

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ESPECIALIZAÇÃO EM ENGENHARIA DE SOFTWARE E INOVAÇÃO

RAFAEL LEITES LUCHESE

**Aplicação de Técnicas de Machine Learning
para Classificação ABC Multicritério**

Monografia de Conclusão de Curso apresentada
como requisito parcial para a obtenção do grau
de Especialista em Engenharia de Software e
Inovação

Orientador: Prof. Dr. Flávio Moreira de Oliveira

Porto Alegre
2021

CIP — CATALOGAÇÃO NA PUBLICAÇÃO

Leites Luchese, Rafael

Aplicação de Técnicas de Machine Learning para Classificação ABC Multicritério / Rafael Leites Luchese. – Porto Alegre: PPGC da UFRGS, 2021.

54 f.: il.

Monografia (especialização) – Universidade Federal do Rio Grande do Sul. Curso de Especialização em Engenharia de Software e Inovação, Porto Alegre, BR-RS, 2021. Orientador: Flávio Moreira de Oliveira.

1. Gestão de Estoques. 2. Classificação ABC. 3. Machine Learning. I. Moreira de Oliveira, Flávio. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões Mendes

Vice-Reitora: Prof^a. Patricia Pranke

Pró-Reitor de Pós-Graduação: Prof. Júlio Otávio Jardim Barcellos

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenadora do Curso: Prof^a. Karin Becker

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“A resposta certa não importa nada:
o essencial é que as perguntas estejam certas.”*

— MÁRIO QUINTANA

AGRADECIMENTOS

Agradeço em primeiro lugar aos meus pais, que sempre me apoiaram e incentivaram nos estudos, comemorando comigo cada conquista. Agradeço também à minha irmã, sempre um exemplo de dedicação e comprometimento a ser seguido. Aos colegas e amigos que, de uma forma ou outra, contribuíram para este trabalho, com ideias, discussões e embasamento técnico, meu muito obrigado. Agradeço ao meu orientador, sempre muito solícito e dando dicas e conselhos de grande valor para este trabalho. Por fim, agradeço à Paula, minha companheira de vida, cujo apoio e carinho incondicionais foram imprescindíveis para a conclusão deste trabalho.

RESUMO

Quando falamos na gestão de estoques de uma empresa, a classificação ABC é uma forma bastante conhecida e amplamente utilizada para priorizar itens de forma a identificar quais são mais e menos relevantes no contexto da empresa. Ao longo do tempo alguns métodos foram propostos para realizar esta classificação, de modo a considerar um ou mais critérios neste análise, sempre com o objetivo de ser mais assertivo no resultado. Dada a ampla gama de aplicações voltadas para classificação que utilizam técnicas de *Machine Learning*, neste trabalho explorou-se a viabilidade da aplicação destas técnicas para o problema da classificação ABC. Utilizando uma base de dados reais, foram aplicadas três técnicas distintas de *Machine Learning* comparando os resultados obtidos pelas técnicas entre si e contra um cenário de referência. A abordagem se mostrou viável, com as técnicas obtendo resultados melhores que o cenário de referência na maior parte dos testes.

Palavras-chave: Gestão de Estoques. Classificação ABC. Machine Learning.

Applying Machine Learning Algorithms for multi-criteria ABC Classification

ABSTRACT

When talking about a company's inventory management, the ABC classification is a well-known and widely applied method to prioritize items in order to identify which ones are more and less relevant in the company's context. Over time, some methods were proposed to carry out this classification, in order to consider one or more criteria in this analysis, always aiming to be more assertive in the result. Given the wide range of Machine Learning Algorithms in classification problems, we explore the feasibility of applying these techniques to the ABC Classification problem. Using a real database, we applied three distinct Machine Learning algorithms comparing their results against each other and against a reference scenario. The approach proved to be viable, with the algorithms obtaining better results than the reference scenario in nearly all tests.

Keywords: Stocks Management, ABC Classification, Machine Learning.

LISTA DE ABREVIATURAS E SIGLAS

ADU	Annual Dolar Usage
ANN	Artificial Neural Networks
CD	Centro de Distribuição
CSL	Customer Service Level
GA	Genetic Algorithms
IA	Inteligência Artificial
KNN	K-nearest Neighbors
ML	Machine Learning
MCIC	Multi-Criteria Inventory Classification
SVM	Support Vector Machines
XGBoost	Extreme Gradient Boost

LISTA DE FIGURAS

Figura 3.1	Visualização gráfica do método ADU, ilustrando o Princípio de Pareto.....	23
Figura 4.1	Visualização temporal da distribuição dos trimestres da base de dados.....	32
Figura 4.2	Acurácia obtida variando o valor de K para o <i>Depósito A</i>	38
Figura 4.3	Acurácia obtida variando o valor de K para o <i>Depósito B</i>	38
Figura 4.4	Acurácia obtida variando o valor de Γ para o <i>Depósito A</i>	41
Figura 4.5	Acurácia obtida variando o valor de Γ para o <i>Depósito B</i>	41
Figura 4.6	Acurácia obtida variando o valor de C para o <i>Depósito A</i>	42
Figura 4.7	Acurácia obtida variando o valor de C para o <i>Depósito B</i>	42
Figura 4.8	Acurácia obtida variando o valor de η para o <i>Depósito A</i>	44
Figura 4.9	Acurácia obtida variando o valor de η para o <i>Depósito B</i>	44
Figura 4.10	Acurácia obtida variando o valor de max_depth para o <i>Depósito A</i>	45
Figura 4.11	Acurácia obtida variando o valor de max_depth para o <i>Depósito B</i>	45
Figura 4.12	Acurácia obtida variando o valor de Γ para o <i>Depósito A</i>	46
Figura 4.13	Acurácia obtida variando o valor de Γ para o <i>Depósito B</i>	46

LISTA DE TABELAS

Tabela 4.1	Tabela dos trimestres contidos na base histórica de pedidos.....	36
Tabela 4.2	Acurácia obtida variando a forma de voto do modelo.....	39
Tabela 4.3	Tabela dos trimestres contidos na base histórica de pedidos.....	40
Tabela 4.4	Resultados obtidos nos trimestres de treino com os métodos aplicados no <i>Depósito A</i>	47
Tabela 4.5	Resumo dos resultados das técnicas no <i>Depósito A</i>	47
Tabela 4.6	Resultados obtidos nos trimestres de treino com os métodos aplicados no <i>Depósito B</i>	48
Tabela 4.7	Resumo dos resultados das técnicas no <i>Depósito B</i>	48
Tabela 4.8	Resumo do custo obtido em cada depósito.....	49
Tabela 4.9	Comparação das técnicas que melhor performaram por trimestre.....	49
Tabela 4.10	Comparação do tempo de execução dos diferentes classificadores, em segundos.....	50

SUMÁRIO

1 INTRODUÇÃO	11
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 Classificação ABC	13
2.2 Política de estoques	14
2.3 Machine Learning	16
2.4 Técnicas de Machine Learning	17
2.4.1 K-nearest neighbors	17
2.4.2 Support vector machine	18
2.4.3 XGBoost	19
2.5 Trabalhos relacionados	20
3 PROPOSTA PARA UTILIZAÇÃO DE MACHINE LEARNING	22
3.1 Método ADU para classificação ABC	22
3.2 Métodos MCIC para classificação ABC	23
3.3 Conjunto de dados utilizado	25
3.4 Tratamento dos dados	27
3.5 Definição de acurácia	28
3.6 Definição de custo	29
4 AVALIAÇÃO EXPERIMENTAL E RESULTADOS	31
4.1 Dados de treino do modelo	31
4.2 Seleção de atributos a serem utilizados	33
4.3 Variação de histórico nos dados de treino	35
4.4 Seleção dos depósitos a serem analisados	36
4.5 Escolha de parâmetros para as técnicas de Machine Learning	37
4.5.1 Parâmetros para a técnica KNN	37
4.5.2 Parâmetros para a técnica SVM	39
4.5.3 Parâmetros para a técnica XGBoost	42
4.6 Resultado da aplicação das técnicas de Machine Learning	46
5 CONCLUSÃO	51
REFERÊNCIAS	53

1 INTRODUÇÃO

A capacidade de gerenciar estoques de maneira eficaz é atualmente uma grande preocupação e tem um papel sem dúvida crucial na gestão da cadeia de suprimentos das empresas de diferentes portes e tamanhos, especialmente para aquelas que possuem uma grande variedade de itens em estoque. Isso acontece porque, com muitos itens, a complexidade de gerenciar este estoque tende a aumentar de forma proporcional. As dificuldades que surgem estão relacionadas com a necessidade de utilizar adequadamente o espaço físico, posicionar os itens adequadamente, gerenciar muitos fornecedores e consumidores deste estoque (GROOVER, 2000).

Com o passar do tempo, muitas empresas necessitam lidar com uma cadeia de suprimentos cada vez mais complexa, com uma diferenciação de produtos que vem constantemente crescendo, buscando atender às necessidades dos consumidores (JOLAI; RAZMI; ROSTAMI, 2011). Um maior número de produtos acabados tende a aumentar também o número de matérias-primas e de fornecedores que a empresa precisa controlar, aumentando assim a complexidade da gestão de seus estoques. Com isso, as empresas estão constantemente buscando formas de balancear o custo dos estoques *versus* seus potenciais benefícios, evidenciando a importância de uma política de estoques adequada (DAS et al., 2012).

Em mercados altamente competitivos, uma gestão adequada e eficaz dos estoques pode inclusive se tornar um diferencial importante de uma empresa, aumentando sua produtividade e sua lucratividade. Em cenários com centenas, às vezes até milhares de itens em estoque, dedicar o mesmo nível de atenção para todos eles é uma tarefa árdua e virtualmente impossível de ser feita, visto que seria necessário um alto investimento em pessoas e tempo, dois recursos geralmente escassos. Desta forma, uma das abordagens mais utilizadas pelas empresas é a criação de uma priorização dos itens em estoque, de modo a identificar quais são os itens mais importantes e indispensáveis (DAS et al., 2012).

Para entender o quão prioritário um item é para o estoque, podemos utilizar uma abordagem de analisar seu histórico de consumo. Por exemplo, uma empresa pode ter um item em estoque que é utilizado com muita frequência, digamos todos os dias. Já outros itens podem ser usados bem eventualmente, apenas uma vez no mês, por exemplo. Alguns itens podem ter um custo bastante alto, enquanto outros podem ser baratos. Essas são algumas das variáveis, mas não as únicas, a se considerar no momento da análise de priorização.

A classificação ABC (TEUNTER; BABAI; SYNTETOS, 2010) é uma forma bastante difundida e largamente utilizada de classificar itens em estoque de uma empresa visando categorizá-los de forma que eles possam ser gerenciados de maneira condizente com a sua relevância no negócio. Inspirada no Princípio de Pareto, também conhecida como regra do 80/20, a classificação ABC tem sido usada nas empresas desde meados do século XX com o objetivo de ajudar a identificar os itens que têm maior impacto no resultado da organização e que por isso são de suma importância, bem como aqueles que são pouco expressivos, e que podem, portanto, ter um controle menos rigoroso.

Desde que começou a ser utilizada pelas empresas, diferentes métodos de classificação ABC têm sido empregados. O primeiro e mais tradicional critério utilizado nas análises é o *Annual Dollar Usage* (ADU), que expressa o quanto de capital da empresa é empregado anualmente naquele determinado item (COHEN; ERNST, 1988). Apesar da sua popularidade e facilidade de uso, com o passar do tempo percebeu-se que, além do capital empregado, haviam outros critérios que poderiam ser úteis na hora de gerar a classificação, como por exemplo: preço, tamanho médio do pedido, frequência com que o item é requisitado, quantidade utilizada, criticidade para o negócio, obsolescência, tempos de ressuprimento, entre outros. Outra abordagem utilizada pelas empresas é a utilização de métodos que consideram múltiplos critérios, conhecidos como *Multi-Criteria Inventory Classification* (MCIC), visando acomodar mais variáveis em sua análise e buscar assim um resultado mais assertivo (FLORES; OLSON; DORAI, 1992).

Mais recentemente, aproveitando o contínuo desenvolvimento e aprimoramento na área de *Artificial Intelligence* (Inteligência Artificial - IA) e valendo-se da grande variedade de aplicações destes conceitos em tarefas de classificação de dados, estudos têm sido propostos utilizando técnicas como redes neurais e algoritmos genéticos na elaboração da classificação ABC (PARTOVI; ANANDARAJAN, 2002). Além disso, trabalhos utilizando métodos de *Machine Learning* (ML), um campo dentro da IA, também tem sido propostos, com base principalmente nos bons resultados obtidos por estes métodos em tarefas de classificação de dados, como em (LAJILI; LADHARI; BABAI, 2016) e (KARTAL; CEBI, 2013).

Neste trabalho realizou-se um estudo de caso sobre a aplicação de técnicas de *Machine Learning* na classificação ABC multicritérios em uma base de dados real de uma indústria brasileira. O principal objetivo foi avaliar a viabilidade da utilização de diferentes técnicas de ML na tarefa de classificar produtos de forma a produzir uma classificação ABC, comparando suas performances entre si e com uma classificação de referência.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção as bases teóricas para o este trabalho são apresentadas, de modo a introduzir os principais conceitos abordados. Nos próximos parágrafos é apresentado em mais detalhes o que é uma classificação ABC e o contexto em que ela se insere dentro de uma empresa. Por fim, é feita uma introdução ao *Machine Learning* e sua utilização, bem como são apresentadas as diferentes técnicas que serão estudadas neste trabalho.

