



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CAMPUS AGRESTE
NÚCLEO DE TECNOLOGIA
CURSO DE ENGENHARIA DE PRODUÇÃO

PEDRO LAURENTINO DA SILVA NETO

**APLICAÇÃO DA CLASSIFICAÇÃO ABC COM APRENDIZADO DE MÁQUINA
SEMI-SUPERVISIONADO NA GESTÃO DE ESTOQUE DE MATERIAIS DE MRO
EM INDÚSTRIAS DE GRANDE PORTE.**

Caruaru
2025

PEDRO LAURENTINO DA SILVA NETO

**APLICAÇÃO DA CLASSIFICAÇÃO ABC COM APRENDIZADO DE MÁQUINA
SEMI-SUPERVISIONADO NA GESTÃO DE ESTOQUE DE MATERIAIS DE MRO
EM INDÚSTRIAS DE GRANDE PORTE.**

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de engenharia de produção do Campus Agreste da Universidade Federal de Pernambuco – UFPE, na modalidade de monografia, como requisito parcial para a obtenção do grau de bacharel em Engenharia de Produção.

Área de concentração: Logística

Orientador (a): Lucimário Gois de Oliveira Silva

Caruaru

2025

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Silva Neto, Pedro Laurentino da.

Aplicação da classificação ABC com aprendizado de máquina semi-supervisionado na gestão de estoque de materiais de MRO em indústrias de grande porte. / Pedro Laurentino da Silva Neto. - Caruaru, 2025.

61 p. : il., tab.

Orientador(a): Lucimário Gois de Oliveira Silva

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro Acadêmico do Agreste, Engenharia de Produção, 2025.
Inclui referências, apêndices.

1. gestão de estoques. 2. classificação ABC. 3. aprendizado de máquina. 4. MRO. I. Oliveira Silva, Lucimário Gois de. (Orientação). II. Título.

620 CDD (22.ed.)

PEDRO LAURENTINO DA SILVA NETO

**APLICAÇÃO DA CLASSIFICAÇÃO ABC COM APRENDIZADO DE MÁQUINA
SEMI-SUPERVISIONADO NA GESTÃO DE ESTOQUE DE MATERIAIS DE MRO
EM INDÚSTRIAS DE GRANDE PORTE.**

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Engenharia de Produção do Campus Agreste da Universidade Federal de Pernambuco – UFPE, na modalidade de monografia, como requisito parcial para a obtenção do grau de bacharel em engenharia de produção.

Aprovado em: 08/08/2025

BANCA EXAMINADORA

Prof. Dr. Lucimário Gois de Oliveira Silva (Orientador)
Universidade Federal de Pernambuco

Prof. Dr. José Leão (Examinador Interno)
Universidade Federal de Pernambuco

Prof. Dr. Isaac Pergher (Examinador Interno)
Universidade Federal de Pernambuco

AGRADECIMENTOS

A Deus, por me dar força, fé e me guiar em todos os momentos.

Agradeço imensamente ao meu orientador, Prof. Lucimário Gois, por ter abraçado e apoiado a proposta deste tema desde o seu início. Sua disposição para as reuniões, o indispensável auxílio técnico e as valiosas ideias sugeridas foram fundamentais para o enriquecimento e a concretização desta pesquisa.

À Universidade Federal de Pernambuco (UFPE) e ao curso de Engenharia de Produção, pela estrutura oferecida e pelo corpo docente que foi essencial em minha formação acadêmica e profissional. Estendo minha gratidão à empresa que abriu suas portas e permitiu a realização deste estudo de caso, pela confiança e pela oportunidade de aplicar os conhecimentos teóricos em um desafio prático e relevante.

A jornada para a conclusão deste trabalho não teria sido possível sem o alicerce da minha família. Aos meus pais, Iranilde Laurentino e Edmilson de Moraes, por serem minha base e inspiração; à minha noiva, Débora Lúcia, por todo o amor e paciência durante as longas horas de estudo e ausência; e à minha irmã, Maria Eduarda. Agradeço a vocês pelo apoio incondicional e por nunca me deixarem desistir. Esta conquista é nossa.

Aos meus amigos e companheiros de jornada, Gabriel Almeida, Juliana Larissa e Michele Santos, minha eterna gratidão. Obrigado pelos anos de estudo compartilhados, pela parceria sólida que formamos e por ouvirem pacientemente cada desabafo sobre os desafios deste TCC.

A todos que, direta ou indiretamente, fizeram parte desta etapa, o meu muito obrigado.

RESUMO

A gestão de estoques de materiais de Manutenção, Reparo e Operações (MRO) em indústrias de grande porte é um desafio complexo, e a classificação ABC tradicional, baseada em variáveis únicas ou subjetivas, mostra-se frequentemente inconsistente e ineficaz para mitigar riscos operacionais. Este trabalho teve como objetivo analisar a aplicação de algoritmos de aprendizado de máquina para desenvolver um modelo de classificação ABC mais robusto e alinhado aos objetivos operacionais de uma indústria. A metodologia baseou-se em um estudo de caso em uma indústria metalúrgica, utilizando um conjunto de dados sintético de 6.530 SKUs para preservar a confidencialidade. Inicialmente, a inconsistência do método de classificação vigente na empresa foi quantificada por meio da abordagem Dominance-based Rough Set Approach (DRSA). Em seguida, uma amostra de referência foi classificada por um especialista e utilizada para treinar e comparar cinco algoritmos de aprendizado supervisionado. O modelo de melhor desempenho, Support Vector Machine (SVM), foi selecionado por meio de validação cruzada e aplicado em um modelo de aprendizado semi-supervisionado para classificar todo o inventário. Os resultados demonstraram que a classificação original da empresa possuía uma qualidade de aproximação de apenas 12%, enquanto a do especialista alcançou 75%. O modelo SVM obteve a maior acurácia média (80%) e, na validação final, a classificação gerada pelo modelo apresentou 80% de concordância com o julgamento do especialista em uma nova amostra. Conclui-se que o método proposto substitui um processo atual e falho por um sistema escalável e que reflete com maior precisão a criticidade dos itens, oferecendo uma ferramenta validada para otimizar a gestão de estoques de MRO e reduzir o risco de rupturas.

Palavras-Chave: gestão de estoques; classificação ABC; aprendizado de máquina; MRO.

ABSTRACT

Inventory management of Maintenance, Repair, and Operations (MRO) materials in large-scale industries is a complex challenge, and the traditional ABC classification, based on single or subjective variables, often proves to be inconsistent and ineffective for mitigating operational risks. This study aimed to analyze the application of machine learning algorithms to develop a more robust ABC classification model aligned with the operational objectives of an industry. The methodology was based on a case study in a metallurgical industry, using a synthetic dataset of 6,530 SKUs to preserve confidentiality. Initially, the inconsistency of the company's current classification method was quantified through the Dominance-based Rough Set Approach (DRSA). Subsequently, a reference sample was classified by an expert and used to train and compare five supervised learning algorithms. The best-performing model, Support Vector Machine (SVM), was selected through cross-validation and applied in a semi-supervised learning framework to classify the entire inventory. The results demonstrated that the company's original classification had a quality of approximation of only 12%, while the expert's classification reached 75%. The SVM model achieved the highest mean accuracy (80%), and in the final validation, the classification generated by the model showed 80% agreement with the expert's judgment on a new sample. It is concluded that the proposed method replaces a subjective and flawed process with a scalable system that more accurately reflects the criticality of the items, offering a validated tool to optimize MRO inventory management and reduce the risk of stockouts.

Keywords: inventory management; ABC classification; machine learning; MRO.

LISTA DE ILUSTRAÇÕES

Figura 1 - Função básica dos estoques	17
Figura 2 - Comportamento do estoque de Produtos Acabados X Comportamento de estoque MRO.	19
Figura 3 - Curva de Pareto para itens em um armazém.	21
Figura 4 - Hierarquia do aprendizado de máquina.	23
Figura 5 - Fluxograma metodológico.	29
Figura 6 - Base de dados do estudo.	30
Figura 7 - Qualidade da aproximação atual gerada pelo jmaf.	34
Figura 8 - Qualidade da aproximação do especialista gerado pelo jmaf.	35
Figura 9 - Classificação atual.	36
Figura 10 - Matriz de confusão do random forest na base inicial.	38
Figura 11 - Matriz de confusão do modelo de rede neural para a base do especialista.....	40
Figura 12 - SVM X Especialista.....	41
Figura 13 - Classificação final.	42
Figura 14 - Qualidade da aproximação do SVM pelo jmaf.	43

LISTA DE TABELAS

Tabela 1 - Parâmetros dos modelos para a base inicial.....	36
Tabela 2 - Resultado dos modelos na base inicial.	37
Tabela 3 - Parâmetros dos modelos para a base de dados do especialista.	39
Tabela 4 - Resultado dos modelos na base do especialista.	39

LISTA DE ABREVIATURA E SIGLAS

MRO	Manutenção, Reparo e Operações
DRSA	Dominance-based Rough Set Approach
RST	Rough Set Theory
SSL	Semi-supervised learning
TD	Tomador de Decisão
MLPNN	Multilayer perceptron neural network
SKUs	Stock Keeping Units
KNN	K-Nearest Neighbors
SVM	Support Vector Machine

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO GERAL.....	13
1.1.1	Objetivos específicos.....	14
1.2	JUSTIFICATIVA	14
1.2.1	Relevância econômica	14
1.2.2	Relevância acadêmica	14
1.2.3	Justificativa empresarial.....	15
2	REFERENCIAL TEÓRICO.....	16
2.1	GESTÃO DE ESTOQUES.....	16
2.2	MANUTENÇÃO, REPARO E OPERAÇÃO (MRO).....	18
2.3	CLASSIFICAÇÃO ABC	20
2.4	APRENDIZADO DE MÁQUINA.....	22
2.4.1	Aprendizado de máquina na classificação ABC.....	24
2.5	DRSA	25
3	METODOLOGIA	28
3.1	CLASSIFICAÇÃO DA PESQUISA.....	28
3.2	OBJETO DO ESTUDO DE CASO.....	28
3.3	ETAPAS DA PESQUISA	29
3.3.1	Coleta e geração dos dados	30
3.3.2	Desenvolvimento dos modelos de aprendizado de máquina.....	30
3.3.3	Análise comparativa entre classificações.....	31
3.3.4	Seleção do modelo	32
3.3.5	Aprendizado semi-supervisionado	32
3.3.6	Análise comparativa do resultado	33
3.4	LIMITAÇÕES DA METODOLOGIA	33
4	RESULTADOS E DISCUSSÕES	34

4.1	ANÁLISE DE CONSISTÊNCIA DAS CLASSIFICAÇÕES	34
4.2	ANÁLISE DOS MODELOS NA CLASSIFICAÇÃO ATUAL.....	35
4.3	ANÁLISE COMPARATIVA E SELEÇÃO DO MODELO	38
4.4	RESULTADOS DA CLASSIFICAÇÃO SEMI-SUPERVISIONADA.....	41
5	CONCLUSÃO	44
	REFERÊNCIAS.....	46
	APÊNDICE A – CÓDIGO DOS TESTES E ESCOLHA DO MODELO	50
	APÊNDICE B – GERAÇÃO DA AMOSTRA PARA O ESPECIALISTA ..	57
	APÊNDICE C – APRENDIZADO SEMI-SUPERVISIONADO	60

1 INTRODUÇÃO

No atual cenário da economia global, as indústrias de grande porte enfrentam o desafio contínuo de otimizar seus processos para garantir sustentabilidade e crescimento. Segundo Ballou (2008), a gestão da cadeia de suprimentos assume um papel estratégico, sendo a administração de estoques uma de suas áreas mais críticas. Uma gestão de estoques eficaz representa um equilíbrio entre garantir a disponibilidade de materiais para não interromper as operações e minimizar o capital imobilizado, que poderia ser alocado em investimentos mais estratégicos.

Dentre as diversas categorias de inventário, os materiais de Manutenção, Reparo e Operações (MRO) apresentam um desafio particular. Diferentemente da matéria-prima, os itens de MRO não compõem o produto final, mas são vitais para a continuidade e a confiabilidade do processo produtivo. A falta de um componente de MRO, por mais barato que seja, pode resultar na paralisação de equipamentos e linhas de produção, gerando prejuízos financeiros significativos (Santos, 2015).

Para classificar e priorizar a gestão desses materiais, a ferramenta mais difundida é a Curva ABC, um método baseado no Princípio de Pareto que segmenta os itens com base em seu valor de consumo anual (Slack; Brandon-Jones; Johnston, 2018).

