

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Ciência da Computação

Pedro Henrique de Souza Martins

**UMA ABORDAGEM CENTRADA A DADOS PARA IDENTIFICAÇÃO
DE FALHAS EM PROCESSOS PRODUTIVOS COMPLEXOS**

Belo Horizonte
2024

Pedro Henrique de Souza Martins

**UMA ABORDAGEM CENTRADA A DADOS PARA IDENTIFICAÇÃO
DE FALHAS EM PROCESSOS PRODUTIVOS COMPLEXOS**

Versão Final

Dissertação apresentada ao Programa de Pós-Graduação em
Ciência da Computação da Universidade Federal de Minas
Gerais, como requisito parcial à obtenção do título de Mestre
em Ciência da Computação.

Orientador: Adriano Alonso Veloso

Belo Horizonte
2024

[Folha de Aprovação em formato PDF]

A folha de aprovação deve estar em formato PDF e deve ser passada como argumento do comando `ppgccufmg` no arquivo principal `.tex`, conforme o exemplo abaixo:

```
\ppgccufmg{  
    ...  
    folhadeaprovacao={folha.pdf}  
}
```

Dedico este trabalho a todos que me apoiaram e ajudaram nessa jornada.

Agradecimentos

Gostaria de pronunciar meu mais sincero agradecimento a todas as pessoas que contribuíram para a realização desta dissertação e principalmente que ajudaram nessa árdua jornada do mestrado.

Primeiramente, agradeço profundamente à minha família, cujo assistência, apoio constante e compreensão foram fundamentais para a realização desta dissertação. Sem o incentivo e a paciência de vocês, este trabalho não seria possível.

Agradeço ao meu orientador, Professor Dr. Adriano Alonso Veloso, por sua orientação, paciência e apoio inestimáveis ao longo de todo o processo de pesquisa. Sua expertise e conselhos foram fundamentais para a realização deste trabalho.

Um sincero agradecimento aos professores e membros do corpo acadêmico do UFMG pelo suporte acadêmico e pelas oportunidades de crescimento oferecidas durante o meu período de estudos.

Gostaria de expressar minha sincera gratidão aos meus colegas de trabalho por sua compreensão e apoio durante as minhas ausências necessárias para a realização desta dissertação. Sua colaboração e flexibilidade foram essenciais para que eu pudesse conciliar meus compromissos profissionais e acadêmicos.

A todos, meu sincero agradecimento.

“A vida é uma aventura ousada ou não é nada.”
(Helen Keller)

Resumo

Para desenvolver uma aplicação de aprendizado de máquina focada na identificação de falhas em processos produtivos complexos, utilizamos dados de séries temporais fornecidos por uma empresa parceira, definimos nosso objetivo com base em hipóteses sobre a relação causal dos dados coletados. Seleccionamos o modelo *LightGBM* pela sua eficácia em lidar com grandes volumes de dados e realizar previsões precisas. Dividimos os dados em 90% para treinamento e 10% para validação, garantindo a cronologia dos dados e utilizando validação cruzada com *TimeSeriesSplit* para assegurar uma avaliação robusta da generalização do modelo.

Nossa contribuição principal está na criação de uma metodologia de expansão e filtragem de *features*, aplicada diretamente no desenvolvimento do modelo. Utilizamos técnicas de engenharia de características e o algoritmo *SHAP* para criar novos atributos e iterativamente remover variáveis irrelevantes, resultando em um modelo mais eficiente e preciso. Este processo permitiu aprimorar a capacidade do modelo de realizar previsões, mantendo apenas as variáveis de maior importância. Avaliamos o desempenho do modelo usando a métrica *MAE* (*Mean Absolute Error*), e os resultados demonstraram que o *LightGBM* obteve o menor erro médio absoluto, destacando a eficácia da nossa abordagem.

A relevância científica desta metodologia reside na sua capacidade de melhorar a interpretabilidade e a eficiência dos modelos de aprendizado de máquina em contextos industriais. Os resultados indicam que nosso modelo pode prever com precisão o tempo até a falha da máquina, contribuindo para o campo da manutenção preditiva. A pesquisa contribui para o campo de manutenção preditiva, oferecendo uma abordagem eficiente para a previsão de falhas em processos industriais complexos.

Palavras-chave: Aprendizado de Máquina, Interpretabilidade, Séries Temporais, Computação.

Abstract

To develop a machine learning application focused on identifying faults in complex production processes, we used time series data provided by a partner company, defining our objective based on hypotheses about the causal relationship of the collected data. We selected the LightGBM model for its effectiveness in handling large volumes of data and making accurate predictions. We split the data into 90% for training and 10% for validation, ensuring data chronology and using TimeSeriesSplit cross-validation to ensure robust model generalization.

Our main contribution involves creating a feature expansion and filtering methodology, applied directly in model development. We used feature engineering techniques and the SHAP algorithm to create new attributes and iteratively remove irrelevant variables, resulting in a more efficient and accurate model. This process enhanced the model's predictive capability by retaining only the most important variables. We evaluated the model's performance using the Mean Absolute Error (MAE) metric, and the results demonstrated that LightGBM achieved the lowest mean absolute error, highlighting the effectiveness of our approach.

The scientific relevance of this methodology lies in its ability to improve the interpretability and efficiency of machine learning models in industrial contexts. The results indicate that our model can accurately predict the time until machine failure, contributing significantly to the field of predictive maintenance by offering an efficient approach to fault prediction in complex industrial processes.

Keywords: Machine Learning, Interpretability, Time Series, Computing.

Lista de Figuras

1.1	Equipamento de desodorização	14
2.1	Tipos de modelos de aprendizado de máquina	18
2.2	Exemplo de Algoritmo de Árvore de Decisão	19
2.3	Modelos Intrinsecamente Interpretáveis <i>versus</i> Modelos de Caixa Preta. Ilustração retirada do artigo [46]	22
2.4	Exemplo simples de como <i>SHAP</i> ajuda na interpretação da predição de modelos.	24
2.5	Gráfico resumido das importâncias das <i>features</i> na predição do modelo de predição de valores de moradia	25
2.6	Exemplificação de séries temporais sobre preço de ações aplicadas ao longo de um período de tempo	27
3.1	Fluxograma mostrando os passos de pré-processamento	31
3.2	Gráfico com exemplo da variável de tempo para falha	34
3.3	Representação em forma de funil do resultado <i>dataset</i>	35
3.4	Processo de escolha de algoritmo	36
4.1	Exemplificação da biblioteca <i>TimeSeriesSplit</i>	39
4.2	Comparação final de algoritmos	40
4.3	Comparação predição <i>versus</i> real	41
4.4	Valores <i>SHAP</i> das principais características	42
4.5	Histograma dos Resíduos	43
4.6	Q-Q <i>plot</i>	44

Lista de Tabelas

4.1	Tabela comparativa de experimentos	39
-----	--	----

Lista de Algoritmos

Sumário

1	Introdução	12
1.1	Motivação	12
1.2	Definição do Problema	13
1.3	Objetivos	13
1.4	Contribuições	15
1.5	Organização da Dissertação	15
2	Referencial Teórico e Trabalhos Relacionados	17
2.1	Aprendizado de Máquina	17
2.1.1	Modelos Supervisionados	19
2.1.2	Modelos Não Supervisionados	20
2.1.3	Modelos de Aprendizado por Reforço	21
2.2	Interpretabilidade de Modelos	21
2.2.1	SHAP	23
2.3	Abordagem Orientada a Dados	25
2.4	Séries Temporais	26
2.5	Trabalhos Relacionados	28
3	Modelo para Predição de Quebra de Vácuo	31
3.1	Coleta de Dados	31
3.2	Pré-Processamento	32
3.3	Definição da variável alvo	33
3.4	Algoritmo	34
3.5	Aplicação do Modelo de Aprendizado de Máquina	36
4	Experimentos e Resultados	38
4.1	Configuração dos experimentos	38
4.2	Discussão dos Resultados	39
5	Conclusão	45
5.1	Conclusão	45
5.2	Limitações do Trabalho	46
5.3	Trabalhos Futuros	46
	Referências Bibliográficas	47

Capítulo 1

Introdução

A utilização de soluções que empregam aprendizado de máquina tem sido cada vez mais impactantes no nosso cotidiano [39]. Já são claras as várias situações onde esses algoritmos podem ser empregados, por exemplo: prevenção de fraudes, análise de textos, previsão de falhas de equipamentos, entre muitas outras utilidades. Após a etapa de criação e treinamento dos modelos, um ponto importante é interpretar e entender o que os modelos predisseram. Não que modelos com alta performance não sejam confiáveis e que podemos apenas ignorar as suas predições. A questão é que uma única métrica, como precisão, é uma descrição incompleta da maioria das tarefas do mundo real [14].

Ao longo desta dissertação, serão abordados temas relacionados a *datacentric*, interpretabilidade de modelos e aprendizado de máquina aplicado à identificação de falhas em processos produtivos. Serão analisados trabalhos relacionados, identificando lacunas no conhecimento e oportunidades para contribuições originais. Além disso, será apresentada uma metodologia robusta para desenvolver e avaliar modelos de aprendizado de máquina com foco na identificação de falhas em ambientes industriais complexos.

1.1 Motivação

Em um setor industrial é interessante que todas as etapas e fluxos estejam alinhados e precisos para garantir qualidade e eficiência na produção. As falhas no processo podem ocorrer por diversas razões: Inadequações dos Processos, Falta de habilidade, Desatenção, entre outros [16].

A identificação de falhas pode levar a uma redução de problemas e consequentemente, à redução de custos e despesas para as empresas durante o processo produtivo, uma vez que diminuir falhas implica na redução de recursos gastos de forma desnecessária e aumento dos lucros.

O emprego de algoritmos de aprendizado de máquina, que são essencialmente ma-

peamentos locais $X \rightarrow y$ relacionando uma combinação de características na instância X para um rótulo y , pode proporcionar ao setor industrial uma visão diferente do ponto de vista de manutenção de equipamentos, sugerindo uma possível abordagem de manutenção preditiva. Tal tipo de tratamento é capaz de projetar o monitoramento do equipamento em tempo real, sendo assim gerar alertas sobre o sistema com antecedência e determinar se será necessária manutenção corretiva.[67]

1.2 Definição do Problema

Dado um conjunto de dados do tipo *timeseries* devidamente rotulados que demonstram o estado do equipamento descrevendo as informações dos sensores internos, tal qual receita do produto, pressão interna, temperatura, vazão entre diversas outras, assume-se que cada registro cronológico da situação da máquina pode ser importante e útil para o reconhecimento de padrões de falha interna.

A máquina de desodorização exemplificada na figura 1.1 utilizada no processo produtivo está equipada com diversos sensores que monitoram parâmetros críticos da mesma. Esses sensores são vitais para garantir o controle preciso das condições operacionais, permitindo a coleta de dados detalhados e em tempo real. Os dados fornecidos por esses sensores são fundamentais para o desenvolvimento de modelos de aprendizado de máquina destinados à identificação de falhas, pois oferecem uma visão abrangente do comportamento e desempenho da máquina ao longo do tempo.

Sendo assim, busca-se nesse trabalho desenvolver um modelo capaz de antever com certo grau de confiabilidade em qual momento e em quais condições o equipamento sofrerá uma falha interna de pressão.

1.3 Objetivos

O objetivo principal deste trabalho é implementar uma metodologia de expansão e filtragem de características do dado capaz de gerar resultados em métricas de avaliação oportunas. O desenvolvimento do trabalho permitiu que fossem gerados ganhos consideráveis de desempenho e este trabalho visa relatar que isso também pode se aplicar em outros sistemas. Durante o desenvolvimento do trabalho, verificou-se que o desempenho

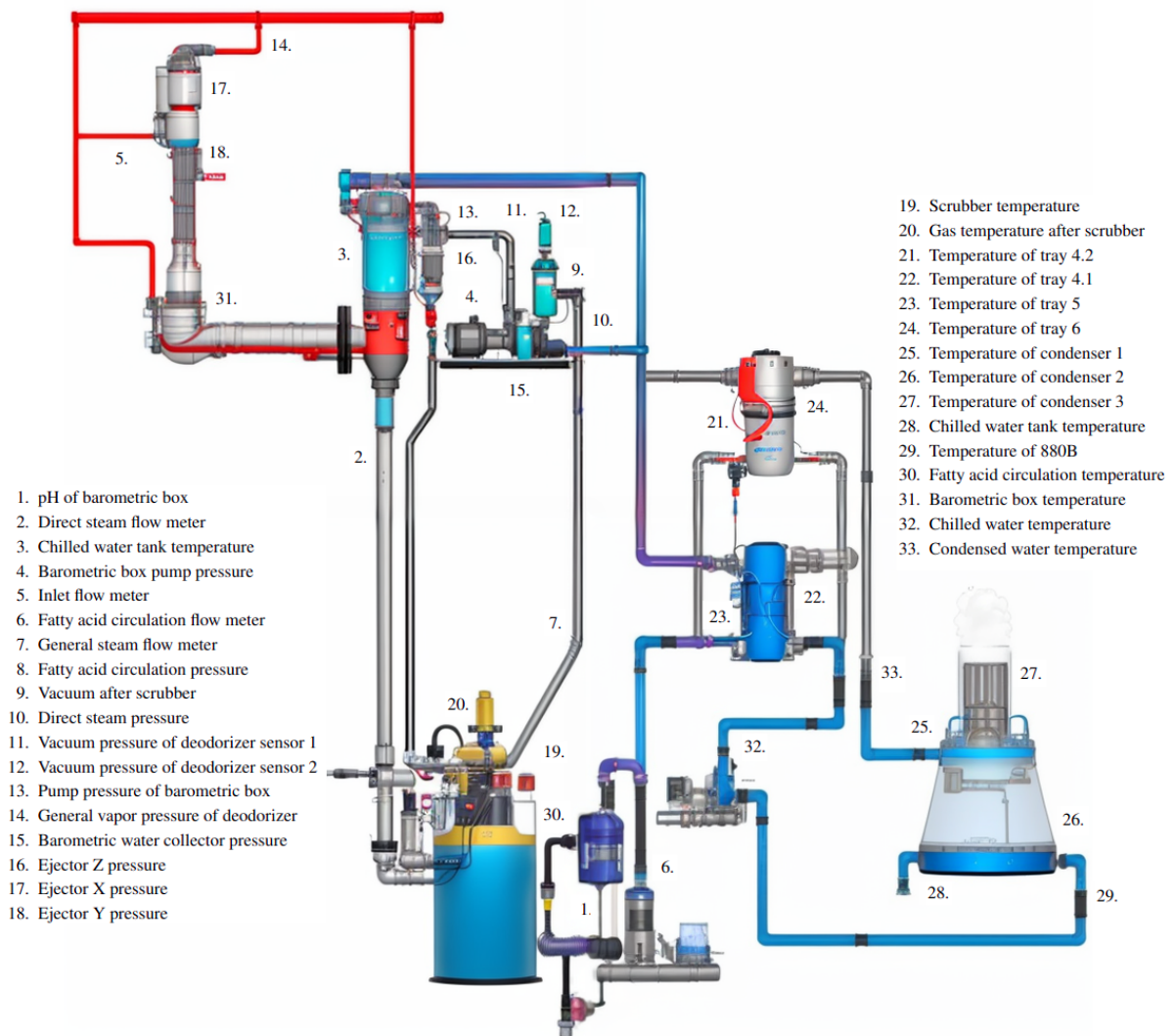


Figura 1.1: Equipamento de desodorização

do modelo base desenvolvido era superior aos *baselines* encontrados no estado-da-arte. Desta forma, determinou-se um objetivo secundário de exercitar o modelo inicial afim de avaliar eventuais melhorias de desempenho. Para tal, foram treinados modelos adicionais modulando valores de filtragem em razão da métrica de avaliação inicial.