2.1 Classificação ABC

A classificação ABC é uma forma de priorização para identificação e agrupamento de itens de forma a expressar a relevância destes para o negócio da empresa. Ela surgiu da necessidade das empresas em tornarem suas gestões de estoques mais eficientes, com o principal objetivo de permitir às pessoas envolvidas no processo de gestão dos estoques identificarem os itens mais importantes para a empresa.

A classificação ABC divide o universo de itens de uma empresa em três diferentes categorias, sendo elas:

- *Categoria A*: itens extremamente importantes para a empresa, são aqueles para os quais se deseja manter um alto nível de controle e atenção, com uma alta recorrência de revisão dos níveis de estoque, pois o custo de uma ruptura ou de um estoque excessivo pode ser bastante prejudicial para a empresa. Em geral são poucos itens, mas que estão relacionados com um grande impacto no negócio;
- *Categoria C*: itens pouco importantes para a empresa. Em geral são bastante numerosos e são itens que têm um pequeno impacto no resultado do negócio. Por serem muito numerosos, um controle pouco recorrente e pouco criterioso deve ser empregado, pois o esforço de um controle mais rigoroso não vale o investimento em tempo e pessoas;
- *Categoria B*: itens moderadamente importantes para a empresa. São os itens entre as categorias A e C, e refletem exatamente itens cuja importância não chega a ser crucial para a empresa, contudo possuem a sua parcela de relevância. É uma categoria intermediária também na quantidade de itens, possuindo mais itens que a categoria A e menos que a C.

A primeira abordagem para se calcular uma classificação ABC foi inspirada no

Princípio de Pareto, devido à percepção dos estoques da empresa se comportarem conforme sugere o princípio, ou seja, que alguns poucos itens são responsáveis pela maior parte do capital empregado pela empresa, enquanto os demais são responsáveis por uma pequena parcela deste.

Um ponto importante a ressaltar é que, quando falamos em estoques, podemos estar nos referindo tanto a matérias-primas como a produtos acabados, e trabalhar com uma classificação ABC é totalmente viável em ambos os contextos, uma vez que ambos precisam ser gerenciados. O que em geral muda para a empresa nestes dois contextos é que o estoque de matéria-prima é proveniente de fornecedores externos, enquanto o de produto acabado é uma produção interna da empresa.

2.2 Política de estoques

A classificação ABC é na verdade um dos passos de um processo maior, que é a definição de uma política de estoque por parte da empresa, cujo principal objetivo é orientar a política da empresa para o atendimento de sua demanda de curto e médio prazo. A política de estoques de uma empresa é que vai definir não apenas quais produtos desejamos ter em estoque, mas também o quanto de cada produto teremos, quais os níveis mínimos e máximos de cada item, visando evitar rupturas e estoques em excesso, observando também outros fatores como: margem de contribuição, complexidade de produção / compra do item, tempos de ressurgimento, ciclo de vida do produto, entre outros.

É neste contexto maior que geralmente é utilizada uma classificação ABC para estoques. Não há uma receita de bolo que defina exatamente como realizar uma política de estoques, nem que de fato fazer uma classificação ABC seja obrigatório; a realidade de cada empresa é diferente e em cada cenário pode fazer ou não sentido aplicar uma ou outra análise. No entanto, em cenários com um número grande de itens para gerenciar e com recursos de tempo e pessoas escassos, o instrumento mais difundido de priorizar os itens a receberem maior atenção é a classificação ABC.

Basicamente a classificação ABC inserida no contexto de uma política de estoques vai buscar ajudar a identificar os itens que são mais ou menos relevantes para a empresa, conforme os critérios que a ela própria definir, visando garantir que seja feito um controle do estoque de cada item de acordo com esta relevância. A política de estoque vai ser responsável por dar todo o direcionamento do estoque da empresa, envolvendo algumas definições como:

- *Estoque mínimo*: quantidade mínima que se deseja manter de um determinado item, com o intuito de evitar rupturas. Está bastante relacionado com a quantidade média consumida do item, com a frequência de utilização, com os tempos de ressuprimento, entre outros;
- *Estoque máximo*: quantidade considerada limite de ter de um determinado item, sendo o excedente um custo indesejado para a empresa. Assim como o estoque mínimo, também está relacionado com o consumo do material, sua frequência, quantidades e sua reposição;
- *Estoque de segurança*: é o estoque que visa absorver a demanda em caso de imprevistos, tanto relacionados com a demanda do item como no tempo de ressuprimento;
- *Periodicidade de controle*: frequência com que os itens devem ser controlados, podendo variar com sua classificação ABC, o número de fornecedores daquele item, o tempo de ressuprimento destes fornecedores;
- *Gestão de pessoas*: direcionamento das ações dos colaboradores da empresa para que o processo ocorra da forma esperada. Divisão das pessoas por setores de responsabilidade e entre as principais atividades, como contagem, recontagem, compra, descartes e o acompanhamento do processo.

Estes são apenas alguns termos comuns que uma política de estoques visa definir, sempre com o objetivo maior de buscar um equilíbrio nos níveis de estoque da empresa. Estoque em demasia pode ser prejudicial, pois representa capital imobilizado, além de possíveis perdas por extravio, perda de qualidade, entre outros. Por outro lado, manter baixos níveis de estoque pode fazer a empresa perder vendas e eficiência produtiva, através de rupturas de estoque de produtos acabados e de matéria-prima, respectivamente.

É preciso também que seja avaliada uma periodicidade para se revisar a política de estoques, que vai depender do cenário no qual a empresa está inserida. Sem dúvida, fatores como sazonalidade, o ciclo de vida dos produtos no portfólio da empresa, mudanças e tendências da demanda do mercado, entre outros, afetam na definição desta periodicidade. Ou seja, é preciso estar sempre em linha com o consumo dos estoques, que invariavelmente muda com o tempo, de modo a estar preparado para atender a demanda.

2.3 Machine Learning

A aprendizagem de máquina (MITCHELL et al., 1997), do termo em inglês *Machine Learning*, é um campo da Inteligência Artificial que estuda algoritmos que buscam, através dos dados, encontrar padrões, visando fazer previsões ou determinações. Esse campo de estudo não é novo, porém tem se desenvolvido mais rapidamente recentemente, principalmente devido ao aumento de poder computacional e de armazenagem de dados visto nos últimos 20 anos, o que possibilitou a ampla utilização destes conceitos em ambientes empresariais.

Uma importante divisão entre as diferentes técnicas de *Machine Learning* são as distintas abordagens de aprendizado possíveis, sendo elas: supervisionado, não supervisionado e aprendizado por reforço. Abaixo é feita uma breve descrição das características de cada abordagem:

- *Aprendizado supervisionado*: técnicas de aprendizado supervisionado são aquelas que são treinadas com um conjunto de dados de teste, ou seja, são fornecidas entradas de exemplo e suas saídas esperadas;
- *Aprendizado não supervisionado*: nas técnicas de aprendizado não supervisionado são fornecidos dados de entrada para o modelo, porém não são fornecidas as saídas esperadas, fazendo com que o modelo busque encontrar padrões nos dados fornecidos;
- *Aprendizado por reforço*: na abordagem de aprendizado por reforço não são fornecidos dados de entrada ao modelo, que por sua vez aprende através da experimentação, no ambiente em que está inserido, visando atingir um objetivo definido, de forma iterativa.

Técnicas de *Machine Learning* têm sido usadas com bastante sucesso em diversas áreas do conhecimento humano. Abaixo são apresentados alguns exemplos que demonstram a variedade de aplicações em que tem se utilizado ML:

- *Análise de crédito*: empresas do setor financeiro utilizam dados dos tomadores de crédito para avaliar o risco do empréstimo;
- *Detecção de fraudes*: bancos utilizam os dados de compras efetuadas em cartões de crédito para detectar fraudes, através da comparação da compra atual com o histórico de compras do cliente;
- *Detecção de spams*: provedores de email utilizam técnicas de *Machine Learning*

para prever se um email é potencialmente um spam ou não;

- *Medicina*: técnicas de *Machine Learning* têm sido amplamente utilizadas em uma variedade de aplicações médicas, desde a classificação de imagens (no diagnóstico de tumores, por exemplo), apoio no diagnóstico e escolha de tratamento de doenças;
- *Sistemas de recomendação*: *e-commerces* e plataformas de *streaming* utilizam o histórico de navegação dos usuários para melhorar as recomendações de novos produtos.

A principal motivação para a utilização de técnicas de *Machine Learning* na classificação ABC é o grande número de aplicações focadas na classificação e clusterização existentes com ML. Juntando isso à complexidade de se trabalhar com muitas variáveis em métodos de classificação ABC multicritérios, é proposto que essa abordagem possa ser uma alternativa a esta alta complexidade e trazer bons resultados (MOHAMMADI-TABAR; GHODSYPOUR; O'BRIEN, 2012).

2.4 Técnicas de Machine Learning

Existe uma grande gama de técnicas de *Machine Learning* que podem ser utilizadas no contexto do trabalho, no entanto, o trabalho limitou-se à utilização de algumas técnicas julgadas mais interessantes de serem utilizadas no problema em questão, baseado principalmente em outros contextos de utilização das técnicas e dos trabalhos relacionados com o presente estudo.

Optou-se por utilizar apenas técnicas de aprendizagem supervisionada para o nosso trabalho, especialmente devido a sua boa adaptação a problemas de classificação de dados e à natureza do problema.

Após estudar alguns métodos como alternativa, os métodos selecionados para serem utilizados neste estudo foram: K-Nearest Neighbors (KNN), Support Vector Machines (SVM) e XGBoost. Nos próximos parágrafos é apresentado um pouco sobre cada método e o motivo de sua escolha.

2.4.1 K-nearest neighbors

O KNN (PETERSON, 2009), acrônimo para K-Nearest Neighbors, é uma técnica de *Machine Learning* supervisionada muito utilizada para classificação e regressão de

dados, na qual o algoritmo busca classificar uma nova entrada de dados com base nas K entradas mais similares do conjunto de entrada conhecido, sendo K um inteiro positivo. O método é baseado no princípio de que instâncias de uma mesma classe vão possuir atributos similares, com isso a classificação de uma nova instância a partir dos dados de teste é dada pela proximidade desta instância com as demais.

Caso utilizado para a regressão de dados, a saída do método é um valor médio dos K vizinhos mais próximos. Já quando utilizado como um classificador, o interesse deste estudo, a saída do método é dada pela classe ou grupo que obtém a maioria dos votos dos vizinhos mais próximos. Este voto pode ser em geral de duas formas: uniforme ou por distância. Se uniforme, cada vizinho vota a classe a ser atribuída à nova instância e a classe mais votada será o resultado. Se por distância, os vizinhos mais próximos (de menor distância) tem uma influência maior na votação.

Um dos parâmetros mais importantes do método é o próprio número de K a ser escolhido. Um número de K muito pequeno pode não produzir um bom resultado, pois o resultado será baseado em poucas amostras dos dados em questão. Já um valor muito alto de K pode não ser muito eficaz em identificar as fronteiras entre as diferentes classes. Por isso não há uma fórmula mágica para se decidir o valor de K , e ele está ligado também aos dados que utilizamos. Para definir tal valor, em geral varia-se o valor de K com o mesmo conjunto de dados e avalia-se a acurácia de cada valor, permitindo assim identificar o valor que produziu o melhor resultado.

Para classificar uma nova instância, o KNN se baseia no cálculo da distância entre as observações de treino e esta nova observação, e para o problema proposto será utilizada a distância Euclidiana no modelo. Por isso, é de suma importância para a obtenção de um bom resultado que os dados sejam preparados corretamente, ou seja, normalizados, antes de serem utilizados no algoritmo. Mais adiante será abordado o processo de normalização. A escolha pelo KNN se deu, portanto, especialmente pela sua grande aplicação em tarefas de categorização.

2.4.2 Support vector machine

O SVM (LORENA; CARVALHO, 2007), abreviação para Support Vector Machines, é uma técnica de *Machine Learning* supervisionada bastante difundida e utilizada em uma grande gama de problemas, especialmente por ser bastante genérica. Assim como a técnica KNN, esta técnica também é bastante empregada na classificação e em análises

de regressão. Ela é um classificador linear binário não probabilístico, o que significa que ela deve ser utilizada com apenas duas saídas possíveis.

A ideia por trás do SVM é dispor os dados de entrada como pontos no espaço, e o método busca então encontrar o hiperplano que maximize a distância entre as duas classes, de modo que uma nova instância possa ser classificada com base na classe mais próxima.

Por ser um classificador binário, é preciso adaptá-lo para ser usado no problema proposto, no qual a saída esperada é constituída de três classes distintas. Duas estratégias são possíveis para tratar este problema, sendo elas:

- *One-vs-one*: é uma estratégia onde cada classe é testada contra outra. No contexto deste trabalho, são gerados 3 classificadores, sendo eles: A contra B, A contra C e B contra C. Quando uma nova entrada é testada, cada um dos classificadores definirá uma classe para essa nova entrada, e a classe majoritária será a escolhida;
- *One-vs-all*: é uma estratégia onde cada classe é testada contra todas as outras. Assim, no contexto deste trabalho, é necessário gerar 3 classificadores binários de forma a classificar os nossos dados, pois é preciso testar a classe A contra o resto, depois a B e finalmente a classe C. Desta forma o classificador consegue decidir a qual classe uma nova observação pertence, buscando entre os 3 classificadores aquele que apresentou a maior similaridade com o modelo.