Apesar de sua ampla utilização, a abordagem tradicional da classificação ABC revela-se frequentemente inadequada para o universo MRO. Ao utilizar como único critério o valor financeiro, o método falha em reconhecer a criticidade operacional dos itens, podendo classificar peças de baixo custo, porém essenciais, como de baixa prioridade (Classe C), enquanto itens caros, mas de menor impacto operacional, podem ser tratados com rigor excessivo (Classe A) (Slack; Brandon-Jones; Johnston, 2018). Essa distorção pode induzir a uma alocação ineficiente de recursos e, mais gravemente, a um risco operacional elevado e mal gerenciado. Por isso, que as tecnologias da Indústria 4.0, como o aprendizado de máquina (*machine learning*), surgem como uma alternativa promissora para modernizar práticas consolidadas (Kaminski Sanches, 2003).

Na empresa estudada, atualmente o sistema de classificação dos materiais é feito pelo setor de manutenção, sendo de uma maneira manual e subjetiva, portanto, sendo um método mais suscetível a inconsistências em suas classificações.

Dominance-based Rough Set Approach (DRSA) surge como uma extensão da Rough Set Theory (RST) para superar uma de suas principais limitações: a incapacidade de lidar com atributos e classes de decisão que possuem uma ordem de preferência. A inovação central do DRSA consiste na substituição da relação de indiscernibilidade da RST clássica por uma relação de dominância, a qual se baseia no princípio de que um objeto com avaliações iguais ou superiores em todos os critérios não pode ser alocado em uma classe de decisão inferior à de um objeto que ele domina (Slowinski; Greco; Matarazzo, 2012).

Sendo assim, o DRSA será utilizado para medir a consistência da classificação atual da empresa e posteriormente no resultado da classificação com aprendizado de máquina.

Nesse sentido, o presente trabalho é norteado pelo seguinte problema de pesquisa: De que maneira o uso de múltiplas dimensões, por meio de algoritmos de aprendizado de máquina, gera uma classificação ABC alinhada aos objetivos operacionais de uma indústria de grande porte, quando comparada ao método tradicional baseado unicamente no valor de consumo anual?

O trabalho está estruturado em capítulos que detalham desde a fundamentação teórica sobre gestão de estoques e aprendizado de máquina, passando pela metodologia aplicada, até a apresentação, análise dos resultados e as considerações finais, que consolidam as contribuições do estudo.

1.1 OBJETIVO GERAL

Para guiar este trabalho de forma estruturada, foi estabelecido como objetivo geral: avaliar o uso do de aprendizado de máquina semi-supervisionado como suporte para classificação ABC de materiais MRO de uma empresa de grande porte.

1.1.1 Objetivos específicos

Portanto, foram delineados os seguintes objetivos específicos:

- Identificar e analisar o processo de gestão de estoque de MRO na indústria estudada e realizar o levantamento dos dados necessários para a análise;
- Analisar a consistência do modelo de classificação atual utilizado na empresa;
- Desenvolver e aplicar um modelo de aprendizado de máquina, utilizando algoritmos de classificação, para segmentar os itens de estoque MRO com base em múltiplas dimensões;
- Analisar comparativamente os resultados das duas classificações (tradicional vs. aprendizado de máquina).

1.2 JUSTIFICATIVA

A busca por uma resposta a essa questão se justifica por sua alta relevância nos cenários econômico e acadêmico, abordando um problema clássico da Engenharia de Produção sob uma ótica inovadora e alinhada às transformações da Indústria 4.0 (Tessarini; Saltorato, 2018).

1.2.1 Relevância econômica

A relevância econômica deste estudo é acentuada pelo cenário de alta competitividade e pela constante pressão por redução de custos que as indústrias de grande porte enfrentam. A gestão de estoques de materiais de MRO representa um desafio crítico nesse contexto (Vieira; Loures, 2016).

1.2.2 Relevância acadêmica

No campo da Engenharia de Produção, a gestão de estoques é uma área de estudo consolidada. No entanto, a literatura acadêmica aponta consistentemente as

limitações da classificação ABC tradicional, especialmente sua ineficácia em ambientes complexos como a gestão de MRO (Póvoa; Chaves, 2014). Este trabalho contribui para o avanço do conhecimento ao preencher essa lacuna de forma prática e contemporânea. Slack, Brandon-Jones e Johnston (2018) fazem um comentário crítico, colocando-se na posição do gestor de estoque, percebendo que esse tipo de classificação pode ser enganosa e mal interpretada, pois a regra não refere-se ao custo do estoque, mas ao seu uso.

1.2.3 Justificativa empresarial

No cenário atual da empresa, a classificação ABC dos materiais é realizada predominantemente pelo setor de manutenção, de forma manual, cada colaborador do time de manutenção determina, com base em seu conhecimento e viés, qual seria a classificação ideal para o material usado, seguindo o método tradicional que considera, na maioria das vezes, apenas o critério do valor de consumo anual dos itens. Contudo, a vivência no planejamento revelou inconsistências nessa abordagem.

Dessa forma, considera-se a oportunidade de conectar este problema prático e real com os conhecimentos adquiridos no curso de Engenharia de Produção. A proposta de aplicar o aprendizado de máquina surgiu como uma solução para desenvolver um modelo de classificação com múltiplas dimensões, capaz de analisar simultaneamente diversas variáveis e gerar uma segmentação ABC mais inteligente e estrategicamente alinhada aos objetivos operacionais da empresa. Este trabalho, portanto, nasce da busca por uma solução acadêmica e tecnológica para um desafio concreto e relevante observado no ambiente industrial.

2 REFERENCIAL TEÓRICO

Este capítulo estabelece a fundamentação teórica que sustenta a presente pesquisa. Inicia-se com a contextualização da gestão de estoques como uma função de relevância estratégica para a cadeia de suprimentos. Em seguida, aprofunda-se nas características singulares e desafiadoras dos materiais de MRO, que constituem o objeto central deste estudo. Posteriormente, realiza-se uma análise crítica do método tradicional de classificação ABC. Esta crítica serve como uma transição lógica para a introdução de métodos analíticos avançados, apresentando o aprendizado de máquina e a abordagem DRSA para análise de consistência da classificação ABC em vigor na empresa, como uma solução robusta para o problema identificado.

2.1 GESTÃO DE ESTOQUES

“Estoque é o termo que usamos para descrever a acumulação de materiais, clientes ou informações à medida que fluem através de processos ou redes” (Slack; Brandon-Jones; Johnston, 2018, p. 630). Segundo Ballou (2008, p. 29), “a Logística/Cadeia de Suprimentos é um conjunto de atividades funcionais (transportes, controle de estoques etc.) que se repetem inúmeras vezes”. A gestão da cadeia de suprimentos consolidou-se como um fator estratégico para as empresas modernas, sendo a administração de estoques uma de suas áreas mais críticas e de maior impacto no desempenho organizacional (Faria; Carvalho; Rossi, 2008).

Segundo Slack, Brandon-Jones e Johnston (2018) longe de ser uma atividade meramente operacional, a gestão de estoques representa uma alavanca estratégica que influencia diretamente a saúde financeira, a eficiência operacional e a capacidade competitiva de uma organização. Por isso, a natureza estratégica é evidenciada pelas funções primordiais que os estoques desempenham em um sistema produtivo.

Nesse sentido, os estoques têm como primeira função o aumento do nível de serviço oferecido aos clientes a partir do atendimento imediato às suas demandas, transmitindo confiança e uma imagem positiva no mercado (BALLOU, 2008). Além desse objetivo, é possível citar como outro grande fator a redução de custos, sem que haja impacto no nível de serviço, de maneira a promover uma economia de escala na

produção dos bens, nas compras e transportes, amortizando efeitos das variações nas operações (Campos; Rezende, 2013).

Figura 1 - Função básica dos estoques.



Fonte: Campos e Rezende (2013).

Peixoto e Pinto (2006) articulam este dilema de forma explícita, apresentando-o como um problema de otimização onde se deve buscar um ponto de equilíbrio que minimize a soma das curvas de custo e de serviço. Para Wanke (2011) embora focada em um aspecto específico, parte da mesma premissa. Ao argumentar sobre os benefícios de um alto nível de serviço, o autor implicitamente reconhece que esta é uma escolha estratégica que envolve custos, reforçando a ideia de que o nível de serviço é uma variável crítica na equação da gestão de estoques, e não um mero resultado operacional.

A gestão de estoques é uma atividade complexa e integrada. Para Cabral Filho (2023) o planejamento de estoques deve ser realizado "em conjunto com os planejamentos de transportes e armazéns". Reforçando a ideia de que a gestão de estoques e todas as outras atividades logísticas devem estar adequadamente coordenadas (Faria; Carvalho; Rossi, 2008).

A gestão de estoques em larga escala, é caracterizada por uma vasta quantidade de itens, fornecedores e clientes, apresenta um ambiente operacional de elevada complexidade e dinamismo. Para Slack, Brandon-Jones e Johnston (2018), o gerenciamento eficaz de tal cenário exige a adoção de duas estratégias gerenciais fundamentais. A primeira consiste na diferenciação dos itens em estoque com base em sua importância relativa, permitindo a aplicação de um grau de controle

proporcional a cada um. A segunda estratégia é o investimento em sistemas de informação robustos, capazes de processar e gerenciar o grande volume de dados e as circunstâncias particulares do controle de inventário da organização.

2.2 MANUTENÇÃO, REPARO E OPERAÇÃO (MRO)

Os estoques de peças de reposição têm como função básica fazer com que a equipe de manutenção consiga manter os equipamentos em condição de funcionamento e produzindo (Kennedy; Patterson; Fredendall, 2002).

Segundo Póvoa e Chaves (2014) os materiais de MRO necessitam de atenção especial, pois tem um nível de complexidade e peculiaridade particular. Campos e Rezende (2013) reforça essa ideia ao declarar que as peças de reposição requerem um tratamento de gestão e planejamento diferenciado. Ele cria um catálogo de atributos desafiadores: consumo irregular, longos tempos de ressuprimento, baixo giro, alto custo unitário e criticidade operacional. É uma visão pragmática, focada nos sintomas e nas dificuldades enfrentadas pelos gestores.

Kennedy, Patterson e Fredendall (2002) aprofundam essa noção, listando aspectos como a dependência entre falhas de peças, a possibilidade de canibalização para atender à demanda e o grande desafio de quantificar os custos de falta de estoque, que podem incluir perda de produção e riscos à segurança.

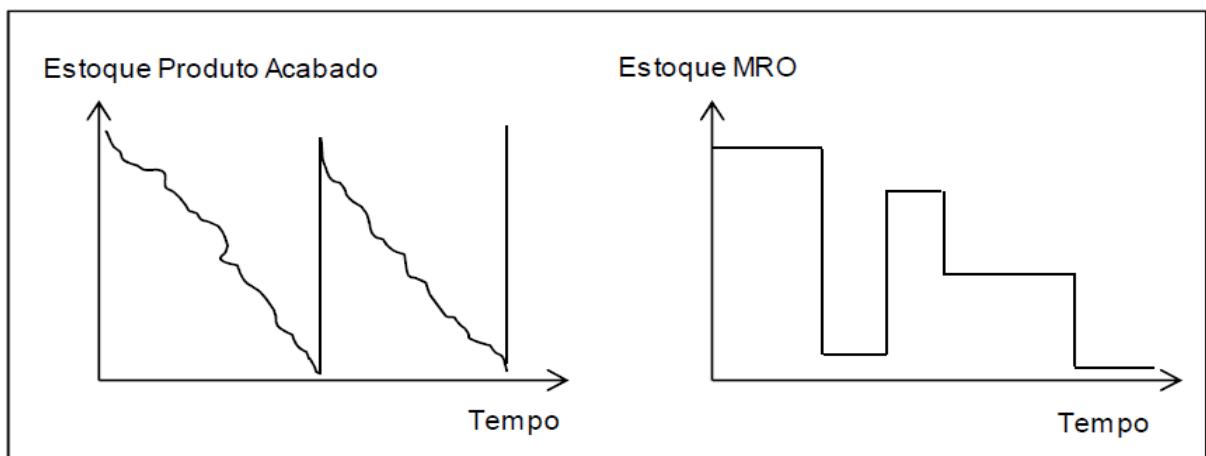
A função dos estoques de peças de reposição é intrinsecamente distinta, visto que seu propósito não é a comercialização para um cliente final, mas sim o de prover suporte à equipe de manutenção para manter os equipamentos em condição de uso operacional. Por isso, as políticas que regem esses inventários diferem daquelas aplicadas a outros estoques. Seus níveis são largamente determinados pelas estratégias de uso e manutenção dos ativos, e não por cronogramas de produção (Kennedy; Patterson; Fredendall, 2002). Decisões como reparar ou substituir um componente, por exemplo, impactam diretamente a demanda por peças.

