



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

Caio Henrique Roman Rodriguez

**Aplicação de autoencoders para redução de dimensionalidade em dados de  
vibração de compressores herméticos para refrigeração doméstica**

Florianópolis  
2023

Caio Henrique Roman Rodriguez

**Aplicação de autoencoders para redução de dimensionalidade em dados de vibração de compressores herméticos para refrigeração doméstica**

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. Rodolfo César Costa Flesch, Dr.  
Supervisor: Prof. Gabriel Thaler, M. Eng.

Florianópolis  
2023

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Roman Rodriguez, Caio Henrique

Aplicação de autoencoders para redução de dimensionalidade em dados de vibração de compressores herméticos para refrigeração doméstica. / Caio Henrique Roman Rodriguez ; orientador, Rodolfo César Costa Flesch, coorientador, Gabriel Thaler, 2023.

80 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia de Controle e Automação, Florianópolis, 2023.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Aprendizado de máquina. 3. Autoencoder. 4. Redução de dimensionalidade de dados. 5. Amaciamento em compressores. I. Costa Flesch, Rodolfo César. II. Thaler, Gabriel . III. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. IV. Título.

Caio Henrique Roman Rodriguez

**Aplicação de autoencoders para redução de dimensionalidade em dados de  
vibração de compressores herméticos para refrigeração doméstica**

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, 10 de Julho de 2023.

Prof. Hector Bessa Silveira, Dr.  
Coordenador do Curso

**Banca Examinadora:**

Prof. Rodolfo César Costa Flesch, Dr.  
Orientador  
UFSC/CTC/DAS

Prof. Gabriel Thaler, M. Eng.  
Supervisor  
UFSC/CTC/DAS/LIAE

João Paulo Zomer Machado, M. Eng.  
Avaliador  
UFSC/CTC/DAS/LIAE

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

## **AGRADECIMENTOS**

Primeiramente, gostaria de expressar minha profunda gratidão aos meus pais, Lilian e Carlos, que nunca deixaram de me apoiar, mesmo quando eu não acreditava em mim mesmo. A origem dos meus valores e da excelente educação que recebi, que inclusive me proporcionou a oportunidade de estudar em uma instituição de excelência, definitivamente se deve a eles. Vocês foram fundamentais para todas as minhas conquistas, inclusive as que ainda estão por vir.

Sou grato também aos meus avós Pepe, Maria, José e Consuelo, por todo o carinho e pela assistência que deram aos meus pais no meu crescimento. Quero dedicar esse trabalho a todos os meus familiares, em especial aos tios Cesar e Goreti que serviram de inspiração para a minha escolha de curso de graduação. Sem esquecer do companheirismo das minhas irmãs, primos e primas, que foram essenciais para a finalização dessa etapa tão importante da minha vida.

Agradeço a minha namorada Mônica, que foi muito importante nessa reta final de graduação, colaborando muito para que esse processo tenha sido mais leve. É importante mencionar, mesmo que de forma geral, por serem muitos, todos os amigos que a UFSC me proporcionou ao longo desses anos, cada um de vocês teve importância significativa no meu desenvolvimento pessoal e profissional.

Ao meu orientador, Professor Rodolfo Flesch, que trouxe oportunidades desde a iniciação científica, guiando o trabalho com excelência e profissionalismo. Me apresentando ao Laboratório de Instrumentação e Automação de Ensaios, que teve uma parcela importante na minha vivência acadêmica. Também sou grato ao meu supervisor no laboratório, Professor Gabriel Thaler, que me concedeu essa oportunidade e me orientou com paciência para que eu alcançasse os objetivos do projeto.

Resumindo, a todos os meus familiares, amigos, colegas do LIAE, professores e todos que estiveram ao meu lado nessa jornada, quero expressar minha gratidão sincera. Cada um de vocês faz parte dessa conquista. Muito obrigado a todos.

## DECLARAÇÃO DE PUBLICIDADE

Florianópolis, 10 de julho de 2023.

Na condição de representante do Laboratório de Instrumentação e Automação de Ensaios na qual o presente trabalho foi realizado, declaro não haver ressalvas quanto ao aspecto de sigilo ou propriedade intelectual sobre as informações contidas neste documento, que impeçam a sua publicação por parte da Universidade Federal de Santa Catarina (UFSC) para acesso pelo público em geral, incluindo a sua disponibilização *online* no Repositório Institucional da Biblioteca Universitária da UFSC. Além disso, declaro ciência de que Caio Henrique Roman Rodriguez, na condição de estudante da UFSC, é obrigado a depositar este documento, por se tratar de um Trabalho de Conclusão de Curso, no referido Repositório Institucional, em atendimento à Resolução Normativa n° 126/2019/CUn.

Por estar de acordo com esses termos, subscrevo-me abaixo.



Documento assinado digitalmente

Gabriel Thaler

Data: 17/07/2023 19:27:54-0300

CPF: \*\*\*.599.659-\*\*

Verifique as assinaturas em <https://v.ufsc.br>

---

Prof. Gabriel Thaler, M. Eng.  
UFSC/CTC/DAS/LIAE

## RESUMO

Este projeto de conclusão de curso tem como objetivo desenvolver técnicas de aprendizado de máquina para reduzir a dimensionalidade de dados adquiridos em uma bancada experimental de ensaios de amaciamento em compressores herméticos alternativos. Os testes de amaciamento referem-se a procedimentos específicos que avaliam o desempenho dos compressores em operação. O principal objetivo deste trabalho é submeter como entrada do algoritmo as medições de vibração e avaliar a implementação de uma técnica que utiliza redes neurais, especificamente o autoencoder *stacked*, para detectar padrões nesses dados. Com o pré-processamento do *dataset* em bandas de potência, o autoencoder proposto utiliza os pesos obtidos no treinamento para identificar as *features* mais relevantes que possam evidenciar o processo de amaciamento. Essa escolha de características para compor o espaço latente pode permitir o mapeamento de perfis de funcionamento, contribuindo para a detecção e a previsão de falhas nesse tipo de equipamento. Ao identificar as faixas de frequência mais importantes associadas ao processo de amaciamento, a abordagem proposta pode fornecer ferramentas adicionais para análise em testes de desempenho de compressores. Explorar métodos de inteligência artificial para auxiliar na análise de dados pode otimizar o monitoramento operacional de equipamentos como compressores herméticos alternativos, possibilitando avanços tecnológicos que podem influenciar positivamente em durabilidade, desempenho energético e outros fatores que melhorem a qualidade desses dispositivos.

**Palavras-chave:** Aprendizado de máquina. Autoencoder. Redução de dimensionalidade de dados. Amaciamento. Compressores herméticos alternativos.

## ABSTRACT

The objective of this project is to develop machine learning techniques to reduce the dimensionality of the data acquired in a test rig for the running-in of hermetic reciprocating compressors. Running-in tests refer to specific procedures that are used to evaluate the performance of a compressor during its operation. The main objective of this undergraduate thesis is to use vibration measurements as input to the algorithm and to evaluate the implementation of a technique that uses neural networks, in particular the stacked autoencoder, to detect patterns in these data. By pre-processing the dataset into power bands, the proposed autoencoder uses the weights obtained in training to identify the most relevant features that may be indicative of the running-in process. This choice of features to compose the latent space may allow the mapping of operating profiles and thus contribute to the detection and prediction of failures in this type of equipment. By identifying key frequency ranges associated with this process, the proposed approach may provide additional indicators to regular compressor performance testing. Exploiting artificial intelligence methods to support data analysis can optimise operational monitoring of equipment such as hermetic reciprocating compressors, facilitating technological advances that can positively influence durability, performance, and other quality factors.

**Keywords:** Machine learning. Autoencoder. Data dimensionality reduction. Running-in. Reciprocating hermetic compressors.

## LISTA DE FIGURAS

Figura 1 – Representação visual em corte de um compressor alternativo como exemplo. . . . .	21
Figura 2 – Etapas do ciclo de compressão no sistema de pistão-cilindro de um compressor alternativo. . . . .	22
Figura 3 – Área de contato real em superfícies rugosas. . . . .	23
Figura 4 – Representação de um neurônio biológico e de uma unidade de limite linear. . . . .	32
Figura 5 – Diagrama simplificado de uma RNA. . . . .	32
Figura 6 – Esquema de autoencoder simplificado. . . . .	33
Figura 7 – Esquema de autoencoder exemplificando um conjunto de dados. . .	34
Figura 8 – Diagrama estrutural de um SAE. . . . .	35
Figura 9 – Algoritmo de <i>backpropagation</i> . . . . .	36
Figura 10 – Função de ativação ReLU. . . . .	37
Figura 11 – Função de ativação Linear. . . . .	38
Figura 12 – Função de ativação Sigmoides. . . . .	38
Figura 13 – Instrumentação utilizada nos compressores testados. . . . .	41
Figura 14 – Exemplo de uma medição de dados de vibração no eixo longitudinal. .	42
Figura 15 – Dados eficazes de vibração em ensaios completos. . . . .	42
Figura 16 – Perfil de um arquivo de medição em 64 bandas de potência. . . . .	45
Figura 17 – Perfil de um arquivo de medição em 128 bandas de potência. . . . .	45
Figura 18 – Perfil de um arquivo de medição em 256 bandas de potência. . . . .	45
Figura 19 – Diagrama simplificado do SAE implementado. . . . .	48
Figura 20 – Curvas da função de perda em testes preliminares com processamento do sinal em 64 bandas de frequência. . . . .	54
Figura 21 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 64 bandas de frequência. . . . .	54
Figura 22 – Erro percentual médio em cada banda de potência na reconstrução em testes preliminares com processamento do sinal em 64 bandas de frequência. . . . .	55
Figura 23 – Curvas da função de perda em testes preliminares com processamento do sinal em 128 bandas de frequência. . . . .	56
Figura 24 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 128 bandas de frequência. . . . .	56
Figura 25 – Erro percentual médio entre cada banda de potência em testes preliminares com processamento do sinal em 128 bandas. . . . .	57
Figura 26 – Curvas da função de perda em testes preliminares com processamento do sinal em 256 bandas de frequência. . . . .	58

Figura 27 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 256 bandas de frequência. . . . .	58
Figura 28 – Erro percentual médio entre cada banda de potência em testes preliminares com processamento do sinal em 256 bandas. . . . .	60
Figura 29 – Curvas da função de perda do treinamento final com processamento do sinal em 64 bandas de frequência. . . . .	63
Figura 30 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 64 bandas de frequência. . . . .	63
Figura 31 – Curvas da função de perda do treinamento final com processamento do sinal em 128 bandas de frequência. . . . .	64
Figura 32 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 128 bandas de frequência. . . . .	65
Figura 33 – Curvas da função de perda do treinamento final com processamento do sinal em 256 bandas de frequência. . . . .	66
Figura 34 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 256 bandas de frequência. . . . .	67
Figura 35 – Desvio padrão de erro relativo médio nas baixas frequências do treinamento final com processamento do sinal em 256 bandas de frequência. . . . .	68
Figura 36 – Pesos médios calculados para cada uma das 64 bandas de potência. . . . .	70
Figura 37 – Intervalos de frequência mais influentes na escolha de 32 <i>features</i> contidas no espaço latente. . . . .	70
Figura 38 – Pesos médios calculados para cada uma das 128 bandas de potência. . . . .	71
Figura 39 – Intervalos de frequência mais influentes na escolha de 64 <i>features</i> contidas no espaço latente. . . . .	72
Figura 40 – Pesos médios calculados para cada uma das 256 bandas de potência. . . . .	72
Figura 41 – Intervalos de frequência mais influentes na escolha de 128 <i>features</i> contidas no espaço latente. . . . .	73
Figura 42 – Comparação entre as influências na escolha de <i>features</i> contidas no espaço latente dos três treinamentos finais. . . . .	74

## LISTA DE QUADROS

Quadro 1 – Classificação de transições tribológicas. . . . .	24
Quadro 2 – Treinamento padrão de um autoencoder SAE com <i>backpropagation</i> . . . . .	36
Quadro 3 – Quantidade de arquivos de medição de vibração. . . . .	41
Quadro 4 – Funcionamento do algoritmo de tratamento de dados. . . . .	43
Quadro 5 – Comparação entre métodos de normalização. . . . .	47
Quadro 6 – Número de neurônios em cada camada do autoencoder (considerando número de bandas de frequência). . . . .	50
Quadro 7 – Resumo da rotina de treinamento do autoencoder implementado. . . . .	51

## LISTA DE TABELAS

Tabela 1 – Comparação dos últimos valores de perda entre os treinamentos preliminares. . . . .	59
Tabela 2 – Comparação entre dimensões de conjuntos em testes preliminares e finais. . . . .	62
Tabela 3 – Comparação dos últimos valores de perda entre todos os treinamentos realizados. . . . .	68
Tabela 4 – Comparação da influência dos intervalos de frequência na análise de importância de <i>features</i> dos treinamentos realizados. . . . .	74

## LISTA DE ABREVIATURAS E SIGLAS

CP	Periódica cíclica (do inglês <i>cyclic, periodic</i> )
CR	Randômica cíclica (do inglês <i>cyclic, randomic</i> )
DFT	Transformada de Fourier discreta
I	Induzida
ISO	Induzida de ocorrência única (do inglês <i>induced, single occurrence</i> )
LIAE	Laboratório de instrumentação e automação de ensaios
MAPE	Erro médio percentual absoluto (do inglês <i>mean absolute percentage error</i> )
N	Não-induzida
NSO	Não-induzida de ocorrência única (do inglês <i>noninduced, single Occurrence</i> )
ReLU	Função identidade para valores positivos (do inglês <i>rectified linear unit</i> )
RNA	Redes neurais artificiais
SAE	Stacked Autoencoders (em português autoencoders empilhados)
SO	Ocorrência única (do inglês <i>single occurrence</i> )
STFT	Transformada de Fourier de curto prazo
RMS	Raiz do valor médio quadrático (do inglês <i>root mean square</i> )

## LISTA DE SÍMBOLOS

$\omega$	Frequência no tempo contínuo
$\Omega$	Frequência no tempo discreto
$T$	Intervalo de amostragem
$n$	Dimensionalidade dos dados de entrada e saída
$m$	Dimensionalidade dos dados codificados
$X$	Conjunto de dados de entrada
$Z$	Conjunto de dados codificados
$\hat{X}$	Conjunto de dados decodificados ou reconstruídos
$r$	Número de exemplos utilizados no treinamento
$n_e$	Quantidade de atributos presentes nas amostras utilizadas para treinamento
$x_i$	Amostra de entrada
$x_{ij}$	Um dos atributos específicos de uma amostra de entrada
$\hat{x}_i$	Reconstrução da amostra de entrada
$\hat{x}_{ij}$	Um dos atributos presentes na reconstrução da amostra de entrada
$X_{std}$	Um valor normalizado contido no conjunto de dados
$X_b$	Um valor original contido no conjunto de dados
$u_b$	Média dos valores originais
$s_b$	Desvio padrão da dos valores originais
$X_{min}$	Valor mínimo do conjunto de dados original
$X_{max}$	Valor máximo do conjunto de dados original

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
1.1	CONTEXTUALIZAÇÃO	16
1.2	EMPRESA	17
1.3	ATIVIDADES E OBJETIVOS	18
<b>1.3.1</b>	<b>Objetivo Geral</b>	<b>18</b>
<b>1.3.2</b>	<b>Objetivos Específicos</b>	<b>18</b>
1.4	ESTRUTURA DO TRABALHO	19
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>20</b>
2.1	COMPRESSORES HERMÉTICOS PARA REFRIGERAÇÃO	20
2.2	AMACIAMENTO	23
2.3	TRATAMENTO DE DADOS	26
<b>2.3.1</b>	<b>Transformada de Fourier</b>	<b>26</b>
<b>2.3.2</b>	<b>Transformada de Fourier de Curto Termo</b>	<b>28</b>
<b>2.3.3</b>	<b>Normalizadores</b>	<b>28</b>
2.3.3.1	Escalonamento Padrão	29
2.3.3.2	Escalonamento Min-Max	30
2.3.3.3	Escalonamento por Máximo Absoluto	30
2.4	APRENDIZADO DE MÁQUINA	31
<b>2.4.1</b>	<b>Redes Neurais Artificiais</b>	<b>31</b>
<b>2.4.2</b>	<b>Autoencoder</b>	<b>32</b>
<b>2.4.3</b>	<b>Stacked Autoencoders</b>	<b>34</b>
2.4.3.1	Funções de Ativação	37
<b>2.4.4</b>	<b>Seleção de Features</b>	<b>38</b>
<b>3</b>	<b>DESENVOLVIMENTO</b>	<b>40</b>
3.1	PRÉ-PROCESSAMENTO DE DADOS	40
<b>3.1.1</b>	<b>Aquisição de Dados</b>	<b>40</b>
<b>3.1.2</b>	<b>Padronização de Dados</b>	<b>43</b>
<b>3.1.3</b>	<b>Normalização</b>	<b>46</b>
3.2	IMPLEMENTAÇÃO	47
<b>3.2.1</b>	<b>Estrutura do SAE</b>	<b>47</b>
<b>3.2.2</b>	<b>Definição de Hiperparâmetros</b>	<b>49</b>
<b>3.2.3</b>	<b>Análise de Desempenho Preliminar</b>	<b>50</b>
3.2.3.1	Treinamento Preliminar – 64 Bandas	53
3.2.3.2	Treinamento Preliminar – 128 Bandas	55
3.2.3.3	Treinamento Preliminar – 256 Bandas	57
3.3	CONSIDERAÇÕES FINAIS	59
<b>4</b>	<b>RESULTADOS</b>	<b>62</b>

4.1	TREINAMENTOS FINAIS . . . . .	62
4.1.1	<b>Pré-processamento dos Dados de Entrada em 64 Bandas de Potência</b> . . . . .	62
4.1.2	<b>Pré-processamento dos Dados de Entrada em 128 Bandas de Potência</b> . . . . .	64
4.1.3	<b>Pré-processamento dos Dados de Entrada em 256 Bandas de Potência</b> . . . . .	65
4.1.4	<b>Considerações Finais de Desempenho entre os Treinamentos</b> .	68
4.2	INTERPRETAÇÃO DAS <i>FEATURES</i> CONTIDAS NO ESPAÇO LATENTE . . . . .	69
4.2.1	<b>Espaço Latente de 32 <i>Features</i></b> . . . . .	69
4.2.2	<b>Espaço Latente de 64 <i>Features</i></b> . . . . .	70
4.2.3	<b>Espaço Latente de 128 <i>Features</i></b> . . . . .	71
4.2.4	<b>Considerações Finais da Análise de Importância das <i>Features</i></b> .	73
5	<b>CONSIDERAÇÕES FINAIS</b> . . . . .	75
5.1	CONCLUSÃO . . . . .	75
5.2	SUGESTÕES PARA TRABALHOS FUTUROS . . . . .	76
	<b>REFERÊNCIAS</b> . . . . .	77

# 1 INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

Os refrigeradores domésticos, possuem grande relevância na sociedade atual. Em 2015, pesquisas estatísticas indicaram que cerca de 97,8% (IBGE, 2016) das residências no Brasil possuíam esses eletrodomésticos, fato que evidencia a dependência da população em relação a sistemas de refrigeração. Por ser um dispositivo amplamente utilizado, é importante que ele apresente alto desempenho e eficiência, além de ser comercializado a um preço acessível. Nesse sentido, as empresas desse ramo têm investido em pesquisas para desenvolver produtos de qualidade e desempenho elevado, sem desconsiderar um custo de venda coerente para o consumidor.

Em 2019, a área de pesquisa e desenvolvimento do Laboratório de Instrumentação e Automação de Ensaios (LIAE) elaborou uma bancada experimental com o propósito de determinar o perfil de amaciamento de compressores herméticos (THALER, 2019). Esse fenômeno pode ser descrito como o contato entre duas superfícies que ainda não foram operadas, gerando choques nas regiões mais irregulares. Apesar da pequena área de contato inicial, os pequenos desarranjos entre as superfícies são gradualmente removidos, permitindo que a área de contato real seja ampliada ao longo do tempo (BLAU; KOMANDURI, 1990). Com isso, o comportamento dos sistemas mecânicos tende a sofrer alteração nas primeiras horas de uso. Para o propósito específico do estudo em compressores herméticos de refrigeração, foram conduzidos ensaios de até 72 h de duração em compressores que nunca haviam sido operados antes e em compressores que já se encontravam amaciados.

A ideia do processo de desenvolvimento de um refrigerador inclui diversos componentes em seu sistema de refrigeração, sendo que um deles é o compressor. Como o compressor é o componente que mais consome energia em um refrigerador, é comum que sejam realizados ensaios de desempenho energético em compressores domésticos tanto para atividades de pesquisa e desenvolvimento quanto para avaliação da qualidade da produção. Por consequência, os ensaios de amaciamento são procedimentos importantes para essa linha de pesquisa e desenvolvimento da indústria, pois deve-se estabelecer um protocolo de amaciamento e monitorar cuidadosamente o processo, para garantir que tenha sido realizado corretamente antes dos ensaios de desempenho. Um procedimento de amaciamento adequado evita tanto o desperdício de recursos quanto que equipamentos não amaciados gerem dados de desempenho energético imprecisos. Geralmente os problemas nesses ensaios se originam na preparação, pela falta de padronização. Isso ocorre devido ao fato de que cada empresa pode definir arbitrariamente o tempo destinado para o processo de amaciamento, que pode ser curto ou longo demais. Com o emprego de métodos que caracterizem adequadamente o amaciamento, os ensaios de desempenho se tornam mais exatos e

confiáveis, gerando dados para tomada de decisões mais assertivas.

Na bancada experimental, foram coletados dados de diversas grandezas físicas possivelmente conectadas ao amaciamento em grande quantidade, incluindo dados de vibração nas calotas superior e inferior do compressor. Em análises preliminares, no domínio do tempo, notou-se que em um dos modelos a curtose e o valor RMS da corrente elétrica possuem comportamento em ensaios de amaciamento diferente do que foi verificado com amostras já amaciadas. No entanto, devido ao longo tempo de ensaio e à alta taxa de aquisição (25,6 kHz), uma análise frequencial pode ser interessante para detectar anomalias, em uma linha parecida com o que é aplicado em outras máquinas rotativas (LALANNE; FERRARIS, 1998). Para facilitar esse tipo de análise, foi proposta uma diminuição da dimensionalidade dessas medições, mas de modo que não prejudique o acesso e futuras análises de perfis de funcionamento contidos nos dados.

## 1.2 EMPRESA

O Laboratório de Instrumentação e Automação de Ensaios (LIAE), fundado em 2007, é uma entidade de atuação multidisciplinar que abrange as áreas de engenharia mecânica, elétrica e de controle e automação. É um laboratório da Universidade Federal de Santa Catarina que atua dentro do Laboratório de Metrologia e Automação (Labmetro) e que tem forte atuação com parceiros industriais. Seus projetos concentram-se em aplicações nas seguintes áreas:

- Automação e Controle de Ensaios: implementação de técnicas de automação e controle para otimizar e aprimorar os processos de ensaios industriais;
- Desenvolvimento de Equipamentos Eletrônicos e Eletromecânicos: projeto e criação de dispositivos eletrônicos e eletromecânicos para medição, atuação e controle em ambientes industriais;
- Avaliação e Garantia da Confiabilidade Metrológica: realização de estudos e análises para garantir a precisão e confiabilidade dos equipamentos e instrumentos de medição utilizados nos ensaios;
- Aplicação de Inteligência Artificial na Metrologia Industrial: utilização de técnicas avançadas de inteligência artificial para otimizar processos de medição e análise de dados em contextos industriais;
- Processamento Digital de Sinais Aplicado à Medição: aplicação de técnicas de processamento digital de sinais para aprimorar a qualidade das medições realizadas nos ensaios.

O LIAE tem como objetivo principal colaborar com a indústria, buscando constantemente o desenvolvimento e aprimoramento de soluções inovadoras e eficientes em instrumentação e automação de ensaios.

### 1.3 ATIVIDADES E OBJETIVOS

A proposta deste projeto é implementar um método de redução de dimensionalidade de dados de vibração para permitir a análise da aplicabilidade desses dados como indicadores de amaciamento de compressores herméticos de refrigeração. Algoritmos de autoencoder são validados como um método indicado para esse tipo de função (HINTON, G. E.; SALAKHUTDINOV, 2006). No treinamento de um autoencoder, são obtidos pesos a partir do conjunto de dados de entrada e esses são utilizados como *features* contidas na versão reduzida feita pelo autoencoder, indicando faixas de frequência possivelmente mais relevantes ao amaciamento de compressores nesta aplicação particular. Com a redução da dimensão desse grande conjunto de dados é esperado que, com base nos pesos calculados, possa ser mais fácil analisar indicadores que evidenciem perfis de operação dos equipamentos.

#### 1.3.1 Objetivo Geral

O objetivo geral deste projeto é analisar as *features* contidas na versão reduzida gerada pelo modelo, considerando a redução de dimensionalidade dos dados coletados em uma bancada de detecção de amaciamento em compressores herméticos.

#### 1.3.2 Objetivos Específicos

Para colaborar com a organização e guiar o atingimento do objetivo geral do projeto, foram propostos os seguintes objetivos específicos:

- mapear os dados de vibração de ensaios experimentais, agrupando informações contendo os diretórios de cada arquivo;
- desenvolver um algoritmo de pré-processamento que separe os dados de medição em bandas de potência;
- propor e implementar um método baseado em autoencoder para redução da dimensionalidade dos dados;
- realizar testes preliminares com o algoritmo de autoencoder desenvolvido, visando determinar hiperparâmetros adequados;
- implementar o autoencoder desenvolvido utilizando como entrada a totalidade dos dados de vibração medidos em um modelo de compressor e avaliar os resultados.

## 1.4 ESTRUTURA DO TRABALHO

Os parágrafos seguintes caracterizam a estrutura desta monografia.

O Capítulo 2 trata da fundamentação teórica utilizada no desenvolvimento do projeto, elucidando conceitos relacionados ao fenômeno de amaciamento, organização de dados e autoencoders.

O Capítulo 3 expõe as escolhas e o desdobramento da elaboração dos algoritmos de tratamento de dados e de implementação do autoencoder, de modo a examinar o processo de escolha de parâmetros e da arquitetura das redes neurais, visando aperfeiçoar o resultado obtido com relação aos dados de entrada.

