



**Bárbara Valeria de Abreu Lavôr**

**Avaliação de parâmetros estruturais baseada  
em dados: explorando ondas ultrassônicas  
guiadas por meio de modelagem computacional  
e aprendizado de máquina**

**Tese de Doutorado**

Tese apresentada como requisito parcial para obtenção do grau de Doutor pelo Programa de Pós-graduação em Engenharia Mecânica da PUC-Rio.

Orientador : Prof. Arthur Martins Barbosa Braga  
Co-orientador: Dra. Paula Aida Sesini

Rio de Janeiro  
Abril de 2024



**Bárbara Valeria de Abreu Lavôr**

**Avaliação de parâmetros estruturais baseada  
em dados: explorando ondas ultrassônicas  
guiadas por meio de modelagem computacional  
e aprendizado de máquina**

Tese apresentada como requisito parcial para obtenção do grau de Doutor pelo Programa de Pós-graduação em Engenharia Mecânica da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

**Prof. Arthur Martins Barbosa Braga**

Orientador

Departamento de Engenharia Mecânica – PUC-Rio

**Dra. Paula Aida Sesini**

Co-orientador

Departamento de Engenharia Mecânica – PUC-Rio

**Prof. Helon Vicente Hultmann Ayala**

Pontifícia Universidade Católica – PUC-Rio

**Dr. Roberth Waldo Angulo Llerena**

Ouro Negro Tecnologias em Equipamentos Industriais S.A. –

Ouro Negro

**Prof. Ricardo Leiderman**

Universidade Federal Fluminense – UFF

**Dr. Sergio Ricardo Kokay Morikawa**

Cenpes – Petrobrás-S.A.

Rio de Janeiro, 19 de Abril de 2024

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

**Bárbara Valeria de Abreu Lavôr**

Ficha Catalográfica

Lavor, B.V.A.

Avaliação de parâmetros estruturais baseada em dados: explorando ondas ultrassônicas guiadas por meio de modelagem computacional e aprendizado de máquina / Bárbara Valeria de Abreu Lavôr; orientador: Arthur Martins Barbosa Braga; co-orientador: Paula Aida Sesini. – Rio de Janeiro: PUC-Rio, Departamento de Engenharia Mecânica, 2024.

v., 103 f: il. ; 30 cm

Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica.

Inclui bibliografia

1. Avaliação Estrutural;. 2. Ensaios Não Destrutivos;. 3. Ondas Ultrassônicas Guiadas;. 4. Método SAFE;. 5. Aprendizado de Máquina;. 6. Aprendizado Supervisionado.. I. Braga, Arthur M. B.. II. Sesini, Paula. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Mecânica. IV. Título.

CDD: 621

## Agradecimentos

Primeiramente agradeço a Deus por ter chegado a reta final do doutorado. Acredito que escrever e defender esta tese foi o desafio mais complicado da minha vida. Mentalmente exaustivo e socialmente subestimado, concluir este ciclo sozinha não seria possível, por isso agradeço de coração a todos os envolvidos.

Devo gratidão aos meus familiares, principalmente minha mãe e minha irmã que fazem parte da minha vida, me apoiam e, por amá-las, tenho um motivo a mais para seguir em frente. Profunda gratidão também ao meu amado noivo, Thales, que é um companheiro de vida admirável, me incentiva e apoia, além de transmitir segurança e leveza.

Obrigada, Prof. José Luiz Freire, por ter acreditado em mim, ter sido um grande mestre e por ter propiciado uma bolsa de projeto, quando eu entrei sem bolsa no doutorado. Não seria possível seguir sem condições financeiras.

Um muito obrigada especial ao Prof. Helon que de maneira despretensiosa e altruísta, me deu conselhos, dicas, ensinamentos e orientação, mesmo sem eu ser oficialmente sua aluna e nunca ter cursado uma matéria com ele. Essa tese não existiria sem a sua intervenção. Obrigada de verdade, professor.

Também devo gratidão ao meu orientador, o Prof. Arthur Braga, uma pessoa que admiro muito por sua inteligência, empreendedorismo, compreensão e disposição em ajudar a todos. E minha co-orientadora a Dra. Paula Sesini, que embarcou nesse projeto e ficou ao meu lado mesmo quando não concordávamos com alguns termos, mas foi fundamental com idéias, conselhos e intervenções.

Obrigada, Prof. Alan Kubrusly, Daniel Boechat e Pedro Henrique Domingues por toda ajuda nas publicações. Aos amigos Eduardo Goicoechea, Claudio Tellez, Ingrid Pires, Mariana Gomes, Marília Assad, Gabriela Martins e Juan Hidalgo pela ajuda nos artigos, textos, estudos e desabafos.

Repito o nome do Eduardo e acrescento o da Chef Val Santos, que compartilharam um teto comigo e o transformaram em lar. Obrigada por tanto, amigos, vocês são parte disso!

Agradeço também aos membros da banca avaliadora, ao Departamento de Engenharia Mecânica da PUC-Rio e todos os seus trabalhadores, destacando Carlucio Souza e Carina Beline e à agência de fomento que me propiciou bolsa.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

## Resumo

Lavor, B.V.A.; Braga, Arthur M. B.; Sesini, Paula. **Avaliação de parâmetros estruturais baseada em dados: explorando ondas ultrassônicas guiadas por meio de modelagem computacional e aprendizado de máquina.** Rio de Janeiro, 2024. 103p. Tese de Doutorado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Métodos não destrutivos desempenham um papel fundamental na análise de integridade de componentes civis e mecânicos. Este trabalho apresenta uma abordagem integrativa que pode ser aplicada em avaliações estruturais, envolvendo o estudo de ondas ultrassônicas guiadas com técnicas de aprendizado de máquina, juntamente com métodos alternativos para ampliar o conjunto de dados necessários para treinamento dos modelos de predição. É proposta uma metodologia para a estimativa de tensão em placas, incorporando dados sintéticos gerados por interpolação linear a partir de resultados experimentais. Paralelamente, utilizando o *Semi-Analytical Finite Element Method* (SAFE), foi examinada a propagação de ondas em guias cilíndricos, comparando resultados com simulações numéricas e posteriormente, aplicando o método SAFE para gerar curvas de dispersão que servem de dados para alimentar uma rede neural para predição de espessura nominal de dutos. Este estudo contribui para a eficiência e precisão das metodologias aplicadas em integridade estrutural, englobando abordagens de dados experimentais e simulados computacionalmente.

## Palavras-chave

Avaliação Estrutural; Ensaios Não Destrutivos; Ondas Ultrassônicas Guiadas; Método SAFE; Aprendizado de Máquina; Aprendizado Supervisionado.

## Abstract

Lavor, B.V.A.; Braga, Arthur M. B. (Advisor); Sesini, Paula (Co-Advisor). **Data-Driven Assessment Structural Parameters: Exploring Ultrasonic Guided Waves through Computational Modeling and Machine Learning**. Rio de Janeiro, 2024. 103p. Tese de doutorado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

Non-destructive methods play a fundamental role in analyzing the integrity of civil and mechanical components. This work presents an integrative approach that can be applied in structural assessments, involving the study of ultrasonic waves guided with machine learning techniques, together with alternative methods to expand the data set necessary for training prediction models. A methodology for estimating stress in plates is proposed, incorporating synthetic data generated by linear interpolation from experimental results. In parallel, using the *Semi-Analytical Finite Element Method* (SAFE), we examined wave propagation in cylindrical guides, comparing results with numerical simulations and subsequently, applying the SAFE method to generate dispersion curves that serve as data to feed a neural network for predicting nominal pipeline thickness. This study contributes to the efficiency and precision of methodologies applied in structural integrity, encompassing experimental and computationally simulated data approaches.

## Keywords

Structural Assessment; Non-Destructive Testing; Ultrasonic Guided Waves; SAFE Method; Machine Learning; Supervised Learning.

## Sumário

1	Introdução	13
1.1	Revisão Bibliográfica	15
1.1.1	Ensaaios Não Destrutivos e Inspeção Acústica	15
1.1.2	Modelagem Computacional de Ondas Guiadas	16
1.1.3	Aprendizado de Máquina na Análise de Ondas Ultrassônicas	18
1.2	Objetivos	22
1.3	Organização da tese	23
2	Conceitos básicos	24
2.1	Propagação de ondas elásticas em sólidos	24
2.1.1	Equação de Navier-Cauchy	24
2.1.2	Teorema de Helmholtz	30
2.1.3	Equações de Onda	31
2.2	Sistemas de Coordenadas	32
2.2.1	Coordenadas Cilíndricas	32
2.3	<i>Machine Learning</i>	34
2.3.1	Aprendizado Supervisionado: Uma Abordagem Fundamentada em Dados	35
2.3.1.1	Ridge regression	36
2.3.1.2	Support vector regression	36
2.3.1.3	Decision tree regression	37
2.3.1.4	K-nearest neighbors regression	37
2.3.1.5	Random Forest Regression	37
2.3.1.6	Gradient boosting regressor	38
2.3.1.7	Redes Neurais: uma abordagem inspirada no cérebro	38
2.3.1.8	Multi-layer Perceptron	39
3	Estimativa de tensão de tração em placas	40
3.1	Construção do modelo e validação	40
3.2	Dados experimentais	41
3.3	Abordagem proposta	42
3.4	Sinais de ondas ultrassônicas sintéticas através de interpolação	44
3.5	Modelos de aprendizado de máquina	45
3.6	Resultados	48
3.7	Considerações sobre este capítulo	59
4	Método SAFE no cálculo de curvas de dispersão em cilindros	61
4.1	Formulação do Problema	61
4.2	Metodologia e Resultados	66
4.3	Considerações sobre este capítulo	68
5	Rede neural para predição da espessura de dutos	69
5.1	Problema Analisado	69
5.2	Metodologia	70

5.2.1	Rede neural usada	71
5.2.2	Arquitetura Proposta	72
5.3	Resultados e Discussão	73
5.3.1	Seleção de Hiperparâmetros	74
5.4	Considerações finais desde capítulo	76
<b>6</b>	<b>Aprendizagem por transferência</b>	<b>78</b>
6.1	Abordagens de aprendizagem por transferência	78
6.1.1	Visual Geometry Group	79
6.1.2	Redes Residuais	79
6.1.3	MobileNet	79
6.1.4	MobileNetV2	80
6.1.5	Xception	80
6.2	Método	80
6.2.1	Redimensionamento em Imagem	81
6.3	Discussão e resultados	83
6.4	Considerações finais desde capítulo	87
<b>7</b>	<b>Conclusão</b>	<b>88</b>
7.1	Implicações e Futuras Direções	89
7.2	Considerações Finais	89
	Referências bibliográficas	<b>90</b>

## Lista de figuras

Figura 2.1	Mudança de configuração de um corpo arbitrário.	24
Figura 3.1	Croqui da bancada experimental: A - Placa de alumínio; B - Transdutor emissor piezoelétrico; C - Extensômetro; D - Transdutor receptor piezoelétrico; E - Suporte de montagem e; F - Haste roscada.	43
Figura 3.2	Visão geral do método proposto.	44
Figura 3.3	Número de componentes principais para os conjuntos de dados sintéticos.	49
Figura 3.4	Erro Quadrático Médio (MSE) para diferentes modelos à medida que o tamanho dos conjuntos de dados sintéticos aumenta (gráfico em escala logarítmica para melhor visualização).	50
Figura 3.5	Gráfico de comparação do Coeficiente de Determinação ( $R^2$ ) entre diferentes modelos de aprendizado de máquina para os 11 conjuntos de dados sintéticos.	51
Figura 3.6	Gráfico de comparação do Erro Médio Absoluto (MAE) entre diferentes modelos de aprendizado de máquina em diferentes conjuntos de dados sintéticos.	52
Figura 4.1	Cilindro com raio interno de $\mathbf{r_1 = 0,30\ m}$ e raio externo de $\mathbf{r_2 = 0,33\ m}$ .	62
Figura 4.2	Funções de forma adotadas.	63
Figura 4.3	A imagem (a) mostra a solução analítica e (b) mostra as curvas de dispersão calculadas pelo método SAFE.	66
Figura 4.4	No gráfico, em (a) são mostradas as curvas de dispersão obtidas com COMSOL e em (b) as curvas de dispersão obtidas com o método SAFE no Matlab.	68
Figura 5.1	Resumo de como foi realizado este estudo para avaliar a espessura nominal de dutos utilizando redes neurais treinadas com curvas de dispersão obtidas pelo método SAFE.	69
Figura 5.2	O modelo de cilindro livre de tensões que foi usado.	70
Figura 5.3	As curvas de dispersão para um cilindro foram calculadas usando o método SAFE.	70
Figura 5.4	Representação esquemática ilustrando o procedimento de pré-processamento e a arquitetura da CNN proposta para avaliação da espessura nominal de dutos.	73
Figura 5.5	LOSS de treinamento e teste.	75
Figura 5.6	MAE para treinamento e teste.	76
Figura 5.7	Diagrama de dispersão das previsões do treinamento	76
Figura 5.8	Diagrama de dispersão dos testes	77
Figura 6.1	Sinal experimental e sua imagem correspondente.	83
Figura 6.2	Sinal sintético e sua respectiva forma remodelada em imagem.	83

Figura 6.3	Sinais de onda remodelados.	84
Figura 6.4	Comparação entre os modelos em termos de MSE.	86

## Lista de tabelas

Tabela 3.1	Configurações de hiperparâmetros para os modelos usando validação cruzada e pesquisa aleatória	48
Tabela 3.2	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>15</b> casos.	53
Tabela 3.3	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>29</b> casos.	53
Tabela 3.4	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>58</b> casos.	54
Tabela 3.5	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>116</b> casos.	54
Tabela 3.6	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>232</b> casos.	55
Tabela 3.7	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>346</b> casos.	55
Tabela 3.8	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>462</b> casos.	56
Tabela 3.9	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de <b>924</b> casos.	56
Tabela 3.10	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de <b>1848</b> casos.	57
Tabela 3.11	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de <b>5082</b> casos.	57
Tabela 3.12	Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de <b>10164</b> casos.	58
Tabela 4.1	Parâmetros do cilindro	67
Tabela 5.1	Métricas de regressão obtidas com espessura prevista dos dutos	74
Tabela 6.1	Valores de MSE para os diferentes modelos, onde FT corresponde a abreviatua de <i>fine-tuning</i> .	86

Tabela 6.2 As informações desta tabela foram integralmente reproduzidas do trabalho citado, mostrando as estatísticas de erro quadrático médio para diferentes modelos testados pelos autores. O valor de MSE do modelo CNN está destacado em negrito. 87

# 1

## Introdução

No início do ano de 2023 um inconveniente diplomático foi amplamente noticiado pela mídia: Um balão meteorológico chinês sobrevoou o espaço aéreo norte americano. A China pediu desculpas e afirmou que o objeto era usado para fins puramente científicos e civis, mas os Estados Unidos desconfiavam tratar-se de espionagem [1, 2]. Aparentemente, o programa dos balões meteorológicos foi suspenso após o ocorrido e os países seguiram tentando estabilizar a relação que já andava tensa por motivos políticos e comerciais [3].

O intrigante nessa história é que a acusação americana pode fazer sentido e, como sugere um velho ditado popular: “o bom julgador por si julga aos outros”.

Durante a Guerra Fria, no final dos anos de 1940, os EUA se empenhavam em investigar o que os soviéticos estavam fazendo [4]. Foi quando um geofísico inventou uma boa maneira de descobrir se estavam testando bombas nucleares. Os instrumentos utilizados para isso eram balões meteorológicos equipados com microfones para captar o som das explosões [5]. Mas, como isso poderia acontecer se os balões eram lançados nos EUA e os testes supostamente ocorriam em um território próximo a Rússia? A resposta para essa pergunta está relacionada com a beleza existente na propagação de ondas acústicas.

Estes balões meteorológicos vão até a troposfera, que fica a 15km de altitude [6, 7]. Conforme o ar vai ficando mais frio, o som tende a desviar nessa direção. Acima de 15km, o ar volta a esquentar devido à radiação ultravioleta. Ou seja, existe uma camada na atmosfera, onde o som pode se propagar por longas distâncias emparelhado por uma camada de ar frio e uma camada de ar mais quente. Tal condição favorece o passeio de ondas acústicas guiadas por quilômetros até que sejam captadas por receptores, que nesse caso, são os microfones instalados nos balões. Dependendo do que fosse captado, poderia haver a indicação de uma explosão de teste nuclear. Essa espécie de “sanduíche de ar” pelo qual ondas “viajam” pode ser explicado com o conceito de um guia de ondas, que é uma estrutura cujas superfícies de contorno confinam total ou parcialmente as ondas eletromagnéticas ou acústicas de forma que a energia seja transportada (guiada) em uma única direção [8, 9].

Fora seu curioso papel nas complexas relações internacionais, não é só

em grandes projetos de espionagem que a física relacionada a propagação de ondas é aproveitada. Ela também desempenha um papel fundamental em aplicações cotidianas. Isso pode ser visto nos cabos coaxiais dos aparelhos de televisão, nas imagens de ultrassom dos exames médicos, nas fibras óticas que transmitem dados de internet, etc. Além disso, esse conceito pode ser habilmente aplicado na engenharia. A capacidade de entender e analisar o comportamento de materiais e componentes sem causar danos é crucial para garantir a segurança e a durabilidade das estruturas. A utilização de ensaios não destrutivos, como testes com ondas ultrassônicas guiadas, tornou-se uma ferramenta valiosa nesse contexto. Esses métodos permitem a detecção precisa de falhas e tensões, contribuindo para a manutenção preditiva e preservando a integridade das estruturas [10].

No âmbito deste trabalho, os conceitos relacionados a ondas ultrassônicas, métodos numéricos e modelos de aprendizado de máquina são explorados devido o seu potencial de aplicação em técnicas não destrutivas, o que possui grande valia no que se refere a avaliação estrutural de componentes civis e mecânicos. Ao integrar métodos semi-analíticos, como o *Semi-Analytical Finite Element Method* (SAFE), com abordagens inovadoras baseadas em *machine learning* (ML), busca-se expandir as perspectivas da avaliação estrutural. As diferentes abordagens de ML, como ferramentas de predição, oferecem boas oportunidades para aprimorar a precisão e eficiência dessas metodologias.