2.4.3 XGBoost

O XGBoost (CHEN; GUESTRIN, 2016), forma diminuta de *Extreme Gradient Boost*, é uma técnica de *Machine Learning* relativamente recente, e que tem ganhado bastante popularidade nos últimos anos devido principalmente às melhorias de performance aplicadas à técnica. O algoritmo tem sido bastante utilizado em problemas de previsão envolvendo dados estruturados, mostrando um excelente resultado em competições do *Kaggle* (uma das principais comunidades de cientistas de dados do mundo).

A técnica é uma implementação de árvores de decisão com *gradient boost*, e esse *gradient boost* é utilizado principalmente para melhorar a acurácia do método. Árvores de decisão é uma técnica de *Machine Learning* também bastante utilizada na classificação e regressão de dados, onde é construída uma estrutura de árvore, na qual partindo da raiz da árvore e percorrendo seus nodos, que são os pontos de decisão do algoritmo (as perguntas

da árvore), até suas folhas, obtemos o valor de saída.

Já o *gradient boost* do método é uma técnica utilizada de forma a combinar a classificação de um conjunto de árvores de decisão de forma a produzir um consenso. No entanto, diferentemente da técnica de *Random Forests*, que também é um algoritmo que busca o consenso de diferentes árvores de decisão, onde as diferentes árvores são independentes entre si e o consenso por maioria é atingido ao final da análise, o XGBoost constrói cada árvore de forma iterativa, uma após a outra, buscando sempre melhorar a árvore seguinte a partir da anterior e combinando os resultados das árvores ao longo do processo de iteração.

2.5 Trabalhos relacionados

O uso de Inteligência Artificial no contexto da classificação ABC já vem sendo estudado há alguns anos e alguns trabalhos já foram propostos. Em (PARTOVI; ANANDARAJAN, 2002) é apresentado um estudo utilizando *Artificial Neural Networks* (ANN) para uma empresa farmacêutica. No estudo, a rede neural é treinada tanto com *Genetic Algorithms* (GA) como usando *BackPropagation*, e o principal objetivo é simular a tomada de decisão de um gestor de suprimentos. Na camada de entrada de dados são fornecidas diferentes informações, como a demanda do item, o tempo de ressuprimento, o custo, entre outros, e a rede produz a saída informando a classe do item.

Em (KARTAL; CEBI, 2013) é feito um estudo sobre o uso da técnica SVM para a classificação ABC em um conjunto de dados de uma empresa real, a partir de alguns critérios pré-definidos, utilizando pesos para os critérios fornecidos pela própria empresa, atingindo resultados promissores.

No estudo apresentado em (YU, 2011) é feito um comparativo entre diferentes técnicas de Inteligência Artificial (redes neurais) e *Machine Learning* com métodos MCIC, utilizando como referência um conjunto de dados bastante popular de 47 itens, encontrado na literatura. Apesar de trabalhar com uma amostra pequena de dados, o estudo mostra que a performance das técnicas aplicadas é superior àquelas das classificações MCIC, fazendo delas grandes candidatas a serem testadas em cenários reais.

Finalmente, em (LAJILI; LADHARI; BABAI, 2016) é apresentado um estudo usando cinco diferentes técnicas de *Machine Learning*, sendo elas: Decision Trees, Naive Bayesian Networks, KNN, SVM e ANN. Diferentemente de outros estudos, este propõe diferentes abordagens de comparação entre os resultados dos métodos, um dos quais en-

volvendo o custo total do estoque da classificação produzida, conforme apresentado em (TEUNTER; BABAI; SYNTETOS, 2010).

3 PROPOSTA PARA UTILIZAÇÃO DE MACHINE LEARNING

Nesta seção será apresentada a forma de cálculo do método ADU para classificação ABC, que será utilizado como cenário de referência na sequência deste trabalho. Além disso, como será abordado adiante, o método ADU também servirá para a preparação dos dados de treino dos modelos de *Machine Learning*. Além disso, será discutido o conjunto de dados que será utilizado e os tratamentos aplicados de forma a prepará-lo para nossa análise. Por fim, serão apresentadas as métricas que serão utilizadas para a comparação entre diferentes classificações.

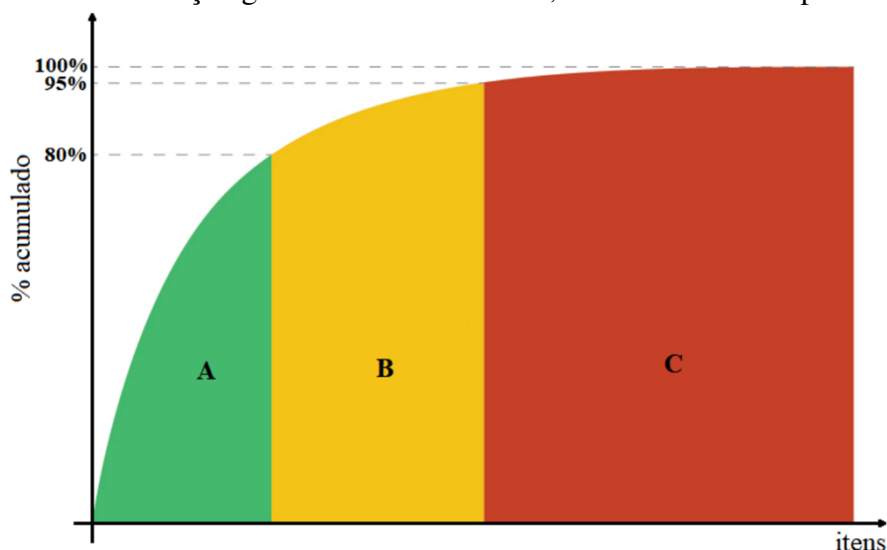
3.1 Método ADU para classificação ABC

O primeiro método que surgiu para a classificação ABC foi o método Annual Dollar Usage (ADU). Observando o comportamento dos itens no estoque, percebeu-se que havia uma distribuição análoga ao Princípio de Pareto (princípio do 80/20), ou seja, que alguns poucos itens (20% dos itens) seriam os responsáveis pelo maior valor de estoque (80% do valor), enquanto o restante dos itens (80% dos itens) seriam responsáveis por uma pequena parcela do capital empregado (20% do valor). O método ADU visa, portanto, classificar os itens com base no capital empregado pela empresa em estocar aquele item, permitindo desta forma ordená-los do de maior valor para o de menor valor.

O método é bastante simples de ser calculado: basta pegar o valor de cada item e multiplicar pela quantidade utilizada ao longo de um determinado período. Como o próprio nome do método sugere, este período pode ser anual, mas basta que seja usado o mesmo período para todos os itens e a análise se torna válida. Com isso, é obtida uma lista dos itens, ordenada de forma decrescente em termos de valor total de capital empregado pela empresa, ficando no topo os itens de maior valor empregado.

Em seguida, esta lista é quebrada em três partes, com base no total acumulado. Não existe uma porcentagem fixa ou ideal para cada categoria, porém a convenção é de que os itens A são os que representam entre 70% e 80% do total de capital empregado pela empresa, os itens B representam entre 15% e 20% do valor empregado, e por fim os itens C representam algo entre 5% e 10% do valor total. As categorias A, B e C com isso estão ligadas ao nível de importância que deve ser empregado para cada item. A figura 3.1 a seguir mostra de maneira visual como é feita a distribuição dos itens entre as categorias A, B e C.

Figura 3.1: Visualização gráfica do método ADU, ilustrando o Princípio de Pareto.



Com o objetivo de não deixar tendenciar nenhum extremo, para este trabalho optou-se por utilizar valores médios de cada classe, e os valores adotados foram: 74% para a classe A, 18% para a classe B e 8% para a classe C.

Este método ainda é bastante utilizado devido especialmente a sua facilidade de uso. Diversos sistemas empresariais permitem automaticamente obter essa classificação, através do histórico de consumo dos itens no estoque, ou através do histórico de pedidos de venda no caso de produtos acabados. Ele será utilizado neste trabalho como um modelo de referência para comparação de resultados com os demais métodos empregados.

Com o passar do tempo, e com o aumento da maturidade dos processos de gestão de estoques, o método passou a ser criticado por focar exclusivamente no fator financeiro do valor do item em estoque, ou seja, fazendo uma análise de uma única dimensão, e conseqüentemente negligenciando outras variáveis que pudessem influenciar na importância do item em estoque. Visando suprir a necessidade de acomodar outras variáveis na análise, surgiram os métodos chamados de multicritérios.

3.2 Métodos MCIC para classificação ABC

Os métodos multicritérios surgiram com o intuito de englobar outras variáveis no processo de classificação ABC. Conforme o ambiente de cada empresa, outras características podem ser de suma importância na hora de gerar a classificação. Por exemplo, em indústrias cujos estoques contêm itens perecíveis, a durabilidade do item pode ser um fator muito importante a se considerar. Neste caso, por exemplo, itens com pouca du-

rabilidade precisam ter seu estoque controlado de forma frequente, porém utilizando-se apenas o critério de capital empregado, esses itens poderiam ser definidos como itens de baixa frequência de controle (itens C).

Este é apenas um exemplo de um critério que pode, em determinados cenários, impactar no resultado de uma classificação mais adequada para o ambiente de cada empresa. Contudo, outros critérios que também podem impactar são citados na literatura, como por exemplo em (CHEN; LI; LIU, 2008) e em (RAMANATHAN, 2006). A seguir, são listados alguns exemplos:

- *Tempo de ressuprimento*: as empresas possuem muitas vezes uma gama bastante ampla de itens em estoque e o ressuprimento destes itens pode ser bem variado, sendo de poucos dias ou de até alguns meses;
- *Frequência de utilização do item*: alguns itens podem ser requisitados em pequenas quantidades e forma constante, enquanto outros em grande quantidade mas muito raramente, o que é negligenciado por uma análise puramente de capital empregado como a ADU;
- *Pedido médio (ou consumo médio, no caso de matérias-primas)*: quantidade média em que o item é requisitado, similar à frequência de utilização;
- *Ciclo de vida do produto*: principalmente as fases de lançamento e descontinuidade do produto, onde o comportamento das vendas é bastante incerto ou inflado / desinflado propositalmente pela empresa.

Frente a este desafio foram propostos diversos métodos multicritérios, denominados *Multi-Criteria Inventory Classification (MCIC)*. Abaixo algumas abordagens que são encontradas na literatura na utilização de métodos multicritérios:

- *Métodos baseados em AHP (Analytical Hierarchy Process)*: são métodos que visam auxiliar na tomada de decisões complexas e são amplamente utilizados em uma série de aplicações, sendo baseados em um modelo de identificação e definição de pesos para os atributos significativos. No contexto da classificação ABC foi utilizado em (PARTOVI; BURTON, 1993). O método é às vezes criticado por incluir uma subjetividade na definição dos pesos e atributos relevantes;
- *Métodos baseados em otimização linear ponderada*: são métodos que buscam encontrar os valores que minimizam uma função linear satisfazendo um determinado conjunto de restrições. Um exemplo de aplicação no contexto do deste trabalho é apresentado em (RAMANATHAN, 2006);

- *Outros métodos*: outros métodos específicos também foram propostos, como por exemplo em (FLORES; WHYBARK, 1987) e (COHEN; ERNST, 1988), usando eventualmente alguns conceitos presentes nos métodos acima mencionados, porém com algumas características peculiares, mas que em geral são métodos que buscam uma melhor forma de ponderar os diferentes critérios para atingir melhores resultados. O método *TOPSIS*, proposto em (BHATTACHARYA; SARKAR; MUKHERJEE, 2007) e métodos utilizando *Data Envelopment Analysis* (DEA), como proposto em (TORABI; HATEFI; PAY, 2012), são exemplos destes métodos.

Assim como o método ADU, estes métodos são bastante utilizados, todavia são menos difundidos especialmente devido a sua maior complexidade e por serem também mais custosos de serem aplicados, seja em termos de custo computacional ou mesmo uma complexidade crescente com o número de itens em estoque, dificultando seu uso em alguns cenários. De forma geral, nestes métodos, a adição de mais critérios aumenta também a complexidade de cálculo, o que é muitas vezes mais um complicador para sua utilização.

3.3 Conjunto de dados utilizado

O conjunto de dados utilizado como objeto de estudo para este trabalho consiste em uma base de dados real de uma empresa do ramo de cerâmicas que atua em todo o território brasileiro. Para atender o seu cliente final, que pode ser tanto uma pessoa física quanto jurídica, em suas lojas espalhadas por todo o Brasil, a empresa dispõe de diferentes centros de distribuição de produtos em diferentes localidades e regiões do país, cada qual atendendo lojas específicas. O fluxo da empresa para atendimento dos clientes finais é descrito a seguir:

1. O cliente final realiza o pedido na loja física;
2. O pedido do cliente é repassado ao centro de distribuição (CD), que é responsável por abastecer a loja física;
3. O CD é atendido pela fábrica da empresa, responsável pela produção dos itens.