O Capítulo 4 analisa os dados obtido por meio de métricas de desempenho dos autoencoders, buscando explicitar se os dados resultantes da execução do algoritmo realmente são condizentes com os dados originais.

O Capítulo 5 detalha as considerações pessoais do autor acerca do projeto desenvolvido, junto com uma visão geral resumida dos passos que conduziram ao resultado final. Além disso, o capítulo oferece sugestões para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados conceitos importantes para a elaboração deste projeto. A Seção 2.1 discorre acerca do compressor hermético alternativo, expondo sua aplicação em circuitos de refrigeração. Por sua vez, a Seção 2.2 resume os conceitos já estabelecidos na literatura no que diz respeito ao processo de amaciamento em sistemas mecânicos. A Seção 2.3 detalha os conteúdos utilizados para a implementação dos algoritmos com um enfoque no tratamento de dados. Por fim, na Seção 2.4 são apresentados fundamentos teóricos para a implementação do autoencoder, dando enfoque também para o método de seleção de *features* utilizado.

### 2.1 COMPRESSORES HERMÉTICOS PARA REFRIGERAÇÃO

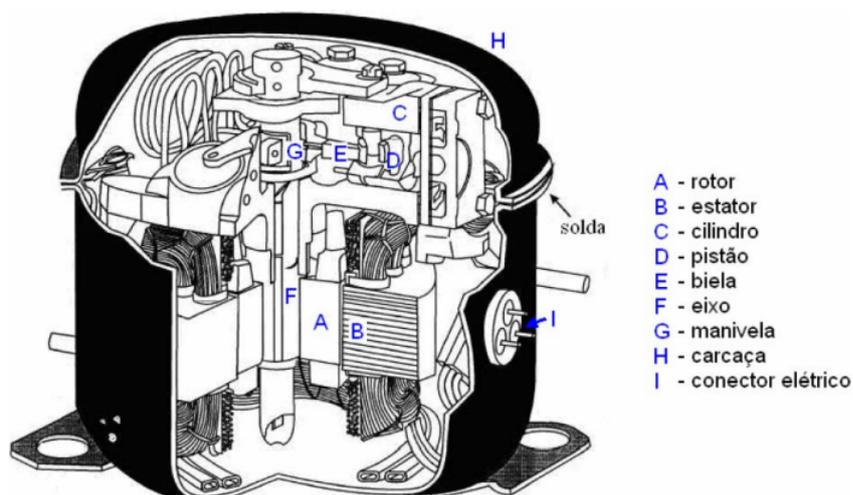
Os compressores são equipamentos que têm como função principal aumentar a pressão de fluidos em estado gasoso ou de vapor (STOECKER; JONES, 1985). O mercado oferece muitos tipos de compressores, cada um com características distintas que o tornam mais adequado para usos específicos. Esses dispositivos podem ser divididos e caracterizados por dois tipos principais: dinâmicos e de deslocamento positivo. Os compressores dinâmicos podem ser centrífugos ou axiais e se caracterizam por aumentar a energia cinética do fluido e transformá-la em aumento de pressão. Já os compressores de deslocamento positivo, comprimem volumes finitos de fluido, gerando aumento de pressão (STOECKER; JONES, 1985; HANLON, 2001; BLOCH; HOEFNER, 1996). Este projeto tem o enfoque em técnicas de amaciamento para compressores herméticos alternativos, que são compressores de deslocamento positivo.

O compressor alternativo converte o movimento rotacional, normalmente gerado por um motor elétrico, em movimento linear por meio de um sistema biela-manivela, com um pistão acoplado na extremidade oposta do sistema, inserido dentro de um cilindro (STOECKER; JONES, 1985; BLOCH; HOEFNER, 1996). Na Figura 1 é apresentada a vista em corte de um exemplo de compressor hermético utilizado em refrigeradores domésticos. Em um exemplo de compressor como esse, os terminais elétricos externos ativam o motor elétrico, composto por estator e rotor. Consequentemente, ocorre a formação de um movimento de rotação, que é então passado para o sistema biela-manivela por intermédio do eixo. O movimento linear do sistema biela-manivela move o pistão, localizado dentro do cilindro, que acaba criando uma câmara de compressão cujo volume varia.

Em compressores do tipo hermético, tanto o motor elétrico quanto o mecanismo de compressão estão posicionados dentro de um corpo hermeticamente selado, dificilmente acessíveis para reparo ou manutenção (HUNDY; TROTT; WELCH, 2008). Esses compressores geralmente são menores e mais silenciosos do que outras variantes semi-herméticas ou abertas do equipamento, tornando-os populares em aplicações

comerciais e domésticas. Entretanto, esses modelos podem deixar de ser tão eficientes por causa do fato de o fluido refrigerante poder ser aquecido pelos sistemas internos do compressor (STOECKER; JONES, 1985).

Figura 1 – Representação visual em corte de um compressor alternativo como exemplo.



Fonte: (HENKLEIN, 2006)

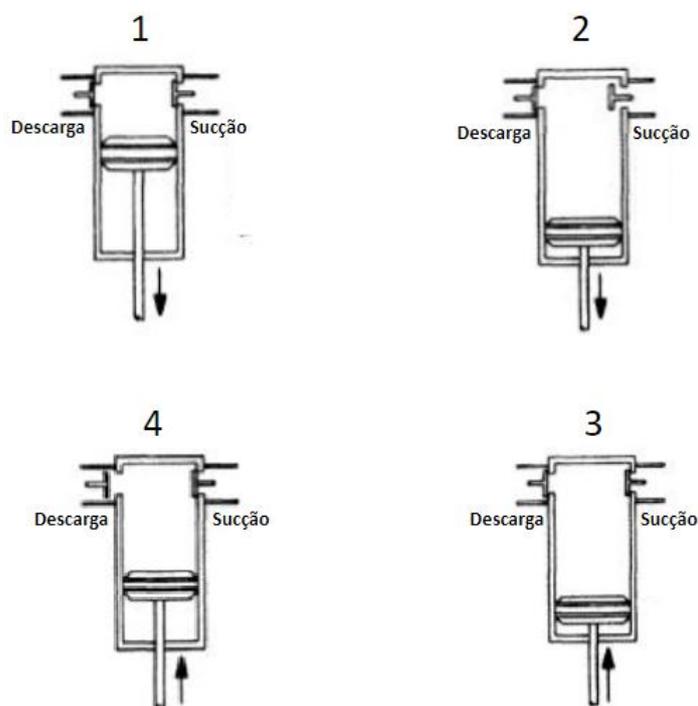
As válvulas de sucção e de descarga têm a responsabilidade de permitir que o fluido escoe em uma única direção e, automaticamente, são acionadas em virtude das diferenças de pressão entre os pontos internos do cilindro e os pontos de descarga e de sucção. Tais válvulas permitem a conexão do interior da câmara de compressão com os passadores de descarga e de sucção, respectivamente. Conseqüentemente, o fluido se desloca da câmara de compressão para o ambiente externo através do passador de descarga, enquanto no caso do passador de sucção, ocorre o escoamento no sentido contrário (DOSSAT, 2004; BOLES; ÇENGEL, 2013).

O ciclo de compressão que ocorre nesse tipo de equipamento é ilustrado na Figura 2. A análise é feita a partir do momento em que as válvulas se fecham, representado na figura pela condição identificada com o número 1. Quando o pistão se move para baixo, a pressão na câmara fica menor que a pressão de sucção e ocorre a abertura da válvula de sucção, fazendo com que a câmara seja preenchida por fluido refrigerante. Quando a pressão interna se equaliza com a pressão de sucção, a válvula fecha e o cilindro inicia o movimento de compressão. Com isso, a câmara se reduz e a pressão do fluido aumenta. Quando a pressão interna for maior do que a pressão de descarga, após a compressão interna, ocorre a abertura da válvula de descarga, permitindo uma liberação do fluido para o circuito externo. Uma vez que a pressão interna se equaliza com a pressão de descarga, a válvula de descarga se fecha e o cilindro inicia o movimento de expansão, até que seja atingida a condição inicial desta análise. Por ser um ciclo, o mesmo volta a se repetir, possibilitando o funcionamento

do sistema.

Nos circuitos de refrigeração que utilizam a compressão de vapor, o compressor desempenha um papel fundamental ao transferir o fluido entre os trocadores de calor e ao converter o trabalho em um diferencial de pressão. Essa função permite, juntamente com o dispositivo de expansão, a transferência de calor em direção oposta à natural, ou seja, da fonte fria para a fonte quente (STOECKER; JONES, 1985; PRATA; BARBOSA JR, 2009).

Figura 2 – Etapas do ciclo de compressão no sistema de pistão-cilindro de um compressor alternativo.



1. Expansão
2. Sucção
3. Compressão
4. Descarga

Fonte: Adaptado de (BLOCH; HOEFNER, 1996)

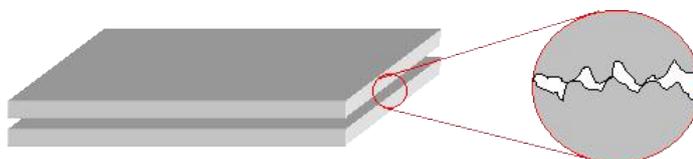
Para o desenvolvimento deste projeto, foi adotado um modelo conhecido como ciclo quente, que consiste em manter o fluido em estado de vapor superaquecido durante todo o processo, sendo utilizado em algumas análises de compressores fora do sistema de refrigeração. Vale ressaltar que esse tipo de circuito possui uma eficiência negativa, o que o inviabiliza para aplicações de refrigeração convencional. No entanto, ele é adequado para avaliar algumas condições de operação do compressor, como

pressão de sucção, pressão de descarga, temperatura, vibração, entre outros. Essa abordagem permite realizar certos testes em compressores sem a necessidade de implementar um circuito de refrigeração habitual, dado que do ponto de vista do compressor o circuito equivalente observado é o mesmo que seria em um circuito com mudança de fase.

## 2.2 AMACIAMENTO

A definição original do amaciamento, documentada em 1933 (ABBOT; FIRES-TONE, 1933), explica o conceito de curva de contato de superfície, relacionando-o com a distância entre os picos das superfícies envolvidas. De acordo com essa definição, quando duas superfícies não operadas entram em contato, elas se tocam apenas nos pontos mais altos das irregularidades, resultando em uma área de contato real bastante reduzida. Conforme as superfícies são submetidas a carga, as irregularidades são gradualmente reduzidas e a área de contato real se expande. A Figura 3 ilustra esse conceito, mostrando a área representada pelo pequeno espaço que separa as duas superfícies.

Figura 3 – Área de contato real em superfícies rugosas.



Fonte: Adaptado de (STACHOWIAK; BATCHELOR, 2013)

Considerando aplicações práticas, o contato entre superfícies rugosas conduz ao alisamento, enquanto o atrito entre superfícies lisas gera rugosidades. Desse modo, o atrito leva as duas superfícies a um estado de equilíbrio em termos de rugosidade (IGOR'V; DOBYČIN; KOMBALOV, 1982). Em Abbot e Firestone (1933), também foi constatado que o desgaste inicial das irregularidades ocorre rapidamente, porém diminui gradualmente à medida que a área de contato aumenta, caracterizando um período inicial de amaciamento. Essas observações iniciais evidenciam a influência do desgaste nos estágios iniciais do contato entre superfícies. Contudo, atualmente é reconhecido que o fenômeno do amaciamento envolve diversos fatores em diferentes escalas, provocando mudanças, por exemplo, na orientação cristalográfica e na tempera dos materiais envolvidos, quando metálicos (BLAU, 2005).

Além disso, o desgaste entre as superfícies durante o amaciamento não se limita ao desgaste abrasivo. Em regiões de alta temperatura geradas pelo atrito, podem ocorrer pontos de microsoldagem entre as superfícies, resultando em desgaste adesivo (KHALIFA, 1996). O aquecimento gerado pela fricção pode criar uma camada de

óxido que evita o desgaste adesivo do metal em contato, prevenindo danos severos à estrutura. Essa camada de óxido, ao se desgastar, forma uma fina camada que age como lubrificante sólido natural entre as superfícies (ANTLER, 1964).

O amaciamento é um fenômeno que ocorre simultaneamente em escalas macro, micro e nanométricas. Do ponto de vista macroscópico, o amaciamento envolve o ajuste do posicionamento das peças. Caso esse ajuste não seja realizado durante essa fase, pode resultar em desgaste prematuro e perda de eficiência (BLAU, 2005, 2008).

O sistema proposto em Blau e Komanduri (1990) estabelece um sistema de classificação para as transições, levando em consideração duas características: sua ocorrência natural no sistema e a frequência com que acontecem. Quando o sistema é operado de maneira diferente com o intuito de provocar uma transição tribológica, ela é categorizada como induzida (I). Por outro lado, se o fenômeno ocorre naturalmente como parte da operação normal do sistema, é classificado como não induzida (N).

A classificação da frequência de transição pode ser feita com base em três tipos: ocorrência única (SO, do inglês *Single Occurrence*), cíclica aleatória (CR, do inglês *Cyclic, Randomic*) ou cíclica periódica (CP, do inglês *Cyclic, Periodic*). O primeiro tipo acontece apenas uma vez na operação do sistema, enquanto as outras duas ocorrem de forma cíclica. O que diferencia o segundo do terceiro tipo é que no segundo a alteração ocorre em intervalos aleatórios, enquanto na terceira ela ocorre em intervalos periodicamente determinados. O modelo de classificação é apresentado no Quadro 1. Assim, considerando que o amaciamento ocorre apenas uma vez no início da operação, pode ser classificado de acordo com esse modelo como uma transição ISO (induzida e de ocorrência única) quando é induzido, ou NSO (não induzido e de ocorrência única) quando ocorre naturalmente.

Quadro 1 – Classificação de transições tribológicas.

<b>Frequência</b>	<b>Induzida</b>	<b>Não-induzida (natural)</b>
<b>Ocorrência única</b>	<b>ISO</b>	<b>NSO</b>
<b>Ocorrência múltipla (aleatória)</b>	<b>ICR</b>	<b>NCR</b>
<b>Ocorrência múltipla (periódica)</b>	<b>ICP</b>	<b>NCP</b>

Fonte: Adaptado de (BLAU; KOMANDURI, 1990)

Na caracterização do fenômeno dentro das transições tribológicas, é importante distinguir entre o amaciamento natural e o amaciamento forçado. O amaciamento natural ocorre como resultado da operação normal do sistema e é o tipo mais comum nos mecanismos. Já o amaciamento forçado ocorre quando são aplicadas rotinas de operação diferentes daquelas usuais, com o objetivo específico de amaciar o sistema, como alterações na velocidade de operação e na lubrificação (BLAU, 2005, 2008).

Durante o amaciamento, a duração da curva de atrito pode ser reduzida ou es-

tendida, dependendo do alinhamento das superfícies e da carga mecânica do contato. Pressões mais elevadas exercidas nas superfícies tendem a promover uma transição mais ágil do sistema para um estado tribológico estável (BLAU, 2005). No entanto, é relevante considerar que a heterogeneidade na composição do sistema pode ocasionar flutuações momentâneas no atrito, sem alterar substancialmente o padrão geral das curvas de atrito.

O amaciamento em sistemas complexos ocorre de maneira simultânea em várias peças, resultando em reações e efeitos distintos em cada superfície de contato (BLAU, 2005). Portanto, é importante examinar tanto o comportamento global do sistema como o comportamento local entre seus componentes. O processo de amaciamento por completo não deve ser considerado um fenômeno reservado, mas sim uma conjunção de distintos perfis de funcionamento que são notados ao início da atuação do equipamento. É recomendado conhecer a maneira que se correlacionam o amaciamento e o desgaste comum do sistema, para distinguir de maneira mais exata o momento em que o compressor pode ser considerado amaciado. Essa distinção pode ser uma tarefa complexa porque o tempo total de amaciamento não está necessariamente relacionado com o tempo necessário para assentamento da taxa de desgaste. Além da medição direta, é possível identificar o amaciamento indiretamente por meio de indicadores, como a concentração de partículas de desgaste no lubrificante, vibração, corrente do motor, entre outros (BLAU, 2005).

Em um estudo realizado por Martin (1992), propõe-se uma abordagem indireta para detectar o período de amaciamento por meio de medições de vibração, utilizando uma função de densidade de probabilidade aplicada ao sinal vibracional obtido. Com base na ideia de que superfícies pouco danificadas exibem uma distribuição de asperezas semelhante a uma curva gaussiana, e que essa distribuição se afasta do formato tradicional em sino à medida que o contato entre as superfícies se modifica, o autor sugere o uso da medida de curtose para identificar o achatamento da distribuição vibracional em relação à curva gaussiana. Usando esse método, o autor demonstra que o valor de curtose se aproxima de três à medida que a superfície é amaciada, observando alterações apenas em casos de danos superficiais.

No artigo de Lockwood e Dalley (1992), é apresentada uma proposta para analisar o óleo lubrificante da superfície, com o objetivo de detectar o desgaste moderado (benigno) nas superfícies, o qual é comum durante o período de amaciamento. A ideia é verificar se as partículas de desgaste estão dentro dos limites detectáveis por meio de análise espectrométrica de metais (inferiores a 10  $\mu\text{m}$ ). Esse método poderia ser utilizado para identificar o amaciamento por desgaste, examinando a concentração e a composição das partículas de desgaste no óleo lubrificante por meio da purga do óleo em diferentes momentos durante as primeiras horas de operação do compressor.

A detecção do amaciamento também pode ser feita por meio da corrente elé-

trica do motor, que age como um transdutor, convertendo informações sobre a carga mecânica aplicada no sistema em sinais elétricos. A análise desses sinais ocorre em duas vertentes: uma delas avalia o valor RMS (raiz do valor médio quadrático, conhecido como *Root Mean Square* em inglês), enquanto a outra observa as flutuações instantâneas da corrente do motor (EISSEBERG; HAYNES, 1992). Esse método pode ser aplicado para avaliar o desgaste em compressores, levando em consideração que a potência do motor pode ser dividida em energia dissipada na compressão do fluido, perdas elétricas, perdas nas válvulas de sucção e descarga, e perdas por fricção (DAGILIS; VAITKUS, 2009). Com um conhecimento adequado do processo e instrumentação do compressor, é possível isolar as perdas por fricção, permitindo a detecção dos fatores relacionados ao amaciamento.

Em Khonsari, Ghatreh Samani e Akbarzadeh (2021), os autores apresentam uma revisão dos estudos teóricos e práticos sobre o amaciamento, os quais investigam o contato entre superfícies de forma isolada da influência de outros mecanismos. Constatou-se, a partir das pesquisas conduzidas, que as mudanças na topografia e no coeficiente de atrito são os indicadores mais claros do processo. Além disso, foram conduzidos trabalhos que analisaram esse comportamento em modelos experimentais com materiais específicos, tais como Zambrano *et al.* (2020) e Blau (2008).

A maneira como o sistema é amaciado pode ter impactos significativos na durabilidade do produto. No setor automotivo, por exemplo, considera-se que um amaciamento adequado das lonas de freio é necessário para garantir torque suficiente durante a frenagem (BLAU, 2005). Além disso, pesquisas em motores de combustão interna também estabelecem uma conexão entre o fenômeno e a qualidade do produto final (BLAU, 2008).

## 2.3 TRATAMENTO DE DADOS

Para fundamentar o algoritmo elaborado para o pré-processamento de dados, foi necessário explorar conceitos detalhados nas Subseções 2.3.1 e 2.3.2, para servir como base para a rotina de padronização do conjunto de dados de entrada.

### 2.3.1 Transformada de Fourier

A transformada de Fourier é uma técnica utilizada para decompor sinais temporais em infinitos componentes de frequência (MÜLLER, 2015). Em outras palavras, ela permite analisar um sinal de tempo contínuo e representá-lo como uma combinação de ondas senoidais complexas. Ao lidar com sinais contínuos e não periódicos, é importante considerar que a superposição dessas ondas senoidais complexas abrange uma gama infinita de frequências, que se estendem de menos infinito a mais infinito.

Para realizar a representação de um sinal originalmente no domínio do tempo

utilizando a transformada de Fourier, utiliza-se a integral apresentada em (2). Essa integral permite considerar todas as possíveis frequências presentes no sinal e ponderar a contribuição de cada uma delas na composição do sinal original.

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega \quad (1)$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (2)$$

Na equação (1), o sinal  $x(t)$  é representado como uma combinação ponderada de senoides sobrepostas. Essa combinação é obtida através de uma integral, em que o peso de cada senoide é determinado por  $\frac{1}{2\pi} X(\omega) d\omega$ . Essa relação estabelece que  $x(t)$  e  $X(\omega)$  formam um par na transformada de Fourier.

Essencialmente, o que a equação (1) mostra é que se pode decompor o sinal  $x(t)$  em suas diferentes componentes de frequência, representadas por senoides complexas. Cada componente possui um peso específico determinado pela integral. Dessa forma, a Transformada de Fourier permite analisar e compreender a composição em frequência do sinal  $x(t)$ , fornecendo a representação no domínio da frequência,  $X(\omega)$ . Esquemáticamente, tem-se que:

$$x(t) \xrightarrow{\text{Transformada de Fourier}} X(j\omega). \quad (3)$$

A transformada de Fourier, representada por  $X(\omega)$ , descreve o sinal  $x(t)$  como uma função da frequência  $\omega$  e é amplamente utilizada como uma representação no domínio de frequência para  $x(t)$  (HAYKIN, S. S.; VAN VEEN, 2001). A equação (2) é conhecida como a transformada de Fourier de  $x(t)$ , pois converte o sinal do domínio do tempo para sua representação no domínio de frequência. Por outro lado, a equação (1) é denominada transformada de Fourier inversa, pois realiza o processo inverso, ou seja, converte a representação no domínio de frequência,  $X(\omega)$ , de volta ao domínio do tempo.

É importante mencionar que essa representação da transformada de Fourier é aplicável a sinais no tempo contínuo. No entanto, no contexto deste trabalho, o sinal adquirido é de tempo discreto. Apesar dessa diferença, há uma correspondência entre a frequência no tempo contínuo ( $\omega$ ) e a frequência no tempo discreto ( $\Omega$ ).

Pode-se definir as senoides complexas como  $x(t) = e^{j\omega t}$  e  $g(k) = e^{j\Omega k}$ . Estabelecemos uma relação entre as frequências dessas senoides ao exigir que  $g(k)$  esteja relacionado com  $x(t)$ . Supomos que  $g(k)$  seja igual às amostras de  $x(t)$  tomadas em intervalos  $T$ , ou seja,  $g(k) = x(kT)$ . Essa condição implica:

$$e^{j\Omega k} = e^{j\omega Tk}. \quad (4)$$

Pode ser estabelecida a relação  $\Omega = \omega T$ , onde  $\Omega$  representa a frequência no tempo discreto e  $\omega$  é a frequência no tempo contínuo. Essa relação indica que a

frequência no tempo discreto é obtida multiplicando a frequência no tempo contínuo pelo intervalo de amostragem  $T$  (HAYKIN, S. S.; VAN VEEN, 2001).

No cálculo do espectro de frequência de um sinal discreto no tempo, uma técnica muito usada é a transformada de Fourier discreta (DFT, do inglês *Discrete Fourier Transform*). Explicando em termos práticos, a DFT é a aplicação da transformada de Fourier em um sinal amostrado considerando um intervalo de tempo limitado.

### 2.3.2 Transformada de Fourier de Curto Termo

A transformada de Fourier de curto termo (STFT, do inglês *Short-Time Fourier Transform*) é um método amplamente adotado para análises espectrais de frequência em sinais não estacionários que se alteram ao longo do tempo. A STFT converte o sinal de entrada, o fazendo com o uso de técnicas de janelamento para posteriormente efetuar a aplicação da DFT em cada banda separada. Assim sendo, a STFT possibilita análises das propriedades espectrais do sinal de entrada em janelas temporais definidas. A STFT pode ser representada pela seguinte equação:

$$\text{STFT}\{x[k]\}(\delta, \omega) = \sum_{k=-\infty}^{\infty} x[k]w[k - \delta]e^{-j\omega k}, \quad (5)$$

em que  $x(k)$  representa o sinal no domínio do tempo a ser transformado,  $w[k - \delta]$  é a função de janelamento aplicada com um deslocamento  $\delta$ , e  $e^{-j\omega k}$  é a unidade complexa (MANHERTZ; BERECZKY, 2021). A STFT é uma técnica amplamente utilizada para analisar a variação de frequência ao longo do tempo em sinais não estacionários, sendo aplicada em áreas como detecção de falhas em rolamentos e diagnósticos de funcionamento nesses equipamentos (BOUDIAF *et al.*, 2016).

### 2.3.3 Normalizadores

No campo de aprendizado de máquina e redes neurais, a normalização de dados ajuda a diminuir o viés nos pesos da rede. Quando os dados de entrada possuem escalas muito diferentes, isso pode resultar em um desequilíbrio nos pesos aprendidos durante o treinamento. Ao normalizar os dados, todas as características passam a ter a escala similar, mitigando, assim, possíveis vieses nos pesos da rede (HASTIE *et al.*, 2009). Além disso, a normalização dos dados tipicamente acelera a convergência do treinamento. Ao ter todas as características projetadas em escalas semelhantes, o otimizador pode ajustar os pesos de maneira mais eficiente para minimizar a função de perda. Isso resulta em um treinamento mais rápido e menos sensível a problemas como gradientes desequilibrados.

A normalização dos dados também melhora o equilíbrio na escala em *datasets* que possuem valores muito distintos. Em algumas situações, as operações numéricas durante o treinamento podem levar a problemas de estabilidade quando os valores dos

dados são muito grandes ou muito pequenos. A normalização dos dados para uma faixa específica, considerando o conjunto de dados, pode ajudar a evitar esses problemas, garantindo que os valores estejam dentro de uma escala razoável (BROWNLEE, 2016).

Outra vantagem da normalização dos dados é lidar com *outliers*. Esses são pontos de dados que são significativamente diferentes da maioria dos outros pontos. Sua presença pode afetar negativamente a representação aprendida por modelos de aprendizado, como o autoencoder, especialmente se tiverem uma escala desproporcional. Em algumas técnicas de normalização dos dados, os *outliers* assumem menor influência, pois são convertidos para a mesma escala dos outros pontos.

Na Subseção 3.1.3, são analisados e comentados os testes realizados com diferentes técnicas de normalização. Nesta seção são detalhados os métodos utilizados neste trabalho, provenientes de uma biblioteca especializada em aprendizado de máquina (PEDREGOSA *et al.*, 2011).