Destacam-se os objetivos específicos:

- Realizar pesquisa sobre utilização de modelos preditivos aplicados a sistemas de produção industriais complexos a fim de averiguar os diferentes métodos existentes na literatura.
- Implementar um modelo preditivo que tenha desempenho comparável aos demais métodos de estado-da-arte.
- Utilizar adaptação no modelo proposto e avaliar o ganho de desempenho.

- Explorar diferentes abordagens para a criação do modelo de aprendizado de máquina final. Ainda, comparar o desempenho de cada uma e suas respectivas vantagens e desvantagens.

1.4 Contribuições

A principal contribuição deste trabalho é elucidar que colocar os dados como o elemento central na construção de modelos de aprendizado de máquina pode levar a uma extração avançada de performance dos algoritmos preditivos existentes. De maneira geral, também destaca-se as seguintes contribuições:

- Foram empregadas diversas técnicas existentes na literatura em conjunto para alcançar o objetivo do trabalho.
- Dadas as representações vetoriais dos sensores de um máquina industrial, foi construído um modelo paramétrico capaz de identificar em que momento futuro uma falha interna ocorrerá no equipamento.
- Foi avaliada a eficácia do modelo na previsão de falhas utilizando técnicas bem consolidadas como validação cruzada e *MAE*. O modelo atinge valores consideravelmente melhores que modelos de *baseline*.
- Mostrar que o modelo proposto internaliza melhor as mudanças no dado e proporciona melhorias substanciais no conjunto de dados.
- Foram propostas variações de filtragem afim de buscar uma explicação mais coerente da solução proposta.

1.5 Organização da Dissertação

O restante desta dissertação está organizado da seguinte forma: no Capítulo 2, serão revisados os principais trabalhos relacionados sobre temas relacionados a *datacentric*, interpretabilidade de modelos e aprendizado de máquina aplicado à identificação de falhas em processos produtivos; No Capítulo 3, serão apresentados os dados coletados e compilados, como foi feito o pré-processamento das informações, definição da variável

alvo que serão utilizados neste trabalho, e o algoritmo de predição de falhas; No capítulo 4 será descrita a configuração experimental, bem como avaliação do modelo e a discussão dos resultados; Por fim, no capítulo 5, são feitas as considerações finais sobre a dissertação e trabalhos futuros.

Capítulo 2

Referencial Teórico e Trabalhos Relacionados

O presente capítulo tratará conceitos relacionados a *Machine Learning*, como: Criação de Modelos de Aprendizagem de Máquina, Avaliação de Modelos de Aprendizado de Máquina, Análise de Dados, Mineração de dados, Séries Temporais, Pré-processamento de Dados e Abordagem Centrada a Dados.

2.1 Aprendizado de Máquina

A partir da grande massa de dados gerados, é preciso executar o processo para obtê-los e, principalmente, para coletar as informações mais valiosas e úteis. Os aportes teórico de [7] serão necessários à esta pesquisa, já que caracterizam que a mineração de dados é parte integrante de um processo mais amplo, conhecido como descoberta de conhecimento em bases de dados (*knowledge discovery in databases, ou KDD*) [23]. Essa etapa de extração de conhecimento de dados conduz a geração de valor [4] e permite com que as soluções sejam mais assertivas e eficientes.

Com os dados apreendidos, é interessante executar a fase de pré-processamento para identificar possíveis irregularidades, ruídos e inconsistências. Essa etapa apesar de consumir tempo [28] permite criar novas informações agregadas, como a obtenção da idade de uma pessoa, por exemplo, sabendo apenas a data de seu nascimento. Assim, a fase de pré-processamento de dados auxiliará na criação dos modelos, que estabelecerá diretos diálogos com [17] que afirmam que essas técnicas podem eliminar ou reduzir problemas presentes nos dados. As teorias de [44], também auxiliarão a pesquisa, pois apresentam outros argumentos que visam corroborar com as necessidades dos pré-processamentos, os de que alguns algoritmos, como redes neurais, modelos frequentemente utilizados para identificação de objetos em imagens [57] e *SVMs*, modelo que busca encontrar um máximo hiperplano do tipo margem no espaço de entrada que separa o conjunto de dados de

treinamento [64], são sensíveis ao dimensionamento dos dados.

Como esta pesquisa pretende abordar os conceitos de interpretabilidade de modelos de aprendizagem de máquina, será necessário entender sobre os variados tipos de modelos que podem ou não serem utilizados, a partir do problema proposto e dos dados obtidos. Os modelos são classificados em três grandes classes, Modelos Supervisionados, Modelos Não Supervisionados e Modelos de Aprendizado por Reforço conforme figura 2.1. Tais conceitos serão melhor descritos nos subtópicos a seguir.

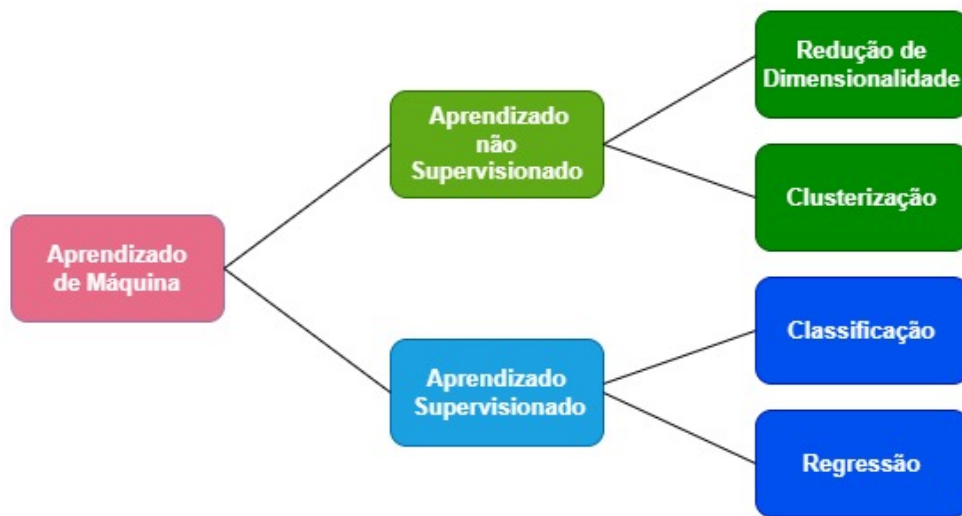


Figura 2.1: Tipos de modelos de aprendizado de máquina

Como [25] argumenta, existem universos de modelos que podem ser utilizados no problema que se deseja resolver, e na maioria dos casos, escolhe-se uma família de modelos para usar os dados e aprender parâmetros ao qual se deseja otimizar.

A avaliação dos modelos é um conceito tão importante quanto os demais, o conhecimento das métricas e medidas de avaliação de modelos são fundamentais para que o trabalho realizado possa ser validado e, caso haja necessidade, o modelo possa sofrer alterações e possíveis correções. [17], também, asseguram que não existe uma técnica universal de aprendizagem de máquina, ou seja, que não existe um modelo de aprendizagem de máquina que se sairá melhor em qualquer tipo de problema. Portanto, conhecer as métricas de erro, pontuação, desempenho [22], testes e hipóteses, se mostram relevantes para a criação de um modelo preciso e, de fato, assertivo.

Todas esses conceitos servirão de base para a construção e avaliação nosso modelo. A intuição é que ao avaliar todos esses conhecimentos e combiná-los como outras técnicas poderá ser possível gerar um trabalho robusto na tarefa de identificação de falhas em processos produtivos complexos.

2.1.1 Modelos Supervisionados

Este tipo de modelo utiliza um conjunto de dados de treinamento com um par de características e respostas rotuladas ou valores de destino [33]. Os exemplos de tarefas que os modelos de aprendizado supervisionados executam são tarefas de classificação e regressão. Existe a tendência a se referir a problemas com resposta quantitativa como problemas de regressão, enquanto aqueles que envolvem uma resposta qualitativa são frequentemente chamados de problemas de classificação [32].

A tarefa de classificação está associada a geração de padrão de dados em categorias distintas, como identificar se um *e-mail* é spam ou não, ou segmentar clientes em diferentes grupos de acordo com seus hábitos de compra. Algoritmos como árvores de decisão 2.2, regressão logística e *Support Vector Machines (SVM)* brilham nesse contexto, dividindo os dados em classes com base em suas características e padrões [18]. A capacidade de categorizar dados com precisão abre um leque de possibilidades para diversas aplicações.

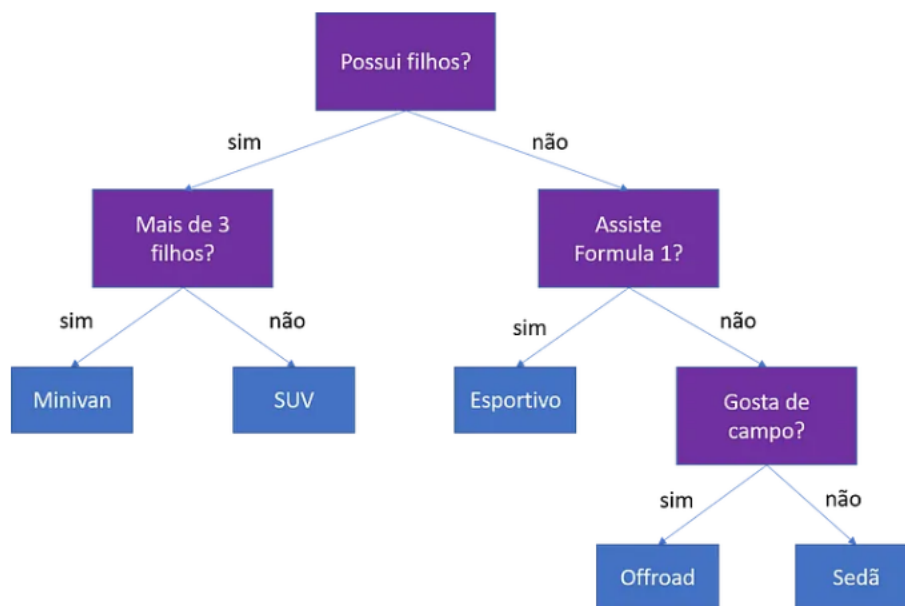


Figura 2.2: Exemplo de Algoritmo de Árvore de Decisão

Na regressão supervisionada, o objetivo central reside na predição de valores numéricos contínuos, como o preço de um imóvel ou a temperatura no dia seguinte. O modelo, guiado por um conjunto de dados rotulados com exemplos de valores de entrada e seus respectivos valores de saída, aprende a mapear as características de entrada para um valor numérico de saída. Técnicas como regressão linear, *K-Nearest Neighbors (KNN)* se consolidam como ferramentas valiosas para essa tarefa, permitindo a construção de modelos robustos e adaptáveis a diferentes cenários.

O aprendizado de máquina supervisionado encontra utilidade em diversos campos, tais como: Previsão do Preço de Imóveis, Detecção de Spam, Diagnóstico de Doenças, Previsão da Demanda, Reconhecimento de Imagens e diversas outras aplicações. Sua capacidade de aprender com exemplos rotulados o torna uma ferramenta poderosa para resolver problemas complexos e tomar decisões inteligentes. A ampla gama de algoritmos e técnicas disponíveis permite a construção de modelos personalizados para cada desafio, impulsionando a inovação e o avanço do conhecimento em diferentes áreas.

2.1.2 Modelos Não Supervisionados

Este tipo de modelo detecta padrões sem quaisquer rótulos pré-existentes ou especificações [11]. Essa liberdade de não depender de rótulos torna os métodos não supervisionados especialmente úteis em situações onde os dados não são rotulados ou onde a tarefa de rotulagem é custosa ou inviável. Além disso, esses modelos frequentemente empregam técnicas eficientes de redução de dimensionalidade, como a extração de características significativas de um espaço dimensional alto para um espaço dimensional menor [3], proporcionando uma representação mais compacta e interpretável dos dados.