Desta forma a gestão de estoque de cada um dos CDs é de suma importância para a empresa, pois está diretamente atrelada ao nível de serviço prometido ao cliente final. Ter uma boa composição dos produtos no estoque de cada CD impacta positivamente o resultado da empresa. Apesar da empresa atuar no Brasil inteiro, os diferentes CDs aten-

dem regiões geográficas diferentes, e estão, desta forma, expostos a mercados diferentes, com diferentes sazonalidades e interesses. Por isso, cada CD conta com sua própria política de estoques e, conseqüentemente, sua própria classificação ABC, visando adequar-se à demanda de sua região.

Este estudo foca especificamente no ponto 2 do fluxo de atendimento mencionado anteriormente, ou seja, a disponibilidade de estoque nos CDs, visando atender as lojas físicas (que por consequência atendem o cliente final). O cenário a ser estudado aqui é, portanto, a classificação ABC aplicada a produtos acabados, a serem armazenados nos diferentes CDs da empresa.

Conforme mencionado anteriormente, o processo de classificação ABC é compreendido por um processo maior, que é a definição de uma política de estoques da empresa, para cada um dos CDs. A empresa revisa sua política de estoque com uma periodicidade trimestral, e dentro desta revisão o primeiro passo é entender, para cada CD, qual a classificação ideal dos produtos para o novo ciclo que se inicia. Esse tempo de três meses está ligado à fatores como:

- Sazonalidade na demanda conforme as diferentes estações do ano;
- Renovação do portfólio da empresa com adição e remoção de produtos;
- Para garantir uma certa estabilidade no processo.

Seria viável trabalhar com um horizonte menor que três meses, buscando ser mais reativo às variações de demanda de curto prazo, porém a empresa mantém este horizonte para garantir uma estabilidade no processo, uma vez que envolve diferentes pessoas e áreas, e na visão da empresa mudanças mais frequentes seriam prejudiciais ao processo como um todo.

O conjunto de dados é uma base histórica da entrada dos pedidos em cada um destes CDs. Esta base contém pedidos desde o primeiro trimestre do ano de 2017 até o primeiro trimestre de 2021. Há dados de seis diferentes CDs, espalhados nas regiões Sul (um), Sudeste (dois), Nordeste (dois) e Centro-Oeste (um) do Brasil. O total de produtos distintos na base de dados é de cerca de 4600, e há em média, em um único CD em um trimestre cerca de 1500 itens.

3.4 Tratamento dos dados

Um ponto bastante importante na utilização das técnicas de *Machine Learning* é com relação ao tratamento dos dados de entrada dos métodos, especialmente para lidarmos com dados não numéricos, dados nulos e também com a normalização dos dados.

Para podermos levar em conta parâmetros textuais ou não numéricos nos algoritmos de *Machine Learning*, precisamos realizar uma técnica chamada de *One Hot Encoding*, que transforma dados categóricos em dados numéricos, igualmente espaçados entre si. A técnica consiste em transformar as diferentes categorias em vetores binários para cada valor, eliminando problemas de hierarquia ou ordenação que uma simples transformação do texto em um número inteiro traria.

Esta é uma tática boa para lidar com dados textuais, porém ela pode aumentar bastante a complexidade do cálculo, uma vez que vetorizando o dado é como se adicionássemos uma nova coluna para cada valor distinto de categoria. Portanto, é necessário avaliar a viabilidade de usar essa técnica em campos com um grande número de valores distintos.

Para os dados nulos, é preciso também tratá-los de forma adequada. Imagine que temos um dado cujos valores variam entre 0 e 100, e temos alguns valores nulos. Se simplesmente substituíssemos os nulos por 0, por exemplo, estaríamos prejudicando a análise, pois o 0 é de fato um valor válido e não é o valor daquela instância. Com isso podemos tratar os nulos de duas formas, conforme fizer mais sentido:

- *Eliminar as amostras*: se o número de amostras com valores nulos for desprezível, pode-se eliminar estas amostras, para não influenciarem o resultado, sem perda significativa de qualidade;
- *Substituir os nulos pela média / moda*: neste tratamento visa-se minimizar os impactos dos valores nulos, substituindo-os para o valor da média ou da moda das demais amostras que possuem o dado.

Já com dados numéricos também é necessário tratá-los antes de utilizá-los nas técnicas de *Machine Learning*. Isso ocorre uma vez que os diferentes atributos utilizados no método podem ter escalas bem diferentes entre si. Por exemplo, uma variável que representa uma porcentagem pode variar entre 0 e 1 enquanto uma outra variável que represente um valor de temperatura pode variar entre 30 graus negativos e 50 graus positivos, ou seja, as escalas são completamente distintas. Deste modo, é bastante comum

normalizarmos estes dados, trazendo todos para a mesma escala, evitando assim esses problemas.

Neste trabalho foram utilizados todos os tratamentos de dados aqui descritos, conforme explicado abaixo:

- *One Hot Enconding*: dados de categoria dos produtos foram utilizados, como por exemplo a unidade de medida do produto além da família de produto, conforme exposto adiante;
- *Eliminação de nulos*: alguns atributos possuíam alguns poucos valores nulos, e nestes casos essas amostras foram eliminadas do conjunto de dados, foi o caso da família do produto;
- *Substituição pela média / moda*: em algumas poucas amostras não havia a informação sobre o cliente ser pessoa física ou jurídica, e nestes casos substituiu-se pela moda da amostragem. O percentual de nulos na base foi inferior a 1% dos dados, ou seja, entendemos que nenhum ruído foi produzido devido a este tratamento;
- *Normalização*: a normalização foi aplicada a todos os campos numéricos.

Por fim, para realizar o trabalho, a base de dados foi anonimizada, eliminando quaisquer informações sensíveis para manter o sigilo dos dados da empresa.

3.5 Definição de acurácia

Para avaliar os resultados dos diferentes métodos e compará-los entre si será utilizada a noção de acurácia comumente utilizada em *Machine Learning* em cenários de classificação e categorização, ou seja, ela é o percentual de predições corretas feitas pelo método. A fórmula abaixo resume o cálculo:

$$\text{Acurácia} = \frac{\text{Número de predições corretas}}{\text{Número total de predições}}$$

De maneira a exemplificar, em uma aplicação do método para classificar 1000 itens, se o método classificar 800 itens da forma correta e 200 itens de forma errônea, a acurácia do método é de 80%.

3.6 Definição de custo

Uma outra forma de avaliar a performance das técnicas de *Machine Learning* específica do problema aqui abordado foi encontrada na literatura. Em (TEUNTER; BABAI; SYNTETOS, 2010) foi proposta uma forma de avaliar duas classificações ABC diferentes em termos do custo total de armazenamento obtido com a classificação.

Desta forma, neste trabalho foi utilizado também como fator de comparação do resultado dos diferentes métodos a metodologia proposta em (LAJILI; LADHARI; BABAI, 2016). O objetivo é obter o custo total de armazenamento de um conjunto de itens quando um nível fixo de serviço (*Customer Service Level* - CSL) é definido para cada uma das classes A, B e C. Para essa definição, dada a natureza do problema, podemos assumir fatores de segurança decrescentes (como é de se esperar, visto que os itens A são os mais importantes e os itens C os menos importantes), e neste estudo serão assumidos os valores de 99%, 95% e 90%, respectivamente, como nível de serviço de cada classe. Para este cálculo foi assumido também que a demanda é regida por uma distribuição normal.

O custo unitário de um item i pode ser calculado como:

$$C_i = h_i \times k_i \times \theta_i$$

Onde:

- h_i : é atribuído a 20% do valor do custo do item i ;
- θ_i : é o desvio padrão da demanda do item i ;
- k_i : é o fator de segurança do item i e pode ser calculado a partir da fórmula a seguir.

Para o fator de segurança k_i , seu cálculo é dados por:

$$k_i = \Phi^{-1} \times (CSL_i)$$

Em que:

- Φ^{-1} : é uma função de distribuição normal de probabilidade;
- (CSL_i) : é o nível de serviço da classe do item i , conforme definimos anteriormente.

E portanto, o custo total de uma classificação ABC é expressa pela soma dos custos individuais dos N itens nela presentes, ou seja, pela fórmula a seguir:

$$C_T = \sum_{i=1}^N h_i \times k_i \times \theta_i$$

Com base nesta metodologia é possível, para uma determinada classificação ABC, calcular o custo de cada item nesta classificação, e a soma de todos os itens nos dá o valor total de armazenamento. Desta forma, podemos comparar duas classificações distintas, e uma classificação é melhor que a outra se possuir um custo total de armazenamento menor.

4 AVALIAÇÃO EXPERIMENTAL E RESULTADOS

O próximo passo do trabalho consiste em preparar o conjunto de dados que temos em mãos de modo a extrair os atributos a serem utilizados na aplicação das técnicas de *Machine Learning*, de modo a realizar a classificação ABC com elas. Nesta etapa também será feito um trabalho de refinamento dos parâmetros a serem utilizados pelos modelos, de modo a obter uma melhor performance. Por fim, a comparação entre as classificações com o método ADU, cenário de referência, com as classificações geradas através das técnicas KNN, SVM e XGBoost serão comparadas em termos de acurácia e de custo.

Manteve-se o período de análise para a classificação ABC igual ao utilizado pela empresa, com isso, é possível obter um comportamento bem próximo de como seria a aplicação real dos métodos na classificação. Valendo-se do fato de possuir uma base histórica da entrada dos pedidos desde 2017 para os CDs, conseguimos com este conjunto de dados emular um comportamento trimestre a trimestre, ou seja, para cada trimestre que se evolui no tempo, é possível utilizar os trimestres anteriores como dado de treino dos modelos de *Machine Learning*.

A base de dados utilizada consiste em um banco de dados SQL Server e a linguagem de programação utilizada para a aplicação das técnicas de ML foi a linguagem Python, através das bibliotecas *scikit-learn* e *matplotlib*.

4.1 Dados de treino do modelo

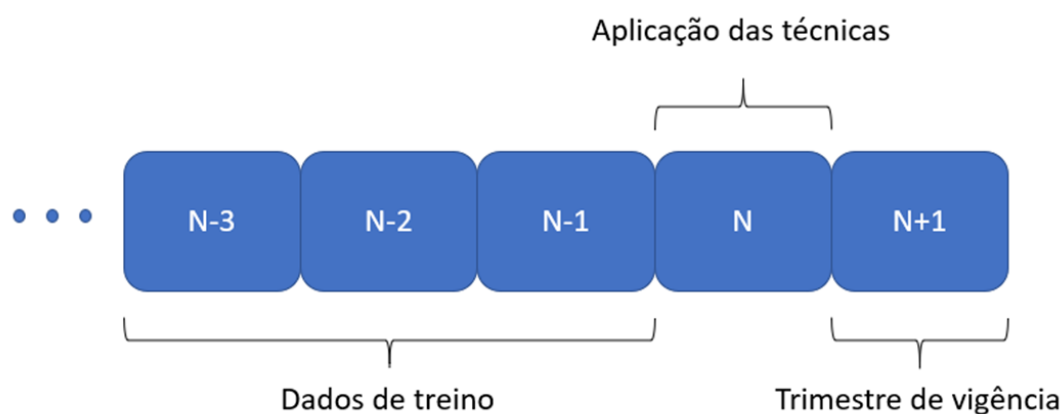
Como são utilizadas técnicas de *Machine Learning* supervisionadas, para treinar os modelos é necessário fornecer os dados de treino com o resultado esperado. Ou seja, é preciso fornecer a informação de qual é a classificação ABC correta para um determinado item. Contudo, não existe uma forma de determinar qual a classificação ABC correta ou ideal para um determinado conjunto de itens. Cada método que se utilizar pode chegar a resultados diferentes.

No entanto, como a base de dados possui o histórico de pedidos a serem atendidos pelos diferentes CDs, é possível analisar estes dados de uma forma temporal, e buscar definir uma classificação ABC de referência com base no que de fato ocorreu no trimestre seguinte à classificação ABC.

O conjunto de dados é composto de um histórico de 17 trimestres, que foram ordenados de 1 a 17, do mais antigo para o mais atual, desde o ano de 2017 até o primeiro

trimestre de 2021. Com isso, é possível realizar uma classificação que estaria vigente para o trimestre $N+1$ usando como dados de treino os trimestres $N-1$ e anteriores, e os dados do trimestre N como os dados a serem aplicados às técnicas de *Machine Learning*, ou seja, os dados de teste. De posse dos dados históricos, é possível verificar a acurácia da classificação obtida com base no realizado, ou seja, o que de fato ocorreu no trimestre $N+1$. Em resumo, o modelo é treinado com os dados de histórico do trimestre $N-1$ e anteriores, em seguida o modelo é aplicado aos dados do trimestre N e por fim é obtida a classificação do trimestre $N+1$, conforme a figura 4.1. Esse comportamento é exemplificado nos próximos parágrafos.

Figura 4.1: Visualização temporal da distribuição dos trimestres da base de dados.