### 2.3.3.1 Escalonamento Padrão

Uma das principais transformações realizadas pelo escalonamento padrão é a centralização dos dados. Isso significa que a média dos dados brutos será ajustada para próximo de zero (BUITINCK *et al.*, 2013). Essa centralização é útil pois permite que o modelo se concentre nas diferenças relativas entre os valores. Essa centralização ocorre com base na equação (6).

$$X_{std} = \frac{x_b - u_b}{s_b} \quad (6)$$

Na equação (6),  $X_{std}$  representa a variável transformada após a normalização, enquanto  $x_b$  é a variável original que está sendo padronizada. O termo  $u_b$  representa a média dos valores originais e  $s_b$  é o desvio-padrão dos valores originais.

Esse tipo de escalonamento também dimensiona os dados para terem um desvio-padrão de um. Essa escala é importante para evitar que recursos com variações muito maiores dominem o treinamento em relação aos recursos com variações menores.

Uma vantagem que a normalização feita por meio da função apresentada em (6) é a preservação da forma da distribuição original dos dados. Se a distribuição original é aproximadamente normal ou se aproxima de uma distribuição normal, ela ainda terá essa forma após a normalização. No entanto, se a distribuição original não é normal, ela permanecerá não normal após a normalização.

É importante mencionar que esse método pode ser mais sensível a *outliers* que outros métodos de normalização. Dessa forma, na presença desses valores discrepantes no conjunto de dados, pode haver distorção da média e do desvio-padrão

utilizados. Isso pode resultar em uma normalização menos eficaz e potencialmente afetar o desempenho do modelo.

### 2.3.3.2 Escalonamento Min-Max

O escalonamento min-max é uma técnica de normalização que valoriza o intervalo entre o menor e o maior valor contido no conjunto de dados de entrada (BUITINCK *et al.*, 2013). Esse cálculo ocorre por meio das equação (7).

$$X_{std} = \frac{X_b - X_{min}}{X_{max} - X_{min}} \quad (7)$$

Na equação (7),  $X_{std}$  representa a variável normalizada, obtida por meio da diferença entre o valor original  $X_b$  e o valor mínimo  $X_{min}$ , dividido pela diferença entre o valor máximo  $X_{max}$  e o valor mínimo  $X_{min}$ .

Uma característica presente nesse método é a preservação da relação de ordem dos dados originais. Isso significa que, após a normalização, a ordem dos valores será mantida, o que se mostra particularmente relevante em problemas nos quais a ordem dos dados possui importância.

Adicionalmente, esse tipo de escalonamento também auxilia a minimizar o problema de variações desproporcionais entre os diferentes recursos do conjunto de dados. Ao reescalar os valores para um intervalo pré-definido, o método garante que todos os recursos possuam uma faixa de valores equivalente, o que é benéfico para algoritmos de aprendizado de máquina sensíveis à escala dos dados. Contudo, é válido ressaltar que essa normalização pode ser tão sensível quanto a técnica descrita na Subseção 2.3.3.1 com relação a presença de *outliers*.

### 2.3.3.3 Escalonamento por Máximo Absoluto

Quando se deve lidar com um *dataset* com ampla gama de magnitudes, um método que também pode ser testado é o escalonamento por máximo absoluto. Apesar de possuir algumas similaridades com o escalonamento por intervalo descrito na Seção 2.3.3.2, ainda assim, foi decidido realizar testes com esse normalizador. Ele é descrito por:

$$X_{std} = X_b / |X_{max}|, \quad (8)$$

em que  $X_{std}$  representa o valor final após a normalização,  $X_b$  é um valor original e  $|X_{max}|$  representa o valor absoluto máximo encontrado nos dados originais.

O método de normalização descrito nesta seção transforma os dados dividindo cada valor pela magnitude máxima absoluta encontrada no conjunto de dados. Essa abordagem, assim como outras, possui a vantagem de preservar a relação de proporção entre os valores, mesmo quando os dados estão em escalas diferentes. Ao

aplicá-lo em conjuntos de dados com grandezas discrepantes, espera-se que a normalização resulte em uma escala uniforme entre zero e um, mantendo as proporções relativas dos valores originais (BUITINCK *et al.*, 2013).

Sua principal vantagem é sua capacidade de preservar as informações originais do conjunto de dados enquanto normaliza as magnitudes. Isso pode ser particularmente benéfico em conjuntos de dados positivos, nos quais a relação entre os valores é fundamental para a interpretação correta dos resultados. Além disso, é um pouco menos sensível a *outliers* em comparação com os métodos citados nas Seções 2.3.3.1 e 2.3.3.2, o que contribui para resultados mais robustos.

## 2.4 APRENDIZADO DE MÁQUINA

Com o objetivo de fornecer uma base sólida para o desenvolvimento do projeto de implementação dos algoritmos de autoencoder, foi de importante o estudo aprofundado dos conceitos de aprendizado de máquina que são apresentados nesta seção. Essas explicações foram importantes, pois para configurar corretamente um autoencoder se faz necessário entender os conceitos básicos para fazer o uso dessas redes na redução de dimensionalidade e reconhecimento de perfis de dados.

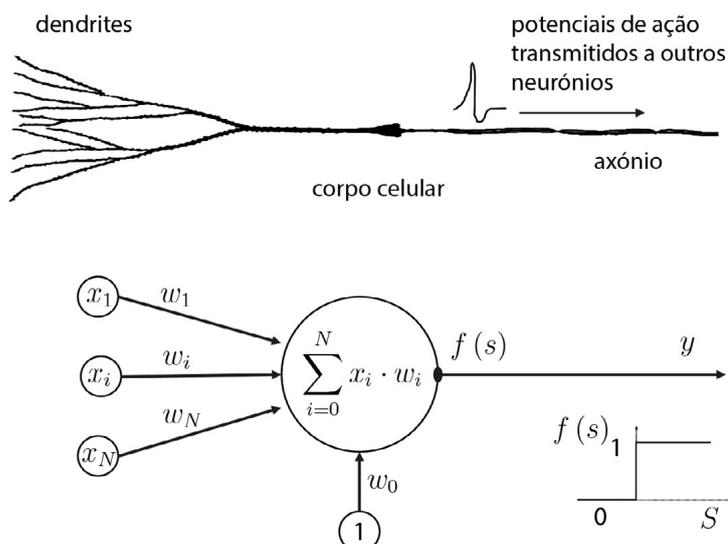
### 2.4.1 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são definidas como um modelo computacional que visa simular mecanismos naturalmente existentes no cérebro de seres vivos. A primeira introdução das RNAs foi feita por McCulloch e Pitts (1943). O elemento fundamental das RNAs é o neurônio artificial, uma unidade de computação simples que possui múltiplas entradas e uma saída. Embora os neurônios artificiais possam variar nas diferentes arquiteturas das RNAs, a ideia central é preservada, assemelhando-se à unidade de limiar linear proposta por Rosenblatt (1957). Essa unidade realiza a soma ponderada das entradas e passa o resultado por uma função de ativação para produzir a saída, conforme ilustrado na Figura 4.

Com base na Figura 5, observa-se que os diversos neurônios artificiais se organizam em paralelo, de modo a formar uma camada. Subsequentemente, essas camadas se conectam em série, constituindo uma RNA.

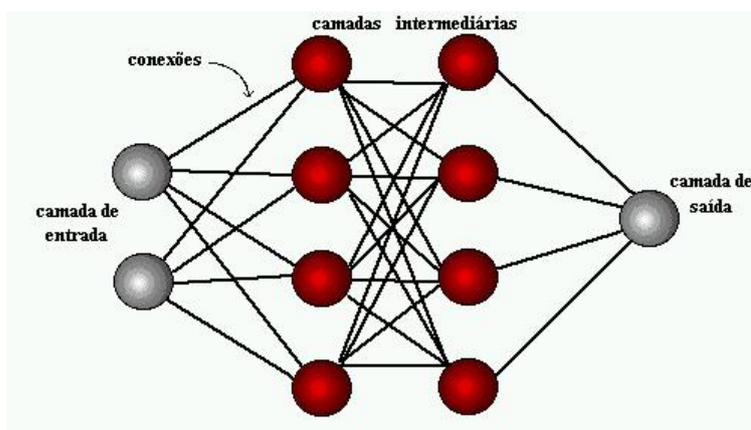
Quando estão em treinamento, os parâmetros da rede, comumente conhecidos como pesos, se ajustam baseando-se na função de perda e, desse modo, a rede pode retornar os resultados esperados em sua implementação. Além dos pesos, há outros parâmetros livres durante o ajuste de uma RNA, mas esses parâmetros de configuração, também conhecidos como hiperparâmetros, tais como número de camadas, taxa de aprendizado, número de épocas, entre outros, são definidos por quem implementa o modelo dependendo da aplicação do mesmo e não durante o processo de treinamento.

Figura 4 – Representação de um neurônio biológico e de uma unidade de limite linear.



Fonte: (AGUIAR, D.; AGUIAR, P., 2022)

Figura 5 – Diagrama simplificado de uma RNA.



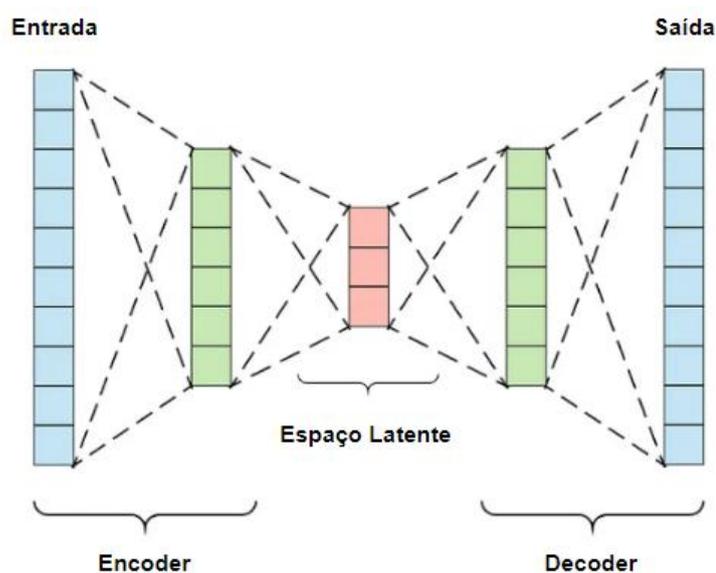
Fonte: (LEON F. DE CARVALHO, s.d.)

### 2.4.2 Autoencoder

Os autoencoders, também conhecidos como redes autoassociativas, se destacam como redes neurais eficientes em funções como extração de *features*, possibilitando que esse tipo de estrutura seja utilizado na criação de mapas de identidade ou replicadores. Os autoencoders são treinados para compreender os perfis mais característicos dos dados de entrada e, dessa forma, retornar uma reconstrução desses dados com base em uma representação com menor dimensionalidade gerada pelo modelo. Esse fato distancia os autoencoders de outras redes neurais, que geralmente são treinadas para predição de classes ou predição de valores. A estrutura de um autoencoder clássico de três camadas envolve camadas de entrada e saída com a mesma dimensão  $n$  e uma camada oculta de dimensão  $m$ , menor que  $n$ , formando

uma rede simétrica e totalmente conectada (HAYKIN, S., 2009). Essa estrutura inclui uma função codificadora e uma função decodificadora, essenciais para a reconstrução dos dados, conforme o esquema da Figura 6. Vale ressaltar que a camada oculta pode ser considerada como um espaço latente, sendo representada de forma simplificada pela Figura 6.

Figura 6 – Esquema de autoencoder simplificado.



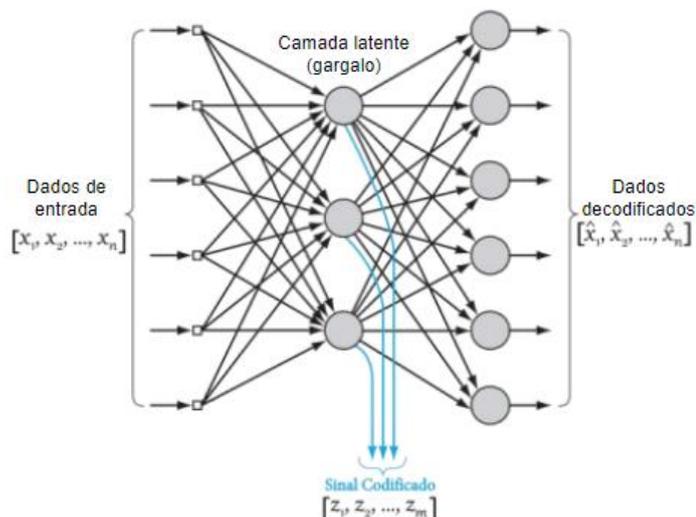
Fonte: adaptado de Stewart (2019)

Os autoencoders são capazes de aprender representações compactas dos dados de entrada, aplicando uma redução de dimensionalidade nas camadas intermediárias. Essa redução de dimensionalidade ocorre durante o processo de reconstrução, no qual os dados são comprimidos e representados de forma mais concisa no espaço latente. Essa representação intermediária de menor dimensão pode ser explorada em várias aplicações, como modelos geradores de dados, extração de variáveis para uso em outros modelos ou detecção de anomalias.

Autoencoders possuem a capacidade de mapear um conjunto de pontos de dados de entrada  $X$  para um conjunto de pontos de dados no espaço latente  $Z$ , onde cada ponto  $z$  representa uma codificação do exemplo de treinamento com dimensionalidade  $m$ . Devido à estrutura do autoencoder, com a mesma quantidade de neurônios nas camadas de entrada e de saída e uma camada oculta de menor dimensão, é possível afirmar que o conjunto de pontos de dados  $Z$  tem uma dimensionalidade menor do que o conjunto de pontos de dados  $X$  e que o conjunto  $Z$  pode reconstruir o conjunto  $X$  (HAYKIN, S., 2009). Esses conjuntos de dados podem ser visualizados no esquema representado pela Figura 7.

Os autoencoders podem ter múltiplos pares de encoders e decoders nas camadas ocultas, desde que sejam observados os requisitos estruturais de simetria e

Figura 7 – Esquema de autoencoder exemplificando um conjunto de dados.



Fonte: adaptado de (HAYKIN, S., 2009)

dimensionalidade (BENGIO *et al.*, 2006). No entanto, a estrutura clássica do autoencoder, conforme representada na Figura 6 e nos requisitos de Simon Haykin (2009), consiste em uma única camada oculta, também conhecida como camada *bottleneck* (BENGIO *et al.*, 2006).

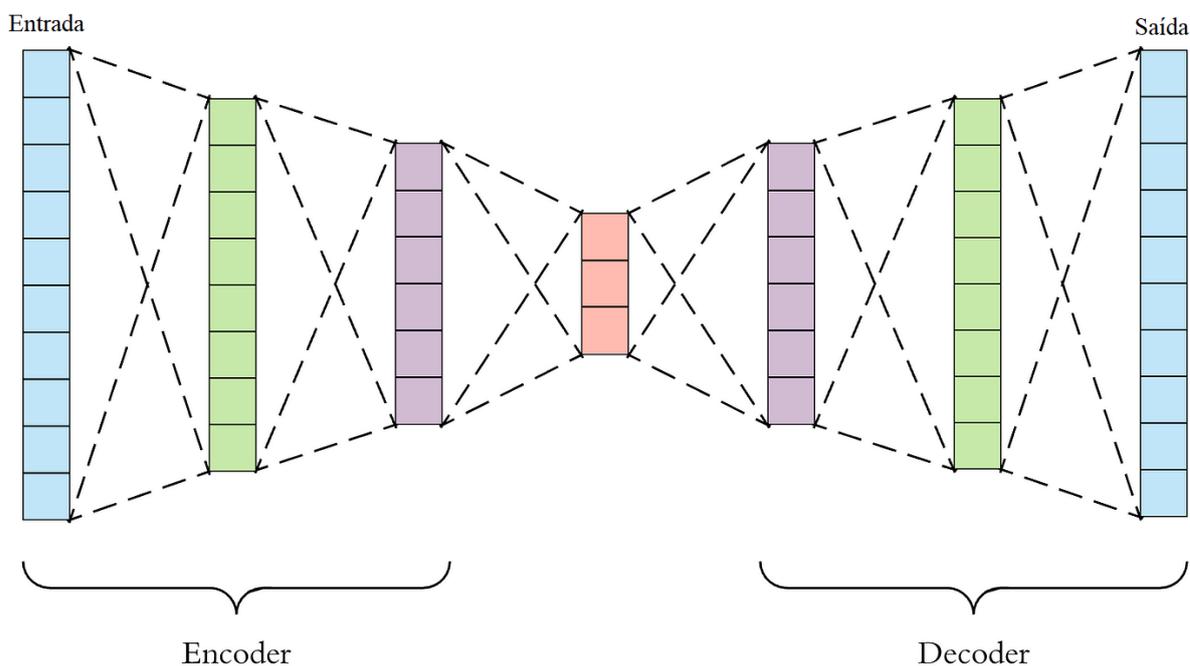
### 2.4.3 Stacked Autoencoders

O tipo de autoencoder utilizado no desenvolvimento do projeto se baseia no conceito de autoencoders “empilhados”, também chamados de *Stacked Autoencoders* (SAE), com arquitetura ilustrada pela Figura 8. Essa é uma técnica de *deep learning* utilizada em RNAs para representação de dados e frequentemente empregada na redução de dimensionalidade (HINTON, G. E.; SALAKHUTDINOV, 2006). A arquitetura dos SAEs é baseada na interconexão de autoencoders, que consistem em um codificador responsável pela codificação e um decodificador responsável pela decodificação (HINTON, G. E.; SALAKHUTDINOV, 2006). O codificador reduz a dimensão dos dados de entrada, enquanto o decodificador restaura a dimensão original dos dados ao processar a saída do codificador.

A diferença principal é que os SAEs podem possuir várias camadas de codificação e decodificação, enquanto os autoencoders genéricos possuem especificamente uma camada de codificação e uma camada de decodificação. Com os SAEs, vários autoencoders são “empilhados” e conectados para formar uma arquitetura mais profunda, tendo mais capacidade de aprender representações mais complexas de dados.

O treinamento dessa estrutura é realizado por meio de um algoritmo conhecido como “pré-treinamento camada a camada” (*layer-wise pretraining*) (BENGIO *et al.*,

Figura 8 – Diagrama estrutural de um SAE.



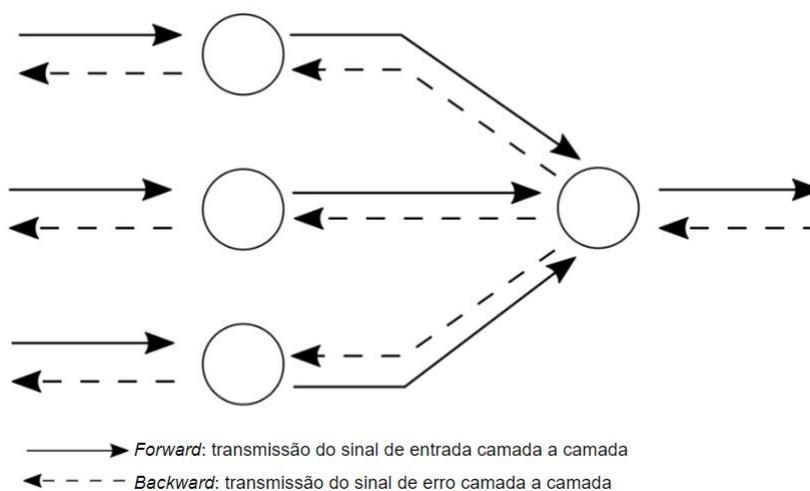
Fonte: adaptado de Stewart (2019)

2006), que é classificado como não supervisionado. Esse treinamento ocorre em cada camada do autoencoder. Portanto, o treinamento de cada camada visa minimizar a função de perda, o que tipicamente é feito utilizando o método do gradiente descendente.

O treinamento começa com o primeiro autoencoder, que, após ser treinado, tem seus pesos fixados e tem a saída de seu codificador usada como entrada para treinar o próximo autoencoder. Esse procedimento se repete até a última camada. Em seguida, é possível realizar um ajuste fino, que consiste em ajustar os pesos de uma rede neural inicializada com os pesos do SAE treinado, utilizando métricas alvo de desempenho determinadas e aplicando a retropropagação do erro de reconstrução na saída do decoder, também conhecida pelo termo em inglês *backpropagation* (LECUN; BENGIO; HINTON, G., 2015). Esse fluxo de alimentação de informações implementado pelo método é representado pelo diagrama demonstrado na Figura 9.

A utilização do SAE se motivou pelo fato de que, embora a retropropagação do erro também seja aplicada em redes neurais rasas, ao construir redes mais profundas observa-se uma convergência mais rápida dos pesos nas camadas próximas à saída em comparação com as camadas próximas à entrada. O pré-treinamento camada a camada tem como objetivo principal inicializar os pesos da rede neural com valores que facilitem a minimização do erro, além de extrair características importantes do conjunto de dados. O Quadro 2 apresenta um resumo do algoritmo de treinamento de um SAE.

É importante ressaltar que para realizar o treinamento de um autoencoder, é ne-

Figura 9 – Algoritmo de *backpropagation*.

Fonte: Adaptado Simon S. Haykin e Van Veen (2001)

Quadro 2 – Treinamento padrão de um autoencoder SAE com *backpropagation*.

Início
<b>Entrada do Algoritmo</b>
Conjunto de pontos de dados pré-processados, no qual cada ponto $x$ (que representa um exemplo de treinamento) com $u$ dimensões.
<b>Etapa 1: Transmissão do Sinal (<i>Forward</i>)</b>
Para cada ponto de entrada $x$ é transmitido o sinal para computar ativações em todas as camadas ocultas e na camada de saída para obter uma saída $\hat{x}$ .
<b>Etapa 2: Cálculo do Erro de Reconstrução</b>
Calcular o erro de reconstrução para cada $\hat{x}$ em relação a entrada $x$ utilizando a função de perda de informação definida como métrica.
<b>Etapa 3: Retropropagação do Erro de Reconstrução (<i>Backward</i>)</b>
Retropropagar o erro através das camadas da rede e realizar atualizações nos pesos.
<b>Etapa 4: Iteração</b>
Repete as etapas 1, 2 e 3, até que um determinado indicador seja satisfeito.
<b>Saída do Algoritmo</b>
Valores de parâmetros da rede (pesos e bias) são ajustados.
Fim

Fonte: adaptado de Simon S. Haykin e Van Veen (2001)

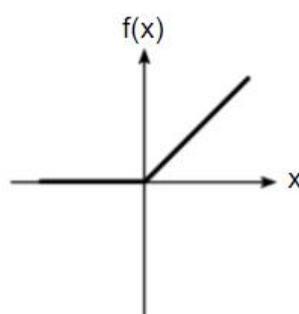
cessário empregar um algoritmo de otimização de ajustes de pesos, que é combinado com o algoritmo de retropropagação, visando calcular e atualizar os conjuntos relativos aos vieses da rede. Esses algoritmos são necessários para ajustar os parâmetros do autoencoder durante o processo de aprendizado, permitindo que a rede se adapte aos dados e otimize seu desempenho.

#### 2.4.3.1 Funções de Ativação

As funções de ativação desempenham um papel importante nas camadas de um autoencoder, pois são responsáveis por introduzir não linearidades nas transformações dos dados. Isso permite que as camadas do autoencoder tenham a capacidade de gerar representações mais complexas, detectando relações não lineares nos dados de entrada (AGGARWAL, 2018). A seguir são citadas as funções que foram mais utilizadas em testes no desenvolvimento do projeto.

- ReLU (*Rectified Linear Unit*): uma função de ativação amplamente utilizada, definida como  $f(x) = \max(0, x)$ . A função ReLU ativa os valores positivos, mantendo-os inalterados, e define os valores negativos como zero. Essa função de ativação é utilizada devido à sua simplicidade computacional e à sua capacidade de mitigar o problema do gradiente desaparecente, que ocorre quando há uma diminuição significativa do valor do gradiente em algoritmos de retropropagação em redes neurais profundas. O gráfico apresentado na Figura 10 mostra o perfil desse tipo de função de ativação.

Figura 10 – Função de ativação ReLU.

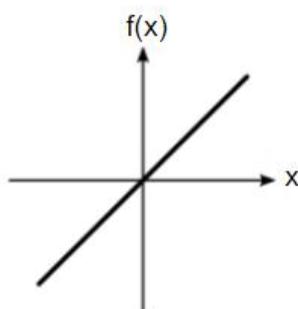


Fonte: autor

- Linear: também conhecida como identidade, onde  $f(x) = x$ . Essa função de ativação simplesmente mantém os valores de entrada inalterados, sem introduzir não linearidade. A função linear é utilizada principalmente na camada de saída de um autoencoder quando se deseja obter uma reconstrução linear dos dados. O

gráfico apresentado na Figura 11 mostra o comportamento desse tipo de função de ativação.

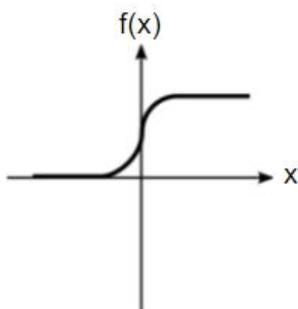
Figura 11 – Função de ativação Linear.



Fonte: autor

- Sigmoidal, ou função logística: é definida como  $f(x) = \frac{1}{1+e^{-x}}$ . Essa função mapeia os valores de entrada para o intervalo entre zero e um, o que a torna adequada para problemas de classificação binária. No contexto do autoencoder, a função sigmoide é frequentemente utilizada nas camadas ocultas para introduzir não linearidades e permitir a aprendizagem de representações mais complexas dos dados. O gráfico apresentado na Figura 12 mostra o perfil desse tipo de função de ativação.

Figura 12 – Função de ativação Sigmoide.



Fonte: autor

#### 2.4.4 Seleção de *Features*

A seleção de características, também conhecida como seleção de *features*, é uma etapa importante no campo da aprendizagem de máquina e análise de dados. Seu objetivo é identificar as características mais relevantes e informativas para um determinado problema. Diversos métodos e abordagens existem para realizar essa se-

leção, mas nessa seção será detalhado apenas o método utilizado no projeto (GUYON; ELISSEEFF, 2003), chamado de seleção de *features* baseada na ponderação.

