



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
DEPARTAMENTO DE AUTOMAÇÃO E SISTEMAS  
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Ana Paula da Rocha

**Manutenção preditiva de máquinas rotativas para detecção de falhas mecânicas  
com técnicas de aprendizado de máquina utilizando dados reais de sensores de  
vibração**

Florianópolis  
2022

Ana Paula da Rocha

**Manutenção preditiva de máquinas rotativas para detecção de falhas mecânicas com técnicas de aprendizado de máquina utilizando dados reais de sensores de vibração**

Relatório final da disciplina DAS5511 (Projeto de Fim de Curso) como Trabalho de Conclusão do Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Catarina em Florianópolis.

Orientador: Prof. Danilo Silva, Ph.D.

Florianópolis

2022

### Ficha de identificação da obra

A ficha de identificação é elaborada pelo próprio autor.

Orientações em:

<http://portalbu.ufsc.br/ficha>

Ana Paula da Rocha

**Manutenção preditiva de máquinas rotativas para detecção de falhas mecânicas com técnicas de aprendizado de máquina utilizando dados reais de sensores de vibração**

Esta monografia foi julgada no contexto da disciplina DAS5511 (Projeto de Fim de Curso) e aprovada em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação

Florianópolis, 16 de Dezembro de 2022.

Prof. Hector Bessa Silveira, Dr.  
Coordenador do Curso

**Banca Examinadora:**

Prof. Danilo Silva, Ph.D.  
Orientador  
UFSC/CTC/EEL

Prof. Danilo Silva, Ph.D.  
Supervisor  
UFSC/CTC/EEL

Prof. Laio Seman, Dr.  
Avaliador  
UFSC

Prof. Eduardo Camponogara, Dr.  
Presidente da Banca  
UFSC/CTC/DAS

Aos meus pais, Lucia e Vilmar.

## **AGRADECIMENTOS**

Ao término do trabalho de fim de curso, é impossível não agradecer àqueles que, de alguma forma, estiveram presentes durante o processo de escrita. Então, vamos lá.

Agradeço, enormemente, aos meus pais por todo o apoio e esforços realizados para eu me formar. Esta conquista não é minha, é nossa. Aos meus parceiros Marcelo e Mimi, por todo o amor e leveza que vocês trazem ao meu mundo.

Ao meu Professor Danilo Silva, por me orientar o ano todo, me ensinando e corrigindo com didática e paciência. Com certeza é uma das pessoas mais incríveis e inteligentes que já conheci.

Ao Rodrigo, por me acompanhar nas atividades do dia a dia e me ajudar a resolver todos os impedimentos e sanar minhas dúvidas no decorrer deste trabalho.

Aos meus melhores amigos Fernanda e Matheus por estarem sempre comigo, deixando tudo mais divertido e animado.

Ao meu colega Vinicius por compartilhar as tardes no laboratório comigo. À Ingrid por me escutar e tranquilizar sempre que necessário.

À Dynamox pela disponibilização dos dados e pelo apoio concedido pela equipe. Em especial ao Lobato, Spillere e Braga por sanar todas as minhas dúvidas e compartilharem o seu conhecimento.

À Universidade Federal de Santa Catarina, por possibilitar a execução deste trabalho e me apresentar diversas oportunidades incríveis.

*“Eu não falhei. Apenas descobri 10 mil maneiras que não funcionam.”*  
*(Thomas Edison)*

## RESUMO

Técnicas de manutenção preditiva são cada vez mais utilizadas na indústria, visto que proporcionam melhorias na eficiência da produção e ganhos financeiros ao aumentarem a vida útil das máquinas e minimizarem o número de manutenções. Considerando este cenário, este trabalho busca implementar um modelo de aprendizado de máquina de *random forest* para detecção de falhas mecânicas a partir de dados reais de sensores de vibração de máquinas rotativas. Para isso, foi realizada uma pesquisa exploratória e quantitativa com os objetivos de: i) analisar o desempenho do modelo ao utilizar dados de telemetria e dados do monitoramento de forma de onda separados e em conjunto; ii) e analisar os efeitos produzidos no desempenho dos modelos ao se alterar a tarefa de detecção, da presença ou não de uma falha mecânica qualquer para a detecção de falhas mecânicas específicas. A metodologia CRISP-DM foi empregada como forma de apoio para o processo de desenvolvimento dos experimentos de aprendizado de máquina. Com os experimentos, verificou-se que alterar a tarefa para detecção de falhas específicas não resulta um desempenho melhor do que considerar apenas a classificação binária de que há ou não falha mecânica. Além disso, o modelo apresenta o melhor desempenho ao utilizar apenas os dados do monitoramento de forma de onda, indicando que embora seja possível identificar padrões com os dados de telemetria, tais dados não contribuem com o desempenho do modelo frente aos dados de forma de onda. Finalmente, no caso da detecção de falhas mecânicas a partir de dados do monitoramento de forma de onda, observou-se que as *features* do sinal de envelope foram as consideradas mais relevantes para o modelo.

**Palavras-chave:** Manutenção Preditiva. Aprendizado de Máquina. *Random Forest*. Dados de vibração.

## ABSTRACT

Predictive maintenance techniques are increasingly used in industry, as they provide improvements in production efficiency and financial gains by increasing the useful life of machines and minimizing the number of maintenances. Considering this scenario, this work seeks to implement a random forest machine learning model for detecting mechanical failures from real data from vibration sensors of rotating machines. For this, an exploratory and quantitative research was carried out with the objectives of: i) analyzing the performance of the model when using telemetry data and waveform monitoring data separately and together; ii) and analyze the effects produced on the performance of the models when changing the detection task, from the presence or absence of any mechanical failure to the detection of specific mechanical failures. The CRISP-DM methodology was used to support the process of developing machine learning experiments. With the experiments, it was verified that changing the task to detect specific failures does not result in a better performance than considering only the binary classification that there is or is not a mechanical failure. In addition, the model presents the best performance when using only the waveform monitoring data, indicating that although it is possible to identify patterns with the telemetry data, such data do not contribute to the performance of the model against the waveform data. . Finally, in the case of detecting mechanical failures from waveform monitoring data, it was observed that the features of the envelope signal were considered the most relevant for the model.

**Keywords:** Predictive Maintenance. Machine Learning. Random forest. Vibration data.

## LISTA DE FIGURAS

Figura 1 – Dispositivos de sensores sem fio de vibração e temperatura da solução DynaPredict. . . . .	17
Figura 2 – Tipos de desalinhamento. . . . .	19
Figura 3 – Tipos de desbalanceamento. . . . .	20
Figura 4 – Exemplo genérico de árvore de decisão. . . . .	22
Figura 5 – Exemplo do processo de <i>ensemble learning</i> com o método <i>hard classifier</i> . . . . .	23
Figura 6 – Exemplo do processo de criação de subconjuntos com <i>bagging</i> . . . . .	24
Figura 7 – Exemplo de variância e viés. . . . .	25
Figura 8 – Origem dos dados para geração do conjunto de treinamento e conjunto de teste. . . . .	27
Figura 9 – <i>Pipeline</i> em alto nível do processamento dos dados para criação do conjunto de treinamento e teste. . . . .	29
Figura 10 – <i>Pipeline</i> simplificada do projeto. . . . .	30
Figura 11 – Relatórios de falhas duplicados. . . . .	33
Figura 12 – Contagem de relatórios por situação ( <i>status</i> ) da máquina para cada falha. . . . .	34
Figura 13 – Sinais de vibração dos dados de telemetria juntamente com os alertas e relatórios de falha cadastrados. . . . .	35
Figura 14 – Características presentes e analisadas da aceleração e deslocamento obtidos nos dados do monitoramento de forma de onda. . . . .	36
Figura 15 – Características presentes e analisadas da velocidade e envelope obtidos nos dados do monitoramento de forma de onda. . . . .	36
Figura 16 – Faixas que adotam altas frequências e estão presentes nos sinais dos dados do monitoramento de forma de onda. . . . .	37
Figura 17 – Faixas que adotam baixas frequências e estão presentes nos sinais dos dados do monitoramento de forma de onda. . . . .	38
Figura 18 – Sinais de vibração dos dados do monitoramento de forma de onda juntamente com os relatórios de falha cadastrados. . . . .	38
Figura 19 – Fluxograma do processo de remoção dos relatórios de falhas com data de criação próximas. . . . .	39
Figura 20 – Exemplo do processo de remoção dos relatórios de falhas com datas de criação próximas. . . . .	40
Figura 21 – <i>Dashboard</i> para seleção das amostras do conjunto de teste. . . . .	41
Figura 22 – Fragmento dos relatórios de falha selecionados através do <i>dashboard</i> . . . . .	41
Figura 23 – Quantidade de amostras que possuem cada <i>feature</i> dos dados do monitoramento de forma de onda fornecido pela Dynamox. . . . .	42

Figura 24 – <i>Features</i> de forma de onda selecionadas para o modelo de <i>random forest</i> . . . . .	43
Figura 25 – Fluxograma do processamento dos dados para criação do conjunto de treinamento e conjunto de teste. . . . .	44
Figura 26 – Exemplo de detecção de máquina parada para um <i>spot</i> . . . . .	45
Figura 27 – Fluxograma da seleção do eixo de amplitude dominante. . . . .	46
Figura 28 – Exemplo de dados de telemetria faltantes no processamento do conjunto de treinamento e conjunto de teste. . . . .	47
Figura 29 – Exemplo de amostras do conjunto de teste. . . . .	47
Figura 30 – Quão maior em porcentagem foi a quantidade de dados não nulos para as 18 amostras do conjunto de teste que utilizaram o método de seleção do eixo de amplitude dominante pelos dados de forma de onda não nulos. Ressalta-se que <i>axis1</i> e <i>axis2</i> são os eixos não dominantes. . . . .	49
Figura 31 – Mudança de eixo dominante para um mesmo <i>spot</i> . . . . .	50
Figura 32 – Desempenho do modelo de detecção de falhas mecânicas com os sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda ao variar o parâmetro <i>n_repeat</i> do método de criação do objeto de validação cruzada. . . . .	51
Figura 33 – Diminuição média da impureza das <i>features</i> consideradas mais relevantes pelo modelo de <i>random forest</i> para detecção de falhas mecânicas. . . . .	57
Figura 34 – Diminuição média da impureza das <i>features</i> consideradas mais relevantes pelo modelo de <i>random forest</i> para detecção da falha de folga rotativa. . . . .	57
Figura 35 – Diminuição média da impureza das <i>features</i> consideradas mais relevantes pelo modelo de <i>random forest</i> para detecção da falha de desalinhamento. . . . .	58
Figura 36 – Diminuição média da impureza das <i>features</i> consideradas mais relevantes pelo modelo de <i>random forest</i> para detecção da falha de desbalanceamento. . . . .	58

## LISTA DE QUADROS

Quadro 1 – Valores da acurácia balanceada ao executar a pipeline de otimização dos hiperparâmetros e treinamento do modelo em 5 iterações.	31
Quadro 2 – Períodos de dados fornecidos pela Dynamox. . . . .	32
Quadro 3 – Quantidade de amostras para cada amostra no conjunto de teste e no conjunto de treinamento. . . . .	48
Quadro 4 – Hiperparâmetros utilizados na busca aleatória com estratégia de validação cruzada. . . . .	52
Quadro 5 – Definição das <i>targets</i> de cada modelo de classificação utilizado. . .	53
Quadro 6 – Métrica acurácia balanceada para os modelos de <i>random forest</i> no conjunto de validação. . . . .	54
Quadro 7 – Métrica acurácia balanceada para os modelos de <i>random forest</i> no conjunto de teste. . . . .	54
Quadro 8 – Métrica de sensibilidade para os modelos de <i>random forest</i> no conjunto de teste. . . . .	55

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>16</b>
2.1	MANUTENÇÃO PREDITIVA	16
2.2	SENSORIAMENTO	16
2.3	SINAIS DE VIBRAÇÃO	18
2.4	FALHAS MECÂNICAS	18
2.5	APRENDIZADO DE MÁQUINA	21
<b>2.5.1</b>	<b>Árvore de Decisão</b>	<b>21</b>
<b>2.5.2</b>	<b><i>Ensemble learning</i></b>	<b>23</b>
<b>2.5.3</b>	<b><i>Random Forest</i></b>	<b>23</b>
<b>3</b>	<b>METODOLOGIA</b>	<b>26</b>
<b>4</b>	<b>ANÁLISE EXPLORATÓRIA</b>	<b>32</b>
4.1	CONJUNTO DE DADOS	32
4.2	RELATÓRIO DE FALHAS	32
4.3	SINAIS DE VIBRAÇÃO	35
<b>5</b>	<b>PROCESSAMENTO DOS DADOS</b>	<b>39</b>
5.1	TRATAMENTO DOS RELATÓRIOS DE FALHA	39
5.2	SELEÇÃO DAS AMOSTRAS DO CONJUNTO DE TESTE	40
5.3	SELEÇÃO DAS <i>FEATURES</i> DE FORMA DE ONDA	41
5.4	<i>PIPELINE</i> DE CRIAÇÃO DO CONJUNTO DE TREINAMENTO E DO CONJUNTO DE TESTE	43
<b>6</b>	<b>MODELAGEM</b>	<b>48</b>
6.1	AMOSTRAS DO CONJUNTO DE TREINAMENTO E DO CONJUNTO DE TESTE	48
6.2	BUSCA DOS HIPERPARÂMETROS	51
6.3	MODELOS DE <i>RANDOM FOREST</i>	52
<b>7</b>	<b>RESULTADOS</b>	<b>54</b>
7.1	DESEMPENHO DOS MODELOS	54
7.2	IMPORTÂNCIA DAS <i>FEATURES</i>	56
<b>8</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>59</b>
	<b>REFERÊNCIAS</b>	<b>61</b>

## 1 INTRODUÇÃO

O tema deste trabalho é a manutenção preditiva de máquinas rotativas com técnicas de aprendizado de máquina utilizando dados reais de sensores de vibração. A manutenção preditiva, segundo (SP *et al.*, 2018), é uma técnica de monitoramento de condição usada para identificar mudanças significativas no maquinário, na qual essas alterações são indicativos de presença de falhas. Com o reconhecimento precoce de problemas, é possível pôr em prática ações preventivas que contribuam para a redução de custos de manutenção e melhora na eficiência das operações.

Para isso, existem diversas técnicas de monitoramento, sendo o monitoramento de vibração, abordado neste trabalho, uma delas. De acordo com (SP *et al.*, 2018), essa técnica é uma das mais efetivas, visto que pode detectar falhas mecânicas em qualquer máquina rotativa, e inclui o processamento dos dados de telemetria e/ou do monitoramento de forma de onda. Os dados do monitoramento de forma de onda possuem sinais com diferentes faixas de frequências, enquanto os dados de telemetria envolvem sinais físicos ou em relação ao tempo.

Com as informações de vibração pode-se aplicar a manutenção preditiva em máquinas rotativas, sendo essa aplicação relevante, visto que esse tipo de máquina é amplamente utilizado na indústria moderna, como aponta (QIAN; LI; JIANG, 2019). As máquinas rotativas, de acordo com (SÁ SÓ MARTINS *et al.*, 2021), geralmente operam sob uma variedade de condições, como mudanças frequentes de carga e altas velocidades, então estão sujeitas a falhas mecânicas e degradação de desempenho. Ainda segundo (SÁ SÓ MARTINS *et al.*, 2021), dependendo da gravidade, as falhas podem causar interrupções inesperadas na produção, gerar impactos ambientais negativos e levar a perdas econômicas. Esses exemplos corroboram para a relevância do monitoramento dos sinais de vibração para a predição de falhas.

Agregar a manutenção preditiva com os métodos de aprendizado tornou-se a solução mais eficaz para lidar com a manufatura inteligente e com a grande quantidade de dados industriais, como argumentado em (ZHANG; YANG; WANG, 2019). Com esse contexto, o trabalho consiste em realizar um estudo sobre a aplicação de um modelo clássico de aprendizado de máquina para detectar se esta apresenta, ou não, (classificação binária) as seguintes falhas mecânicas: desalinhamento, desbalanceamento e/ou folga rotativa.

Serão utilizados dados reais de máquinas rotativas das empresas parceiras da Dynamox, que contêm dados de telemetria e dados do monitoramento de forma de onda, além de relatórios associando esses sinais aos diagnósticos de falhas ou normalidade, sendo esses preenchidos por analistas que acompanham o funcionamento das máquinas em estudo. Com essas informações, será desenvolvida uma *pipeline* para criar um conjunto de dados de treino e teste, que serão aplicados no modelo de

aprendizado de máquina de *random forest* para classificar se há, ou não, uma falha mecânica (específica ou não) presente na máquina.

A Dynamox é “uma empresa de alta tecnologia que desenvolve sistemas de monitoramento e aquisição de dados de vibração e temperatura. É especialista em análise de vibrações e monitoramento da condição de ativos industriais. Possuem serviços personalizados e integrados, importantes recursos para uma gestão eficiente de ativos para a equipe preditiva, além de especialistas em vibração do departamento de P&D, que focam em oferecer serviços alinhados às necessidades da indústria” (DYNAMOX, 2022c).

A solução DynaPredict conta com os sensores DynaLoggers que são ideais para a manutenção preditiva, no monitoramento contínuo de máquinas e componentes rotativos (DYNAMOX, 2022a). Essa solução realiza a coleta de dados automatizada e possui aplicativo para configurar e coletar dados e plataforma para análise e tomada de decisão (DYNAMOX, 2022a).

Dito isso, é importante mencionar que essa pesquisa será realizada com o propósito de desenvolver um modelo preditivo, sendo investigado suas variações e possibilidades. Serão respondidos dois problemas de pesquisa com esse trabalho:

1. Ao se alterar a tarefa de detecção, da presença ou não de uma falha mecânica qualquer para a detecção de falhas mecânicas específicas, é obtido um desempenho melhor no modelo de *random forest*?
2. Com base nos dados de telemetria e nos dados do monitoramento de forma de onda, com qual conjunto de sinais é obtido o melhor desempenho nos modelos?

A Dynamox possui dados de sensores de vibração de múltiplos equipamentos de diversas empresas parceiras que fornecem um conjunto de dados já inspecionados por um sistema de monitoração. Contudo, ainda faz-se necessária a análise dos dados por um analista de vibração para uma conclusão correta do problema e identificação da falha. Atualmente, os analistas possuem muitos equipamentos e, conseqüentemente, muita informação para verificar. Isso dificulta a execução da manutenção antes do problema se tornar mais crítico.

Com um modelo de aprendizado de máquina para classificar se a falha está presente ou não, seria agregado valor no sistema de monitoramento já existente na Dynamox, o que contribuiria no trabalho do analista para direcionar o foco em máquinas específicas com uma possível falha já detectada. Foi definido aplicar o modelo em falhas mecânicas, visto que são falhas que aparecem mais comumente em máquinas. O objetivo é validar se a aplicação dos dados reais disponíveis em um modelo de aprendizado de máquina clássico agrega valor à empresa.

Resta pontuar os procedimentos metodológicos realizados para o desenvolvimento deste trabalho. Em primeiro lugar, foi desenvolvida a revisão de literatura, pois

é por meio dela que o tema será contextualizado, situando-o dentro da área de Engenharia de Controle e Automação. Nesse sentido, definiram-se os autores pertinentes a fim de fundamentar a pesquisa, apresentando os conhecimentos já produzidos previamente, isto é, os procedimentos, os resultados, as discussões e as conclusões relevantes para esse estudo.