Com este cenário em mente, é possível delinear a direção desse trabalho. Primeiramente, com uma metodologia baseada em aprendizado de máquina para estimar tensão em placas, incorporando dados sintéticos gerados por interpolação linear de resultados experimentais. A validação desses métodos propostos será realizada comparando os resultados obtidos experimentalmente. Além disso, também é abordado o estudo da propagação de ondas ultrassônicas em guias de onda cilíndricos utilizando métodos analíticos e numéricos computacionais, destacando o papel do método SAFE. Em seguida, é implementada uma abordagem, incorporando redes neurais para avaliação da espessura nominal de dutos.

A interseção entre ondas acústicas guiadas, métodos semi-analíticos e *machine learning* representa um avanço significativo na busca por avaliações estruturais mais precisas e adaptativas. Explorar essas conexões multidisciplinares, não é apenas compreender a física subjacente, mas também impulsionar inovações práticas e outros estudos que possam transformar a maneira como são abordados os desafios da integridade estrutural.

## 1.1

### Revisão Bibliográfica

Os próximos parágrafos, destacam a importância dos ensaios não destrutivos, da inspeção acústica, da modelagem computacional e *machine learning* na avaliação de parâmetros estruturais em setores como engenharia, indústria aeroespacial e de petróleo e gás. Esses pontos são abordados por meio de citações de alguns livros e publicações recentes, ajudando a fornecer um contexto para os tópicos discutidos nesta tese.

#### 1.1.1

### Ensaio Não Destrutivo e Inspeção Acústica

Os ensaios não destrutivos (END ou em inglês *Nondestructive testing - NDT*), incluindo a inspeção acústica, são ferramentas essenciais para avaliar a integridade estrutural sem comprometer o funcionamento de sistemas em operação. Na indústria do petróleo, por exemplo, quando um poço atinge o seu limite, e o ritmo de produção deixa de ser rentável, as atividades desse poço devem ser encerradas [11]. O que é chamado de operação de tamponamento e abandono de poços (do inglês *Plug and Abandonment* ou P&A). Nesse caso, a inspeção acústica é utilizada para verificar a integridade dos componentes, permitindo identificar danos por meio da análise dos sinais de ondas mecânicas propagantes.

Em dutos off-shore de transporte de petróleo e derivados, existem vários tipos de danos que podem ocorrer, como colapso localizado por flambagem. Esse tipo de falha muitas vezes é causada por ovalização de fábrica, que pode ser agravada durante a instalação, ou operação [12], pressão externa e variações de pressão interna, geração de amassado por impactos com objetos (como âncora ou equipamento lançado de embarcação que opera na área, por exemplo) e desgaste por corrosão ou erosão [13, 14].

Nesse sentido, realizar manutenções e inspeções faz parte da rotina da indústria de petróleo e gás [15–17]. Para isso, são utilizadas técnicas não destrutivas, como inspeções por ultrassom. O desenvolvimento e aprimoramento dessas técnicas ainda é objeto de estudo e avanços tecnológicos, pois permitem testes com o mínimo de interferências e a propagação de ondas ultrassônicas é uma importante ferramenta para medir a integridade de estruturas [18–20].

Um dos capítulos deste trabalho, aborda o estudo de tensões em uma placa de alumínio. A análise de tensão em placas é fundamental nas indústrias aeroespacial, de engenharia estrutural e automotiva [21–23]. A estimativa de tensão em placas de forma precisa é essencial para garantir a integridade

estrutural, prever a vida útil dos componentes e evitar falhas catastróficas [24, 25].

Normalmente, para encontrar tensões em um determinado material, recorre-se a ensaios destrutivos, que consistem em aplicar forças a uma amostra de teste e medir sua deformação [26]. No entanto, este processo exige que a amostra de teste seja separada do produto final, resultando em desperdício de materiais, aumento do tempo necessário e custos mais elevados [27]. Além disso, não pode ser usado para monitoramento da qualidade em tempo real durante a fabricação. Portanto, vários estudos tem sido realizados para estimar a deformação por métodos não destrutivos [28–30].

Técnicas não destrutivas, incluindo testes de ondas guiadas ultrassônicas, desempenham um papel crucial nesse tipo de análise, permitindo a detecção e monitoramento de tensão, preservando a integridade do material [10]. Existe uma técnica chamada *ultrasonic tensile testing*[30] (UTT) que foi usada para estimar curvas tensão-deformação em ligas de alumínio. Os resultados demonstraram que o ultrassom pode fornecer estimativas consistentes de propriedades mecânicas, como limite de escoamento, resistência máxima à tração e alongamento total quando comparados com curvas obtidas por testes destrutivos de tração. Embora esse tipo de teste exija uma série de passos que não são detalhados aqui, como calibração em diferentes configurações de medição, é importante notar que uma abordagem de aprendizado profundo com redes neurais convolucionais e recorrentes é um caminho interessante e uma tendência para realizar esse tipo de análise.

Nos ensaios não destrutivos com técnica acústica, são emitidos pulsos que se propagam pela estrutura. Ao comparar o comportamento das ondas mecânicas, quanto à forma como elas se propagam em um ambiente íntegro, e as características do meio lido a partir do sinal registrado nos receptores, é possível identificar danos nas camadas de revestimento, uma vez que a ocorrência de defeitos influencia o formato das curvas de dispersão e os modos guiados [31]. Porém, resolver os sistemas de equações necessários para traçar as curvas de dispersão requer um certo custo computacional.

### 1.1.2

#### **Modelagem Computacional de Ondas Guiadas**

Um exemplo de guia de ondas complexo é a estrutura de um poço de petróleo. Existe a parede de rocha, a camada cimentícia, o revestimento metálico, o fluido e o duto de produção. A manutenção dessas estruturas, possui grande importância, já que os derivados de petróleo representam grande parte da matriz energética mundial. Suas operações de exploração,

transporte, tamponamento e abandono ao final do ciclo produtivo do poço exigem dedicação ao gerenciamento de riscos e garantia da segurança das atividades [32, 33]. Juntamente com isso, os oleodutos offshore são grandes estruturas que custam centenas de milhões de dólares [34]. Levando em conta fatores como o local onde a linha está instalada, o valor significativo da mercadoria transportada e sua composição, é fácil imaginar a consequência de uma falha que leva à ruína uma estrutura que, além de causar danos ambientais catastróficos, impactos, também podem representar perdas financeiras e até levar à perda de vidas humanas. É importante prevenir esses acidentes e, sobretudo, garantir a integridade física das pessoas.

A modelagem de ondas guiadas em estruturas complexas, como vigas "T", trilhos e tubos de perfuração, representa um desafio devido à falta de soluções analíticas diretas [35–38]. Para encontrar as propriedades do sistema que fazem parte da construção das curvas de dispersão, são utilizados algoritmos numéricos que localizam as raízes da equação do problema de autovalores que permite calcular os modos naturais de vibração. Normalmente esses algoritmos são iterativos e utilizam um passo, que deve ter um tamanho ajustado para encontrar todas as ondas. Se a distância entre os passos for grande, algumas raízes podem ser perdidas. Porém, se as raízes estiverem muito espaçadas, ou repetidas, ao determinar de um passo muito pequeno para a busca, há um custo computacional desnecessário.

Dada a natureza contínua das equações de onda, quando há a impossibilidade de resolver o problema na forma analítica, alguns métodos numéricos são frequentemente usados para discretizar essas equações, resolver o problema de autovalores numericamente e obter as características do sistema. Por exemplo, é comum discretizar as equações com o método dos elementos finitos (MEF), o método das diferenças finitas (FDM) ou o método dos elementos de contorno (BEM). Essas técnicas convencionais geralmente requerem de quatro a seis nós por comprimento de onda para expressar com precisão sua forma de onda ([36]). Consequentemente, para representar a propagação de ondas em guias que são estruturas de grande porte, é necessário um elevado número de nós na direção em que a onda se propaga, o que acarreta longos tempos de processamento e necessidade de muita memória computacional.

Uma abordagem eficaz para reduzir custos computacionais na simulação da propagação de ondas em estruturas complexas é o método SAFE (Semi-Analytical Finite Element). Este método combina solução analítica e elementos finitos, permitindo uma representação precisa da propagação de ondas com um menor custo computacional. A discretização da estrutura em elementos finitos é feita na seção transversal, permitindo encontrar um campo de deslocamentos

aproximado. Ao longo da estrutura, na direção de propagação da onda, é utilizada a solução analítica com a representação harmônica da onda no domínio do tempo [36, 39].

Neste trabalho, é utilizado o método SAFE para a simulação da propagação de ondas em estruturas cilíndricas. É adaptada uma solução analítica harmônica nas direções axial e angular e, o método dos elementos finitos na direção radial. Desta forma, o problema resultante é unidimensional, o que representa uma redução no custo computacional comparado com as abordagens bidimensionais. Na implementação do método SAFE que foi publicada pela autora desta tese [40], houve uma interessante redução no custo computacional para obtenção das curvas de dispersão de um cilindro: utilizando um software comercial de elementos finitos, o tempo de processamento foi superior a 1 dia e 5 horas, contrastando com os 8 minutos de duração do processamento no mesmo computador, utilizando o SAFE implementado em Matlab. Devido ao baixo custo computacional inerente a este método, esta mesma implementação foi utilizada para gerar simulações cujos dados serviram de entrada para o modelo de redes neurais que também é mostrado nessa tese.

As ondas ultrassônicas apresentam pequenas variações em sua velocidade e amplitude de propagação [41] devido a alterações no meio de propagação [42]. No entanto, compreender estas mudanças nos sinais recebidos não é uma tarefa simples. O sinal recebido pode se tornar ainda mais complexo ao lidar com características da estrutura, como defeitos, reflexão ou variações de espessura. Essas características podem levar a fenômenos como espalhamento ou conversão de modo, complicando ainda mais a interpretação do sinal [41, 43].

### 1.1.3

#### **Aprendizado de Máquina na Análise de Ondas Ultrassônicas**

O aprendizado de máquina, incluindo técnicas como redes neurais convolucionais, tem se mostrado eficaz na análise e interpretação de dados obtidos por ensaios não destrutivos, como ultrassom. Estudos recentes demonstram sua aplicação na previsão de propriedades mecânicas, detecção de falhas em equipamentos industriais e análise de adesivos estruturais, oferecendo uma abordagem promissora para aprimorar a análise de ondas ultrassônicas e a detecção de danos.

O progresso nas técnicas de aprendizado de máquina permite a redução da intervenção humana na análise dos defeitos estruturais ou patologias [44–47]. Entre as diversas técnicas que foram desenvolvidas nos últimos tempos, as Redes Neurais Convolucionais (CNN) já demonstraram excelentes capacidades em tarefas como reconhecimento de padrões, visão computacional [48] e

classificação de séries temporais [49], comprovando sua capacidade de lidar com problemas complexos e não lineares [50, 51]. Foi projetada para aprender e extrair automaticamente recursos de imagens, permitindo realizar tarefas como reconhecimento, segmentação e classificação de objetos [52–54]. Combinar o uso de um método semianalítico para obtenção de dados de entrada com aprendizado de máquina para avaliação de amostras pode trazer mais agilidade à análise dos resultados e reduzir a necessidade de fornecer recursos para coleta de dados e montagem de bancadas de testes experimentais [44].

Yasniy *et. al* [55] exploraram a aplicação de técnicas de aprendizado de máquina para prever o diagrama tensão-deformação de uma liga de alumínio em diferentes temperaturas usando dois métodos de aprendizado de máquina: *k-nearest neighbors* (KNN) e *random forest*. Apesar dos erros de previsão (9,6% e 5,9% para as regiões lineares e não lineares, respectivamente, no método dos KNN, e 15% e 13,7% no método da random forest), no geral, este estudo destaca o potencial do aprendizado de máquina na modelagem das propriedades mecânicas dos materiais, enfatizando a necessidade contínua de aprimoramento e expansão dessas técnicas.

Nos últimos anos, métodos inovadores, incluindo *generative adversarial neural networks* (GANs), foram integrados a simulações numéricas para gerar dados sintéticos para testes ultrassônicos em ensaios não destrutivos [56]. Esses métodos abordam a escassez de dados aumentando os conjuntos de dados de treinamento, especialmente para defeitos complexos. Várias técnicas de geração de dados sintéticos, como *generalized point source synthesis* (GPSS), superposição de *gaussian beams* (GB), *pencil method* (PM), métodos numéricos e GANs combinados com dados experimentais, foram explorados. Desafios relacionados aos parâmetros de treinamento GAN e qualidade dos dados foram identificados em [56]. A escassez de dados realistas de defeitos sintéticos para cenários complexos dificulta aplicações mais amplas de aprendizado de máquina.

A abordagem delineada por Alkhalifah *et. al.* [57] aborda o desafio de *covariate shift* [58] entre dados de treinamento sintéticos e dados de aplicativos do mundo real, propondo transformações denominadas “MLReal” para melhorar o desempenho do modelo. Embora sejam bem-sucedidas em tarefas geofísicas e sísmicas, estas transformações podem não se adequar a todos os cenários. Mais investigações são necessárias para validar a aplicabilidade da abordagem em vários domínios de aprendizado de máquina. Além disso, pesquisas focadas em técnicas diretas de geração de dados, como a interpolação, são necessárias para avaliar o desempenho do modelo em contextos do mundo real. Isto enfatiza a necessidade de estudos abrangentes sobre hiperparâmetros

e escolhas de modelos para garantir a eficácia.

Em um estudo publicado recentemente [59], os autores propuseram uma nova estratégia computacional para prognóstico de falhas de uma bomba de engrenagens externa usando ML. A abordagem para geração de dados sintéticos combina simulações CFD com métodos de perturbação e interpolação. Para prognósticos de falhas, dois tipos de algoritmos são utilizados, a saber, algoritmos *Multilayer Perceptron* (MLP) e *Support Vector Machine* (SVM); o primeiro foi considerado superior. O trabalho apresentado por Kahar *et. al.* [60] consiste na otimização de coletores de energia baseados em sistemas microeletromecânicos (em inglês *Micro-Electro-Mechanical Systems* - MEMS) para prever a potência de saída e a frequência natural de operação através de algoritmos de dados sintéticos e ML. Os dados sintéticos foram obtidos através de simulação numérica utilizando o software COMSOL Multiphysics™. Diferentes modelos foram utilizados e observou-se que o Extreme Gradient Boost (XG-Boost) foi o modelo de ML mais adequado para prever frequência e potência com  $R^2$  de 97% e 86%, respectivamente.

Ademais, o aprendizado de máquina pode fornecer suporte para fases preliminares do projeto estrutural. Também pode ser citada a aplicação de SVM, SVR (*Support Vector Regression*) e rede neural de retropropagação para prever as respostas estruturais de placas reforçadas [61], que utilizou uma base de dados obtida através do método numérico Euler-Lagrange acoplado (CEL), ou seja, foram empregados dados sintéticos. A aplicação do algoritmo SVM investigou com sucesso o critério crítico para ruptura de placa reforçada, enquanto os algoritmos SVR e Backpropagation Neural Network produziram resultados promissores para previsão de fratura e deformação plástica.

Em outro estudo recente [62], os autores empregaram técnicas de aprendizado de máquina, particularmente redes neurais de camada oculta, para prever respostas a danos em cenários complexos envolvendo explosões de contato subaquático em placas reforçadas. Essa abordagem captura interações intrincadas entre fluido e estrutura, não linearidades de material e dinâmica geométrica, permitindo previsões precisas de danos. No entanto, o trabalho supracitado destacou o desafio da disponibilidade limitada de dados devido ao custo computacional das simulações, o que afeta o treinamento e a validação do modelo. Da mesma forma, outras técnicas de aprendizado de máquina poderiam ter sido investigadas para comparar e melhorar ainda mais as previsões.

Com o avanço da tecnologia e o crescente volume de dados disponíveis, ML oferece vantagens significativas em relação aos métodos tradicionais de análise e modelagem. Em primeiro lugar, os modelos de aprendizagem automática têm a capacidade de lidar com grandes quantidades de dados de forma efici-

ente e automatizada [63, 64]. À medida que os dados aumentam, seja através de sensores ou simulações computacionais, a capacidade de processar, analisar e extrair informações relevantes torna-se crítica [65]. Nesse sentido, o ML oferece algoritmos e técnicas que permitem rápido processamento e extração de padrões complexos nos dados, contribuindo para a análise de tensões em placas sólidas [66]. Devido à capacidade de aprender com os dados, é possível identificar correlações e relações não lineares que podem estar presentes na análise de tensões mecânicas.

Ao contrário das abordagens tradicionais, baseadas em modelos físicos ou métodos analíticos, o ML pode capturar características complexas e interações entre parâmetros do sistema [64, 67]. Os modelos de aprendizado de máquina também podem lidar com incertezas e variações nos dados. No contexto da análise de tensões em placas, existem vários fatores que podem introduzir incerteza nos dados, como ruído e condições ambientais. Os trabalhos já citados até aqui mostraram resultados promissores ao combinar técnicas de ML com métodos de análise tradicionais, resultando em melhorias na precisão e eficiência da estimativa de tensão.

Uma metodologia que inclui a adoção de redes neurais artificiais para análise computacional de adesivos estruturais em simulações de elementos finitos [68] foi publicada recentemente. As redes neurais são treinadas e avaliadas usando ferramentas de aprendizado de máquina usando caminhos de deformação como dados de entrada. Em cenários de dados de alta dimensão, a utilização de técnicas de extração de características, como PCA (*principal component analysis*) ou LDA (*linear discriminant analysis*), revela-se essencial. Ao mitigar a maldição da dimensionalidade [69–71], essas abordagens não apenas simplificam a interpretação dos dados, mas também melhoram significativamente a eficiência computacional.

Uma outra publicação [72] mostra a utilização da LDA para selecionar características a partir de dados de ultrassom, e um modelo de SVM como classificador linear para avaliar a qualidade das juntas adesivas de epóxi em placas de alumínio. Os resultados alcançaram uma precisão de classificação superior a 97% para amostras com desvios significativos. LDA emergiu como a técnica mais eficiente quando comparada a outros métodos de redução de dimensionalidade. Embora o estudo aponte algumas limitações, como a falta de conexão entre o espaço de características transformado pelo LDA e as características iniciais, fica claro que estabelecer um *pipeline* bem definido com redução de dimensionalidade, treinamento de modelo, testes e otimização de hiperparâmetros é crucial para otimizar o desempenho e garantir resultados mais precisos e interpretáveis.