Uma abordagem comumente utilizada para seleção de *features* é o método intrínseco, também conhecido como método incorporado. Os métodos incorporados integram a seleção de características diretamente no processo de treinamento do modelo. Eles exploram as características e sua relação com a variável de destino durante a etapa de construção do modelo, incorporando assim a seleção de *features* como parte integrante do processo (CHANDRASHEKAR; SAHIN, 2014).

Dentre os métodos intrínsecos, também chamados de *embedded*, um que merece destaque é o método da importância das características baseado na ponderação. Esse método atribui um peso ou uma pontuação de importância a cada característica com base na sua contribuição para o desempenho do modelo. Essa contribuição é geralmente avaliada por meio da análise dos coeficientes ou pesos associados às características no modelo. Características com pesos mais altos são consideradas mais relevantes e têm maior influência no processo de aprendizado do modelo.

Esse método fornece uma medida quantitativa da relevância de cada característica, permitindo uma seleção eficiente das *features* mais relevantes. Ao selecionar as características com pesos mais altos, o método visa melhorar o desempenho do modelo, reduzir a dimensionalidade do espaço de características e aumentar a interpretabilidade dos resultados. Isso ajuda a identificar as características mais informativas e descartar as menos relevantes, contribuindo para uma melhor compreensão do problema e reduzindo a complexidade computacional (LIU; YU, 2005).

### 3 DESENVOLVIMENTO

Este capítulo tem como objetivo descrever detalhadamente os passos utilizados para a implementação e treinamento do autoencoder, desde o pré-processamento dos dados até a codificação da solução completa. Na Seção 3.1, é explicada a abordagem para tratar o conjunto de dados bruto, incluindo a padronização e organização para obter uma melhor eficácia aos dados tratados no autoencoder. Já na Seção 3.2, são apresentados de forma mais específica os requisitos e os ajustes de parâmetros utilizados para o funcionamento do algoritmo que serve de base para a redução de dimensionalidade do projeto, bem como os resultados dos testes preliminares realizados.

#### 3.1 PRÉ-PROCESSAMENTO DE DADOS

O pré-processamento dos dados foi realizado buscando selecionar parâmetros e métodos que melhorem o desempenho do autoencoder para redução de dimensionalidade. Um conjunto de dados padronizado contribui de maneira significativa para as rotinas de treinamento do autoencoder (IOFFE; SZEGEDY, 2015), tanto ao considerar um conjunto limitado e aleatório de ensaios (testes preliminares), quanto ao executar treinos com uma versão ajustada do algoritmo e utilizando bases de dados completas como entrada.

Nesta seção, são relatadas de forma mais detalhada as etapas relevantes na organização do conjunto de dados bruto. Na Subseção 3.1.1 são explicados a origem dos dados utilizados e o processo de familiarização com o *dataset*. Na Subseção 3.1.2 é demonstrada a aplicação de algoritmos de padronização dos dados, levando em consideração os conceitos descritos na Subseção 2.3. Por fim, na Subseção 3.1.3, são descritos os experimentos realizados com o intuito de definir o método de normalização a ser utilizado nos treinamentos.

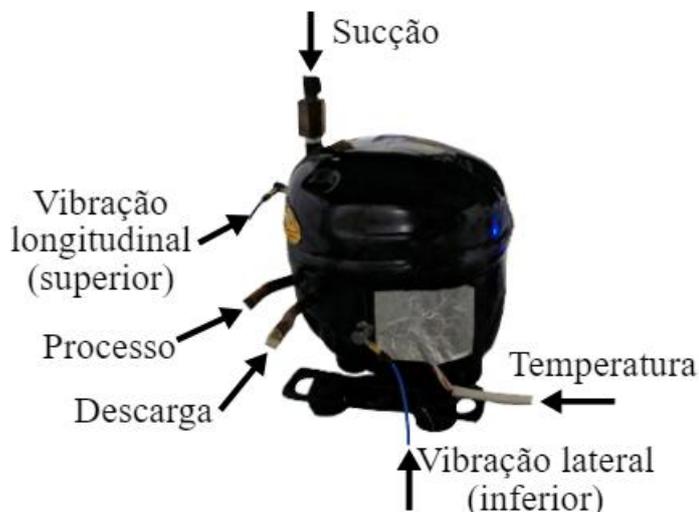
##### 3.1.1 Aquisição de Dados

No projeto de bancada experimental conduzido por Thaler (2021), foram realizados ensaios de amaciamento para examinar o desempenho de dois modelos de compressores com capacidade de refrigeração fixa: o Modelo A e o Modelo B. A Figura 13 apresenta como foram instrumentados os compressores ensaiados.

Cada modelo de compressor passou por no mínimo dois ensaios de duração pré-definida. O primeiro ensaio foi dedicado exclusivamente ao amaciamento da unidade, enquanto que os ensaios subsequentes foram realizados após o amaciamento ter ocorrido. A duração dos ensaios foi estipulada com base em dados experimentais, que demonstraram que o modelo A requer um período de 24 h para um amaciamento

seguro, enquanto que o modelo B demanda um tempo mais longo de 40 h. A Quadro 3 especifica a quantidade de dados de vibração adquiridos.

Figura 13 – Instrumentação utilizada nos compressores testados.



Fonte: (THALER *et al.*, 2021)

Quadro 3 – Quantidade de arquivos de medição de vibração.

Unidades	Nº de medições	Total por tipo	Total de medições
Tipo A (amaciado)	18446	30499	131649
Tipo A (não amaciado)	12053		
Tipo B (amaciado)	87657	101150	
Tipo B (não amaciado)	13493		

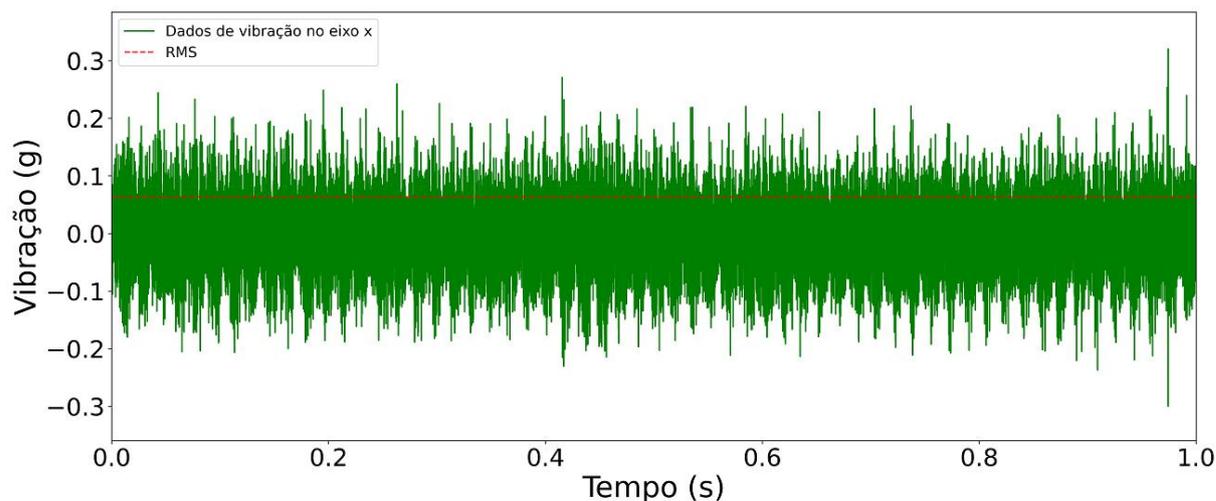
Fonte: autor

A coleta de dados de vibração foi realizada em blocos, utilizando uma taxa de aquisição dada por  $f_s = 25.6$  kHz. Em cada minuto de ensaio, um segundo de dados de vibração é adquirido e armazenado em um arquivo de texto, permitindo uma análise mais precisa do comportamento das unidades durante o ensaio. Para exemplificar o perfil dos dados de cada arquivo, a Figura 14 representa um segundo de medição de vibração de uma das unidades do modelo A depois de já ter sido amaciada.

No total, foram avaliadas quatro unidades do modelo A, identificadas como Unidade A1 a Unidade A4, e onze unidades do modelo B, denominadas com o mesmo padrão sequencial (Unidade B3, por exemplo), seguindo a ordem dos ensaios realizados explicada no parágrafo anterior.

No contexto específico dos dados de vibração, esses foram adquiridos utilizando acelerômetros posicionados no corpo dos compressores, durante os ensaios mencionados anteriormente. Essas medições permitiram a extração de dados bidimensionais, medidos no eixo lateral e longitudinal do compressor. Na Figura 15, são demonstrados dois exemplos de perfis de vibração no eixo longitudinal para ensaios completos.

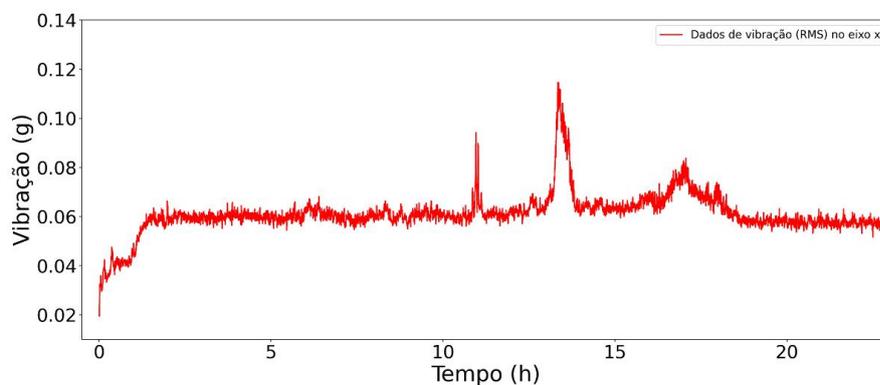
Figura 14 – Exemplo de uma medição de dados de vibração no eixo longitudinal.



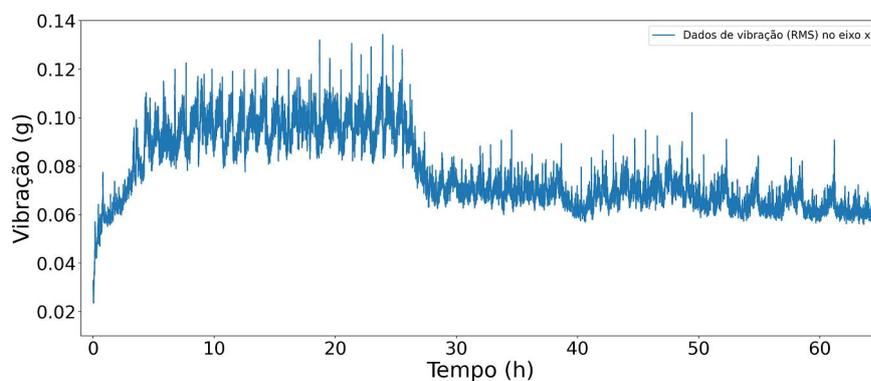
Fonte: autor

Figura 15 – Dados eficazes de vibração em ensaios completos.

(a) Unidade não amaciada.



(b) Unidade amaciada.



Fonte: autor

Os dados obtidos foram armazenados em arquivos de texto no formato “.dat”. Ter familiaridade com a formatação padrão do conjunto de dados foi importante para definir estratégias de conversão desses valores, originalmente em arquivos de texto, para estruturas de dados manipuláveis, a fim de organizar os dados de forma padronizada e garantir que essa etapa não prejudique o progresso do projeto. É importante comentar que para o pré-processamento desses dados foram consideradas apenas as medições referentes ao eixo longitudinal.

### 3.1.2 Padronização de Dados

Com o objetivo de organizar de forma padronizada o conjunto de dados de vibração adquiridos, conforme explicado em 3.1.1, foi desenvolvido um algoritmo que reestrutura a formatação desses dados. O objetivo dessa reestruturação é facilitar a aplicação deles nos treinamentos do autoencoder, visando a compreensão dos perfis intrínsecos ao processo de amaciamento de compressores, com base em medições realizadas por meio dos acelerômetros. O código elaborado para essa função é explicado sucintamente no Quadro 4.

Quadro 4 – Funcionamento do algoritmo de tratamento de dados.

Início
<b>Etapa 1: Definição do número de ensaios</b>
- define-se o diretório onde todos os arquivos de dados de vibração estão localizados; - deve-se determinar um número arquivos de medição a serem escolhidos aleatoriamente; - é criada uma lista de strings contendo os diretórios de cada arquivo escolhido.
<b>Etapa 2: Conversão de formatação de dados</b>
- utiliza-se um loop para iterar sobre a lista gerada na etapa 1, lendo os arquivos selecionados um por um; - os arquivos de texto são processados para extrair colunas contendo medições de vibração nos dois eixos do compressor.
<b>Etapa 3: Organização de dados</b>
- adiciona uma coluna com o índice do arquivo original a cada linha; - dados são concatenados em uma lista, contendo 4 colunas.
<b>Etapa 4: Processamento dos dados em bandas de potência</b>
- é criada a rotina que calcula a banda de potência entre cada faixa de frequência contida de 0 Hz a 12 800 Hz; - a lista criada na etapa 3 é iterada calculando as bandas de potência para cada ensaio utilizando o índice, determinando o passo para dividir uniformemente a frequência de amostragem.
<b>Etapa 5: Armazenamento dos Dados Tratados</b>
- é criada uma lista contendo as bandas de frequência para cada ensaio de vibração no eixo longitudinal; - cada linha dessa lista contém dados dessa natureza referentes a ensaios diferentes.
Fim

Fonte: autor

Um detalhamento do passo a passo apresentado pelo Quadro 4 é descrito abaixo.

**Etapa 1:** Nessa rotina, o diretório no qual se localizam todos os dados dos ensaios é especificado. Uma lista de *strings* é criada mapeando o caminho até todos os

dados de ensaios de vibração. Esses processos são feitos usando um dicionário criado para organizar os dados dos ensaios, separando-os entre modelos de compressor, ensaios amaciados e não amaciados, desconsiderando arquivos em que o modelo de compressor testado se encontrava desligado e armazenando o diretório de cada lista de medições salvas em formato “.dat”. Para finalizar essa etapa, deve-se definir um número de arquivos de medição a serem tratados, esses são escolhidos aleatoriamente utilizando como base a lista de todos os arquivos referentes aos dados coletados pelos acelerômetros.

**Etapa 2:** Nesse trecho, um laço *for* percorre a lista com os diretórios dos  $n$  ensaios aleatórios abrindo os diretórios desses arquivos, separadamente, a cada iteração. Dentro do laço, os arquivos são lidos e convertidos para um formato padronizado, transformando as variáveis presentes nos arquivos de texto em variáveis numéricas.

**Etapa 3:** O índice  $i$ , baseado na ordem de leitura dos dados na etapa 2, é adicionado a cada linha de dados usando uma compreensão de lista, fornecendo uma referência às linhas de medições longitudinais extraídas de cada arquivo original do ensaio.

**Etapa 4:** A função criada `bandpower` recebe um sinal  $x$ , uma taxa de amostragem  $f_s$ , uma frequência mínima  $f_{min}$  e uma frequência máxima  $f_{max}$ . Ela calcula o periodograma do sinal, obtendo as frequências  $f$  e a densidade espectral de potência  $p_{xx}$ . Em seguida, encontra os índices correspondentes às frequências mínima e máxima e calcula a área sob a curva do periodograma entre essas frequências, retornando assim a potência de banda.

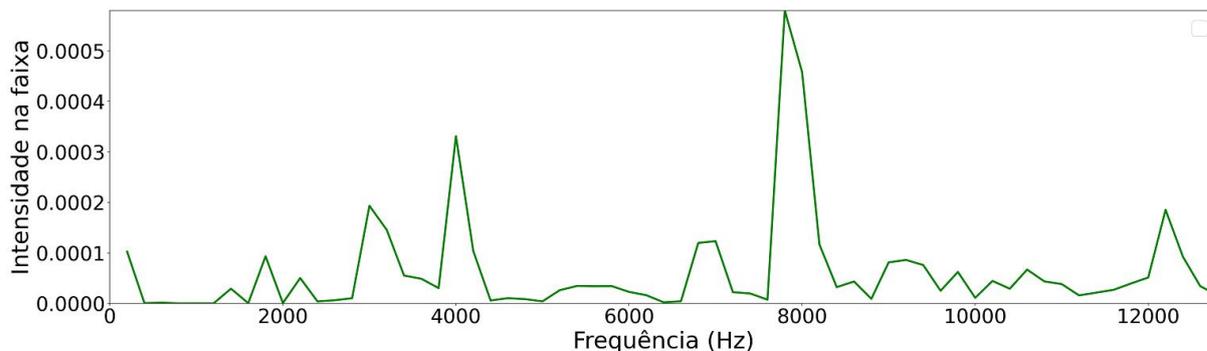
Foi definida uma função que recebe a lista criada na etapa 3, uma taxa de amostragem  $f_s$  e um parâmetro arbitral  $n$ . Para cada grupo de dados, a função extrai um subconjunto de dados que corresponde a um determinado ensaio. Em seguida, calcula a potência de banda para várias faixas de frequência utilizando a função criada na etapa anterior e  $n$  como valor que define o passo uniforme entre frequência mínima e máxima de cada banda de potência.

**Etapa 5:** Uma lista é gerada para armazenar as bandas de frequência correspondentes a cada ensaio de vibração no eixo  $x$ . Cada linha dessa lista contém dados relacionados a diferentes ensaios, representando as informações específicas das bandas de frequência para cada um deles. Essa estrutura organizada permite a análise e comparação dos resultados de diferentes ensaios de vibração, facilitando a interpretação do algoritmo de redução de dimensionalidade.

Para exemplificar o perfil do resultado do tratamento das medições de vibração utilizando o algoritmo descrito acima, são mostradas as Figuras 16, 17 e 18. Cada

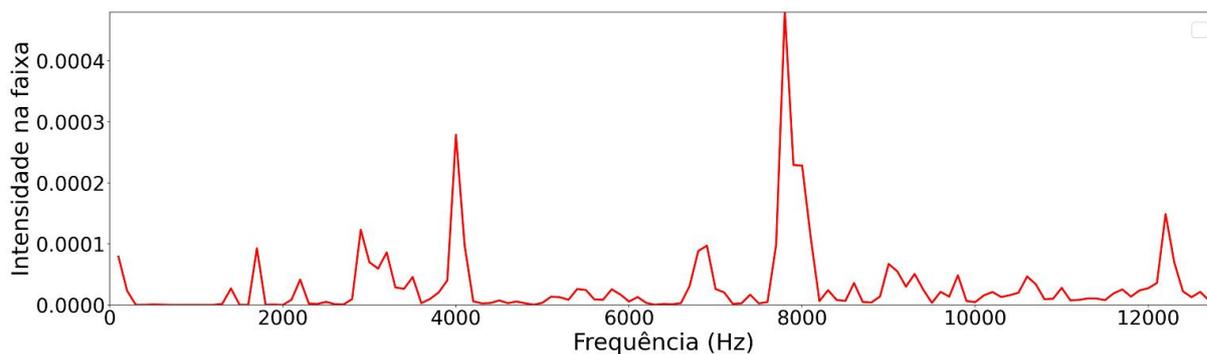
Figura representa um arquivo de medição, tratados com intervalos de frequência com passo uniforme de 200 Hz (total de 64 bandas), 100 Hz (total de 128 bandas) e 50 Hz (total de 256 bandas), respectivamente.

Figura 16 – Perfil de um arquivo de medição em 64 bandas de potência.



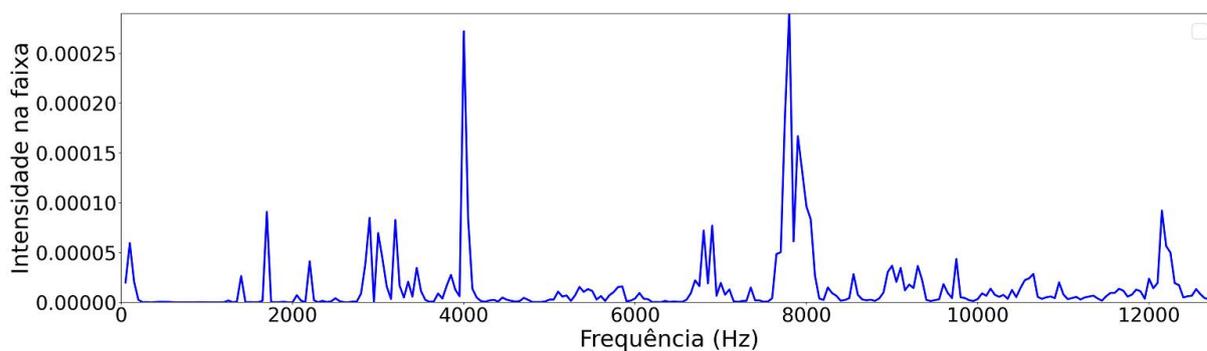
Fonte: autor

Figura 17 – Perfil de um arquivo de medição em 128 bandas de potência.



Fonte: autor

Figura 18 – Perfil de um arquivo de medição em 256 bandas de potência.



Fonte: autor

Seguindo a abordagem proposta para a preparação dos dados de entrada do autoencoder, o próximo passo importante é avaliar e selecionar uma técnica de normalização adequada para o perfil específico do conjunto de dados.

### 3.1.3 Normalização

Após finalizar a organização dos dados em bandas de potência, a normalização torna-se a última etapa do pré-processamento dos dados de vibração antes de utilizá-los como entrada. Foram feitos testes iniciais utilizando diferentes métodos de normalização disponíveis na biblioteca (PEDREGOSA *et al.*, 2011), visando entender qual técnica se adaptava melhor para as características do *dataset*. Seguem explicações acerca de cada um dos métodos experimentados, explicitando também análises breves que levaram à melhor escolha de normalizador para o posterior uso desses dados pré-processados no autoencoder:

- 1: Durante a avaliação do funcionamento do escalonamento padrão, constatou-se que os *outliers* tiveram um impacto significativo no desempenho do modelo, resultando em uma exatidão comprometida do autoencoder. Isso ocorreu devido à atribuição de um peso maior aos dados com características de *outliers* do que deveria ter sido atribuído, distorcendo o processo de aprendizado.
- 2: No decorrer da aplicação do escalonamento min-max, observou-se uma melhoria no desempenho do autoencoder em comparação à técnica descrita na Seção 2.3.3.1. No entanto, à medida que a quantidade de dados pré-processados aumentava, tornava-se evidente um problema relacionado aos *outliers*.
- 3: O escalonamento por máximo absoluto foi o mais eficaz experimentado para normalizar os dados organizados em bandas de frequência. Sua capacidade de preservar as relações proporcionais entre os valores originais e sua robustez em relação a *outliers* são atributos que fizeram com que essa técnica trouxesse melhores resultados em análises estatísticas e no algoritmo do autoencoder.

Foram definidas algumas métricas para a escolha do método mais compatível ao *dataset* utilizado. A partir da Subseção 2.3.3, foram elencadas características com relação à suavização de *outliers* e preservação da distribuição para cada um dos métodos, correlacionando essas características com os valores mínimos da função perda alcançados com cada uma delas. Ou seja, a adaptabilidade do *dataset* tem conexão direta com o valor da função de perda.

Com base nos treinamentos realizados com as técnicas descritas anteriormente, foi decidido manter a utilização do escalonamento por máximo absoluto para os testes mais aprofundados na implementação e nos ajustes do algoritmo do autoencoder. A fim de fornecer uma visão mais ilustrativa, é apresentado a seguir o Quadro 5, que representa um comparativo das três técnicas explicadas anteriormente com características referentes à natureza do conjunto de dados de vibração e o erro relativo médio na reconstrução dos dados atingido por cada uma delas.

Quadro 5 – Comparação entre métodos de normalização.

	Preservação da distribuição	Suavização de <i>outliers</i>	Adaptabilidade ao <i>dataset</i>	Função de perda
Escalonamento padrão	Regular	Ruim	Ruim	95%
Escalonamento min-max	Ruim	Ruim	Regular	90%
Escalonamento por máximo absoluto	Bom	Regular	Bom	20%

Fonte: autor

Considerando as informações do Quadro 5, observa-se que a suavização de *outliers* possui impacto similar na adaptabilidade aos dados com relação à preservação da distribuição original desses. Dessa forma, orientando-se também pela métrica do erro relativo, pode-se escolher o normalizador mais adequado para atuar nesse tipo de *dataset*. Durante os treinamentos iniciais do algoritmo, utilizando métodos mais sensíveis a pontos discrepantes, constatou-se que o processamento adequado desses dados extremos é um dos aspectos mais críticos. Mas observou-se também, que por esses dados apresentarem uma grande variabilidade de magnitudes entre seus valores, a distribuição desses se prova uma característica ainda mais importante.

## 3.2 IMPLEMENTAÇÃO

Nesta seção são exploradas em maior detalhe as etapas de desenvolvimento que foram executadas para definir e projetar o autoencoder. A discussão inicia pela Seção 2.4.3, na qual é abordada a estrutura relacionada ao tipo de autoencoder selecionado para resolver o problema em questão. Em seguida, na Seção 3.2.2, é discutido o processo de seleção dos hiperparâmetros do modelo. Por fim, na Seção 3.2.3, é descrita a rotina de testes do modelo, com o objetivo de avaliar e embasar a escolha dos hiperparâmetros e da técnica de normalização mais adequada, demonstrando resultados experimentais.

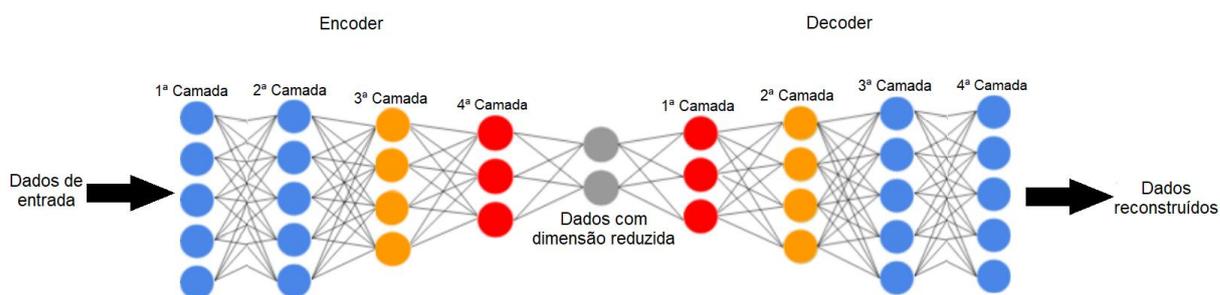
### 3.2.1 Estrutura do SAE

Foram avaliadas diversas estruturas de autoencoder baseando-se na Subseção 2.4.3, mas o modelo elaborado que obteve os melhores resultados tem sua estrutura apresentada na Figura 19.