Junto a tal pesquisa foi utilizada a metodologia CRISP-DM (Cross Industry Standard Process for Data Mining), que consiste em um conjunto de etapas e boas práticas para se realizar o processo de mineração de dados, conforme (SCHRÖER; KRUSE; GÓMEZ, 2021). Tal metodologia consiste em seis passos: i) entendimento do negócio; ii) entendimento dos dados; iii) preparação dos dados; iv) modelagem dos dados; v) avaliação do modelo; vi) e colocar em produção. É essencial destacar que o método foi adaptado, haja vista que não foi realizado o sexto passo, pois o objetivo inicial da empresa é estudar a possibilidade para, posteriormente, decidir ou não implementar. De qualquer modo, as cinco etapas foram suficientes para segmentar o objetivo principal em pequenos objetivos e validar cada etapa do projeto.

A autora desenvolveu as etapas de entendimento, modelagem e avaliação dos dados, bem como a aplicação e validação dos modelos. Para isso, possuía o apoio do mestrando para a solução de impedimentos no decorrer do projeto, além de validação semanal das atividades realizadas e dos resultados obtidos pelo professor e pela equipe da Dynamox. As etapas relacionadas à criação e manutenção do banco de dados foram realizadas pelo mestrando e o planejamento das atividades do projeto era definido pelo professor e pelo mestrando.

O trabalho foi organizado em 8 capítulos que esclarecem o desenvolvimento do projeto realizado. No capítulo 2 são documentadas as teorias e modelos usados na concepção da solução do problema. No capítulo 3, é detalhada a metodologia adotada para a execução do trabalho.

No capítulo 4, são detalhadas as análises exploratórias dos dados fornecidos pela Dynamox. No capítulo 5, são explicadas as implementações feitas para processar os dados. No capítulo 6, são apresentados o conjunto de treinamento e conjunto de teste final, além dos modelos usados nos experimentos. No capítulo 7, são discutidos e comparados os resultados para cada modelo. Por último, no capítulo 8, são apresentadas as considerações finais do trabalho e sugestões para futuras implementações.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos de engenharia e de programação utilizados para o desenvolvimento do projeto. Inicialmente são apresentados os conceitos de manutenção preditiva e os elementos que compõem essa técnica. Posteriormente são abordados os tópicos relacionados a aprendizado de máquina, como os modelos e estratégias utilizadas.

### 2.1 MANUTENÇÃO PREDITIVA

A manutenção preditiva consiste, segundo (GIRDHAR; SCHEFFER, 2004a), em programar a execução de manutenções apenas quando uma falha funcional é detectada, sendo que a detecção ocorre graças ao monitoramento periódico das máquinas, o que permite constatar quando há uma predisposição de problema. É necessário atentar-se a um conjunto de itens para que a predição apresente uma boa confiabilidade, por exemplo, os dados das máquinas estarem corretos e disponíveis - para isso é necessário o uso de sensores próprios para cada informação coletada. Também é necessário garantir que o desempenho do método seja avaliado corretamente, sendo, dessa forma, assegurado que o rastreamento de falhas incorretas seja mínimo.

Essa técnica de manutenção, de acordo com (GIRDHAR; SCHEFFER, 2004a), é vista como ferramenta essencial na indústria, pois resulta em uma diminuição dos custos de manutenção, tempo de inatividade operacional e retenção de estoque. Outras vantagens, como aponta (GIRDHAR; SCHEFFER, 2004a), são: i) Minimizar o número de manutenções, realizando-as apenas conforme o necessário; ii) Diminuição dos danos, visto que as máquinas são paradas antes da ocorrência das falhas; iii) conseqüentemente, há o aumento da vida útil das máquinas, visto que são mais conservadas; iv) e os recursos de reparo podem ser planejados, sendo possível alocar peças e trabalhadores para a função de manutenção.

Um dos monitoramentos possíveis da manutenção preditiva é o monitoramento de vibrações e sua posterior análise. De acordo com (GIRDHAR; SCHEFFER, 2004b), “a análise de vibração é usada para determinar a condição operacional e mecânica dos equipamentos”. Falhas como frouxidão, engrenagens gastas ou quebradas e rolamentos deteriorados ou defeituosos podem ser identificados pelo monitoramento da vibração (GIRDHAR; SCHEFFER, 2004b). Além disso, as falhas de desalinhamento e desbalanceamento também podem ser identificadas antes que a máquina se deteriore.

### 2.2 SENSORIAMENTO

Os DynaLoggers são sensores sem fio de monitoramento de vibração e temperatura que possuem como vantagens: "i) redução de ações de manutenção não

planejadas; ii) aumento da disponibilidade e confiabilidade de ativos monitorados; iii) aumento seguro ao maquinário, alinhado à NR12; iv) potencial redução de custos com manutenção corretiva e preventiva; v) dados centralizados para análise e prognóstico remoto; vi) e troca eficiente de peças de reposição, prolongando a vida útil" (DYNAMOX, 2022a). A solução DynaPredict conta com três dispositivos de sensores sem fio de vibração e temperatura, sendo eles:

- DynaLogger TcAg, dispositivo laranja apresentado na Figura 1, é um data *logger Bluetooth* que monitora a condição de máquina. É simples, robusto e performante. Realiza o monitoramento contínuo minuto a minuto e a sua fixação é feita usando cola (DYNAMOX, 2022a);
- DynaLogger HF+ (*High Frequency*), dispositivo azul apresentado na Figura 1, é um data *logger Bluetooth* que monitora a condição de máquina e possui capacidade de análise espectral e telemetria até 13.3 kHz (DYNAMOX, 2022a);
- DynaLogger TcAs, dispositivo verde apresentado na Figura 1, é um data *logger Bluetooth* que monitora a condição de máquina e possui capacidade de análise espectral e telemetria até 2.5 kHz (DYNAMOX, 2022a).

Figura 1 – Dispositivos de sensores sem fio de vibração e temperatura da solução DynaPredict.



Fonte: (DYNAMOX, 2022b).

## 2.3 SINAIS DE VIBRAÇÃO

De acordo com (GIRDHAR; SCHEFFER, 2004b), “as características de deslocamento, velocidade e aceleração da vibração são medidas que determinam a severidade da vibração e essas são muitas vezes referidas como a “amplitude” da vibração”. A partir desses sinais é possível obter outros parâmetros como:

- Curtose: De acordo com (FILHO, s.d.), “para mancais de rolamento, o valor da curtose varia conforme o progresso da falha, sendo que cresce rapidamente quando a falha se inicia”;
- Valor RMS (Raiz Quadrada Média): Esse parâmetro depende da energia global do sinal e aumenta com o desenvolvimento da falha (FILHO, s.d.);
- Valor de pico a pico: Maior amplitude medida pelo sensor. O valor de pico aumenta antes do valor RMS quando há o surgimento de uma possível falha, pois é um valor pontual máximo medido (FILHO, s.d.);
- Fator de crista: É o quociente entre o valor de pico e o valor RMS. À medida que a falha se desenvolve, o fator de crista tende a apresentar valores bruscos, sendo um indicativo do aumento de um defeito localizado (FILHO, s.d.).

Em termos de operação da máquina, “a amplitude da vibração é o primeiro indicador que aponta quão boa ou ruim está a condição da máquina” (GIRDHAR; SCHEFFER, 2004b). Assim, sabendo que há uma deterioração na amplitude de vibração da máquina, para ser possível identificar qual a falha, é necessário considerar a frequência de análise. Movimentos abaixo de 10 Hz produzem vibrações consideráveis em termo de deslocamento; entre 10 Hz e 1000 Hz, as vibrações são bem caracterizadas pela velocidade e, acima de 1000 Hz, a aceleração é o melhor indicador para caracterizar as vibrações (GIRDHAR; SCHEFFER, 2004b).

## 2.4 FALHAS MECÂNICAS

Os elementos que compõem as máquinas apresentam comportamentos relacionados à determinadas frequências, sendo que cada componente é atribuído a uma parcela das frequências que formam os sinais de vibração da máquina. A partir disso, é possível detectar qual falha está se desenvolvendo de acordo com os padrões estudados na literatura que caracterizam problemas nos maquinários.

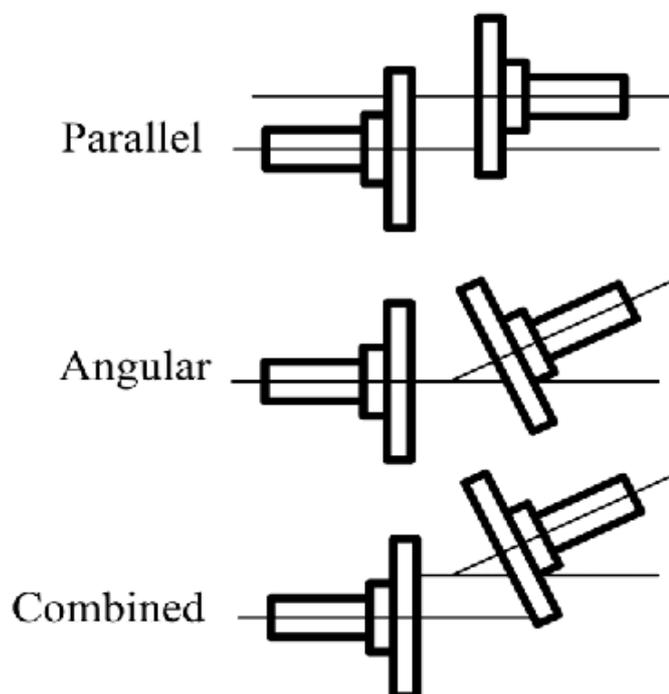
As falhas abordadas neste trabalho são: desalinhamento, desbalanceamento, folga rotativa, folga estrutural, folga do mancal, perda de rigidez e engrenagens desalinhadas. Todas são falhas mecânicas que podem ser detectadas pela análise do sinal de vibração das máquinas, sendo descritas as três falhas previstas separadamente neste trabalho:

- Desalinhamento

O desalinhamento do eixo pode ocorrer em direções paralelas e/ou angulares. O desalinhamento angular ocorre de acordo com (GIRDHAR; SCHEFFER, 2004c) quando “a linha central do eixo dos dois eixos se encontra em ângulo entre si”, enquanto o desalinhamento paralelo ocorre quando “a linha central do eixo das duas máquinas é paralela entre si e tem um deslocamento” (GIRDHAR; SCHEFFER, 2004c). A Figura 2 apresenta exemplos de desalinhamento do eixo, sendo eles: paralelo, angular e combinados.

As seguintes consequências, segundo (GIRDHAR; SCHEFFER, 2004c), são causadas no maquinário devido ao desalinhamento: i) desgaste e aquecimento do acoplamento; ii) e tensões anormais nos rolamentos de antifricção, devido às forças de fricção e flexão dentro do acoplamento; iii) possível falha do eixo; iv) transmissão de vibrações para outras partes da máquina por causa da fadiga gerada pelo aumento das cargas cíclicas.

Figura 2 – Tipos de desalinhamento.



Fonte: (BOSSIO, J. M.; BOSSIO, G. R.; ANGELO, 2009)

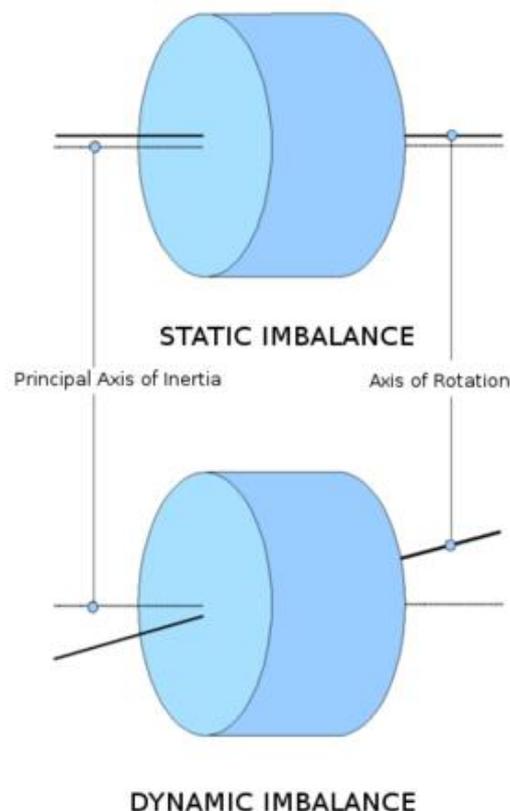
- Desbalanceamento

O desbalanceamento é definido muitas vezes como “a distribuição desigual de peso sobre o eixo de um rotor” (GIRDHAR; SCHEFFER, 2004c). É possível complementar essa definição com a visão de (ZHANG; YANG; WANG, 2019)

de que “o desbalanceamento ocorre quando o centro de rotação do rotor for diferente de seu centro de massa”. Também ressalta-se que o desbalanceamento resulta em vibrações indesejadas nos maquinários. As causas que resultam o desbalanceamento durante a operação normal das máquinas são: i) depósito irregular de produto em torno do ventilador ou impulsor da bomba; ii) lâminas ou palhetas danificadas ou ausentes; iii) e distorção térmica do rotor devido a desvios de temperatura no processo (GIRDHAR; SCHEFFER, 2004c).

A Figura 3 apresenta os dois tipos mais comuns de desbalanceamento do rotor. O desbalanceamento estático ocorre quando o eixo de rotação e o eixo principal de inércia estão paralelos entre si. No geral isso ocorre devido à má distribuição de massa em volta do eixo de rotação. Já o desbalanceamento dinâmico acontece quando o eixo principal de inércia não é paralelo e nem intercepta o eixo de rotação da máquina.

Figura 3 – Tipos de desbalanceamento.



Fonte: WikiWand (2020).<sup>1</sup>

- Folga rotativa

A folga rotativa é ocasionada, de acordo com (INC., 2022), por “folgas excessivas entre componentes rotativos e não rotativos ou estacionários, como rolamentos”.

A folga rotativa, segundo a equipe Dynamox, normalmente gera harmônicos nos dados do monitoramento de forma de onda e causa o aumento de ruído, impactando outras frequências. Os fatores que levam a máquina a apresentar essa falha são desgaste dos componentes e projeto inadequado (INC., 2022).

## 2.5 APRENDIZADO DE MÁQUINA

### 2.5.1 Árvore de Decisão

Os algoritmos de aprendizado de máquina podem ser simplificados para fins de entendimento como uma função que mapeia as variáveis de entrada, chamadas de *features*, para variáveis de saída, chamadas de *targets*. Cada algoritmo assume hipóteses sobre a função de mapeamento com o objetivo de realizar classificações (a *target* é categórica) ou regressões (*target* é um dado contínuo).

As árvores de decisão, segundo (GÉRON, 2019), são algoritmos versáteis de aprendizado de máquina, visto que conseguem realizar tarefas de classificação e regressão, e são fundamentais para o algoritmo de árvores aleatórias que será abordado no subcapítulo 2.5.3. É um método de aprendizado supervisionado, ou seja, é necessário passar um conjunto de dados rotulados para o algoritmo aprender quais os resultados de saída esperados. Além de ser um modelo não paramétrico, “é aquele que não pode ser caracterizado por um conjunto limitado de parâmetros” (GÉRON, 2019). Assim, esse algoritmo não faz suposições fortes sobre a função de mapeamento.

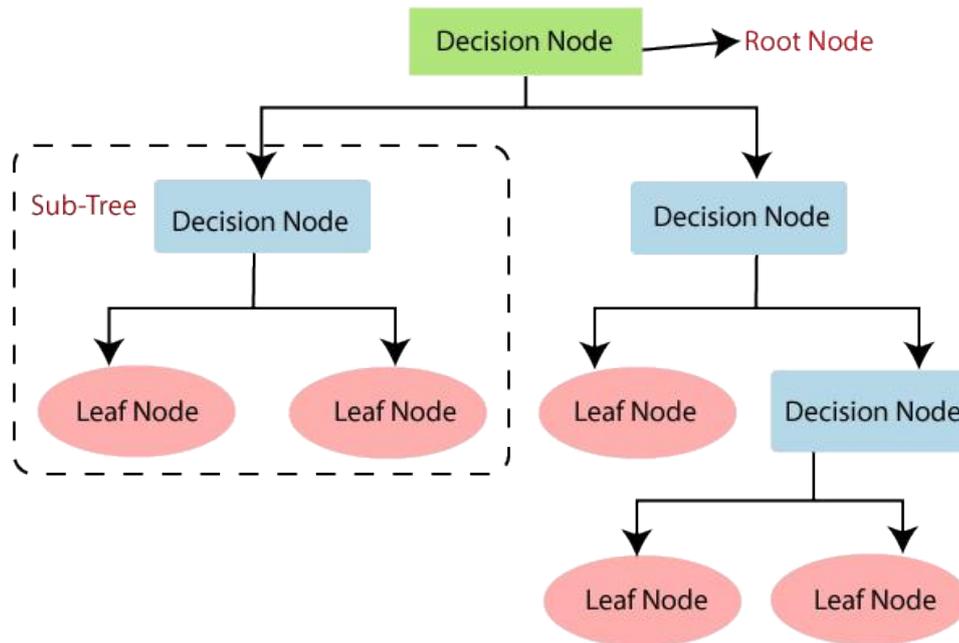
A árvore é representada de acordo com a Figura 4, sendo que o algoritmo começa no nó raiz (*root node*) com profundidade 0, verificando um primeiro atributo que, dependendo do resultado, direciona para um dos dois nós de decisão (*decision node*) com profundidade 1. Caso o nó da direita seja escolhido, então a árvore é direcionada para dois nós folhas (*leaf node*). Nesse caso, não há nenhum nó folha, e a decisão é obtida.

Para o algoritmo realizar as decisões e obter uma predição, são necessários alguns parâmetros, como aponta (GÉRON, 2019), sendo eles: i) quantidade de instâncias utilizadas de cada *target*; ii) quantidade de instâncias de cada *target* associada a cada nó de decisão, e iii) a impureza dos nós, sendo que um nó só é puro se todas as instâncias utilizadas pertencem a uma mesma *target*. Para mensurar a impureza dos nós, geralmente, são utilizadas as métricas Gini e entropia, sendo a métrica Gini calculada de acordo com a equação 1:

$$G_i = 1 - \sum_{k=1}^n p_{i,k}^2, \quad (1)$$

<sup>1</sup> Disponível em <[https://www.wikiwand.com/en/Tire\\_balance](https://www.wikiwand.com/en/Tire_balance)>.

Figura 4 – Exemplo genérico de árvore de decisão.



Fonte: Site Javatpoint.<sup>2</sup>

na qual,  $p_{i,k}$  é a proporção de instâncias de classe  $k$  entre as instâncias de treinamento no  $i^{\text{th}}$  nó.

Enquanto a entropia é calculada de acordo com a equação 2:

$$H_i = - \sum_{k=1}^n p_{i,k} \log_2(p_{i,k}), \quad (2)$$

sendo que  $p_{i,k} \neq 0$ .

A métrica Gini é mais rápida, pois não usa logaritmo como a entropia e pode prover um melhor desempenho do modelo ao ser utilizada (GÉRON, 2019). Além dos parâmetros citados, outros parâmetros também precisam ser definidos, sendo estes chamados de hiperparâmetros, pois são fixos antes do treinamento do modelo.