Essa flexibilidade e eficiência na aprendizagem de representações [69] são características-chave do aprendizado não supervisionado, tornando-o uma ferramenta valiosa em uma variedade de aplicações, desde análise exploratória de dados até reconhecimento de padrões e segmentação de mercado.

O aprendizado não supervisionado, em particular a técnica de clusterização, desempenha um papel crucial na análise exploratória de dados e na descoberta de estruturas subjacentes em conjuntos de dados não rotulados. A clusterização visa agrupar os dados em *clusters* ou grupos que compartilham características semelhantes, permitindo a identificação de padrões intrínsecos nos dados. Esses *clusters* podem representar diferentes categorias, classes ou grupos de comportamento, dependendo do domínio da aplicação. Algoritmos de clusterização, como o *k-means*, *hierarchical clustering* e *DBSCAN*, são frequentemente utilizados para segmentar os dados e identificar agrupamentos naturais, oferecendo *insights* valiosos para a compreensão da estrutura subjacente dos dados [74].

A redução de dimensionalidade é uma técnica essencial no campo do aprendizado não supervisionado, buscando simplificar conjuntos de dados complexos e de alta dimensionalidade, preservando ao máximo suas características informativas. Por meio de algoritmos como Análise de Componentes Principais (PCA), *T-Distributed Stochastic Neighbor Embedding (t-SNE)* e *Autoencoders*, a redução de dimensão visa projetar os dados em espaços de menor dimensão, mantendo o máximo possível de variação original.

Essa técnica permite uma representação mais compacta e interpretável dos dados, facilitando a visualização, análise e interpretação dos padrões subjacentes. Além disso, a redução de dimensionalidade pode ser útil para mitigar o problema da maldição da dimensionalidade [38], melhorando a eficiência computacional e evitando overfitting em modelos de aprendizado de máquina. Assim, a redução de dimensão desempenha um papel crucial na simplificação e interpretação de conjuntos de dados complexos, contribuindo para uma compreensão mais profunda e eficaz dos fenômenos estudados.

2.1.3 Modelos de Aprendizado por Reforço

Nesse tipo de modelo agentes inteligentes interagem com o ambiente para cumprir uma meta de longo prazo [47]. Em vez de fornecer pares de entrada e saída, é descrito o estado atual do sistema, especificada uma meta, fornecida uma lista de ações permitidas e restrições e o modelo busca atingir a meta usando o princípio de tentativa e erro para maximizar uma recompensa [33].

Ao contrário do aprendizado supervisionado, nesses modelos descreve-se o estado atual do sistema, especifica-se uma meta ou objetivo a ser alcançado e fornece-se uma lista de ações possíveis e suas respectivas restrições [56]. O agente, então, busca maximizar uma recompensa ao longo do tempo, realizando tentativas e erros para aprender uma política de ação ótima que o leve a atingir o objetivo desejado [35]. Esse processo de interação agente-ambiente, baseado em *feedback* de recompensa, é fundamental para o desenvolvimento de estratégias eficazes de tomada de decisão em ambientes complexos e dinâmicos.

O aprendizado por reforço tem sido aplicado com sucesso em uma variedade de domínios, desde jogos de tabuleiro até controle de robótica, destacando-se como uma abordagem poderosa para resolver problemas de tomada de decisão sequencial em ambientes complexos e incertos.

2.2 Interpretabilidade de Modelos

O aprendizado de máquina interpretável é um termo um tanto quanto genérico mas útil que pode ser entendido como a extração de conhecimento relevante de um modelo de aprendizado de máquina sobre relacionamentos contidos em dados ou aprendidos pelo

modelo [45]. Quanto maior a interpretabilidade de um modelo de aprendizado de máquina, mais fácil é para entender como as decisões e previsões o modelo toma.

Além de confiar em previsões individuais, também é necessário avaliar o modelo como um todo antes de começar a usá-lo com dados diferentes dos dados utilizados no modelo. Para ter essa resposta, os responsáveis pelo modelo precisam ter certeza da eficiência do modelo com dados reais utilizando métricas relevantes [10]. Atualmente, os modelos são avaliados usando métricas de precisão em um conjunto de dados de validação disponível [29]. No entanto, os dados do mundo real costumam ser significativamente diferentes e, além disso, a métrica de avaliação pode não ser indicativa do objetivo do produto [49].

Para interpretar como os modelos correspondem as entradas dadas, temos duas abordagens principais. A interpretação local [55] que visa entender como cada característica individual contribui para a previsão. A interpretação global busca entender como o modelo funciona como um todo e quais tendências o modelo usa para fazer as previsões o que pode ser difícil caso o modelo seja muito complexo [48].

Atualmente existem diversos métodos para interpretação dos diversos modelos de aprendizado de máquina. Nesse sentido de interpretação de modelos, os mesmos podem ser separados em Modelos Intrinsecamente Interpretáveis e Modelos de Caixa Preta como é mostrado na imagem 2.3.

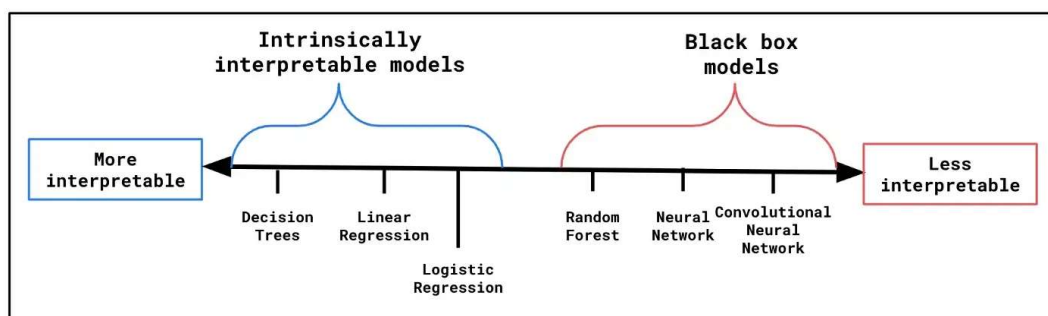


Figura 2.3: Modelos Intrinsecamente Interpretáveis *versus* Modelos de Caixa Preta. Ilustração retirada do artigo [46]

Os modelos chamados de Intrinsecamente Interpretáveis são aqueles cujo a interpretação do resultado é de fácil entendimento ao ser humano. Ou seja, de acordo com a sua própria construção a interpretabilidade é facilitada [15], um exemplo muito conhecido é o modelo de árvore de decisão, onde as condições referentes as características do conjunto de dados são avaliadas e os níveis da árvore são construídos [12].

Os modelos de caixa preta são muito complicados para serem entendidos diretamente por humanos. Para entender um modelo *Random Forest*, seria necessário entender simultaneamente todas as árvores de decisão. Da mesma forma, uma rede neural terá muitos parâmetros para compreender de uma só vez. Precisamos de métodos adicionais para examinar a caixa preta [46].

Para poder interpretar melhor os modelos de caixa preta, geralmente utiliza-se métodos de modelos agnósticos para auxiliar na interpretabilidade desses modelos. Os métodos mais comuns são os *PDPs* e *ICE Plots*, métodos utilizados para interpretação global. Enquanto *SHAP* (*Shapley Additive exPlanation*) e *LIME* (*Local interpretable model-agnostic explanations*) são utilizados para interpretação das previsões de modelos individuais com base na aproximação local do modelo em torno de uma determinada previsão [41].

O *SHAP* atribui a cada *feature* um valor de importância para uma previsão específica. Seus componentes incluem: a identificação de uma classe de medidas de importância de características e resultados mostrando que existe uma solução única em tal classe com um conjunto de propriedades desejáveis [41].

O *LIME* é um algoritmo que pode explicar as previsões de qualquer classificador ou regressor de forma fiel, por aproximação localmente com um modelo interpretável [49].

Uma outra maneira de interpretar o resultado dos modelos é utilizando técnicas contrafactuais. As explicações contrafactuais são um dos métodos mais populares para fazer previsões de modelos de aprendizado de máquina de caixa preta interpretáveis, fornecendo explicações na forma de "cenários hipotéticos" [9]. Essas técnicas buscam explicar por que um determinado resultado não foi alcançado, ajudam a compreender potenciais oposições de um dado resultado e fornecem orientação sobre como a previsão desejada pode ser alcançada no futuro [43].

Esse tipo de técnica contrafactual tem como particularidade encobrir o desafio de explicar o funcionamento interno de um modelo de aprendizado de máquina [6]. Por outro lado, tais técnicas fornecem informações ao titular dos dados que são facilmente digeríveis e úteis na prática para entender os motivos de uma decisão, contestá-los e alterar o comportamento futuro para um melhor resultado [63].

2.2.1 SHAP

A interpretação dos modelos de aprendizado de máquina oferece uma nova maneira de aplicar essas técnicas de predição. Com modelos interpretáveis, os profissionais têm acesso a informações adicionais que ajudam a tomar decisões mais informadas sobre os



Figura 2.4: Exemplo simples de como *SHAP* ajuda na interpretação da predição de modelos.

problemas que enfrentam, e principalmente ajudando a entender o próprio *dataset*.

O *SHAP* (*Shapley Additive Explanations*) é uma das ferramentas mais usadas para explicar modelos complexos, que são comumente considerados caixas pretas pela falta de transparência. Com o *SHAP*, é possível entender e interpretar 2.4 modelos estatísticos como *XGBoost*, *LightGBM* e redes neurais convolucionais (*CNN*), facilitando a compreensão das predições dos modelos para seres humanos [41].

Como abordam os autores [27], a natureza agnóstica a modelo do *SHAP* oferece liberdade na escolha do algoritmo de aprendizado de máquina, permitindo priorizar a precisão enquanto ainda fornece *insights* detalhados sobre esses modelos específicos de local, fornecendo tanto explicações locais de pontos de operação individuais quanto interpretações globais dos modelos.

O *SHAP* tem sido amplamente aplicado em uma variedade de domínios para explicar os resultados de modelos, tanto localmente quanto globalmente.

No entanto, há alguns pontos importantes que os usuários finais devem estar cientes ao aplicar o *SHAP*. Primeiro, o *SHAP* é um método dependente de modelo. Isso quer dizer que o resultado do *SHAP* depende do modelo de aprendizado de máquina usado para a tarefa de classificação/regressão, o que pode levar a diferentes escores de explicabilidade. Assim, quando diferentes modelos são aplicados à mesma tarefa usando os mesmos dados, as principais características identificadas pelo *SHAP* podem variar entre os modelos de aprendizado de máquina [53].

O *SHAP* mede o impacto que cada uma das *features* tem no valor decisivo do modelo. A ferramenta utiliza os valores de *Shapley* que são uma média ponderada da contribuição marginal de cada uma das características [51].

A figura 2.5 representa um gráfico *SHAP* para previsão dos preços de moradias na Califórnia a partir de um *dataset* público conhecido. Cada um dos pontos é uma amostra com o valor de suas características. Os valores *SHAP* auxiliam a mostrar a distribuição dos impactos de cada característica na saída do modelo. A cor representa o valor da

característica (vermelho para alto, azul para baixo). Isso revela, para esse exemplo, que rendas medianas mais altas melhoram o preço de casa previsto.

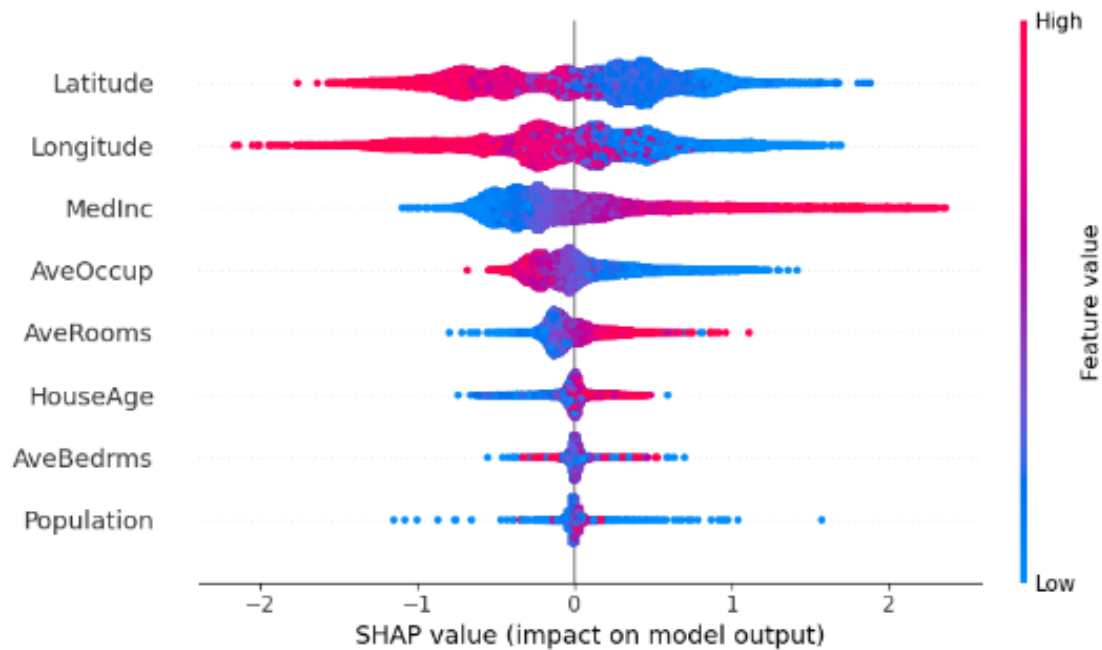


Figura 2.5: Gráfico resumido das importâncias das *features* na predição do modelo de predição de valores de moradia

2.3 Abordagem Orientada a Dados

A abordagem orientada a dados é cada vez mais reconhecida como fundamental para o desenvolvimento de sistemas eficazes e inovadores, a demanda por conjuntos de dados relevantes e abrangentes torna-se imperativa [8]. A abordagem centrada em dados entende que os dados são o recurso mais valioso para impulsionar o desenvolvimento de modelos e soluções inteligentes.