O comportamento do consumo entre produtos acabados e materiais de MRO justifica a necessidade de políticas de estoque distintas para cada categoria. A demanda por produtos acabados é, em geral, externa e contínua, impulsionada por

previsões de vendas e pedidos de clientes externos, o que permite a aplicação de modelos estatísticos de previsão com maior assertividade (Faria; Carvalho; Rossi, 2008).

Portanto, a demanda por itens de MRO é interna, ou seja, gerada pela necessidade interna de manter a capacidade produtiva das máquinas. Essa distinção é crucial, pois a natureza desse consumo é fundamentalmente diferente, com consumo esporádico e eventos independentes, podendo ser influenciado pelo plano de manutenção. (Campos; Rezende, 2013). Quando o setor de manutenção realiza o plano, isso reduz os riscos e melhora a previsão.

Figura 2 - Comportamento do estoque de Produtos Acabados X Comportamento de estoque MRO.



Fonte: Póvoa e Chaves (2014).

A gestão de peças de reposição para equipamentos em processo de obsolescência representa um desafio de bastante complexo na gestão de estoques de MRO (Kennedy; Patterson; Fredendall, 2002). Para Campos e Rezende (2013), e Slack, Brandon-Jones e Johnston (2018) essa complexidade emerge de um dilema estratégico: a necessidade de balancear o risco de paradas prolongadas de um equipamento crítico, caso uma peça não esteja disponível, contra o risco financeiro de que o item em estoque se torne obsoleto antes mesmo de ser utilizado.

Vieira e Loures (2016) corroboram e expandem essa visão no contexto da indústria aeronáutica, descrevendo o MRO como um processo complexo, pois tem

requisitos por parte das áreas de manutenção, considerados rigorosos e precisos. Desse modo os serviços prestados no mercado de MRO, por parte das fabricantes, ainda é relativamente inexplorado.

Um exemplo disso é quando uma máquina do setor produtivo está com manutenção programada, no manual técnico possuem as peças de reposição e indicam as marcas específicas para serem usadas. Essa diversidade é mostrada por Vieira e Loures (2016) da seguinte forma:

- Barato vs caro;
- Sem movimentação ou com movimentação lenta vs com movimentação alta;
- Reparável vs consumível;
- Crítico vs não crítico;
- Padrão vs específico;
- Prazos de entrega dos fornecedores.

Dito isso, Vieira e Loures (2016) referem-se a esses desafios em um contexto estratégico e sistêmico, demonstrando como as características listadas por Kennedy, Patterson e Fredendall (2002) são amplificadas por fatores como uma cadeia de suprimentos restrita, tais características sendo: políticas de manutenção, informações de vida útil da peça, falhas de peças que são dependentes entre si, obsolescência e entre outros. Juntos, os autores justificam a necessidade de modelos de gestão avançados, pois mostram que o problema do MRO não é apenas uma questão de gerenciar itens com demanda irregular, mas de navegar em um complexo ambiente de negócios com barreiras de entrada, custos elevados e dinâmicas competitivas próprias.

2.3 CLASSIFICAÇÃO ABC

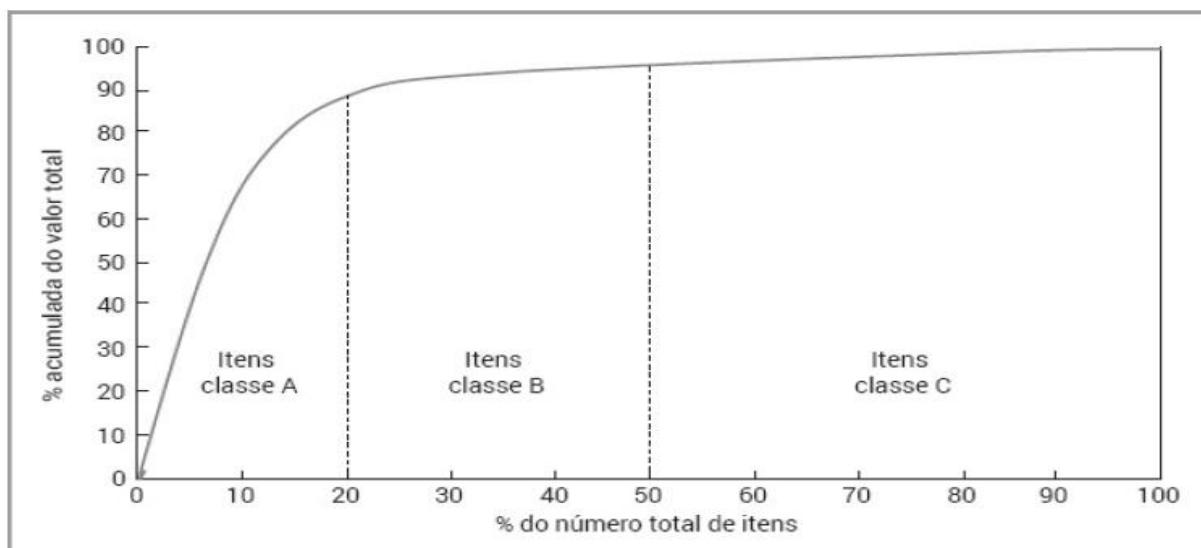
Em inventários com múltiplos itens, é natural que alguns possuam maior relevância para a organização do que outros. Para Slack, Brandon-Jones e Johnston (2018), essa relevância pode derivar de fatores como a alta frequência de demanda, onde uma ruptura impactaria muitos clientes, ou o elevado valor unitário, que tornaria

difícil manter níveis de estoque excessivos. Uma metodologia comum para a priorização de itens consiste em classificá-los com base no seu valor de movimentação, um indicador obtido pela multiplicação da taxa de consumo pelo seu valor individual.

Segundo Cabral Filho (2023) e Camargo Izahc (2020), é a partir dessa classificação que os itens com alto valor de movimentação são submetidos a um controle mais rigoroso, enquanto aqueles de baixo valor, podem ser gerenciados com menor intensidade. Este método de classificação fundamenta-se no Princípio de Pareto, também conhecido como regra 80/20, que postula que uma pequena porcentagem do total de itens (20%) é responsável pela maior parte do valor total movimentado (80%) (Slack; Brandon-Jones; Johnston, 2018). Para Cabral Filho (2023), essa é a regra de classificação com base em seu percentual acumulado do valor total:

- Itens classe A: mais importantes de todos (20%)
- Itens classe B: de importância intermediária (30%)
- Itens classe C: menos importantes (50%)

Figura 3 - Curva de Pareto para itens em um armazém.



Fonte: Slack, Brandon-Jones e Johnston (2018).

2.4 APRENDIZADO DE MÁQUINA

O aprendizado de máquina é um ramo da inteligência artificial que se baseia na ideia de que sistemas podem aprender a partir de dados, identificar padrões complexos e tomar decisões com mínima intervenção humana. Para Kaminski Sanches (2003), o estudo de aprendizado de máquina é essencial para a área de pesquisa de IA, englobando estudos de métodos computacionais, para assim, desenvolver novos métodos para organizar conhecimentos já existentes.

Embora o aprendizado de máquina englobe um campo vasto e diversificado de métodos, um objetivo central para muitas dessas abordagens é a generalização a partir de informações conhecidas. O processo consiste, fundamentalmente, em analisar um conjunto de exemplos ou dados de treinamento para identificar e modelar padrões implícitos (Hartshorn, 2016). Posteriormente, o modelo desenvolvido é utilizado para reconhecer características análogas ou realizar previsões em novos conjuntos de dados cujos resultados são desconhecidos.

As tarefas executadas por algoritmos de aprendizado de máquina podem ser categorizadas em dois tipos: supervisionado e não supervisionado.

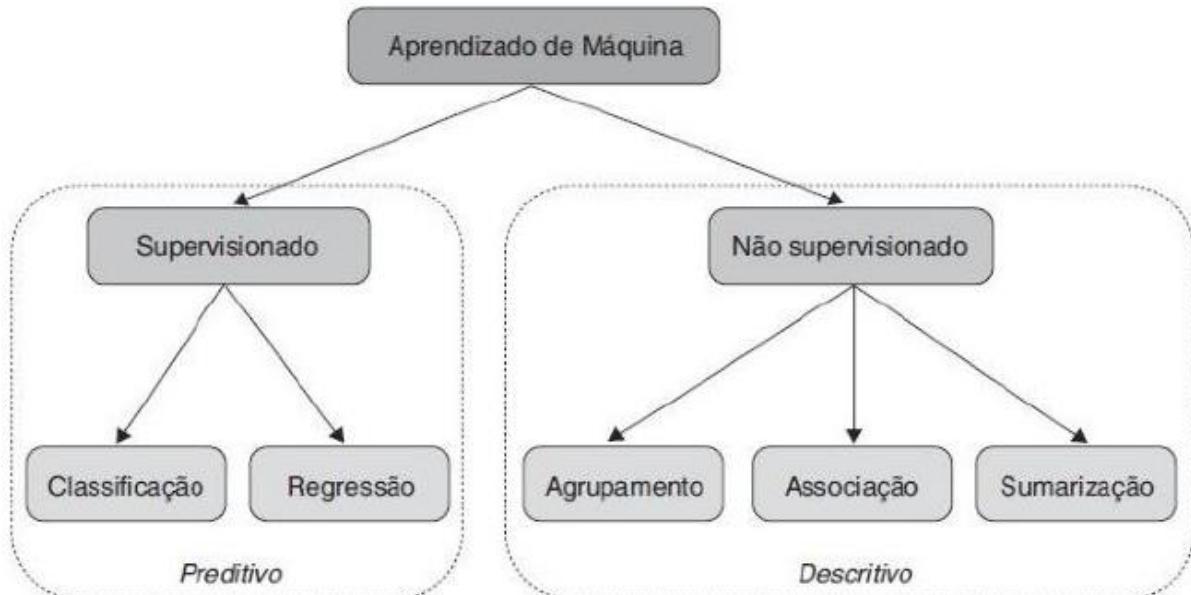
Os algoritmos supervisionados utilizam métodos aplicados a conjuntos de dados previamente rotulados (dados de treinamento) para induzir um modelo com capacidade de predição. O objetivo é estimar o valor de um atributo-alvo para um novo objeto, com base em seus atributos definidos. Um exemplo prático seria a predição do diagnóstico de um paciente a partir de seus sintomas (Gaia, 2024).

Por isso, o termo "supervisionado" refere-se à utilização de rótulos conhecidos (o atributo-alvo), o objetivo fundamental consiste em identificar um modelo ou uma função que estabeleça um mapeamento preciso entre um conjunto de entradas e suas respectivas saídas (Taylor, 2017).

Segundo Gaia (2024), Taylor (2017) e Hartshorn (2016) os algoritmos não supervisionados não visam à predição de um valor específico, mas sim à extração de padrões e estruturas dos dados, operando sem o conhecimento prévio de um rótulo. Entre as principais tarefas descritivas, destacam-se o agrupamento de dados (clustering), cujo objetivo é identificar grupos de objetos com alta similaridade entre si,

e a mineração de regras de associação, que busca identificar relações entre subconjuntos de atributos.

Figura 4 - Hierarquia do aprendizado de máquina.



Fonte: Gaia (2024).

Diante disso, um termo ainda pouco discutido na literatura acadêmica é o de aprendizado semi-supervisionado. Segundo Kaminski Sanches (2003) e Taylor (2017), o aprendizado semissupervisionado (*Semi-supervised learning* - SSL) representa um paradigma híbrido, situado entre o aprendizado supervisionado e o não supervisionado. Essa abordagem utiliza conjuntos de dados que contêm uma combinação de dados rotulados e não rotulados, geralmente com uma pequena fração de exemplos rotulados e um volume substancialmente maior de exemplos não rotuladas. Consequentemente, os algoritmos de SSL são projetados para identificar de forma autônoma as estruturas e os padrões implícitos nos dados, ao mesmo tempo que se utilizam das entradas rotuladas como guia para o refinamento do modelo.

Uma aplicação ilustrativa é a classificação categórica de páginas web, onde a rotulagem manual de milhares de sites é inviável devido a custos e tempo de equipes de trabalho. Em um framework de SSL, o modelo pode ser treinado inicialmente com o grande conjunto de dados não rotulados para aprender as distribuições gerais dos

dados e, posteriormente, passar por um ajuste fino utilizando o pequeno subconjunto rotulado para aprimorar a acurácia da classificação (Taylor, 2017).