Os hiperparâmetros são importantes para restringir o algoritmo durante o treinamento, visto que a árvore de decisão é um modelo não paramétrico, então “se o algoritmo for deixado sem restrições, a estrutura da árvore se adaptará aos dados de treinamento, ajustando-se muito bem a eles, e resultando em *overfitting*” (GÉRON, 2019). O *overfitting* é discutido com mais detalhes no subcapítulo 2.5.3.

Alguns dos hiperparâmetros da árvore de decisão, que podem ser encontrados em (PEDREGOSA *et al.*, 2011), são: i) a profundidade máxima da árvore; ii) o número mínimo de amostras que um nó folha deve ter para ser considerado uma solução; iii) o

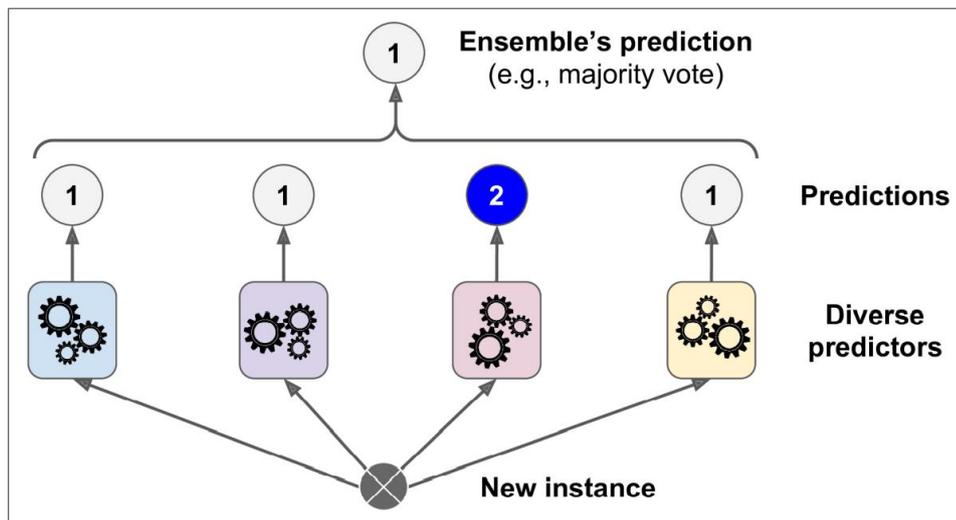
<sup>2</sup> Disponível em <<https://static.javatpoint.com/tutorial/machine-learning/images/decision-tree-classification-algorithm.png>>.

número máximo de nós folhas presentes na árvore; e iv) número máximo de recursos que são avaliados para divisão em cada nó.

### 2.5.2 Ensemble learning

A ideia do método *ensemble learning* é “selecionar uma coleção de hipóteses do espaço de hipóteses e combinar suas previsões” (GÉRON, 2019). A Figura 5 apresenta um tipo de *ensemble learning* chamado *hard classifier*, no qual são utilizados modelos diferentes, e escolhida a previsão que foi encontrada em mais modelos. A junção dos modelos resulta em um melhor desempenho do que utilizar apenas o melhor classificador do conjunto (GÉRON, 2019).

Figura 5 – Exemplo do processo de *ensemble learning* com o método *hard classifier*.

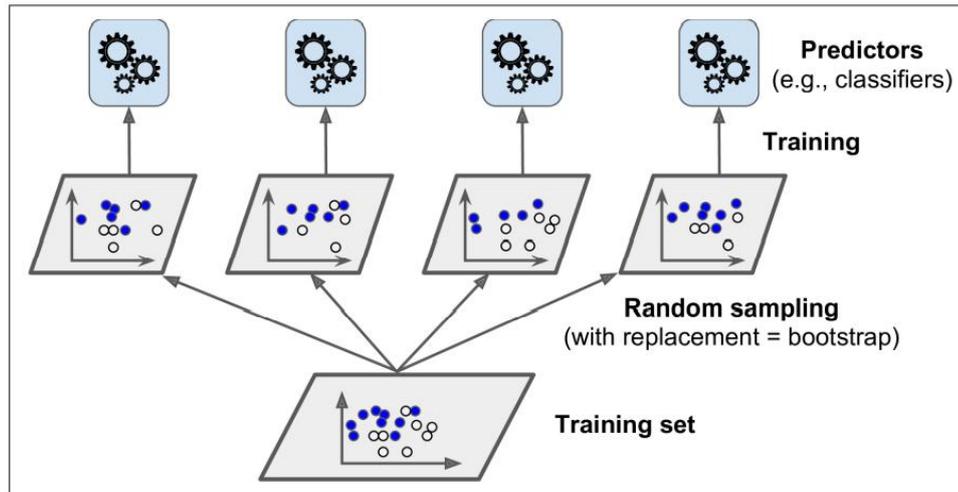


Fonte: (GÉRON, 2019).

Outro tipo de *ensemble learning*, ainda conforme (GÉRON, 2019), é utilizar o mesmo algoritmo de treinamento com subconjuntos aleatórios do conjunto de treinamento, sendo possível obter o subconjunto através de dois métodos: i) *bagging*, que resulta em um subconjunto que pode conter dados repetidos; ii) e *pasting*, que gera um subconjunto que não contém dados repetidos. A Figura 6 mostra como é realizado esse processo, no qual um conjunto de treinamento é dividido em subconjuntos aleatórios e cada subconjunto é aplicado em algoritmos do mesmo método.

### 2.5.3 Random Forest

O modelo de *random forest* é “um conjunto de árvores de decisões que introduz aleatoriedade extra ao procurar o melhor recurso entre um subconjunto aleatório de recursos” (GÉRON, 2019). Ainda, segundo (PEDREGOSA *et al.*, 2011), o algoritmo é

Figura 6 – Exemplo do processo de criação de subconjuntos com *bagging*.

Fonte: (GÉRON, 2019).

um *ensemble learning* que agrega várias árvores de decisão independentes e calcula a média de suas previsões probabilísticas.

Além dos hiperparâmetros da árvore de decisão, vistos no capítulo 2.5.1, outros precisam ser definidos, como a quantidade de árvores na floresta, sendo que quanto mais árvores, melhor o desempenho do modelo. Contudo, o aumento da quantidade de árvores resulta em um aumento do tempo de execução e também há um valor crítico, no qual o desempenho do algoritmo não aumenta de modo considerável.

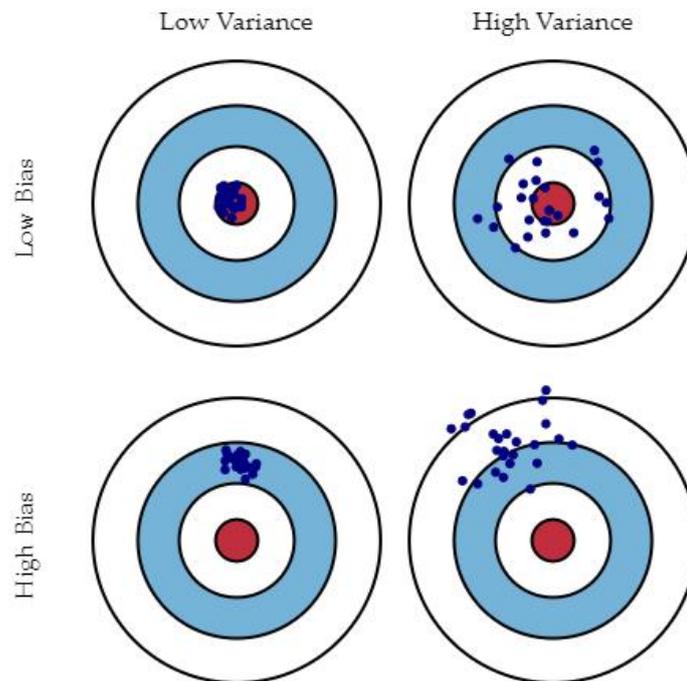
Outro hiperparâmetro é o tamanho dos subconjuntos aleatórios, sendo que com um tamanho menor, há uma redução da variância e um aumento do viés. Isso ocorre, pois ao aumentar o tamanho dos subconjuntos, está sendo diminuída a complexidade do modelo.

Ao aumentar a complexidade do modelo, geralmente resultará em um aumento da variância e redução do viés; o contrário ocorrendo ao diminuir a complexidade do modelo (GÉRON, 2019). Conforme (FORTMANN-ROE, 2017), o viés é definido como o erro causado por suposições erradas dos dados, por exemplo, assumir que os dados são lineares, mas na verdade, são quadráticos. Já a variância é o erro associado à sensibilidade do modelo para pequenas variações no conjunto de treinamento (FORTMANN-ROE, 2017).

A Figura 7 exemplifica as ocorrências desses erros. Nota-se que um bias alto faz com que os resultados fiquem distantes do valor correto (círculo central vermelho), enquanto uma variância alta, faz com que os resultados fiquem dispersos para um mesmo conjunto de dados.

O impacto do viés e variância estão relacionados diretamente com o *overfitting* e *underfitting* do modelo. Com uma variância alta, ocorre o *overfitting*, ou seja, o

Figura 7 – Exemplo de variância e viés.



Fonte: (FORTMANN-ROE, 2017).

modelo não está aprendendo com os dados de treinamento, mas se ajustando a eles, e isso resulta em um erro elevado nos dados de teste. Já com um alto viés, ocorre o *underfitting*, no qual o modelo não aprende com os dados de treinamento e resulta em um desempenho ruim tanto nos dados de treino como nos de teste. É necessário escolher hiperparâmetros que equilibrem ambos os erros, sendo isso feito por experimentos.

### 3 METODOLOGIA

Para a realização dessa pesquisa, foram realizados diversos procedimentos metodológicos. Em primeiro lugar, foi feita a análise exploratória dos dados fornecidos pela Dynamox, com o intuito de identificar quais informações são relevantes para o problema e investigar a qualidade dos dados, verificando possíveis anomalias e valores discrepantes que poderiam agregar ruído no modelo.

A análise mostrou que alguns relatórios de falhas apresentavam inconsistências. Diante disso, foi aplicado um processamento nos relatórios a fim de eliminar as falhas com inconsistências e realizar as análises dos dados de vibração com falhas válidas. A *pipeline* de processamento consistia em:

1. Remoção de relatórios de falhas que possuíam na coluna diagnóstico descrições como “desconsiderar”, “sensor parou” e “dados não estão sendo coletados”;
2. Remoção de relatórios que apresentavam cadastro de uma falha e situação de máquina A0, que é indicativo de normalidade;
3. Remoção de relatórios sem cadastro de falha, mas com situação de máquina A1 ou A2, que são indicativos de possível início de falha e falha presente, respectivamente.

Também foi incluído um processamento para os relatórios de normalidade, assim, apenas os relatórios que possuíam no diagnóstico pelo menos uma das seguintes frases foram considerados: i) equipamento em bom estado; ii) equipamento com níveis normais de vibração; iii) equipamento apresenta vibração estável; iv) e equipamento apresenta valores globais dentro dos limites. Uma precaução adotada nesse processamento foi impor que as letras da coluna diagnóstico fossem todas minúsculas, desse modo garantindo que o procedimento abrangeria mais casos.

Outro processamento realizado nos relatórios foi a seleção das falhas mecânicas desejadas: desalinhamento, desbalanceamento, folga estrutural, folga rotativa, folga do mancal, perda de rigidez e engrenagens desalinhadas, pois são as falhas que mais possuem relatórios nos dados fornecidos pela Dynamox e também são as falhas mais comuns em máquina rotativas. Além da remoção de falhas de um mesmo *spot* com datas de cadastro, chamadas de data de criação, muito próximas, pois isso resultaria em amostras com dados de sinais de vibração iguais. Esse *spot* é uma região da máquina na qual os sinais de vibração estão sendo medidos e coletados, sendo que uma máquina pode apresentar mais de um *spot*

A partir disso, definiu-se o conjunto de treinamento e o conjunto de teste. Cada amostra presente em ambos os conjuntos é associada a uma falha de um determinado *spot* de uma máquina e a sua data de criação. Assim, a falha também está relacionada

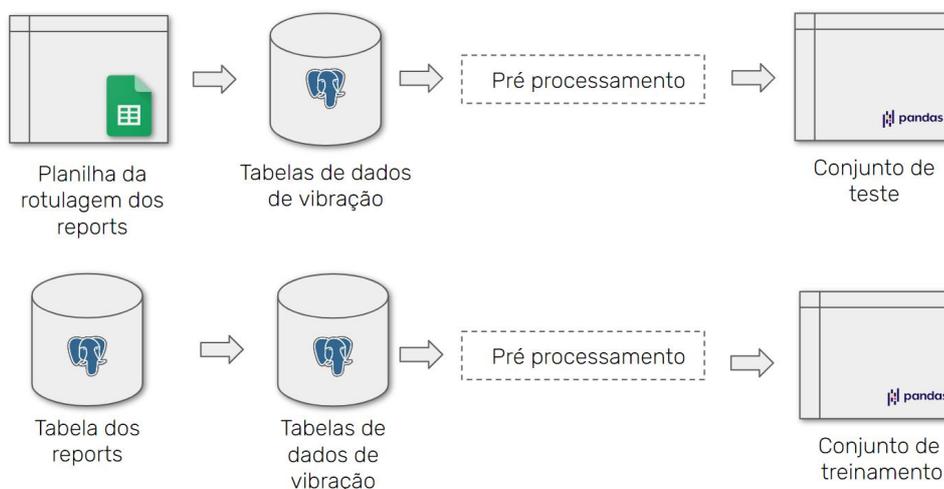
ao conjunto de características de vibração dos dados de telemetria e dos dados de forma de onda do *spot*.

O conjunto de treinamento foi constituído de todas as amostras que não entraram no conjunto de teste, sendo ele constituído de amostras selecionadas manualmente, baseadas nos dados de telemetria a partir de um *dashboard*. Optou-se pelo uso dos dados de telemetria, pois as falhas mecânicas utilizadas neste trabalho se manifestam em baixa frequência e, teoricamente, deveriam apresentar um comportamento de falha nos dados de telemetria. A seleção manual é importante para garantir que realmente existe uma falha na data de criação relatada nos dados de vibração.

Após a seleção inicial das amostras, foram definidas as *features* a serem utilizadas nos modelos de *random forest*. Em relação aos dados de telemetria, foram consideradas as seis *features* existentes: aceleração e velocidade para os eixos X, Y e Z. Enquanto, para a escolha das *features* dos dados do monitoramento de forma de onda, foi realizada outra análise exploratória dos dados fornecidos inicialmente pela Dynamox, pois havia em torno de 259 *features*. Priorizou-se selecionar *features* com dados não nulos presentes na maior quantidade de amostras possíveis.

Para finalizar a geração do conjunto de treinamento e do conjunto de teste, foi definida a *pipeline* para o processamento dos dados, sendo os procedimentos aplicados baseados nas informações descobertas com as análises exploratórias realizadas anteriormente. A Figura 8 apresenta a origem dos dados para a geração de cada conjunto, sendo que as amostras presentes na planilha e que foram usadas para a geração do conjunto de teste são desconsideradas na geração do conjunto de treinamento.

Figura 8 – Origem dos dados para geração do conjunto de treinamento e conjunto de teste.



Fonte: Autoria Própria.

O objetivo da *pipeline* de processamento é gerar um *dataset* final sem dados

nulos, contendo os sinais de vibração de 14 dias de dados de telemetria e 1 dia de dados do monitoramento de forma de onda, pois esses dados possuem mais informações sobre as características do sinal do que os dados de telemetria, então apenas 1 dia já foi considerado suficiente.

A escolha de 14 dias de dados para os dados de telemetria foi baseada na ideia de que é preciso selecionar tempo suficiente para o modelo conseguir aprender o comportamento vibratório daquela parte da máquina e detectar se há, ou não, falha mecânica. Contudo, o intervalo não pode ser muito longo, visto que será adicionado ruído ao modelo. Assim, pelas análises exploratórias realizadas com o *dashboard*, foi verificado que 14 dias antes da data de criação seriam suficientes para a detecção de falha mecânica.

A *pipeline* de processamento é apresentada em alto nível na Figura 9. Inicialmente, são selecionados 14 dias de dados para os dados de telemetria e os dados do monitoramento de forma de onda, depois são removidos os intervalos de tempo que a máquina se encontra parada, pois nenhum conhecimento é obtido sobre a máquina quando esta não se encontra operando. Além de ser selecionado o eixo de amplitude dominante para cada *feature*.

Essa seleção é importante, pois com base nos alinhamentos feitos com a equipe da Dynamox, geralmente é em algum eixo que a falha se manifesta. Assim, seria possivelmente redundante ter os outros dois eixos. Em seguida, é feita uma interpolação linear, caso os sinais de vibração dos dados de telemetria possuam dados faltantes e, por último, são salvos em um *dataset* os 14 dias de dados de telemetria e o último dia de dados não nulos imediatamente antes da data de criação dos dados do monitoramento de forma de onda.

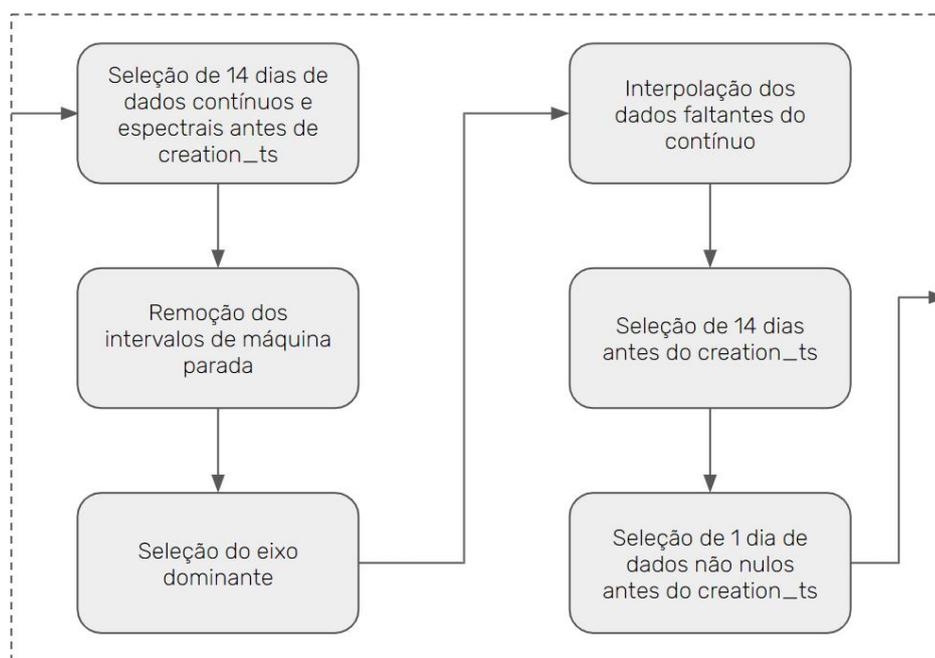
Com o conjunto de treinamento e conjunto de teste definidos, a modelagem é a próxima etapa desenvolvida no projeto. A busca dos hiperparâmetros do modelo foi realizada com o algoritmo de busca aleatória<sup>5</sup> com estratégia de validação cruzada para a divisão dos dados em treinamento e validação. Optou-se pela busca aleatória devido ao tempo computacional ser menor em comparação com a busca exaustiva (*Grid Search*<sup>6</sup>), além da validação cruzada para garantir que o modelo apresenta bons resultados, visto que são criados diferentes conjuntos de treino e validação.