Outra definição importante para o funcionamento do autoencoder é a da função de perda. Nesse caso, foi empregado o erro percentual absoluto médio (MAPE, do inglês *Mean Absolute Percentage Error*), pois considerando a aplicação do modelo desenvolvido, foi a métrica mais próxima do erro relativo de reconstrução dos dados de entrada. Dessa forma, ela foi utilizada no algoritmo de *backpropagation* para ajustes dos parâmetros da rede. A expressão para a função de perda é dada por:

$$MAPE = \frac{1}{r} \sum_{i=1}^r \frac{100}{n_e} \sum_{j=1}^{n_e} \left| \frac{x_{ij} - \hat{x}_{ij}}{x_{ij}} \right|, \quad (9)$$

Figura 19 – Diagrama simplificado do SAE implementado.



Fonte: autor

na qual a variável  $r$  representa o número de exemplos utilizados no treinamento,  $n_e$  indica a quantidade de atributos presentes nas amostras utilizadas para treinamento,  $x_i$  refere-se a uma amostra de entrada,  $x_{ij}$  representa um dos atributos específicos de  $x_i$ ,  $\hat{x}_i$  representa uma reconstrução da amostra de entrada  $x_i$  e  $\hat{x}_{ij}$  é um dos atributos presentes na reconstrução de  $\hat{x}_i$ .

Quanto aos conjuntos de treinamento, validação e teste, no contexto dos treinamentos preliminares, a separação foi determinada em 20% dos dados de entrada com o objetivo de avaliar o desempenho dos hiperparâmetros escolhidos diante de um conjunto independente em relação aos dados de treinamento do modelo. Com relação aos treinamentos finais, foi definido que 20% do total das medições seriam separadas para o conjunto de testes e que com os arquivos restantes seria feita a divisão feita na mesma proporção dos testes preliminares.

Para analisar o desempenho do modelo nos treinamentos preliminares, foi utilizada a validação cruzada. Nessa abordagem, foi considerado que o conjunto de dados é dividido em dois subconjuntos principais: o conjunto de treinamento e o conjunto de validação. Os desempenhos dos dois são comparados utilizando as curvas referentes à função perda para cada um desses subconjuntos. Já nos treinamentos finais, a única diferença é que o conjunto de dados de medição foi dividido em três subconjuntos.

Considerando os pesos atribuídos pelo modelo no processo de treinamento e utilizando como base os treinamentos finais detalhados no Capítulo 4, foram feitas análises referentes à significância dos pesos escolhidos para formar o espaço latente. Esses foram ordenados, em ordem decrescente, e foi utilizada a metade desses pesos que possuía os maiores valores. Para avaliar a importância que as bandas de frequência possuem com relação aos pesos contidos no espaço latente, foram consideradas as bandas que originaram esses pesos escolhidos.

É importante ressaltar que devido a limitação computacional, os treinamentos contidos neste trabalho foram feitos exclusivamente com dados de entrada originários das medições realizadas com compressores de modelo A. Foi tomada essa decisão pelo fato de a quantidade de arquivos referentes ao modelo B equivalerem a mais que

o triplo do que havia disponível em compressores de modelo A. Dessa forma, essa quantidade foi considerado suficiente para explorar a utilização de autoencoders em dados dessa natureza.

Na Subseção 3.2.2, são especificadas as definições dos parâmetros de configuração do modelo implementado. É importante ter cautela ao definir alguns hiperparâmetros, pois, dependendo da configuração, o modelo pode sofrer de *overfitting*. Isso ocorre quando um modelo se ajusta bem aos dados de treinamento, mas tem dificuldade em generalizar para novos dados não vistos anteriormente.

### 3.2.2 Definição de Hiperparâmetros

Tendo sido escolhido o tipo de autoencoder utilizado no projeto, deve-se definir a arquitetura e os parâmetros de configuração do mesmo. Foram realizadas variações na quantidade de neurônios por camada, considerando a dimensão dos dados de entrada e utilizando os testes realizados e descritos na Seção 3.2.3. Além disso, ao encontrar um número adequado de neurônios em cada camada, determinam-se hiperparâmetros para o desempenho do autoencoder.

É relevante destacar que um algoritmo de otimização aleatória foi desenvolvido para auxiliar na escolha do tamanho do lote e na taxa de aprendizado do modelo. Esse algoritmo testou várias combinações de valores para determinar qual delas proporcionaria o melhor desempenho dentro de uma faixa definida por experimentos críticos baseados nos valores iniciais contidos nas funções da biblioteca Keras (s.d.).

Para realizar a escolha de quase todos os parâmetros de configuração, com exceção da taxa de aprendizado e tamanho de lote, foi utilizado o ajuste fino. Esse procedimento envolve a modificação manual desses hiperparâmetros do modelo com base na compreensão do problema específico. Ele permite refinar o desempenho do modelo, explorando configurações específicas que podem não ter sido consideradas na busca automatizada.

Foram realizados testes preliminares para a definição dos parâmetros de configuração escolhidos, considerando apenas os experimentos bem-sucedidos apresentados na Seção 3.2.3, os hiperparâmetros selecionados são citados a seguir:

- número de camadas: quatro camadas para o encoder e quatro camadas para o decoder;
- número de neurônios por camada: o número de neurônios em cada camada foi definido em relação à entrada, e os valores para o parâmetro são apresentados no Quadro 6 para cada uma das três divisões de banda avaliadas;
- Função de ativação: após testes das funções descritas na Seção 2.4.3.1, observou-se que a combinação que melhor se adaptou ao encoder foi ReLU-ReLU-ReLU-

ReLU, enquanto que para o decoder foi ReLU-ReLU-ReLU-Sigmoide que apresentou os melhores resultados;

- Otimizador: o método selecionado foi o Adam (*Adaptive moment estimation*), pois esse algoritmo obteve bons resultados nos experimentos mais bem-sucedidos e também utiliza o conceito de *backpropagation* para atualização dos pesos;
- Taxa de aprendizado: definida ao configurar a rotina do algoritmo de otimização aplicado no modelo, assumindo valores entre 0,00004 e 0,0006 na implementação deste projeto;
- Tamanho do lote (*batch size*): também fixado ao configurar parâmetros do otimizador, foi definido em 100;
- Número de épocas: o número de épocas é determinado durante o treinamento do autoencoder e declarado na rotina do otimizador. O valor padronizado foi 20000 épocas, para avaliar o rendimento independentemente da quantidade de bandas selecionadas no pré-processamento dos dados.

Quadro 6 – Número de neurônios em cada camada do autoencoder (considerando número de bandas de frequência).

Nº de entradas	Encoder				Decoder			
	Primeira	Segunda	Terceira	Quarta	Primeira	Segunda	Terceira	Quarta
Bandas de frequência (64)	64	64	48	32	32	48	64	64
Bandas de frequência (128)	128	128	96	64	64	96	128	128
Bandas de frequência (256)	256	256	192	128	128	192	256	256

Fonte: autor

### 3.2.3 Análise de Desempenho Preliminar

Foram realizados diversos testes com o objetivo de consolidar a escolha dos hiperparâmetros e da arquitetura do autoencoder. A maioria desses testes envolveu a definição manual dos parâmetros, utilizando informações disponíveis na literatura citadas na Seção 2.4.2. Nesta seção, são apresentados os resultados dos testes mais bem-sucedidos, considerando um conjunto de dados menor, composto por 2000 medições de dados de vibração.

Com o objetivo de facilitar a compreensão de cada parte do código desenvolvido, o Quadro 7 apresenta um resumo da funcionalidade da metodologia utilizada nos treinamentos descritos nas Seções 3.2.3.1, 3.2.3.2 e 3.2.3.3. Isso foi feito visando definir variáveis importantes na elaboração e na adaptação do autoencoder a um conjunto extenso de dados de medição.

Quadro 7 – Resumo da rotina de treinamento do autoencoder implementado.

Início
<p><b>Etapa 1: Inicialização</b></p> <ul style="list-style-type: none"> <li>- considerando que não há informações prévias disponíveis, os pesos sinápticos e os limiares (bias) são inicializados com valores aleatórios seguindo uma distribuição uniforme.</li> </ul>
<p><b>Etapa 2: Definição dos Dados de Entrada</b></p> <ul style="list-style-type: none"> <li>- é fornecido à rede, para cada um dos treinamentos, um conjunto de dados com dimensão <math>2000 \times 64</math>, <math>2000 \times 128</math> ou <math>2000 \times 256</math>;</li> <li>- os dados de entrada são separados em treinamento (80%) e validação (20%);</li> <li>- para cada exemplo de treinamento, as etapas 3 e 4 são executadas.</li> </ul>
<p><b>Etapa 3: Treinamento</b></p> <ul style="list-style-type: none"> <li>- os potenciais de ativação e os sinais funcionais de saída são calculados para cada neurônio, camada por camada, utilizando a função de ativação correspondente;</li> <li>- na camada de saída, o sinal de erro (MAPE) é calculado considerando os valores originais e uma predição do modelo a partir da versão reduzida parcial;</li> <li>- os gradientes locais são calculados para a camada de saída e para todas as outras camadas;</li> <li>- realiza-se o cálculo do ajuste dos pesos sinápticos e limiares da rede;</li> <li>- os pesos sinápticos e limiares das redes são atualizados, camada por camada;</li> <li>- a etapa 3 é repetida, iterando sobre o conjunto de 2000 ensaios, até que a rede alcance as 20000 épocas definidas;</li> <li>- saída do treinamento, valores dos componentes da rede (pesos e bias) ajustados.</li> </ul>
<p><b>Etapa 4: Reconstrução</b></p> <ul style="list-style-type: none"> <li>- após a última época, é gerada a versão reduzida final do conjunto de dados de entrada considerando as <i>features</i> selecionadas a partir dos pesos ajustados;</li> <li>- para avaliar o desempenho do autoencoder, reconstrói-se os dados originais a partir da versão reduzida final. Isso possibilita a comparação final entre o conjunto de dados reconstruídos e o conjunto de dados inicial.</li> </ul>
Fim

Fonte: Adaptado de Simon S. Haykin e Van Veen (2001)

**Etapa 1:** Nessa etapa, os pesos sinápticos e os limiares (bias) de cada neurônio da rede podem ser inicializados com valores aleatórios, os quais seguem uma distribuição uniforme no intervalo de  $[0, 1]$ .

**Etapa 2:** O autoencoder é treinado com conjuntos de dados que têm dimensões entre  $2000 \times 64$  e  $2000 \times 256$ , representando a quantidade de bandas de frequência em que as 2000 medições de vibração são divididas. Os dados são divididos em conjuntos de treinamento (80%) e validação (20%) para ajustar os pesos e parâmetros da rede neural e avaliar seu desempenho. Cada exemplo de treinamento passa pelas etapas 3 e 4, nas quais é alimentado na rede neural e os pesos e parâmetros são atualizados com base no erro. Essas etapas são repetidas em cada treinamento, permitindo que a rede se adapte gradualmente aos dados e melhore seu desempenho geral.

**Etapa 3:** Em modelos de autoencoder, os potenciais de ativação e os sinais funcionais de saída são calculados para cada neurônio em cada camada, levando em consideração os valores de entrada ponderados pelos pesos sinápticos. Esses potenciais de ativação são então transformados em sinais funcionais de saída por meio de funções de ativação, que determinam a resposta não linear dos neurônios. Na

camada de saída, é calculado o sinal de erro (MAPE), que mede a discrepância entre os valores originais dos dados e as previsões feitas pelo modelo a partir de uma versão reduzida, parcial, com 50% da dimensão de entrada.

Os gradientes locais são calculados para cada camada, incluindo a camada de saída e todas as outras. Os gradientes locais representam a taxa de variação do erro em relação aos pesos sinápticos e limiares de cada neurônio. Esses gradientes são calculados usando o algoritmo otimização escolhido, que propaga o erro da camada de saída de volta para as camadas anteriores. Uma vez obtidos os gradientes locais, realiza-se o cálculo do ajuste dos pesos sinápticos e limiares da rede. Esse ajuste é realizado usando o método da descida do gradiente, no qual os valores dos pesos e limiares são atualizados na direção oposta ao gradiente local, buscando minimizar o erro. Essa redefinição começa pela camada de saída e progredindo em direção às camadas anteriores.

Durante o treinamento, a propagação e o ajuste de pesos são repetidos para cada ensaio do conjunto de treinamento. Essa repetição ocorre até que a rede alcance o número pré-definido de 20000 épocas. O objetivo é otimizar os componentes da rede, como pesos e bias, visando reduzir o erro. Ao final do treinamento, espera-se que os valores dos componentes da rede estejam ajustados de forma a entregar um erro menor de reconstrução, contribuindo para uma melhor capacidade de previsão e desempenho com relação aos dados originais de treinamento e validação.

**Etapa 4:** Sendo realizada a última época de treinamento, o autoencoder gera a versão reduzida final dos dados de entrada, selecionando um número definido de *features* com base nos valores finais dos pesos. Em seguida, os dados originais são reconstruídos a partir dessa versão reduzida, usando a estrutura do decoder. Essa reconstrução permite comparar os dados reconstruídos com o conjunto de dados inicial, avaliando a exatidão desse processo. Quanto menor o erro de reconstrução, melhor terá sido o desempenho do autoencoder em compreender as informações importantes do *dataset*.

São demonstrados três treinamentos preliminares distintos, mas ainda seguindo a mesma rotina descrita anteriormente. No entanto, há uma variação quanto ao número de bandas de potência, definidas no pré-processamento das medições de vibração, que são realizadas em 2000 arquivos aleatórios selecionados nas medições referentes aos compressores do tipo A. Essa escolha aleatória foi feita para restringir a seleção, uma vez que os perfis de funcionamento dos dois tipos de equipamentos testados possuem características significativamente distintas. Assim, é necessário utilizar esses dados separadamente.

Todos os testes foram realizados utilizando os hiperparâmetros mencionados na Seção 3.2.2, evidenciando dados que comprovam sua seleção para obter uma boa fidelidade na reconstrução dos dados originais. As representações gráficas geradas a partir de cada treinamento têm como objetivo não apenas avaliar um número eficiente de épocas para futuras aplicações do algoritmo, mas também fornecer informações sobre os erros relativos médios acompanhados do desvio padrão em cada banda de frequência. Isso permite avaliar a resposta do autoencoder em diferentes faixas de frequência e divisões dessas no processamento dos dados de vibração. Essas conclusões ajudam a antecipar o desempenho do SAE quando alimentado com um conjunto de dados consideravelmente maior do que o utilizado nos experimentos preliminares.

### 3.2.3.1 Treinamento Preliminar – 64 Bandas

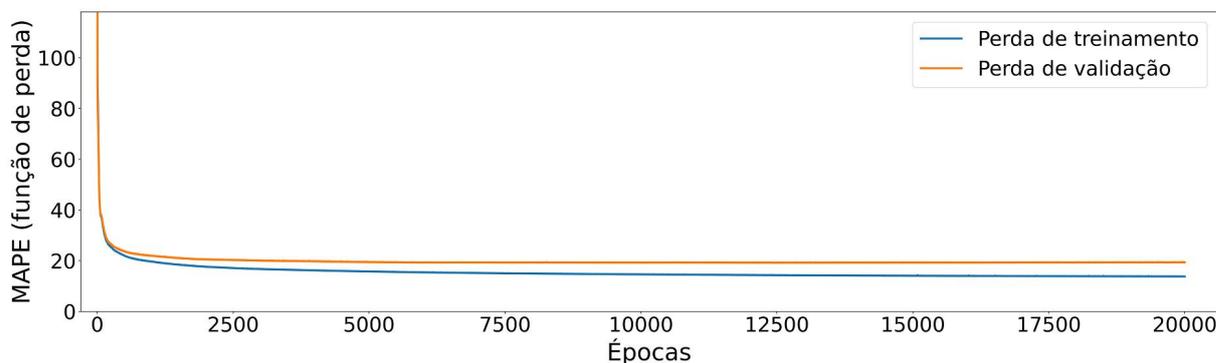
Com base no Quadro 7, foi realizado o primeiro treinamento para validar as definições dos parâmetros de configuração das RNAs implementadas. O treinamento envolveu o pré-processamento dos dados utilizando o método descrito na Seção 3.1.2. Os arquivos correspondentes às 2000 medições foram separados em 64 bandas de potência. Cada faixa de frequência tinha um intervalo uniforme de 200 Hz, começando com a faixa de 0 Hz a 200 Hz e terminando com a faixa de 12 600 Hz a 12 800 Hz. Conseqüentemente, os dados de entrada foram tratados de forma que cada uma das 2000 linhas representasse uma medição de vibração e as colunas representassem as 64 bandas de potência nas quais os dados foram organizados. A taxa de aprendizado definida nesse experimento foi de 0,0006.

A Figura 20 representa a evolução da função de perda com relação ao número de épocas definido para o treinamento do algoritmo. Percebe-se que as duas curvas decrescem de modo acentuado até valores próximos a 20%. Após isso, o autoencoder ainda consegue minimizar a função de perda com relação ao conjunto de treinamento, mas com uma taxa de redução consideravelmente menor. Quando analisado o erro percentual com relação ao conjunto de validação nota-se que a função de perda se estabiliza a partir do intervalo entre 2000 e 2500 épocas.

Considerando a época na qual a função de perda alcançou o mínimo referente ao conjunto de validação, foi decidido analisar as derivadas relativas das curvas contidas na Figura 20 de uma perspectiva mais detalhada, o que é apresentado na Figura 21. Nota-se que antes mesmo da época 7000, aproximadamente, ocorre a tendência de mudança no sentido da taxa de variação da curva de erro de validação, e efetivamente muda-se o sentido da derivada em torno da época 10000. Dessa forma, nas épocas restantes de treinamento do modelo, a tendência é que essa taxa cresça ainda mais, caracterizando que o modelo pode começar a se tornar cada vez menos generalista, podendo ser um indício de *overfitting*.

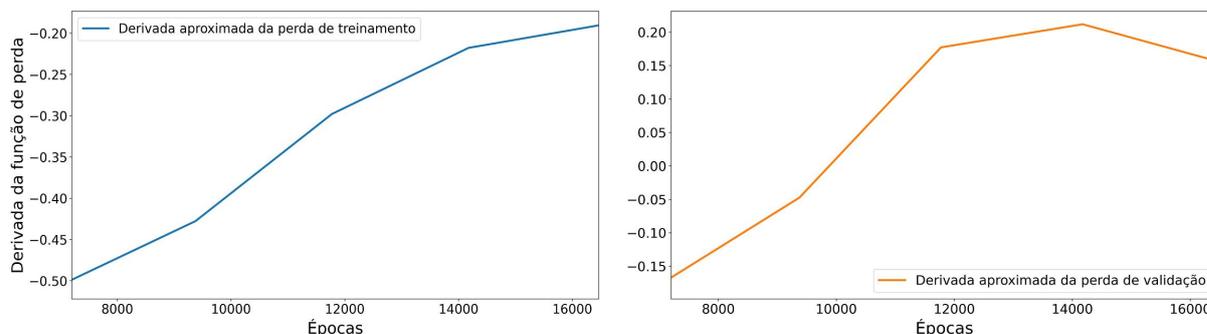
A análise anterior é confirmada quando se observa a curva que representa

Figura 20 – Curvas da função de perda em testes preliminares com processamento do sinal em 64 bandas de frequência.



Fonte: autor

Figura 21 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 64 bandas de frequência.

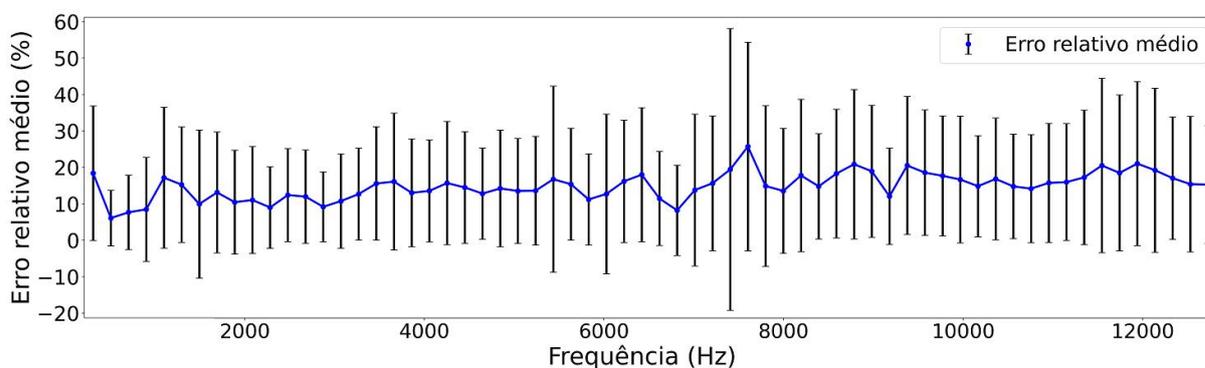


Fonte: Autor

o erro com relação ao conjunto de treinamento na mesma Figura 21. A tendência apresentada pelas derivadas desses dados expõe uma diminuição gradual na taxa de decrescimento do valor da função de perda, mas o autoencoder continua sendo capaz de reduzir o erro percentual médio quando comparado aos dados de entrada, referentes ao conjunto de treinamento, até a última época exibida pelo gráfico. Deve-se ponderar e realizar mais análises gráficas para entender se esse comportamento, evidenciado pelas tendências nos valores da função de perda, resultou em um desempenho inadequado do modelo.

Conforme apresentado na Figura 22, verifica-se que realmente o valor de MAPE apresentado pela Tabela 1, referente a esse treinamento, se comprova pelo perfil das médias dos erros relativos de cada faixa de frequência. No caso da divisão dos 2000 ensaios em 64 faixas, observou-se um desvio padrão, em cada ponto, equilibrado entre -5% e +35% na maioria dos dados, com exceção de algumas faixas. A faixa mais crítica com relação a esse equilíbrio ocorre na 37ª (de 7200 Hz a 7400 Hz), que apresenta desvios aproximados entre -20% e +55%.

Figura 22 – Erro percentual médio em cada banda de potência na reconstrução em testes preliminares com processamento do sinal em 64 bandas de frequência.



Fonte: autor

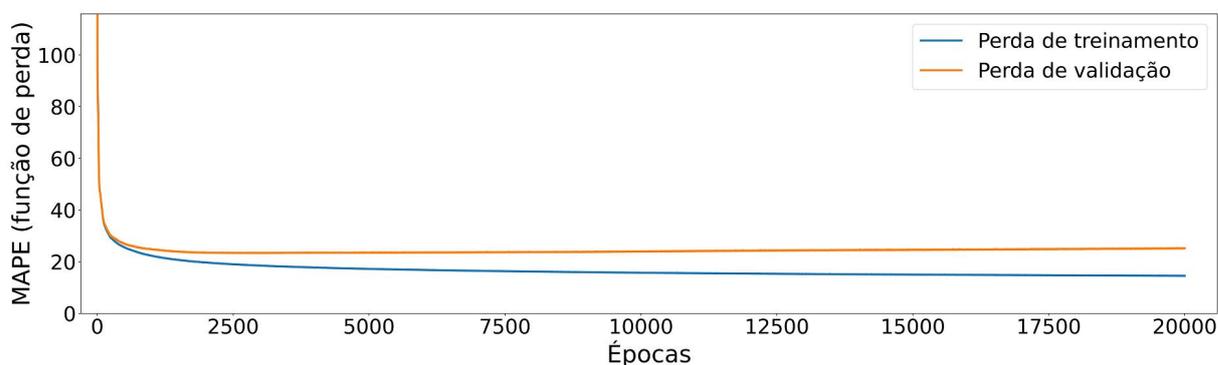
### 3.2.3.2 Treinamento Preliminar – 128 Bandas

Utilizando a mesma rotina de treinamento proposta na Seção 3.2.3, foi executado um segundo treinamento para testar se o algoritmo desenvolvido consegue obter uma exatidão similar ao alcançado na Seção 3.2.3.1 com mais dados de entrada com mais divisões. Esse treinamento também foi realizado com o pré-processamento de 2000 medições de vibração selecionadas aleatoriamente. Os arquivos selecionados foram divididos em 128 bandas de potência, ou seja, cada faixa possuía um intervalo uniforme de 100 Hz até metade da frequência de amostragem (12800 Hz). Outra alteração com relação às configurações do primeiro treinamento foi a definição de uma taxa de aprendizado menor, determinada nesse teste em 0,0002. Essa alteração foi realizada devido ao fato que em experimentos mais breves notou-se que o valor de 0,0006 resultava em um desempenho abaixo do esperado, devido ao incremento no número de entradas do autoencoder.

Analisando a Figura 23, que apresenta o processo de minimização da função de perda, é observado um comportamento similar do treinamento preliminar descrito na Seção 3.2.3.1. Entretanto, é possível constatar uma diferença maior entre os valores finais de erro de validação e de treinamento, fato que pode ser confirmado na Tabela 1, que apresenta os valores que comprovam essa maior diferença entre as curvas. Isso se explica pelo aumento considerável no número de bandas de potência.

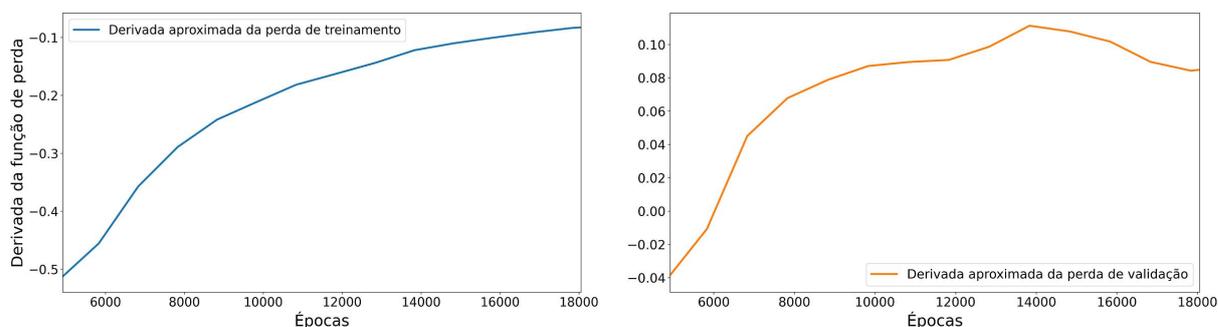
Comparando com o que foi visto na Seção 3.2.3.1, tem-se que a época em que o erro de validação atinge o valor mínimo ocorre cerca de 3000 épocas antes. Essa análise se pauta na Figura 24, a partir da qual também é possível afirmar que a tendência da função de perda de validação se mostra inconstante, mas denotando a mudança no sentido da taxa de variação cerca de 4000 épocas antes do que foi visto no primeiro treinamento. Esse fato aumenta a probabilidade de que o modelo se torne rapidamente mais específico e, conseqüentemente, menos generalista.