É possível notar que as abordagens propostas acima mencionadas, [57], [59], [72] e [60], apresentam resultados satisfatórios para previsão de falhas. Porém, nesses estudos, a simulação numérica foi utilizada combinada com outras abordagens para obtenção de dados sintéticos. Assim, o custo computacional é ainda maior quando comparado a outros métodos como interpolação e métodos de ML, sem a necessidade de simulações numéricas [56].

## 1.2 Objetivos

O principal intuito deste trabalho é explorar a aplicação de técnicas de *machine learning* para predição com diferentes composições de conjuntos de dados: experimentais, sintéticos obtidos por interpolação e obtidos computacionalmente com o método SAFE. O objetivo principal de desdobra nas seguintes metas intermediárias:

- Investigar a propagação de ondas ultrassônicas guiadas em tubos cilíndricos utilizando solução analítica;
- Comparar resultados obtidos pelo método SAFE com aqueles da solução analítica;
- Implementar uma rede neural para avaliação da espessura nominal de dutos;
- Utilizar curvas de dispersão obtidas pelo método SAFE como entrada para as redes neurais, explorando a interseção entre métodos semi-analíticos e *machine learning*;
- Propor uma metodologia baseada em aprendizado de máquina para estimar tensão em placas, utilizando dados sintéticos gerados por interpolação de resultados experimentais.
- Validar a precisão e a eficácia dessa metodologia proposta comparando os resultados obtidos com dados experimentais.

### 1.3

#### Organização da tese

Este trabalho está organizado em sete capítulos. Este primeiro apresentou a Introdução, delineando os objetivos deste estudo, além de fornecer uma revisão bibliográfica abrangente, os objetivos e a pertinência. O Capítulo 2 explora os conceitos fundamentais relacionados à propagação de ondas e ao aprendizado de máquina. No Capítulo 3 é apresentado o método proposto para estimativa de tensão em placas. O Capítulo 4 contém uma análise de curvas de dispersão em guias de ondas cilíndricos utilizando um método semi-analítico para cálculo. O Capítulo 5 aborda a predição da espessura de dutos por meio de uma rede neural. No Capítulo 6 é abordado o tema de *transfer learning*. Por fim, o Capítulo 7 resume as principais conclusões e sugere direções para trabalhos futuros.

## 2

### Conceitos básicos

Este capítulo apresenta uma smula de conceitos relevantes para propagao de ondas elsticas guiadas e *machine learning*. Para o entendimento mais aprofundado de alguns tpicos relacionados, recomendada-se que os artigos, livros e documentos reunidos nas referncias bibliogrficas sejam consultados.

Antes de abordar a propagao de ondas elsticas,  necessrio contextualizar os estudos na mecnica aplicada. Como o assunto  relacionado com ondulatria, acstica e elastodinmica,  recomendvel ter alguma noo da matemtica envolvida com o estudo de campos (identidades do clculo vetorial), assim como  importante visitar conceitos bsicos que podem ser encontrados em livros clssicos de Fundamentos da Fsica [73], Mecnica do Contnuo e Teoria da Elasticidade [74–76].

#### 2.1

##### Propagao de ondas elsticas em slidos

Na mecnica do contnuo, o espao de referncia e o espao atual esto relacionados por meio de uma funo de movimento  $\chi(X, t)$  e tensores derivados desta, como mostra a Figura 2.1.

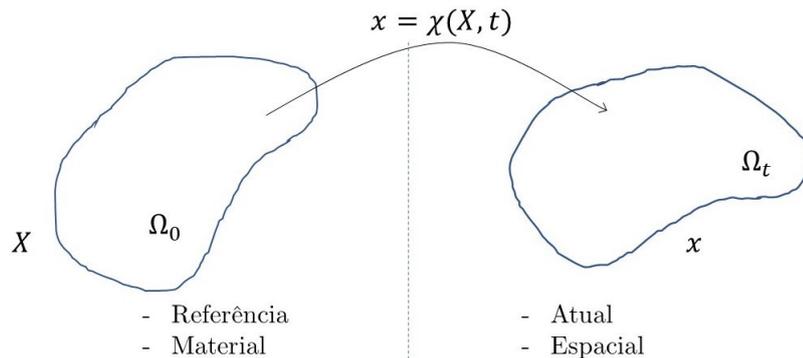


Figura 2.1: Mudana de configurao de um corpo arbitrrio.

##### 2.1.1

##### Equao de Navier-Cauchy

De acordo com a teoria da elasticidade, a soluo de problemas de elastodinmica leva em considerao o equilbrio, a cinemtica do sistema e as

relações constitutivas do material. Dessa forma, há algumas relações e tensores que são importantes de mencionar.

### Gradiente de Deformações

$$\mathbf{F} = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix} = \nabla x = \text{Grad}(x) \quad (2-1)$$

### Tensor de Cauchy-Green à Direita

$$\mathbf{C} = \mathbf{F}^T \mathbf{F} \quad (2-2)$$

### Tensor de deformações de Green-Lagrange

$$\boldsymbol{\varepsilon} = \frac{1}{2} (\mathbf{C} - \mathbb{I}), \quad (2-3)$$

onde  $\mathbb{I}$  é o tensor identidade.

A equação constitutiva que relaciona as grandezas térmicas ou mecânicas de um sistema físico depende da organização molecular do material que compõe esse sistema. Considerando que as variáveis mecânicas de interesse são tensão e deformação, a relação entre essas grandezas é dada pela lei de Hooke (2-4).

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}, \quad (2-4)$$

na qual o tensor constitutivo  $\mathbf{D}$  é dado por (2-5), considerando um material isotrópico em regime linear elástico no estado plano de tensões.

$$\mathbf{D} = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix} \quad (2-5)$$

onde  $E$  é o módulo de Young e  $\nu$  é o coeficiente de Poisson.

É possível também usar os parâmetros de Lamé para representar a relação constitutiva do material elástico linear, conforme:

$$\boldsymbol{\sigma} = \lambda \text{tr}(\boldsymbol{\varepsilon})\mathbb{I} + 2\mu\boldsymbol{\varepsilon}, \quad (2-6)$$

onde  $\lambda$  é o primeiro parâmetro de Lamé e  $\mu$  é segundo parâmetro de Lamé, dados por (2-8). O tensor de deformações de Green-Lagrange é  $\boldsymbol{\varepsilon}$ , enquanto

$\text{tr}(\boldsymbol{\varepsilon})\mathbb{I}$  é a dilatação cúbica. Usando a notação indicial a equação pode ser reescrita conforme abaixo:

$$\sigma_{ij} = \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij}, \quad (2-7)$$

$$\lambda = \frac{\nu E}{(1 - 2\nu)(1 + \nu)} \quad (2-8)$$

$$\mu = \frac{E}{2(1 + \nu)},$$

Antes de continuarmos, a rigor, vale ressaltar que a relação constitutiva com os parâmetros de Lamé utiliza o segundo tensor de Piola, como mostra a (2-9). No entanto, para o regime linear elástico, a introdução de hipóteses simplificadoras faz com que o tensor de Cauchy, o primeiro e o segundo tensor de Piola-Cauchy sejam considerados iguais  $\boldsymbol{\sigma} = \mathbf{P} = \mathbf{S}$ . A seguir, a sequência de equações mostra isso de uma maneira simplificada.

$$\mathbf{S} = \lambda \text{tr}(\boldsymbol{\varepsilon})\mathbb{I} + 2\mu \boldsymbol{\varepsilon}, \quad (2-9)$$

O valor de  $x$  é dado pela função  $\chi(x, t)$ :

$$\mathbf{x} = \chi(\mathbf{X}, \mathbf{t}) = \mathbf{X} + \mathbf{u}, \quad (2-10)$$

onde  $\mathbf{u}$  é o campo de deslocamentos.

$$\mathbf{F} = \nabla \mathbf{x} = \nabla \mathbf{X} + \nabla \mathbf{u} \quad (2-11)$$

$$\nabla \mathbf{X} = \frac{\partial X_i}{\partial X_j} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbb{I} \quad (2-12)$$

$$(2-13)$$

Assim,  $\mathbf{F} = \mathbb{I} + \nabla \mathbf{u}$ .

Como a variação do deslocamento é muito pequena, com  $\nabla \mathbf{u} \approx 0$ , resulta que  $\mathbf{F} \approx \mathbb{I}$ .

$$\mathbf{P} = \mathbf{F}\mathbf{S} = \mathbb{I}\mathbf{S} = \mathbf{S} \quad (2-14)$$

O gradiente de deformações  $\mathbf{F}$ , pode ser dado como o tensor identidade, isso possui outras implicações:

$$\mathbf{F} = \mathbb{I} + \nabla \mathbf{u} \quad (2-15)$$

$$\mathbf{C} = \mathbf{F}^T \mathbf{F} = (\mathbb{I} \times \nabla \mathbf{u})^T + \mathbb{I} + \nabla \mathbf{u} \quad (2-16)$$

$$\mathbf{C} = \mathbb{I} + \nabla \mathbf{u} + \nabla \mathbf{u}^T + \nabla \mathbf{u}^T \nabla \mathbf{u} \quad (2-17)$$

$$(2-18)$$

sendo o valor da parcela  $\nabla \mathbf{u}^T \nabla \mathbf{u}$  desprezado por se tratar de um termo de ordem superior (não linear, muito pequeno), então linearizando:

$$\mathbf{C} = \mathbb{I} + \nabla \mathbf{u} + \nabla \mathbf{u}^T \quad (2-19)$$

Retomando a Equação (2-3), temos:

$$\boldsymbol{\varepsilon} = \frac{1}{2} (\mathbf{C} - \mathbb{I}) = \frac{1}{2} (\mathbb{I} + \nabla \mathbf{u} + \nabla \mathbf{u}^T - \mathbb{I}) \quad (2-20)$$

Este resultado nos leva a cinemática do sistema, ou seja, a equação de compatibilidade (2-21). Em um regime elástico linear, a deformação é fisicamente possível quando é compatível com um campo de deslocamentos.

$$\boldsymbol{\varepsilon} = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) \quad (2-21)$$

Ou, em notação indicial:

$$\varepsilon_{ij} = \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \quad (2-22)$$

O vetor de deslocamentos  $\mathbf{u}$  é dado por:  $\mathbf{u} = [u_1, u_2, u_3]^T$  e seu gradiente é:

$$\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_1}{\partial x_2} & \frac{\partial u_1}{\partial x_3} \\ \frac{\partial u_2}{\partial x_1} & \frac{\partial u_2}{\partial x_2} & \frac{\partial u_2}{\partial x_3} \\ \frac{\partial u_3}{\partial x_1} & \frac{\partial u_3}{\partial x_2} & \frac{\partial u_3}{\partial x_3} \end{bmatrix} \quad (2-23)$$

Substituindo na Equação (2-21), temos:

$$\boldsymbol{\varepsilon} = \frac{1}{2} \left[ \begin{array}{c} \left( \begin{array}{ccc} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_1}{\partial x_2} & \frac{\partial u_1}{\partial x_3} \\ \frac{\partial u_2}{\partial x_1} & \frac{\partial u_2}{\partial x_2} & \frac{\partial u_2}{\partial x_3} \\ \frac{\partial u_3}{\partial x_1} & \frac{\partial u_3}{\partial x_2} & \frac{\partial u_3}{\partial x_3} \end{array} \right) + \left( \begin{array}{ccc} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_2}{\partial x_1} & \frac{\partial u_3}{\partial x_1} \\ \frac{\partial u_1}{\partial x_2} & \frac{\partial u_2}{\partial x_2} & \frac{\partial u_3}{\partial x_2} \\ \frac{\partial u_1}{\partial x_3} & \frac{\partial u_3}{\partial x_3} & \frac{\partial u_3}{\partial x_3} \end{array} \right) \end{array} \right] \quad (2-24)$$

$$\boldsymbol{\varepsilon} = \left[ \begin{array}{ccc} \frac{\partial u_1}{\partial x_1} & \frac{1}{2} \left( \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) & \frac{1}{2} \left( \frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1} \right) \\ & \frac{\partial u_2}{\partial x_2} & \frac{1}{2} \left( \frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \right) \\ \text{Symm} & & \frac{\partial u_3}{\partial x_3} \end{array} \right] \quad (2-25)$$

Retomando a relação constitutiva com os parâmetros de Lamé da Equação (2-7) e substituindo os valores de (2-25), temos os valores das componentes do tensor de tensões  $\boldsymbol{\sigma}$ .

Abaixo, seguem quatro componentes para exemplificar:

$$\sigma_{xx} = \lambda \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} \right) + 2\mu \frac{\partial u_1}{\partial x_1} \quad (2-26)$$

$$\sigma_{xy} = \mu \left( \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \quad (2-27)$$

$$\sigma_{xz} = \mu \left( \frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1} \right) \quad (2-28)$$

$$\sigma_{yy} = \lambda \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} \right) + 2\mu \frac{\partial u_2}{\partial x_2} \quad (2-29)$$

Uma vez que o tensor de tensões é encontrado, ele pode ser utilizado na equação de equilíbrio que envolve as leis de conservação de massa, da conservação de movimento linear e angular, bem como conservação de energia. A segunda lei de Newton define o equilíbrio como:

$$\nabla \cdot \boldsymbol{\sigma}^T + \mathbf{f}_b = \rho \frac{d^2 \mathbf{u}}{dt^2}, \quad (2-30)$$

Considerando que as forças de corpo,  $\mathbf{f}_b$ , são nulas, temos:

$$\nabla \cdot (\lambda \operatorname{tr}(\boldsymbol{\varepsilon})\mathbb{I} + 2\mu\boldsymbol{\varepsilon}) = \rho \frac{d^2 \mathbf{u}}{dt^2}, \quad (2-31)$$

$$2\mu \nabla \cdot \boldsymbol{\varepsilon} + \lambda \nabla \cdot \text{tr}(\boldsymbol{\varepsilon}) \mathbb{I} = \rho \frac{d^2 \mathbf{u}}{dt^2}, \quad (2-32)$$

$$(\nabla \cdot \boldsymbol{\varepsilon})_x = \frac{\partial \varepsilon_{ij}}{\partial x_j} = \frac{\partial \varepsilon_{11}}{\partial x_1} + \frac{\partial \varepsilon_{12}}{\partial x_2} + \frac{\partial \varepsilon_{13}}{\partial x_3} \quad (2-33)$$

$$(\nabla \cdot \boldsymbol{\varepsilon})_x = \frac{\partial^2 u_1}{\partial x_1^2} + \frac{1}{2} \left( \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_1 \partial x_2} \right) + \frac{1}{2} \left( \frac{\partial^2 u_1}{\partial x_3^2} + \frac{\partial^2 u_3}{\partial x_1 \partial x_3} \right) \quad (2-34)$$

$$\lambda \text{tr}(\boldsymbol{\varepsilon}) \mathbb{I} = \lambda \begin{bmatrix} \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} & 0 & 0 \\ 0 & \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} & 0 \\ 0 & 0 & \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} \end{bmatrix} \quad (2-35)$$

$$(\nabla \cdot \lambda \text{tr}(\boldsymbol{\varepsilon}) \mathbb{I})_x = \frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial u_2}{\partial x_2 \partial x_1} + \frac{\partial u_3}{\partial x_3 \partial x_1} \quad (2-36)$$

Montando a equação em termos de  $x$ , temos:

$$2\mu \left( \frac{\partial^2 u_1}{\partial x_1^2} + \frac{1}{2} \frac{\partial^2 u_1}{\partial x_2^2} + \frac{1}{2} \frac{\partial^2 u_2}{\partial x_1 \partial x_2} + \frac{1}{2} \frac{\partial^2 u_1}{\partial x_3^2} + \frac{1}{2} \frac{\partial^2 u_3}{\partial x_1 \partial x_3} \right) + \lambda \left( \frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial u_2}{\partial x_2 \partial x_1} + \frac{\partial u_3}{\partial x_3 \partial x_1} \right) = \rho \ddot{\mathbf{u}} \quad (2-37)$$

Rearrmando, obtemos:

$$(\mu + \lambda) \left( \frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial u_2}{\partial x_2 \partial x_1} + \frac{\partial u_3}{\partial x_3 \partial x_1} \right) + \mu \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_1}{\partial x_2} + \frac{\partial u_1}{\partial x_3} \right) = \rho \ddot{\mathbf{u}} \quad (2-38)$$

Utilizando os conceitos de gradiente e divergente, temos a Equação de Navier-Cauchy, representada na forma vetorial em (2-39), que engloba as leis constitutivas, de equilíbrio e compatibilidade.

$$(\lambda + \mu) \nabla (\nabla \cdot \mathbf{u}) + \mu \nabla^2 \mathbf{u} = \rho \ddot{\mathbf{u}} \quad (2-39)$$

É importante observar que a equação (2-39) está acoplada e por isso possui uma estrutura mais complexa que a equação da onda a qual estamos acostumados, por isso recorre-se a uma forma de decompô-la, usando o teorema de Hemholtz.