Os dados tem papel importante no ciclo de aprendizado de máquina. Tal ciclo é composto por diversas etapas as quais são comumente associadas característica de serem centrada em dados, centrada em problema e centradas em modelo. Até agora, a pesquisa em aprendizado de máquina tem sido predominantemente centrada no modelo e focada no design e na avaliação do modelo. Esta predominância enfatiza principalmente a otimização da precisão e eficiência dos próprios modelos e enxerga o conjunto de dados mais como um *benchmark* estático do que uma representação dinâmica da aplicação.[50]

A qualidade dos dados pode ser compreendida de diferentes pontos de vista. No entanto, a qualidade dos dados depende da qualidade das informações contidas [21]. Tradicionalmente, a pesquisa sobre qualidade de dados tem sido mais direcionada em dados operacionais. No entanto, no aprendizado de máquina, frequentemente os dados não estão em uma forma adequada para aprendizagem. Variáveis/características são identificadas e extraídas dos dados brutos. Embora as características tendam a ser específicas do domínio, há uma necessidade de estabelecer padrões genéricos que ajudem a identificar as características [26].

Atualmente existem diversas técnicas e métodos utilizados para extrair *insights* significativos de conjuntos de dados. Seja para utilizar em aprendizado de máquina, aprendizado profundo ou mineração de dados, os conceitos são aplicados para analisar e interpretar informações de maneira eficaz. Uma dessas técnicas é a *Data Augmentation* (Aumento de dados), técnica para aumentar o tamanho e a diversidade dos dados, criando artificialmente variações dos dados existentes, o que frequentemente pode melhorar o desempenho do modelo. [71]

Um conceito interessante associado a abordagem centrada a dados é o de aceleração de dados, cujo objetivo é construir uma infraestrutura eficiente de dados para facilitar a rápida aquisição de dados.[70]

Por outro lado, é necessário dar importância a gestão eficaz dos dados para garantir a qualidade e a confiabilidade dos resultados da IA. Além disso, existem desafios e oportunidades associados à abordagem centrada em dados, incluindo aspectos de conhecimento de domínio do problema, sociais e técnicos [31] que surgem no processo de manipulação e análise de dados.

2.4 Séries Temporais

Dados do tipo de séries temporais são definidos como um conjunto de observações sobre uma variável, ordenado no tempo e registrado em períodos regulares. Essa caracterização pela ordem sequencial e dependências temporais, encapsulam informações valiosas sobre a dinâmica de diversos sistemas e processos [13].

Diversos tipos de dados de séries temporais, como preço de ações 2.6, fluxo de tráfego e eletricidade, trazem desafios e oportunidades únicos para a análise computacional, exigindo abordagens específicas para entender suas características. A análise e compreensão de séries temporais é uma parte importante da mineração de dados, facilitando percepções e decisões cruciais em muitos domínios [65], incluindo finanças, saúde, computação em nuvem e energia.

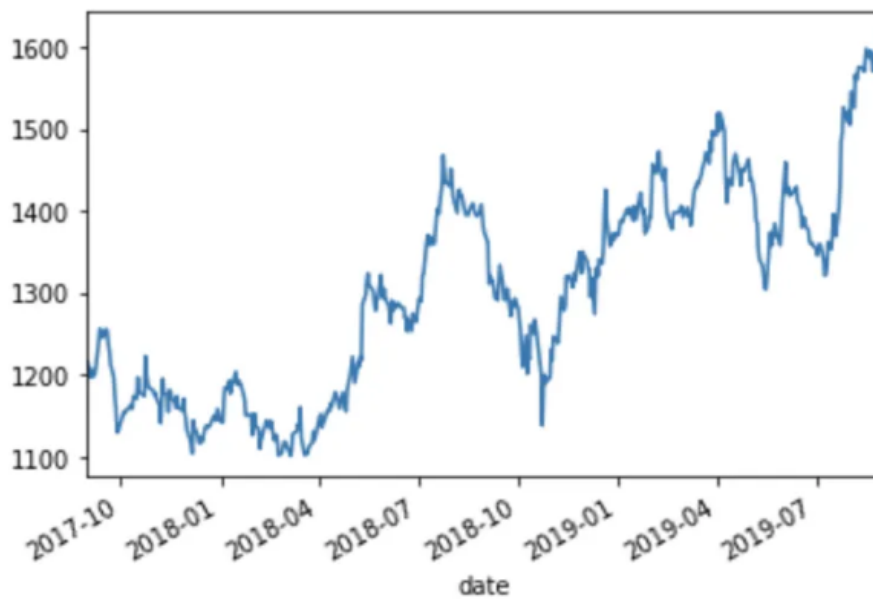


Figura 2.6: Exemplificação de séries temporais sobre preço de ações aplicadas ao longo de um período de tempo

Outro ponto importante relacionado a séries temporais é a detecção de anomalias, tal abordagem tem sido reconhecida como de importância crítica para a operação confiável e eficiente de sistemas do mundo real [72]. Muitos métodos de detecção de anomalias foram desenvolvidos com base em diferentes suposições sobre as características das anomalias.

No entanto, como apresentam [52], devido à natureza complexa dos dados do mundo real, diferentes anomalias dentro de uma série temporal geralmente apresentam perfis variados que se alinham a diferentes suposições de anomalias. Isso torna difícil encontrar um único detector de anomalias que possa consistentemente superar outros modelos.

O objetivo principal da análise de séries temporais é entender as relações entre os pontos de dados e incluir esses padrões nas previsões geradas. [19] e [60] destacam a importância da identificação e modelagem de componentes como tendência, sazonalidade e padrões cíclicos para uma análise precisa.

A modelagem de séries temporais envolve a seleção e aplicação de modelos estatísticos adequados para capturar a estrutura subjacente dos dados. Os autores [5] mencionam métodos autoregressivos, de médias móveis e suas combinações, como *ARIMA*, destacando a importância da avaliação de modelos por meio de ferramentas como funções de autocorrelação e critérios de informação. Essa abordagem estatística permite não apenas entender os padrões presentes nos dados históricos, mas também fazer previsões precisas sobre o comportamento futuro da série.

No entanto, as séries temporais frequentemente apresentam diversos desafios, como a presença de *outliers*, ruído e não estacionariedade. Para tratar tais empasses, técnicas

avanzadas, como modelos de suavização exponencial e decomposição sazonal, podem ajudar a lidar com tais complexidades [30]. Além disso, métodos baseados em aprendizado de máquina, como redes neurais e árvores de decisão, têm sido cada vez mais explorados para modelagem e previsão de séries temporais em contextos específicos.

À medida que a análise de séries temporais continua a evoluir, é relevante a integração de abordagens teóricas e práticas para lidar com problemas do mundo real [59]. Essa abordagem interdisciplinar não apenas promove uma compreensão mais profunda dos processos subjacentes aos dados, mas também capacita os profissionais a tomar decisões informadas e estratégicas em diversas áreas de aplicação. Em resumo, o estudo de séries temporais não apenas proporciona *insights* valiosos sobre o comportamento temporal dos fenômenos, mas também fornece ferramentas poderosas para previsão e tomada de decisões em cenários de constante mudança.

2.5 Trabalhos Relacionados

A aplicação de séries temporais na predição de defeitos e manutenção de equipamentos tem recebido crescente atenção devido à sua capacidade de identificar padrões de falha e antecipar necessidades de manutenção, contribuindo assim para a redução de custos e aumento da eficiência operacional.

A predição de falhas em processos produtivos complexos se configura como um desafio crucial para a indústria moderna, exigindo ferramentas robustas e precisas para garantir a otimização da produção, a segurança dos trabalhadores e a redução de custos. As séries temporais, caracterizadas por dados coletados ao longo do tempo, emergem como um aliado poderoso nessa jornada, possibilitando a identificação de padrões e anomalias que podem indicar falhas iminentes.

A Indústria 4.0 utiliza *IoT* e outras tecnologias de manutenção preditiva para melhorar os processos de negócios e obter vantagem competitiva. Nesse cenário, a Detecção e Diagnóstico de Falhas em Tempo Real (*RT-FDD*) é fundamental para aumentar a confiabilidade dos sistemas de produção, evitando quebras.[40]

Ainda sobre o tema de *IoT*, tais integrações de dados de séries temporais com informações de Internet das Coisas (*IoT*) e monitoramento remoto, fornecem às organizações a capacidade de desenvolver uma vantagem competitiva sustentada, uma vez que a crescente quantidade de dados e dados com cada vez mais qualidade tem a capacidade de gerar *insights* significativos [19]. A utilização de dados de sensores de séries temporais para monitorar a condição do chão de fábrica, destacam a importância da coleta e análise contínua de dados para garantir a segurança e confiabilidade de sistemas

críticos.

Além disso, a utilização de técnicas de séries temporais na manutenção preditiva de equipamentos é amplamente relevante. Uma vez que artigos como [75] e [68] destacam a importância da análise contínua de dados de sensores para monitorar o desempenho de equipamentos, identificar tendências de degradação e prever falhas com antecedência, permitindo intervenções proativas de manutenção.

Como apresentam os autores [58], a grande quantidade de dados gerados por plantas industriais possibilita o uso de *Machine Learning (ML)* para entender melhor o comportamento das operações e auxiliar os operadores na gestão eficiente. A detecção de anomalias é um exemplo de como *ML* pode identificar se uma planta está funcionando normalmente ou se precisa de ações corretivas. Contudo, a falta de interpretabilidade dos modelos de *ML* é uma queixa comum. Para resolver isso, o campo de Inteligência Artificial Explicável (*XAI*) tem ganhado destaque. O estudo investiga várias técnicas de *XAI* para detectar anomalias em dados de séries temporais multivariadas, predominantes em sistemas industriais, usando *autoencoders*. Entre as sete técnicas desenvolvidas, um explicador baseado em *SHAP*, chamado *DTFS*, se destacou por identificar corretamente a causa das anomalias com 86% de precisão e em 1,53 segundos.

A necessidade de abordagens personalizadas e adaptativas para diferentes tipos de equipamentos e setores industriais, ressalta o potencial das séries temporais na otimização de estratégias de manutenção e na maximização da vida útil dos ativos. Em harmonia com o que pensam os autores [62], a previsão de falhas em dados multivariados industriais é crucial para a implementação de estratégias eficazes de manutenção preditiva para reduzir o tempo de inatividade e aumentar a produtividade e o tempo operacional. Contudo, atingir esse objetivo utilizando aprendizado de máquina pode ser um desafio e requer uma compreensão profunda dos dados de entrada.

Ainda tratando de dados multivariados, [2] discutem em seu artigo como as várias novas abordagens de detecção de anomalias em séries temporais multivariadas baseadas em *deep learning* foram desenvolvidas, reduzindo drasticamente a quantidade de computação necessária. Os pesquisadores ponderam como os dados industriais são amostrados com alta frequência, diferentes tamanhos de janela e são considerados para avaliar seu efeito no desempenho dos modelos de *deep learning*.

Os autores [24] utilizaram a ferramenta para abordar o problema da interpretabilidade de um modelo de aprendizado de máquina desenvolvido para prever séries temporais de qualidade do ar. O artigo demonstra como as explicações aditivas de *Shapley* podem ser usadas para interpretar as saídas de uma rede neural profunda projetada para prever as concentrações de dióxido de nitrogênio em Madrid.

O artigo elaborado por [37] buscar entender como o aumento do calor causado pelas mudanças climáticas piorou o ambiente térmico urbano e prejudicou a saúde humana, resultando em mortalidade relacionada ao calor. Os resultados da estimativa do modelo

foram interpretados a partir das perspectivas global e local, utilizando o método *SHAP*. Como resultado da interpretação, os setores demográfico, socioeconômico e climático foram identificados como os que mais contribuíram para o processo de estimativa. Esse é um texto interessante sobre como a interpretação de um modelo de aprendizado de máquina pode ajudar na análise de mortalidade relacionada ao calor.

Uma outra área interessante que a ferramenta *SHAP* está sendo utilizada é na área de veículos elétricos. Os autores [61] utilizaram a abordagem de explicação aditiva de *Shapley* (*SHAP*) para lidar com os problemas de não interpretabilidade do algoritmo de aprendizado de máquina, interpretando as saídas do modelo *XGBoost*. Os gráficos de valores *SHAP* obtidos demonstraram a relação não linear entre as variáveis explicativas e o tempo de carregamento de veículos elétricos.

Outro ponto que pode dificultar o desenvolvimento de modelos de aprendizado de máquina é que séries temporais industriais frequentemente incluem intervalos de tempo irregulares e valores ausentes, devido a sensores ou infraestrutura de coleta de dados mal configurados ou com falha, ou devido a falta de manutenção e atualização. Portanto, conforme [34] citam, não é sempre claro se as informações necessárias para previsão precisa de falhas está presente nos dados e o que a precisão da previsão é alcançável com os dados.

O artigo desenvolvido por [66] traz uma contribuição dupla interessante. Tais pesquisadores discutem os princípios do aprendizado de máquina supervisionado e dados em formato de séries temporais para a detecção de descargas parciais (*PD*) em linhas de energia, que podem levar a interrupções de energia ou até causar incêndios. Os autores realizam um estudo comparativo detalhado entre vários sistemas de detecção de *PD* baseados em dois paradigmas de aprendizado: modelos de sequência e aprendizado em conjunto. A conclusão final do artigo leva a percepção que focar na melhoria da qualidade dos dados desempenha um papel crucial na criação de modelos detectores de falhas indesejadas.

Um assunto pertinente ao tema tratado na dissertação é a Análise Exploratória de Dados (*EDA*), mencionada no artigo de [42] como uma etapa crucial para identificar falhas, oferecendo uma visão inicial dos possíveis problemas sem necessariamente identificar suas causas específicas. Essa abordagem orienta os diagnósticos subsequentes ao destacar áreas de preocupação. Em um contexto onde a rápida detecção de falhas pode evitar custosos períodos de inatividade ou danos significativos, a importância da *EDA* é evidente. O estudo também conclui que a integração do aprendizado de máquina com sistemas complexos pode aprimorar a segurança e a confiabilidade das operações envolvidas.