Com essa base histórica consegue-se emular o comportamento da aplicação real das técnicas. Por exemplo, se estivéssemos no final do terceiro trimestre de 2018, buscando gerar a classificação ABC para o quarto e último trimestre de 2018, com base na figura acima, estaríamos no momento em que N é igual a 7 e pode-se obter sua classificação ABC da seguinte forma:

- Pegando-se os dados de pedidos do trimestre 7, podemos realizar uma classificação ABC utilizando o método ADU. A esta classificação é atribuída o nome de ADU_i , onde i representa o trimestre para o qual está sendo gerada a classificação. Neste exemplo temos a ADU_8 , que é a classificação ADU para o trimestre 8, gerada com base nos dados do trimestre 7;
- Pegando-se os dados de pedidos que de fato ocorreram no trimestre 8, é possível também realizar uma classificação ABC utilizando o método ADU. A essa classificação é atribuído o nome de $REAL_i$, onde i representa o trimestre no qual os pedidos ocorreram. Neste exemplo, i é igual 8. O nome *REAL* faz alusão ao realizado, e esta classificação será a classificação correta, a ser fornecida aos dados de

treino do modelo;

- Os dados de pedidos do trimestre 6 (e anteriores) podem então ser fornecidos como dados de treino para as técnicas de *Machine Learning*, passando a classificação $REAL_6$ (e equivalentes para os anteriores) como o valor correto da classificação;
- As classificações geradas por cada um dos métodos serão, portanto, chamadas de KNN_i , SVM_i , XGB_i , onde i representa o trimestre da classificação, respectivamente para as técnicas KNN, SVM e XGBoost. Neste exemplo, i é 8;
- As classificações KNN_8 , SVM_8 , XGB_8 podem então ser confrontadas com a classificação ADU_8 com base nos critérios definidos de acurácia e custo.

Devido a esse comportamento temporal, o primeiro trimestre para o qual é possível realizar a análise é com N igual a 2 na figura 4.1, ou seja, seriam geradas as classificações ADU_3 , KNN_3 , SVM_3 , XGB_3 e $REAL_3$, pois é necessário ao menos um trimestre para treinar os modelos. De forma análoga, o último trimestre no qual é possível fazer nossa análise é o de N igual a 16, pois é o último em que é possível realizar a classificação $REAL_{17}$ para poder analisar os resultados obtidos.

4.2 Seleção de atributos a serem utilizados

A base de dados utilizada possui uma grande quantidade de atributos disponíveis para serem utilizados na experimentação. A seguir é apresentada a lista de todos os atributos possíveis de serem utilizados, onde entre parênteses é apresentado o tipo de dado do atributo, juntamente com uma breve descrição qualitativa da informação que cada um contém. Muitos destes atributos foram calculados a partir da base de dados, porém alguns são de fato dados crus provenientes da própria base.

- *Quantidade de pedidos (numérico)*: contador do total de pedidos distintos do item no trimestre em questão;
- *Valor total (numérico)*: valor total do item no trimestre, obtido multiplicando a quantidade total pelo valor unitário médio (caso o item tenha sofrido variação de preço ao longo do trimestre);
- *Pedidos de pessoas jurídica (numérico)*: o percentual dos pedidos que foram feitos por pessoas jurídicas, sendo o restante é de pessoa física, pois só há na base estas duas opções;

- *Peso do produto (numérico)*: peso do produto em quilogramas, de certa forma correlacionado com as dimensões do produto;
- *Quantidade por pallet do produto (numérico)*: quantidade do produto que cabe em um *pallet*, sendo bastante associado ao volume do item;
- *Pedido mínimo (numérico)*: o menor pedido efetuado daquele item, em quantidade;
- *Pedido máximo (numérico)*: o maior pedido efetuado daquele item, em quantidade;
- *Pedido médio (numérico)*: a média da quantidade de todos os pedidos para o item;
- *Desvio padrão da quantidade do pedido (numérico)*: com base em todos os pedidos efetuados no período em questão, obtém-se o desvio padrão daqueles pedidos para o item. Esse valor dá uma ideia se o item é requisitado geralmente em quantidades similares ou distintas, onde quanto maior o valor mais variação na quantidade pedida a cada solicitação;
- *Preço médio (numérico)*: preço médio do item ao longo do trimestre. Como ele pode sofrer pequenas variações devido a promoções ou ações específicas, utilizamos a média do trimestre;
- *Frequência temporal de pedidos (numérico)*: contador do número de dias distintos que o produto foi pedido ao longo do trimestre;
- *Frequência total do item (numérico)*: percentual total de pedidos do trimestre que solicitaram aquele determinado produto;
- *Unidade de medida (categoria)*: unidade de medida do produto, o campo possui cinco valores distintos;
- *Família de produtos (categoria)*: agrupador dos produtos, identificando linhas de produtos específicas, que dentro de si possuem produtos similares. Possui no total vinte e sete valores distintos;
- *Marca (categoria)*: representa as diferentes marcas comercializadas pela empresa. Possui três valores distintos;
- *Trimestre no ano (numérico)*: trimestre o qual o pedido foi feito, sendo os possíveis valores de 1 a 4. Assim, dados do segundo trimestre do ano 2018 e 2019 tem o mesmo valor de trimestre do ano, neste exemplo o valor 2 (segundo semestre).

A definição de quais atributos devem ser utilizados envolve um conhecimento intrínseco do negócio da empresa, buscando identificar quais as variáveis mais importantes para a análise. Desta forma, foi conduzido um estudo junto à pessoa responsável por definir a política de estoques da empresa, de modo a fazer essa seleção criteriosa. Após esse

exercício, os campos escolhidos para serem utilizados na análise são mostrados a seguir, além de uma breve descrição do motivo de terem sido escolhidos:

- *Quantidade de pedidos*: considerado importante pois é relativo à frequência com que o item é requisitado;
- *Valor total*: considerado importante pois está intrinsicamente ligado ao custo do estoque;
- *Pedido médio*: considerado importante pois balanceia a frequência que o item é pedido, ou seja, contrapõe itens requisitados com muita frequência, porém em pequenas quantidades e vice-versa. Ele acaba trabalhando como uma regulação para *outliers* (dados discrepantes, que se diferenciam drasticamente dos demais);
- *Frequência temporal de pedidos*: considerado importante pois relaciona a frequência com que o item é requisitado com uma questão temporal, ou seja, itens que são consistentemente demandados ao longo do trimestre, ou apenas em momentos específicos, como promoções, divulgações de marketing e liquidações;
- *Frequência total do item*: considerado importante pois permite identificar itens que podem ser pouco importantes em aspectos de valor, porém estão presentes em um grande percentual dos pedidos, ou seja, acabam sempre compondo a venda.

Esses foram os campos considerados mais importantes para a análise, e eles foram escolhidos de forma a evitar redundâncias (por exemplo, escolher o valor total e a quantidade pedida torna o preço médio redundante, uma vez que as três variáveis são correlatas) e de forma a permitir às técnicas levar em consideração fatores importantes do ponto de vista de negócio como: controlar o custo em estoque nos CDs e garantir um bom nível de serviço através de uma alta disponibilidade dos produtos mais requisitados.

4.3 Variação de histórico nos dados de treino

Conforme visto anteriormente, é possível usar um ou mais trimestres de teste para treinar os modelos. Claro, se partirmos do trimestre onde N é igual a 2, temos apenas um trimestre de histórico a nossa disposição. Porém, à medida que vamos avançando no tempo, o número de trimestres no histórico passíveis de serem utilizados no treino do modelo aumenta. Por exemplo, para gerar classificação do trimestre onde N é igual a 10 seria possível utilizar os dados dos trimestres 1 ao 9.

Como o método ADU utiliza apenas um trimestre de histórico, e o objetivo aqui é

comparar a performance das técnicas de *Machine Learning* entre si e com a ADU, decidiu-se utilizar também apenas um trimestre de histórico na análise. Com essa definição se tem dados válidos para uso já a partir do segundo trimestre do conjunto de dados.

Para a sequência deste estudo, será utilizado um horizonte de 11 trimestres que servirão para calibragem dos parâmetros de cada uma das técnicas, sendo estes os trimestres de 2 a 12 da base. Na parte final deste trabalho, onde os modelos serão de fato testados como se estivessem em um ambiente real, os trimestres de 13 a 16 da base serão utilizados. Na tabela 4.1 é mostrado um resumo desta abordagem.

Tabela 4.1: Tabela dos trimestres contidos na base histórica de pedidos.

<i>Trimestre</i>	<i>Valor de N</i>	<i>Utilidade</i>
T1-2017	1	Sem uso
T2-2017	2	Calibragem
T3-2017	3	Calibragem
T4-2017	4	Calibragem
T1-2018	5	Calibragem
T2-2018	6	Calibragem
T3-2018	7	Calibragem
T4-2018	8	Calibragem
T1-2019	9	Calibragem
T2-2019	10	Calibragem
T3-2019	11	Calibragem
T4-2019	12	Calibragem
T1-2020	13	Teste
T2-2020	14	Teste
T3-2020	15	Teste
T4-2020	16	Teste
T1-2021	17	Sem uso

4.4 Seleção dos depósitos a serem analisados

O conjunto de dados possui dados de 6 depósitos, porém o histórico vendas só é completo para apenas 4 destes 6 depósitos. Os outros 2 foram depósitos criados após 2017, e com isso possuem um histórico de pedidos menor que os demais. Ainda que hajam dados suficientes para analisar estes depósitos (mais de 5 trimestres), os trimestres iniciais da operação de um novo depósito podem ser bastante atípicos, uma vez que diversos pontos são bastante incertos. Em geral o primeiro ano de operação de um novo depósito é de muito ajuste e adaptação, e por isso se decidiu eliminá-los da análise.

Dos 4 depósitos restantes, foram selecionados 2 deles para serem analisados no trabalho. Foram escolhidos os 2 depósitos com maiores volumes de operação, ou seja, aqueles que atendem os maiores volumes (tanto em quantidade de pedidos quanto em volume de mercadoria). Estes depósitos serão aqui em diante denominados como *Depósito A* e *Depósito B*, identificando respectivamente um dos depósitos da região sudeste e um dos depósitos da região sul. A escolha pelos maiores volumes de operação se dá pelo maior número de dados à disposição. Vale salientar que se tem um volume bem diferente de dados entre os dois depósitos, tendo o *Depósito A* em média 4 a 5 vezes mais pedidos do que o *Depósito B*.

4.5 Escolha de parâmetros para as técnicas de Machine Learning

Com a definição das variáveis, dos meses de histórico a serem utilizados e dos depósitos a serem analisados, serão avaliadas as performances individuais de cada uma das técnicas de *Machine Learning*, e nesta avaliação se buscará variar os parâmetros de cada um dos métodos de forma a melhorar a performance do algoritmo para o problema proposto.

Esta metodologia é chamada de *Hyperparameter tuning* ou *Hyperparameter optimization* (FEURER; HUTTER, 2019), e consiste em encontrar os melhores valores de parâmetros para cada uma das técnicas de modo a atingir uma melhor acurácia. Será utilizada uma configuração padrão de cada uma das técnicas utilizadas e com a variação dos parâmetros será avaliado o impacto no resultado. Para esta calibragem dos modelos será utilizado unicamente a acurácia como norteador.

4.5.1 Parâmetros para a técnica KNN

O primeiro método de *Machine Learning* aplicado foi o KNN. Para este método foram realizadas análises variando os seguintes parâmetros:

- *Valor de K*: o valor de K é sem dúvida o parâmetro mais importante. Ele decide quantas serão as amostras que poderão votar na classificação de uma nova instância;
- *Forma de voto*: as duas possibilidades para esse parâmetro são por distância ou uniforme, conforme explicado anteriormente.

Existem outros parâmetros que podem ser variados no método KNN, porém o foco foram estes dois parâmetros pelo entendimento que são os mais relevantes para a análise. Como a amostra possui muitos dados, o valor de K foi variado com valores entre 1 e 100. As figuras 4.2 e 4.3 mostram os gráficos de acurácia obtidos para cada um dos depósitos, respectivamente.

Figura 4.2: Acurácia obtida variando o valor de K para o *Depósito A*.

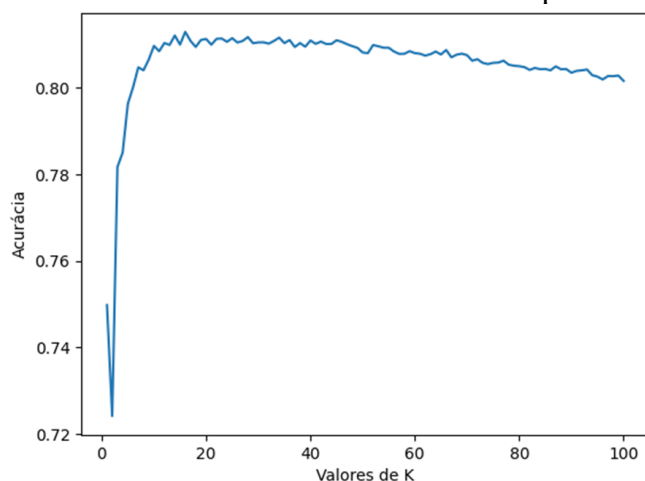
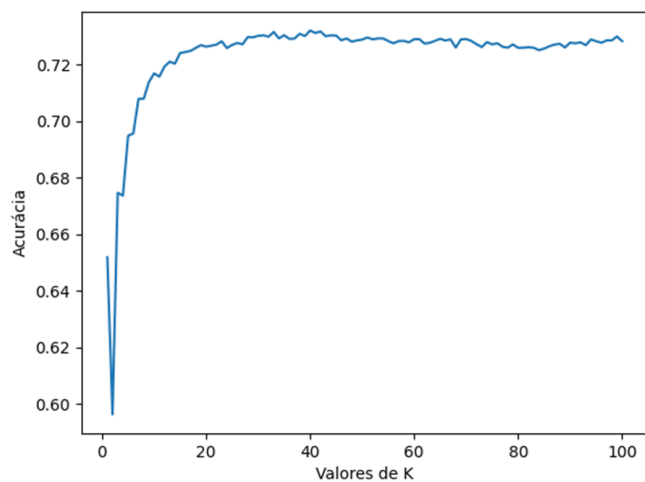


Figura 4.3: Acurácia obtida variando o valor de K para o *Depósito B*.