### 2.1.2

#### Teorema de Helmholtz

A equação de Navier-Cauchy é utilizada para solução de problemas envolvendo ondas mecânicas elásticas que se propagam em placas, porém, devido a complexidade matemática para a sua resolução, recorre-se ao teorema de Helmholtz em que “todo campo vetorial finito, uniforme, contínuo e que decai no infinito pode ser escrito como a soma do gradiente de um escalar e o rotacional de um campo de divergente nulo” [36].

$$\mathbf{u} = \nabla\phi + \nabla \times \boldsymbol{\psi}. \quad (2-40)$$

onde  $\nabla\phi$  corresponde a parcela do gradiente do potencial escalar e  $\nabla \times \boldsymbol{\psi}$  é o rotacional do potencial vetorial e são dados por (2-41) e (2-42):

$$\nabla\phi = \begin{bmatrix} \frac{\partial\phi}{\partial x_1} \\ \frac{\partial\phi}{\partial x_2} \\ \frac{\partial\phi}{\partial x_3} \end{bmatrix} \quad (2-41)$$

$$\nabla \times \boldsymbol{\psi} = Det \begin{bmatrix} e_1 & e_2 & e_3 \\ \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_3} \\ \psi_{x_1} & \psi_{x_2} & \psi_{x_3} \end{bmatrix} \quad (2-42)$$

$$\begin{aligned} \nabla \times \boldsymbol{\psi} &= \frac{\partial\psi_3}{\partial x_2}e_1 + \frac{\partial\psi_1}{\partial x_3}e_2 + \frac{\partial\psi_2}{\partial x_1}e_3 \\ &\quad - \frac{\partial\psi_1}{\partial x_2}e_3 - \frac{\partial\psi_3}{\partial x_1}e_2 - \frac{\partial\psi_2}{\partial x_3}e_1 \end{aligned}$$

Com isso, a Equação de Helmholtz (2-40) pode ser reescrita como:

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} \phi_{,1} \\ \phi_{,2} \\ \phi_{,3} \end{pmatrix} + \begin{pmatrix} \psi_{3,2} - \psi_{2,3} \\ \psi_{1,3} - \psi_{3,1} \\ \psi_{2,1} - \psi_{1,2} \end{pmatrix} \quad (2-43)$$

### 2.1.3 Equações de Onda

Substituindo o valor do campo de deslocamentos  $\mathbf{u}$  de (2-40) em (2-39), chega-se nas duas equações de onda para o potencial irrotacional e o potencial solenoidal, conforme equações (2-44) e (2-45)

$$\nabla^2 \phi - \frac{1}{c_L^2} \ddot{\phi} = 0, \quad (2-44)$$

$$\nabla^2 \psi - \frac{1}{c_T^2} \ddot{\psi} = 0. \quad (2-45)$$

Os valores das velocidades de onda  $c_L^2$  (longitudinal)  $c_T^2$  (transversal) são dados em (2-46)

$$c_L = \sqrt{\frac{\lambda + 2\mu}{\rho}} \quad ; \quad c_T = \sqrt{\frac{\mu}{\rho}} \quad (2-46)$$

Podemos assumir que a solução para as equações diferenciais dadas em (2-44) e (2-45), possuem forma harmônica, assim:

$$\phi = \mathbf{D}_1(z) e^{j(\omega t - \kappa x)}, \quad (2-47)$$

$$\psi = \mathbf{D}_2(z) e^{j(\omega t - \kappa x)}, \quad (2-48)$$

onde  $\mathbf{D}_1$  e  $\mathbf{D}_2$  são as amplitudes do deslocamento da onda levando em consideração a equação do potencial e descrevem como o campo varia naquela coordenada espacial,  $z$  é a coordenada no espaço,  $j$  é o número complexo,  $\omega$  é a frequência,  $\kappa$  é o número de onda (ou frequência espacial) e  $x$  é a direção onde a onda se propaga.

Para resolver o sistema de equações, além de usar as formas das soluções, é necessário se valer das condições de contorno e isso vai depender do problema em questão, levando em consideração as características da onda e do meio onde ela se propaga.

As equações apresentadas até aqui estão escritas em termos de coordenadas retangulares cartesianas. Para outros tipos de coordenadas devem ser realizadas transformações.

De uma maneira geral e resumida, as equações da onda são usadas para calcular os valores dos potenciais (assumindo formas harmônicas de solução) e esses valores são substituídos na equação dos deslocamentos. Com os deslocamentos é possível calcular as tensões e derivar as velocidades. A partir daí, podem ser encontradas as impedâncias, os fluxos de potência, as curvas de dispersão, os modos, etc.

## 2.2

### Sistemas de Coordenadas

Quando o guia de ondas possui uma seção transversal circular, o sistema pode ser descrito com mais facilidade em coordenadas cilíndricas. Existem 3 famílias de modos nesse tipo de estrutura, modos axiais, flexurais e torcionais e a solução é dada de acordo com essas diferentes famílias.

Para reescrever as equações da elasticidade em coordenadas cilíndricas são consideradas as hipóteses de que a estrutura é axissimétrica, possui comprimento infinito e está imerso no vácuo, além das demais considerações de material linear elástico, homogêneo e isotrópico, características das leis de Hooke. Dessa forma, não há vazamento de energia, nem atenuação devido o material.

#### 2.2.1

##### Coordenadas Cilíndricas

Para coordenadas cilíndricas, o potencial vetorial da Equação (2-40) pode ser decomposto em dois potenciais escalares  $\eta$  e  $\zeta$  [77, 78], o que resulta na Equação (2-49) abaixo:

$$\boldsymbol{\psi} = \eta \mathbf{e}_3 + \nabla \times (\zeta \mathbf{e}_3), \quad (2-49)$$

onde

$$\nabla^2 \eta - \frac{1}{c_T^2} \ddot{\eta} = 0 \quad \text{e} \quad \nabla^2 \zeta - \frac{1}{c_T^2} \ddot{\zeta} = 0, \quad (2-50)$$

portanto:

$$\mathbf{u} = \nabla \phi + \nabla \times (\eta \mathbf{e}_3) + \nabla \times \nabla \times (\zeta \mathbf{e}_3) \quad (2-51)$$

Em coordenadas cilíndricas, o campo de deslocamentos é dado por:

$$\begin{aligned} u_r &= \frac{\partial^2 \phi}{\partial r^2} + \frac{1}{r} \frac{\partial \eta}{\partial \theta} + \frac{\partial^2 \zeta}{\partial z \partial \theta} \\ u_\theta &= \frac{1}{r} \frac{\partial^2 \phi}{\partial r^2} - \frac{\partial \eta}{\partial r} + \frac{1}{r} \frac{\partial^2 \zeta}{\partial z \partial \theta} \\ u_z &= \frac{\partial^2 \phi}{\partial z^2} + \frac{\partial^2 \zeta}{\partial z^2} - \frac{1}{c_T^2} \frac{\partial^2 \zeta}{\partial t^2} \end{aligned} \quad (2-52)$$

Pelas equações acima é possível separar o operador linear  $\mathbb{A}$ , que relaciona os deslocamentos com os potenciais,  $\mathbf{u} = \mathbb{A}\boldsymbol{\Phi}$ . Sendo,

$$\mathbb{A} = \begin{bmatrix} \frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial}{\partial \theta} & \frac{\partial}{\partial z \partial r} \\ \frac{1}{r} \frac{\partial}{\partial \theta} & -\frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial^2}{\partial z \partial \theta} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial z} - \frac{1}{c_T^2} \frac{\partial^2}{\partial t^2} \end{bmatrix} \quad (2-53)$$

e o vetor de potenciais:

$$\Phi = \begin{bmatrix} \phi \\ \eta \\ \zeta \end{bmatrix} \quad (2-54)$$

Após o cálculo dos deslocamentos e com o valor das constantes de Lamé, usando a Equação 2-7, pode-se calcular o valor das tensões, conforme mostra a (2-55).

$$\begin{Bmatrix} \sigma_{rr} \\ \sigma_{r\theta} \\ \sigma_{rz} \end{Bmatrix} = \begin{bmatrix} (\lambda + 2\mu) \frac{\partial}{\partial r} + \lambda \frac{1}{r} & \lambda \frac{1}{r} \frac{\partial}{\partial \theta} & \lambda \frac{\partial}{\partial z} \\ \mu \frac{1}{r} \frac{\partial}{\partial \theta} & \mu \left( \frac{\partial}{\partial r} - \frac{1}{r} \right) & 0 \\ \mu \frac{\partial}{\partial z} & 0 & \mu \frac{\partial}{\partial r} \end{bmatrix} \begin{bmatrix} u_r \\ u_\theta \\ u_z \end{bmatrix} \quad (2-55)$$

A tensão ortogonal na direção radial,  $\mathbf{t}_r$  é dada por (2-56). Sabendo que o vetor  $\mathbf{u}$  é igual a  $\mathbb{A}\Phi$ , podemos introduzir a matriz  $\mathcal{L}$  para obtenção da tensão axial, dada na Equação (2-56).

$$\mathbf{t}_r = \begin{Bmatrix} \sigma_{rr} \\ \sigma_{r\theta} \\ \sigma_{rz} \end{Bmatrix} = \mathcal{L} \begin{bmatrix} \phi \\ \eta \\ \zeta \end{bmatrix}, \quad (2-56)$$

com

$$\mathcal{L} = \begin{bmatrix} (\lambda + 2\mu) \frac{\partial}{\partial r} + \lambda \frac{1}{r} & \lambda \frac{1}{r} \frac{\partial}{\partial \theta} & \lambda \frac{\partial}{\partial z} \\ \mu \frac{1}{r} \frac{\partial}{\partial \theta} & \mu \left( \frac{\partial}{\partial r} - \frac{1}{r} \right) & 0 \\ \mu \frac{\partial}{\partial z} & 0 & \mu \frac{\partial}{\partial r} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial}{\partial \theta} & \frac{\partial}{\partial z \partial r} \\ \frac{1}{r} \frac{\partial}{\partial \theta} & -\frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial^2}{\partial z \partial \theta} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial z} - \frac{1}{c_T^2} \frac{\partial^2}{\partial t^2} \end{bmatrix}. \quad (2-57)$$

Um procedimento análogo pode ser empregado para obter a matriz  $\mathcal{S}$  que relaciona o vetor de tensões axiais com o vetor de potenciais (não confundir a matriz  $\mathcal{S}$  com o segundo tensor de Piola-Cauchy  $\mathbf{S}$ ).

$$\begin{Bmatrix} \sigma_{zr} \\ \sigma_{\theta z} \\ \sigma_{zz} \end{Bmatrix} = \begin{bmatrix} \mu \frac{\partial}{\partial z} & 0 & \mu \frac{\partial}{\partial r} \\ 0 & \mu \frac{\partial}{\partial z} & \mu \frac{1}{r} \frac{\partial}{\partial \theta} \\ \lambda \left( \frac{\partial}{\partial r} + \frac{1}{r} \right) & \lambda \frac{1}{r} \frac{\partial}{\partial \theta} & (\lambda + 2\mu) \frac{\partial}{\partial z} \end{bmatrix} \begin{bmatrix} u_r \\ u_\theta \\ u_z \end{bmatrix} \quad (2-58)$$

$$\mathbf{t}_a = \begin{Bmatrix} \sigma_{zr} \\ \sigma_{\theta z} \\ \sigma_{zz} \end{Bmatrix} = \mathcal{S} \Phi, \quad (2-59)$$

com

$$\mathcal{S} = \begin{bmatrix} \mu \frac{\partial}{\partial z} & 0 & \mu \frac{\partial}{\partial r} \\ 0 & \mu \frac{\partial}{\partial z} & \mu \frac{1}{r} \frac{\partial}{\partial \theta} \\ \lambda \left( \frac{\partial}{\partial r} + \frac{1}{r} \right) & \lambda \frac{1}{r} \frac{\partial}{\partial \theta} & (\lambda + 2\mu) \frac{\partial}{\partial z} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial}{\partial \theta} & \frac{\partial}{\partial z \partial r} \\ \frac{1}{r} \frac{\partial}{\partial \theta} & -\frac{\partial}{\partial r} & \frac{1}{r} \frac{\partial^2}{\partial z \partial \theta} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial z} - \frac{1}{c_T^2} \frac{\partial^2}{\partial t^2} \end{bmatrix} \quad (2-60)$$

## 2.3

### **Machine Learning**

*Machine Learning*, ou traduzindo para o português, Aprendizado de Máquina, é um campo da inteligência artificial que se concentra no desenvolvimento de algoritmos e modelos que capacitam os sistemas a aprender e melho-

rar automaticamente a partir da experiência [79]. Em vez de serem explicitamente programados para realizar uma tarefa específica, os sistemas de *machine learning* utilizam dados para aprender padrões e realizar previsões ou tomar decisões sem intervenção humana direta [80].

Existem várias abordagens dentro do campo de ML, incluindo aprendizado supervisionado, onde o modelo é treinado em um conjunto rotulado de dados; aprendizado não supervisionado, onde o modelo identifica padrões sem dados rotulados; e aprendizado por reforço, onde o modelo aprende através da interação com um ambiente e feedback positivo ou negativo [64]. O *machine learning* tem aplicações abrangentes, desde reconhecimento de padrões e processamento de linguagem natural até análise de dados e tomada de decisões em tempo real, desempenhando um papel essencial em diversas áreas da ciência e da indústria.

Esta seção apresenta os conceitos básicos relacionados aos modelos utilizados neste trabalho. A abordagem empregada é de Aprendizagem Supervisionada, ou seja, é empregado um conjunto de dados que são rotulados, pois possuem pares de entrada e de saída correspondentes [81]. Para mapear os *inputs* com seus respectivos *outputs* são utilizados diferentes modelos de regressão, uma vez que a previsão refere-se a valores contínuos.

### 2.3.1

#### **Aprendizado Supervisionado: Uma Abordagem Fundamentada em Dados**

A aprendizagem supervisionada é um paradigma em *machine learning*, onde um modelo é treinado para mapear entradas e saídas com base em um conjunto rotulado de dados de treinamento. Esse processo envolve alimentar o modelo com um conjunto de exemplos, onde cada exemplo consiste em uma entrada associada a uma saída conhecida. O modelo ajusta seus parâmetros iterativamente a fim de minimizar a diferença entre as saídas previstas e as saídas reais, utilizando uma função de perda ou custo.

Formalmente, consiste em um dado conjunto de treinamento com  $N$  pares de entradas e saídas  $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ , onde  $y$  é determinado por uma função desconhecida  $f(x) = y$ . O objetivo do aprendizado supervisionado é encontrar uma função  $h$  que aproxime a função verdadeira  $f$  [82]. Esse processo é essencial para diversas aplicações, desde reconhecimento de padrões até análise de dados.

No aprendizado supervisionado, o algoritmo recebe um *feedback* explícito durante o treinamento, ajustando seus parâmetros para minimizar a diferença entre suas previsões e os rótulos reais dos dados de treinamento [79]. Isso é feito através de uma função de perda, que quantifica a discrepância entre a

saída prevista e a saída real.

A aprendizagem supervisionada é altamente dependente da qualidade e representatividade do conjunto de dados de treinamento [80]. Um conjunto diversificado e representativo é essencial para permitir que o modelo generalize bem para novas instâncias.

No contexto de avaliação estrutural, a aprendizagem supervisionada pode ser aplicada, por exemplo, para prever tensões em materiais com base em dados de ondas ultrassônicas guiadas. A utilização de redes neurais, máquinas de vetor de suporte (SVM) e outros algoritmos populares são comuns nesse domínio. A seguir são brevemente apresentadas algumas técnicas de regressão em aprendizado supervisionado, que serão utilizadas nos capítulos seguintes.

### 2.3.1.1

#### Ridge regression

Os modelos lineares tendem a ser os mais simples e fáceis de interpretar e implementar. O modelo Ridge ajuda a evitar *overfitting* adicionando um termo de regularização à função de perda. É útil para conjuntos de dados com muitas variáveis explicativas (ou variáveis preditivas) [83]. O modelo de regressão Ridge, ou LRG é representado por

$$\hat{y} = \theta_0 + X\theta_{i+1}, \quad (2-61)$$

onde  $\theta \in \mathbb{R}^{p+1}$  é o vetor que contém os pesos do modelo obtidos a partir da otimização da função de custo

$$J = \|e\| + \alpha \|\theta\|^2, \quad (2-62)$$

com gradiente descendente, onde  $\alpha$  controla o encolhimento do modelo regularizando a amplitude dos pesos. Mais detalhes sobre esta classe de modelo podem ser encontrados em [70].

### 2.3.1.2

#### Support vector regression

O método *support vector regression* (SVR) possui um procedimento de treinamento eficiente e é considerado o método de aprendizado de máquina mais robusto e preciso [84–86]. Ele usa vetores de suporte para encontrar o hiperplano que melhor separa as classes no espaço de recursos. É eficaz em espaços de alta dimensão, embora exija mais esforço computacional e ajuste de hiperparâmetros [83]. O problema de construção do modelo SVR é definido como:

$$\left\{ \begin{array}{l} \min_{w, \xi, \xi^*} \left[ \frac{1}{2} w^T w + \sum_i \xi_i + \xi_i^* \right] \\ \text{s. t.} \quad \hat{y}_i - y_i \leq \epsilon + \xi \\ \quad \quad y_i - \hat{y}_i \leq \epsilon + \xi^* \\ \quad \quad \xi, \xi^* \geq 0, i = 1, \dots, n, \end{array} \right. \quad (2-63)$$

onde  $\hat{y}$  é a previsão do modelo dada por  $f(x_i, \theta) = w^T \phi(x_i) + b$ . Os parâmetros do modelo são  $w, b \in \theta$  e  $\phi(x_i) = \gamma K(x_i - x'_i)$  é a função kernel que mapeia o vetor de entrada, que normalmente é definido como funções de base radial, polinômios de ordem superior, sigmóides ou funções lineares.  $C$  é o fator de regularização e  $\xi, \xi^*$  garante que sejam penalizadas as amostras que são o valor  $\epsilon$  mais distantes de  $y$ .

### 2.3.1.3

#### Decision tree regression

O modelo DTR (*Decision tree regression*) divide recursivamente os dados em subconjuntos com base nos recursos mais importantes para regressão. Embora tenha a desvantagem de tender ao *overfitting* dos dados, é simples de interpretar [83]. Um modelo DTR é composto de nós e “folhas”, podendo ser representado como [87]

$$\hat{y} = f(x, \theta) = \sum_k c_k 1_{(x \in R_k)}, \quad (2-64)$$

onde  $c_k$  é o  $k$ -ésimo coeficiente alvo de regressão obtido para cada região  $R_k$  não sobreposta, que são retângulos de alta dimensão do espaço de entrada.

### 2.3.1.4

#### K-nearest neighbors regression

Este modelo é um mecanismo de inferência baseado em instâncias, o que significa que não necessita de nenhum processo de treinamento para construção do modelo [43]. Ele funciona encontrando os pontos  $k$  de dados mais próximos da entrada e calculando a média de suas saídas para regressão como sendo a previsão [88]. Mais detalhes sobre a regressão de *k-nearest neighbors regression* (KNN) podem ser vistos em [89].