Esses artigos fornecem percepções valiosas sobre o uso de séries temporais na predição de defeitos e manutenção de equipamentos, demonstrando o papel crucial dessa abordagem na melhoria da confiabilidade e eficiência dos sistemas industriais e de infraestrutura.

Capítulo 3

Modelo para Predição de Quebra de Vácuo

Nesse capítulo são descritos os detalhes de implementação da metodologia desenvolvida. Também serão descritos como foram abordados os procedimentos de implementação do modelo de predição de quebra de vácuo. Primeiramente, detalhamos a estrutura do modelo de aprendizado de máquina adotado, o LightGBM, e as técnicas de pré-processamento aplicadas aos dados, incluindo a seleção e agregação de características relevantes. Em seguida, destacamos uma contribuição crucial deste estudo: a definição da variável alvo, que representa o tempo até a falha, permitindo uma abordagem proativa na identificação de padrões precedentes.

Por fim, explicamos a aplicação do modelo de aprendizado de máquina ao conjunto de dados processado, incluindo a otimização de hiperparâmetros usando a biblioteca Optuna. Esses passos, descritos na figura 3.1 são essenciais para a construção de um modelo eficaz na previsão de quebra de vácuo no processo de desodorização de produtos.

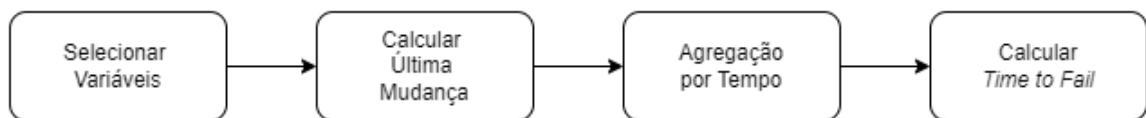


Figura 3.1: Fluxograma mostrando os passos de pré-processamento

3.1 Coleta de Dados

A base de dados utilizada no trabalho foi gentilmente fornecida pela empresa M. Dias Branco consiste em informações de sensores de uma das máquinas responsáveis pelo

processo de desodorização dos produtos fabricados pela empresa.

Os dados foram coletados internamente pela empresa compreendem o período de 2023-03-06 00:00:00 a 2023-05-22 23:59:57. A frequência de coleta dos dados, realizada a cada 3 segundos, é crucial para acompanhar a evolução do desempenho da máquina ao longo do tempo.

O conjunto de dados utilizado é comumente denominado como séries temporais e consiste em 103 campos dispostos em diversos tipos, categóricos e numéricos e 1.949.816 registros. Como descrito anteriormente, tais campos são relativos aos sensores do equipamento e significam valores de medidores, temperaturas de sensores, pH interno, quais tipos de produto estavam dentro da máquina.

Para garantir a confiabilidade dos dados coletados, serão implementadas técnicas de validação e pré-processamento, incluindo verificação de completude, detecção de anomalias, normalização de dados e validação cruzada. A fim de ampliar ainda mais a riqueza da base de dados e aprimorar a precisão dos modelos preditivos, serão avaliadas algumas possibilidades: monitoramento contínuo, integração de dados adicionais, agregação de dados e transformação de dados existentes.

A base de dados da M. Dias Branco, abrangente e rica em detalhes, oferece uma grande oportunidade para a construção de modelos preditivos precisos de quebras de vácuo na máquina de desodorização. Através da implementação de técnicas rigorosas de coleta, pré-processamento e análise de dados, aliadas à exploração de fontes de dados adicionais e à implementação de novas tecnologias de monitoramento, espera-se alcançar um profundo conhecimento do processo de desodorização e desenvolver um modelo robusto que contribuirá para a otimização da produção, a minimização de falhas e o aumento da eficiência da empresa parceira.

3.2 Pré-Processamento

Visando melhorar a qualidade do modelo final, foram utilizadas algumas técnicas de *feature engineering*. Tal processo usa conhecimento de domínio para extrair, transformar ou selecionar características relevantes (características, propriedades, atributos) de dados brutos e para compreender a essência dos dados em uma forma que seja benéfica para o processamento posterior [20].

Durante a etapa de pré-processamento do *dataset* foram realizadas diversas operações visando preparar os dados para a modelagem. Em consulta com especialistas da empresa parceira, foram identificadas colunas que tinham pouca relevância e que não contribuíam significativamente para o objetivo do estudo, portanto, tais colunas foram removidas do

dataset.

Nesta mesma etapa de preparação foram feitas remoções de algumas colunas que em contato com especialistas da empresa parceira verificamos que tais variáveis fariam pouco sentido para serem utilizadas no modelo.

A fim de investigar empiricamente o conjunto de dados, foi feita a exploração de variáveis pensando em momento de mudança. O processo se deu em calcular para algumas variáveis significativas qual era o valor anterior ao valor da mudança de parâmetro. Para explicitar de forma clara podemos pensar que uma variável significa um tipo de produto utilizado na máquina que em determinado tempo era X e trinta minutos após o produto utilizado era Y. Portanto, de forma experimental criamos novos atributos baseados em mudança de valores dos atributos já existentes.

Um dos pontos cruciais do trabalho foi fazer a agregação dos dados em faixas de tempo. Fizemos a agregação dos registros em agrupamentos de 15 segundos, 30 segundos e 60 segundos gerando novas colunas para cada coluna já existente, exemplificando, a coluna 'xxx' gerou três novas colunas etiquetadas com o nome de qual agrupamento de tempo a nova variável pertence.

Ao final da etapa de pré-processamento, foram geradas no total 940 colunas.

Algorithm 1 Pseudocódigo do Processo de Pré-Processamento das *features*

- 1: Renomear Variáveis
 - 2: Remoção de Variáveis com Pouco Significado
 - 3: Criar Variáveis de Última Mudança
 - 4: Calcular Nova Variável Alvo *Time To Fail*
 - 5: Criar Novas Variáveis a Partir de Agregação por Tempo
 - 6: Separar *dataset* em Treino e Validação
-

3.3 Definição da variável alvo

Inicialmente, foi considerado utilizar a variável alvo *PT_846F_2.EU* do conjunto de dados que significa a pressão de vácuo do desodorizador. No entanto, dada a natureza do problema e o objetivo de desenvolver um modelo supervisionado para prever o tempo até a falha, optou-se por criar uma nova variável alvo que indica o tempo restante até que ocorra uma falha. Esta abordagem permite direcionar o modelo para identificar padrões que antecedem as falhas, possibilitando uma intervenção proativa.

Após alguns estudos e conversas com os responsáveis pelo equipamento que tem domínio e conhecimento sobre o processo, definiu-se que valores acima de 10 para tal variável caracterizam uma quebra de vácuo. Portanto, o tempo para falha (*TTF*) foi

calculado considerando o momento em que a pressão interna se manteve abaixo de 10 até ser igual ou superior a 10.

No modelo de previsão de quebra de vácuo da M. Dias Branco, o *feature engineering* na variável alvo "tempo para falha" (TTF) é fundamental para extrair informações valiosas, identificar padrões ocultos e reduzir o ruído nos dados. Essa abordagem estratégica amplia a base de conhecimento do modelo, aprimora a precisão das previsões e possibilita a descoberta de *insights* relevantes [73], propiciando a otimização do processo de desodorização e a prevenção de falhas.

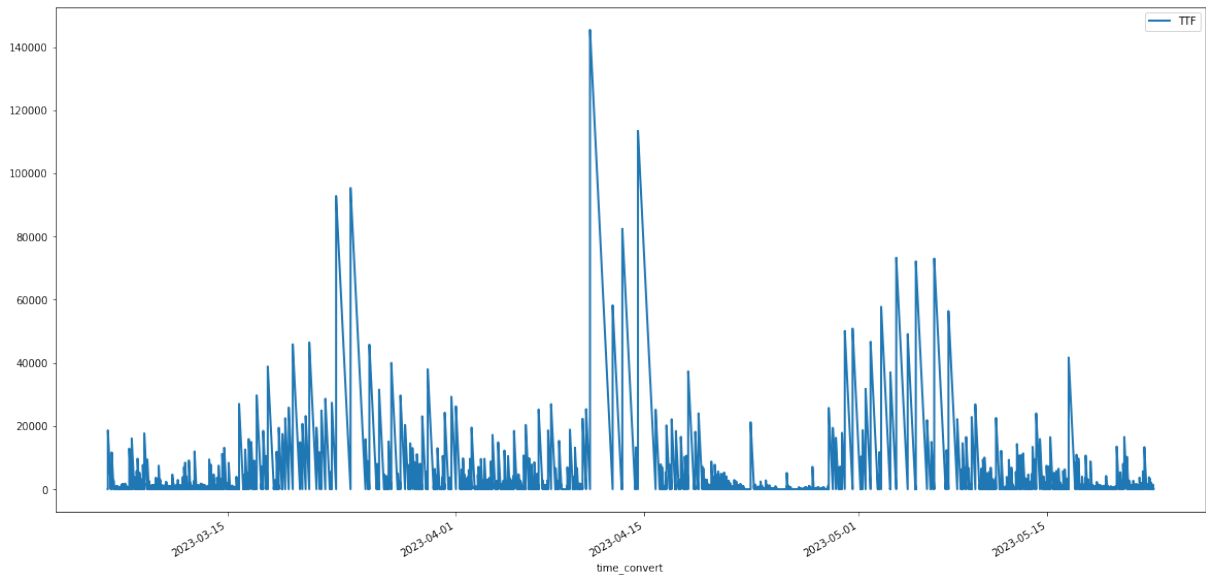


Figura 3.2: Gráfico com exemplo da variável de tempo para falha

3.4 Algoritmo

O algoritmo de filtragem desenvolvido realiza uma seleção iterativa de variáveis baseadas no valor de *SHAP*. Inicialmente, é criada uma lista vazia para armazenar variáveis a serem excluídas. Em um *loop* que se repete até 15 vezes, valor escolhido de forma arbitrária, o algoritmo treina um modelo e calcula os valores de *SHAP* para cada variável.

Em cada iteração, as variáveis com valores de *SHAP* abaixo de um determinado limite são identificadas e armazenadas, e essas variáveis são então excluídas em iterações subsequentes. O processo continua até que o número máximo de iterações seja atingido.

ou não existam mais características com valor de *SHAP* igual a 0.

Algorithm 2 Pseudocódigo do método de filtragem desenvolvido

```

1:  $VariáveisAExcluir \leftarrow ListaVazia$ 
2:  $N \leftarrow 0$ 
3: enquanto  $N \leq 15$  faça
4:   se  $N > 0$  então
5:      $VariáveisAExcluir \leftarrow VariáveisExcluidasAnteriormente$ 
6:   fim se
7:   Treinar o modelo
8:   Calcular variáveis com valor abaixo do limite SHAP
9:   Salvar variáveis
10: fim enquanto
  
```

A imagem 3.3 ajuda a entender de forma visual apresentando em forma de funil invertido a redução na quantidade de variáveis ao longo de várias iterações. No eixo vertical esquerdo, está o "Número de Iterações", indicando que o processo ocorre ao longo de várias etapas. No eixo vertical direito, a "Quantidade de Variáveis" mostra como essa quantidade diminui conforme o processo avança.

Tal figura auxilia no entendimento da seleção de variáveis, onde inicialmente se começa com um grande conjunto de variáveis, e à medida que o modelo é treinado, são identificadas e removidas aquelas que têm pouca ou nenhuma importância, resultando em um modelo mais enxuto e eficiente. A parte superior do funil representa o início do processo, com um grande número de variáveis, que vai diminuindo ao longo das iterações, conforme variáveis menos relevantes são eliminadas.

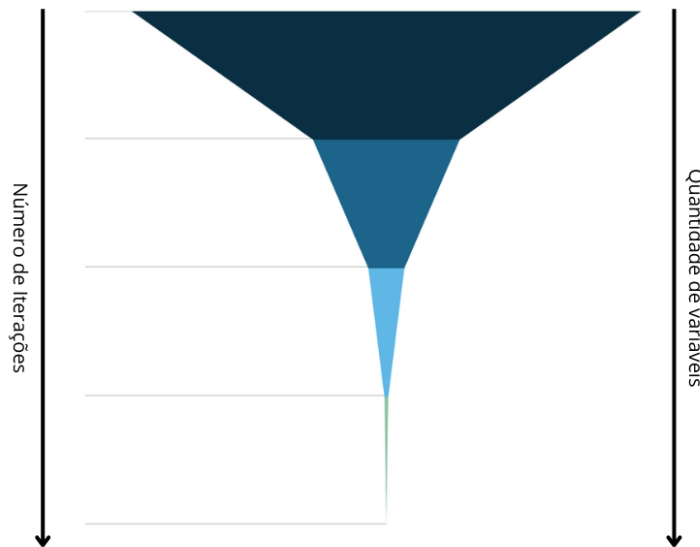


Figura 3.3: Representação em forma de funil do resultado *dataset*

3.5 Aplicação do Modelo de Aprendizado de Máquina

Após o processo de configuração do *dataset* e de processamento dos dados em forma de *pipeline*, o próximo passo é utilizar modelo de aprendizado de máquina previamente definido para encontrar os padrões de quebra de vácuo do desodorizador. A seleção do algoritmo utilizado para suportar a pesquisa foi feita após estudo em etapas conforme figura 3.4. Após eliminar abordagens incompatíveis para a resolução do problema, selecionar a técnica que pode prover desempenho para o algoritmo e escolher de fato o algoritmo que será utilizado para suportar a pesquisa, foi decidido, para o presente trabalho, o algoritmo *LightGBM*. Entende-se portanto, que o desempenho de um algoritmo representa o desempenho aproximado de toda a técnica [20].