2.4.1 Aprendizado de máquina na classificação ABC

Apesar de ser um dos métodos de classificação de estoque mais difundidos, a análise ABC tradicional é alvo de críticas significativas na literatura acadêmica. A principal preocupação é que a abordagem usada se concentra apenas no valor de consumo ao longo do ano, deixando de lado outros fatores importantes para uma gestão eficiente (Lekha *et al.*, 2022).

Nesse sentido, Lekha *et al.* (2022) exploraram a aplicação de técnicas como VIKOR e TOPSIS, demonstrando como abordagens que consideram múltiplos critérios podem ser empregadas na classificação de estoques. Além disso, Ng (2007) propôs um método classificador que evidencia um importante dilema: a utilização de um número reduzido de critérios torna a classificação altamente dependente da ponderação ou ranking atribuído a eles, ao passo que a inclusão de muitos critérios pode sobrecarregar a capacidade de análise do tomador de decisão.

Ferramentas como o Processo Analítico Hierárquico (AHP) foram aplicadas, porém, este método demanda julgamentos implícitos na realização de comparações justas entre os critérios. Segundo Ng (2007), algoritmos genéticos (AG) e redes neurais artificiais (RNA), também foram empregadas para solucionar o problema. No entanto, tais métodos são frequentemente considerados complexos em sua aplicação e de difícil compreensão para os gestores.

Hu *et al.* (2017) propõem um framework de três fases (aprendizado, validação e generalização) para resolver um problema clássico de gestão de operações: a classificação de peças, utilizando 7 tipos diferentes de algoritmos de aprendizado de máquina. Sendo assim, comparou-se o resultado dos algoritmos, medindo o nível de acuracidade de precisão dos resultados, alguns dos algoritmos propostos desse estudo estão alinhados com o estudo dos autores.

Dito isso, o resultado do seu estudo indica que o DRSA apresenta desempenho superior ao de todos os outros métodos em todas as métricas estatísticas. Embora os resultados dos métodos rede perceptron multicamadas (*Multilayer perceptron neural*

network - MLPNN) e regra de classificação difusa também sejam consideravelmente elevados, estes apresentam uma limitação significativa: não funcionam como ferramentas interativas para o tomador de decisão, uma vez que não permitem a sugestão ou alteração de mudanças de forma fácil.

2.5 DRSA

A abordagem DRSA surge como uma metodologia relativamente sofisticada e adequada para o problema em questão. DRSA é uma extensão da Teoria dos Conjuntos Aproximados (RST), proposta originalmente por Pawlak. A RST clássica é uma ferramenta poderosa para lidar com a incerteza e a imprecisão em dados, ela trabalha com aproximações inferior e superior, usando uma relação de indiscernibilidade. O objetivo da DRSA é substituir essa relação por uma relação de dominância, tornando a análise mais prática e clara (Slowinski; Greco; Matarazzo, 2012).

Desse modo, RST clássica é incapaz de lidar com atributos e classes que possuem uma ordem de preferência, como é o caso da classificação ABC.

Slowinski, Greco e Matarazzo (2012) corroboraram para o princípio de dominância, definindo no contexto de classificação, afirmando que um objeto não pode ser designado a uma classe de decisão pior que a de outro objeto que ele domina. Segundo Couto e Gomes (2016, p. 133) "objetos com uma avaliação melhor ou pelo menos igual não podem ser associados a uma classe de decisão pior".

O método DRSA é uma ótima maneira de identificar os problemas de classificação existentes em uma organização. Hu *et al.* (2017) reconhecem a capacidade do DRSA de analisar dados que contêm inconsistências (ou seja, exemplos que violam o princípio da dominância) sem a necessidade de pré-processamento para removê-las, o que é uma vantagem significativa em cenários do mundo real.

A abordagem DRSA inicia-se com uma estrutura de dados conhecida como tabela de decisão. Matematicamente, ela é definida como uma tupla. Błaszczyński *et al.* (2011) representa matematicamente por $S=(U,Q,V,f)$, onde U é um universo finito

de objetos (ex: itens de estoque), Q é um conjunto finito de atributos (divididos em atributos condicionais C, ou critérios, e um atributo de decisão D), V é o domínio dos valores que os atributos podem assumir, e f é uma função de informação que atribui um valor a cada objeto para cada atributo.

O mecanismo central do DRSA é a relação de P-dominância, denotada por $xDPy$, que ocorre se um objeto x for pelo menos tão bom quanto um objeto y em todos os critérios pertencentes a um conjunto $P \subseteq C$ (Couto; Gomes, 2016). A partir dessa relação, são definidos os "grânulos de conhecimento", chamados de cones de dominância, definido por (Błaszczyński *et al.*, 2011):

- Conjunto P-dominante: $D_p^+(x) = \{ y \in U \mid y DPx \}$, que é o conjunto de todos os objetos que dominam x.
- Conjunto P-dominado: $D_p^-(x) = \{ y \in U \mid x DPy \}$, que é o conjunto de todos os objetos dominados por x.

Sendo assim, esses grânulos são então utilizados para calcular as aproximações inferior e superior das uniões de classes. No contexto do DRSA, duas métricas principais são utilizadas para avaliar a consistência e a precisão de uma classificação multicritério.

Segundo Błaszczyński *et al.* (2011), a qualidade da aproximação (representada pela letra grega gama, γ) de uma partição de classes Cl, por meio de um conjunto de critérios P, é definida como a razão entre o número de objetos classificados corretamente pelo conjunto P e o número total de objetos no sistema. Objetos corretamente classificados são aqueles que não pertencem a nenhuma região de fronteira, ou seja, que são consistentes com o princípio da dominância. Matematicamente, é expressa como:

$$\gamma_P(CL) = \frac{|U - ((U_{t \in T} Bnp(Cl_t^{\geq}) \cup U_{t \in T} Bnp(Cl_t^{\leq})))|}{|U|} \quad (1)$$

Adicionalmente, a acurácia da aproximação (representada pela letra grega alfa, α) de uma união de classes específica é calculada como a razão entre o número de

objetos na aproximação inferior e o número de objetos na aproximação superior (Hu *et al.*, 2017). A fórmula é dada por:

$$\alpha p(Cl_t^o) = \frac{|\underline{P}(Cl_t^o)|}{|\bar{P}(Cl_t^o)|} \text{ onde } o \in \{\geq, \leq\} \quad (2)$$

O valor de α e γ varia de 0 a 1. Um valor igual a 1 indica uma aproximação perfeita, onde a região de fronteira é vazia (a aproximação inferior e a superior são iguais). Por outro lado, um valor igual a 0 ocorre quando a aproximação inferior é vazia, indicando ausência de conhecimento certo sobre aquela união de classes.

3 METODOLOGIA

Este capítulo descreve os procedimentos metodológicos adotados para o desenvolvimento do presente trabalho. A estrutura aborda a classificação da pesquisa, o objeto do estudo de caso, as etapas sequenciais de coleta e análise de dados, e, por fim, as limitações inerentes ao método empregado.

3.1 CLASSIFICAÇÃO DA PESQUISA

A presente pesquisa classifica-se quanto à sua natureza como aplicada, pois visa à solução de um problema prático e específico dentro de um contexto organizacional. A abordagem adotada foi quantitativa, uma vez que se baseou na coleta e análise de dados numéricos para a modelagem e comparação de sistemas de classificação de estoque.

Quanto aos objetivos, a pesquisa tem um caráter exploratório, buscando investigar as potencialidades da aplicação de técnicas de aprendizado de máquina em um problema tradicionalmente abordado por métodos subjetivos. O esboço técnico da pesquisa foi o estudo de caso, método que permitiu uma investigação aprofundada da classificação ABC para itens de estoque de MRO em uma única empresa, viabilizando uma análise detalhada e contextualizada do cenário.

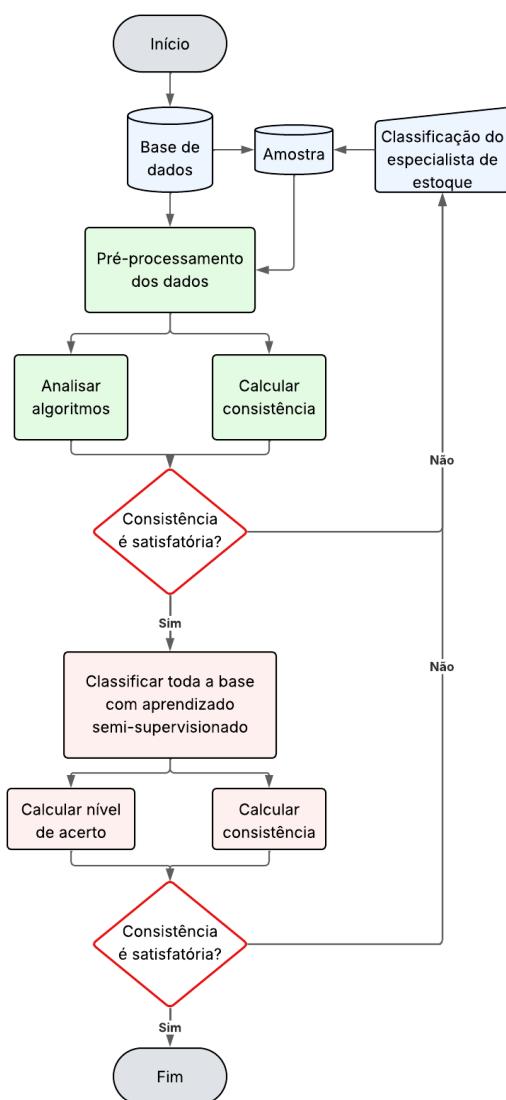
3.2 OBJETO DO ESTUDO DE CASO

O estudo foi conduzido em uma indústria de grande porte do setor metalúrgico, cujo nome será mantido em sigilo para preservar a confidencialidade das informações. A organização gerencia um inventário com mais de 9.000 SKUs (*Stock Keeping Units*) de materiais de MRO. O contexto que motivou este trabalho foi a constatação de que o método de classificação de estoque atual, baseado unicamente na percepção de importância do setor de manutenção, resultava em um controle ineficiente, gerando rupturas de estoque de itens críticos para a operação.

3.3 ETAPAS DA PESQUISA

O desenvolvimento do estudo seguiu um roteiro metodológico estruturado nas etapas descritas a seguir e na figura 5.

Figura 5 - Fluxograma metodológico.



Fonte: O autor (2025).

3.3.1 Coleta e geração dos dados

A caracterização do problema e a definição das variáveis relevantes foram obtidas por meio de entrevistas informais com o gestor do almoxarifado e seus especialistas. Para preservar a confidencialidade dos dados da empresa, optou-se pela geração de um conjunto de dados sintético por meio de Inteligência Artificial, que simula as características e a distribuição do inventário real.

Foram gerados dados que representam um histórico de 24 meses de consumo, compreendendo o período de janeiro de 2023 a dezembro de 2024. As variáveis consideradas no conjunto de dados foram: família do material, consumo médio anual e preço médio, juntamente com suas respectivas classes, totalizando 6530 materiais gerados para a base de dados.

A figura 6 mostra as primeiras linhas da base de dados e como os dados estão estruturados.

Figura 6 - Base de dados do estudo.

	Familia	Consumo Médio Anual	Preço Médio	ABC
0	MAT ELETRICO	12.333333	4.9533	A
1	VEDACAO	584.666667	8.2675	B
2	PARAFUSOS	59.666667	3.0400	C
3	MAT MECANICO	1.000000	26.4200	B
4	LUBRIFICANTES	56.666667	84.2033	A
...

Fonte: O autor (2025).

3.3.2 Desenvolvimento dos modelos de aprendizado de máquina

Esta etapa constituiu o núcleo da pesquisa e envolveu o desenvolvimento e a comparação de diferentes modelos preditivos.

- **Ferramentas e Ambiente:** O desenvolvimento foi realizado na linguagem Python 3, no formato de notebook do Jupyter, no ambiente de desenvolvimento Microsoft VS Code nas dependências do Anaconda, com o suporte das bibliotecas Pandas e Numpy para a manipulação e o tratamento dos dados, Scikit-learn para a implementação dos algoritmos, Matplotlib e Plotly para a geração de visualizações gráficas, e Yellowbrick para a criação da matriz de confusão. A biblioteca Pickle foi utilizada para salvar os modelos treinados. Adicionalmente, o software jMAF foi empregado para analisar a consistência da classificação de referência. Também foi utilizado o software Excel, para as classificações manuais e manipulação dos dados.
- **Análise de consistência:** A base de dados foi adaptada para ser interpretada pelo software Jmaf (Błaszczyński *et al.*, 2011). Após a adaptação, foi inserido os dados no software para calcular a consistência da classificação atual.
- **Pré-processamento dos Dados:** Antes do treinamento, os dados passaram por um pré-processamento. Os atributos numéricos (como custo e consumo) foram normalizados pela técnica de padronização, para que tivessem média zero e desvio padrão um, evitando que critérios com ordens de grandeza distintas dominassem os modelos. Também foi realizado o tratamento dos atributos categóricos para convertê-los em um formato numérico adequado.
- **Modelagem e Treinamento:** A base foi segmentada em conjuntos de treinamento e teste. Foram implementados e avaliados cinco algoritmos de classificação supervisionada distintos: Regressão Logística, Random Forest Classifier, K-Nearest Neighbors (KNN), Support Vector Machine (SVM) e Rede Neural. Inicialmente, cada algoritmo foi treinado com seus parâmetros padrão, sua performance foi avaliada por meio da matriz de confusão e do cálculo da acurácia, comparando as previsões com os rótulos do conjunto de teste.