### 2.3.1.5

#### Random Forest Regression

O modelo *Random Forest Regression* (RFR) consiste em um conjunto de árvores de decisão, onde a saída é a média das previsões de todas as árvores.

Tem a vantagem de reduzir o *overfitting* e melhorar a precisão, no entanto, pode ser lento para treinar [83]

### 2.3.1.6

#### Gradient boosting regressor

É um algoritmo de aprendizagem supervisionada baseado em árvores de decisão. Este método constrói várias árvores sequencialmente, onde cada nova árvore é treinada para corrigir os erros residuais das árvores anteriores, melhorando a precisão geral. O *Gradient boosting regressor* (GBR) é conhecido por fornecer previsões precisas, superando muitos outros algoritmos e não é sensível à escala de atributos, tornando-o mais robusto nesse aspecto. No entanto, se os parâmetros não forem definidos corretamente ou se o número de estimadores for muito alto, o GBR pode tornar-se suscetível ao *overfitting* [90].

### 2.3.1.7

#### Redes Neurais: uma abordagem inspirada no cérebro

As redes neurais, uma classe fundamental de algoritmos de aprendizado de máquina, foram inicialmente concebidas com inspiração na estrutura e funcionamento do cérebro humano. Essa abordagem computacional complexa é composta por camadas de unidades de processamento chamadas neurônios, conectadas por pesos sinápticos, permitindo a aprendizagem de padrões e representações hierárquicas.

A estrutura básica de uma rede neural é representada por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios da camada subsequente, formando uma rede de interconexões. Essas conexões têm pesos associados, ajustados durante o treinamento para otimizar a capacidade da rede de fazer previsões ou classificações.

A função de ativação, que determina a saída de um neurônio dado um conjunto de entradas ponderadas, é um componente crucial nas redes neurais. Funções comuns incluem a função logística

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2-65)$$

e a unidade linear retificada (ReLU)

$$g(x) = \max(0, x) \quad (2-66)$$

Rumelhart et al. (1986) [91] introduziram o algoritmo de retropropagação (*backpropagation*), um método eficiente para treinar redes neurais. Este

algoritmo ajusta iterativamente os pesos da rede para minimizar uma função de erro, propaga o erro da saída para as camadas anteriores e atualiza os pesos de acordo com o gradiente negativo da função de erro.

Para aplicações específicas, é possível personalizar a arquitetura das redes neurais. Redes convolucionais (*Convolutional Neural Networks - CNNs*) são especialmente eficazes em tarefas de visão computacional, enquanto redes recorrentes (*Recurrent Neural Networks - RNNs*) são úteis para lidar com dados sequenciais, como séries temporais.

As redes neurais têm sido aplicadas com sucesso em diversas áreas, desde reconhecimento de padrões até processamento de linguagem natural. No contexto deste trabalho, como pode ser visto no Capítulo 5 as redes neurais desempenham um papel fundamental na avaliação estrutural, fornecendo uma estrutura adaptável para analisar dados complexos.

### 2.3.1.8

#### Multi-layer Perceptron

Como já foi falado anteriormente, Redes Neurais Artificiais são modelos matemáticos computacionais inspirados na arquitetura neural de organismos inteligentes que adquirem conhecimento por meio de aprendizagem experiencial [92]. Dentre estas, destaca-se o MLP (Multi-Layer Perceptron), composto por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. É uma rede *feedforward*, o que significa que nenhuma saída de um neurônio na camada  $k$  serve como entrada para um neurônio em uma camada menor ou igual a  $k$ . Além disso, é totalmente conectada, permitindo que cada neurônio forneça sua saída para cada unidade na camada subsequente [93–95]. Juntamente com os modelos convencionais mencionados anteriormente, no Capítulo 3, a abordagem utilizada incorpora um MLP para aproveitar os benefícios do aprendizado de redes neurais. Vale ressaltar que as Redes MLP apresentam um poder computacional significativamente maior em comparação com Redes Neurais sem camadas ocultas, como Perceptron, Adaline ou Rede Neural de Camada Única, por exemplo [96].

### 3

## Estimativa de tensão de tração em placas

Este capítulo está relacionado ao artigo intitulado *Tensile stress estimation in plates using synthetic ultrasonic guided waves and supervised learning*, submetido ao periódico Ultrasonics, e que foi desenvolvido pela autora desta tese juntamente com outros pesquisadores.

A estimativa precisa da tensão é crucial para análise estrutural e concepção de projetos. A utilização de ondas guiadas ultrassônicas, devido ao efeito acustoelástico, oferece uma boa maneira de estimar a tensão em estruturas. Neste capítulo, é proposto um método de modelagem baseado em dados para estimar a tensão mecânica em placas usando aprendizado de máquina e sinais sintéticos de ondas guiadas gerados através da interpolação de resultados experimentais.

A ideia é usar os dados sintéticos para treinar os modelos de aprendizado de máquina, uma vez que os dados experimentais podem estar indisponíveis ou ter uma aquisição onerosa. Este estudo emprega seis modelos de regressão diferentes, explora combinações de hiperparâmetros e utiliza análise de componentes principais para extração das principais características.

O método aqui exposto demonstra desempenho robusto, mostrando o potencial dos dados sintéticos para aprimorar conjuntos de dados de treinamento de aprendizado de máquina. Esta pesquisa estabelece a viabilidade de modelos de regressão e uso de dados sintéticos para estimativa de tensão, contribuindo com as metodologias existentes.

### 3.1

#### Construção do modelo e validação

A escolha de um método para extração de características depende da natureza do problema, da disponibilidade de dados e das restrições computacionais. Cada abordagem tem suas vantagens e desvantagens, e a seleção adequada é crucial para o desempenho e a eficácia dos modelos de aprendizado de máquina. Um método que se adequa bem para problemas acústicos é a análise de componentes principais (em inglês a sigla é PCA) [43]. O PCA procura identificar a maior variação nos dados e projeta esses pontos em um novo espaço menor. Isto permite capturar as características mais relevantes e

eliminar redundâncias [97]. A principal vantagem do PCA é a sua capacidade de reduzir a dimensionalidade dos dados, encontrando as direções de maior variação e projetando os pontos em novos eixos principais, mantendo a maior parte da variabilidade original. Esses novos eixos, chamados de componentes principais, são ordenados em ordem decrescente de acordo com a quantidade de variação capturada, que é calculada através da Decomposição de Valores Singulares (SVD do inglês *Singular Value Decomposition*) [98]. Desta forma, é possível representar os dados num espaço de dimensão reduzida, mantendo a maior parte da informação relevante [97].

Para validar a arquitetura proposta neste trabalho, é utilizado o método de validação cruzada repetida com  $k$ -partições (*repeated CV k-fold*). A validação cruzada, ou *cross validation* é uma técnica amplamente utilizada para estimar o desempenho de modelos de aprendizado de máquina, dividindo o conjunto de dados em conjuntos de treinamento e teste. No *repeated CV k-fold*, o conjunto de dados é dividido em  $k$  subconjuntos (partições) de tamanho aproximadamente igual. Em cada repetição,  $k-1$  partição são usadas para treinamento do modelo, enquanto a partição restante é usada para teste. Este processo é repetido diversas vezes, alternando as partições utilizadas para treinamento e teste, a fim de garantir uma avaliação robusta e estável do modelo. Uma das principais vantagens do *repeated CV k-fold* é que ele permite uma avaliação mais precisa e confiável do desempenho do modelo, reduzindo a dependência da aleatoriedade na seleção das partições. Além disso, também ajuda a mitigar o enviesamento que pode ser causado por uma única divisão arbitrária de dados em treinamento e teste [99–101].

## 3.2

### Dados experimentais

Os dados experimentais utilizados aqui não foram obtidos pela autora. Eles foram coletados pelos autores de outro estudo [102] e também já foram objeto de análise em outro artigo [43]. Nesta tese, os dados são empregados em duas aplicações diferentes: primeiro para gerar sinais sintéticos de ondas ultrassônicas por meio de interpolação e a seguir para testar os modelos de ML. A configuração experimental usada neste artigo é ilustrada na Fig. 3.1, que foi adaptada de [43].

O aparato experimental é constituído por uma placa de alumínio com 3 mm de espessura, 100 mm de largura e 800 mm de comprimento montada sobre uma estrutura usada para impor tensão de tração na direção longitudinal da placa. Dois transdutores piezoelétricos ultrassônicos foram colados na superfície da placa, próximos às suas extremidades, para gerar e receber ondas

ultrassônicas. A excitação foi realizada por meio de um sinal tipo sinc com largura de banda de 2,5 MHz.

Devido à excitação de banda larga, vários modos de onda guiada podem se propagar. A direção de propagação da onda é paralela à tensão de tração. De acordo com a teoria acustoelástica, cada modo e frequência de onda guiada apresenta uma sensibilidade distinta a tensão [103–105]. Portanto, uma vez que a forma de onda recebida é composta por vários modos de onda e reflexões, devido a tensão fornecida à placa, ela muda de uma forma não trivial. Mais detalhes da configuração experimental podem ser encontrados em [102].

Diversas leituras de sinais foram realizadas para diferentes valores de deformação, abrangendo uma faixa de até cerca de  $150 \mu m/m$ , obtendo uma correspondência entre tensão mecânica, deformação e forma de onda recebida. A aquisição do sinal ocorreu em uma janela de tempo que inicia em  $130 \mu s$  e termina em  $949,1 \mu s$ , totalizando um intervalo de tempo de  $819,1 \mu s$ . Os sinais foram amostrados a uma frequência de 10 MHz, produzindo 8.192 amostras ao longo do tempo para cada um dos 499 casos de medição.

Em outras palavras, foram coletados 499 sinais para diferentes níveis de deformação, abrangendo uma faixa de até cerca de  $150 \mu m/m$ . Cada nível de deformação aplicado resultou em múltiplos sinais ultrassônicos ao longo do tempo recebidos devido à excitação de banda larga utilizada no experimento. Como a resposta ultrassônica da placa varia com o nível de deformação, cada nível de deformação resulta em uma série temporal única de sinais ultrassônicos. Portanto, para diferentes níveis de deformação, há múltiplas séries temporais de sinais ultrassônicos correspondentes.

Além disso, a medição da deformação real foi realizada utilizando-se um *strain gauge* resistivo posicionado no centro da placa de alumínio (representado pela letra C na Fig. 3.1). Este sensor de deformação foi fundamental para correlacionar os sinais recebidos com os diferentes níveis de deformação aplicados. Assim, a combinação do sensor de deformação e dos sinais ultrassônicos coletados resultou na base de dados experimentais, representando o comportamento do material sob diferentes condições de carga.

### 3.3

#### Abordagem proposta

Nesta seção, é apresentada a abordagem empregada neste estudo para análise de tensão utilizando aprendizado de máquina. A estrutura metodológica abrange duas fases principais: a geração de sinais sintéticos de ondas ultrassônicas por meio de interpolação linear e a utilização de vários modelos de aprendizado de máquina para estimativa de tensão. O processo começa

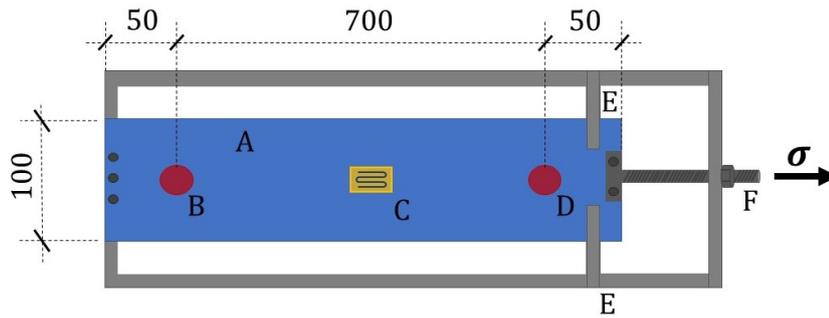


Figura 3.1: Croqui da bancada experimental: A - Placa de alumínio; B - Transdutor emissor piezoelétrico; C - Extensômetro; D - Transdutor receptor piezoelétrico; E - Suporte de montagem e; F - Haste roscada.

com a utilização de dados experimentais relativos à onda guiada ultrassônica dentro de uma placa de alumínio submetida a carregamentos de tensão, conforme descrito na Seção 3.2. Porém, vale ressaltar que os dados experimentais não foram empregados diretamente para treinar os modelos de aprendizado de máquina.

Em vez disso, o resultado experimental e seus rótulos, foram empregados para gerar sinais artificiais de ondas por meio de interpolação linear e seus respectivos rótulos, simulando condições do mundo real. Esses conjuntos de dados sintéticos, compreendendo diversos sinais correspondentes a casos hipotéticos de medição, foram fundamentais no treinamento dos modelos de aprendizado de máquina.

Antes do treinamento do modelo, uma análise dos componentes principais (PCA) foi aplicada aos conjuntos de dados sintéticos para extrair características relevantes e aumentar a eficiência do processo de aprendizagem subsequente. Como são várias medições (15, 29, 58, 116, 232, 346, 462, 924, 1848, 5082 e 10164) com 8192 amostras ao longo do tempo cada, foram criadas matrizes de dimensões "quantidade de sinais" x 8192, onde cada linha representa uma única execução do experimento com uma configuração de tensão diferente.

A extração de características está relacionada à criação de informações úteis e não redundantes que podem ser alimentadas para a construção de modelos preditivos. Para isso, utilizamos o PCA, diretamente nas matrizes criadas com todos os sinais ultrassônicos sintéticos. É importante ressaltar que este processo foi realizado sem o uso de qualquer informação sobre tensão ou deformação, sendo assim, este procedimento é um processo totalmente orientado por dados.

Os dados experimentais foram reservados para testar o desempenho dos modelos, validando efetivamente a sua utilidade prática. A Figura 3.2 representa esquematicamente o método aqui proposto: Os sinais de ondas guiadas

ultrassônicas foram obtidos usando um aparato experimental, cujo croqui esquemático foi ilustrado na Fig. 3.1. Os sinais ultrassônicos recebidos pelos transdutores piezoelétricos após a excitação da placa de alumínio representam as ondas ultrassônicas propagadas na placa e são detectados pelos transdutores na forma de sinais elétricos. Portanto, esses sinais estão na forma de voltagem de saída dos transdutores.

Os sinais recebidos foram utilizados para gerar os dados sintéticos, por meio de interpolação linear. A seguir, com PCA, foram extraídas as principais características desses dados sintéticos. Este conjunto de dados sintético é então dividido aleatoriamente várias vezes em conjuntos de dados de treinamento e teste. O conjunto de dados de treinamento é usado para obter os hiperparâmetros usando busca em grade e validação cruzada. O melhor modelo encontrado é então usado para realizar as previsões. As seções seguintes se aprofundam nas etapas dessa abordagem.

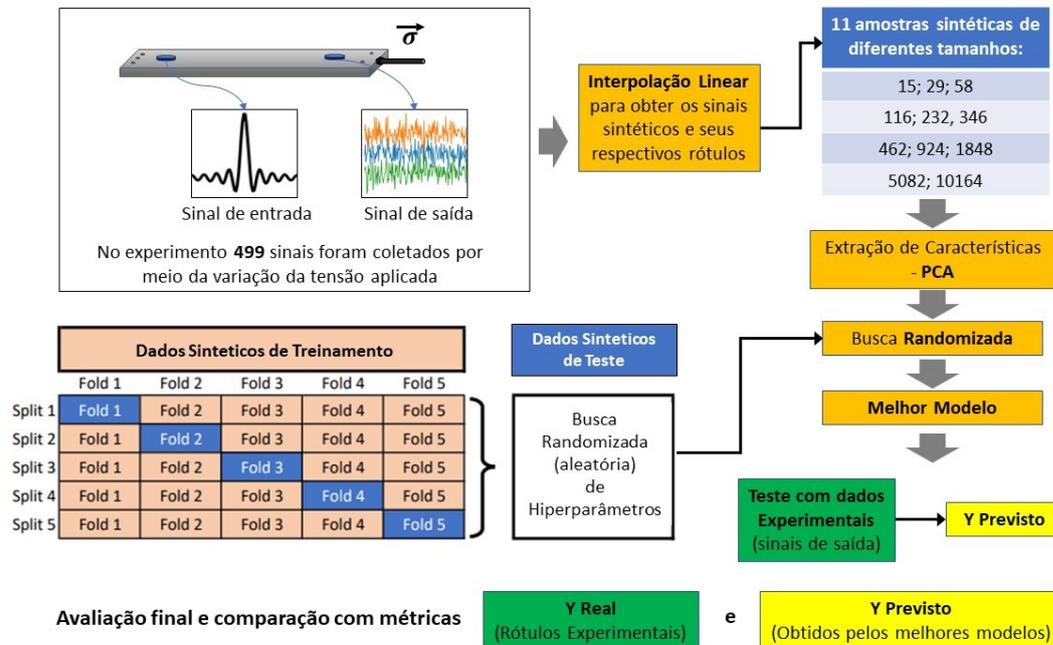


Figura 3.2: Visão geral do método proposto.

### 3.4

#### Sinais de ondas ultrassônicas sintéticas através de interpolação

Como já foi mencionado anteriormente e descrito na na Seção 3.2, os dados experimentais utilizados neste estudo consistem na medição dos sinais de ondas ultrassônicas guiadas em uma placa de alumínio que sofre deformação por tração. O valor das deformações foi usado como rótulo para o problema de estimativa de tensão.

No entanto, é importante enfatizar mais uma vez, que os dados experimentais em si não foram empregados diretamente no treinamento dos modelos. Inicialmente, os dados experimentais foram utilizados para gerar sinais de ondas sintéticas e seus rótulos correspondentes através de interpolação linear. A criação deste conjunto de dados sintéticos permitiu a emulação de condições do mundo real. Posteriormente, os dados experimentais foram utilizados para fins de teste para avaliar o desempenho dos modelos. Todo o sinal foi interpolado, e não a amostragem dentro do sinal, que permaneceu constante em 8.192 amostras de tempo.