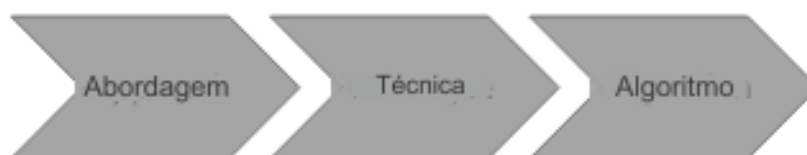


Figura 3.4: Processo de escolha de algoritmo

O *LightGBM* é um modelo de aprendizado de máquina do tipo supervisionado. Para caso específico tratado no trabalho foi o algoritmo utilizado no modo de regressão pois o resultado da predição criada não está de fato contida em classes ou categorias pré-determinadas.

A escolha do algoritmo *LightGBM* para desvendar os padrões da quebra de vácuo na máquina de desodorização da empresa parceira se baseou em seus atributos notáveis e na capacidade de superar desafios específicos do problema. Sua alta precisão e eficiência [36] garantem previsões confiáveis.

Para melhorar o desempenho do modelo e lidar com a distribuição assimétrica da variável alvo, foi aplicada a transformação logarítmica natural (*np.log*) durante o treinamento. Isso ajudou a estabilizar a variância e a reduzir a influência de *outliers*, facilitando o aprendizado do modelo. Após o treinamento, foi utilizada a transformação inversa (*np.exp*) nas predições para retornar os valores à sua escala original. Esse processo garantiu predições mais precisas e comparáveis diretamente com os valores reais.

A escalabilidade e velocidade excepcionais do *LightGBM* permitem lidar com o conjunto de dados massivo de forma eficiente, otimizando o tempo de processamento e garantindo a viabilidade da aplicação em cenários reais. Em suma, o algoritmo em questão se destaca como a escolha ideal pois provê alta confiabilidade, otimizando o processo

produtivo e prevenindo falhas dispendiosas.

A escolha da biblioteca *Optuna* para otimizar os hiperparâmetros do modelo de previsão de quebra de vácuo se baseia em sua capacidade de automatizar o processo de busca pela melhor configuração, otimizando o desempenho do modelo e maximizando sua precisão [1], garantindo a melhor configuração do modelo, evitando a necessidade de ajustes manuais demorados e trabalhosos, além de maximizar o desempenho do modelo.

A *Optuna* se destaca por sua flexibilidade e robustez, permitindo lidar com diversos tipos de modelos de aprendizado de máquina, incluindo o *LightGBM* utilizado neste trabalho. Através de um algoritmo de busca eficiente, a *Optuna* explora o espaço de hiperparâmetros de forma direcionada, evitando quedas em mínimos locais e garantindo a convergência para a melhor configuração possível.

Além disso, a *Optuna* oferece recursos avançados como poda de galhos e reinicialização de buscas, permitindo que o processo de otimização seja mais eficiente e evite o desperdício de tempo em áreas menos promissoras do espaço de busca [54].

Capítulo 4

Experimentos e Resultados

4.1 Configuração dos experimentos

Na configuração dos experimentos, é fundamental considerar a divisão adequada dos dados em conjuntos de treinamento e teste, especialmente em problemas de séries temporais. Neste estudo, o *dataset* modificado foi dividido em 90% para treinamento e 10% para validação, seguindo a ordem cronológica dos dados. Essa abordagem é crucial, pois reflete melhor a distribuição temporal dos dados e evita vazamento de informação do futuro para o modelo durante o treinamento. Tal técnica é essencial porque garante que o modelo seja avaliado em condições que simulem a realidade, prevendo o futuro com base no passado. Isso ajuda a garantir que o modelo seja robusto e confiável quando aplicado a dados futuros reais, o que é fundamental em muitas aplicações práticas.

A validação cruzada é uma etapa importante para avaliar a generalização do modelo. No caso do *LightGBM*, optamos pelo algoritmo *TimeSeriesSplit* da biblioteca *sklearn* com 5 dobras. Esse algoritmo fornece índices de treinamento/teste adequados para séries temporais como exemplificado na figura 4.1, garantindo que as amostras de dados sejam divididas corretamente em conjuntos de treinamento/teste, levando em consideração a ordem temporal das observações. Em cada iteração, usa-se uma parte do histórico como conjunto de treinamento e uma subsequente como conjunto de teste. Esse método imita o fluxo natural de tempo, onde o passado é conhecido e o futuro não.

Um conceito importante explorado neste estudo foi o de iterações que assemelham-se a "épocas". Além de derivar novas características dos dados iniciais, uma parte crucial do processo foi criar uma filtragem de variáveis para garantir a eficiência do modelo. Desenvolvemos o conceito de "épocas" utilizando o algoritmo *SHAP*, onde variáveis com baixa importância foram eliminadas do *dataset*. Isso resultou em um modelo mais leve e rápido, mantendo a relevância das variáveis mais importantes.

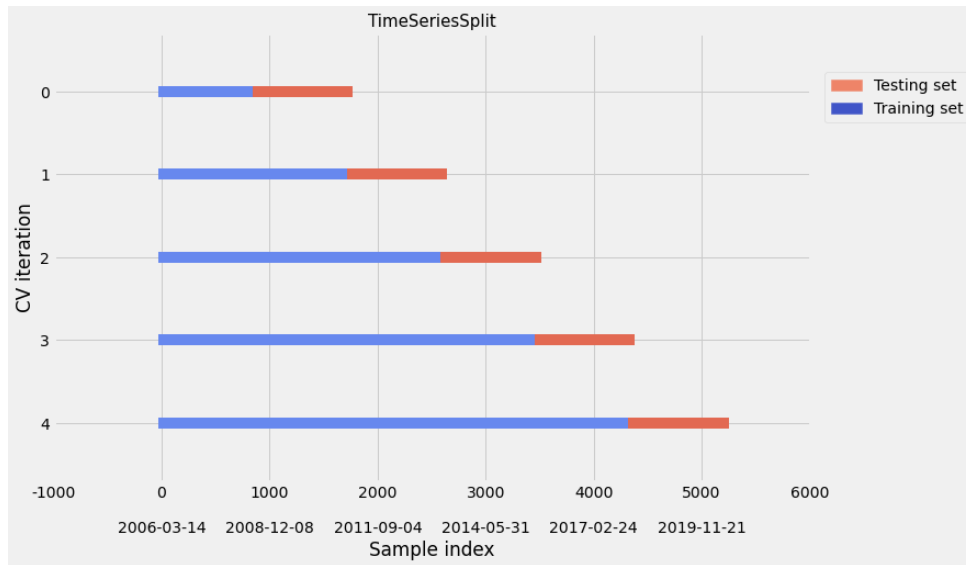


Figura 4.1: Exemplificação da biblioteca *TimeSeriesSplit*.

4.2 Discussão dos Resultados

Como pode ser visto na tabela 4.1 foram exploradas algumas medidas limites de valores do *SHAP* para tentar entender se haveria um limiar onde a métrica utilizada para avaliar o modelo destoaria muito do valor já encontrado sem a criação dos filtros.

Valor Limite <i>SHAP</i>	<i>MAE</i>	ÉPOCAS
0	774.98	5
25	585.35	2
50	628.54	2
100	646.47	2
200	800.82	2

Tabela 4.1: Tabela comparativa de experimentos

Inicialmente foi considerado o valor 0 do *SHAP* como marco para remover as *features* do conjunto de dados. O valor de *SHAP* (*SHapley Additive exPlanations*) igual a 0 indica que a variável correspondente não contribui para o desvio da previsão do modelo em relação ao valor médio esperado. Em outras palavras, o valor de *SHAP* zero sugere que a característica específica não tem impacto no resultado do modelo para aquela previsão específica. Isso significa que, para esse ponto de dados, a presença ou ausência da variável não altera a previsão do modelo, tornando-a irrelevante nesse contexto particular.

À medida que o valor limite aumenta, o erro médio absoluto também tende a crescer, indicando que restringir variáveis consideradas pouco úteis pode reduzir a assertividade do modelo. Na metodologia desenvolvida, os melhores resultados foram obtidos

quando o valor de *SHAP* foi igual a 0, o que sugere que essas características não influenciaram o resultado do modelo em previsões específicas, permitindo que o modelo mantivesse sua precisão ao se concentrar nas variáveis realmente relevantes.

Portanto, embora a eliminação de variáveis menos úteis possa simplificar o modelo, ela também pode comprometer sua assertividade, mostrando que o equilíbrio entre simplicidade e precisão é fundamental para o desempenho ideal do modelo.

Para avaliar o modelo foi utilizada a métrica *MAE* (*Mean Absolute Error*) e conforme é possível visualizar na figura 4.2 para o modelo criado com a estratégia proposta o valor do *MAE* foi o menor existente.

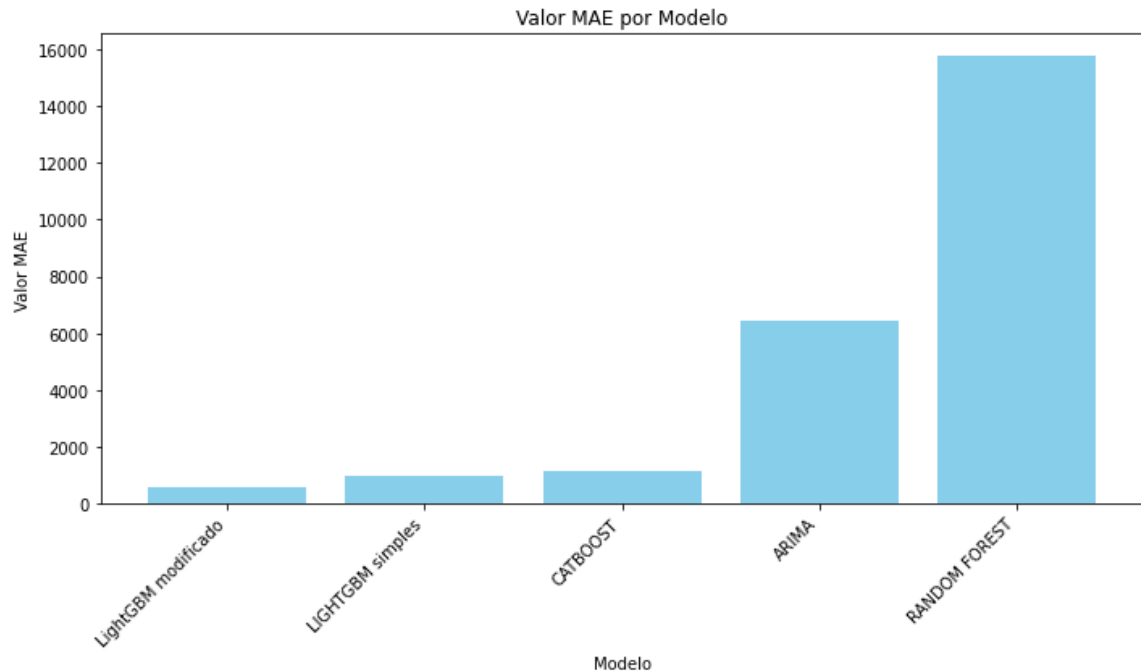


Figura 4.2: Comparação final de algoritmos

Como era esperado, o modelo desenvolvido usando o *LightGBM* e a abordagem de tratamento do *dataset* original obteve o menor valor de erro médio absoluto. Isso sugere que a estratégia adotada foi eficaz na predição do tempo até a falha da máquina.

Para poder visualizar melhor a eficiência do modelo, um gráfico 4.3 plotando os reais valores *versus* os valores preditos pelo modelo foi gerado. O Eixo X mostra a linha do tempo dos dados, um recorte do período de 06/03/2023 a 15/03/2023. O eixo Y mostra os valores do *TTF*, que variam em magnitude ao longo do tempo. As linhas azuis indicam os valores reais de *TTF* ao longo do tempo. As linhas vermelhas pontilhadas indicam os valores previstos pelo modelo. Estas previsões são mais suaves e em algumas momentos tem uma diferença dos valores reais.

O gráfico permite visualizar como o modelo de previsão se comporta em relação aos valores reais, mostrando a proximidade entre as previsões e os valores observados.

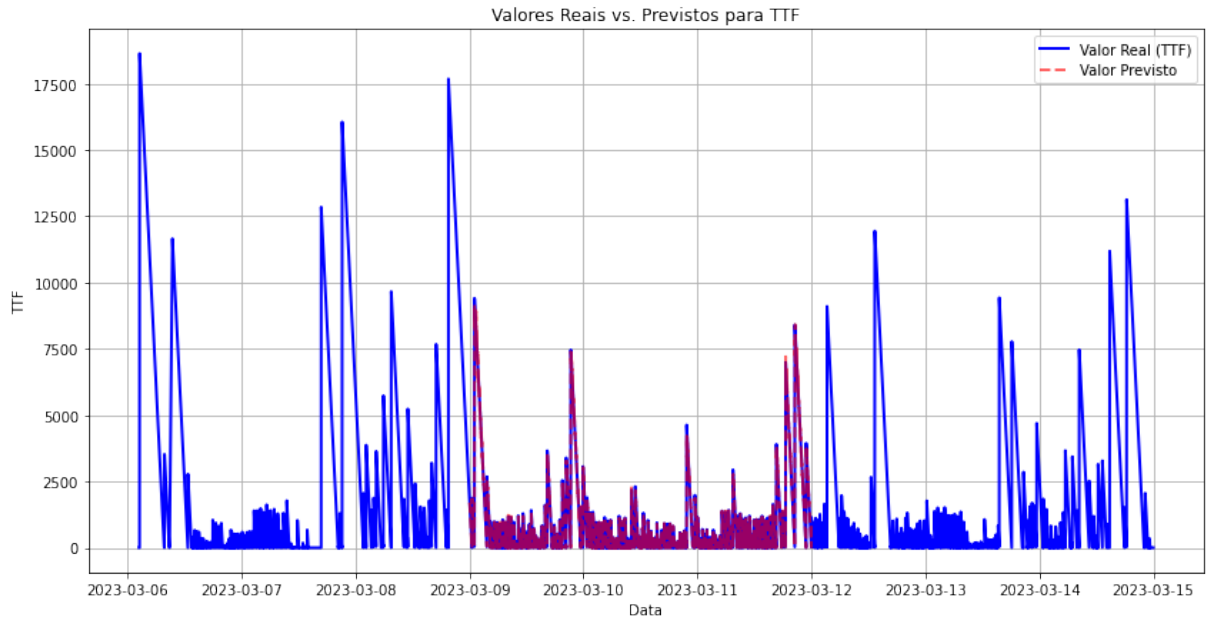


Figura 4.3: Comparação predição *versus* real

Como relatado anteriormente, a metodologia criada na dissertação utiliza como base os valores do *SHAP* para fazer a filtragem e remoção das colunas fora do limite considerado. O gráfico 4.4, conhecido como "*summary plot*", ajuda a entender como cada característica afeta a saída do modelo e qual a importância das características do modelo de predição. O eixo Y lista as características do modelo, cada linha representa uma característica diferente. O eixo X demonstra os valores *SHAP* que mostram o impacto de cada característica na predição do modelo. Tais valores, se positivos indicam que a característica aumenta a predição, enquanto valores negativos indicam que a diminui. A cor de cada ponto representa o valor da característica: vermelho indica um valor alto e azul indica um valor baixo. Cada ponto no gráfico representa um valor *SHAP* para uma observação específica no conjunto de dados. A dispersão dos pontos ao longo do eixo X mostra a variabilidade do impacto da característica em diferentes observações.