3.3.3 Análise comparativa entre classificações

A princípio aplicou-se os testes dos algoritmos de classificação para os 6530 materiais para a geração das estatísticas dos modelos. Paralelamente foi medido o

nível de consistência da classificação atual dos materiais de MRO, por meio da metodologia do DRSA.

Uma amostra aleatória de 400 materiais da base completa foi retirada para um especialista da área de estoque da empresa classificar. Essa amostra foi gerada por meio de código em python, para garantir que seja uma representação do todo e que tenha todas as famílias dos materiais existentes. Essa nova base passou pelo mesmo processo de testes dos algoritmos de classificação e posteriormente gerado os dados estatísticos dos resultados. Igualmente a base completa, também foi medido o nível de consistência da classificação do especialista pelo software jmaf.

3.3.4 Seleção do modelo

Para identificar o algoritmo com melhor desempenho para o conjunto de dados em estudo, foi realizado um procedimento de validação. Primeiramente, utilizou-se a técnica Grid Search para encontrar a melhor combinação de hiper parâmetros para cada um dos cinco algoritmos.

Posteriormente, com os modelos otimizados, foi executada uma validação cruzada com 30 rodadas de testes, em cada rodada a base de dados foi dividida em dez partes aleatórias, nove para treinamento e uma para teste, vale ressaltar que para cada rodada os valores das amostras mudam. Este procedimento permitiu calcular a média de acuracidade e a variação (desvio padrão) do desempenho de cada modelo. O critério de seleção do melhor algoritmo foi a combinação de uma maior média de acuracidade com a menor variação entre os testes, indicando não apenas um modelo preciso, mas também estável e confiável.

3.3.5 Aprendizado semi-supervisionado

Após a escolha do melhor algoritmo de classificação, os 400 materiais previamente classificado pelo especialista foram substituídos na base original, e os demais rótulos foram apagados, resultando em valores em branco, gerando assim uma base de dados com 400 materiais rotulados e 6130 materiais não rotulados.

Nesse sentido, aplicou-se um algoritmo de classificação semi-supervisionado para ser treinado com as classificações existentes, gerando novos rótulos e repetindo o processo até se enquadrar dentro do parâmetro de acurácia pré-definido no algoritmo. O algoritmo base foi o modelo que apresentou o melhor resultado presente na seção anterior.

3.3.6 Análise comparativa do resultado

Novamente o especialista que classificou os 400 materiais foi abordado para classificar mais materiais, dessa vez materiais que anteriormente ele não classificou, para servir de modelo comparativo entre a percepção de classificação do especialista e o resultado que o modelo de aprendizado de máquina nos trouxe.

Sendo assim, após a segunda classificação, por parte do especialista, foi gerado uma coluna de comparação no Excel, para calcularmos quantos materiais foram classificados como “Iguais” e quantos resultaram como “Diferentes”. Plotou-se em um gráfico de pizza esse indicador, para gerar uma melhor interpretação do resultado.

3.4 LIMITAÇÕES DA METODOLOGIA

Reconhece-se que o presente estudo possui limitações intrínsecas à sua abordagem. Primeiramente, por se tratar de um estudo de caso único, os resultados obtidos e o modelo selecionado podem não ser diretamente generalizáveis para outras empresas, mesmo que do mesmo setor, devido a particularidades operacionais e de inventário. Em segundo lugar, a definição de criticidade dos materiais, que serviu como base para a classificação de referência, foi fundamentada na percepção de um único especialista da área de MRO, introduzindo um grau de subjetividade no rótulo dos dados de treinamento.

4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados obtidos a partir da aplicação da metodologia descrita anteriormente. A estrutura desta seção inicia-se com a análise de consistência das classificações de referência, seguida pela análise comparativa dos algoritmos de aprendizado de máquina e a seleção do modelo de melhor desempenho. Posteriormente, são detalhados os resultados da classificação semi-supervisionada de todo o inventário e, por fim, realiza-se a validação final e a discussão aprofundada das implicações práticas e teóricas dos achados.

4.1 ANÁLISE DE CONSISTÊNCIA DAS CLASSIFICAÇÕES

A primeira etapa quantitativa consistiu em avaliar a consistência interna das classificações e a métrica de Qualidade da Aproximação (γ) do DRSA.

A análise da classificação original da empresa, que era baseada na percepção subjetiva do setor de manutenção, revelou uma γ de apenas 12%, como mostra a figura 7. Este valor extremamente baixo indica muita inconsistência, onde itens com características similares recebiam classificações distintas, validando a premissa inicial de que o método atual é falho e pouco confiável.

Figura 7 - Qualidade da aproximação atual gerada pelo jmaf.

Quality of approximation: 0.121	
Union name	Accuracy
> At most C	0.085
> At least B	0.156
> At most B	0.322
> At least A	0.047

Fonte: O autor (2025).

Em contrapartida, a amostra de 400 itens classificada pelo especialista da área de estoque apresentou uma γ de aproximadamente 75%, como pode ser observado

na figura 8. Este resultado, significativamente superior, demonstra um padrão de decisão muito mais coerente e lógico, tornando esta amostra uma base de referência robusta e confiável para o treinamento e a validação dos modelos de aprendizado de máquina. A discrepância entre os dois valores reforça a necessidade de um novo método de classificação, mais estruturado e consistente.

Figura 8 - Qualidade da aproximação do especialista gerado pelo jmaf.

Quality of approximation: 0.747	
Union name	Accuracy
At most C	0.713
At least B	0.819
At most B	0.769
At least A	0.699

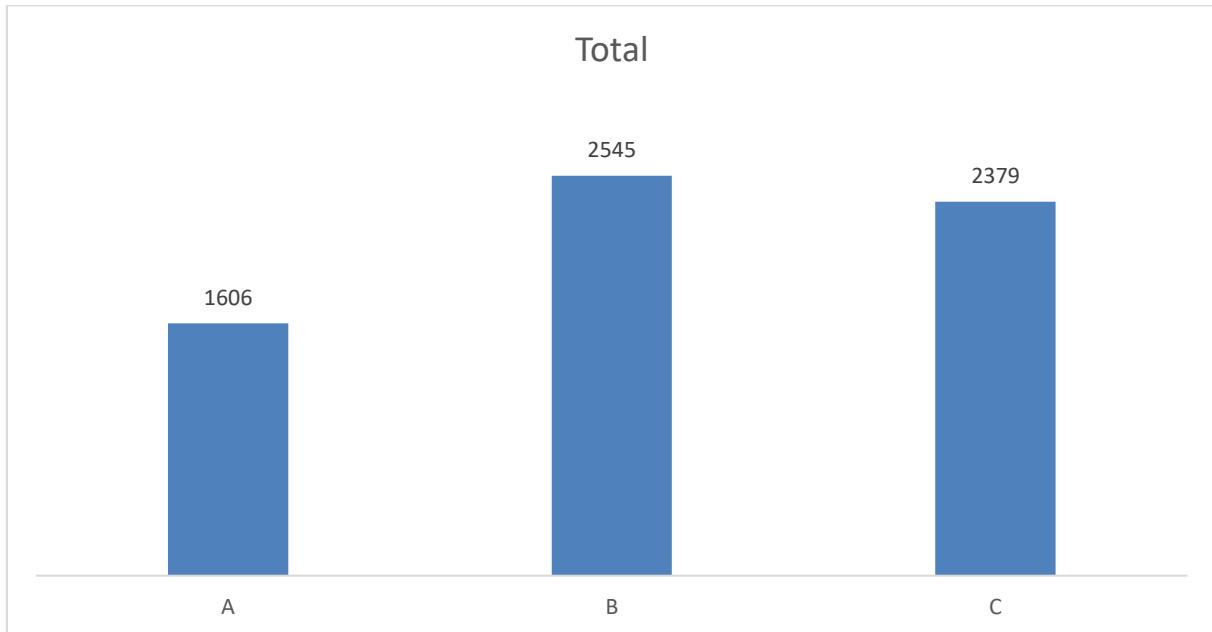
Fonte: O autor (2025).

Essa diferença nos resultados na qualidade da aproximação entre classificação atual versus classificação do especialista, mostra uma maior concordância com as variáveis que são escolhidas para determinar a classe do material. Sendo assim, vale ressaltar que um maior γ não garante que a classificação seja considerada a “correta”, mas sim que existe uma maior concordância entre variáveis e classe escolhida.

4.2 ANÁLISE DOS MODELOS NA CLASSIFICAÇÃO ATUAL

Na base de dados inicial, contou-se o número de materiais que estavam classificados em cada uma das classes ABC. Como mostra na figura 9, nota-se uma quantidade de materiais mais classificados como B, posteriormente como C e por final como A.

Figura 9 - Classificação atual.



Fonte: O autor (2025).

Nessa fase inicial, os melhores parâmetros encontrados pelo método de otimização gread search se encontram na tabela 1.

Tabela 1 - Parâmetros dos modelos para a base inicial.

Algoritmo			
Regressão logística	C = 1.5	Solver = 'lbfgs'	tol = 0.0001
KNN	n_neighb ors = 25	p=2	
Rede Neural	Activation = 'relu'	batch_size = 56	solver = 'sgd'
Random Forest	criterion = 'gini'	min_samples_lea f = 1	min_samples_split=1 0
SVM	kernel = 'poly'	C = 2.0	tol= 0.001

Fonte: O autor (2025).

Após a escolha dos melhores parâmetros para a base de dados, um teste dos algoritmos de classificação foi feito, para medir o nível de acurácia dos algoritmos e como comportaram-se. Os 30 testes foram gerados e os resultados se encontram na tabela 2.

Tabela 2 - Resultado dos modelos na base inicial.

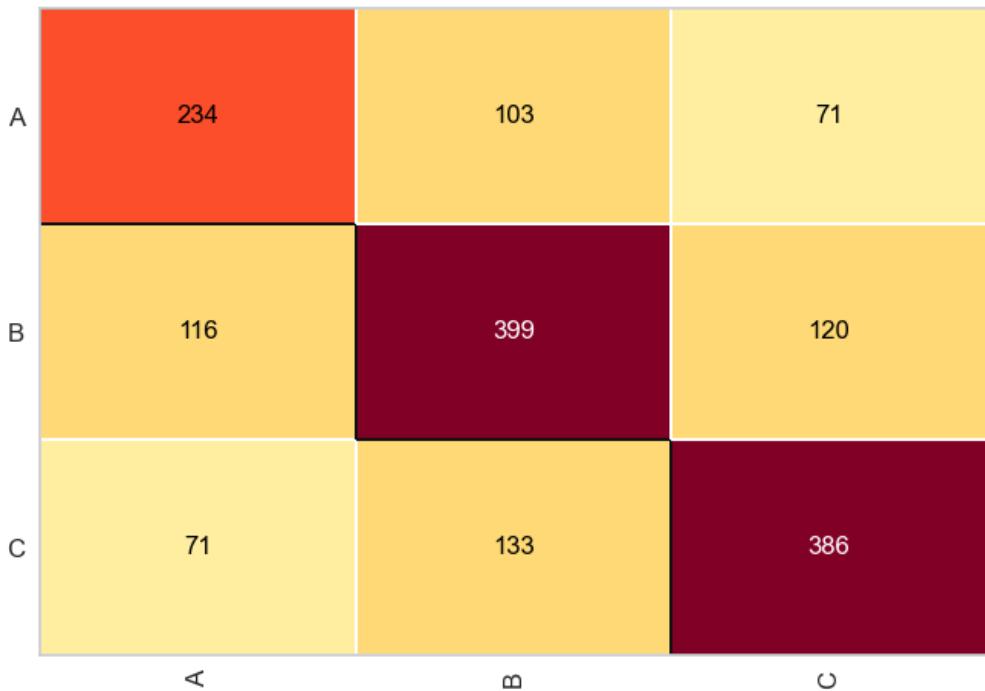
Algoritmo	Média da acurácia	Desvio padrão da acurácia
Regressão logística	60,18%	0,002
KNN	63,66%	0,002
Rede Neural	63,82%	0,002
Random Forest	64,42%	0,002
SVM	61,02%	0,001

Fonte: O autor (2025).