Figura 23 – Curvas da função de perda em testes preliminares com processamento do sinal em 128 bandas de frequência.



Fonte: autor

Figura 24 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 128 bandas de frequência.



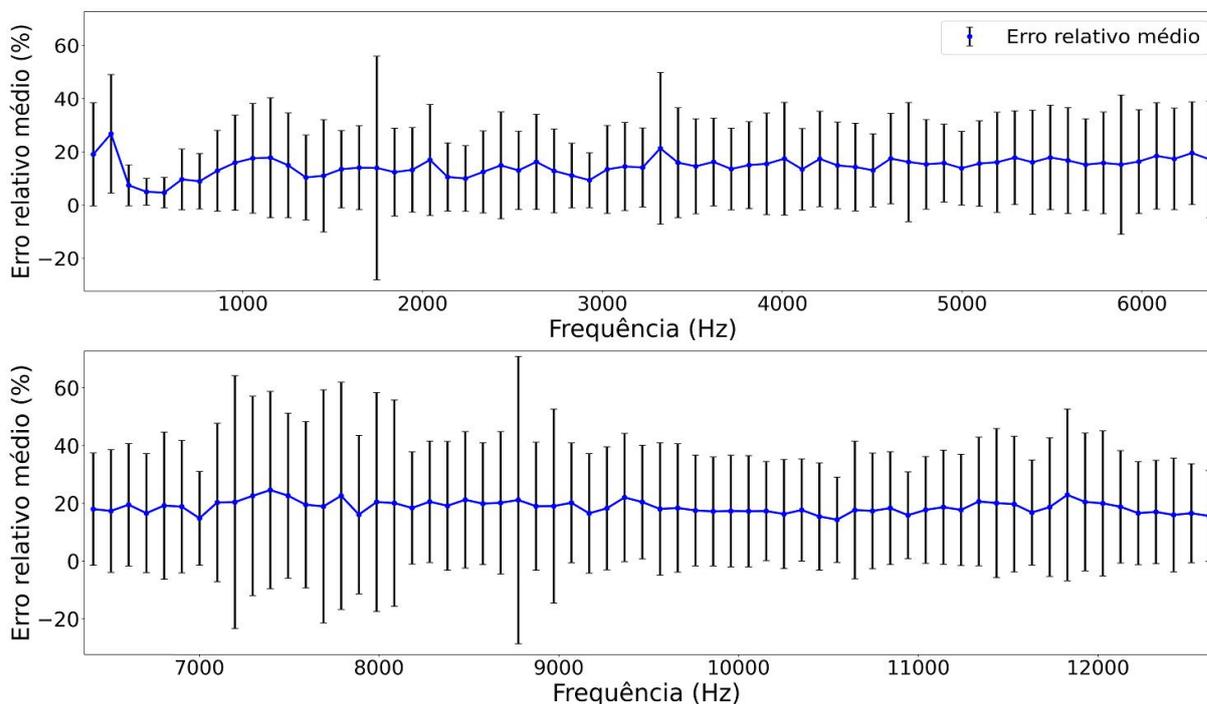
Fonte: autor

Quando se trata da curva de derivadas da função de perda do conjunto de treinamento, pode-se constatar que na metade das épocas do treinamento do modelo a taxa de decrescimento do erro de treinamento já se torna baixa. Tal comportamento pode facilitar que ocorra um caso de *overfitting*, pois apesar de a rede continuar minimizando a função de perda relativa aos dados de treinamento, a distância entre essa curva e a de validação tende a aumentar cada vez mais até o fim das épocas.

A Figura 25 apresenta os erros relativos médios e desvios de todas as 128 bandas de potência. É possível notar que o valor final de MAPE para o conjunto de treinamento, apresentado na Tabela 1, mostra que esse valor ficou menos de 1% maior do que o valor correspondente ao primeiro treinamento. Porém, ao notar a diferença entre os valores finais referentes a esse treinamento, é possível concluir que isso afetou a predição de alguns pontos nesse experimento. Quanto aos erros relativos médios, foi obtido um resultado satisfatório.

Considerando os desvios em cada faixa de frequência, valida-se o que já era esperado para esse teste: ou seja, quanto menor for o passo entre  $f_{min}$  e  $f_{max}$ , maior terá de ser a profundidade do aprendizado das RNAs. Comentando mais especificamente

Figura 25 – Erro percentual médio entre cada banda de potência em testes preliminares com processamento do sinal em 128 bandas.



Fonte: autor

sobre os desvios, ainda é possível validar que na maioria das bandas de potência o desvio fica entre  $-10\%$  e  $+40\%$ , com exceção de poucas faixas que possuem desvios entre  $-25\%$  e  $+60\%$ . Assim sendo, é possível inferir que, nessas condições de treinamento, fica mais clara a necessidade de treinar o algoritmo em uma base de dados maior do que a observada na Seção 3.2.3.1, desde que seja comprovada certa resposta do modelo na Seção 3.2.3.3.

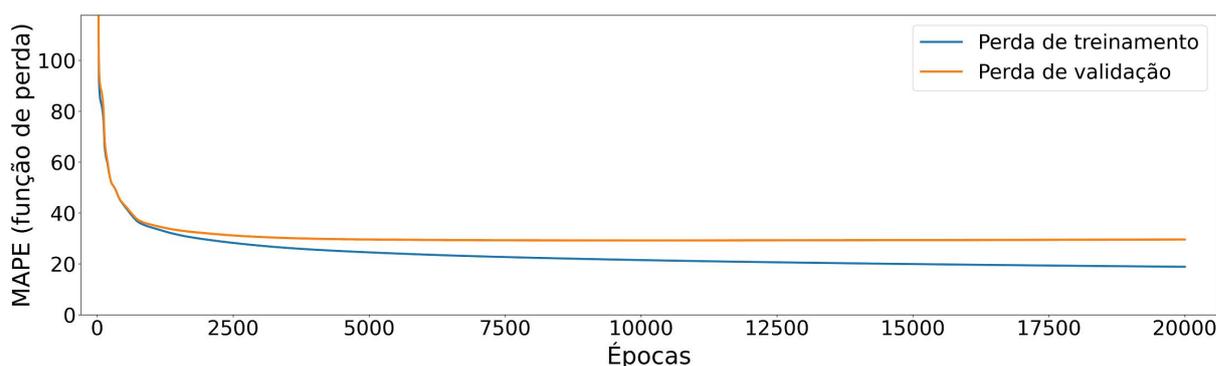
### 3.2.3.3 Treinamento Preliminar – 256 Bandas

Para finalizar a rotina de treinamentos preliminares do autoencoder, o modelo foi submetido a um conjunto de dados mais complexo do que os outros dois realizados nas Seções 3.2.3.1 e 3.2.3.2. Considerando as mesmas etapas utilizadas anteriormente, o terceiro treinamento também contou com 2000 medições de vibração, mas dessa vez separados em 256 faixas de frequência, ou seja, definindo o passo uniforme em 50 Hz. Devido a complexidade desse padrão de conjunto de dados de entrada, para esse cenário o valor de taxa de aprendizado foi 0,00007.

Demonstrando a evolução das curvas relativas à função de perda e considerando as 20000 épocas determinadas para os três ensaios, a Figura 26, apresenta os dados do treinamento considerando 256 bandas de potência. Pode-se afirmar que o aspecto das curvas permanece semelhante ao apresentado nos outros treinamentos,

mas pode ser notada uma diferença ainda maior na estabilização das curvas de erro de validação e treinamento, resultando em uma diferença próxima a 11% entre os valores finais de validação e treinamento, considerando a função de perda para esses conjuntos de dados.

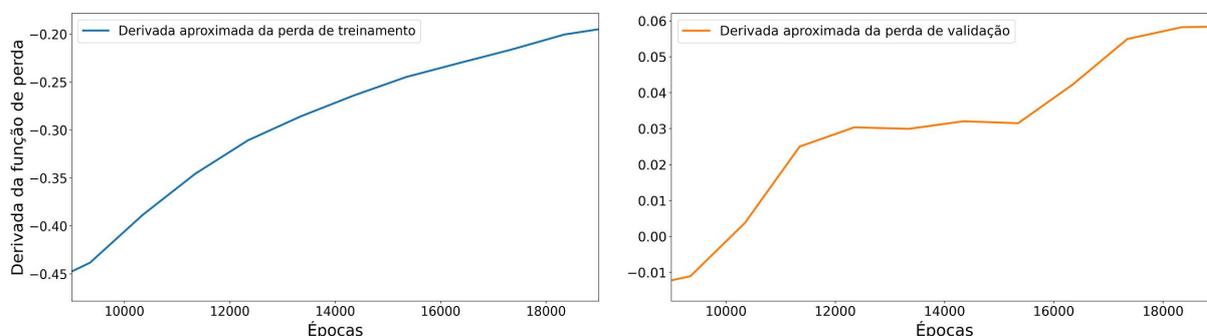
Figura 26 – Curvas da função de perda em testes preliminares com processamento do sinal em 256 bandas de frequência.



Fonte: autor

O perfil visto na Figura 27, se assemelha ao que consta na Seção 3.2.3.1. Essa semelhança não se confere com base no formato das derivadas aproximadas das duas curvas, mas sim pelo fato de a troca do sinal da taxa de variação do erro de validação ocorrer próximo à época 10000. A curva de tendência de variação do erro de validação possui um comportamento diferente do que foi visto nos outros experimentos. Isso se explica em decorrência do crescimento da derivada aproximada dessa curva ocorrer de maneira mais lenta. Provavelmente a origem desse resultado está no fato de que a escolha da taxa de aprendizado possa ter sido mais adequada para esse cenário do que as escolhas em testes anteriores.

Figura 27 – Derivadas aproximadas da função de perda em testes preliminares com processamento do sinal em 256 bandas de frequência.



Fonte: autor

Quanto à curva de minimização da função de perda referente ao conjunto de treinamento, pode ser verificado que a taxa de variação só atinge um valor próximo a

-0,2 quando faltam menos de duas épocas para o fim do treinamento. Considerando isso, após ser alcançado o erro mínimo de validação ainda houve um decréscimo considerável no erro de treinamento, mas isso não impede o fato de que a discrepância entre os valores finais dos valores da função de perda influenciem no desempenho do modelo.

Considerando a Figura 28, de imediato percebe-se que em certas faixas de frequência o modelo gera previsões menos exatas do que o obtido nos últimos dois treinamentos. Esse fato pode ser observado, por exemplo, na faixa próxima a 180, que equivale a frequências em torno de 9000 Hz, contando com o erro relativo variando entre -50% e +110%. Outra banda importante para análise é a primeira (0 Hz a 50 Hz), pois nesse ponto o desvio pode chegar próximo a 100% e ainda o próprio erro relativo médio da faixa fica próximo a 60%. Esse fato pode ser explicado baseado nas figuras apresentadas na Seção 3.1.2, visto que há um pico de vibração em frequências próximas a 60 Hz, provavelmente fazendo com que o modelo considere que a primeira faixa de frequência, nesse treinamento, esteja em um patamar mais próximo da banda subsequente. A Tabela 1 apresenta os valores finais da função de perda, referentes aos três treinamentos detalhados nas seções seguintes.

Para concluir, pode ser observado que quanto maior a divisão entre faixas de frequência, mais comprometida fica o resultado da previsão do modelo. Além de uma ocorrência maior de valores elevados de desvio padrão, o desvio médio ficou cada vez maior comparando a ordem dos três treinamentos realizados. Para diminuir a diferença entre erro de treinamento e de validação, é indicado que o modelo seja testado com um *dataset* consideravelmente maior, teste que será realizado no Capítulo 4.

Tabela 1 – Comparação dos últimos valores de perda entre os treinamentos preliminares.

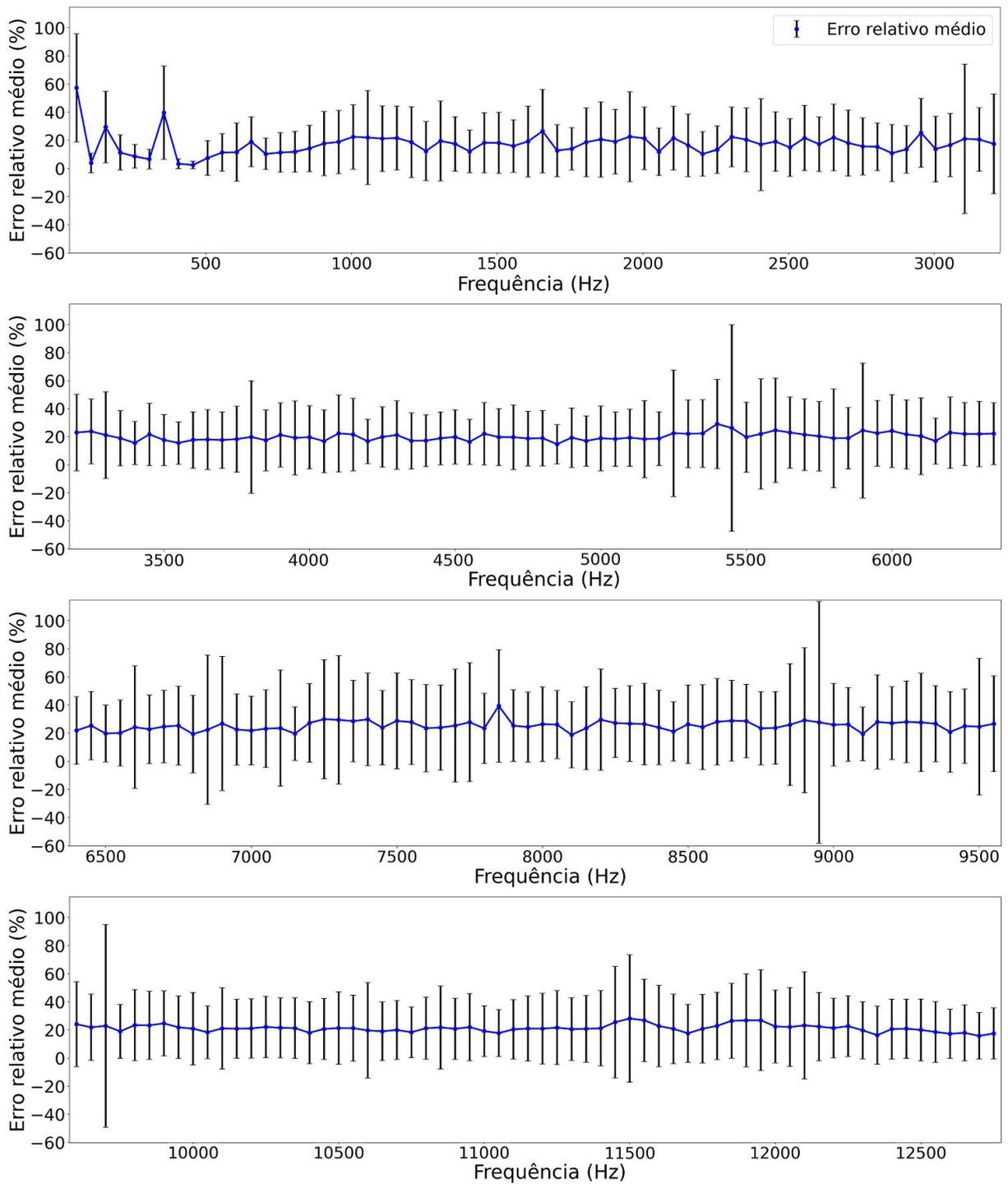
<b>Função Perda (MAPE)</b>	<b>Treinamento</b>	<b>Validação</b>
<b>Treinamento Preliminar – 64 Bandas</b>	<b>13,8%</b>	<b>19,4%</b>
<b>Treinamento Preliminar – 128 Bandas</b>	<b>14,5%</b>	<b>25,2%</b>
<b>Treinamento Preliminar – 256 Bandas</b>	<b>18,9%</b>	<b>29,6%</b>

Fonte: autor

### 3.3 CONSIDERAÇÕES FINAIS

Nesse capítulo foram apresentadas as etapas referentes à elaboração da rotina de pré-processamento dos dados de entrada em bandas de potência, o processo de escolha de técnica de normalização e como foi feita a decisão dos parâmetros de configuração do autoencoder desenvolvido. Por fim, foram apresentados os resultados dos três treinamentos preliminares, realizados com o objetivo de validar o desempenho das redes neurais e fazer os últimos ajustes de configuração. Os resultados dos treinamen-

Figura 28 – Erro percentual médio entre cada banda de potência em testes preliminares com processamento do sinal em 256 bandas.



Fonte: autor

tos representados pelas Seções 3.2.3.1, 3.2.3.2 e 3.2.3.3, corroboram a necessidade de testar o autoencoder utilizando como entrada um conjunto consideravelmente maior de dados. Esse experimento será relatado no Capítulo 4, no qual são utilizadas como dados de entrada todas as medições de vibração adquiridas das unidades do modelo A de compressores.

## 4 RESULTADOS

Neste capítulo são apresentados os resultados finais deste projeto, considerando a utilização dos melhores parâmetros de configuração obtidos no Capítulo 3. Assim sendo, a maior alteração com relação aos treinamentos preliminares do modelo se baseia na quantidade dos dados de entrada. Nos treinamentos descritos neste capítulo, todas as medições de vibração adquiridas nos compressores do modelo A foram pré-processadas e serviram como dados de entrada do autoencoder em treinamentos e testes de desempenho.

### 4.1 TREINAMENTOS FINAIS

É importante ressaltar que, nos treinamentos finais, os dados de vibração foram separados em três conjuntos, diferentemente do que foi feito na Seção 3.2.3. A divisão adicional se refere a um conjunto de testes, que possui o propósito de analisar o desempenho do modelo final perante a dados nunca antes processados pela rede neural, simulando assim cenários mais próximos da aplicação real. A Tabela 2 apresenta o número de amostras que denotam diferença na quantidade de arquivos de medição utilizados nesses treinamentos.

Tabela 2 – Comparação entre dimensões de conjuntos em testes preliminares e finais.

Conjuntos	Treinamento	Validação	Teste	Total
Treinamento preliminares	1600	400	0	2000
Treinamentos finais	19520	4880	6099	30499

Fonte: autor

Com o intuito de validar o desempenho da versão final do autoencoder desenvolvido, foram feitos três treinamentos com as mesmas configurações de separação de bandas de potência utilizadas na Seção 3.2.3. Dessa forma, são mais diretas as comparações quanto à influência de um conjunto de treinamento mais abrangente no desempenho do algoritmo.

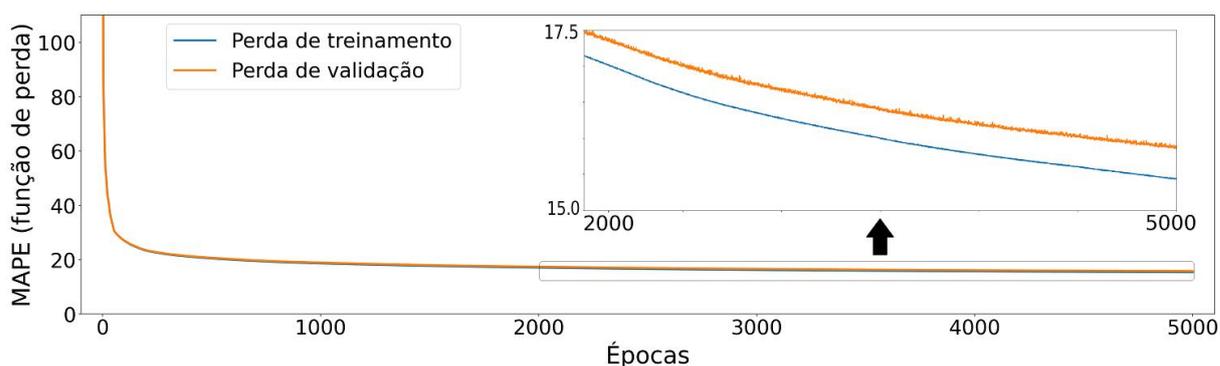
#### 4.1.1 Pré-processamento dos Dados de Entrada em 64 Bandas de Potência

O primeiro dos treinamentos finais foi executado utilizando a mesma configuração de pré-processamento de dados da Seção 3.2.3.1, ou seja, todos os dados de medição foram divididos em 64 bandas de potência. Quanto aos parâmetros de configuração, foram usados os hiperparâmetros definidos na Seção 3.2.2 e o valor de taxa de aprendizado foi de 0,0006.

Analisando a Figura 29, é possível observar que as curvas se encontram relativamente assentadas em valores pouco menores que 20% a partir da época 1000.

Mas para analisar melhor essas curvas também é apresentada uma versão ampliada da mesma, para compreender o comportamento dessas curvas em uma escala reduzida, visando avaliar se ainda havia tendência de redução tanto na perda de validação quanto na perda de treinamento.

Figura 29 – Curvas da função de perda do treinamento final com processamento do sinal em 64 bandas de frequência.

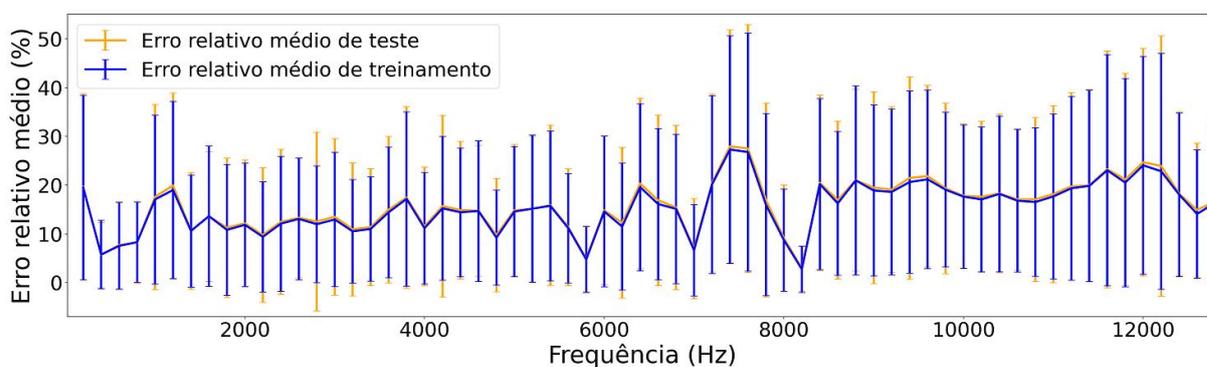


Fonte: autor

A representação ampliada demonstra que as duas curvas ainda não atingiram seus valores mínimos, mas a taxa de decrescimento não está significativa. Pode-se concluir, a partir disso, que o modelo provavelmente não apresentou *overfitting* e está próximo do menor valor de função de perda para essa configuração de autoencoder.

Assim como foi visto na Seção 3.2.3.1, essa divisão em 64 faixas de frequência gerou um modelo consideravelmente generalista. Isso se explica pelo fato de as curvas referentes aos erros médios de treinamento e de teste demonstrarem perfis muito próximos, fato também confirmado ao comparar os desvios de cada curva. O maior desvio que pode ser notado na Figura 30 está em torno de 55%, mas poucas bandas de potência apresentam tal patamar de desvio.

Figura 30 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 64 bandas de frequência.



Fonte: autor

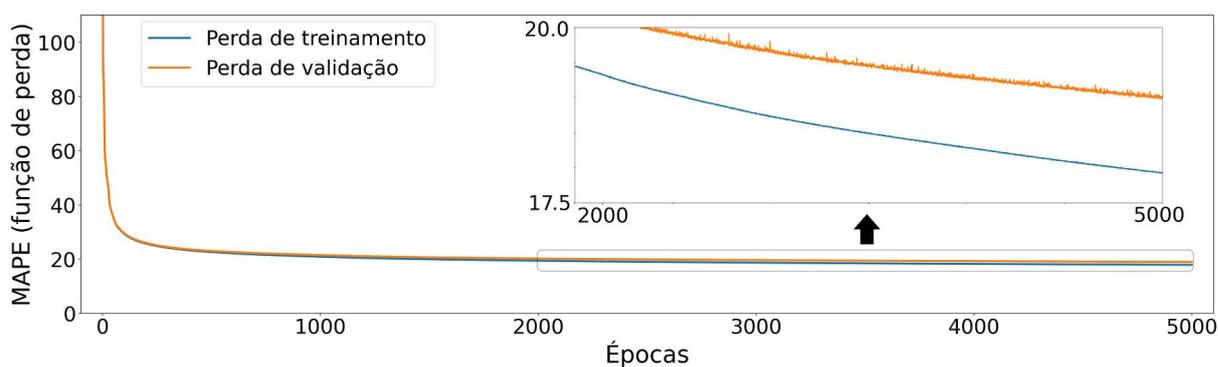
Quando o autoencoder foi utilizado para predição dos dados do conjunto de testes (dados que não participaram do treinamento), obteve um resultado similar ao produzido quando comparado com a reconstrução do conjunto de treinamento. Dessa forma, é possível afirmar que, devido à similaridade entre as duas curvas, o modelo pode ser considerado generalista o suficiente.

#### 4.1.2 Pré-processamento dos Dados de Entrada em 128 Bandas de Potência

O segundo treinamento final foi feito configurando o pré-processamento de dados de entrada que resulte em 128 bandas de potência. Os hiperparâmetros utilizados seguem as mesmas diretrizes detalhadas na Seção 3.2.2 e utilizadas em todos os treinamentos, definindo o valor da taxa de aprendizado em 0,0002.