O gráfico 4.4 ilustra que as características de Sensor 2 Pressão de Vácuo nos tempos 15, 60 e 30 se destacam por ter um impacto significativo nas predições do modelo. Isto sugere que os valores de pressão do Sensor 2 são críticos para a precisão do modelo. A presença dessas características em múltiplos intervalos de tempo (15, 60, 30) indica que a pressão do vácuo é um fator altamente relevante e deve ser monitorada continuamente. De certo modo era esperado que tal característica tivesse forte influência no modelo pois foi a nova variável alvo era proveniente dessa característica.

—como falar q isso era de fato esperado?

Uma outra *feature* que também se mostrou relevante foi Vácuo Após *Scrubber* nos tempos 60, 15 e 30. Tais características inclusive demonstram uma influência importante, embora com uma menor variabilidade em comparação aos sensores de pressão de vácuo.

Não menos importante para o modelo é a *feature* Pressão Ejetor X nos tempos 30 e 60. Estas características mostram um impacto considerável no modelo, indicando que as variações na pressão do ejedor também são relevantes.

As demais características, como Pressão da Bomba da Caixa Barométrica e Pressão Ejetor Y em diversos tempos, tem menor impacto mas ainda sim suas devidas influências são relevantes. Elas não são tão críticas para a precisão do modelo quanto as características de maior impacto.

Em termos gerais, os padrões observados nos valores *SHAP* indicam que as características relacionadas à pressão de vácuo, especialmente dos sensores 1 e 2, são as mais influentes para o modelo de predição. As características relacionadas ao vácuo após o *scrubber* e à pressão do ejedor também são importantes, mas com um impacto mais consistente e previsível.

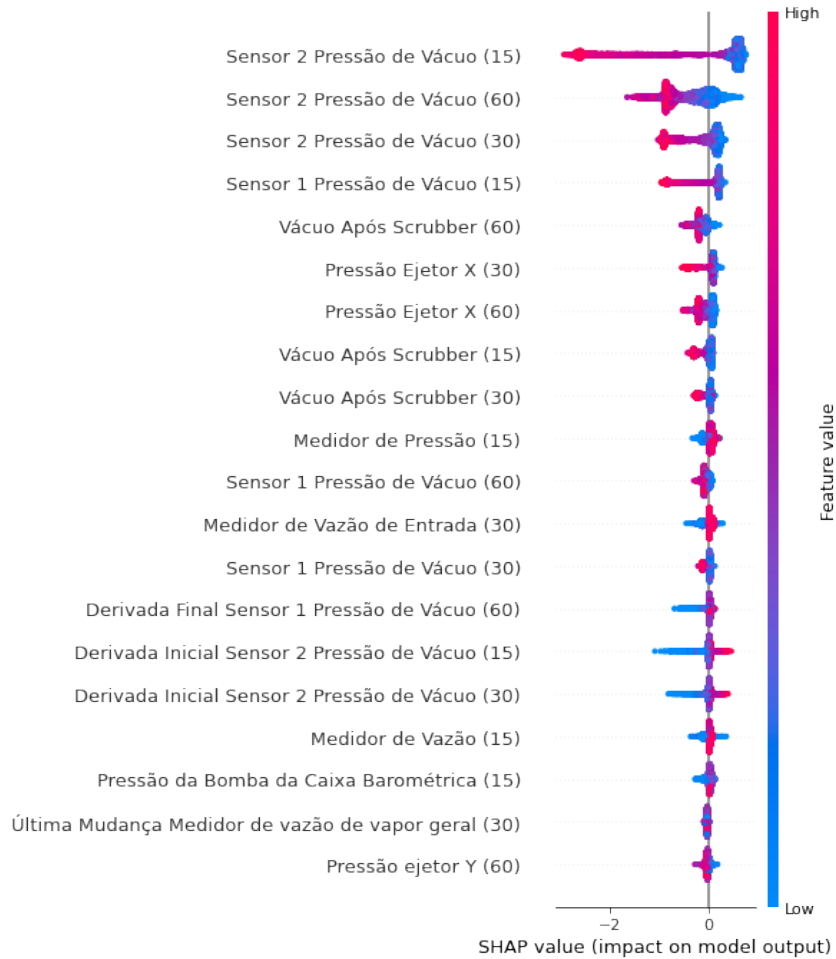


Figura 4.4: Valores *SHAP* das principais características

O histograma dos resíduos apresentado na figura 4.5 mostra a distribuição das

diferenças entre os valores reais e os valores preditos pelo modelo. Pode se observar que a maior parte dos resíduos está concentrada em torno de zero, o que indica que o modelo faz previsões próximas aos valores reais na maioria dos casos. A simetria em torno do zero sugere que o modelo não possui um viés significativo, ou seja, ele não tende a superestimar ou subestimar consistentemente as previsões. O pico central reforça que os erros de previsão são, em sua maioria, pequenos. A presença de resíduos mais distantes do centro pode indicar a existência de alguns erros maiores, sugerindo que, em certas situações, o modelo ainda pode fazer previsões menos precisas.

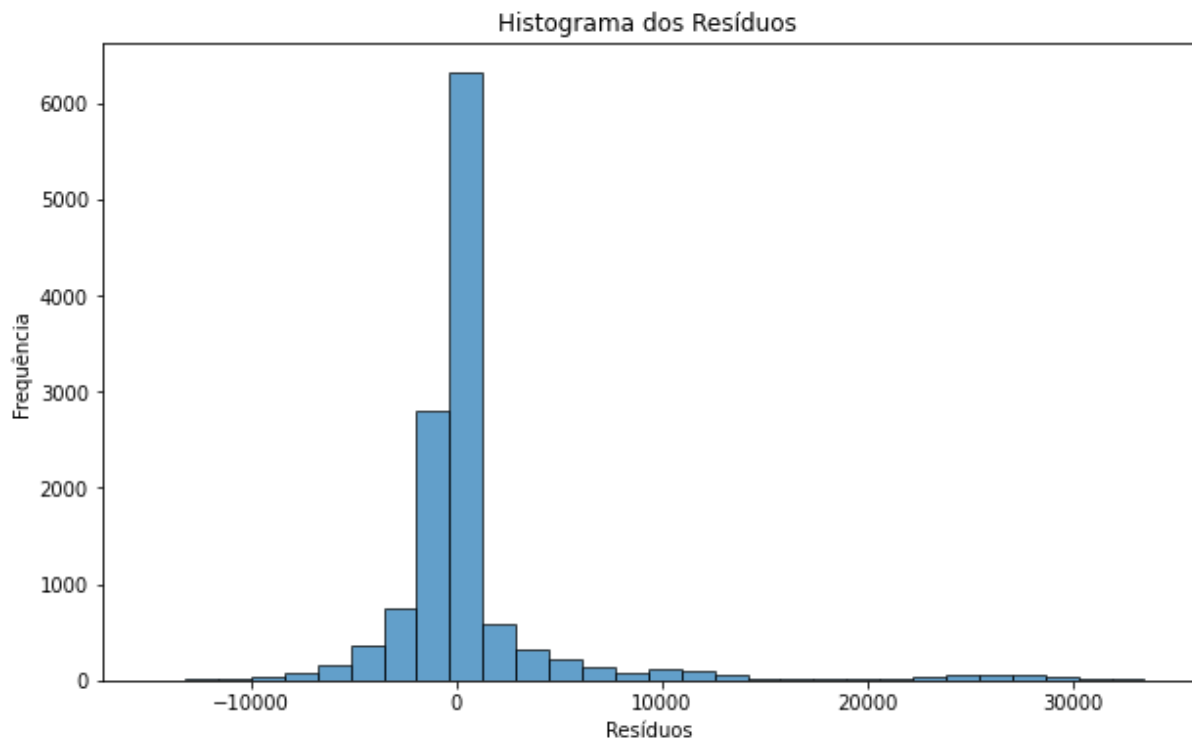
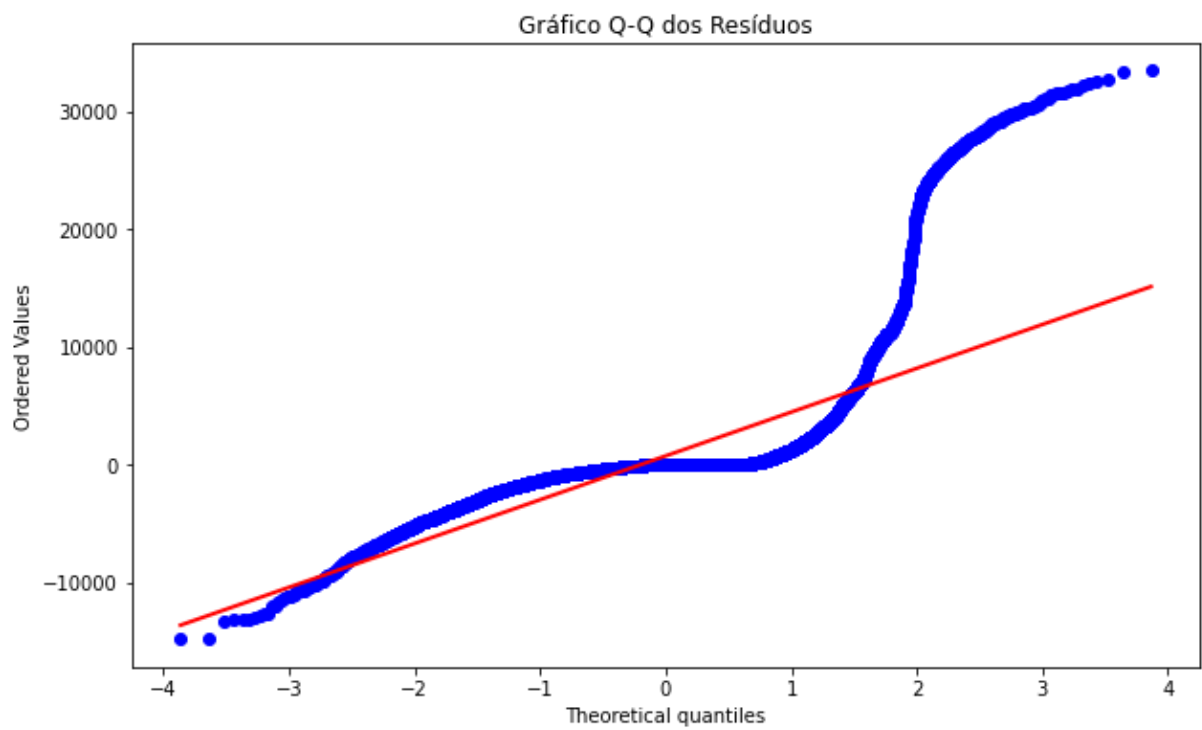


Figura 4.5: Histograma dos Resíduos

A figura 4.6 é um gráfico Q-Q (*quantile-quantile*) que auxilia na avaliação se os resíduos do modelo seguem uma distribuição normal. Neste gráfico, os quantis dos resíduos são comparados com os quantis teóricos de uma distribuição normal. A linha diagonal vermelha representa a linha onde os resíduos se alinhariam se seguissem perfeitamente a distribuição normal. Os pontos do gráfico, representando os quantis dos resíduos, se desviam da linha vermelha em ambas as extremidades (caudas) e na região central. A forma curvada dos pontos indica que os resíduos possuem uma distribuição com caudas mais pesadas ou que há assimetria. Este padrão indica que os resíduos têm variações sistemáticas que o modelo não capturou adequadamente.

Figura 4.6: Q-Q *plot*

Capítulo 5

Conclusão

5.1 Conclusão

Ao avaliar o modelo, utilizamos a métrica *MAE* e verificamos que o modelo criado com a estratégia proposta apresentou o menor erro médio absoluto em comparação com outros modelos. Isso indica uma boa capacidade de previsão do tempo até a falha da máquina, demonstrando a eficácia da abordagem adotada.

O exercício de comparação de diferentes valores limites de *SHAP* para mensurar o desempenho do modelo de forma precisa foi crucial para o trabalho. O objetivo foi identificar o ponto em que a filtragem de variáveis começava a impactar negativamente a precisão do modelo. Testando uma gama de valores, foi possível observar como cada ajuste influenciava o erro médio absoluto (*MAE*).

À medida que o valor limite para a seleção de variáveis aumenta, observa-se uma elevação correspondente no erro médio absoluto. Isso sugere que, ao restringir mais rigorosamente as variáveis consideradas de pouca relevância, o modelo perde parte de sua precisão preditiva.