Foi utilizada a métrica de acurácia balanceada para o *score* da otimização de hiperparâmetros. Essa métrica mede o quão bem o modelo está predizendo em relação a cada classe do conjunto de dados. Assim, ela é útil na otimização, pois o *score* é calculado independentemente do desbalanceamento entre as classes. Também foi

<sup>5</sup> Disponível em <[https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.RandomizedSearchCV.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html)>.

<sup>6</sup> Disponível em <[https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.GridSearchCV.html?highlight=grid+search#sklearn.model\\_selection.GridSearchCV](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html?highlight=grid+search#sklearn.model_selection.GridSearchCV)>.

Figura 9 – Pipeline em alto nível do processamento dos dados para criação do conjunto de treinamento e teste.



Fonte: Autoria Própria.

fixado o parâmetro de *random seed* para os métodos de busca aleatória e validação cruzada. Isso garante uma comparação mais justa dos resultados, pois será utilizado o mesmo conjunto de treinamento e validação ao utilizar diferentes hiperparâmetros e modelos.

Em relação aos experimentos realizados neste trabalho, foram definidos um total de 6 experimentos, que consistem em duas rotulações diferentes de falhas mecânicas e 3 variações de dados de entrada. Assim, para cada rotulação, foram aplicados três conjuntos de dados de entrada diferentes. Em relação à rotulação, os experimentos realizados foram:

- Aplicação de um modelo de classificação binária de *random forest* distinto para cada uma das falhas mecânicas específicas: desbalanceamento, desalinhamento e folga rotativa;
- Aplicação de um modelo de classificação binária de *random forest* para todas as falhas mecânicas: desalinhamento, desbalanceamento, folga rotativa, engrenagens desalinhadas, folga estrutural, perda de rigidez e folga no mancal;

Enquanto os três conjuntos de dados de entrada aplicados foram:

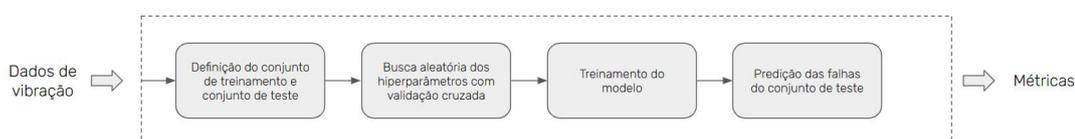
- Conjunto contendo apenas os dados de telemetria;

- Conjunto contendo apenas os dados do monitoramento de forma de onda;
- Conjunto contendo os dados de telemetria e do monitoramento de forma de onda.

Por último, para a validação e análise do desempenho dos modelos foram utilizadas as métricas de sensibilidade e acurácia balanceada, além de ser verificada a importância das *features* adotada pelo modelo de *random forest* por meio da análise dos valores médios de redução de impurezas. Portanto, todas as 5 primeiras etapas da metodologia CRISP-DM foram abordadas e desenvolvidas no decorrer deste trabalho.

A Figura 10 apresenta a *pipeline* simplificada do projeto, desenvolvidas a partir dos dados de vibração e que retornam no final as métricas para avaliação dos resultados. Destaca-se que as etapas de busca aleatória dos hiperparâmetros e treinamento do modelo são executadas 5 vezes e posteriormente é retirado o valor médio das métricas dessas 5 execuções. Esse procedimento garante que é obtido uma medida mais precisa da performance do modelo, pois fatores aleatórios, como a inicialização dos pesos do modelo, são atenuados. O Quadro 1 apresenta os valores de acurácia balanceada ao executar os modelos de falhas específicas em cada uma das 5 iterações, no qual nota-se que as diferenças de valores são na faixa de 0.04.

Figura 10 – *Pipeline* simplificada do projeto.



Fonte: Autoria Própria.

Quadro 1 – Valores da acurácia balanceada ao executar a pipeline de otimização dos hiperparâmetros e treinamento do modelo em 5 iterações.

		Valor da acurácia balanceada				
Falhas	Features	Iteração				
		0	1	2	3	4
Desalinhamento	Dados de telemetria e dados do monitoramento de forma de onda	0,568	0,546	0,588	0,569	0,590
	Dados de telemetria	0,544	0,522	0,538	0,522	0,584
	Dados do monitoramento de forma de onda	0,641	0,594	0,643	0,612	0,615
Folga rotativa	Dados de telemetria e dados do monitoramento de forma de onda	0,709	0,709	0,715	0,692	0,695
	Dados de telemetria	0,633	0,630	0,624	0,633	0,619
	Dados do monitoramento de forma de onda	0,792	0,792	0,789	0,792	0,792
Desbalanceamento	Dados de telemetria	0,579	0,497	0,549	0,570	0,573
	Dados de telemetria	0,543	0,558	0,572	0,549	0,561
	Dados do monitoramento de forma de onda	0,628	0,631	0,605	0,561	0,587

Fonte: Autoria Própria.

## 4 ANÁLISE EXPLORATÓRIA

Neste capítulo, são descritas as análises realizadas nos dados fornecidos pela Dynamox e os *insights* obtidos, que serão, posteriormente, usados para a definição dos processamentos para o tratamento dos dados e seleção das *features*.

### 4.1 CONJUNTO DE DADOS

Os dados foram fornecidos pela Dynamox por meio de arquivos no formato *pickle* que é um formato de dados específico do Python. Cada arquivo *pickle* era formado por um dicionário contendo as informações de um *spot* específico. Ao todo foram fornecidos 1699 *spots* com informações entre 02/09/2021 até 21/10/2022, sendo que esse período variava em 4 possibilidades, como mostrado no Quadro 2. Inicialmente, foram fornecidos os dados de março até abril, depois de abril até maio e, por último, de maio até outubro.

Quadro 2 – Períodos de dados fornecidos pela Dynamox.

Data inicial	Data final
03/10/2021	01/04/2022
02/09/2021	01/03/2022
02/11/2021	01/05/2022
01/05/2022	21/10/2022

Fonte: Autoria Própria.

Para cada *spot*, as informações dos arquivos *pickle* consistiam em: i) sinais de vibração dos dados de telemetria; ii) sinais de vibração dos dados do monitoramento de forma de onda; iii) alertas para os dados de telemetria que indicavam os possíveis valores máximos que o sinal não deveria ultrapassar para preservar a saúde da máquina, sendo esses valores definidos e cadastrados pelo próprio analista para cada máquina ; iv) *asset* que o *spot* é associado, ou seja, qual grupo que esse *spot* pertence; v) condições de alerta; vi) e os relatórios de falha que eram uma lista de dicionários, sendo que cada um deles continha o relatório de uma falha associada a esse *spot*.

### 4.2 RELATÓRIO DE FALHAS

Como o foco do trabalho é identificar as falhas das máquinas, inicialmente foram analisados os relatórios de falha disponibilizados pela Dynamox, no qual verificou-se que as informações que cada *spot* possui são: i) o identificador do *spot*, que se refere à identidade única de cada *spot* cadastrado no sistema; ii) falha que foi detectada pelo analista de vibração; iii) data de cadastro da falha no sistema, sendo abreviada

para data de criação; iv) situação que a máquina se encontra, abreviada para *status* e com possibilidade de marcar A0, A1 e A2; v) e diagnóstico da falha, no qual o analista descreve informações que achar relevante sobre a sua inspeção.

Foi constatado, com as análises, que havia relatórios duplicados, ou seja, para um mesmo *spot* foi cadastrado a mesma falha com a mesma data de criação e com o mesmo diagnóstico, como mostrado na Figura 11. Isso ocorreu, de acordo com a equipe da Dynamox, pois o analista não podia excluir um registro após este ter sido cadastrado na ferramenta.

Figura 11 – Relatórios de falhas duplicados.

creation_ts	spot_id	status	failure	diagnostic
2022-02-10 13:11:03.914000+00:00	1	A2	unbalance	Foi identificado sintomas de desbalanceamento ...
2022-02-10 13:11:03.914000+00:00	1	A2	unbalance	Foi identificado sintomas de desbalanceamento ...
2022-02-07 17:06:24.757000+00:00	2	A2	eccentric-gears	IE-VB-DEFEITO NAS ENGRENAGENS DO REDUTOR
2022-02-07 17:06:24.757000+00:00	2	A2	eccentric-gears	IE-VB-DEFEITO NAS ENGRENAGENS DO REDUTOR
2022-01-06 19:59:11.560000+00:00	3	A2	unbalance	Foi identificado sintomas de 1xRPM do rotor po...
2022-01-06 19:59:11.560000+00:00	3	A2	unbalance	Foi identificado sintomas de 1xRPM do rotor po...

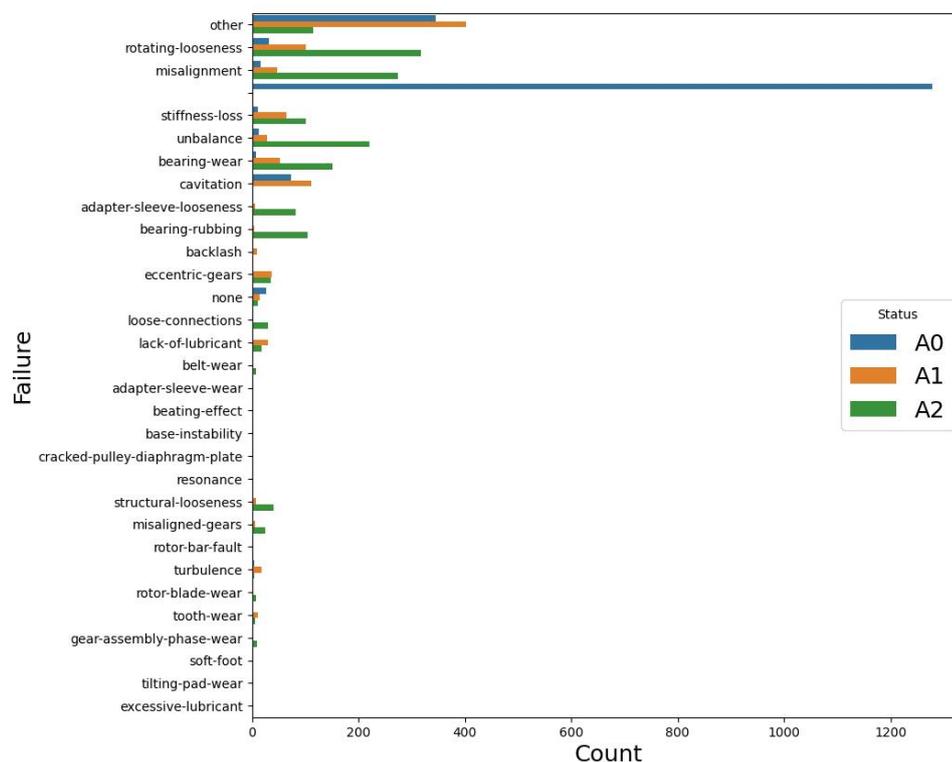
Fonte: Autoria Própria.

As informações dos relatórios de falha são preenchidas pelos analistas de vibração, então são passíveis de erro humano e também dependem da interpretação de cada pessoa. Por exemplo, a situação da máquina é algo relativo para cada analista e, mesmo com o diagnóstico, pode não estar claro o que o analista quis expor. Assim, a partir das conversas com a equipe da Dynamox e exploração dos dados, verificou-se que havia algumas inconsistências nos relatórios preenchidos.

É esperado que um relatório com *status* A0 indique que a máquina não apresenta falha, ou seja, se encontre em situação de normalidade. Contudo, diversos relatórios de falha apresentavam *status* A0 e alguma falha associada. É assumido para esses casos que, mesmo com uma falha presente, a máquina estava operando bem e o analista optou por marcar *status* A0 e sinalizar no diagnóstico os pormenores da saúde da máquina.

Em outros relatórios, o cenário inverso também foi encontrado, ou seja, o *status* foi cadastrado como A1 ou A2, mas nenhuma falha foi associada, sendo que, idealmente, apenas o *status* A0 não seria associado a nenhuma falha. A Figura 12 apresenta a contagem de relatórios por *status* da máquina para cada falha dividida. Idealmente, o item “none” deveria conter apenas *status* A0, enquanto as demais falhas apenas *status* A1 ou A2.

Também foi verificado que alguns relatórios tiveram os diagnósticos preenchidos com conteúdos controversos, como “teste”, “desconsiderar”, “sensor parou”, “dados não estão sendo coletados”. Esses comentários são um forte indicativo que o relatório,

Figura 12 – Contagem de relatórios por situação (*status*) da máquina para cada falha.

Fonte: Autoria Própria.

possivelmente, não é confiável para ser usado no treinamento e teste do modelo, pois não faz sentido treinar e testar o modelo com dados errados.

Além disso, outra incoerência foi encontrada nos dados fornecidos dos meses de Maio até Outubro. Esse novo conjunto de dados possuía novas informações, sendo elas: i) data de detecção da falha; ii) data para tratar o problema (*deadline*); iii) ação a ser realizada; iv) nota, na qual o analista pode adicionar as observações que julgar necessárias; v) e se o relatório foi deletado ou não. Contudo, esse dado possuía apenas Falso como valor, então não poderia ser usado para remover relatórios duplicados erroneamente.

Com a data de detecção da falha, foi verificado que em alguns casos havia uma diferença negativa entre a data de criação do relatório e a data de detecção de falha, ou seja, a detecção da falha ocorreu depois da criação do relatório. De acordo com a equipe da Dynamox, a data de detecção deveria ser preenchida com a data que a falha tornou-se visível nos sinais de vibração, enquanto a data de criação é a data em que o analista efetivamente viu os dados e relatou a falha. Assim, não faria sentido o analista registrar a falha antes que ela fosse visível nos sinais de vibração.

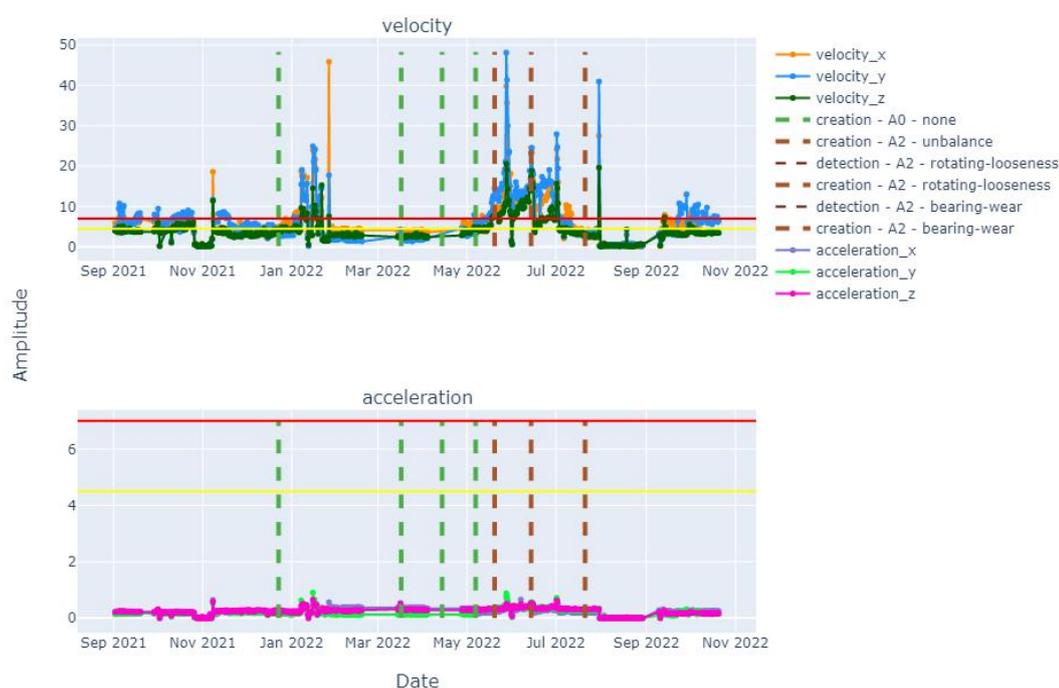
A partir dessas análises foi concluído que seria necessário realizar um conjunto de processamentos nos relatórios, a fim de garantir que as falhas utilizadas para o

treinamento e teste do modelo realmente fossem factuais e condizentes com os dados do sinal de vibração.

### 4.3 SINAIS DE VIBRAÇÃO

Os sinais de vibração dos dados de telemetria consistem em aceleração RMS e velocidade RMS para os eixos X, Y e Z, como mostrado na Figura 13. Todos os *spots* possuem pelo menos 1 dia de dados não nulos de aceleração e velocidade. Também é possível visualizar, pela Figura 13, os valores de alerta associados ao *spot*, sendo que pode-se ter os valores de alerta para a velocidade e/ou para a aceleração. Como dito no capítulo 4.1, esses alertas indicam os valores máximos que os sinais de vibração deveriam estar para a máquina operar no seu regime ideal.

Figura 13 – Sinais de vibração dos dados de telemetria juntamente com os alertas e relatórios de falha cadastrados.



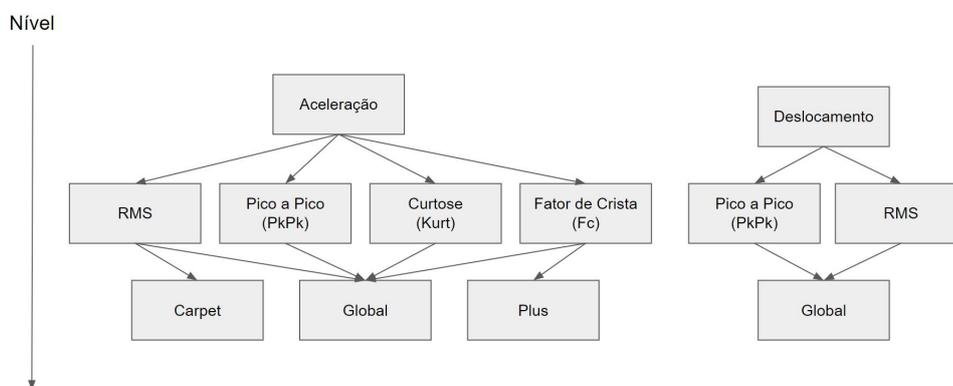
Fonte: Autoria Própria.

Na Figura 13, as faixas amarelas representam a região na qual a máquina já está operando acima do seu regime ideal, enquanto as faixas vermelhas indicam a região de operação considerada perigosa para o seu funcionamento. Contudo, não é significativa a relação entre os valores de alerta e a existência de uma falha na máquina de acordo com a equipe da Dynamox. O que em geral acontece é que as máquinas

trabalham acima da faixa amarela e isso não é considerado necessariamente um alerta para os analistas.