Os gráficos foram gerados pegando-se a acurácia média obtida com cada valor de K nos trimestres definidos como período de teste. De forma similar, foi feito um estudo observando a acurácia média obtida variando a forma de voto do modelo. A tabela 4.2 mostra os resultados obtidos com essa variação.

Com base nos valores de acurácia obtido, pode-se ver que a melhor forma de voto foi a *uniforme*, apesar de os valores serem bastante próximos entre si. Para os valores de K os valores que produziram uma melhor acurácia foram de 16 para o Depósito A e de 40 para o depósito B. Considerou-se prudente adotar um valor de K para cada depósito

Tabela 4.2: Acurácia obtida variando a forma de voto do modelo.

<i>Depósito</i>	<i>Forma de voto</i>	<i>Acurácia Média</i>
A	Uniforme	82,14%
A	Distância	82,04%
B	Uniforme	74,02%
B	Distância	73,95%

devido às peculiaridades de cada depósito, tanto em termos do volume de dados como da própria entrada e composição dos pedidos.

4.5.2 Parâmetros para a técnica SVM

Na sequência, o método SVM foi aplicado à amostra de dados de treino, e os parâmetros estudados para esta técnica foram:

- *Regularizador C*: este parâmetro tem como objetivo permitir ser mais ou menos estrito na classificação de dados de treino. O valor deve ser sempre positivo, porém um valor de C baixo informa ao modelo que aceitamos que hajam dados de treinos que sejam classificados de forma errônea. Já quando o valor de C é alto, o SVM corrige a fronteira de decisão para os dados de treino, de modo a evitar que ocorram classificações erradas. Ou seja, um valor alto pode trazer uma precisão melhor, mas pode também deixar o modelo viciado com os dados de treino, o que pode ser prejudicial para a performance. O parâmetro está, portanto, intimamente ligado com a possibilidade de *overfitting* do modelo;
- *Tipo de kernel*: este parâmetro afeta a forma como o modelo entende a similaridade entre as observações, e está bastante ligado com a dispersão geométrica do conjunto de dados sendo trabalhado. Os diferentes valores possíveis são: *linear* (função linear), *RBF* (função de base radial), *poly* (função polinomial) e *sigmoid* (função sigmoid);
- *Valor de Gamma*: o *Gamma* parâmetro que define quão próximo ou longe estarão as observações que definirão a maior distância entre as classes, na busca pelo plano que maximiza essa distância. Um valor alto de *Gamma* indica que apenas as observações mais próximas devem ser levadas em conta, enquanto um valor baixo indica que observações mais distantes também serão levadas em conta;
- *Função de decisão*: este parâmetro define qual função de decisão será utilizada na

classificação, se a função *one-vs-one* ou *one-vs-rest*, conforme discutido anteriormente. No caso do problema proposto ambas as funções produzem o mesmo número de classificadores, e, portanto, não possuem grande diferença de performance computacional, com isso é possível testar ambas as abordagens sem prejuízo. Em cenários com muitas classes a função *one-vs-one* produz um número maior de classificadores do que a função *one-vs-rest*, eventualmente inviabilizando seu uso.

Devido ao grande número de combinações possíveis entre os diferentes valores de cada parâmetro será utilizada uma abordagem de fixar os parâmetros em um valor padrão e variar apenas um parâmetro por vez, de forma a poder avaliar o impacto na acurácia do método de cada parâmetro individualmente. Como valores padrões de cada parâmetro foram definidos os valores de: C igual a 1, Γ igual a 1 e função de decisão *OVR* (*one-vs-rest*). Uma vez definido um parâmetro, o valor decidido como padrão é incorporado para a decisão do próximo parâmetro, e a ordem que se optou por avaliar os parâmetros foi: kernel, função de decisão, Γ e por fim o regulador C .

Na tabela 4.3 é mostrado o resultado obtido ao se variar o kernel utilizado, mantendo os demais parâmetros em seus valores padrão. Os valores são as acurácias médias obtidas para cada depósito ao longo dos trimestres de teste.

Tabela 4.3: Tabela dos trimestres contidos na base histórica de pedidos

<i>Depósito</i>	<i>Kernel</i>	<i>Função de Decisão</i>	C	Γ	<i>Acurácia Média</i>
A	Linear	OVR	1,0	1,0	66,90%
A	RFB	OVR	1,0	1,0	70,55%
A	Poly	OVR	1,0	1,0	61,19%
A	Sigmoid	OVR	1,0	1,0	66,92%
B	Linear	OVR	1,0	1,0	65,76%
B	RFB	OVR	1,0	1,0	68,25%
B	Poly	OVR	1,0	1,0	60,48%
B	Sigmoid	OVR	1,0	1,0	65,76%

Como pode-se ver os melhores resultados foram obtidos utilizando o kernel *RFB*. Em seguida, o parâmetro variado foi a função de decisão, no entanto os mesmos valores de acurácia foram obtidos com ambas as funções de decisão, exatamente os mesmos resultados obtidos na tabela 4.3. Como esse parâmetro não afetou o resultado, seu valor foi fixado no valor original, ou seja, o *OVR*.

O próximo parâmetro variado será o valor de Γ . Como Γ é um valor numérico, as possibilidades são infinitas, e, portanto, será utilizada uma abordagem de teste parecida com o valor de K no método KNN, ou seja, o modelo é testado contra

um conjunto pré-definido de valores. Os valores utilizados de teste foram variados entre 0,001 e 150, distribuídos de forma espaçada, porém não uniforme (de forma a concentrar mais valores próximos de 0 e 1). Nas figuras 4.4 e 4.5 pode-se ver o resultado da variação do parâmetro, onde o gráfico mostra a acurácia média obtida (no eixo y) com o valor de *Gamma* (no eixo x) em questão.

Figura 4.4: Acurácia obtida variando o valor de *Gamma* para o *Depósito A*.

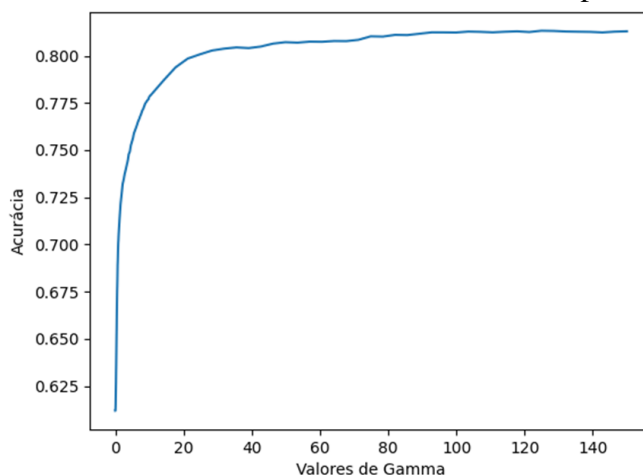
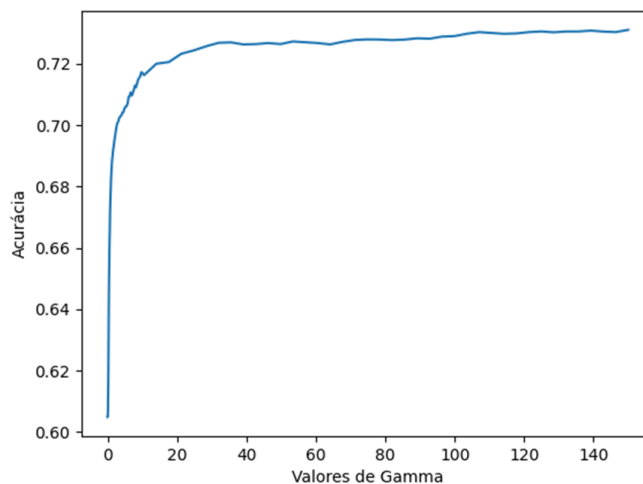


Figura 4.5: Acurácia obtida variando o valor de *Gamma* para o *Depósito B*.



Como pode ser visto nos gráficos, os melhores resultados de acurácia foram obtidos com valores de *Gamma* próximos a 110, que será, portanto, o valor utilizado para a sequência deste trabalho. Importante notar que a variação deste parâmetro produziu um ganho bastante interessante de acurácia, com um salto de 71% para aproximadamente 80% no *Depósito A* e de 68% para aproximadamente 72% no *Depósito B*.

Por fim o parâmetro *C* foi variado, novamente realizando uma distribuição de 100 valores não uniformes entre 0,001 e 30. Nas figuras 4.6 e 4.7 podemos visualizar o resultado obtido, novamente com a acurácia no eixo y e os valores de *C* no eixo x:

Figura 4.6: Acurácia obtida variando o valor de C para o *Depósito A*.

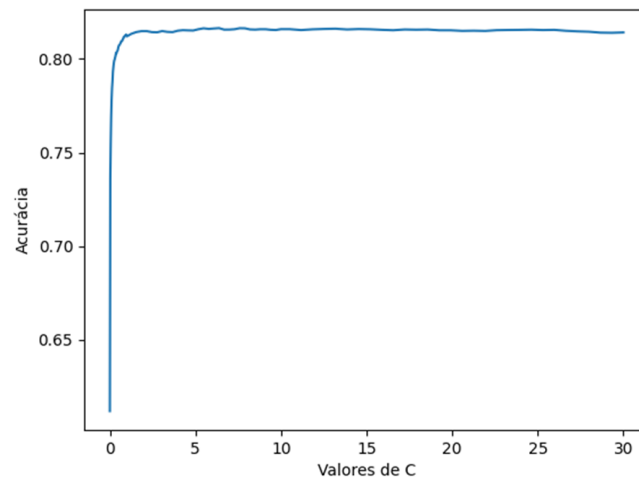
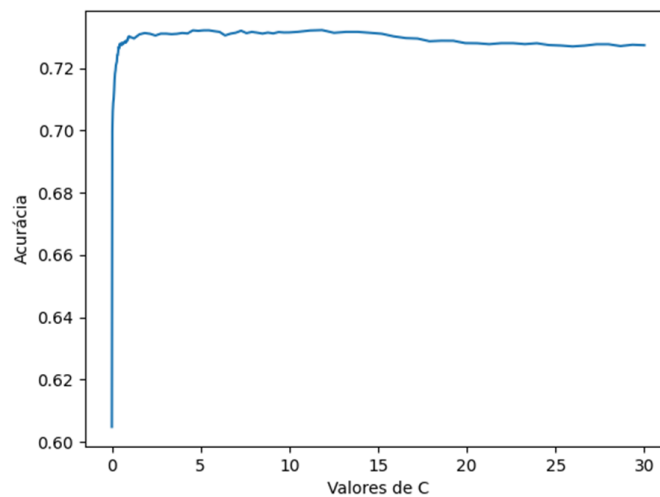


Figura 4.7: Acurácia obtida variando o valor de C para o *Depósito B*.



Podemos notar que os melhores resultados de acurácia do método foram obtidos com C próximo de 5, onde a partir deste valor a acurácia permanece quase que constante, ou até decrescente. A configuração que será adotada para a continuação deste trabalho é:

- *Regularizador C*: 5
- *Tipo de kernel*: *RFB*
- *Valor do Gamma*: 110
- *Função de decisão*: *OVR*

4.5.3 Parâmetros para a técnica XGBoost

A terceira e última técnica aplicada foi a XGBoost, que conta com uma variedade grande de parâmetros. Devido ao grande número de parâmetros, foram selecionados

aqueles julgados mais interessantes de serem variados neste estudo, que produziriam os maiores efeitos na acurácia do método. Os parâmetros testados foram:

- *Valor do eta*: também chamado de taxa de aprendizado, ele afeta a forma como o modelo cria árvores para lidar com os erros residuais dos passos anteriores. Nesse processo iterativo, o modelo pode acabar criando um *overfitting* nos dados, e esse parâmetro visa controlar esse problema;
- *Valor da profundidade máxima (max_depth)*: é a profundidade máxima da árvore, e acaba tendo um papel importante também no controle do *overfitting*, uma vez que uma profundidade maior permite ao modelo buscar se adaptar a casos mais específicos dos dados de teste;
- *Valor de Gamma*: este parâmetro atua na decisão de quando particionar ou não um nodo da árvore, funcionando como um regularizador. Quanto mais alto o valor de *Gamma*, mais conservador o modelo será.