Portanto, com o apoio da análise dos resultados, constatou-se que o modelo desenvolvido com o algoritmo *LightGBM* e a abordagem específica para o tratamento do *dataset* original apresentou o menor valor de erro médio absoluto (*MAE*) entre os modelos avaliados.

Este desempenho superior reforça a eficácia da estratégia adotada, indicando que as técnicas utilizadas foram bem-sucedidas em capturar padrões críticos nos dados, permitindo uma previsão mais precisa do tempo até a falha da máquina. Este resultado valida a metodologia aplicada, destacando sua importância para a área de manutenção preditiva.

5.2 Limitações do Trabalho

Embora os resultados alcançados sejam promissores, o modelo apresenta algumas limitações importantes. Uma delas é sua forte dependência de dados históricos, o que pode comprometer a precisão das previsões em contextos onde as condições do processo mudam significativamente.

Outro ponto de reflexão é sobre o intervalo de tempo capturado no *dataset* inicial, pois ele pode não ter abrangido a totalidade das variações possíveis no processo e deixado cenários de falha do equipamento relevantes para traçar padrões nos dados. Tal fato pode limitar a capacidade preditiva do modelo em diferentes cenários e criar desafios adicionais na generalização das previsões.

Além disso, há margem para melhorias tanto na seleção de variáveis quanto na otimização dos hiperparâmetros, o que poderia aumentar ainda mais a eficiência e a precisão do modelo. Essas áreas representam oportunidades para refinamento, buscando maior robustez e adaptabilidade em diferentes cenários industriais.

5.3 Trabalhos Futuros

Futuros trabalhos podem se concentrar na otimização da seleção de variáveis e dos hiperparâmetros para melhorar ainda mais o desempenho do modelo. Além disso, uma investigação mais aprofundada sobre os valores limite do *SHAP* pode revelar *insights* adicionais. A avaliação de diferentes limiares de *SHAP*, até encontrar uma divergência significativa no *MAE* ou uma interpretação percentual mais robusta, também pode ser explorada para refinar ainda mais o modelo.

Referências Bibliográficas

- [1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework, 2019.
- [2] Mohammad Ali Nemer, Joseph Azar, Jacques Demerjian, Abdallah Makhoul, and Julien Bourgeois. A review of research on industrial time series classification for machinery based on deep learning. In *2022 4th IEEE Middle East and North Africa COMMUNICATIONS Conference (MENACOMM)*, pages 89–94, 2022.
- [3] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1 edition, 2006.
- [4] Mahdi Bohlouli, Jens Dalter, Mareike Dornhöfer, Johannes Zenkert, and Madjid Fathi. Knowledge discovery from social media using big data provided sentiment analysis (somabit), 2020.
- [5] Peter J. Brockwell and Richard A. Davis. *Introduction to Time Series and Forecasting*. Springer Texts in Statistics. Springer International Publishing, 2016.
- [6] Jenna Burrell. How the machine ‘thinks’: Understanding opacity in machine learning algorithms. BIG DATA SOC, June 2016.
- [7] Leandro Nunes Castro and Daniel Gomes Ferrari. *Introdução à mineração de dados*. Saraiva, São Paulo, 2016.
- [8] Lingjiao Chen, Bilge Acun, Newsha Ardalani, Yifan Sun, Feiyang Kang, Hanrui Lyu, Yongchan Kwon, Ruoxi Jia, Carole-Jean Wu, Matei Zaharia, and James Zou. Data acquisition: A new frontier in data-centric ai, 2023.
- [9] Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. Multi-objective counterfactual explanations. In *Parallel Problem Solving from Nature – PPSN XVI*, pages 448–469. Springer International Publishing, 2020.
- [10] Kwetishe Joro Danjuma. Performance evaluation of machine learning algorithms in post-operative life expectancy in the lung cancer patients. 2015.
- [11] Virginia R. de Sa and Dana H. Ballard. Category learning through multimodality sensing. *Neural Computation*, 10(5):1097–1117, 1998.

- [12] Yongqi Dong, Kejia Chen, Yinxuan Peng, and Zhiyuan Ma. Comparative study on supervised versus semi-supervised machine learning for anomaly detection of in-vehicle can network, 2022.
- [13] Ming Jin Qingsong Wen Shirui Pan Yuqi Nie Yushan Jiang Yuxuan Liang Dongjin Song, Haomin Wen. Foundation models for time series analysis: A tutorial and survey, 2024.
- [14] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning, 2017.
- [15] Mengnan Du, Ninghao Liu, and Xia Hu. Techniques for interpretable machine learning, 2018.
- [16] Amy C. Edmondson. Strategies for learning from failure, Abril 2011.
- [17] Katti Faceli, Ana Carolina Lorena, João Gama, and André Carlos Ponce de Leon Ferreira de Carvalho. *Inteligência artificial: uma abordagem de aprendizado de máquina*. LTC, 2011.
- [18] Ko-Hui Michael Fan, Chih-Chung Chang, and Kuang-Hsiao-Yin Kongguoluo. Quadratic multiform separation: A new classification model in machine learning, 2022.
- [19] Mojtaba A. Farahani, M. R. McCormick, Ramy Harik, and Thorsten Wuest. Time-series classification in smart manufacturing systems: An experimental evaluation of state-of-the-art machine learning algorithms, 2023.
- [20] Mojtaba A. Farahani, M.R. McCormick, Robert Gianinny, Frank Hudacheck, Ramy Harik, Zhichao Liu, and Thorsten Wuest. Time-series pattern recognition in smart manufacturing systems: A literature review and ontology. *Journal of Manufacturing Systems*, 69:208–241, August 2023.
- [21] Giuseppe Fenza, Mariacristina Gallo, Vincenzo Loia, Francesco Orciuoli, and Enrique Herrera-Viedma. Data set quality in machine learning: Consistency measure based on group decision making. *Appl. Soft Comput.*, 106(C), jul 2021.
- [22] Luciana Ferrer. Analysis and comparison of classification metrics, 2022.
- [23] Esther Galbrun. The minimum description length principle for pattern mining: a survey. *Data Mining and Knowledge Discovery*, 36(5):1679–1727, jul 2022.
- [24] María García and José Aznarte. Shapley additive explanations for no2 forecasting. *Ecological Informatics*, 56:101039, 03 2020.
- [25] Joel Grus. *Data Science from Scratch: First Principles with Python*. O’Reilly Media, 2019.

- [26] Venkat N. Gudivada, Amy W. Apon, and Junhua Ding. Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations. 2017.
- [27] Robert I. Hamilton and Panagiotis N. Papadopoulos. Using shap values and machine learning to understand trends in the transient stability limit, 2023.
- [28] Jeff Heaton. An empirical analysis of feature engineering for predictive modeling. In *SoutheastCon 2016*. IEEE, mar 2016.
- [29] Jane Huang. Causal inference, Novembro 2020.
- [30] Robin John Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, Australia, 2nd edition, 2018.
- [31] Johannes Jakubik, Michael Vössing, Niklas Kühl, Jannis Walk, and Gerhard Satzger. Data-centric artificial intelligence, 2024.
- [32] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2013.
- [33] Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, apr 2021.
- [34] Femke Jansen, Mike Holenderski, Tanir Ozcelebi, Paulien Dam, and Bas Tijmsma. Predicting machine failures from industrial time series data. In *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 1091–1096, 2018.
- [35] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey, 1996.
- [36] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc.
- [37] Yesuel Kim and Youngchul Kim. Explainable heat-related mortality with random forest and shapley additive explanations (shap) models. *Sustainable Cities and Society*, 79:103677, 2022.
- [38] Katiana Kontolati, Dimitrios Loukrezis, Dimitrios G. Giovanis, Lohit Vandanapu, and Michael D. Shields. A survey of unsupervised learning methods for high-dimensional uncertainty quantification in black-box-type problems. *Journal of Computational Physics*, 464:111313, September 2022.

- [39] Michał Kuźba and Przemysław Biecek. What would you ask the machine learning model? identification of user needs for model explanations based on human-model conversations. In *ECML PKDD 2020 Workshops*, pages 447–459. Springer International Publishing, 2020.
- [40] Denis Leite, Aldonso Martins, Diego Rativa, Joao F. L. De Oliveira, and Alexandre M. A. Maciel. An automated machine learning approach for real-time fault detection and diagnosis. *Sensors*, 22(16), 2022.
- [41] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions, 2017.
- [42] Alexey Margun, Radda Iureva, and Daniil Antonov. Machine learning approaches for failures detection in inertial navigation systems. In *2024 International Russian Smart Industry Conference (SmartIndustryCon)*, pages 532–537, 2024.
- [43] Christoph Molnar. *Interpretable Machine Learning*. 2 edition, 2022.
- [44] Andreas Mueller and Sarah Guido. *Introduction to Machine Learning with Python*. O’Reilly Media, 2016.
- [45] W. James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.
- [46] Conor O’Sullivan. What is explainable ai (xai)?, Setembro 2022.
- [47] Yunpeng Qing, Shunyu Liu, Jie Song, and Mingli Song. A survey on explainable reinforcement learning: Concepts, algorithms, challenges, 2022.
- [48] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning, 2016.
- [49] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. ”why should i trust you?”: Explaining the predictions of any classifier, 2016.
- [50] Ribana Roscher, Marc Rußwurm, Caroline Gevaert, Michael Kampffmeyer, Jeferson A. dos Santos, Maria Vakalopoulou, Ronny Hänsch, Stine Hansen, Keiller Nogueira, Jonathan Prexl, and Devis Tuia. Data-centric machine learning for geospatial remote sensing data, 2023.
- [51] Alvin E. Roth. *The Shapley Value: Essays in Honor of Lloyd S. Shapley*. Cambridge University Press, 1988.

- [52] Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Gregoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich, and Klaus-Robert Muller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, May 2021.
- [53] Ahmed M. Salih, Zahra Raisi-Estabragh, Ilaria Boscolo Galazzo, Petia Radeva, Steffen E. Petersen, Karim Lekadir, and Gloria Menegaz. A perspective on explainable artificial intelligence methods: Shap and lime. *Advanced Intelligent Systems*, June 2024.
- [54] Shashank Shekhar, Adesh Bansode, and Asif Salim. A comparative study of hyperparameter optimization tools, 2022.
- [55] Dylan Slack, Sorelle A. Friedler, Carlos Scheidegger, and Chitradeep Dutta Roy. Assessing the local interpretability of machine learning models, 2019.
- [56] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.
- [57] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.
- [58] Sarthak Manas Tripathy, Ashish Chouhan, Marcel Dix, Arzam Kotriwala, Benjamin Klöpper, and Ajinkya Prabhune. Explaining anomalies in industrial multivariate time-series data with the help of explainable ai. In *2022 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 226–233, 2022.
- [59] Ruey Tsay. Analysis of financial time series: Third edition. *Analysis of Financial Time Series: Third Edition*, 08 2010.
- [60] Granville Tunncliffe Wilson. Time series analysis: Forecasting and control, 5th edition, by george e. p. box, gwilym m. jenkins, gregory c. reinsel and greta m. ljun, 2015. published by john wiley and sons inc., hoboken, new jersey, pp. 712. isbn: 978-1-118-67502-1. *Journal of Time Series Analysis*, 37:n/a–n/a, 03 2016.
- [61] Irfan Ullah, Kai Liu, Toshiyuki Yamamoto, Muhammad Zahid, and Arshad Jamal. Prediction of electric vehicle charging duration time using ensemble machine learning algorithm and shapley additive explanations. *International Journal of Energy Research*, 46(11):15211–15230, 2022.
- [62] Nicolò Oreste Pincirolì Vago, Francesca Forbicini, and Piero Fraternali. Predicting machine failures from multivariate time series: an industrial case study, 2024.

- [63] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. 2017.
- [64] Huajun Wang, Yuanhai Shao, Shenglong Zhou, Ce Zhang, and Naihua Xiu. Support vector machine classifier via $l_{0/1}$ soft-margin loss, 2019.
- [65] Qingsong Wen, Linxiao Yang, Tian Zhou, and Liang Sun. Robust time series analysis and applications: An industrial perspective. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, page 4836–4837, New York, NY, USA, 2022. Association for Computing Machinery.
- [66] Hadjer Ykhlef and Sarah Rouizi. Time series fault detection for power line condition monitoring: A comparative study. In *2022 IEEE/ACS 19th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–8, 2022.
- [67] Wenjin Yu, Tharam Dillon, Fahed Mostafa, Wenny Rahayu, and Yuehua Liu. A global manufacturing big data ecosystem for fault detection in predictive maintenance. *IEEE Transactions on Industrial Informatics*, 16(1):183–192, 2020.
- [68] Wenjin Yu, Tharam Dillon, Fahed Mostafa, Wenny Rahayu, and Yuehua Liu. A global manufacturing big data ecosystem for fault detection in predictive maintenance. *IEEE Transactions on Industrial Informatics*, 16(1):183–192, 2020.
- [69] Larijn Zaadnoordijk, Tarek R. Besold, and Rhodri Cusack. The next big thing(s) in unsupervised machine learning: Five lessons from infant learning, 2020.
- [70] Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, and Xia Hu. Data-centric ai: Perspectives and challenges, 2023.
- [71] Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Hu. Data-centric artificial intelligence: A survey, 2023.
- [72] Jiuqi Elise Zhang, Di Wu, and Benoit Boulet. Time series anomaly detection via reinforcement learning-based model selection, 2022.
- [73] A. Zheng and A. Casari. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. O'Reilly, 2018.
- [74] Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester. A comprehensive survey on deep clustering: Taxonomy, challenges, and future directions, 2022.
- [75] Tiago Zonta, Cristiano André da Costa, Rodrigo da Rosa Righi, Miromar José de Lima, Eduardo Silveira da Trindade, and Guann Pyng Li. Predictive maintenance in the industry 4.0: A systematic literature review. *Computers Industrial Engineering*, 150:106889, 2020.