Um exemplo do nível de acurácia do melhor modelo para a primeira rodada de teste é observado pela matriz de confusão de sua classificação. A matriz se dá de forma visual pelo número de acertos do modelo. A figura 9 mostra a matriz de confusão do algoritmo random forest, no qual foi o modelo que teve melhor média de acurácia e um desvio padrão não tão diferente dos demais.

Na matriz de confusão da figura 9, o eixo X representa os valores reais das classes, enquanto o eixo Y representa os valores que o algoritmo classificou, com base na fase de testes e treinamento.

Figura 10 - Matriz de confusão do random forest na base inicial.



Fonte: O autor (2025).

Observa-se que o nível de precisão do algoritmo é baixo, podendo causar falhas nas decisões de classificação dos materiais para estoque.

4.3 ANÁLISE COMPARATIVA E SELEÇÃO DO MODELO

Para identificar o algoritmo mais adequado para o problema, cinco modelos de classificação supervisionada foram treinados e avaliados utilizando a amostra de 400 itens classificada pelo especialista.

Após essa classificação por parte do especialista, o mesmo código foi testado, incluindo os ajustes dos parâmetros pelo gread search, os novos parâmetros para cada modelo estão representados na tabela 3.

Tabela 3 - Parâmetros dos modelos para a base de dados do especialista.

Algoritmo				
Regressão logística	C = 2.0	solver = 'saga'	tol = 0.000001	
KNN	n_neighb ors = 5	p=2		
Rede Neural	Activation = 'relu'	batch_size = 10	solver = 'adam'	
Random Forest	criterion = 'gini'	min_samples_lea f = 1	min_samples_split=5	n_estimators s = 100
SVM	kernel = 'rbf'	C = 1.0	tol= 0.0001	

Fonte: O autor (2025).

Quando comparado a tabela 1 com a tabela 3, a medida que a consistência dos dados mudou, os seus parâmetros também sofreram alterações, isso nos mostra que não existe uma configuração de parâmetros universal para cada modelo, mas sim que eles devem se adequar a base de dados.

A performance e a estabilidade de cada modelo foram mensuradas por meio de uma validação cruzada com 30 testes. Os resultados consolidados, incluindo a média e o desvio padrão da acurácia, são apresentados na Tabela 4.

Tabela 4 - Resultado dos modelos na base do especialista.

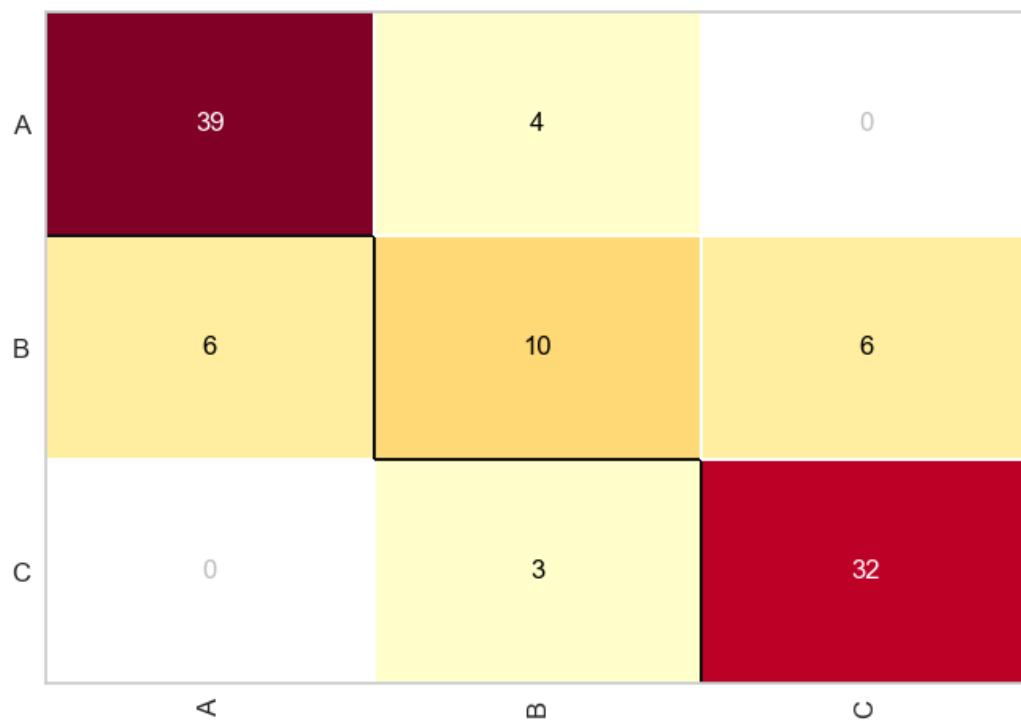
Algoritmo	Média da acurácia	Desvio padrão da acurácia
Regressão logística	79,25%	0,0108
KNN	76,5%	0,0109
Rede Neural	78,22%	0,0104
Random forest	77,7%	0,0105
SVM	80,8%	0,008

Fonte: O autor (2025).

A análise da Tabela 4 indica que o algoritmo SVM foi o modelo de melhor desempenho. Ele não apenas alcançou a maior média de acurácia (80,8%), mas também apresentou o menor desvio padrão (0,008). Essa combinação é ideal, pois sugere que o SVM é o modelo não apenas mais preciso, mas também o mais estável e confiável para este conjunto de dados, sendo, portanto, o escolhido para a etapa subsequente de classificação.

Dito isso, a matriz de confusão do modelo de rede neural, feito de forma manual antes dos ajustes dos parâmetros, mostra de forma visual o nível de acerto de sua classificação. Como se pode observar na figura 11.

Figura 11 - Matriz de confusão do modelo de rede neural para a base do especialista.



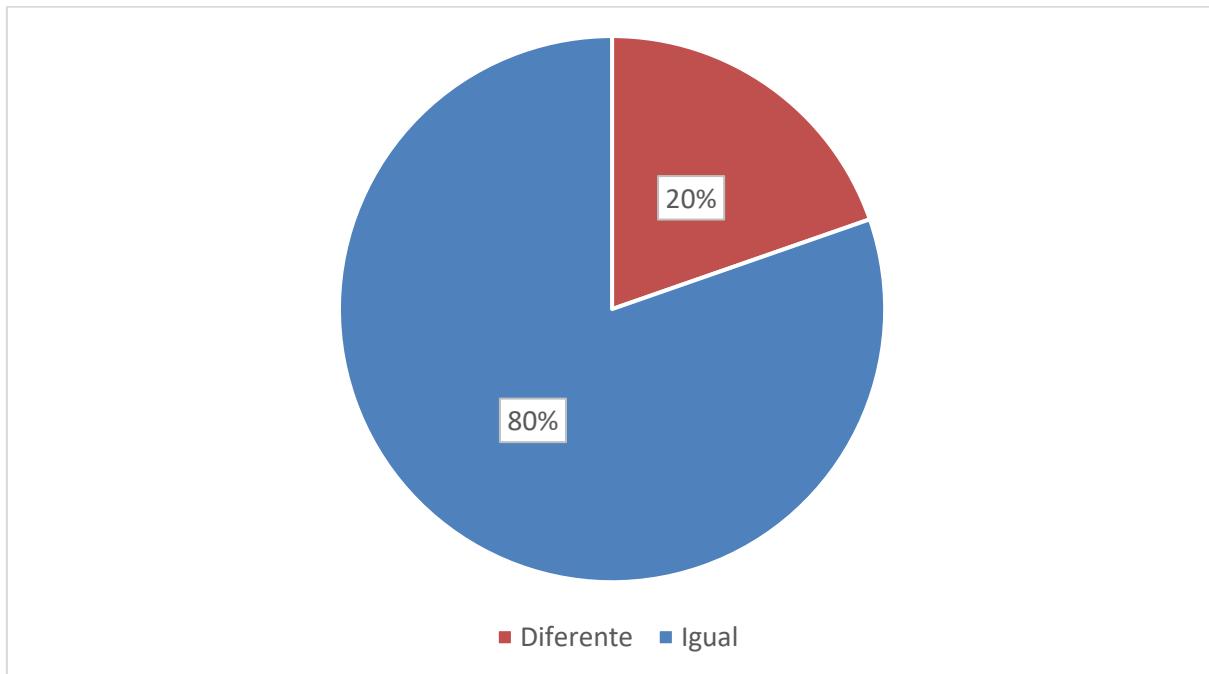
Fonte: O autor (2025).

4.4 RESULTADOS DA CLASSIFICAÇÃO SEMI-SUPERVISIONADA

Sendo assim, com o modelo SVM otimizado e selecionado, procedeu-se à classificação dos 6.130 itens restantes da base de dados, utilizando uma abordagem de aprendizado semi-supervisionado.

Com o modelo aplicado na nova base, foi pedido ao especialista que fizesse uma classificação cega dos materiais que ele ainda não tinha classificado, sendo 239 materiais, para compararmos com o resultado do modelo final. O resultado é mostrado na figura 12, refletindo o nível de acurácia anteriormente medido na etapa da escolha do modelo.

Figura 12 - SVM X Especialista.



Fonte: O autor (2025).

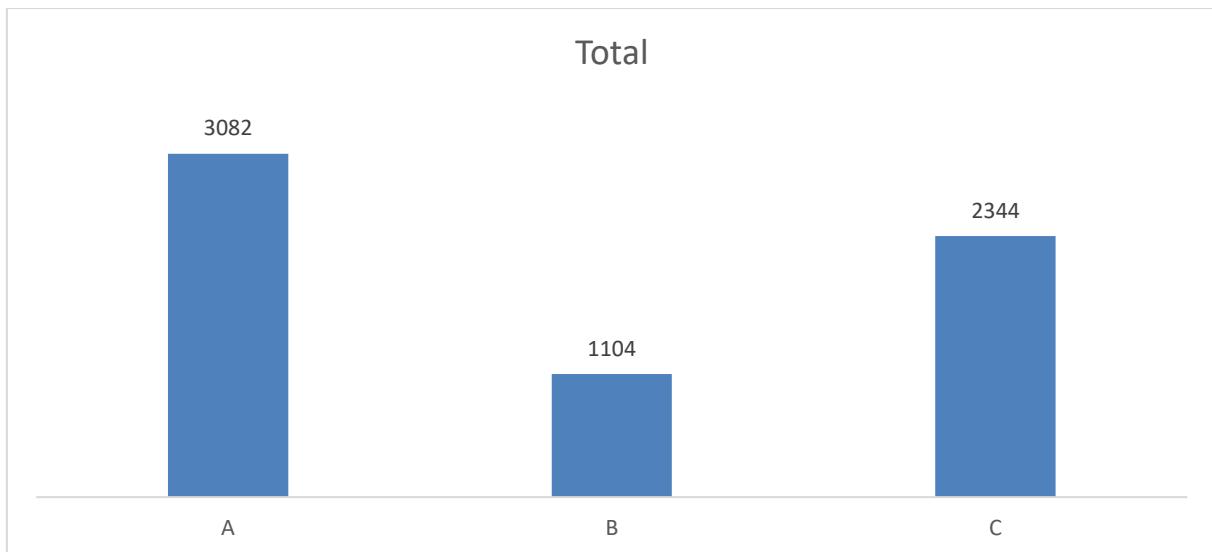
Foram feitas 239 novas classificações por parte do especialista para gerar essa comparação. Como mostra a figura 12, o algoritmo acertou 80% dos materiais que o especialista classificou, resultando em 192 classificações iguais e errou 47 classificações, o que representa 20% do total comparado.

Após a seleção do modelo SVM como o de melhor desempenho e a aplicação do aprendizado semi-supervisionado para classificar todo o inventário, a etapa final da pesquisa consistiu em uma validação prática da eficácia do resultado. O objetivo desta análise comparativa foi verificar o alinhamento entre as previsões do modelo e a lógica de decisão humana em um conjunto de itens que não fizeram parte da amostra original de treinamento, servindo como a validação final da capacidade de generalização e da acurácia do sistema proposto em um cenário prático.

Dito isso, observou-se uma taxa de concordância de 80%. Para estimar a precisão deste resultado, foi calculado o erro padrão da proporção, que resultou em aproximadamente 2,59%. Com base nesse valor, foi construído um intervalo de confiança de 95%. A análise estatística indica, com 95% de confiança, que a verdadeira taxa de concordância entre o modelo e o especialista, se aplicada a toda a população de materiais, estaria contida no intervalo entre 75% e 85%. Este resultado confere uma alta e estatisticamente significativa confiabilidade ao modelo preditivo desenvolvido.