Com base na Figura 31, pode-se afirmar que a diferença entre os valores finais das curvas da função de perda, ao final das épocas, ficou maior do que o observado na Figura 29. Ao visualizar a versão ampliada para o treinamento dos dados separados em 128 faixas de frequência, percebe-se que o valor mínimo atingido pela curva referente aos dados de treinamento, foi maior do que o obtido em relação ao treinamento preliminar 2 (Seção 3.2.3.2). Em contrapartida, a diferença entre as curvas de treinamento e validação nesse treinamento final foi consideravelmente melhor do que o obtido no treinamento preliminar equivalente.

Figura 31 – Curvas da função de perda do treinamento final com processamento do sinal em 128 bandas de frequência.



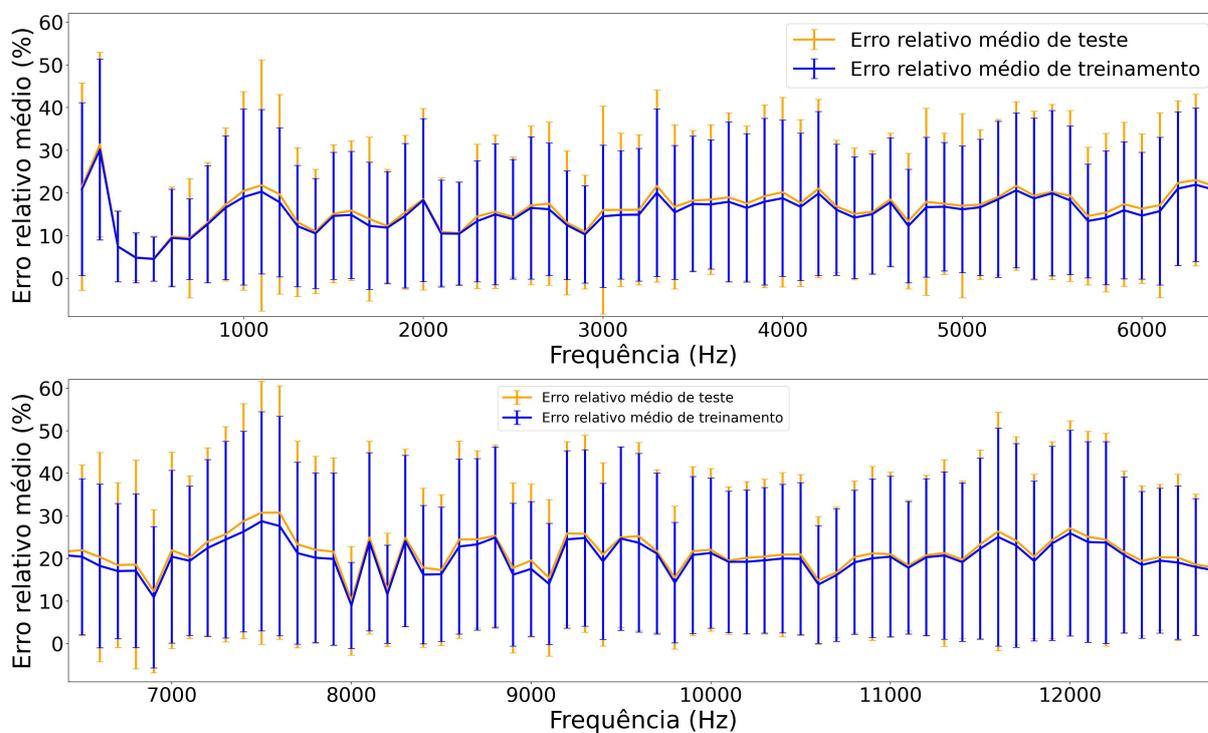
Fonte: autor

Tanto a curva de validação quanto a de treinamento, ainda não alcançaram seu valor mínimo, mas a tendência indica que estão próximos disso. Isso se explica pelo fato de o recorte ampliado do gráfico, entre 2000 e 5000 épocas, demonstrar um decrescimento baixo com base no intervalo de 3000 épocas.

A Figura 32 indica que, para essa configuração com dados contendo 128 faixas de frequência, os resultados obtidos são mais consistentes do que os obtidos na Seção 3.2.3.2. O fato que explica essa afirmação se apoia no comportamento mais

equilibrado dos desvios referentes a cada faixa, variando entre valores próximos a 0% e 50%, considerando a maioria dos pontos.

Figura 32 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 128 bandas de frequência.



Fonte: autor

Uma análise interessante desse treinamento final foi a comparação dos erros relativos médios de treinamento com as mesmas métricas adquiridas com base no conjunto de teste. Ou seja, em um teste de desempenho considerado imparcial, o modelo gerou erros similares ao que foi notado no treinamento, atestando o resultado do autoencoder treinado para operar com dados alheios ao treinamento.

#### 4.1.3 Pré-processamento dos Dados de Entrada em 256 Bandas de Potência

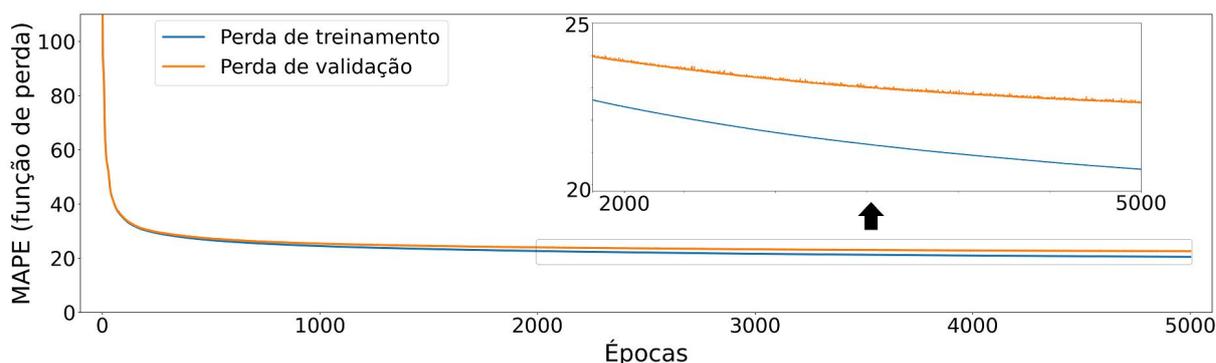
Para completar a rotina de treinamentos finais do modelo desenvolvido, foi realizado o tratamento das medições de vibração em 256 bandas de potência, para avaliar o desempenho do autoencoder nesse cenário. Considerando os resultados da Seção 3.2.3.3, essa configuração foi a mais crítica dentre os treinamentos preliminares. Dessa forma, espera-se um melhor desempenho do modelo quando treinado com o total de medições de vibração dos compressores de tipo A. A taxa de aprendizado foi determinada em 0,00007.

Com base na Figura 33 e fazendo um paralelo com o que foi obtido como resultado na Seção 3.2.3.3, esse treinamento gerou a maior diferença entre as curvas de erro de validação e treinamento, quando comparado aos outros treinamentos finais.

Ou seja, mesmo com uma quantidade de dados de maior, o modelo tem mais dificuldade para generalizar os perfis contidos nesses dados. Mas comparando com o treino preliminar equivalente, percebe-se uma diferença entre as curvas bem menor do que na Figura 26.

Considerando a versão ampliada da minimização das curvas de função de perda apresentada pela Figura 33, assim como nos outros treinamentos finais, tanto o erro de validação quanto o erro de treinamento não alcançaram o mínimo. Mas, usando como exemplo a curva de validação, em um intervalo de 3000 épocas esse erro foi reduzido em aproximadamente apenas 1%, esse fato já demonstra que está ocorrendo uma queda gradual da taxa de decrescimento dessa curva. Quanto à curva de treinamento, nota-se um comportamento parecido, mas se fixando em um valor de MAPE em torno de 20%.

Figura 33 – Curvas da função de perda do treinamento final com processamento do sinal em 256 bandas de frequência.

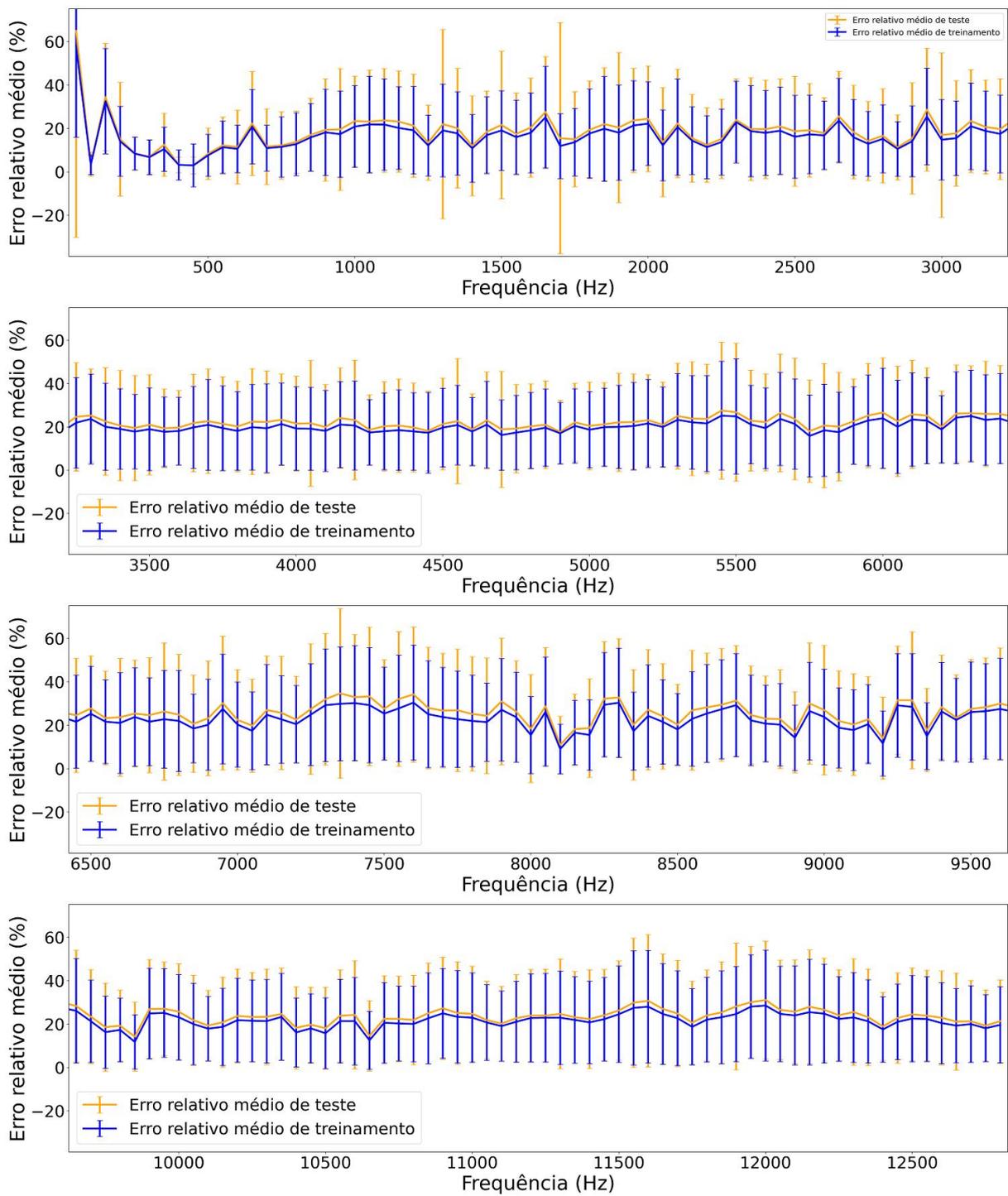


Fonte: autor

Comparando a Figura 34 com os resultados alcançados no último treinamento preliminar (Seção 3.2.3.3), nesse treinamento final houve uma incidência consideravelmente menor com relação a picos de desvios no erro relativo médio para cada banda de potência. Nota-se que na maioria dos pontos os desvios ficam entre  $-10\%$  e  $+50\%$ , tendo como base tanto os erros de teste quanto o de treinamento. As exceções se concentram, quase que exclusivamente, em uma faixa de bandas de frequência entre 0 Hz a 3000 Hz, denotando uma disparidade maior entre os desvios referentes ao conjunto de treinamento e a mesma métrica no contexto do conjunto de teste.

A Figura 35 evidencia de maneira mais detalhada o maior pico de desvio-padrão dentre todos os pontos da Figura 34. Esse resultado demonstrou que mesmo contando com uma quantidade muito maior de dados para o treinamento, o modelo não conseguiu atenuar o erro relativo médio para a primeira banda de frequência. Segundo a Figura 35, o autoencoder manteve esse erro médio em valores próximos a 60%, com desvio padrão entre 15% e 100%, mas o que mais indica essa falha de desempenho é a amplificação do desvio padrão relacionado ao conjunto de teste, que ficou entre

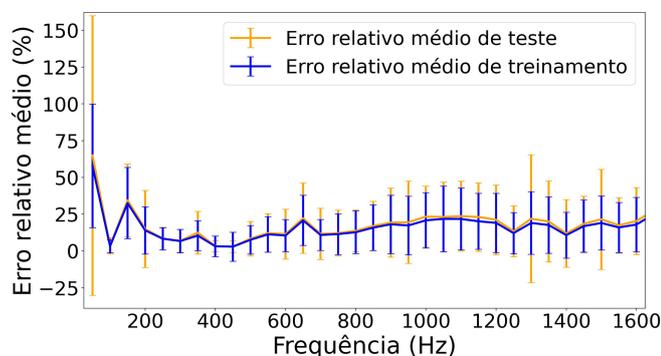
Figura 34 – Comparação entre erros de treinamento e de teste no treinamento final com processamento do sinal em 256 bandas de frequência.



Fonte: autor

-25% e +150%.

Figura 35 – Desvio padrão de erro relativo médio nas baixas frequências do treinamento final com processamento do sinal em 256 bandas de frequência.



Fonte: autor

#### 4.1.4 Considerações Finais de Desempenho entre os Treinamentos

A Tabela 3 demonstra os valores finais da função perda MAPE para todos os treinamentos realizados, para que seja mais representativa uma comparação visando concluir sobre a generalidade do modelo de autoencoder desenvolvido.

Tabela 3 – Comparação dos últimos valores de perda entre todos os treinamentos realizados.

Conjuntos	Bandas de Potência	Treinamento	Validação	Teste
Treinamento preliminar	64	13,8%	19,4%	-
Treinamentos final		15,4%	15,9%	15,8%
Treinamento preliminar	128	14,5%	25,2%	-
Treinamentos final		17,9%	19,0%	18,9%
Treinamento preliminar	256	18,9%	29,6%	-
Treinamentos final		20,4%	22,5%	22,4%

Fonte: autor

Concluindo de forma geral sobre os treinamentos, percebe-se que quando o modelo foi treinado com um grande conjunto de dados de medição, o autoencoder conseguiu ser mais generalista. Isso ocorre pois as métricas mais interessantes para avaliar tal característica em um modelo desse tipo são os erros de validação e principalmente o de teste, por serem conjuntos de dados novos para o autoencoder. Desse modo, mesmo que a função de perda nos experimentos preliminares tenha alcançado valores menores em relação ao conjunto de treinamento, o comportamento da curva de validação foi muito mais desequilibrado e, conseqüentemente, isso indica desempenho pior na capacidade de generalização do modelo.

Com relação às diferentes configurações de bandas de potência, os resultados associados a dados de entrada tratados em 64 e 128 faixas fizeram com que o modelo

conseguisse ser mais generalista e manter um erro mais equilibrado. Esse fato denota a maior complexidade de representar e generalizar perfis de funcionamento de dados de vibração divididos em 256 faixas de frequência.

## 4.2 INTERPRETAÇÃO DAS *FEATURES* CONTIDAS NO ESPAÇO LATENTE

Outro objetivo do projeto é analisar faixas de frequência mais significativas dos dados de vibração com o auxílio do autoencoder. Para realizar tal análise, foram utilizados os pesos calculados pelo modelo em cada um dos três treinamentos finais e foi seguido o padrão da dimensão do espaço latente baseado na dimensão do dado de entrada. Dessa forma, como nos treinamentos realizados, o espaço latente foi configurado para ter a metade da dimensão da entrada.

Para auxiliar na análise das *features* escolhidas para formar a versão com dimensão reduzida, foram apresentados gráficos no qual são ponderados os valores dos pesos selecionados para cada intervalo de frequência. Esses gráficos indicam o percentual que cada intervalo contribui na proporção total, considerando a soma dos pesos que constituem o espaço latente e a soma dos pesos referentes às bandas de potência contidas em cada intervalo de frequência.

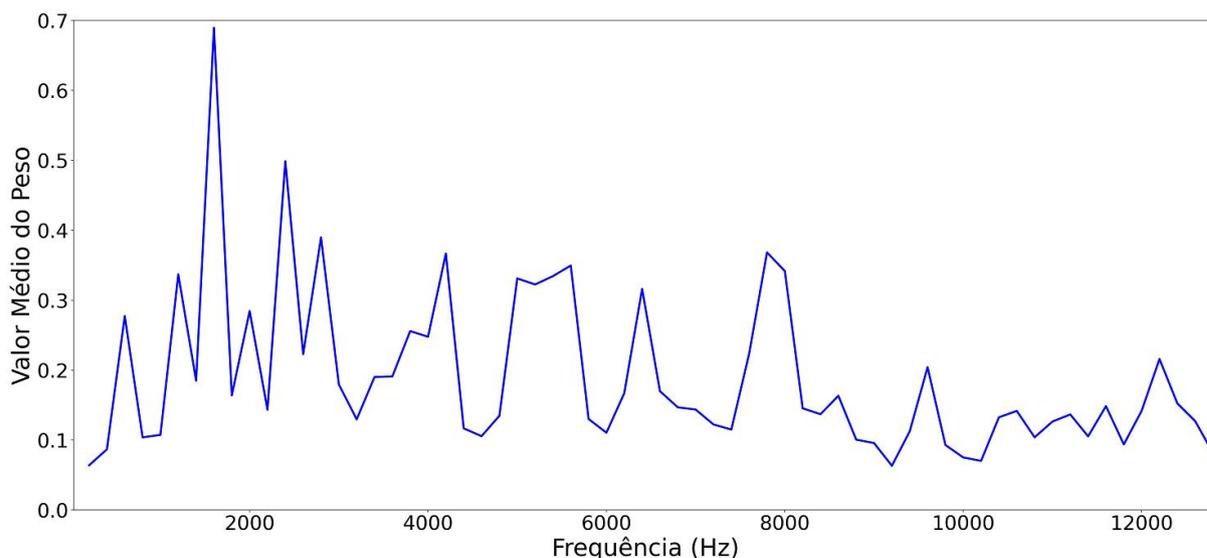
### 4.2.1 Espaço Latente de 32 *Features*

A Figura 36 apresenta os pesos médios que foram definidos pelo modelo para cada uma das 64 bandas de frequência, com base no treinamento final apresentado na Seção 4.1.1. Pode-se notar que o maior pico em termos de peso adotado pelo autoencoder seria referente a, aproximadamente, 1600 Hz. Outro fato importante é que em bandas de frequência maiores que 8000 Hz, o algoritmo atribui um patamar menor de valores nos pesos médios, ou seja, considera que tais faixas não possuem tanta influência na generalização do perfil dos dados de entrada.

Para facilitar a análise das 32 *features* contidas no espaço latente, que equivalem aos 32 pesos médios com o maior valor da Figura 36, foi realizada uma segunda divisão que abrange oito intervalos de frequência. No caso de um treinamento com entrada de dimensão 64, cada intervalo de frequência abrange 8 bandas. Dentre os 32 pesos médios escolhidos, cada um deles faz referência a uma banda de frequência original. Baseado nisso, os pesos foram divididos entre os intervalos, para depois calcular o percentual de influência das bandas presentes em cada intervalo de forma conjunta dentre o valor total dos 32 pesos.

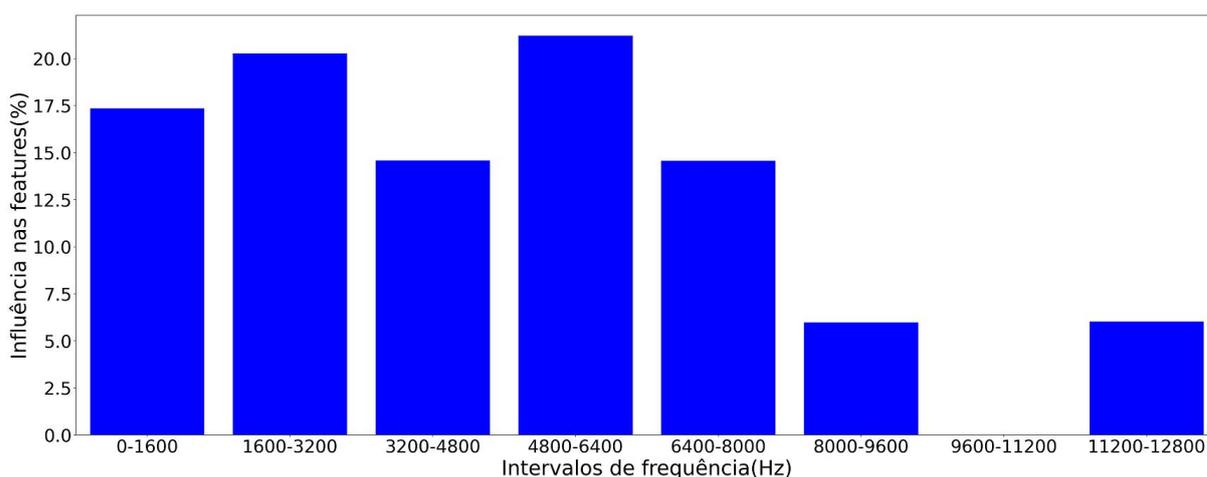
Com base na Figura 37, é possível afirmar que o intervalo de frequência que mais influencia na escolha dos 32 pesos foi entre 4800 Hz a 6400 Hz, ou seja, as bandas de frequência contidas nesse intervalo foram as *features* mais significantes. Percebe-se que, especificamente, o penúltimo intervalo de frequência não teve ne-

Figura 36 – Pesos médios calculados para cada uma das 64 bandas de potência.



Fonte: autor

nhuma *feature* presente no espaço latente, mas, de modo geral, o patamar de influência cai consideravelmente a partir de 8000 Hz. Isso significa que poucas bandas de frequência contidas nos últimos três intervalos tiveram seus pesos médios entre os 32 maiores separados para constituir uma versão reduzida dos dados de entrada.

Figura 37 – Intervalos de frequência mais influentes na escolha de 32 *features* contidas no espaço latente.

Fonte: autor

#### 4.2.2 Espaço Latente de 64 *Features*

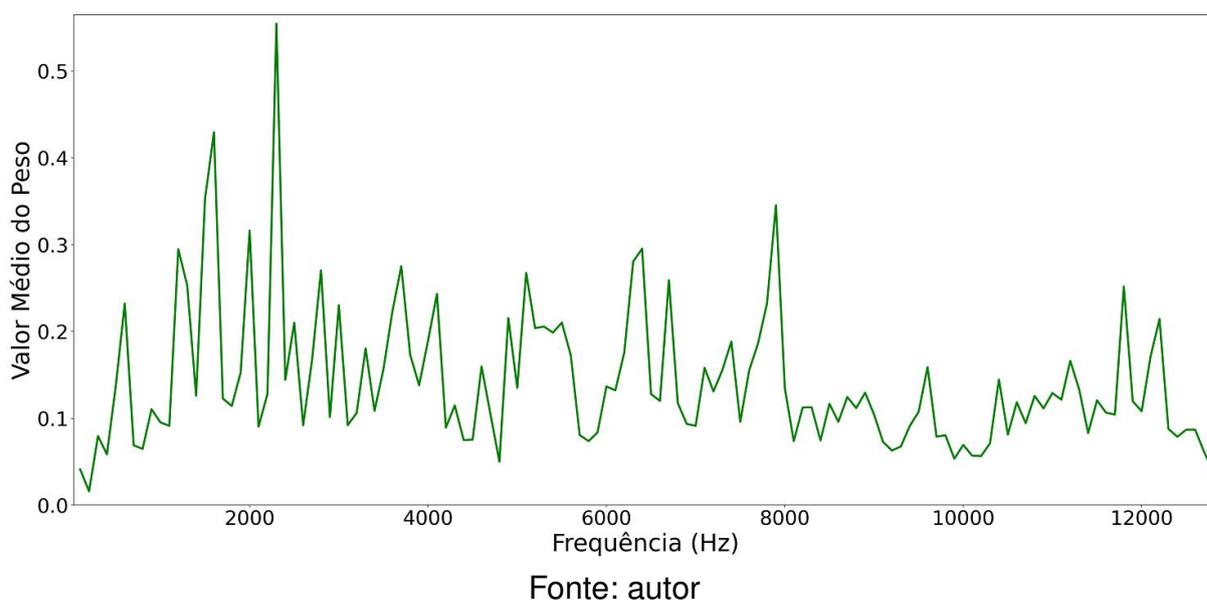
Considerando o treinamento final exposto pela Seção 4.1.2, foi apresentada a Figura 38, no qual ficam explícitos os pesos médios determinados pelo modelo para cada uma das 128 bandas de frequência referentes ao dado de entrada desse

treinamento. Visando analisar o comportamento dos pesos tendo como referencia um número maior de bandas de entrada e como isso afetaria a escolha de *features* contidas no espaço latente.

No caso da Figura 38, de forma equivalente ao que foi visto na Figura 36, há uma atenuação nos valores dos pesos a partir de 8000 Hz. Mas, por se tratar de um treinamento em que o perfil dos dados de entrada é mais específico do que no treinamento detalhado na Seção 4.1.1, nota-se uma distribuição mais equilibrada nos valores atribuídos pelo autoencoder no geral.

Observando a Figura 39, percebe-se que ainda há uma predominância das bandas de frequência contidas de 0 Hz a 8000 Hz, quando considerada a influência nas 64 *features* contidas no espaço latente. Dessa forma, ainda ocorre uma redução considerável tanto na quantidade quanto no valor dos pesos escolhidos, correspondentes às bandas contidas nos intervalos de frequência restantes, algo similar com o que foi visto na Figura 37. Ou seja, dentre os 64 pesos escolhidos, apenas uma minoria desses se originaram das 48 bandas de frequência contidas de 8000 Hz a 12 800 Hz.