Por definição, a interpolação permite a obtenção de um novo conjunto de dados a partir de outro conjunto previamente conhecido que pode ser obtido através de amostragem ou medições experimentais [106, 107], como é o caso deste estudo. Essencialmente, envolve a construção de uma função  $z' = f(x', y')$ , que aproxima os dados conhecidos. Este processo pode ser descrito pela seguinte equação:

$$z' = z_1 + (z_2 - z_1) \cdot \left( \frac{x' - x_1}{x_2 - x_1} \right) + (z_2 - z_1) \cdot \left( \frac{y' - y_1}{y_2 - y_1} \right), \quad (3-1)$$

onde  $z$  representa os sinais de onda;  $x$  é o conjunto de dados rotulado (deformações) e  $y$  corresponde ao conjunto de dados de tempo;  $z_1$  e  $z_2$  representam os valores dos pontos conhecidos  $(x_1, y_1)$  e  $(x_2, y_2)$ ;  $z'$  é o valor interpolado no novo par de pontos  $(x', y')$ , que por sua vez são as coordenadas nas quais o valor interpolado é estimado.

### 3.5 Modelos de aprendizado de máquina

No processo de desenvolvimento de modelos de aprendizado de máquina, encontrar a configuração ideal de hiperparâmetros é essencial, pois isso afeta diretamente o desempenho e a capacidade de generalização [87]. Assim, uma abordagem de busca aleatória de hiperparâmetros foi empregada em conjunto com repetidas validações cruzadas. Este procedimento foi realizado para os modelos LRG Ridge, RFR, KNN, DTR, GBR, SVR e MLP, com o objetivo de identificar as melhores configurações de hiperparâmetros para cada um.

É importante pontuar que os hiperparâmetros são parâmetros do modelo de aprendizado de máquina que não são aprendidos diretamente dos dados, ao contrário dos pesos do modelo [64]. Em vez disso, eles são configurados antes do treinamento e desempenham um papel crucial na determinação do comportamento do modelo e da sua arquitetura [70, 108]. Eles podem incluir a taxa de aprendizagem em algoritmos, o número de camadas e neurônios em redes neurais, ou a profundidade máxima em árvores de decisão, por exemplo.

São críticos porque controlam a capacidade do modelo de ajustar os dados de treinamento e, portanto, afetam diretamente o desempenho e a capacidade de generalização. No contexto da otimização de hiperparâmetros, o objetivo é encontrar os valores ideais para essas configurações.

Neste estudo, empregamos uma abordagem de busca aleatória, uma técnica amplamente utilizada na otimização de hiperparâmetros de aprendizado de máquina [109]. A busca randômica envolve explorar um espaço predefinido selecionando aleatoriamente combinações de hiperparâmetros em cada iteração. Matematicamente, pode ser representado da seguinte forma:

$$\theta^* = \arg \min_{\theta} L(\theta), \quad (3-2)$$

onde  $\theta^*$  representa o conjunto ideal de hiperparâmetros através da minimização da função  $L(\theta)$  que quantifica o desempenho do modelo com uma configuração de hiperparâmetros específica. A busca randômica explora esse espaço com eficiência de forma iterativa com amostragem de diferentes valores de  $\theta$  e avalia seu desempenho usando técnicas como a validação cruzada. Esta formulação é um conceito fundamental na teoria de otimização [110, 111].

Este método oferece diversas vantagens, incluindo uma exploração mais eficiente do espaço de hiperparâmetros em comparação com a busca em grade, que testa todas as combinações possíveis [112, 113]. Ao fazer a amostragem hiperparâmetros de forma aleatória, há uma chance maior de descobrir configurações efetivas sem ficar restrito a uma grade fixa.

Para implementar a busca aleatória, foi aproveitado o módulo `Randomized Search CV` fornecido pela biblioteca `scikit-learn`, que é uma biblioteca versátil de aprendizado de máquina na linguagem Python e este módulo simplifica o processo de realização de buscas aleatórias de hiperparâmetros. Ele permite especificar um intervalo de valores e distribuições de hiperparâmetros, como sendo uniforme ou logarítmico [114]. Além disso, lida com a tarefa repetitiva de ajustar e avaliar modelos com diferentes combinações, agilizando o processo de otimização. Essa escolha foi feita para aproveitar as ferramentas eficientes e amigáveis do `scikit-learn` para ajuste de hiperparâmetros, garantindo uma otimização robusta dos modelos de aprendizado de máquina. Neste processo foram especificadas diversas configurações de hiperparâmetros para cada modelo, como pode ser visto na Tabela 3.1. Por exemplo, para o modelo LRG Ridge, a busca considerou valores diferentes para o hiperparâmetro alfa e algumas opções de solver.

A pesquisa aleatória foi configurada para ser executada em 40 iterações, com cada iteração explorando uma nova combinação de hiperparâmetros. Além disso, para avaliar a robustez do desempenho do modelo, todo o processo foi

repetido 100 vezes de acordo com os recursos computacionais disponíveis e obtendo um número suficientemente grande para tirar conclusões significativas. Este número também permite uma exploração abrangente do espaço de hiperparâmetros. A validação cruzada com 5 partições e 5 repetições garante que o ajuste de hiperparâmetros seja executado diversas vezes com diferentes divisões de dados, proporcionando uma avaliação de desempenho mais precisa.

A importância da repetição é justificada porque a busca por hiperparâmetros ótimos é um processo estocástico, o que significa que pode produzir resultados ligeiramente diferentes cada vez que é executado devido à aleatoriedade [87, 112]. Ao repetir a busca aleatória 100 vezes com validação cruzada de 5 repetições, é possível demonstrar a robustez da otimização de hiperparâmetros. Esta robustez é crucial porque mostra que os hiperparâmetros escolhidos são consistentemente eficazes em diferentes divisões de dados, reduzindo a probabilidade de ajuste excessivo a uma configuração de dados específica e aumentando a confiabilidade do modelo.

O valor 100 também foi utilizado como semente para controlar a aleatoriedade do processo de busca, garantindo que as amostras aleatórias geradas em cada iteração sejam possíveis de reproduzir [115]. Isto é particularmente importante para fins de comparação e reprodução de resultados.

Modelo	Hiperparametro	Opção
LRG	alpha	stats.loguniform(1e-2, 1e2)
	solver	auto, cholesky, lsqr, sag, saga
RFR	n_estimators	stats.randint(2,100)
	criterion	friedman_mse, poisson
	max_depth	stats.randint(5,10)
	max_features	stats.uniform(0, 1)
KNN	n_neighbors	stats.randint(2, 10)
	weights	uniforms, distance
DTR	criterion	absolute_error, friedman_mse
	splitter	best, random
	max_features	np.linspace(0.1, 1.0, 10)
SVR	C	stats.loguniform(1e-1, 1e3)
	kernel	linear, poly, rbf, sigmoid
	degree	stats.randint(2, 5)
	gamma	stats.loguniform(1e-4, 1e0)
	epsilon	stats.loguniform(1e-4, 1e1)
GBR	n_estimators	np.arange(50, 200)
	learning_rate	np.linspace(0.01, 0.1, 10)
	max_depth	np.arange(3, 10)
	loss	quantile, absolute_error
MLP	hidden_layer_sizes	[2,3]
	neurons_per_layers	[16, 32, 64]
	activation	relu, tanh
	max_iter	[10, 20, 50]
	learning_rate	[1e-4, 5e-4, 1e-3, 5e-3, 1e-2]

Tabela 3.1: Configurações de hiperparâmetros para os modelos usando validação cruzada e pesquisa aleatória

### 3.6

#### Resultados

Esta seção apresenta uma avaliação do desempenho do modelo com base nas métricas de Erro Quadrático Médio (MSE), Coeficiente de Determinação (R2) e Erro Médio Absoluto (MAE). Esta análise abrange todos os tamanhos de amostra examinados, fornecendo uma visão de como os modelos se comportam em diferentes escalas de conjuntos de dados. Através destas comparações, é possível ter uma percepção sobre os pontos fortes e fracos de cada modelo, de acordo com as suas habilidades na captura de padrões subjacentes e produção de previsões precisas.

Neste trabalho, o método PCA foi empregado para extração de características do conjunto de dados sintético. Esta técnica reduz efetivamente a dimensionalidade dos dados, ao mesmo tempo que retém as informações mais significativas.

Para determinar o número ótimo de componentes principais a reter, adotou-se o critério de 95% da variância total. Esta abordagem está representada na Figura 3.3, que ilustra graficamente a variação cumulativa em função do número de componentes principais, que se mostraram ser 8 para o conjunto de dados em questão. Através do PCA, os dados de alta dimensão dos conjuntos de dados sintéticos foram transformados em uma representação de menor dimensão, facilitando a análise subsequente e o treinamento do modelo.

Sabe-se que apenas alguns intervalos de tempo são importantes para detectar mudanças no estresse. A análise de componentes principais permite automatizar o processo de seleção de intervalos que seriam necessários se quiséssemos analisar diretamente os sinais ultrassônicos, de forma orientada por dados e automatizada. O uso de tal método é importante no contexto da modelagem preditiva para aplicações de monitoramento, pois permite otimizar o processo de inferência e construir modelos mais eficazes.

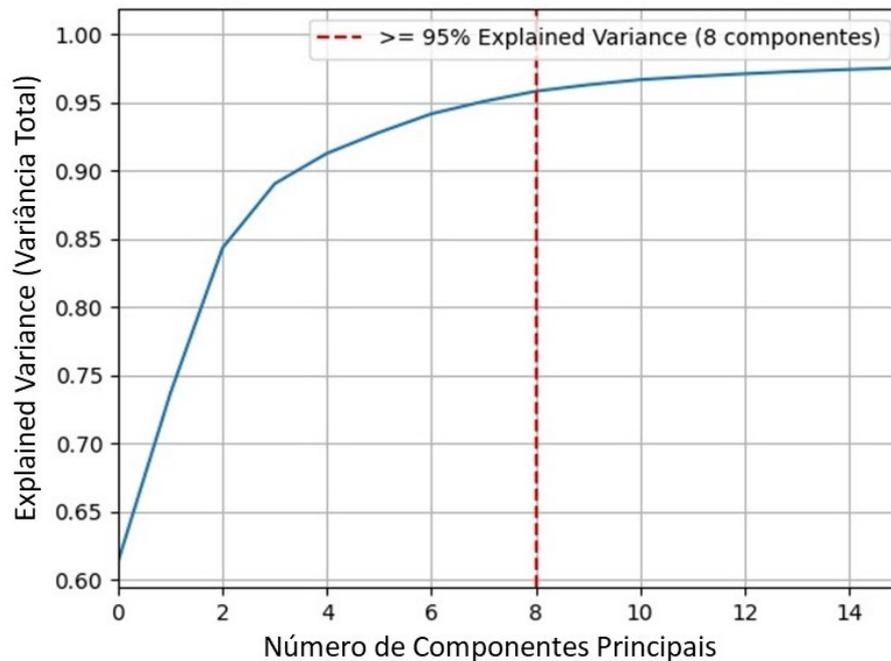


Figura 3.3: Número de componentes principais para os conjuntos de dados sintéticos.

A Figura 3.4 mostra as tendências do Erro Quadrático Médio (MSE) para diferentes modelos à medida que o tamanho dos conjuntos de dados sintéticos aumenta. O desempenho de todos os modelos evolui à medida que mais dados são utilizados. Como pode ser observado, os modelos SVR, DTR e RFR são mais sensíveis, apresentando as maiores mudanças pelo menos até o conjunto de dados sintéticos com 58 amostras de deformação. Aparentemente, isso ocorre para amostras de deformação com uma menor quantidade de dados.

É possível observar como os diferentes modelos se comportam em termos de MSE para diferentes tamanhos de amostras de deformação. Os modelos com o menor MSE para conjuntos de dados maiores são RFR, SVR e KNN, ou seja, são aqueles com o melhor desempenho de previsão. Diferentes padrões de desempenho podem ser observados à medida que o número de amostras no conjuntos de dados sintéticos varia. Os modelos GBR, KNN e LRG parecem apresentar um desempenho mais consistente.

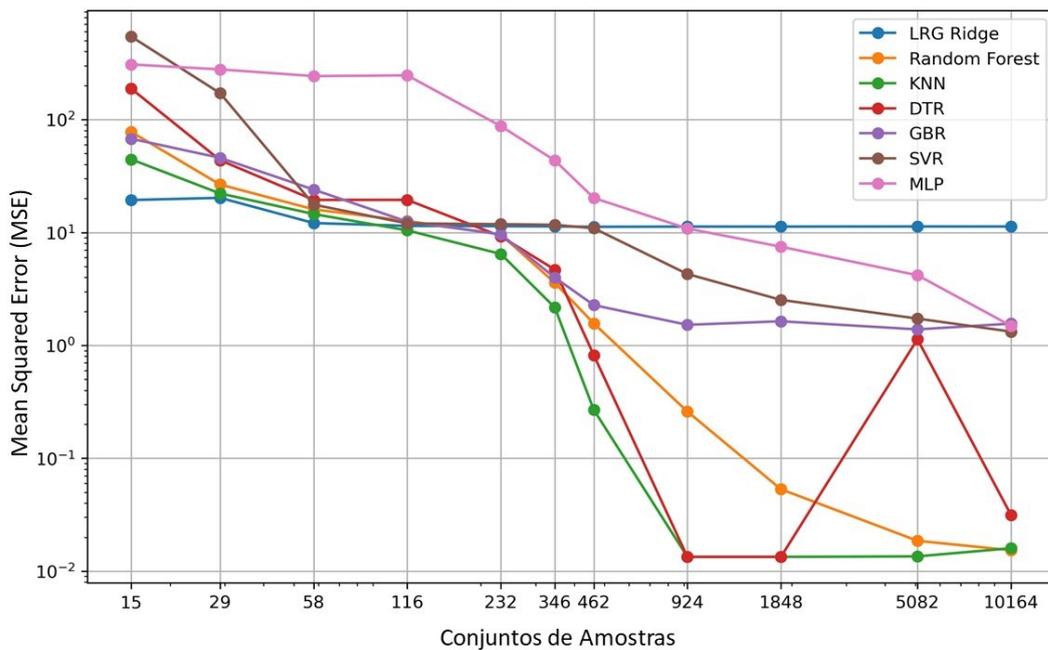


Figura 3.4: Erro Quadrático Médio (MSE) para diferentes modelos à medida que o tamanho dos conjuntos de dados sintéticos aumenta (gráfico em escala logarítmica para melhor visualização).

A Figura 3.5 mostra como o coeficiente de determinação  $R^2$  varia com o aumento dos tamanhos dos conjuntos de dados sintéticos. Em geral, espera-se que o  $R^2$  aumente à medida que o número de dados aumenta, pois os modelos ajustados a conjuntos de dados maiores tendem a capturar melhor os padrões. Comparando as linhas de todos os modelos usados, podemos ver quais modelos têm o melhor desempenho em amostras de deformação com diferentes tamanhos de conjuntos de dados sintéticos.

Os modelos LGR, KNN e GBR têm linhas mais altas e mais próximas de 1.0, mesmo com conjuntos pequenos (15, 29 e 58), o que indica um melhor ajuste aos dados, enquanto modelos com linhas mais baixas podem estar subajustando os dados. A partir da amostra de 462 casos, todos os modelos apresentam bom desempenho. Na Figura 3.6, é possível ver mais claramente que os modelos MLP e SVR realmente mostram uma melhora quando o

tamanho da amostra é superior a 462, enquanto os outros modelos não mostram uma melhoria adicional.

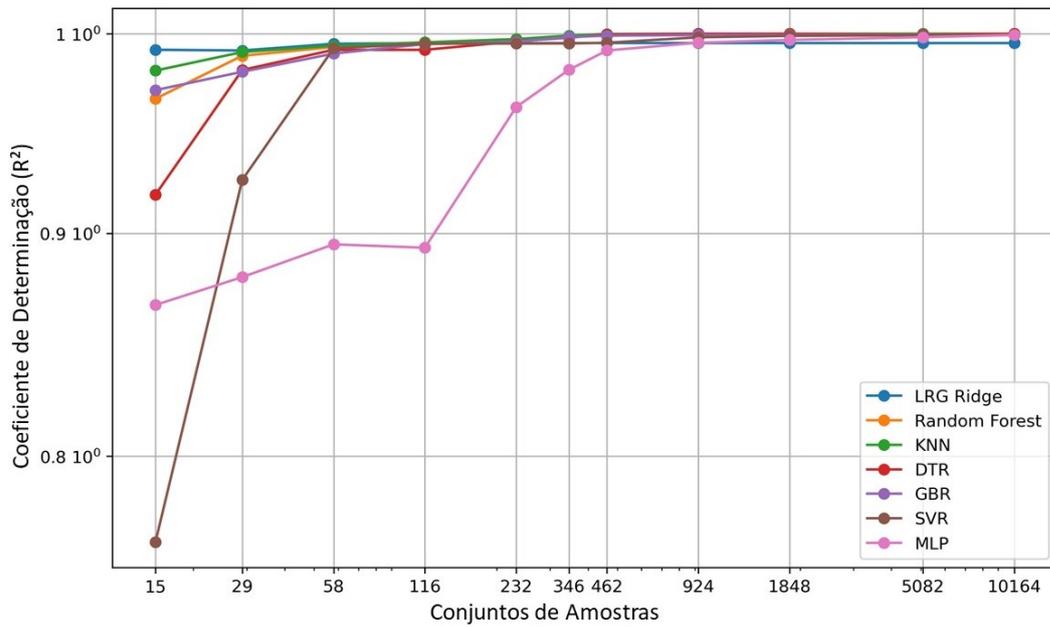


Figura 3.5: Gráfico de comparação do Coeficiente de Determinação ( $R^2$ ) entre diferentes modelos de aprendizado de máquina para os 11 conjuntos de dados sintéticos.