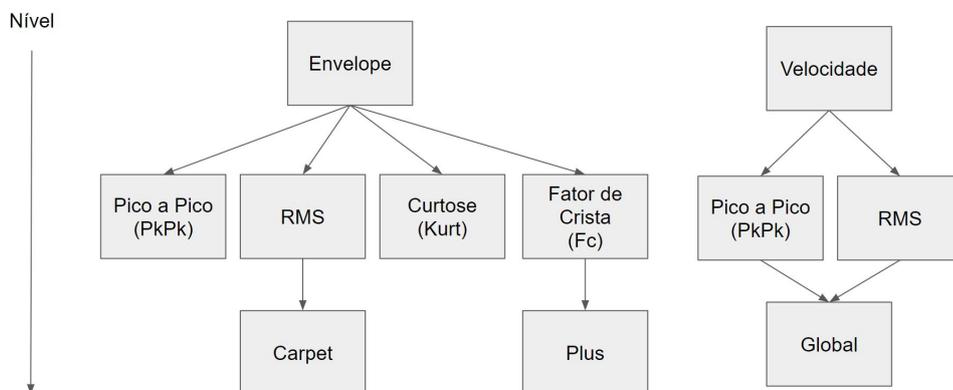
Em relação aos sinais de vibração dos dados do monitoramento de forma de onda, são analisados os sinais de aceleração, deslocamento, velocidade e envelope para cada eixo X, Y e Z e para diferentes faixas de frequência. A partir desses sinais é possível obter outros parâmetros, como apresentado nas Figuras 14 e 15. Por exemplo, o sinal de deslocamento, apresentado na Figura 14, possui as *features*: Deslocamento Pico a Pico Global e Deslocamento RMS Global. A mesma lógica é aplicada para os demais sinais.

Figura 14 – Características presentes e analisadas da aceleração e deslocamento obtidos nos dados do monitoramento de forma de onda.



Fonte: Autoria Própria.

Figura 15 – Características presentes e analisadas da velocidade e envelope obtidos nos dados do monitoramento de forma de onda.

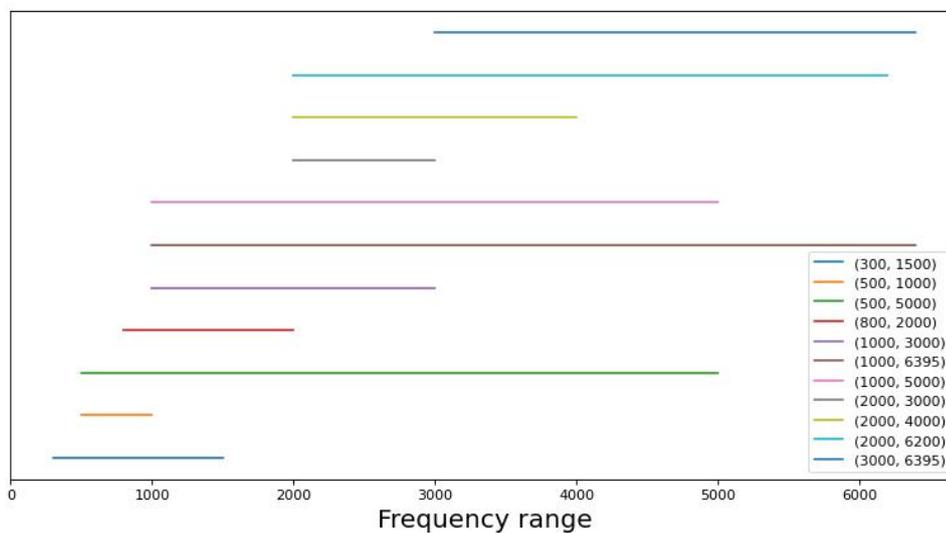


Fonte: Autoria Própria.

Também há outros parâmetros presentes no conjunto de dados fornecidos pela

Dynamox, como aceleração *Cepstral* e aceleração *Skewness*. Contudo, essas medidas aparecem em poucas amostras e possuem dados faltantes. Dessa forma, foram desconsideradas das análises, sendo concentradas nas verificações das *features* apresentadas nas Figuras 16 e 17. Por último, constatou-se que há um total de 22 faixas de frequências nas *features* de forma de onda e que elas englobam baixas e altas frequências, variando de 5 Hz até 6000 Hz. Outra verificação é que as faixas de frequência se cortam, por exemplo, a faixa de frequência de 5 Hz a 20 Hz é englobada pela faixa de frequência de 5 a 50 Hz, como mostrado na Figura 17.

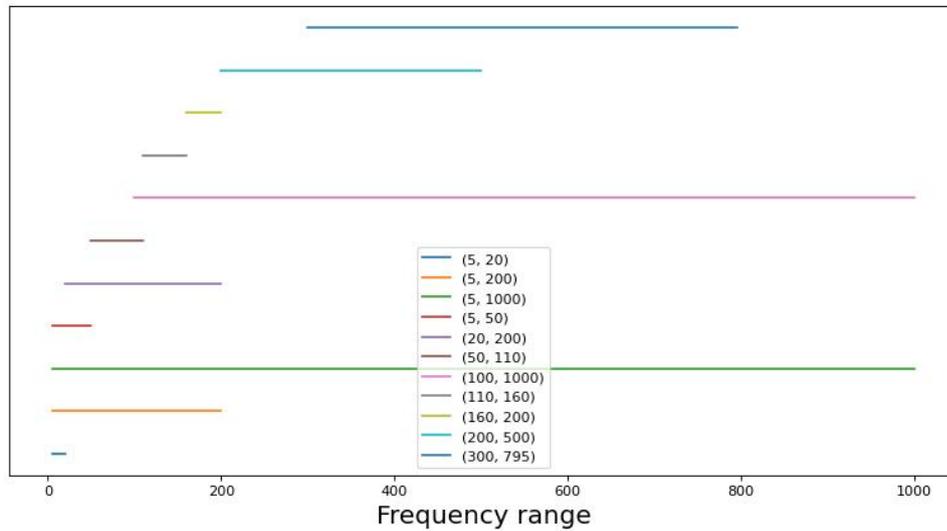
Figura 16 – Faixas que adotam altas frequências e estão presentes nos sinais dos dados do monitoramento de forma de onda.



Fonte: Autoria Própria.

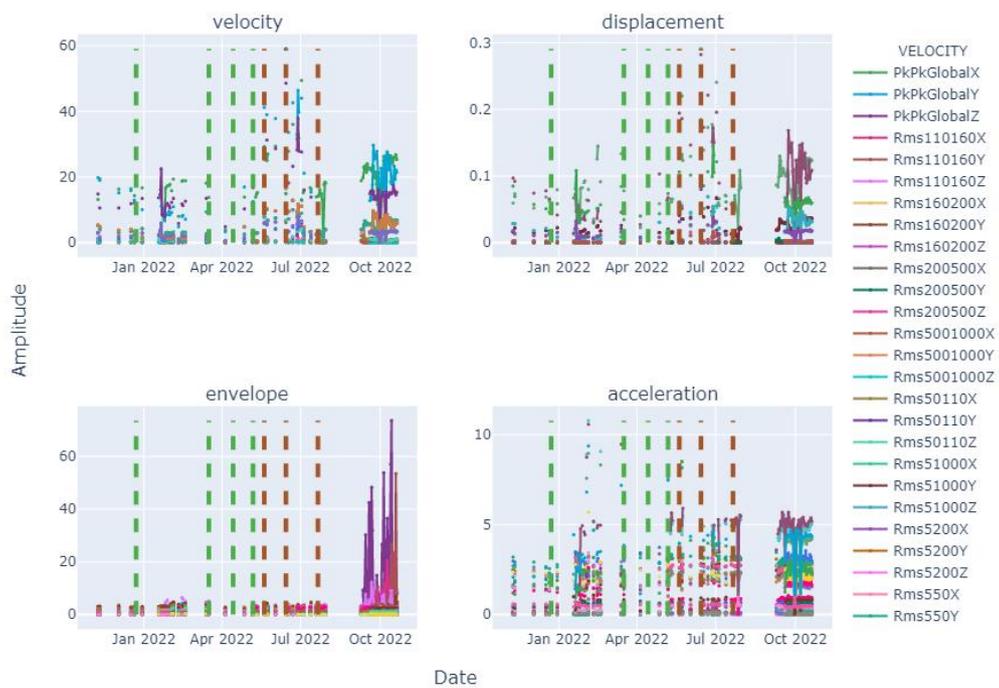
A Figura 18 apresenta os sinais de vibração associados a um *spot*. Nota-se que há uma grande quantidade de *features* para cada *spot*. Em média, os *spots* apresentam 259 *features*, contudo os dados em sua maioria são faltantes, sendo necessário ponderar a quantidade de dados nulos na escolha das *features* que serão utilizadas no modelo.

Figura 17 – Faixas que adotam baixas frequências e estão presentes nos sinais dos dados do monitoramento de forma de onda.



Fonte: Autoria Própria.

Figura 18 – Sinais de vibração dos dados do monitoramento de forma de onda juntamente com os relatórios de falha cadastrados.



Fonte: Autoria Própria.

## 5 PROCESSAMENTO DOS DADOS

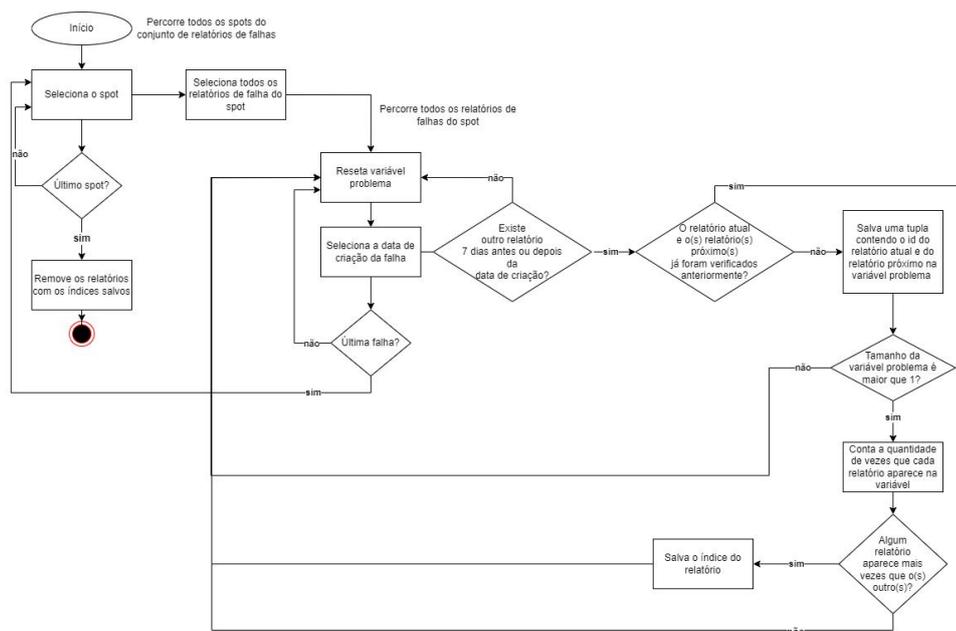
Neste capítulo, serão apresentadas as implementações das pipelines de processamento dos dados para geração do conjunto de treinamento e conjunto de teste. O processamento engloba o tratamento dos relatórios de falha e dos sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda.

### 5.1 TRATAMENTO DOS RELATÓRIOS DE FALHA

No tratamento dos relatórios de falha, optou-se por aplicar processamentos que eliminassem o máximo de relatórios com as inconsistências observadas na análise exploratória. Assim, é garantida uma maior confiabilidade nas falhas utilizadas para o treinamento e teste do modelo.

A Figura 19 apresenta o fluxograma do processo de remoção dos relatórios de falhas com data de criação próximas, no qual para cada *spot*, do conjunto de relatórios de falhas, é verificado se há falhas próximas considerando 7 dias antes ou depois da data de criação do relatório. Após essa verificação para cada relatório de falha, caso haja relatórios próximos, é removido o relatório que mais possui data de criação próxima de outros. Se todos estiverem próximos na mesma quantidade, é removido aleatoriamente um relatório.

Figura 19 – Fluxograma do processo de remoção dos relatórios de falhas com data de criação próximas.

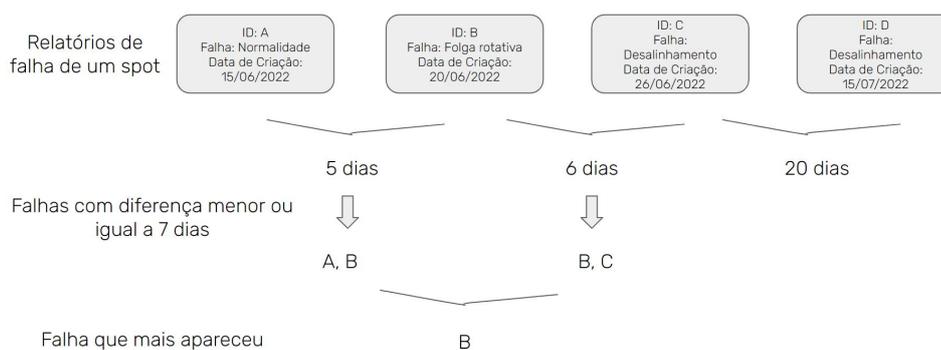


Fonte: Autoria Própria.

Para clarificar o procedimento de relatórios com data de criação próxima, é

apresentado um exemplo na Figura 20, no qual a falha com ID B será desconsiderada como amostra, pois possui a data de criação com uma diferença menor que 7 dias em relação às datas de criação de outros 2 relatórios.

Figura 20 – Exemplo do processo de remoção dos relatórios de falhas com datas de criação próximas.



Fonte: Autoria Própria.

## 5.2 SELEÇÃO DAS AMOSTRAS DO CONJUNTO DE TESTE

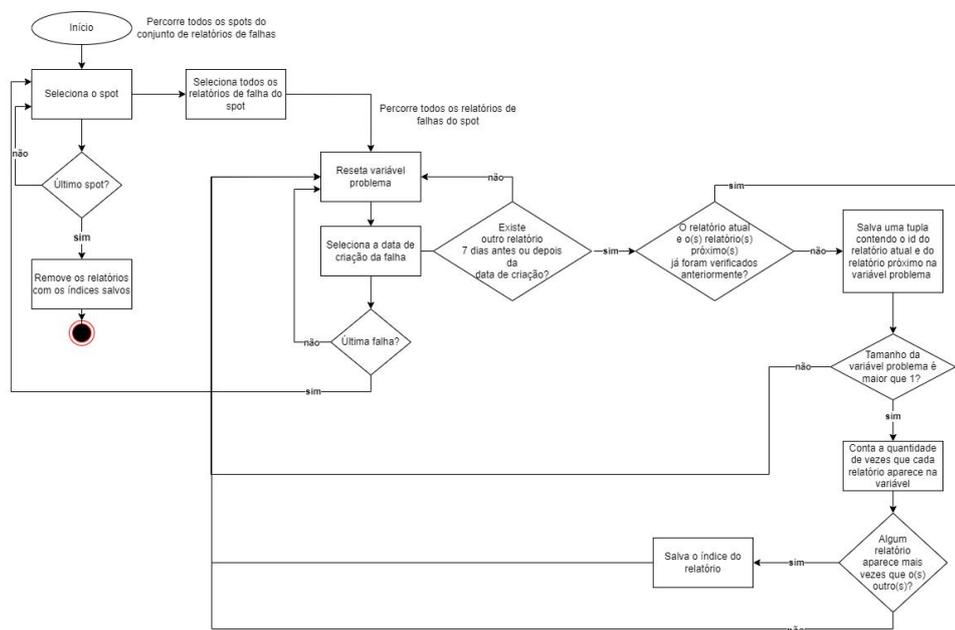
As amostras do conjunto de teste foram selecionadas através do *dashboard* da Figura 21. Ele consiste em: i) um menu *dropdown* para a seleção do relatório de falha; ii) um *checkboxlist* para selecionar os *spots* associados ao relatório de falha; iii) um botão de submeter para salvar em um arquivo de texto os *spots* que realmente apresentam falha nos sinais de vibração dos dados de telemetria; iv) um texto para informar se os *spots* do relatório de falha apresentado já foram verificados; v) e as séries temporais da velocidade e aceleração dos *spots* selecionados.

É possível que um relatório de falha esteja associado a mais de um *spot*, visto que o analista consegue replicar um relatório para todos os *spots* de uma máquina na ferramenta da Dynamox. Assim, mesmo sem o *spot* apresentar a falha nos seus sinais de vibração, o analista pode registrar um relatório de falha por replicação.

Os relatórios mostrados no *dashboard* são aqueles que contêm falhas mecânicas, possuem situação de falha A2 e mais de um *spot* associado à falha, sendo que nos relatórios são informados os *spots* que a falha registrada está associada, na coluna *report group*. A partir desse dado, são preenchidos os *spots* apresentados no *checkboxlist* do *dashboard*.

Com o uso do *dashboard* foram selecionados 114 relatórios de falhas, que foram salvos em uma planilha de dados contendo: a falha, sua data de criação, o *spot* associado, as datas de início e fim em que realmente foi verificada a presença de falha nos dados de telemetria e o número do intervalo, pois um relatório de falha poderia

Figura 21 – Dashboard para seleção das amostras do conjunto de teste.



Fonte: Autoria Própria.

ter mais de um período, no qual a falha foi realmente verificada. Um fragmento do *dataset* gerado é apresentado na Figura 22, na qual é possível verificar que o *spot* das linhas 7 e 8 apresenta dois intervalos, porque a falha de desbalanceamento criada em 16/12/2021 foi verificada em dois intervalos de tempo. O mesmo também acontece para a falha de folga rotativa, criada em 23/12/2021.

Figura 22 – Fragmento dos relatórios de falha selecionados através do *dashboard*.

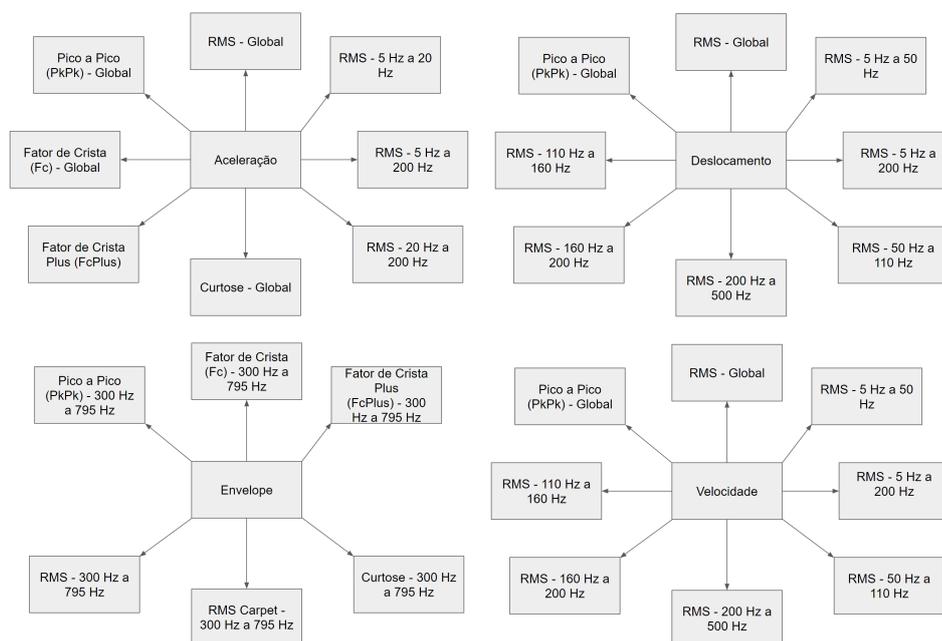
interval	spot_id	creation_ts	failure	start_ts	end_ts
1	1	2021-12-16	unbalance	2021-12-02	2022-01-18
2	1	2021-12-16	unbalance	2022-02-03	2022-02-28
2	2	2021-12-23	rotating-looseness	2022-02-26	2022-03-10
1	2	2021-12-23	rotating-looseness	2021-10-18	2022-01-02

Fonte: Autoria Própria.