A distribuição final de todos os 6.530 itens de MRO entre as classes A, B e C, conforme determinado pelo modelo, se encontra na figura 13.

Figura 13 - Classificação final.



Fonte: O autor (2025).

Para a classificação que o modelo escolhido realizou na base de dados, também foi medido a qualidade de aproximação γ e o resultado reafirma o resultado obtido na seção de escolha do modelo. Percebe-se na figura 14 que o γ foi de aproximadamente 87%.

Figura 14 - Qualidade da aproximação do SVM pelo jmaf.

Quality of approximation: 0.873	
Union name	Accuracy
At most C	0.736
At least B	0.803
At most B	0.831
At least A	0.777

Fonte: O autor (2025).

5 CONCLUSÃO

A pesquisa partiu da premissa de que métodos de classificação tradicionais ou subjetivos são frequentemente inconsistentes e ineficazes para gerenciar a complexidade do estoque de MRO. Norteados pelo seguinte problema de pesquisa: De que maneira a incorporação de múltiplas variáveis, por meio de algoritmos de aprendizado de máquina, gera uma classificação ABC alinhada aos objetivos operacionais de uma indústria de grande porte, quando comparada ao método tradicional?

Ao final da pesquisa, constatou-se que todos os objetivos, geral e específicos, foram alcançados com sucesso. O objetivo geral de analisar a aplicação da classificação ABC com aprendizado de máquina foi plenamente atendido por meio da execução dos seguintes objetivos específicos: foi caracterizado o processo de gestão da empresa e gerado um conjunto de dados representativo; foi analisado o método de classificação atual, quantificando seu nível de consistência; foram desenvolvidos, treinados e otimizados cinco diferentes algoritmos de classificação supervisionado; foi realizada uma análise comparativa para selecionar o modelo de melhor desempenho; e, finalmente, este modelo foi aplicado em um framework semi-supervisionado para classificar todo o inventário, com sua eficácia sendo validada posteriormente.

A pesquisa demonstrou que a incorporação de múltiplas variáveis via aprendizado de máquina gera uma classificação consideravelmente mais consistente, robusta e alinhada aos objetivos operacionais. Enquanto o método original da empresa apresentou uma qualidade de aproximação de apenas 12%, indicando alta inconsistência, o modelo de aprendizado de máquina (SVM), destacou-se com a maior média de acurácia (80,88%) e a menor variância, sendo selecionado como o modelo mais adequado, treinado com base em uma amostra consistente classificada por um especialista ($\gamma = 80\%$), foi capaz de replicar a lógica deste com alta fidelidade.

Sendo assim, prova que o método proposto substitui um processo subjetivo e falho por um sistema escalável e que reflete com maior precisão a criticidade dos itens.

As contribuições do estudo são tanto práticas quanto acadêmicas. Para a empresa, o trabalho oferece uma metodologia validada para reestruturar sua política de estoque de MRO.

Para a academia, a pesquisa contribui com um framework prático para a aplicação e comparação de múltiplos algoritmos de classificação, além de demonstrar uma abordagem semi-supervisionada para escalar a rotulagem de dados em contextos industriais.

Reconhece-se que o estudo possui limitações. Por ser um estudo de caso único, a generalização dos resultados para outros setores deve ser feita com cautela. A utilização de dados sintéticos, embora necessária para proteger a confidencialidade, elimina algumas das complexidades de dados reais. Adicionalmente, a classificação de referência foi baseada na expertise de um único especialista, podendo conter um viés individual.

Diante de tais considerações, recomenda-se para trabalhos futuros um maior aprofundamento sobre expandir o número de variáveis no modelo, incluindo variáveis como número de fornecedores e obsolescência dos itens. Explorar algoritmos mais avançados, como técnicas de deep learning e outros algoritmos de aprendizado semi-supervisionado, como clusterização. Desenvolver uma abordagem de decisão em grupo para criar a base de referência, utilizando técnicas de consolidação de opinião de múltiplos especialistas para mitigar o viés individual.

REFERÊNCIAS

Ballou, Ronald. **Gerenciamento da cadeia de suprimentos/logística empresarial.** 5^a. ed. Porto Alegre: Bookman, 2008. Acesso em: 22 Junho 2025.

Błaszczyński, Jerzy et al. **Dominance-based Rough Set Approach Data Analysis Framework.** Institute of Computing Science. Poznań. 2011.

Cabral Filho, Djalma Alves. **Getão logística e tendências da logística 4.0.** 1^a. ed. Ponta Grossa: Atena Editora, 2023. 121 p. Acesso em: 23 Junho 2025.

Camargo Izahc, Jussara. **Gestão de estoques de itens mro através das curvas de classificação e agrupamento de volume e criticidade aplicada em uma empresa fabricante de equipamentos de transporte ferroviário de carga.** Laboratório de Aprendizagem em Logística e Transportes. Campinas, p. 16. 2020.

Campos, Eduardo Bessa Sales; Rezende, Raphael Sodero. **Gestão de peças para MRO: estudo de caso de um operador ferroviário.** Engenharia de Produção. Rio de Janeiro, p. 77. 2013.

Couto, Ayrton Benedito Gaia do; Gomes, Luiz Flavio Autran Monteiro. Multi-criteria web mining with DRSA. **Procedia Computer Science**, Rio de Janeiro-RJ, 91, n. 1, 6 Agosto 2016. 131-140. Acesso em: 9 Julho 2025.

Faria, Marina Dias; Carvalho, José Luis Felicio ; Rossi, Raphael. Planejamento colaborativo da demanda e sistemas de informação: gestão de estoques na cadeia de suprimentos. **ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO**, Rio de Janeiro, 16 Outubro 2008. 14. Disponível em:
https://abepro.org.br/biblioteca/enegep2008_tn_sto_069_492_11140.pdf. Acesso em: 23 Junho 2025.

Gaia, Ayrton Benedito. **Avaliação da sustentabilidade e do desempenho de empresas do setor elétrico brasileiro: aplicação do apoio multicritério à**

decisão e aprendizado de máquina. Universidade Federal Fluminense - UFF. Niterói, p. 189. 2024.

Gaia, Ayrton Benedito; Rangel, Luís Duncan Alberto. Model proposition for predicting sustainability classes using multicriteria decision support and artificial intelligence.

Gestão e produção, 7 Novembro 2022. 25. Acesso em: 29 Junho 2025.

Hartshorn, Scott. **Machine learning with random forests and decision trees.** [S.I.]: [s.n.], 2016. Acesso em: 28 Junho 2025.

Hu, Qiwei *et al.* Spare parts classification in industrial manufacturing using the dominance-based rough set approach. **European Journal of Operational Research**, 262, n. 3, 1 Novembro 2017. 1136-1163. Acesso em: 12 Julho 2025.

Kaminski Sanches, Marcelo. **Aprendizado de máquina semi-supervisionado: proposta de um algoritmo para rotular exemplos a partir de poucos exemplos rotulados.** São Carlos. 2003.

Kennedy, W.J. ; Patterson, J. Wayne ; Fredendall, Lawrence D.. An overview of recent literature on spare parts inventories. **International production economics**, 76, n. 2^a, 21 Março 2002. 201-215. Acesso em: 23 Junho 2025.

Lekha, Dharani *et al.* Multi-Criteria inventory classification using machine learning algorithms. **International Conference on Industrial Engineering and Operations Management, Warangal, India**, Warangal, 16 Agosto 2022. Acesso em: 28 Junho 2025.

Ng, Wan Lung. A simple classifier for multiple criteria ABC analysis. **European Journal of Operational Research**, Hong Kong, 177, n. 1, 16 Fevereiro 2007. 344-353. Acesso em: 29 Junho 2025.

Peixoto, Eduardo Carrara; Pinto, Luiz Ricardo. Gerenciamento de estoques via previsão de vendas agregadas utilizando simulação. **Associação Brasileira de Engenharia de Produção**, São José dos Campos, 16, n. 3, 13 Fevereiro 2007. 569-

581. Disponível em:

<https://www.scielo.br/j/prod/a/kft3c3dwd437DWQ5wTSgkXB/?lang=pt>. Acesso em: 23 Junho 2025.

Póvoa, Bárbara Batista; Chaves, Gisele de Lorena Diniz. Ponto de ressuprimento para materiais com demanda sazonal e aleatória: o caso de uma empresa de grande porte. **Revista Gestão Industrial**, Campus Ponta Grossa, 10, n. 3, 17 Dezembro 2014. 584-618. Acesso em: 23 Junho 2025.

Santos, Marcos Rogério da Silva. **Aplicação da distribuição de poisson para determinação de estoque mínimo de itens de MRO de baixo giro aplicados em manutenção de ativos industriais**. Engenharia de Produção. Bauru, p. 116. 2015.

Slack, Nigel; Brandon-Jones, Alistair; Johnston, Robert. **Administração da Produção**. 8^a. ed. São Paulo: Atlas, 2018. Acesso em: 22 Junho 2025.

Slowinski, Roman; Greco, Salvatore; Matarazzo, Benedetto. rough set and rule-based multicriteria decision aiding. **Pesquisa Operacional**, 32, n. 2, 4 Setembro 2012. 213-269. Acesso em: 9 Julho 2025.

Taylor, Michael. **Neural networks. A visual introduction for beginners**. [S.I.]: Blue Windmill Media, 2017. Acesso em: 28 Junho 2025.

Tessarini, Geraldo; Saltorato, Patrícia. Impactos da indústria 4.0 na organização do trabalho: uma revisão sistemática da literatura. **Revista Produção Online**, 18, n. 2, 15 Junho 2018. 743–769. Disponível em:
<https://www.producaoonline.org.br/rpo/article/view/2967>. Acesso em: 22 Junho 2025.

Vieira, Darli Rodrigues ; Loures, Paula Lavorato. Maintenance, Repair and Overhaul (MRO) Fundamentals and Strategies: An Aeronautical Industry Overview. **International Journal of Computer Applications**, Nova Iorque, 135, n. 12^a, 12 Fevereiro 2016. 21-29. Acesso em: 24 Junho 2025.

Wanke, Peter. **Gestão De Estoques Na Cadeia De Suprimento: Decisões E Modelos Quantitativos.** 3^a. ed. São Paulo: Editora Atlas, 2011. Acesso em: 23 Junho 2025.

APÊNDICE A – CÓDIGO DOS TESTES E ESCOLHA DO MODELO

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import pickle
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from yellowbrick.classifier import ConfusionMatrix

# Carregamento da base de dados classificada pelo especialista
df_tcc = pd.read_excel('BD_TCC.xlsx', sheet_name='BD_Analista')

# Remoção de colunas não utilizadas na modelagem
df_tcc.drop(['Loja', 'Tipo de Consumo', 'Lead Time'], axis=1, inplace=True)

# Separação das variáveis independentes (X) e da variável dependente (y)
x_classe = df_tcc.iloc[:, 0:3].values
y_classe = df_tcc.iloc[:, 3].values

# Tratamento de atributos categóricos (variável 'Familia')
# 1. Label Encoder para transformar as categorias em números
label_enconder_familia = LabelEncoder()
x_classe[:,0] = label_enconder_familia.fit_transform(x_classe[:,0])

# 2. OneHotEncoder para transformar os números em colunas binárias
onehotencoder_classe = ColumnTransformer(transformers=[('OneHot',
OneHotEncoder(sparse_output=False), [0])], remainder='passthrough')
```

```
x_classe = onehotencoder_classe.fit_transform(x_classe)

# Escalonamento dos valores para padronização
scaler_classe = StandardScaler()
x_classe = scaler_classe.fit_transform(x_classe)

# Divisão da base em conjuntos de treinamento e teste
x_classe_treinamento, x_classe_teste, y_classe_treinamento, y_classe_teste =
train_test_split(x_classe, y_classe, test_size=0.25, random_state=0)

# Salvando as variáveis pré-processadas para uso futuro
with open('abc.pkl', mode = 'wb') as f:
    pickle.dump([x_classe_treinamento, y_classe_treinamento, x_classe_teste,
y_classe_teste], f)

#REGRESSÃO LOGÍSTICA
from sklearn.linear_model import LogisticRegression

# Carregamento das variáveis salvas
with open('abc.pkl', 'rb') as f:
    x_classe_treinamento, y_classe_treinamento, x_classe_teste, y_classe_teste =
pickle.load(f)

# Treinamento do modelo
logistic_classe = LogisticRegression(random_state=1)
logistic_classe.fit(x_classe_treinamento, y_classe_treinamento)