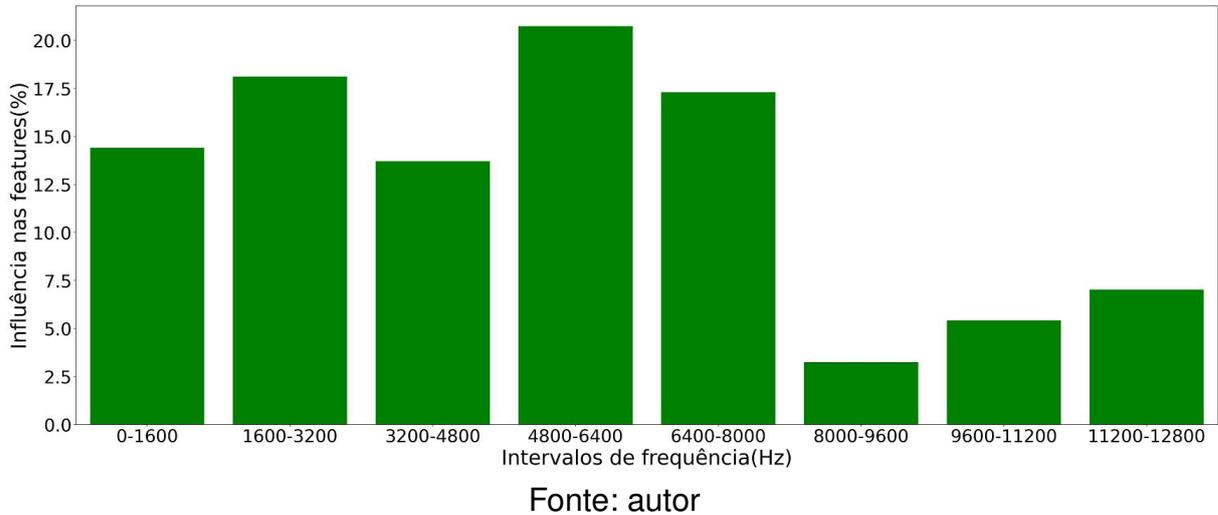
Figura 38 – Pesos médios calculados para cada uma das 128 bandas de potência.



### 4.2.3 Espaço Latente de 128 *Features*

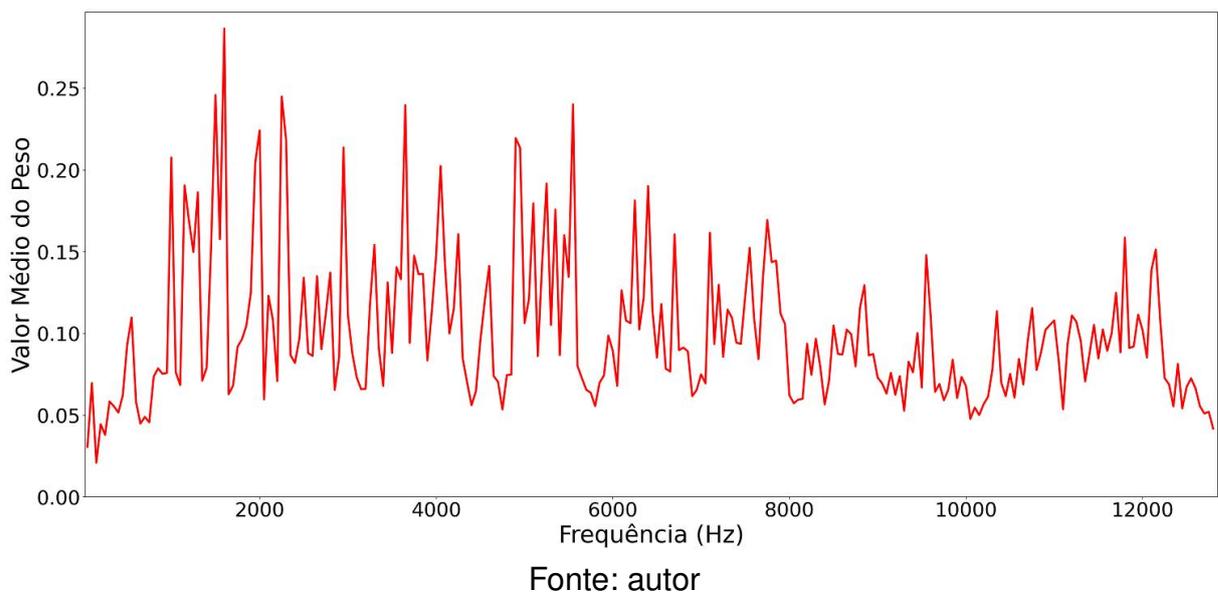
Baseando-se na Figura 40, que utiliza os pesos médios calculados no treinamento 4.1.3, fica evidente que há uma ainda maior distribuição desses valores atribuídos pelo modelo a essas 256 bandas originárias desse cenário de dados de pré-processamento. Uma análise dos valores finais dos pesos do autoencoder foi feita, com o objetivo de avaliar as *features* presentes no espaço latente, considerando a entrada de dados de maior complexidade entre todos os treinamentos detalhados na Seção 4.1.

Figura 39 – Intervalos de frequência mais influentes na escolha de 64 *features* contidas no espaço latente.



Verificando a representação dos pesos médios apresentada pela Figura 40, observa-se que o patamar geral dos 256 valores diminuiu com relação ao que foi notado nas duas análises anteriormente descritas (Seções 4.2.1 e 4.2.2). Ainda assim, é possível constatar uma queda no nível dos pesos atribuídos para bandas de frequência a partir de 8000 Hz, similar ao comportamento detectado em treinamentos finais anteriores, porém, observou-se um decrescimento mais suave nesse caso.

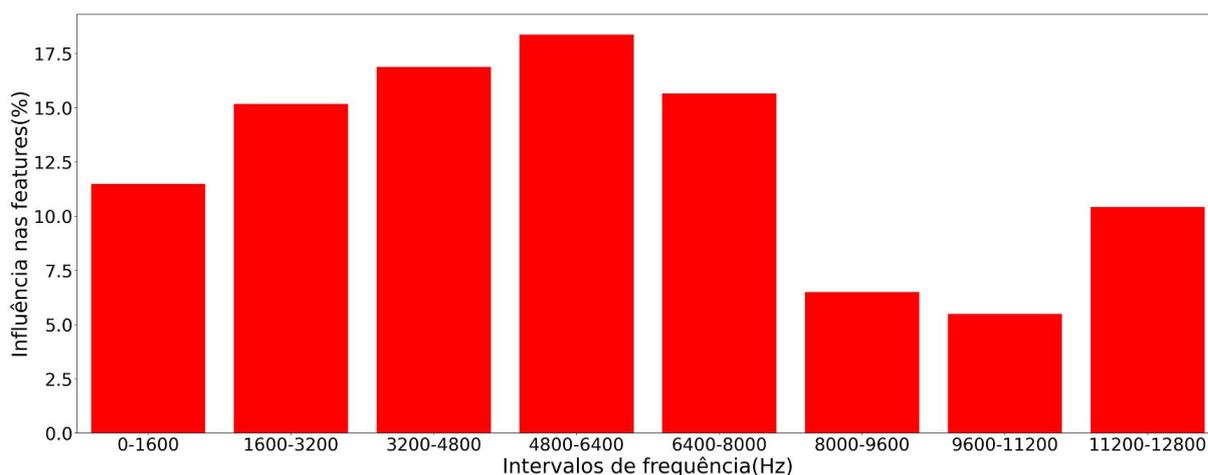
Figura 40 – Pesos médios calculados para cada uma das 256 bandas de potência.



Uma particularidade observada na influência da escolha de *features* contidas no espaço latente, reproduzida pela Figura 41, foi uma maior valorização na escolha dos pesos médios assumidos pelo modelo às bandas de potência incluídas no último intervalo de frequência analisado, de 11 200 Hz a 12 800 Hz. Quanto à influência dos

intervalos de frequência contidos entre 8000 Hz a 11 200 Hz, o patamar segue a queda vista em outros cenários, ou seja, poucos dos pesos provenientes das 64 bandas de potência contidas nessa faixa estiveram entre as 128 *features* contidas no espaço latente.

Figura 41 – Intervalos de frequência mais influentes na escolha de 128 *features* contidas no espaço latente.



Fonte: autor

#### 4.2.4 Considerações Finais da Análise de Importância das *Features*

Utilizando como referência os três procedimentos de análise de importância de *features* detalhados anteriormente (Seções 4.2.1, 4.2.2 e 4.2.3), esta seção tem como objetivo comparar os resultados obtidos com o método implementado, com ênfase na influência dos intervalos de frequência na seleção dos pesos mais significativos para representar o perfil dos dados de entrada.

A Tabela 4 especifica a proporção de influência, com base nos intervalos de frequência, considerando a soma dos pesos médios escolhidos contidos em cada faixa, visando determinar os grupos de bandas de potência que mais contribuíram para a definição das características mais relevantes ao modelo. Comparando os resultados obtidos, percebe-se que nos três treinamentos o intervalo de maior influência nessas escolhas compreendia bandas de frequência contidas entre 4800 Hz e 6400 Hz. Por outro lado, fica evidente que quanto maior for a dimensão dos dados de entrada, mais atenuada fica essa influência. Analisando de modo mais geral, em intervalos de frequência incluídos entre 0 Hz e 8000 Hz, pode-se considerar que o patamar de influência se mantém similar para diferentes números de bandas de frequência na entrada. A única exceção ocorre na análise de importância de 128 *features*, na qual a proporção indicada para o intervalo de frequência que abrange bandas entre 11 200 Hz a 12 800 Hz atingiu valores acima de 10%.

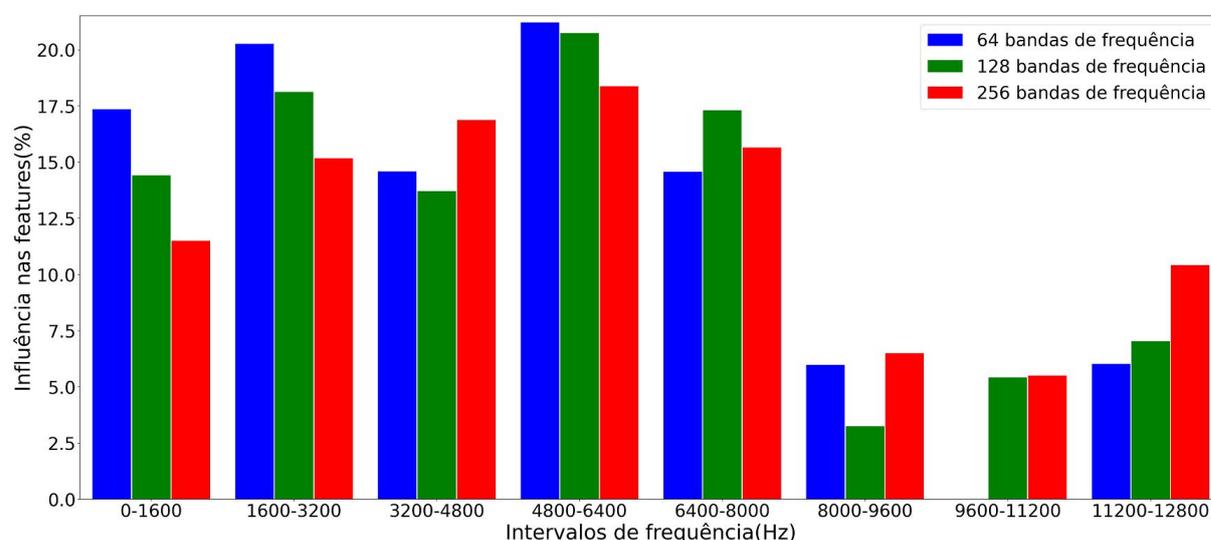
Analisando a Figura 42, que compara a influência das faixas de frequência para diferentes divisões de bandas, fica mais evidente a predominância dos cinco primeiros intervalos de frequência, com porcentagens sempre acima de 10%, mesmo considerando todos os treinamentos. Também fica perceptível o crescimento da influência referente a frequências de 11 200 Hz a 12 800 Hz, indicada pela análise de importância de *features* com base na entrada organizada em 256 bandas de frequência.

Tabela 4 – Comparação da influência dos intervalos de frequência na análise de importância de *features* dos treinamentos realizados.

Intervalos de frequência	Espaço latente de 32 <i>features</i>	Espaço latente de 64 <i>features</i>	Espaço latente de 128 <i>features</i>
0 Hz a 1600 Hz	17,3%	14,4%	11,5%
1600 Hz a 3200 Hz	20,3%	18,1%	15,2%
3200 Hz a 4800 Hz	14,6%	13,7%	16,9%
4800 Hz a 6400 Hz	21,2%	20,7%	18,4%
6400 Hz a 8000 Hz	14,6%	17,3%	15,6%
8000 Hz a 9600 Hz	06,0%	03,3%	06,5%
9600 Hz a 11 200 Hz	00,0%	05,4%	05,5%
11 200 Hz a 12 800 Hz	06,0%	07,1%	10,4%
0 Hz a 12 800 Hz	100%	100%	100%

Fonte: autor

Figura 42 – Comparação entre as influências na escolha de *features* contidas no espaço latente dos três treinamentos finais.



Fonte: autor

Para concluir, pode-se considerar que o resultado mais recorrente quanto ao padrão das *features* contidas no espaço latente realmente foi a importância das bandas contidas entre 0 Hz a 8000 Hz na reconstrução por autoencoder. Como o modelo considerou esses intervalos como os mais importantes para destacar o perfil dos dados de entrada, essa informação pode direcionar para que o monitoramento desses equipamentos, no contexto de medições de vibração, seja mais focado nessa faixa de frequências.

## 5 CONSIDERAÇÕES FINAIS

Este trabalho cumpriu com os objetivos propostos ao desenvolver um método de seleção de *features* e redução de dimensionalidade dos dados de vibração coletados em ensaios de amaciamento de compressores herméticos de refrigeração. O autoencoder projetado demonstrou capacidade de reconstruir, de maneira satisfatória, o perfil das medições pré-processadas em bandas de potência, utilizando uma versão reduzida dos dados de entrada com base nas *features* selecionadas. Os treinamentos realizados utilizando os dados adquiridos em ensaios com o compressor do modelo A mostraram um erro relativo médio em torno de 20%, fato que serve como prova de conceito com relação à competência do modelo tanto na análise do espaço latente quanto na reconstituição da dimensão original dos dados. Os resultados obtidos podem servir como base para trabalhos futuros visando minimizar o erro de reconstrução.

### 5.1 CONCLUSÃO

A familiarização com a natureza dos dados referentes às medições auxiliaram na elaboração de um algoritmo de pré-processamento eficiente, visando contribuir para a melhora de desempenho do modelo ao disponibilizar conjuntos de dados de entrada padronizados e condizentes com os perfis dos dados de vibração no domínio da frequência.

Para a escolha dos parâmetros de configuração do autoencoder foram utilizados ajustes manuais baseados no entendimento da estrutura dos dados disponíveis, somado ao auxílio de métodos de otimização aleatória a partir de faixas de valores obtidas em testes críticos. A métrica utilizada para avaliar escolhas de configuração foi o erro médio percentual absoluto, que é a função perda que melhor se adequou ao contexto do trabalho, avaliando o erro de reconstrução no treinamento do autoencoder, que foi considerado o indicador mais importante para avaliar o desempenho do modelo.

Considerando os treinamentos finais realizados, observou-se que o autoencoder encontrou mais dificuldades na generalização com dados de entrada organizados em 256 bandas de frequência. Consequentemente, a reconstrução desse padrão de dados afetou o desempenho do modelo, resultando em desvios relativos consideravelmente maiores em comparação com as outras configurações de pré-processamento utilizadas.

Com relação à análise das *features* contidas no espaço latente, foi possível extrair informações importantes referentes ao perfil dos dados de vibração. O modelo atribuiu pesos fundamentados na dimensão da entrada de cada cenário de pré-processamento. A partir desses pesos, foi selecionada a metade com os valores mais altos para correlacionar essas características com as faixas de frequência mais influentes na generalização desses dados.

Concluiu-se que, por conta do padrão das *features* selecionadas, existe um intervalo de frequência comprovadamente mais influente no reconhecimento do perfil desses dados pelo autoencoder. Bandas de potência situadas entre 0 Hz e 8000 Hz são responsáveis pela atribuição dos maiores pesos e podem ser consideradas como características mais importantes do conjunto de dados de entrada. Dessa forma, possibilita-se uma análise mais aprofundada com relação à resposta em frequência da vibração do compressor, visando no futuro contribuir como ferramenta adicional tanto em métodos de monitoramento do processo de amaciamento quanto na predição de falhas em compressores.

Por fim, este projeto ressalta que técnicas embasadas em inteligência artificial devem ser exploradas com o objetivo de otimizar a análise de dados de monitoramento de dispositivos, como os compressores herméticos alternativos. Os resultados deste trabalho permitem que técnicas mais avançadas, considerando também um maior custo computacional, sejam aplicadas por causa da redução dimensional dos dados. Esse tipo de abordagem pode contribuir para avanços tecnológicos, capazes de viabilizar melhorias em desempenho, durabilidade e outros fatores significativos para aprimorar a qualidade desses equipamentos.

## 5.2 SUGESTÕES PARA TRABALHOS FUTUROS

Com base nos conhecimentos adquiridos na elaboração deste projeto e na interpretação dos resultados detalhados no presente documento, foram consideradas sugestões de aprimoramento e estudos mais avançados, com o intuito de otimizar o desempenho da técnica utilizada visando o desenvolvimento de trabalhos futuros. A seguir, são apresentadas algumas sugestões:

- avaliar alternativas de implementação de autoencoders mais complexos, tais como modelos em paralelo, comumente chamados de *ensembles* de autoencoders;
- realizar o pré-processamento de bandas de frequência utilizando intervalos logarítmicos, baseando a divisão de frequências em escala de oitavas;
- executar testes e análises utilizando técnicas de otimização mais complexas para a escolha dos hiperparâmetros, como algoritmos genéticos e otimização bayesiana;
- aplicar os método desenvolvido em diferentes modelos de compressores.

## REFERÊNCIAS

ABBOT, E. J.; FIRESTONE, F. A. Specifying surface quality. **Mechanical Engineering**, v. 55, n. 9, p. 569–572, 1933.

AGGARWAL, Charu C. **Neural networks and deep learning**. New York, EUA: Springer, 2018. v. 10, p. 3.

AGUIAR, Daniel; AGUIAR, Paulo. Percepções e redes neuronais artificiais. **Revista de Ciência Elementar**, Casa das Ciências, v. 10, n. 1, 2022.

ANTLER, Morton. Processes of metal transfer and wear. **Wear**, Elsevier, v. 7, n. 2, p. 181–203, 1964.

BENGIO, Yoshua; LAMBLIN, Pascal; POPOVICI, Dan; LAROCHELLE, Hugo. Greedy layer-wise training of deep networks. **Advances in Neural Information Processing Systems**, v. 19, 2006.

BLAU, Peter J. **Friction science and technology: from concepts to applications**. Tennessee, EUA: CRC press, 2008.

BLAU, Peter J. On the nature of running-in. **Tribology international**, Elsevier, Tennessee, EUA, v. 38, n. 11-12, p. 1007–1012, 2005.

BLAU, Peter J.; KOMANDURI, Ranga. Friction and wear transitions of materials: Break-in, run-in, and wear-in. Tennessee, EUA, 1990.

BLOCH, Heinz P; HOEFNER, John J. **Reciprocating compressors: operation and maintenance**. Germany: Elsevier, 1996.

BOLES, M.; ÇENGEL, Y. **Termodinâmica**. 7. ed. São Paulo: AMGH Editora, 2013.

BOUDIAF, Adel; DJEBALA, Abederrazek; BENDJMA, Hocine; BALASKA, Adel; DAHANE, Amine. A summary of vibration analysis techniques for fault detection and diagnosis in bearing. *In*: ANNABA UNIVERSITY. 2016 8th International Conference on Modelling, Identification and Control (ICMIC). Algeria: IEEE, 2016. P. 37–42.

BROWNLEE, Jason. How to prepare your data for machine learning in python with scikit-learn. **Machine Learning Mastery**, v. 18, 2016.

BUITINCK, Lars *et al.* API design for machine learning software: experiences from the scikit-learn project. *In: ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. UK: Scikit-learn, 2013. P. 108–122.

CHANDRASHEKAR, Girish; SAHIN, Ferat. A survey on feature selection methods. **Computers & Electrical Engineering**, Elsevier, v. 40, n. 1, p. 16–28, 2014.

DAGILIS, V.; VAITKUS, L. Experimental investigations and analysis of compressor's friction losses. **Mechanics**, v. 79, n. 5, p. 28–35, 2009.

DOSSAT, Roy J. **Princípios de refrigeração**. Louisiana, EUA: Hemus, 2004.

EISSENBERG, D. M.; HAYNES, H. D. Motor-current signature analysis. **ASM International, ASM Handbook**, v. 18, p. 313–318, 1992.

GUYON, Isabelle; ELISSEEFF, André. An introduction to variable and feature selection. **Journal of Machine Learning Research**, v. 3, Mar, p. 1157–1182, 2003.

HANLON, Paul C. **Compressor handbook**. New York, EUA: McGraw-Hill Professional, 2001.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome H.; FRIEDMAN, Jerome H. **The elements of statistical learning: data mining, inference, and prediction**. New Jersey, EUA: Springer, 2009. v. 2.

HAYKIN, Simon. **Neural Networks and Learning Machines**. 3. ed. Ontario, Canada: Pearson Education, 2009.

HAYKIN, Simon S.; VAN VEEN, Barry. **Sinais e Sistemas**. Ontario, Canada: Bookman, 2001.

HENKLEIN, Pedro. **Análise de ensaios de desempenho de compressores herméticos durante o transitório**. 2006. Diss. (Mestrado) – Florianópolis, SC.

HINTON, Geoffrey E.; SALAKHUTDINOV, Ruslan R. Reducing the dimensionality of data with neural networks. **Science**, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006.

HUNDY, Guy F.; TROTT, Albert Runcorn; WELCH, T. C. **Refrigeration and air-conditioning**. London, UK: Butterworth-Heinemann, 2008.

IBGE. **Pesquisa nacional por amostra de domicílios: síntese de indicadores 2015**. Rio de Janeiro: IBGE, 2016.

IGOR'V, Kragel'skij; DOBYČIN, Michail N; KOMBALOV, Vjačeslav S. **Friction and wear: calculation methods**. Latvia: Pergamon Press, 1982.

IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *In*: UNIVERSITY OF CALIFORNIA. INTERNATIONAL Conference on Machine Learning. California, EUA: PMLR, 2015. P. 448–456.

KERAS. **Keras - Layer activation functions**. [S.l.: s.n.]. Acesso em 1 jun. 2023. Disponível em: <https://keras.io/api/layers/activations/>.

KHALIFA, HE. Break-in Behavior of Scroll Compressors, 1996.

KHONSARI, M. M.; GHATREHSAMANI, Sahar; AKBARZADEH, Saleh. On the running-in nature of metallic tribo-components: A review. **Wear**, Elsevier, v. 474, p. 203871, 2021.

LALANNE, Michel; FERRARIS, Guy. **Rotordynamics prediction in engineering**. Quebec, Canada: John Wiley & Sons, 1998. v. 2.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **Nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LEON F. DE CARVALHO, André Ponce de. **Redes Neurais Artificiais**. São Paulo, Brasil: Instituto de Ciências Matemáticas e de Computação (ICMC) - Universidade de São Paulo (USP). Acesso em 1 jun. 2023. Disponível em: <https://sites.icmc.usp.br/andre/research/neural/>.

LIU, Huan; YU, Lei. Toward integrating feature selection algorithms for classification and clustering. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 17, n. 4, p. 491–502, 2005.

LOCKWOOD, F. E.; DALLEY, R. Lubricant analysis. **ASM International, ASM Handbook**, v. 18, p. 299–312, 1992.

MANHERTZ, Gabor; BERECZKY, Akos. STFT spectrogram based hybrid evaluation method for rotating machine transient vibration analysis. **Mechanical Systems and Signal Processing**, Elsevier, v. 154, p. 107583, 2021.

MARTIN, HR. Vibration analysis, in: Friction, Lubrication, and Wear Technology. **ASM Handbook**, ASM, v. 18, p. 293, 1992.

MCCULLOCH, Warren S.; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, v. 5, p. 115–133, 1943.

MÜLLER, Meinard. **Fundamentals of music processing: Audio, analysis, algorithms, applications**. Nürnberg, Germany: Springer, 2015. v. 5.

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

PRATA, Alvaro T.; BARBOSA JR, Jader R. Role of the thermodynamics, heat transfer, and fluid mechanics of lubricant oil in hermetic reciprocating compressors. **Heat Transfer Engineering**, Taylor & Francis, v. 30, n. 7, p. 533–548, 2009.

ROSENBLATT, Frank. **The perceptron, a perceiving and recognizing automaton Project Para**. New York, EUA: Cornell Aeronautical Laboratory, 1957.

STACHOWIAK, Gwidon W.; BATCHELOR, Andrew W. **Engineering tribology**. Australia: Butterworth-heinemann, 2013.

STEWART, Matthew. **Comprehensive introduction to autoencoders**. Massachusetts, EUA: Harvard University, 2019. Acesso em 8 jun. 2023. Disponível em: <https://towardsdatascience.com/generating-images-with-autoencoders-77fd3a8dd368>.

STOECKER, Wilbert F.; JONES, Jerold W. **Refrigeração e ar condicionado**. Illinois, EUA: McGraw-Hill, 1985.

THALER, Gabriel. **Desenvolvimento de métodos não invasivos para avaliação do processo de amaciamento de compressores herméticos alternativos**. 2021. Diss. (Mestrado) – Florianópolis, Brasil.

THALER, Gabriel. Projeto e implementação de uma bancada experimental para determinação do perfil de amaciamento em compressores herméticos alternativos. Florianópolis, SC., 2019.

THALER, Gabriel; NUNES, Nicolas A.; NASCIMENTO, Ahryman S. B. de S.; PACHECO, Antonio L. S.; FLESCH, Rodolfo C. C. Aplicação de aprendizado não supervisionado para identificação não destrutiva do amaciamento em compressores. *In: XV Simpósio Brasileiro de Automação Inteligente*. Rio Grande: SBA, 2021.

ZAMBRANO, O. A.; MUÑOZ, E. C.; RODRIGUEZ, S. A.; CORONADO, J. J. Running-in period for the abrasive wear of austenitic steels. **Wear**, Elsevier, v. 452, p. 203298, 2020.