### 5.3 SELEÇÃO DAS *FEATURES* DE FORMA DE ONDA

Para a seleção das *features* de forma de onda foi realizada uma análise dos dados de vibração das amostras do conjunto de teste. O objetivo é garantir a seleção da maior quantidade de *features* com dados não nulos, pois será considerado apenas



Figura 24 – *Features* de forma de onda selecionadas para o modelo de *random forest*.

Fonte: Autoria Própria.

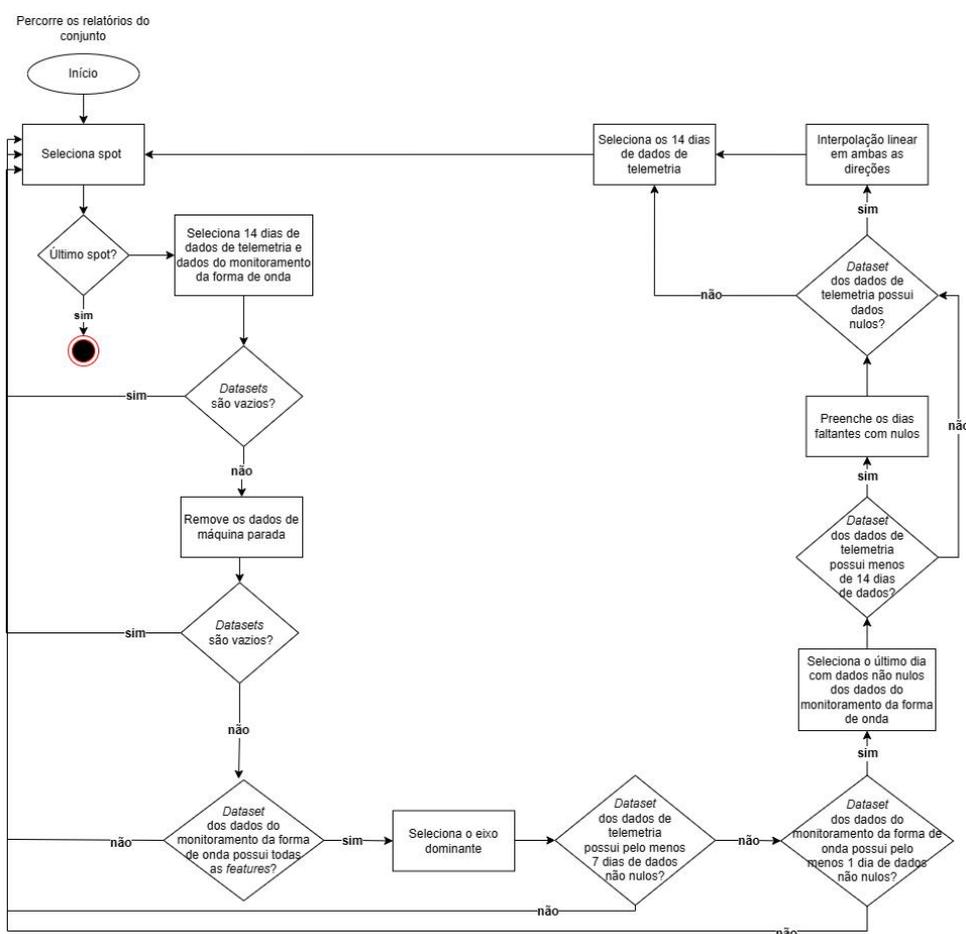
#### 5.4 PIPELINE DE CRIAÇÃO DO CONJUNTO DE TREINAMENTO E DO CONJUNTO DE TESTE

Com os relatórios de falhas tratados, é possível gerar o conjunto de treinamento e conjunto de teste baseado nas amostras de falhas. A Figura 25 apresenta o fluxograma do processamento para criação desses conjuntos, no qual para cada amostra são selecionados 14 dias de dados de telemetria e do monitoramento de forma de onda antes da data de criação da falha. A partir dessa seleção, são gerados dois *datasets*: *dataset* de telemetria, que contém os sinais de vibração dos dados de telemetria; e o *dataset* de forma de onda, que contém os sinais de vibração dos dados do monitoramento de forma de onda.

Se existir pelo menos 1 dia de dados nesse intervalo de 14 dias tanto nos dados de telemetria quanto nos dados do monitoramento de forma de onda, é aplicado o procedimento de remoção de máquina parada. Neste trabalho, a detecção de máquina parada foi feita verificando se a aceleração RMS estava abaixo de um valor limite. A escolha da aceleração RMS como uma boa *feature* para detectar se a máquina encontra-se ou não parada foi verificada pelo gráfico dos sinais de vibração e indicados pela equipe da Dynamox. Nos sinais de vibração, foi possível identificar padrões claros quando a máquina estava parada em comparação com a velocidade.

É verificado se para cada dia de dados, a aceleração RMS máxima entre os

Figura 25 – Fluxograma do processamento dos dados para criação do conjunto de treinamento e conjunto de teste.



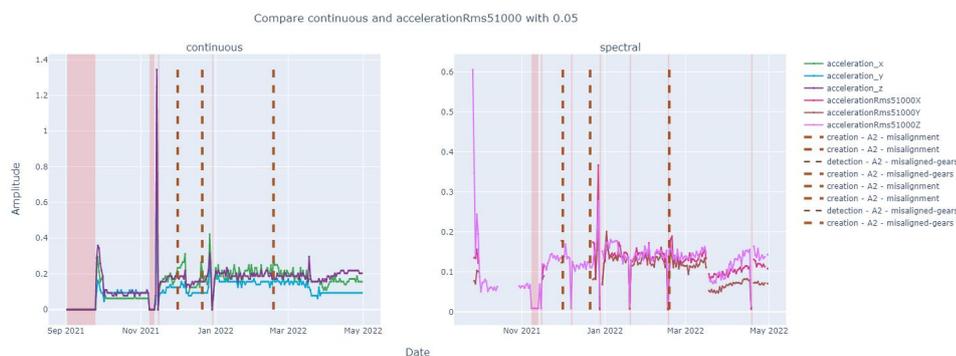
Fonte: Autoria Própria.

eixos X, Y e Z é maior que o valor limite. Caso seja, a data é salva para posterior remoção desses dias do *dataset* de dados. Como a coleta dos dados de telemetria e dos dados do monitoramento de forma de onda é realizada separadamente pelos sensores, foram escolhidas *features* diferentes para cada caso. Para os dados de telemetria, foi escolhido  $0.05 \text{ mm/s}^2$  como valor limite, logo se a aceleração RMS máxima entre os eixos X, Y, Z for menor que  $0.05 \text{ mm/s}^2$ , esse dia é considerado como máquina parada.

Enquanto para os dados do monitoramento de forma de onda, foi constatado e novamente validado, pela equipe Dynamox, que a aceleração RMS para a faixa de frequência de 5 Hz a 1000 Hz é uma boa característica para se detectar máquina parada, sendo adotado também  $0.05 \text{ mm/s}^2$  como valor limite. Como os sensores só capturam faixas de frequência mais baixas para o RMS nos dados de telemetria, foi escolhida uma faixa de frequência nos dados do monitoramento de forma de onda que se aproximasse dos dados de telemetria. A Figura 26 apresenta os intervalos de

máquina parada detectados nos dados de telemetria e nos dados do monitoramento de forma de onda para um mesmo *spot*. É possível verificar que nesse caso as datas não são necessariamente as mesmas.

Figura 26 – Exemplo de detecção de máquina parada para um *spot*.



Fonte: Autoria Própria.

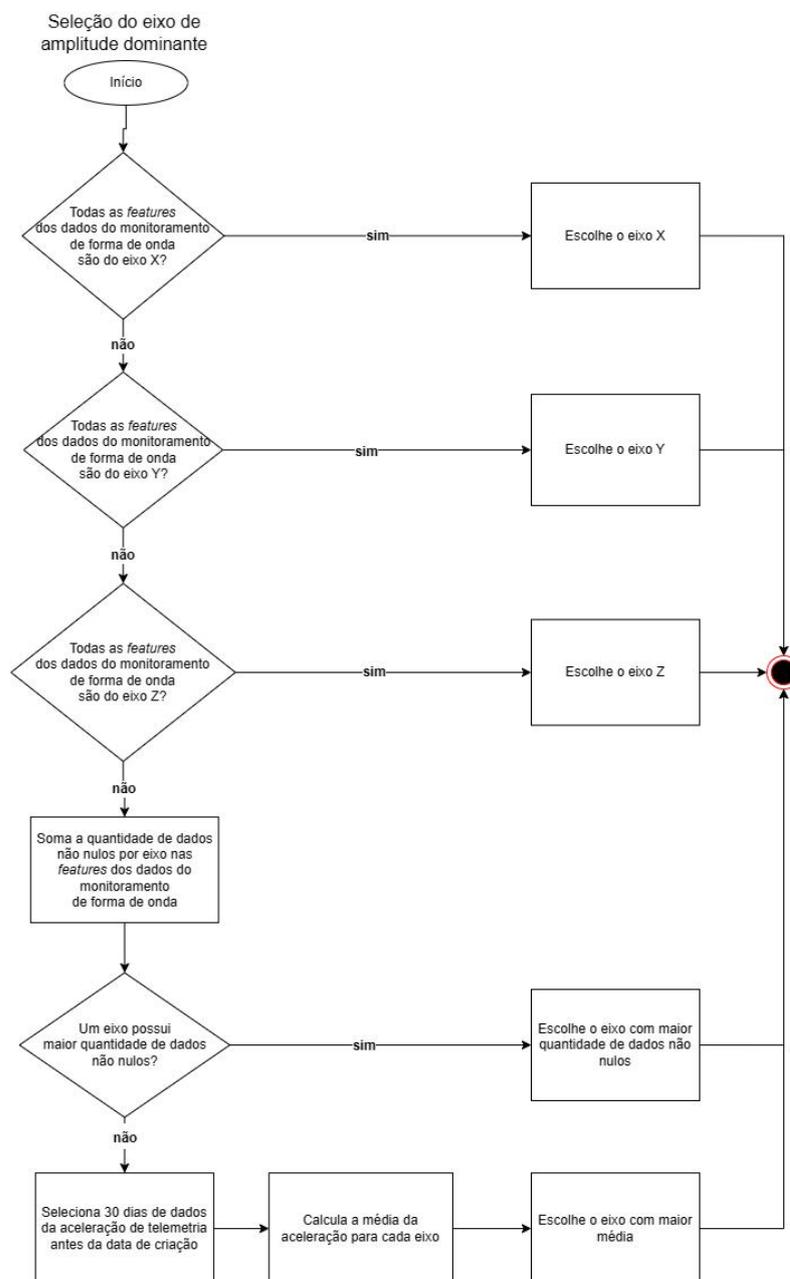
Após a remoção de máquina parada, de acordo com a Figura 25, é verificado novamente se os *datasets* estão vazios, pois essa remoção pode eliminar todos os dados presentes. Caso ainda existam dados em ambos os *datasets* e o *dataset* de forma de onda apresente todas as *features* escolhidas, é aplicado o procedimento de seleção do eixo de amplitude dominante do sinal em cada *feature* dos conjuntos.

O fluxograma da seleção do eixo de amplitude dominante é apresentado na Figura 27. Inicialmente, é verificado o eixo em relação aos sinais de vibração dos dados do monitoramento de forma de onda, pois, pelas análises realizadas, a diferença de eixo com amplitude dominante é mais evidente do que nos dados de telemetria. Ressalta-se que para essas análises foram consideradas as *features* escolhidas anteriormente e ao analisar os dados do sinal de vibração para o período de 14 dias antes da data de criação da falha associada à amostra. Se todas as *features* de forma de onda possuírem dados não nulos em apenas um eixo, esse eixo é selecionado como dominante.

Caso não seja possível definir o eixo dominante por esse método, é aplicado outro procedimento, que é baseado na soma do número de dados não nulos para o conjunto de *features* agrupadas por eixo X, Y e Z. Se nenhum eixo possuir uma maior quantidade de dados não nulos, são analisados os dados de telemetria. Para isso, são selecionados 30 dias de dados, antes da data de criação da falha associada da aceleração RMS e calculado a média da aceleração RMS para cada eixo, sendo escolhido como dominante o eixo de maior amplitude.

Para a finalização da *pipeline* de geração dos conjuntos, é verificado se o *dataset* de forma de onda possui pelo menos 1 dia de dados não nulos e se o *dataset*

Figura 27 – Fluxograma da seleção do eixo de amplitude dominante.

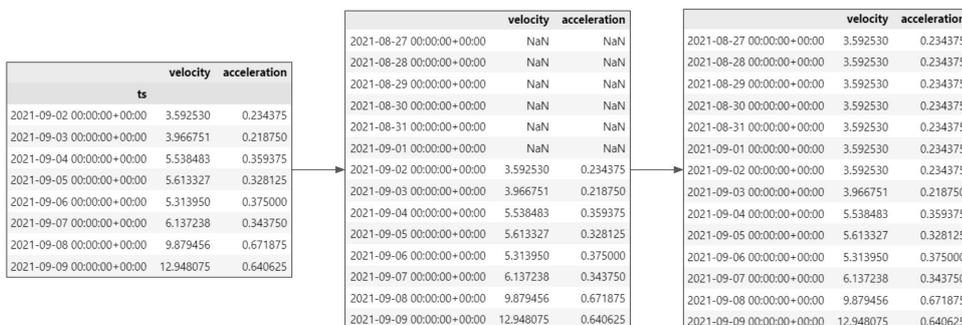


Fonte: Autoria Própria.

de telemetria possui pelo menos 7 dias de dados não nulos. Ao garantir que há pelo menos metade da quantidade total de dados desejados (14 dias) no *dataset* de telemetria, é possível interpolar os dados faltantes sem perder de forma excessiva o comportamento real do sinal. Se ambas as verificações forem positivas, essa amostra é salva em um *dataset* final que contém os sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda. Contudo, antes de salvar a amostra, é verificado se há dados faltantes no *dataset* de telemetria. Como mostrado na

Figura 28, é preciso primeiro acrescentar as datas dos dias faltantes no *dataset* para, posteriormente, realizar a interpolação linear.

Figura 28 – Exemplo de dados de telemetria faltantes no processamento do conjunto de treinamento e conjunto de teste.



Fonte: Autoria Própria.

Com isso, é gerado o *dataset* final contendo os sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda. A Figura 29 apresenta 10 amostras do conjunto de teste e as *features* finais que serão aplicadas no modelo. Ao todo são 59 *features*, sendo 28 delas dos dados de telemetria e 31 dos dados do monitoramento de forma de onda. A *feature* aceleração 1 é o valor da aceleração registrado 14 dias antes da data de criação da amostra; a *feature* aceleração 2 é o valor da aceleração registrado 13 dias antes da data de criação da amostra e esse raciocínio é aplicado para as demais *features* dos dados de telemetria.

Figura 29 – Exemplo de amostras do conjunto de teste.

spot_id	failure	creation_ts	acceleration_1	velocity_1	acceleration_2	velocity_2	acceleration_3	velocity_3	acceleration_4	velocity_4	acceleration_5	velocity_5	...	envelopeRms300795	envelopeRmsCarpets300795	velocityPkGlobal	
1	misalignment	2022-02-16	0.656250	8.307724	0.890625	13.097764	0.937500	15.043716	0.937500	17.738114	0.562500	8.681946	...	0.252589		0.138598	58.631229
2	misalignment	2022-02-16	0.546875	8.204439	0.734375	12.070898	0.703125	11.599380	0.890625	24.519014	0.562500	14.711409	...	0.422015		0.244761	44.145785
3	unbalance	2022-02-11	0.320312	3.966751	0.234375	3.218308	0.210938	4.116440	0.250000	4.340973	0.265625	5.014572	...	0.161072		0.101989	66.893283
4	unbalance	2022-02-11	1.734375	14.669496	1.093750	8.607102	1.218750	9.355546	1.218750	9.131012	1.312500	9.804612	...	0.685905		0.421027	65.849411
5	unbalance	2022-02-11	1.375000	16.503182	0.953125	10.184821	0.984375	10.187308	1.140625	11.036250	1.078125	12.356827	...	0.748981		0.458270	63.795425
6	misalignment	2022-01-25	0.203125	3.667374	0.328125	5.546773	0.359375	5.837860	0.265625	3.667374	0.328125	4.715195	...	0.282130		0.141783	22.436540
7	rotating-looseness	2021-11-24	1.656250	17.513581	1.609375	17.663269	1.828125	21.704866	1.625000	19.906011	1.765625	20.133134	...	0.774784		0.476939	81.144495
8	unbalance	2021-12-16	1.687500	46.786154	1.875000	45.654231	1.781250	39.994615	1.687500	41.503846	0.781250	22.261154	...	1.851232		1.088634	131.303364
9	rotating-looseness	2021-12-23	0.906250	9.879456	0.906250	9.280701	0.968750	10.328522	0.937500	10.927277	1.093750	10.927277	...	0.515320		0.309485	32.942180
10	rotating-looseness	2022-02-16	0.437500	5.659615	0.390625	5.093654	0.421875	5.942596	0.468750	5.376635	0.421875	5.376635	...	0.308636		0.158257	24.986945

Fonte: Autoria Própria.

## 6 MODELAGEM

Neste capítulo, serão descritas as estatísticas do conjunto final utilizado para o treinamento e teste do modelo, como quantidade de amostras para cada falha, métodos utilizados para seleção do eixo de amplitude dominante, além da explicação dos hiperparâmetros utilizados no algoritmo de busca aleatória.

### 6.1 AMOSTRAS DO CONJUNTO DE TREINAMENTO E DO CONJUNTO DE TESTE

Com a aplicação dos procedimentos explicados no capítulo 5, o conjunto de treinamento foi formado por 732 amostras, enquanto o conjunto de teste foi formado por 188 amostras. O Quadro 3 apresenta a quantidade de amostras para cada falha do conjunto de teste e do conjunto de treinamento, respectivamente. O fato de falhas como folga estrutural e engrenagens desalinhadas apresentarem poucas amostras não é um problema, visto que essas falhas são agrupadas na modelagem.

Quadro 3 – Quantidade de amostras para cada amostra no conjunto de teste e no conjunto de treinamento.

Falha	Quantidade de amostras	
	Teste	Treinamento
Normalidade	88	351
Desalinhamento	23	113
Desbalanceamento	19	62
Folga rotativa	17	125
Engrenagens desalinhadas	5	8
Folga estrutural	4	2
Perda de rigidez	16	36
bucha de fixação do rolamento	16	35

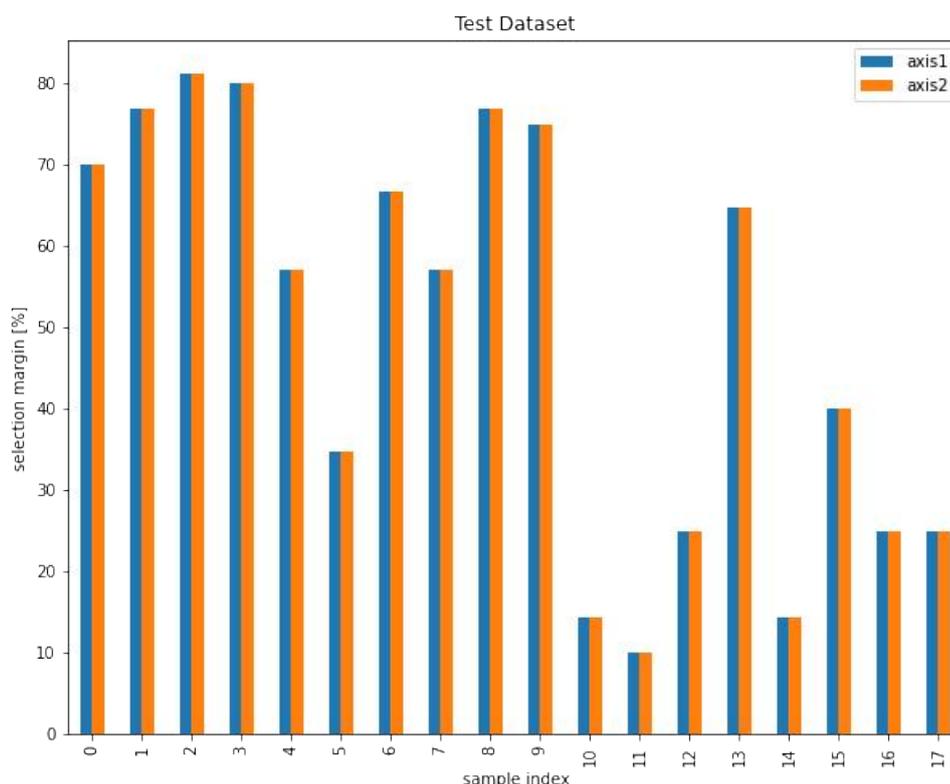
Fonte: Autoria Própria.