Além dos parâmetros definidos acima, para usar corretamente o classificador XG-Boost para a natureza do problema proposto, foi necessário definir mais alguns parâmetros com valores específicos, conforme mostrado abaixo:

- *Valor do booster*: define qual o booster utilizado, neste caso será utilizado árvores, ou seja, usa-se o valor *gbtree*;
- *Valor do objective*: define qual a função de aprendizado. Será utilizado o valor *multi:softmax*, adequado para o problema de classificação multiclases;
- *Valor do num_class*: é o número de classes em classificações multiclases, e neste caso o valor é 3, representando as classes A, B e C;
- *Valor do eval_metric*: o valor escolhido é o *mlogloss*, utilizado para cenários multiclases.

Similarmente ao que foi feito no SVM anteriormente, aqui também será variado um parâmetro de cada vez, partindo dos valores padrões de cada um deles, e analisando como afetam a performance. A ordem em que os parâmetros serão variados foi definida como: *eta*, *max_depth* e *Gamma*. Os valores padrões escolhidos para análise foram: *max_depth* igual a 6, *Gamma* igual a 0.

A forma de analisar o impacto na acurácia da variação os parâmetros é análoga ao que foi feito anteriormente no KNN e SVM, avaliando a acurácia média obtida nos trimestres de teste utilizando uma gama pré-definida de valores para cada parâmetro, uma vez que são todos parâmetros numéricos. Os resultados obtidos na variação dos parâmetros

em termos de acurácia do método são apresentados nas figuras 4.8 e 4.9.

Figura 4.8: Acurácia obtida variando o valor de η para o *Depósito A*.

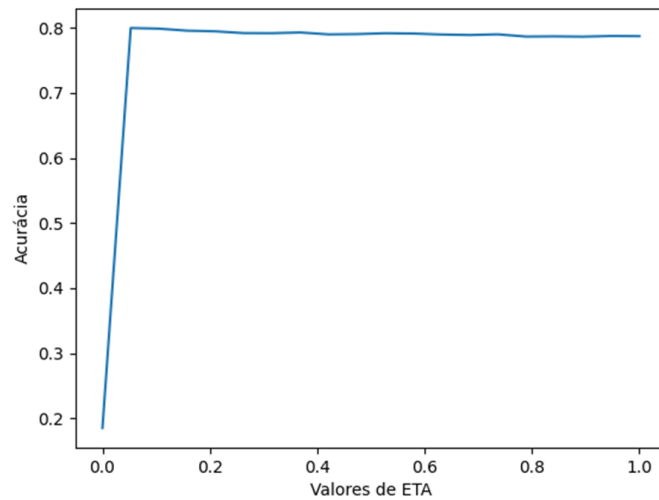
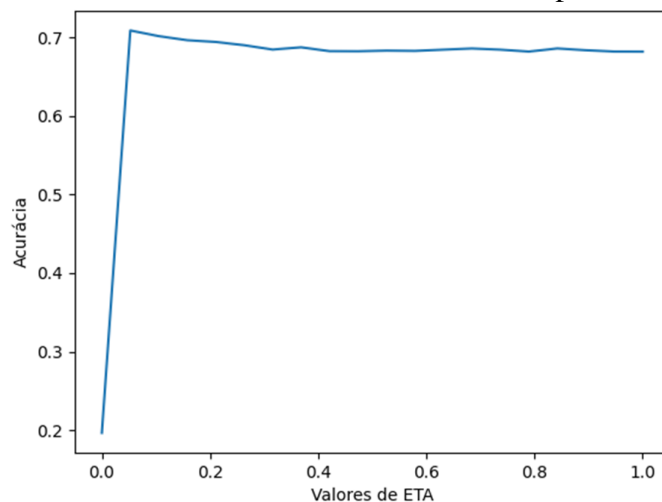


Figura 4.9: Acurácia obtida variando o valor de η para o *Depósito B*.



O η pode variar entre 0 e 1, e para o teste foram utilizados 20 valores distintos, uniformemente espalhados entre os valores máximo e mínimo permitidos. Podemos notar pelos gráficos acima que o melhor valor para ambos os depósitos ficou próximo de 0,05, que será o valor utilizado doravante.

O próximo parâmetro variado foi o valor da profundidade máxima. Os valores possíveis aqui são apenas os inteiros positivos, então variou-se o valor entre 1 e 30, de modo a avaliar como o parâmetro afeta na acurácia. Os resultados são mostrados nas figuras 4.10 e 4.11 e podemos ver que os melhores resultados foram obtidos com valores bem baixos de profundidade.

Por fim, foi variado o parâmetro Γ , o qual permite qualquer valor entre 0 e infinito. Desta forma, foram pré-definidos 100 valores distribuídos entre 0,001 e 100,

Figura 4.10: Acurácia obtida variando o valor de max_depth para o Depósito A.

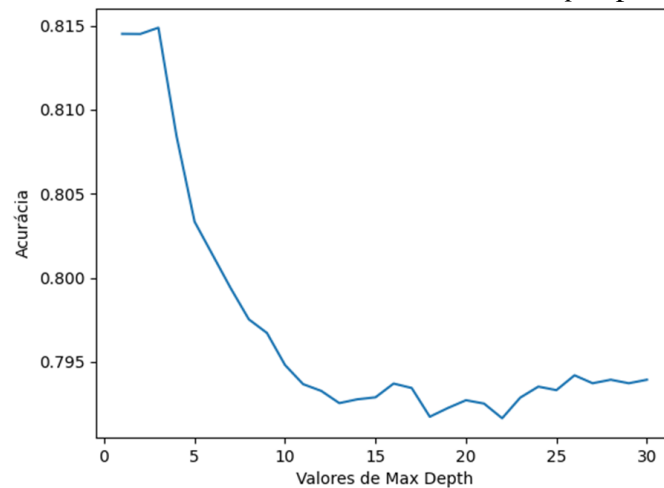
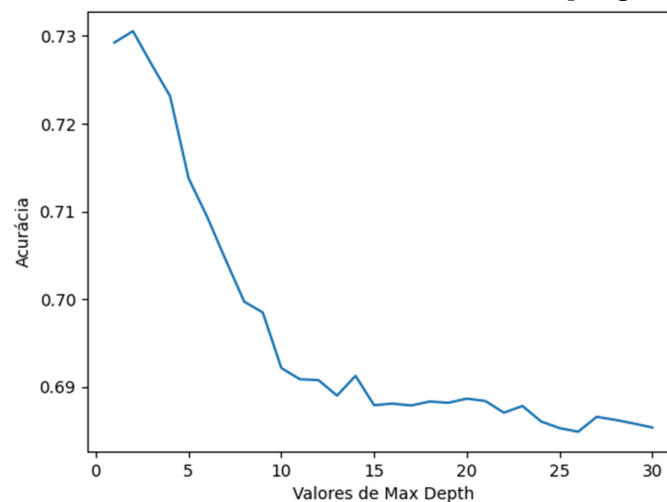


Figura 4.11: Acurácia obtida variando o valor de max_depth para o Depósito B.



de maneira não uniforme. Os valores de Γ que produziram os melhores resultados foram valores próximos de 2,5, conforme podemos ver nas figuras 4.12 e 4.13.

Conforme os dados obtidos, podemos ver que a configuração de parâmetros que produziu a melhor performance do método, e que, portanto, será utilizada na sequência deste trabalho, foram os valores a seguir:

- Valor do η : 0,05
- Valor da profundidade máxima (max_depth): 3
- Valor de Γ : 2,5

Figura 4.12: Acurácia obtida variando o valor de *Gamma* para o *Depósito A*.

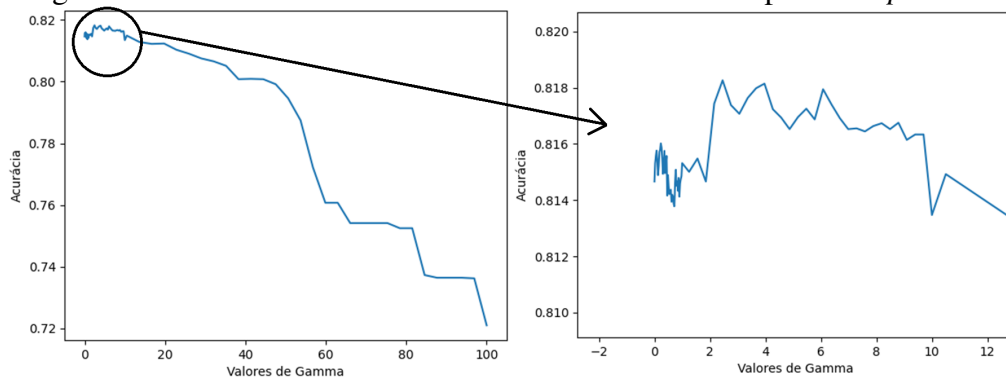
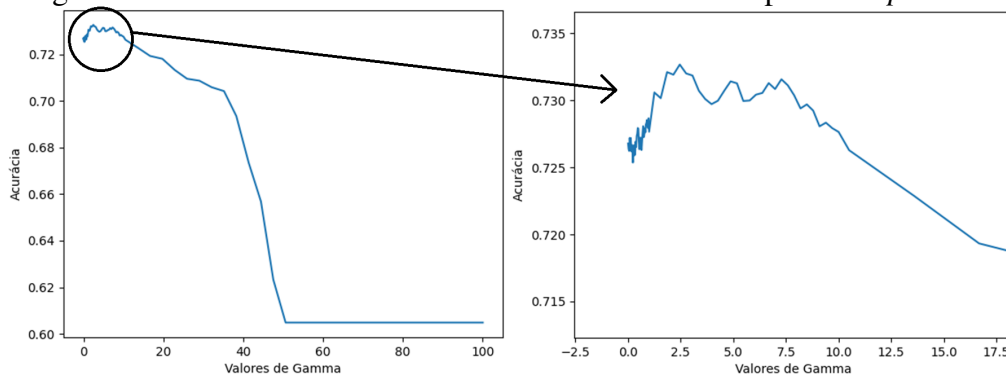


Figura 4.13: Acurácia obtida variando o valor de *Gamma* para o *Depósito B*.



4.6 Resultado da aplicação das técnicas de Machine Learning

Com as definições feitas até aqui, o próximo passo é aplicar os modelos de *Machine Learning* no conjunto de dados de teste e avaliar a performance obtida, comparando os cenários obtidos com as diferentes técnicas com o cenário referência.

Até aqui foram definidos: campos a serem utilizados, meses de histórico, depósitos a serem avaliados e os parâmetros específicos para cada uma das técnicas. Vamos então aplicar às técnicas aos trimestres de teste definidos anteriormente, ou seja, aos trimestres 13 a 16.

Primeiro será analisada a performance obtida pelas técnicas de *Machine Learning* no *Depósito A*. As métricas aqui analisadas são a acurácia e o custo obtido para a classificação ABC efetuada, com base nas fórmulas apresentadas anteriormente. Os melhores métodos são aqueles com uma acurácia maior e um custo menor. A tabela 4.4 mostra os resultados obtidos ao aplicar as diferentes técnicas nos trimestres de teste, e em destaque (negrito) estão os melhores resultados de cada trimestre.

Como podemos ver na tabela 4.4, com exceção do trimestre 14, o resultado obtido pelas diferentes técnicas de *Machine Learning* foi levemente superior ao resultado obtido pela ADU em termos de acurácia, porém consistentemente superior. No trimestre 16 é

Tabela 4.4: Resultados obtidos nos trimestres de treino com os métodos aplicados no Depósito A.

<i>Técnica</i>	<i>Trimestre</i>	<i>Acurácia (%)</i>	<i>Custo (R\$)</i>
ADU	13	75,58	1.866.526,00
KNN	13	76,78	1.755.027,00
SVM	13	77,22	1.742.839,00
XGBoost	13	76,67	1.764.401,00
ADU	14	75,11	1.731.101,00
KNN	14	74,45	1.643.051,00
SVM	14	73,67	1.594.790,00
XGBoost	14	75,11	1.664.494,00
ADU	15	75,66	2.197.281,00
KNN	15	76,18	2.128.926,00
SVM	15	80,88	2.082.926,00
XGBoost	15	78,02	2.112.601,00
ADU	16	59,71	2.887.193,00
KNN	16	72,78	2.720.227,00
SVM	16	78,54	2.610.398,00
XGBoost	16	74,58	2.677.491,00

onde houve a diferença mais significativa, com quase 20 pontos percentuais de diferença na acurácia do melhor método de ML em comparação com a ADU.

Comparando-se os métodos de *Machine Learning* entre si, nota-se que a melhor performance em termos tanto de acurácia como custo foi obtida pelo SVM, com exceção da acurácia no trimestre 14. Analisando os resultados fica claro que as três técnicas estudadas apresentaram resultados na média melhores que aqueles obtidos pela classificação utilizando o método ADU. A tabela 4.5 mostra a acurácia média e o custo total obtidos com cada técnica de classificação a título de comparação.

Tabela 4.5: Resumo dos resultados das técnicas no Depósito A.

<i>Técnica (posição)</i>	<i>Acurácia Média (%)</i>	<i>Custo Total (R\$)</i>
ADU(4)	71,52	8.682.101,00
KNN(3)	75,05	8.247.206,00
SVM(1)	77,56	8.030.953,00
XGB(2)	76,10	8.218.987,00

Para o *Depósito B* foi feita a mesma análise, tendo sido aplicadas todas as técnicas nos trimestres definidos como teste. Conforme visto anteriormente, apenas a técnica do KNN possui parâmetros diferentes entre os dois depósitos, sendo as demais técnicas configuradas de forma idêntica. Os resultados são mostrados na tabela 4.6, de forma análoga aos apresentados anteriormente:

Tabela 4.6: Resultados obtidos nos trimestres de treino com os métodos aplicados no Depósito B.