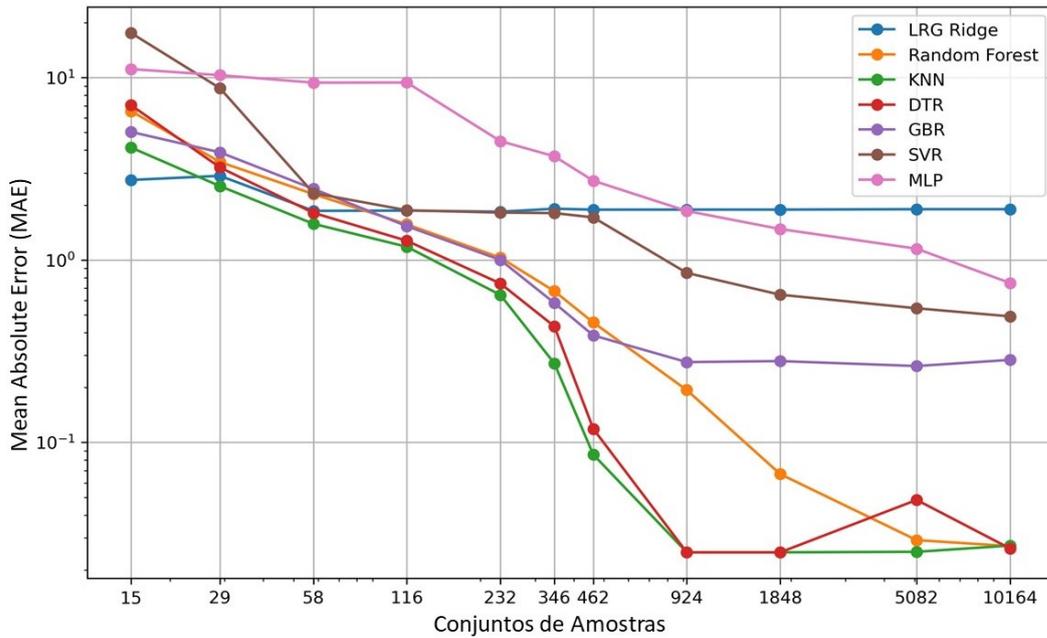


Figura 3.6: Gráfico de comparação do Erro Médio Absoluto (MAE) entre diferentes modelos de aprendizado de máquina em diferentes conjuntos de dados sintéticos.

As Tabelas 3.2 a 3.12 mostram as estatísticas do erro médio quadrático (MSE) para diferentes modelos testados com conjuntos de dados sintéticos de diferentes tamanhos. Elas estão organizadas de forma a classificar os modelos em ordem crescente de MSE médio. Isso auxilia na análise do desempenho do modelo à medida que o tamanho da amostra aumenta. As estatísticas fornecidas incluem média, desvio padrão, mínimo, quartis (25%, 50%, 75%), máximo e porcentagem (pct) de ocorrência. A porcentagem de ocorrência indica quantas vezes um modelo específico alcançou o MSE mais baixo em todas as repetições.

Os modelos de aprendizado de máquina como KNN, GBR, SVR e MLP parecem exibir uma maior variabilidade em seus resultados, quando comparados com o modelo de regressão linear LRG, especialmente quando o tamanho da amostra é pequeno. Certos modelos podem se destacar em amostras menores, mas seu desempenho pode diminuir à medida que o tamanho da amostra aumenta, provavelmente devido à sua capacidade de generalização.

Na Tabela 3.2, o LRG Ridge exibe o MSE mais baixo, mas isso muda à medida que o número de amostras do conjuntos de dados aumenta para 116. O modelo KNN tende a apresentar um MSE mais alto em comparação com outros modelos em várias amostras. Por outro lado, SVR e DTR frequentemente exibem valores de MSE mais altos, sugerindo um desempenho relativamente inferior em comparação com outros modelos. Em contraste, GBR e RFR

parecem oferecer um desempenho mais consistente e melhorado em termos de MSE na maioria das amostras.

modelo	média	std	min	25%	50%	75%	max	pct
LRG	19.37	72.58	0.00	0.52	2.42	14.06	1082.50	0.00
KNN	44.38	120.87	0.00	0.62	4.40	30.47	2409.17	129.12
GBR	67.57	169.46	0.00	0.57	6.41	40.80	2616.39	248.87
RFR	77.72	147.49	0.00	5.66	23.53	77.32	2207.78	301.24
DTR	187.78	693.59	0.00	0.22	6.19	58.24	13851.34	869.46
SVR	542.55	808.66	0.00	17.26	188.40	757.96	6745.53	2701.12
MLP	307.44	1083.79	0.00	9.40	48.59	205.29	17213.55	1487.19

Tabela 3.2: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **15** casos.

modelo	média	std	min	25%	50%	75%	max	pct
LRG	20.26	72.38	0.00	0.82	3.59	13.84	1061.72	0.00
KNN	22.13	94.97	0.00	0.17	1.02	10.15	1582.63	9.27
RFR	26.56	73.49	0.00	0.90	4.99	20.57	1145.67	31.11
DTR	43.64	190.00	0.00	0.05	0.54	12.94	6955.62	115.43
GBR	45.78	123.79	0.00	0.19	2.47	23.50	1454.93	126.01
SVR	171.02	382.52	0.00	2.51	26.14	163.41	4993.59	744.35
MLP	277.85	1117.92	0.00	10.13	44.10	157.55	17256.16	1271.42

Tabela 3.3: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **29** casos.

modelo	média	std	min	25%	50%	75%	max	pct
LRG	12.11	63.12	0.00	0.32	1.27	4.16	881.63	0.00
KNN	14.56	72.75	0.00	0.03	0.33	1.80	939.22	20.23
RFR	16.01	67.52	0.00	0.41	2.01	7.48	927.27	32.24
SVR	17.63	63.34	0.00	0.28	1.56	5.88	923.95	45.59
DTR	19.42	108.42	0.00	0.01	0.17	2.25	5854.63	60.41
GBR	23.94	86.88	0.00	0.07	0.67	6.83	1344.98	97.75
MLP	242.51	975.90	0.00	6.54	30.74	127.57	17150.36	1902.55

Tabela 3.4: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **58** casos.

modelo	média	std	min	25%	50%	75%	max	pct
KNN	10.44	68.11	0.00	0.00	0.19	1.13	939.64	0.00
LRG	11.43	58.30	0.00	0.33	1.61	4.34	824.27	9.54
SVR	12.02	59.85	0.00	0.21	1.36	4.75	871.14	15.18
GBR	12.47	59.50	0.00	0.01	0.21	2.21	1132.01	19.51
RFR	12.65	67.54	0.00	0.08	0.48	2.34	1043.20	21.20
DTR	19.44	212.82	0.00	0.00	0.06	0.32	12820.34	86.27
MLP	246.19	1003.10	0.00	6.84	32.79	128.12	17400.17	2258.14

Tabela 3.5: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **116** casos.

	média	std	min	25%	50%	75%	max	pct
modelo								
KNN	6.45	49.02	0.00	0.00	0.01	0.20	647.65	0.00
DTR	9.24	79.85	0.00	0.00	0.00	0.15	8176.77	43.38
RFR	9.58	63.82	0.00	0.02	0.10	0.50	1077.52	48.61
GBR	9.62	57.94	0.00	0.00	0.03	0.43	910.02	49.18
LRG	11.35	58.40	0.00	0.29	1.35	4.67	827.60	76.12
SVR	11.84	61.40	0.00	0.21	1.23	4.22	896.81	83.61
MLP	87.40	639.41	0.00	1.24	6.18	22.42	13472.24	1255.04

Tabela 3.6: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **232** casos.

	média	std	min	25%	50%	75%	max	pct
modelo								
KNN	2.16	23.67	0.00	0.00	0.00	0.02	459.56	0.00
RFR	3.58	22.42	0.00	0.01	0.04	0.22	583.33	65.50
GBR	3.99	28.15	0.00	0.00	0.00	0.13	464.56	84.34
DTR	4.69	36.40	0.00	0.00	0.00	0.03	884.00	116.49
LRG	11.31	56.18	0.00	0.25	1.66	5.92	795.87	422.48
SVR	11.68	61.76	0.00	0.21	1.29	4.06	1268.40	439.73
MLP	43.43	216.90	0.00	0.93	4.55	16.62	7880.86	1910.65

Tabela 3.7: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **346** casos.

modelo	média	std	min	25%	50%	75%	max	pct
KNN	0.27	3.72	0.00	0.00	0.00	0.00	86.54	0.00
DTR	0.82	17.56	0.00	0.00	0.00	0.00	1519.28	204.98
RFR	1.56	8.55	0.00	0.00	0.02	0.10	159.52	482.44
GBR	2.27	20.63	0.00	0.00	0.00	0.06	598.71	746.19
SVR	10.89	59.11	0.00	0.18	1.05	3.70	887.95	3954.88
LRG	11.22	56.53	0.00	0.31	1.53	5.53	798.49	4079.38
MLP	20.21	72.84	0.00	0.59	2.87	10.00	2015.15	7385.18

Tabela 3.8: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **462** casos.

modelo	média	std	min	25%	50%	75%	max	pct
DTR	0.01	0.07	0.00	0.00	0.00	0.00	0.64	0.00
KNN	0.01	0.07	0.00	0.00	0.00	0.00	0.64	0.00
RFR	0.26	1.49	0.00	0.00	0.00	0.02	62.25	1835.95
GBR	1.52	15.46	0.00	0.00	0.00	0.02	551.00	11245.16
SVR	4.29	35.75	0.00	0.02	0.16	0.82	889.11	31877.91
MLP	10.84	46.39	0.00	0.29	1.28	4.29	904.62	80614.81
LRG	11.27	56.87	0.00	0.29	1.50	5.55	806.71	83974.73

Tabela 3.9: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando a amostra de **924** casos.

modelo	média	std	min	25%	50%	75%	max	pct
DTR	0.01	0.07	0.00	0.00	0.00	0.00	0.64	0.00
KNN	0.01	0.07	0.00	0.00	0.00	0.00	0.64	0.00
RFR	0.05	0.51	0.00	0.00	0.00	0.00	23.67	297.24
GBR	1.64	17.45	0.00	0.00	0.00	0.02	627.05	12100.74
SVR	2.53	25.05	0.00	0.01	0.09	0.53	876.32	18763.10
MLP	7.48	37.84	0.00	0.19	0.82	2.72	717.69	55307.41
LRG	11.27	57.00	0.00	0.29	1.44	5.23	801.98	83958.11

Tabela 3.10: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de **1848** casos.

modelo	média	std	min	25%	50%	75%	max	pct
KNN	0.01	0.07	0.00	0.00	0.00	0.00	0.82	0.00
RFR	0.02	0.19	0.00	0.00	0.00	0.00	20.81	37.41
DTR	1.14	58.25	0.00	0.00	0.00	0.00	3836.52	8288.84
GBR	1.38	14.45	0.00	0.00	0.00	0.02	458.63	10134.80
SVR	1.73	19.03	0.00	0.01	0.07	0.41	753.68	12660.57
MLP	4.18	22.01	0.00	0.11	0.52	1.79	608.14	32053.85
LRG	11.30	56.82	0.00	0.30	1.49	5.53	802.62	83377.55

Tabela 3.11: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de **5082** casos.

modelo	média	std	min	25%	50%	75%	max	pct
RFR	0.02	0.11	0.00	0.00	0.00	0.00	14.97	0.00
KNN	0.02	0.09	0.00	0.00	0.00	0.00	1.36	4.20
DTR	0.03	3.42	0.00	0.00	0.00	0.00	761.77	105.03
SVR	1.32	15.54	0.00	0.00	0.06	0.37	651.36	8531.50
MLP	1.49	7.78	0.00	0.05	0.24	0.87	300.29	9271.07
GBR	1.56	15.88	0.00	0.00	0.00	0.02	768.95	10062.78
LRG	11.29	56.80	0.00	0.30	1.52	5.62	804.41	73492.43

Tabela 3.12: Estatísticas de erro quadrático para todos os modelos testados, ordenadas em ordem crescente de MSE utilizando o conjunto de **10164** casos.

Em diferentes tamanhos de conjuntos de dados, modelos específicos demonstraram consistentemente desempenho superior em termos de erro quadrático médio. Por exemplo, na amostra de 15 sinais, o LRG apresentou a média mais baixa, registrando um valor de 19,37. Esta tendência persistiu na amostra de 29 sinais, onde o LRG atingiu novamente o valor mais baixo, com um valor de 20,26. Da mesma forma, na amostra de 58 sinais, o modelo com menor MSE foi novamente o LRG, registrando um valor de 12,11. A tendência continuou na amostra de 116 sinais, onde o LRG atingiu o valor de 10,44. Fazendo a transição para os conjuntos de dados seguintes, a amostra de 232 casos de medição, o KNN é o líder de desempenho em termos de MSE, com um valor de 2,16. Esta tendência foi reafirmada na amostra de 346 sinais, já que o KNN demonstrou mais uma vez a sua eficácia com uma média de 0,27. Na amostra com 462 sinais, o KNN manteve seu desempenho consistente, registrando 0,01.

Com base nos valores MSE, o modelo KNN parece ser uma escolha promissora para conjuntos de dados menores. No entanto, à medida que o tamanho do conjunto de dados aumenta, os modelos podem apresentar desempenho variável. Alguns modelos podem ter desempenho consistentemente melhor ou pior em termos de MSE, enquanto outros podem variar.

À medida que o tamanho dos conjuntos de dados aumentou, surgiu um padrão claro com outro modelo desempenhando melhor e exibindo a média mais baixa. Para o conjunto de 924 pares sinal sintético x deformação, o RFR demonstrou sua eficácia em relação aos demais ao atingir o MSE mais baixo, registrando um valor de 0,02. Esta tendência persistiu no conjunto de 1848 dados, onde RFR tem um MSE médio de 0,02. Demonstrando consistência, a amostra do sinal de 5.082 ondas continuou a enfatizar a superioridade do RFR, registrando 0,02. Este padrão permaneceu ininterrupto mesmo no maior conjunto de dados, onde o RFR continuou a exibir sua robustez ao atingir o MSE mais baixo de 0,02. Assim, com base nos valores médios do MSE, o

modelo RFR parece ser a escolha mais consistente para amostras maiores, demonstrando desempenho robusto com baixo erro quadrático médio.

### 3.7

#### Considerações sobre este capítulo

A partir dos dados experimentais do presente estudo, constituído por uma placa de alumínio submetida a diferentes valores de tensão, foram obtidos dados sintéticos através da aplicação de um método de interpolação linear. Desta forma, foram gerados diversos conjuntos de dados sintéticos, cada um com um número diferente de amostras de deformações e seus respectivos sinais de onda. O tamanho das amostras de deformação geradas varia entre 15 e 10.146 pares de sinal x deformação. Como método de redução da dimensionalidade dos conjuntos de dados, foi utilizado o PCA, que diminuiu o tamanho da matriz de dados sintéticos de 8.192 para 8 componentes. Diferentes conjuntos de dados sintéticos gerados foram utilizados para treinar vários modelos de ML (LRG, SVR, RFR, KNN, DTR, GBR e MLP) com os sinais sendo as entradas dos modelos e as deformações iniciais aplicadas à placa sendo suas saídas, ou seja, os valores previstos para cada modelo.

Dados experimentais foram utilizados para avaliação final dos modelos, comparando os valores reais e os valores previstos pelos modelos. O objetivo principal deste capítulo foi realizar uma análise comparativa de diversos modelos de ML, utilizando no treinamento dados sintéticos gerados por interpolação linear dos dados experimentais, que por sua vez, foram usados para validação.

De forma geral, observou-se que os modelos apresentam uma precisão satisfatória para todos os conjuntos de dados utilizados, variando entre 95% e 99% sendo os melhores modelos RFR e GBR para os casos com maiores amostras. Porém, para casos com amostras menores, os melhores modelos são o LRG e o KNN que apresentam melhor acurácia.

Diferentemente das abordagens anteriores que dependem de métodos numéricos, esta abordagem é bastante adequada na geração de dados sintéticos, expandindo significativamente o conjunto de treinamento para modelos de aprendizado de máquina. O pipeline proposto, com extração de características e busca aleatória, resulta em modelos com até 99% de precisão.

Os modelos de aprendizado de máquina como RFR e GBR mostraram alta precisão para conjuntos de dados maiores, enquanto LRG e KNN se destacaram para conjuntos de dados menores. Notavelmente, além de 924 amostras, a melhoria torna-se marginal, indicando um equilíbrio entre melhores resultados e custos computacionais.

Em resumo, a geração eficiente de dados sintéticos e o desempenho

robusto dos modelos de aprendizado de máquina oferecem uma contribuição significativa para a análise de tensão em placas. A abordagem de interpolação linear a partir de dados experimentais destaca-se como uma solução viável, eliminando a necessidade de métodos complexos para expansão da base de dados.

## 4

### Método SAFE no cálculo de curvas de dispersão em cilindros

Este capítulo aborda a aplicação do método SAFE para cálculo de curvas de dispersão em guias de onda cilíndricos e está relacionado ao artigo intitulado *A comparative Study of Dispersion Curves in Cylindrical Waveguide Using the Semi-Analytical Finite Element Method*, que foi apresentado no congresso DINAME2023.

As inspeções ultrassônicas são uma técnica não destrutiva amplamente utilizada na análise de estruturas civis e mecânicas, sem afetar o funcionamento dos equipamentos ou prejudicar sua integridade. No procedimento, um transmissor emite pulsos acústicos que se propagam pelos elementos estruturais. De acordo com a resposta registrada pelos receptores, é possível detectar avarias, pois a presença de danos, ou falhas altera a forma das curvas de dispersão e dos modos de onda, em comparação com a resposta acústica de uma estrutura sem anomalias.

Obter as curvas de dispersão significa encontrar as raízes de um sistema de equações, para o qual podem ser utilizados algoritmos computacionais iterativos. Uma maneira de resolver o sistema é usar métodos numéricos, como por exemplo, elementos finitos. Porém, em guias de onda mais complexos ou muito longos, os custos computacionais são elevados ou inviáveis. Esta dificuldade pode ser superada com uma combinação de solução analítica e método dos elementos finitos. Neste capítulo é feita uma comparação considerando as respostas obtidas na propagação de ondas em guias cilíndricos utilizando o MEF do software COMSOL e o SAFE implementado em Matlab.