Ao todo havia 203 possíveis amostras para o conjunto de teste, sendo que 15 delas foram removidas, pois 8 não possuíam dados de forma de onda e 7 não possuíam mais de 7 dias de dados de telemetria não nulos após a seleção do eixo de amplitude dominante. Em relação ao método de escolha do eixo de amplitude dominante das amostras válidas para o conjunto de teste têm-se que: i) 3 amostras possuíam apenas *features* de forma de onda no eixo X; ii) duas amostras possuíam apenas *features* de forma de onda no eixo Y; iii) 23 possuíam apenas *features* de forma de onda no eixo Z; iv) 142 amostras utilizaram os dados de telemetria; v) e 18 possuem um eixo do espectral com mais dados não nulos

A Figura 30 apresenta um gráfico de barras indicando, em porcentagem, a diferença da quantidade de dados não nulos nas 18 amostras que utilizaram o método que

analisa a quantidade de dados não nulos no espectral para seleção do eixo dominante. Para, por exemplo, a amostra 0, há uma diferença de 65% de dados não nulos entre o eixo dominante e os demais eixos.

Figura 30 – Quão maior em porcentagem foi a quantidade de dados não nulos para as 18 amostras do conjunto de teste que utilizaram o método de seleção do eixo de amplitude dominante pelos dados de forma de onda não nulos. Ressalta-se que *axis1* e *axis2* são os eixos não dominantes.



Fonte: Autoria Própria.

Enquanto para o conjunto de treinamento, das 1316 possíveis amostras, 732 foram selecionadas, sendo que as outras 584 foram removidas, pois i) 170 possuíam os *datasets* dos dados de telemetria **E** do espectral vazios; ii) 160 amostras possuíam o *dataset* dos dados de vibração de telemetria vazios; iii) 80 possuíam o *dataset* espectral vazio; iv) 33 ficaram sem dados de forma de onda após a remoção de máquina parada; v) 14 ficaram sem dados de telemetria após a remoção de máquina parada; vi) e 127 não possuíam mais de 7 dias de dados de telemetria não nulos após a seleção do eixo de amplitude dominante.

Já na seleção do eixo dominante para o conjunto de amostras válidas para o conjunto de treino têm-se que: i) 13 amostras possuem todas as *features* do espectral para o eixo X; ii) 9 possuem todas as *features* de forma de onda para o eixo Y; iii) 126 possuem todas as *features* de forma de onda para o eixo Z; iv) 83 possuem um eixo

do espectral com mais dados não nulos; v) e para 501 amostras foram utilizados os dados de telemetria para definir o eixo de amplitude dominante.

Uma questão levantada ao realizar a escolha do eixo de amplitude dominante foi que estão sendo utilizados 5 métodos diferentes para essa escolha, então o eixo dominante pode variar de acordo com o método escolhido. Idealmente, um *spot* deveria ter apenas um eixo dominante, contudo, como mostrado na Figura 31, não é isso que ocorre, porque é possível verificar a mudança do eixo de amplitude dominante no conjunto de treinamento, sendo apresentado o pior caso. O mesmo *spot* sofre uma mudança de eixo dominante 4 vezes, entretanto, nota-se que as primeiras duas mudanças acontecem quando o *spot* encontra-se em normalidade.

Foi assumida a hipótese que em normalidade as mudanças de eixo dominante não são problemáticas, pois os sinais de vibração estão no limite aceitável de operação da máquina, enquanto as mudanças de eixo dominante quando há falha na máquina são consideradas uma limitação que a metodologia adotada neste trabalho possui.

Figura 31 – Mudança de eixo dominante para um mesmo *spot*.

<b>spot_id</b>	<b>failure</b>	<b>creation_ts</b>	<b>dominant_axis</b>	<b>method</b>
1	none	2021-12-23	Z	cont
1	none	2022-03-17	X	cont
1	none	2022-04-14	X	cont
1	none	2022-05-07	Y	non_null
1	unbalance	2022-05-20	X	non_null
1	rotating-looseness	2022-06-14	Y	cont

Fonte: Autoria Própria.

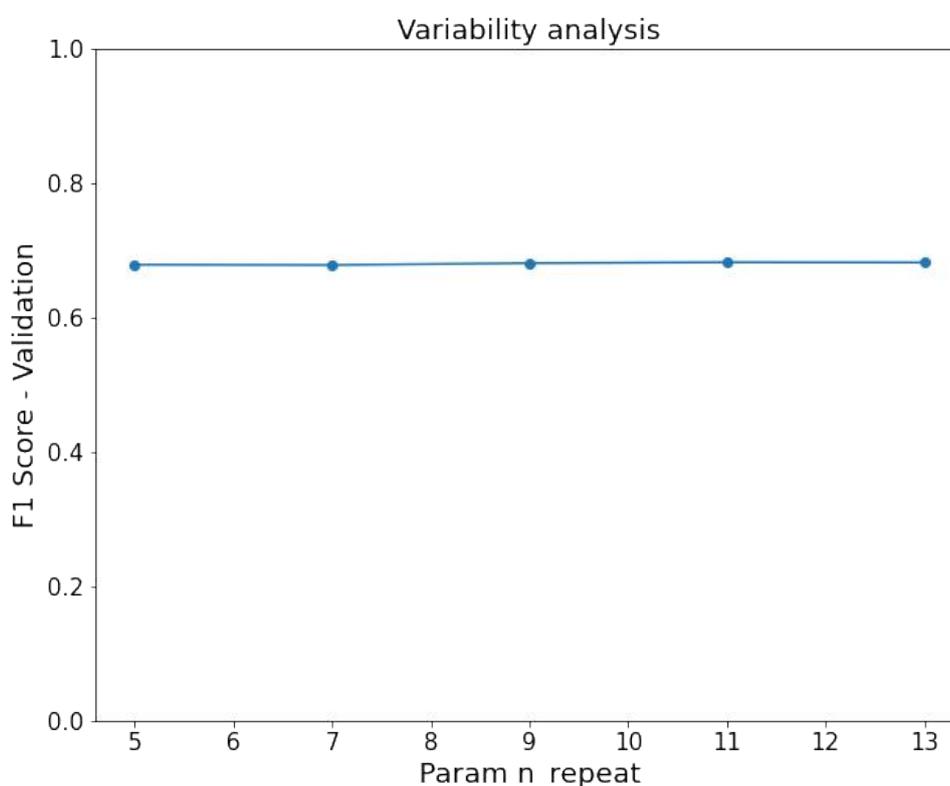
Por fim, o fato de ter sido utilizado um conjunto de teste com amostras selecionadas manualmente e validadas pela equipe Dynamox, garante que os resultados apresentados e os desempenhos obtidos para os modelos sejam mais confiáveis do que simplesmente realizar um *split* aleatório de todo o conjunto de dados processados. A desvantagem dessa seleção é que um viés é introduzido no modelo, pois a escolha das amostras não foi puramente aleatória. Então é esperado que esse processamento adotado para a seleção das amostras atrapalhe o aprendizado do modelo para a detecção de falhas.

Idealmente, o conjunto de treinamento deveria ter a mesma confiabilidade que o conjunto de teste possui, ou seja, garantir que realmente existe uma falha nos dias que antecedem a data de criação do relatório de falha. Assim, também teria uma maior confiabilidade na análise da importância das *features* para o aprendizado do modelo.

## 6.2 BUSCA DOS HIPERPARÂMETROS

Como dito no capítulo 3, será utilizada a busca aleatória com estratégia de validação cruzada. Na escolha dos atributos da validação cruzada foi mantido o valor padrão de 5 *folds*, e definido como 9 o número de repetições que a validação cruzada será executada. Como mostrado na Figura 32, ao variar a quantidade de repetições de execução, não há variabilidade no resultado do modelo.

Figura 32 – Desempenho do modelo de detecção de falhas mecânicas com os sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda ao variar o parâmetro *n\_repeat* do método de criação do objeto de validação cruzada.



Fonte: Autoria Própria.

Para a definição dos hiperparâmetros que seriam utilizados no algoritmo de busca aleatória foram realizados experimentos nos quais os parâmetros eram variados um a um, com o intuito de verificar quais eram significativos para o aumento do desempenho do modelo. Os valores iniciais aplicados nos experimentos foram baseados em trabalhos anteriores realizados pelo professor orientador e que resultaram um bom desempenho no modelo de *random forest*. A partir deles foram analisados quais hiperparâmetros eram significativos com o conjunto de dados formado pelos dados de telemetria e dados do monitoramento de forma de onda e, posteriormente, calibrados os valores mínimo e máximo das distribuições.

Assim, foi definido que: i) o número de árvores na floresta seria de 400; ii) o critério para o cálculo de impurezas seria o Gini, pois, como visto no capítulo 2, é mais rápido que a entropia e apresenta bons resultados; iii) para o peso das classes foi adotada a opção *balanced subsample* que considera pesos para cada árvore e estima pesos nas classes, sendo esse peso inversamente proporcional às frequências das classes <sup>3</sup>; iv) e o parâmetro *bootstrap* foi mantido como verdadeiro, assim é realizada a amostragem com reposição.

Os hiperparâmetros *min\_samples\_split* e *min\_samples\_leaf* foram definidos como distribuições uniformes a partir da função *randint* do módulo Scipy do Python, que “gera uma variável aleatória discreta uniforme” <sup>4</sup>. Já o hiperparâmetro *min\_weight\_fraction\_leaf* foi definido como uma distribuição uniforme contínua com a função *uniform* do módulo Scipy do Python <sup>5</sup>. O Quadro 4 apresenta os valores finais utilizados nos hiperparâmetros do modelo.

Quadro 4 – Hiperparâmetros utilizados na busca aleatória com estratégia de validação cruzada.

Hiperparâmetro	Valor
<i>n_estimators</i>	400
<i>criterion</i>	Gini
<i>class_weight</i>	<i>balanced_subsample</i>
<i>bootstrap</i>	<i>True</i>
<i>min_impurity_decrease</i>	0
<i>min_weight_fraction_leaf</i>	<i>inuniform(0,0.03)</i>
<i>min_samples_split</i>	<i>randint(4,60)</i>
<i>min_samples_leaf</i>	<i>randint(4,40)</i>

Fonte: Autoria Própria.

### 6.3 MODELOS DE *RANDOM FOREST*

Com os hiperparâmetros e as suas faixas definidas, foram treinados os modelos de *random forest* para os três conjuntos de dados definidos no capítulo 3. O Quadro 5 apresenta como foram definidas as *targets* de cada modelo de classificação utilizados no projeto.

<sup>3</sup> Explicação dos parâmetros disponível em [https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html#sklearn.ensemble.RandomForestClassifier.predict\\_proba](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html#sklearn.ensemble.RandomForestClassifier.predict_proba).

<sup>4</sup> Disponível em <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.randint.html>.

<sup>5</sup> Disponível em <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.uniform.html?highlight=uniform#scipy.stats.uniform>.

Quadro 5 – Definição das *targets* de cada modelo de classificação utilizado.

Target associada a cada modelo de predição				
Tipo de Classificação	Falhas mecânicas específicas			Todas as falhas juntas
Falha	Desbalanceamento	Folga Rotativa	Desalinhamento	Falhas mecânicas
Normalidade	0	0	0	0
Folga rotativa	0	1	0	1
Desbalanceamento	1	0	0	1
Desalinhamento	0	0	1	1
Perda de rigidez	0	0	0	1
bucha de fixação do rolamento	0	0	0	1
Engrenagens desalinhas	0	0	0	1
Folga estrutural	0	0	0	1

Fonte: Autoria Própria.

## 7 RESULTADOS

Neste capítulo, será apresentada a avaliação de desempenho obtida pelos modelos nos experimentos. O objetivo é estimar se os resultados obtidos são satisfatórios.

### 7.1 DESEMPENHO DOS MODELOS

O Quadro 6 apresenta o desempenho, com a métrica acurácia balanceada, para o conjunto de validação utilizado no treinamento do modelo, enquanto o Quadro 7 apresenta o desempenho para o conjunto de teste. O modelo que aborda todas as falhas mecânicas apresentou um resultado melhor do que aqueles que abordam as falhas separadamente para os diferentes conjuntos de *features* adotados como mostrado no Quadro 7. Contudo, para o conjunto de *features* dos dados do monitoramento de forma de onda, a detecção de folga rotativa teve um melhor desempenho que o modelo de detecção de falhas mecânicas.

Quadro 6 – Métrica acurácia balanceada para os modelos de *random forest* no conjunto de validação.

Acurácia balanceada para o conjunto de validação			
Falhas \ <i>Features</i>	Dados de telemetria e Dados do monitoramento de forma de onda	Dados de telemetria	Dados do monitoramento de forma de onda
Falhas mecânicas	0,6771	0,6139	0,6548
Desalinhamento	0,6261	0,5594	0,6204
Folga rotativa	0,6779	0,6709	0,6433
Desbalanceamento	0,6650	0,6756	0,6431

Fonte: Autoria Própria.

Quadro 7 – Métrica acurácia balanceada para os modelos de *random forest* no conjunto de teste.

Acurácia balanceada para o conjunto de teste			
Problema \ <i>Features</i>	Dados de telemetria e Dados do monitoramento de forma de onda	Dados de telemetria	Dados do monitoramento de forma de onda
Falhas mecânicas	0,686	0,633	0,73
Desalinhamento	0,572	0,542	0,621
Folga rotativa	0,704	0,628	0,791
Desbalanceamento	0,553	0,556	0,602

Fonte: Autoria Própria.

Uma possível explicação para os modelos de detecção de falhas específicas apresentarem em sua maioria um resultado inferior é de que as falhas apresentam características parecidas nos dados de vibração e estão sendo marcadas como zero

no conjunto de dados, com exceção da *target* alvo. Então, o modelo não consegue identificar as falhas separadamente de uma maneira eficiente.

Ainda em relação ao problema de predição das falhas específicas, os modelos apresentaram um melhor desempenho no conjunto de validação do que no conjunto de teste para as falhas de desalinhamento e desbalanceamento. Uma possível explicação para esse resultado é de que há um grau considerável de *overfitting* no modelo, no qual este ficou enviesado pelo ruído presente no conjunto de treinamento. Então, os desempenhos parecem ser bons no conjunto de validação, contudo, ao aplicar no conjunto de teste, há uma queda de desempenho, pois esse conjunto possui mais garantia do não uso de dados ruidosos.

Essa diferença de resultados com o conjunto de validação e de teste também pode ser justificada pela discrepância entre a quantidade de amostras do conjunto de treino e do conjunto de teste como apresentado no capítulo 6.1, pois houve a introdução de viés no modelo ao selecionar as amostras de teste manualmente pelo *dashboard*.

Enquanto o desempenho do modelo de predição de falhas mecânicas é consistente tanto no conjunto de validação quanto no conjunto de teste, os valores entre os conjuntos não são discrepantes e a acurácia balanceada foi maior para o conjunto de teste. Isso é um indicativo de que o modelo está generalizando bem seu desempenho para dados que não tenham sido vistos durante o treinamento e pode ser confiável para uso em situações reais.

Outra métrica analisada foi a sensibilidade, que indica a capacidade do modelo de identificar todas as ocorrências de cada classe. Pelo Quadro 8, nota-se que a sensibilidade do modelo para o problema de predição de falhas mecânicas com as *features* dos dados do monitoramento de forma de onda está acima de 50% para todas as falhas. O melhor desempenho foi para a falha de folga estrutural, no qual a taxa de acerto foi de 95%, ou seja, das vezes em que o modelo deve prever corretamente a folga estrutural, ele acerta 95% das vezes.

Quadro 8 – Métrica de sensibilidade para os modelos de *random forest* no conjunto de teste.

Problema	Features / Falhas	Recall para cada falha							
		Bucha de fixação do rolamento	engrenagens desalinhadas	desalinhamento	normalidade	folga rotativa	perda de rigidez	Folga estrutural	desbalanceamento
Falhas mecânicas	Dados de telemetria e dados do monitoramento de forma de onda	0,5625	0,6000	0,6957	0,6409	1,0000	0,5625	0,7500	0,8526
	Dados de telemetria	0,4375	0,6000	0,6609	0,6364	0,8706	0,3500	0,7000	0,7684
	Dados do monitoramento de forma de onda	0,5500	0,6000	0,6609	0,7227	1,0000	0,5625	0,9500	0,8947

Fonte: Autoria Própria.

Por último, nos resultados obtidos em cada grupo de *features* adotado no conjunto de dados, têm-se que: o desempenho foi melhor ao utilizar apenas os dados de monitoramento de forma de onda. Indicando que, mesmo sendo possível identificar alguma informação usando os dados de telemetria, os dados temporais atrapalham o

modelo devido ao seu ruído e é possível detectar melhor a diferença entre as falhas pela forma de onda. Já para o problema de classificação de falhas mecânicas, a acurácia balanceada com os dados de monitoramento de forma de onda foi respectivamente 10% e 5% maior do que com os dados de telemetria e do que os dados de telemetria e forma de onda em conjunto.

## 7.2 IMPORTÂNCIA DAS *FEATURES*

As *features* associadas aos nós da árvore de decisão são definidas a partir do grau de importância delas, sendo calculado de acordo com a documentação do Scikit Learn (PEDREGOSA *et al.*, 2011) como a média e o desvio padrão do acúmulo da diminuição da impureza dentro de cada árvore. Para o modelo de classificação de falhas mecânicas, a Figura 33 apresenta a diminuição média da impureza para as 11 *features* mais importantes ao utilizar os sinais de vibração dos dados de telemetria e dos dados do monitoramento de forma de onda separados e/ou em conjunto.