<i>Técnica</i>	<i>Trimestre</i>	<i>Acurácia (%)</i>	<i>Custo (R\$)</i>
ADU	13	67,85	794.915,00
KNN	13	73,77	699.116,00
SVM	13	74,11	709.146,00
XGBoost	13	75,47	701.763,00
ADU	14	66,29	793.028,00
KNN	14	71,91	693.928,00
SVM	14	71,54	676.067,00
XGBoost	14	72,47	701.866,00
ADU	15	69,04	813.496,00
KNN	15	75,62	769.066,00
SVM	15	75,44	756.570,00
XGBoost	15	74,38	777.468,00
ADU	16	53,30	907.041,00
KNN	16	81,28	788.309,00
SVM	16	81,28	799.496,00
XGBoost	16	75,77	839.283,00

No Depósito B obtive-se resultados um pouco diferentes daqueles do Depósito A, no entanto também é possível notar, ainda mais claramente que nos dados anteriores, que as técnicas de *Machine Learning* apresentaram um desempenho consistentemente superior à classificação realizada com o método ADU. No trimestre 16 obteve-se a maior diferença em termos de acurácia, onde tanto o KNN como o SVM obtiveram aproximadamente 28 pontos percentuais a mais de acurácia que o ADU.

Comparando as técnicas de *Machine Learning* entre si podemos ver que os melhores resultados foram obtidos pelas técnicas SVM e KNN, que tiveram performance bastante similares tanto em termos de acurácia como custo, praticamente empatadas. O XGBoost teve uma performance levemente inferior aos outros dois, mas ainda assim bastante superior ao ADU, conforme podemos observar na tabela 4.7 que resume os dados.

Tabela 4.7: Resumo dos resultados das técnicas no Depósito B.

<i>Técnica (posição)</i>	<i>Acurácia Média (%)</i>	<i>Custo Total (R\$)</i>
ADU(3)	64,12	3.308.480,00
KNN(1*)	75,65	2.950.419,00
SVM(1*)	75,59	2.941.279,00
XGB(2)	74,52	3.020.380

*Considerados empatados devido a pequena diferença percentual entre os resultados, considerando tanto custo como acurácia.

Outro ponto a ser destacado é com relação ao custo total obtido com as diferentes

classificações, que apresentaram uma variação não tão acentuada, porém tampouco desprezível. Considerando os valores totais de custo, que nada mais são do que a soma dos custos de cada trimestre individualmente, podemos ver na tabela 4.8 que temos reduções que chegam a até 7,5% e 11% comparadas ao resultado da classificação ADU, para os depósitos *A* e *B*, respectivamente. A tabela 4.8 mostra uma comparação dos resultados de custo obtidos por cada uma das técnicas e sua variação em relação a classificação de referência, a do método ADU.

Tabela 4.8: Resumo do custo obtido em cada depósito.

<i>Técnica</i>	<i>Depósito</i>	<i>Custo Total (R\$)</i>	<i>Redução em relação ao ADU (%)</i>
ADU	A	8.682.101,00	-
KNN	A	8.247.206,00	5,01
SVM	A	8.030.953,00	7,50
XGBoost	A	8.218.987,00	5,33
ADU	B	3.308.480,00	-
KNN	B	2.950.419,00	10,82
SVM	B	2.941.279,00	11,10
XGBoost	B	3.020.380,00	8,71

Por fim, na tabela 4.9 é apresentado um resumo com os resultados condensados de ambos os depósitos, com o intuito de facilitar a compreensão da técnica que melhor performou em termos tanto de acurácia quanto custo. Para o *Depósito A* fica evidente que a melhor técnica foi o SVM, enquanto para o *Depósito B* tivemos ambos SVM e KNN com uma ótima performance.

Tabela 4.9: Comparação das técnicas que melhor performaram por trimestre.

<i>Métrica</i>	<i>Depósito</i>	<i>Trimestre 13</i>	<i>Trimestre 14</i>	<i>Trimestre 15</i>	<i>Trimestre 16</i>
Acurácia	A	SVM	ADU/SVM	SVM	SVM
Acurácia	B	XGBoost	XGBoost	KNN	KNN/SVM
Custo	A	SVM	SVM	SVM	SVM
Custo	B	KNN	SVM	SVM	KNN

Na média, a técnica que melhor performou, e que seria, portanto, a escolhida para ser utilizada pela empresa, foi a técnica SVM, especialmente no critério de custo, obtendo os melhores resultados. No entanto, conforme veremos no próximo parágrafo, devido ao volume de dados relativamente pequeno do nosso cenário, seria totalmente viável para a empresa trabalhar com todos os classificadores e avaliar posteriormente qual se deseja utilizar, podendo-se inclusive escolher classificadores diferentes para depósitos diferentes, o que faz sentido, devido às diferentes características de cada depósito, como demanda e volume de pedidos.

Em termos de performance computacional todas as técnicas aplicadas apresentaram tempos de execução baixo para a realização da classificação de um trimestre, considerando o volume de dados utilizado. O *Depósito A*, que possui o maior volume de dados do nosso conjunto de dados, possui cerca de 1800 produtos por trimestre. A tabela 4.10 mostra um comparativo dos tempos de execução do método ADU e as técnicas de *Machine Learning*. Podemos ver que as técnicas foram duas a três vezes mais lentas que o método ADU, que por sua vez é bastante simples de ser calculado, porém com o volume de dados do nosso conjunto a performance computacional foi bastante satisfatórios.

Tabela 4.10: Comparação do tempo de execução dos diferentes classificadores, em segundos.

<i>Depósito</i>	<i>ADU (s)</i>	<i>KNN (s)</i>	<i>SVM (s)</i>	<i>XGBoost (s)</i>
A	12	23	31	34
B	8	19	25	23

5 CONCLUSÃO

Neste trabalho realizou-se um estudo de caso da aplicação de técnicas de *Machine Learning* empregadas na tarefa de realizar uma classificação ABC para a armazenagem de produtos nos diferentes depósitos de uma indústria de cerâmica brasileira. Através de uma base histórica de pedidos composta de 17 períodos trimestrais, foram utilizados 11 períodos para treino e otimização de parâmetros e 4 períodos para teste dos modelos, simulando um comportamento similar ao que seria o uso real dos classificadores.

Como foi possível observar nos resultados obtidos, as técnicas estudadas neste trabalho se mostraram uma boa alternativa para gerarmos uma classificação ABC multi-critérios. Os resultados mostraram que, para o conjunto de dados utilizados, as técnicas apresentaram uma performance superior à classificação feita com o método ADU, tanto em termos de acurácia como em termos de custo. Além disso, a utilização de técnicas de *Machine Learning* aparece como uma boa alternativa a métodos de classificação MCIC, devido à possibilidade da inclusão de novas variáveis sem grande efeito na performance computacional e complexidade de aplicação do método, como observado em alguns métodos MCIC.

Uma das dificuldades encontradas neste trabalho foi com relação ao conjunto de dados de treino do modelo. Uma vez que diversos interesses estão em jogo, não há uma forma certa ou ideal de se determinar qual é a classificação ABC correta de um conjunto de produtos. Por possuir uma base histórica, foi possível gerar uma classificação ADU nos pedidos realizados e usar esse resultado como dado de treino dos modelos. Contudo, outras classificações poderiam ser utilizadas para treinar os modelos, como por exemplo a própria classificação ABC real da empresa, atualmente gerada com método MCIC próprio.

Uma outra limitação encontrada diz respeito ao grande número de combinações possíveis em termos de possibilidades de parâmetros dos diferentes modelos. Especialmente as técnicas SVM e XGB possuem muitos parâmetros e foi necessário limitar a análise a apenas alguns deles. É possível que uma otimização de outros parâmetros que não foram abordados neste trabalho pudessem levar a ganhos ainda maiores do que os obtidos.

Este estudo abre uma grande possibilidade de trabalhos futuros visando ampliar as possibilidades de aplicação de técnicas de *Machine Learning* ao problema proposto. Neste trabalho foram utilizadas apenas técnicas de aprendizado supervisionado, porém,

dados os resultados encontrados nas referências deste trabalho, entende-se que a utilização de técnicas de aprendizado não supervisionado merecem ser avaliadas. Outra possível contribuição futura é comparar os resultados obtidos utilizando-se as técnicas de ML com outros métodos de classificação ABC, como os métodos MCIC aqui citados.

No que diz respeito ao conjunto de dados da empresa, a sequência deste trabalho pode ampliar a análise para os demais depósitos da empresa, aumentar a gama de possíveis parâmetros de entrada (com informações como tempo de ressuprimento, por exemplo), reavaliar novas combinações de parâmetros de entrada dos modelos e variar o número de meses de históricos utilizado na análise.

REFERÊNCIAS

BHATTACHARYA, A.; SARKAR, B.; MUKHERJEE, S. K. Distance-based consensus method for abc analysis. **International Journal of Production Research**, Taylor & Francis, v. 45, n. 15, p. 3405–3420, 2007.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: **Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining**. [S.l.: s.n.], 2016. p. 785–794.

CHEN, Y.; LI, K. W.; LIU, S.-f. A comparative study on multicriteria abc analysis in inventory management. In: IEEE. **2008 IEEE International Conference on Systems, Man and Cybernetics**. [S.l.], 2008. p. 3280–3285.

COHEN, M. A.; ERNST, R. Multi-item classification and generic inventory stock contr. **Production and Inventory Management Journal**, American Production & Inventory Control Society, Inc., v. 29, n. 3, p. 6, 1988.

DAS, D. et al. Two-warehouse production model for deteriorating inventory items with stock-dependent demand under inflation over a random planning horizon. **Central Eur. J. Oper. Res.**, v. 20, n. 2, p. 251–280, 2012.

FEURER, M.; HUTTER, F. Hyperparameter optimization. In: **Automated machine learning**. [S.l.]: Springer, Cham, 2019. p. 3–33.

FLORES, B. E.; OLSON, D. L.; DORAI, V. Management of multicriteria inventory classification. **Mathematical and Computer Modelling**, v. 16, n. 12, p. 71–82, 1992. ISSN 0895-7177.

FLORES, B. E.; WHYBARK, D. C. Implementing multiple criteria abc analysis. **Journal of Operations Management**, Wiley Online Library, v. 7, n. 1-2, p. 79–85, 1987.

GROOVER, M. P. **Automation, Production Systems, and Computer-Integrated Manufacturing**. 2nd. ed. USA: Prentice Hall PTR, 2000. ISBN 0130889784.

JOLAI, F.; RAZMI, J.; ROSTAMI, N. A fuzzy goal programming and meta heuristic algorithms for solving integrated production: distribution planning problem. **Central European Journal of Operations Research**, v. 19, p. 547–569, 2011.

KARTAL, H.; CEBI, F. Support vector machines for multi-attribute abc analysis. **International Journal of Machine Learning and Computing**, p. 154–157, 01 2013.

LAJILI, I.; LADHARI, T.; BABAI, Z. Adaptive machine learning classifiers for the class imbalance problem in abc inventory classification. In: **6th International Conference on Information Systems, Logistics and Supply Chain. ILS Conference, Bordeaux, France**. [S.l.: s.n.], 2016.

LORENA, A. C.; CARVALHO, A. C. de. Uma introdução às support vector machines. **Revista de Informática Teórica e Aplicada**, v. 14, n. 2, p. 43–67, 2007.

MITCHELL, T. M. et al. **Machine learning**. McGraw-hill New York, 1997.

MOHAMMADITABAR, D.; GHODSYPOUR, S. H.; O'BRIEN, C. Inventory control system design by integrating inventory classification and policy selection. **International Journal of Production Economics**, Elsevier, v. 140, n. 2, p. 655–659, 2012.

PARTOVI, F.; ANANDARAJAN, M. Classifying inventory using an artificial neural network approach. **Computers Industrial Engineering**, v. 41, p. 389–404, 02 2002.

PARTOVI, F. Y.; BURTON, J. Using the analytic hierarchy process for abc analysis. **International Journal of Operations & Production Management**, MCB UP Ltd, 1993.

PETERSON, L. E. K-nearest neighbor. **Scholarpedia**, v. 4, n. 2, p. 1883, 2009.

RAMANATHAN, R. Abc inventory classification with multiple-criteria using weighted linear optimization. **Computers & operations research**, Elsevier, v. 33, n. 3, p. 695–700, 2006.

TEUNTER, R.; BABAI, M. Z.; SYNTETOS, A. Abc classification: Service levels and inventory costs. **Production and Operations Management**, v. 19, p. 343 – 352, 05 2010.

TORABI, S. A.; HATEFI, S. M.; PAY, B. S. Abc inventory classification in the presence of both quantitative and qualitative criteria. **Computers & Industrial Engineering**, Elsevier, v. 63, n. 2, p. 530–537, 2012.

YU, M.-C. Multi-criteria abc analysis using artificial-intelligence-based classification techniques. **Expert Systems with Applications**, Elsevier, v. 38, n. 4, p. 3416–3421, 2011.