# Realização das previsões no conjunto de teste
previsoes = logistic_classe.predict(x_classe_teste)

# Avaliação da acurácia e visualização da matriz de confusão
print(accuracy_score(y_classe_teste, previsoes))
cm = ConfusionMatrix(logistic_classe)
cm.fit(x_classe_treinamento, y_classe_treinamento)
```

```
cm.score(x_classe_teste, y_classe_teste)
print(classification_report(y_classe_teste, previsoes))

#RANDOM FOREST
from sklearn.ensemble import RandomForestClassifier
random_forest_classe = RandomForestClassifier(n_estimators=30,
criterion='entropy', random_state=0)
random_forest_classe.fit(x_classe_treinamento, y_classe_treinamento)
previsoes = random_forest_classe.predict(x_classe_teste)
accuracy_score(y_classe_teste, previsoes)
cm = ConfusionMatrix(random_forest_classe)
cm.fit(x_classe_treinamento, y_classe_treinamento)
cm.score(x_classe_teste, y_classe_teste)
print(classification_report(y_classe_teste, previsoes))
```

```
#KNN
from sklearn.neighbors import KNeighborsClassifier
knn_classe = KNeighborsClassifier(n_neighbors=5)
knn_classe.fit(x_classe_treinamento, y_classe_treinamento)
previsoes = knn_classe.predict(x_classe_teste)
previsões
accuracy_score(y_classe_teste, previsoes)
cm = ConfusionMatrix(knn_classe)
cm.fit(x_classe_treinamento, y_classe_treinamento)
cm.score(x_classe_teste, y_classe_teste)
print(classification_report(y_classe_teste, previsoes))
```

```
#SVM
from sklearn.svm import SVC
svm_classe = SVC(kernel='poly', random_state=1)
svm_classe.fit(x_classe_treinamento, y_classe_treinamento)
previsoes = svm_classe.predict(x_classe_teste)
previsões
accuracy_score(y_classe_teste, previsoes)
```

```

cm = ConfusionMatrix(svm_classe)
cm.fit(x_classe_treinamento, y_classe_treinamento)
cm.score(x_classe_teste, y_classe_teste)
print(classification_report(y_classe_teste, previsoes))

#REDE NEURAL
from sklearn.neural_network import MLPClassifier
rede_neural_classe = MLPClassifier(verbose=True, max_iter=1500, tol=0.000010,
                                    hidden_layer_sizes=(19,19))
rede_neural_classe.fit(x_classe_treinamento, y_classe_treinamento)
previsoes = rede_neural_classe.predict(x_classe_teste)
previsões
accuracy_score(y_classe_teste, previsoes)
cm = ConfusionMatrix(rede_neural_classe)
cm.fit(x_classe_treinamento, y_classe_treinamento)
cm.score(x_classe_teste, y_classe_teste)
print(classification_report(y_classe_teste, previsoes))

#GRID SEARCH
from sklearn.model_selection import GridSearchCV
with open('abc.pkl', 'rb') as f:
    x_classe_treinamento, y_classe_treinamento, x_classe_teste, y_classe_teste =
        pickle.load(f)
    x_classe = np.concatenate((x_classe_treinamento, x_classe_teste), axis=0)
    y_classe = np.concatenate((y_classe_treinamento, y_classe_teste), axis=0)
parametros = {
    'tol': [0.0001, 0.00001, 0.000001],
    'C': [1.0, 1.5, 2.0],
    'solver': ['lbfgs', 'sag', 'saga']
}
grid_search = GridSearchCV(estimator=LogisticRegression(),
                           param_grid=parametros)
grid_search.fit(x_classe, y_classe)
melhores_parametros = grid_search.best_params_

```

```
melhor_resultado = grid_search.best_score_
print(melhores_parametros)
print(melhor_resultado)

parametros = {
    'criterion': ['gini', 'entropy'],
    'n_estimators': [10,40,100,150],
    'min_samples_split': [2,5,10],
    'min_samples_leaf': [1,5,10]
}

grid_search = GridSearchCV(estimator=RandomForestClassifier(),
param_grid=parametros)
grid_search.fit(x_classe, y_classe)
melhores_parametros = grid_search.best_params_
melhor_resultado = grid_search.best_score_
print(melhores_parametros)
print(melhor_resultado)

parametros = {
    'n_neighbors': [3,4,5,6,7,10,20,25,30],
    'p': [1,2]
}

grid_search = GridSearchCV(estimator=KNeighborsClassifier(),
param_grid=parametros)
grid_search.fit(x_classe, y_classe)
melhores_parametros = grid_search.best_params_
melhor_resultado = grid_search.best_score_
print(melhores_parametros)
print(melhor_resultado)

parametros = {
    'tol': [0.0001, 0.00001, 0.000001],
    'C': [1.0,1.5,2.0],
    'kernel': ['rbf', 'linear', 'poly', 'sigmoid']
```

```

    }

grid_search = GridSearchCV(estimator=SVC(), param_grid=parametros)
grid_search.fit(x_classe, y_classe)
melhores_parametros = grid_search.best_params_
melhor_resultado = grid_search.best_score_
print(melhores_parametros)
print(melhor_resultado)

parametros = {
    'activation': ['relu', 'logistic', 'tahn'],
    'solver': ['adam', 'sgd'],
    'batch_size': [10,56]
}

grid_search = GridSearchCV(estimator=MLPClassifier(), param_grid=parametros)
grid_search.fit(x_classe, y_classe)
melhores_parametros = grid_search.best_params_
melhor_resultado = grid_search.best_score_
print(melhores_parametros)
print(melhor_resultado)

#VALIDAÇÃO CRUZADA
from sklearn.model_selection import cross_val_score, KFold
resultados_random_forest = []
resultados_knn = []
resultados_logistica = []
resultados_svm = []
resultados_rede_neural = []

for i in range(30):
    print(i)
    kfold = KFold(n_splits=10, shuffle=True, random_state=i)

    random_forest = RandomForestClassifier(criterion = 'gini', min_samples_leaf = 1,
min_samples_split=5, n_estimators = 100)

```

```
scores = cross_val_score(random_forest, x_classe, y_classe, cv = kfold)
resultados_random_forest.append(scores.mean())

knn = KNeighborsClassifier(n_neighbors=5, p=2)
scores = cross_val_score(knn, x_classe, y_classe, cv = kfold)
resultados_knn.append(scores.mean())

logistica = LogisticRegression(C = 2.0, solver = 'saga', tol = 0.000001)
scores = cross_val_score(logistica, x_classe, y_classe, cv = kfold)
resultados_logistica.append(scores.mean())

svm = SVC(C= 1.0, kernel= 'rbf', tol= 0.0001)
scores = cross_val_score(svm, x_classe, y_classe, cv = kfold)
resultados_svm.append(scores.mean())

rede_neural = MLPClassifier(activation = 'relu', batch_size = 10, solver = 'adam')
scores = cross_val_score(rede_neural, x_classe, y_classe, cv = kfold)
resultados_rede_neural.append(scores.mean())
resultados = pd.DataFrame({'Random forest': resultados_random_forest,
                           'KNN': resultados_knn, 'Rede neural': resultados_rede_neural,
                           'logistica': resultados_logistica,
                           'SVM': resultados_svm})

resultados

resultados.describe()
```

APÊNDICE B – GERAÇÃO DA AMOSTRA PARA O ESPECIALISTA

```

import pandas as pd
import numpy as np # Usado para np.nan no exemplo de dados
import openpyxl

df_tcc = pd.read_excel("BD_TCC.xlsx", sheet_name="BD_Classificados")
df_tcc

def criar_amostra_estratificada_unica_simplificada(df_original, coluna_familia,
tamanho_amostra=200, seed_aleatoria=None):
    """
    Cria uma única amostra aleatória estratificada de forma simplificada,
    assumindo que os dados estão limpos e as condições são ideais:
    - Menos famílias que o tamanho da amostra.
    - Sem NaNs na coluna família.
    - Itens suficientes.
    """

    Args:
        df_original (pd.DataFrame): DataFrame contendo os dados completos.
        coluna_familia (str): Nome da coluna com a família do material.
        tamanho_amostra (int): Tamanho desejado para a amostra.
        seed_aleatoria (int, optional): Seed para reproduzibilidade.

    Returns:
        pd.DataFrame: DataFrame contendo a amostra gerada.
    """

    df = df_original.copy() # Trabalhar com uma cópia

    familias_unicas = df[coluna_familia].unique() # Não precisa de .dropna()

    # Parte 1: Selecionar um item garantido de cada família
    itens_garantidos_lista = []
    df_copia_para_selecao = df.copy()

```

Args:

- df_original (pd.DataFrame): DataFrame contendo os dados completos.
- coluna_familia (str): Nome da coluna com a família do material.
- tamanho_amostra (int): Tamanho desejado para a amostra.
- seed_aleatoria (int, optional): Seed para reproduzibilidade.

Returns:

pd.DataFrame: DataFrame contendo a amostra gerada.

"""

```
df = df_original.copy() # Trabalhar com uma cópia
```

```
familias_unicas = df[coluna_familia].unique() # Não precisa de .dropna()
```

```
# Parte 1: Selecionar um item garantido de cada família
```

```
itens_garantidos_lista = []
```

```
df_copia_para_selecao = df.copy()
```

```

for familia in familias_unicas:
    itens_da_familia =
        df_copia_para_selecao[df_copia_para_selecao[coluna_familia] == familia]
    # Assume-se que itens_da_familia nunca estará vazio para uma família em
    familias_unicas
        item_selecionado = itens_da_familia.sample(n=1,
random_state=seed_aleatoria)
        itens_garantidos_lista.append(item_selecionado)
        df_copia_para_selecao = df_copia_para_selecao.drop(item_selecionado.index)

amostra_parte_1 = pd.concat(itens_garantidos_lista)

# Parte 2: Selecionar os itens restantes para completar a amostra
num_itens_adicionais_necessarios = tamanho_amostra - len(amostra_parte_1)

# Assume-se que num_itens_adicionais_necessarios será >= 0
# e que df_copia_para_selecao terá itens suficientes.
if num_itens_adicionais_necessarios > 0:
    amostra_parte_2 =
        df_copia_para_selecao.sample(n=num_itens_adicionais_necessarios,
random_state=seed_aleatoria)
        amostra_final = pd.concat([amostra_parte_1,
amostra_parte_2]).reset_index(drop=True)
else: # Caso len(amostra_parte_1) == tamanho_amostra (nenhum item adicional
necessário)
    amostra_final = amostra_parte_1.reset_index(drop=True)

return amostra_final

amostra_3 = criar_amostra_estratificada_unica_simplificada(df_original=df_tcc,
coluna_familia="Familia", seed_aleatoria=None)

```

```
amostra_3.to_excel("TCC_Amostra_3.xlsx", index= False)
```

APÊNDICE C – APRENDIZADO SEMI-SUPERVISIONADO

```

from sklearn.svm import SVC
from sklearn.semi_supervised import SelfTrainingClassifier
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
import numpy as np
import pandas as pd

df = pd.read_excel('BD_TCC.xlsx', sheet_name= "BD_Classificados")
df

# Separar variáveis
cat_features = ['Familia'] #Categóricas
num_features = ['Consumo Médio Anual', 'Preço Médio'] #Numéricas
X = df[cat_features + num_features]
# Criar vetor de rótulos: -1 para não rotulados
y = df['ABC Analista'].copy()
y = y.fillna(-1)

# Pré-processamento
preprocessor = ColumnTransformer([
    ('cat', OneHotEncoder(handle_unknown='ignore'), cat_features),
    ('num', StandardScaler(), num_features)
])
# Classificador base
base_classificador_svm = SVC(probability=True, C= 1.0, kernel= 'rbf', tol= 0.0001,
random_state=42)
# Modelo semi-supervisionado
self_training = SelfTrainingClassifier(
    base_classificador_svm,
    threshold=0.8,
    verbose=True
)

```

```
)  
  
# Pipeline completo  
pipeline = Pipeline([  
    ('preproc', preprocessor),  
    ('self_train', self_training)  
])  
  
# Treinamento  
print("Iniciando o treinamento do modelo semi-supervisionado com SVM...")  
# Treina o modelo com todos os dados (rotulados e não rotulados)  
pipeline.fit(X, y)  
print("-----")  
print("Treinamento concluído!")  
  
print("Realizando a previsão para todos os materiais...")  
y_pred = pipeline.predict(X)  
  
# Adiciona os resultados como uma nova coluna no DataFrame original  
df['ABC_SVM'] = y_pred  
# Salva o DataFrame completo com as previsões em um novo arquivo Excel  
salvar = 'Resultado_final_com_previsao_svm.xlsx'  
df.to_excel(salvar, index=False)
```