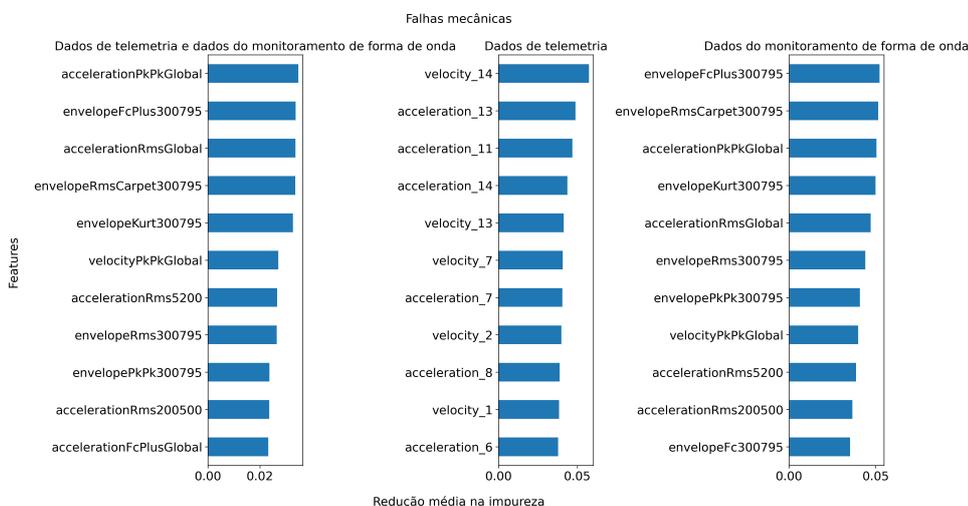
A partir da Figura 33 é possível notar que, para o conjunto formado pelos dados de telemetria e forma de onda, as *features* de forma de onda possuem mais relevância, sendo que a *feature* de aceleração foi a que mais se manifestou, seguida pela *feature* envelope e pela *feature* velocidade. No conjunto de treinamento são utilizadas 8 *features* de deslocamento, contudo nenhum delas apareceu entre as 11 mais importantes.

Ainda de acordo com a Figura 33, ao analisar o gráfico de redução média na impureza para o conjunto que resultou no melhor desempenho de detecção de falhas mecânicas, ou seja, o conjunto formado pelos dados do monitoramento de forma de onda, têm-se que as *features* mais relevantes são de envelope, seguida pelas *features* da aceleração. Ressalta-se que o conjunto de dados era formado por 6 *features* de envelope, sendo que todas estão entre as 11 mais importantes.

Para o modelo de detecção de folga rotativa, as *features* com maior redução média de impurezas, como mostrado na Figura 34, foram a aceleração dos dados de telemetria e as *features* de envelope. Essas importâncias foram considerando o conjunto formado pelos dados de telemetria e pelos dados de monitoramento de forma de onda. Já considerando apenas o conjunto formado pelos dados de onda, tem-se que as 6 *features* de envelope presentes no conjunto de dados também se encontram entre as 11 *features* com maior redução média na impureza, como mostrado na Figura 34. Esse resultado é semelhante ao do modelo de detecção de falhas mecânicas.

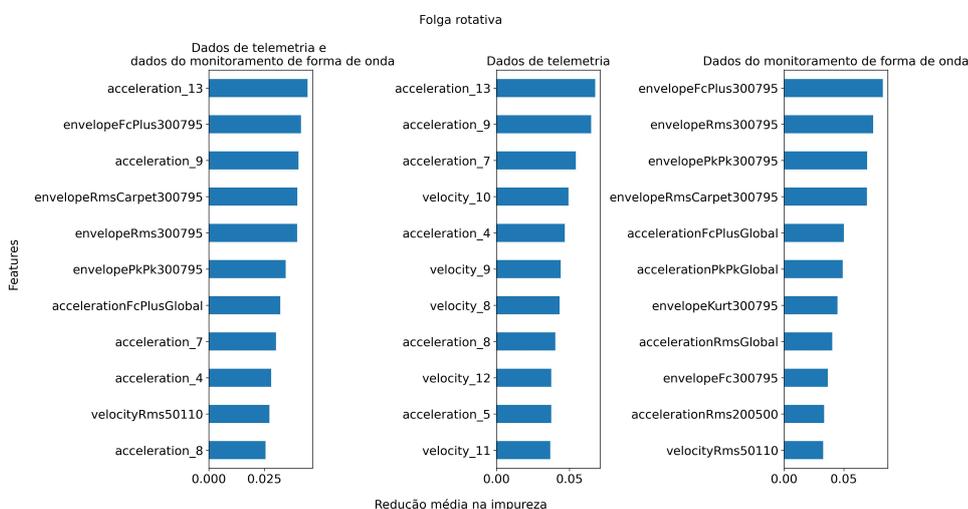
Para o modelo de detecção de desalinhamento, as *features* com maior redução média de impurezas, como mostrado na Figura 35, foram as *features* de aceleração dos dados de forma de onda, considerando o conjunto formado pelos dados de telemetria e de forma de onda. Enquanto que em relação ao conjunto formado apenas pelos dados de forma de onda, têm-se que as *features* de aceleração continuam sendo as

Figura 33 – Diminuição média da impureza das *features* consideradas mais relevantes pelo modelo de *random forest* para detecção de falhas mecânicas.



Fonte: Autoria Própria.

Figura 34 – Diminuição média da impureza das *features* consideradas mais relevantes pelo modelo de *random forest* para detecção da falha de folga rotativa.

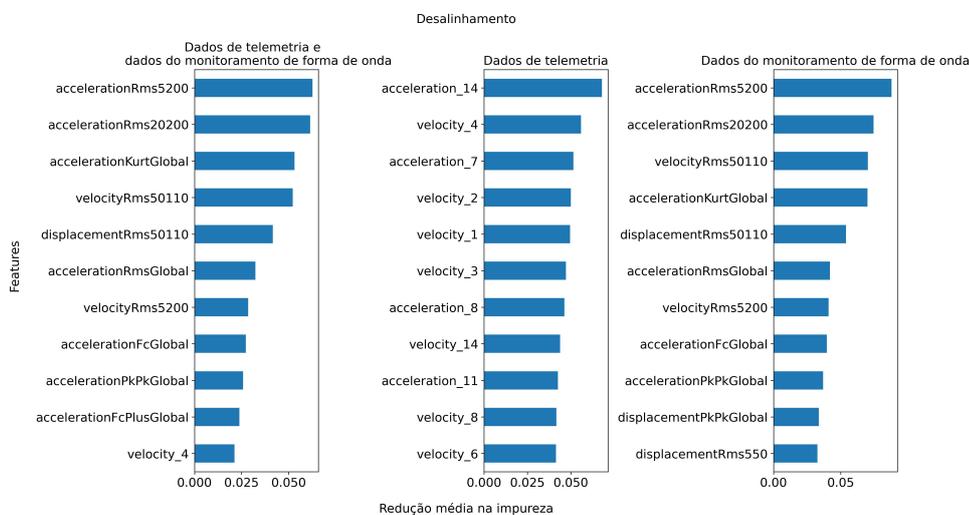


Fonte: Autoria Própria.

mais relevantes, seguida pelas *features* de deslocamento. Esse resultado é o oposto do obtido para os modelos de detecção de falhas mecânicas e folga rotativa, pois nesse modelo não apareceu nenhuma *feature* de envelope entre as 11 *features* mais importantes, além de que 3 *features* de deslocamento foram consideradas relevantes para a detecção de desalinhamento.

Por último, para o modelo de detecção de desbalanceamento, tem-se que, ao considerar o conjunto formado pelos dados de telemetria e forma de onda, as *features*

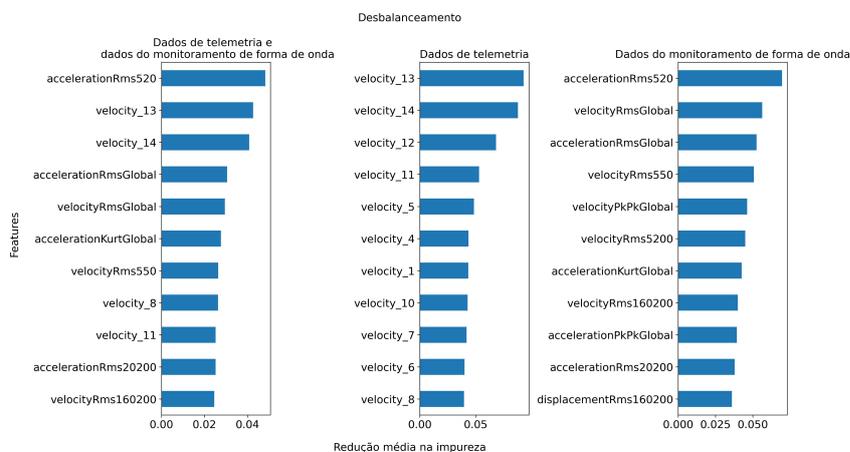
Figura 35 – Diminuição média da impureza das *features* consideradas mais relevantes pelo modelo de *random forest* para detecção da falha de desalinhamento.



Fonte: Autoria Própria.

consideradas mais importantes são as *features* de aceleração dos dados de forma de onda e as *features* de velocidade dos dados de telemetria, como mostrado na Figura 36. Além disso, ao analisar a relevância das *features* para o conjunto formado apenas pelos dados de forma de onda, tem-se que as *features* de aceleração continuam sendo as mais relevantes, seguida pelas *features* de velocidade.

Figura 36 – Diminuição média da impureza das *features* consideradas mais relevantes pelo modelo de *random forest* para detecção da falha de desbalanceamento.



Fonte: Autoria Própria.

## 8 CONSIDERAÇÕES FINAIS

A manutenção preditiva de máquinas rotativas, a partir do monitoramento dos sinais de vibração, é uma técnica eficaz que não requer o desligamento da máquina para investigação e detecção de falhas. Assim, é possível identificar precocemente as falhas e executar ações que evitem a quebra do maquinário, gerando redução de custos e aumento na eficiência das manutenções.

A detecção de falhas mecânicas com a técnica de monitoramento dos sinais de vibração é executável, pois esse tipo de falha apresenta características e padrões específicos nos sinais de vibração. Neste contexto, a idealização do projeto teve como ponto principal o desenvolvimento de uma *pipeline* para a aplicação do modelo de *random forest* no conjunto de dados formado pelos sinais de vibração reais de máquinas rotativas e relatórios de falhas cadastrados por analistas das empresas parceiras e da própria Dynamox.

Com isso, foi investigado como poderia ser agregado valor ao sistema de monitoramento já existente na Dynamox com um modelo de aprendizado de máquina de *random forest* para detecção de falhas, pois a empresa possui uma grande quantidade de dados contendo os sinais de vibração e os relatórios de falhas. Como procedimentos metodológicos adotados no trabalho, têm-se: uma revisão da literatura para contextualizar o tema e os conteúdos abordados no trabalho dentro da área de engenharia; implementação de uma abordagem quantitativa para validação do desempenho dos modelos e uso da metodologia CRISP-DM como forma de apoio na realização do projeto.

Os dados sugerem que os relatórios de falhas possuem informações subjetivas e que dependem da interpretação de cada analista, além das informações apresentarem inconsistências. Uma possibilidade para contornar essa situação seria realizar verificações nos dados antes de estes serem salvos na plataforma da Dynamox, evitando que situações como *spot* com situação A0, que indica normalidade, seja associado a uma falha.

Em relação à escolha do eixo de amplitude dominante, o procedimento adotado no trabalho não garante que sempre seja o mesmo eixo dominante para um *spot*. Verificou-se que o método mais usado se baseia nos sinais dos dados de telemetria. Contudo, não é possível adotá-lo unicamente, pois, por exemplo, pode ocorrer de ser selecionado, pelos dados de telemetria, o eixo X como dominante e algumas *features* de forma de onda não apresentarem dados não nulos para esse eixo. Então, a definição de mais de um método para a escolha do eixo de amplitude dominante foi devido às limitações que os dados utilizados no trabalho possuem.

Os resultados dos experimentos sugerem que o modelo de detecção de falhas mecânicas é melhor que os modelos de detecção das falhas mecânicas específicas:

desbalanceamento, desalinhamento e folga rotativa. Isso foi atribuído às semelhanças que as falhas possuem nos sinais de vibração. Além disso, com apenas 1 dia de dados do monitoramento de forma de onda foi obtido um melhor desempenho do que utilizando apenas os dados de telemetria ou em conjunto com os dados do monitoramento de forma de onda. Isso indica que os sinais de velocidade RMS e aceleração RMS dos dados de telemetria não são considerados como *features* importantes na detecção das falhas se comparados com as *features* dos dados do monitoramento de forma de onda.

Na análise da importância das *features* constatou-se que as dos dados de monitoramento de forma de onda apresentam uma diminuição média das impurezas maior que as dos dados de telemetria no modelo de *random forest* ao utilizar o conjunto contendo os dados de telemetria e os dados do monitoramento de forma de onda. Investigando as *features* de forma de onda, verificou-se que: para o problema de detecção de falhas mecânicas e detecção de folga rotativa, as *features* do sinal de envelope dos dados do monitoramento de forma de onda utilizadas são consideradas mais relevantes para a predição de falhas no modelo de *random forest*; enquanto que para os problemas de detecção de desalinhamento e desbalanceamento, as *features* do sinal de aceleração dos dados do monitoramento de forma de onda são consideradas mais relevantes para a predição de falhas.

Também foi verificado que não há um padrão exato na importância das *features* dos dados de telemetria em relação à distância da data de criação. Assim, os valores de aceleração RMS e velocidade RMS um dia antes da data de criação não são necessariamente mais importantes que os valores dois ou três dias antes da data de criação. Esses resultados foram importantes para responder aos problemas de pesquisa e validar as hipóteses assumidas neste trabalho.

Por fim, o projeto atendeu às especificações estabelecidas e foi possível responder aos problemas de pesquisa do trabalho, mas ainda há uma série de aprimoramentos e experimentos que podem ser realizados, como definir um valor de *threshold* mais adaptável ao comportamento dos dados do *spot* da máquina, em vez de definir um valor fixo para todos os *spots*. Também seria interessante testar outros modelos de classificação e comparar os desempenhos para o conjunto de treinamento e conjunto de teste criados neste trabalho. Um possível modelo seria o XGBoost, que também é baseado em árvore de decisão.

Outro teste futuro seria alterar a quantidade de dias selecionados para os sinais de vibração dos dados do monitoramento de forma de onda, pois foi considerado 1 dia de dados para as *features* de forma de onda e foi constatado que foram dados mais importantes no aprendizado do modelo. Então, aumentar a quantidade de dias selecionados poderia aumentar o desempenho dos modelos.

## REFERÊNCIAS

BOSSIO, José M.; BOSSIO, Guillermo R.; ANGELO, Cristian Hernán de. Angular misalignment in induction motors with flexible coupling. **2009 35th Annual Conference of IEEE Industrial Electronics**, p. 1033–1038, 2009.

DYNAMOX. **DynaLoggers**. [S.l.: s.n.], 2022a. Acessado em 18 de Novembro de 2022. Disponível em: <https://dynamox.net/dynaloggers>.

DYNAMOX. **DynaPredict Solution**. [S.l.: s.n.], 2022b. Acessado em 18 de Novembro de 2022. Disponível em: <https://dynamox.net/en/dynapredict>.

DYNAMOX. **Sobre Nós**. [S.l.: s.n.], 2022c. Acessado em 18 de Novembro de 2022. Disponível em: <https://dynamox.net/about-us>.

FILHO, Julio De Mesquista. **Ferramentas de diagnóstico de máquinas**. [S.l.: s.n.]. Acessado em 19 de Novembro de 2022. Disponível em:  
<https://view.officeapps.live.com/op/view.aspx?src=https%3A%2F%2Ffacervodigital.unesp.br%2Fbitstream%2F123456789%2F46460%2F6%5C%2FAula%252006%2520-%5C%2520Ferramentas%2520de%2520processamento%2520-%5C%2520parte%25201.ppt&wdOrigin=BROWSELINK>.

FORTMANN-ROE, Scott. **Understanding the Bias-Variance Tradeoff**. [S.l.: s.n.], 2017. Acessado em 19 de Novembro de 2022. Disponível em:  
<http://scott.fortmann-roe.com/docs/BiasVariance.html>.

GÉRON, Aurélien. **Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems**. 2. ed. CA 95472: O Reilly, 2019.

GIRDHAR, Paresh; SCHEFFER, C. 1 - Predictive maintenance techniques: Part 1 predictive maintenance basics. In: GIRDHAR, Paresh; SCHEFFER, C. (Ed.). **Practical Machinery Vibration Analysis and Predictive Maintenance**. Oxford: Newnes, 2004a. P. 1–10. ISBN 978-0-7506-6275-8. DOI:  
<https://doi.org/10.1016/B978-075066275-8/50001-1>. Disponível em:  
<https://www.sciencedirect.com/science/article/pii/B9780750662758500011>.

GIRDHAR, Paresh; SCHEFFER, C. 2 - Predictive maintenance techniques: Part 2 vibration basics. *In*: GIRDHAR, Paresh; SCHEFFER, C. (Ed.). **Practical Machinery Vibration Analysis and Predictive Maintenance**. Oxford: Newnes, 2004b. P. 11–28. ISBN 978-0-7506-6275-8. DOI:

<https://doi.org/10.1016/B978-075066275-8/50002-3>. Disponível em:

<https://www.sciencedirect.com/science/article/pii/B9780750662758500023>.

GIRDHAR, Paresh; SCHEFFER, C. 5 - Machinery fault diagnosis using vibration analysis. *In*: GIRDHAR, Paresh; SCHEFFER, C. (Ed.). **Practical Machinery Vibration Analysis and Predictive Maintenance**. Oxford: Newnes, 2004c. P. 89–133. ISBN 978-0-7506-6275-8. DOI: <https://doi.org/10.1016/B978-075066275-8/50005-9>. Disponível em:

<https://www.sciencedirect.com/science/article/pii/B9780750662758500059>.

INC., Perpetual Industries. **Folga Mecânica**. [S.l.: s.n.], 2022. Acessado em 18 de Novembro de 2022. Disponível em:

<https://xyobalancer.com/mechanical-looseness/>.

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

QIAN, Weiwei; LI, Shunming; JIANG, Xingxing. Deep transfer network for rotating machine fault analysis. **Pattern Recognition**, v. 96, p. 106993, 2019. ISSN 0031-3203. DOI: <https://doi.org/10.1016/j.patcog.2019.106993>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320319302961>.

SÁ SÓ MARTINS, Dionísio Henrique Carvalho de *et al.* Diagnostic and severity analysis of combined failures composed by imbalance and misalignment in rotating machines. **The International Journal of Advanced Manufacturing Technology**, v. 114, n. 9, p. 3077–3092, jun. 2021.

SCHRÖER, Christoph; KRUSE, Felix; GÓMEZ, Jorge Marx. A Systematic Literature Review on Applying CRISP-DM Process Model. **Procedia Computer Science**, v. 181, p. 526–534, 2021. CENTERIS 2020 - International Conference on ENTERprise Information Systems / ProjMAN 2020 - International Conference on Project MANagement / HCist 2020 - International Conference on Health and Social Care Information Systems and Technologies 2020, CENTERIS/ProjMAN/HCist 2020. ISSN 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2021.01.199>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050921002416>.

SP, Sujesh; MARULIAIAH, Lokesha; KUMAR, Kiran; K R, Srinivas. Vibration based Fault Diagnosis Techniques for Rotating Mechanical Components: Review Paper. **IOP Conference Series: Materials Science and Engineering**, v. 376, p. 012109, jun. 2018. DOI: 10.1088/1757-899X/376/1/012109.

ZHANG, Weiting; YANG, Dong; WANG, Hongchao. Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey. **IEEE Systems Journal**, v. 13, n. 3, p. 2213–2227, 2019. DOI: 10.1109/JSYST.2019.2905565.