#### 4.1

##### Formulação do Problema

A estrutura analisada é um cilindro oco sem tensões (Fig. 4.1) com comportamento linear elástico, modelado em coordenadas cilíndricas. Considerando o já mencionado método semianalítico, as soluções harmônicas analíticas exatas serão utilizadas nas direções  $\theta$  e  $z$  e a aproximação por elementos finitos será aplicada na dimensão radial.

Para resolver problemas envolvendo ondas mecânicas elásticas que se propagam em materiais sólidos utiliza-se a equação de Navier-Cauchy (4-1)

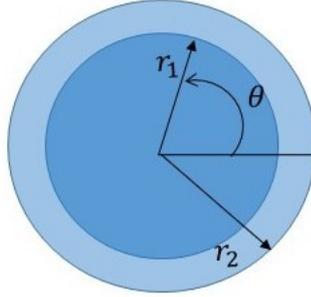


Figura 4.1: Cilindro com raio interno de  $r_1 = 0,30 \text{ m}$  e raio externo de  $r_2 = 0,33 \text{ m}$ .

já apresentada no Capítulo 2. Para sua resolução, aplica-se o teorema de Helmholtz (4-2)

$$(\lambda + \mu)\nabla(\nabla \cdot \mathbf{u}) + \mu\nabla^2\mathbf{u} = \rho\ddot{\mathbf{u}} \quad (4-1)$$

$$\mathbf{u} = \nabla\phi + \nabla \times \boldsymbol{\psi}. \quad (4-2)$$

Em coordenadas cilíndricas, o vetor potencial da equação 4-2 pode ser decomposto em dois potenciais escalares [77, 78], cujo resultado é a equação (4-3) abaixo:

$$\boldsymbol{\psi} = \eta\mathbf{e}_3 + \nabla \times (\chi\mathbf{e}_3), \quad (4-3)$$

assim:

$$\mathbf{u} = \nabla\phi + \nabla \times (\eta\mathbf{e}_3) + \nabla \times \nabla \times (\chi\mathbf{e}_3) \quad (4-4)$$

onde

$$\nabla^2\phi - \frac{1}{c_L^2}\ddot{\phi} = 0, \quad \nabla^2\eta - \frac{1}{c_T^2}\ddot{\eta} = 0 \quad \text{e} \quad \nabla^2\chi - \frac{1}{c_T^2}\ddot{\chi} = 0, \quad (4-5)$$

Passando para coordenadas polares, o campo de deslocamento é dado por:

$$\begin{aligned} u_r &= \frac{\partial^2\phi}{\partial r^2} + \frac{1}{r}\frac{\partial\eta}{\partial\theta} + \frac{\partial^2\chi}{\partial z\partial\theta} \\ u_\theta &= \frac{1}{r}\frac{\partial^2\phi}{\partial r^2} - \frac{\partial\eta}{\partial r} + \frac{1}{r}\frac{\partial^2\chi}{\partial z\partial\theta} \\ u_z &= \frac{\partial^2\phi}{\partial z^2} + \frac{\partial^2\chi}{\partial z^2} - \frac{1}{c_T^2}\frac{\partial^2\chi}{\partial t^2} \end{aligned} \quad (4-6)$$

É necessário resolver o sistema de equações diferenciais e encontrar o campo de deslocamentos ao longo do tempo. De acordo com a natureza do problema em questão e tendo em vista o reconhecimento das propriedades

físicas constantes do guia de ondas ao longo da direção de propagação, utilizando a abordagem semianalítica, o campo de deslocamento que representa o movimento da onda é considerado como:

$$\mathbf{u}^{(e)}(r, \theta, z, t) = \mathbf{U}(x, y)e^{i(kz+n\theta-\omega t)}, \quad (4-7)$$

Para o propósito deste trabalho, métodos numéricos podem ser utilizados, como o método dos elementos finitos. A discretização por elementos finitos leva a uma formulação fraca da equação de energia. O problema da propagação das ondas se reduz a um sistema de equações algébricas, a partir das quais a dispersão pode ser obtida.

Neste trabalho, o método SAFE é utilizado. Em outras palavras, esta é a solução analítica juntamente com o método dos elementos finitos. A solução depende do tempo  $t$  e da coordenada de propagação  $z$ , assim, o problema assume a seguinte forma (4-8):

$$\mathbf{u}^{(e)}(r, \theta, z, t) = \sum_{j=1}^n \mathbf{N}(r)\mathbf{q}^{(e)}e^{i(kz+n\theta-\omega t)}, \quad (4-8)$$

onde  $\mathbf{u}^{(e)}$  representa o campo de deslocamentos,  $x, y$  e  $z$  são as coordenadas espaciais,  $\mathbf{q}^{(e)}$  é o vetor de deslocamento nodal,  $k$  é o número de onda na direção axial,  $n$  é o harmônico circunferencial representado por um número inteiro,  $\omega$  é a frequência, e  $t$  corresponde ao tempo. As funções de forma  $\mathbf{N}$  que correspondem ao elemento da Fig 4.2, são dadas por:

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 \end{bmatrix}, \quad (4-9)$$

com as funções de forma lineares

$$N_1 = \frac{1}{2}(1 - \xi) \quad \text{and} \quad N_2 = \frac{1}{2}(1 + \xi). \quad (4-10)$$

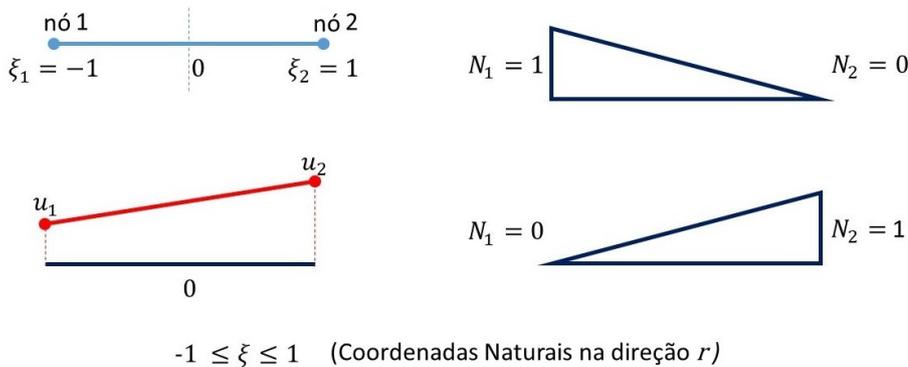


Figura 4.2: Funções de forma adotadas.

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{rr} \\ \varepsilon_{\theta\theta} \\ \varepsilon_{zz} \\ \gamma_{\theta z} \\ \gamma_{rz} \\ \gamma_{r\theta} \end{bmatrix} = \begin{bmatrix} \frac{\partial u_r}{\partial r} \\ \frac{u_r}{r} + \frac{1}{r} \frac{\partial u_\theta}{\partial \theta} \\ \frac{\partial u_z}{\partial z} \\ \frac{1}{r} \frac{\partial u_z}{\partial \theta} + \frac{\partial u_\theta}{\partial z} \\ \frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \\ \frac{1}{r} \frac{\partial u_r}{\partial \theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r} \end{bmatrix} \Rightarrow \boldsymbol{\varepsilon} = (\mathbf{L}_1 + \mathbf{L}_2) \mathbf{u}, \quad (4-11)$$

com

$$\mathbf{L}_1 = \begin{bmatrix} \frac{\partial}{\partial r} & 0 & 0 \\ \frac{1}{r} & \frac{1}{r} \frac{\partial}{\partial \theta} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{r} \frac{\partial}{\partial \theta} \\ 0 & 0 & \frac{\partial}{\partial r} \\ \frac{1}{r} \frac{\partial}{\partial \theta} & \frac{\partial}{\partial r} - \frac{1}{r} & 0 \end{bmatrix} \quad \text{e} \quad \mathbf{L}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ 0 & \frac{\partial}{\partial z} & 0 \\ \frac{\partial}{\partial z} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (4-12)$$

Substituindo as funções de forma nas relações deformação-deslocamento, ou seja, substituindo 4-8 por 4-11, a deformação no elemento pode ser expressa por:

$$\boldsymbol{\varepsilon} = (\mathbf{L}_1 + \mathbf{L}_2) \mathbf{N}(r) \mathbf{q}^{(e)} e^{i(kz+n\theta-\omega t)} = (\mathbf{B}_1 + ik\mathbf{B}_2) \mathbf{q}^{(e)} e^{i(kz+n\theta-\omega t)}, \quad (4-13)$$

onde a matriz  $\mathbf{B}$  é a matriz de compatibilidade geométrica, composta pelas funções de forma e suas derivadas:

$$\mathbf{B}_1 = \begin{bmatrix} \frac{dN_1}{dr} & 0 & 0 & \frac{dN_2}{dr} & 0 & 0 \\ \frac{N_1}{r} & i\frac{n}{r}N_1 & 0 & \frac{N_2}{r} & i\frac{n}{r}N_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & i\frac{n}{r}N_1 & 0 & 0 & i\frac{n}{r}N_2 \\ 0 & 0 & \frac{dN_1}{dr} & 0 & 0 & \frac{dN_2}{dr} \\ i\frac{n}{r}N_1 & \frac{dN_1}{dr} - \frac{N_1}{r} & 0 & i\frac{n}{r}N_2 & \frac{dN_2}{dr} - \frac{N_2}{r} & 0 \end{bmatrix},$$

$$\mathbf{B}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 \\ 0 & N_1 & 0 & 0 & N_2 & 0 \\ N_1 & 0 & 0 & N_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (4-14)$$

A equação 4-13 pode ser substituída na relação constitutiva para obter os componentes do tensor de tensões conforme a seguir:

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon} = \mathbf{C}(\mathbf{B}_1 + ik\mathbf{B}_2) \mathbf{q}^{(e)} e^{i(kz+n\theta-\omega t)}, \quad (4-15)$$

onde  $\boldsymbol{\sigma}$  é a tensão e  $\mathbf{C}$  é o tensor constitutivo para materiais elásticos.

Substituindo os valores de deslocamento, deformação e tensão na equação que governa o problema, temos a equação elementar:

$$(\mathbf{K}_1^{(e)} + ik\mathbf{K}_2^{(e)} + k^2\mathbf{K}_3^{(e)}) \mathbf{U}^{(e)} - \omega^2\mathbf{M}^{(e)}\mathbf{U}^{(e)} = 0, \quad (4-16)$$

com

$$\begin{aligned}
 \mathbf{K}_1^{(e)} &= \int_r \int_\theta \mathbf{B}_1^T \mathbf{C} \mathbf{B}_1 r dr d\theta \\
 \mathbf{K}_2^{(e)} &= \int_r \int_\theta (\mathbf{B}_1^T \mathbf{C} \mathbf{B}_2 - \mathbf{B}_2^T \mathbf{C} \mathbf{B}_1) r dr d\theta \\
 \mathbf{K}_3^{(e)} &= \int_r \int_\theta \mathbf{B}_2^T \mathbf{C} \mathbf{B}_2 r dr d\theta \\
 \mathbf{M}^{(e)} &= \rho \int_r \int_\theta \mathbf{N}^T \mathbf{N} r dr d\theta
 \end{aligned} \tag{4-17}$$

Finalmente, o problema de autovalor generalizado em coordenadas globais pode ser obtido adotando uma metodologia convencional de montagem de elementos finitos:

$$(\mathbf{K}_1 + ik\mathbf{K}_2 + k^2\mathbf{K}_3 - \omega^2\mathbf{M}) \mathbf{Q} = 0, \tag{4-18}$$

## 4.2 Metodologia e Resultados

O método SAFE foi implementado em Matlab, utilizando elementos unidimensionais na direção radial da seção transversal do cilindro e aplicando a solução analítica nas direções longitudinal e circunferencial. Para validar a implementação foi feita uma comparação com as curvas de dispersão obtidas analiticamente [116], conforme mostrado na Fig 5.3.

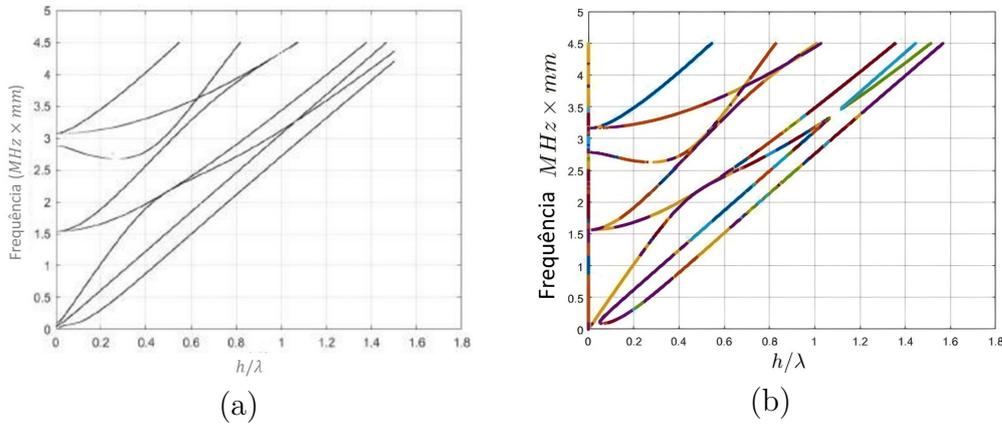


Figura 4.3: A imagem (a) mostra a solução analítica e (b) mostra as curvas de dispersão calculadas pelo método SAFE.

As curvas de dispersão calculadas pelo método SAFE apresentam boa concordância com as curvas obtidas analiticamente. Uma vez validada a implementação, um modelo diferente foi construído no COMSOL para que os resultados pudessem ser avaliados pelo método dos elementos finitos. Porém, devido às limitações de memória computacional e à necessidade de

discretização mais refinada, o modelo numérico precisa ter características diferentes em relação ao modelo inicial.

A primeira modificação foi realizada na faixa de frequência utilizada. Foi reduzido de 5 MHz para 50 kHz. A geometria também foi modificada. A relação entre raio e espessura afeta a curva de dispersão [117], possibilitando a obtenção de mais pontos numa faixa de frequência mais baixa com uma relação  $R/d$  mais alta. Desta forma, as dimensões do modelo foram alteradas.

A estrutura cilíndrica semelhante à mostrada na Fig. 4.1 com as características apresentadas na Tab. 4.1 foi modelado no COMSOL Multiphysics, que é uma plataforma para resolver equações diferenciais usando métodos numéricos. As respostas obtidas foram utilizadas em um algoritmo Matlab para obtenção das curvas de dispersão. Estas curvas de dispersão foram comparadas com o resultado obtido pelo método SAFE, também implementado em Matlab.

Tabela 4.1: Parâmetros do cilindro

Parâmetro	Símbolo	Valor	Units
Módulo de Young	$E_0$	$193 \times 10^9$	Pa
Coefficiente de Poisson	$\nu$	0.27	-
Densidade	$\rho$	7860	kg/m <sup>3</sup>
Frequência	$f_0$	100,000	Hz
Comprimento	$L$	1.0	m
Raio interno	$r_1$	0.10	m
Raio externo	$r_2$	0.20	m
Tempo	$t_f$	$9.23 \times 10^{-3}$	s
Tamanho do intervalo de tempo	$\Delta t$	$7.69 \times 10^{-7}$	s
Velocidade da onda	$c_s$	4.955	m/s

A partir das duas imagens mostradas na Fig. 4.4 é possível ver uma correlação aceitável. Para obter as curvas mostradas em (a), foi necessário um tempo de processamento de 104.575 s, ou seja, 1 dia, 5 horas, 2 minutos e 55 segundos. As curvas mostradas em (b) foram obtidas após um processamento de 474 segundos. Ambos os modelos foram calculados em um computador com 3,60 GHz (4 processadores) e 2,0 TB de RAM. Para melhorar a comparação seria necessário aumentar os intervalos de tempo e reduzir ainda mais a discretização da estrutura modelada no COMSOL, o que incorreria em custos computacionais ainda maiores.

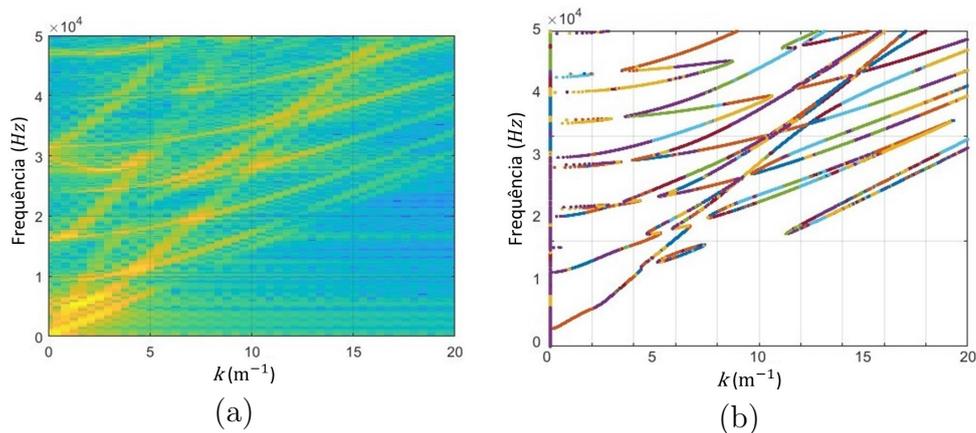


Figura 4.4: No gráfico, em (a) são mostradas as curvas de dispersão obtidas com COMSOL e em (b) as curvas de dispersão obtidas com o método SAFE no Matlab.

### 4.3

#### Considerações sobre este capítulo

O estudo da propagação de ondas elásticas em guias de onda muito longos pode resultar em elevados custos computacionais devido à discretização necessária para representar os comprimentos de onda na direção longitudinal. Portanto, a implementação do método SAFE tem uma boa aceitação para a análise deste tipo de estrutura.

O método semianalítico foi implementado em Matlab e validado através da conferência com a solução analítica. Sua eficiência foi comprovada através da comparação com a aproximação da solução numérica processada no COMSOL Multiphysics resultando em economia de tempo de processamento.

## 5

### Rede neural para predição da espessura de dutos

Este capítulo trata da avaliação da espessura nominal de dutos utilizando redes neurais e está relacionado ao artigo intitulado *Evaluation of the nominal thickness of pipelines using neural networks through the dispersion curves drawn with the SAFE method*, que foi enviado ao CBIC2023.

A inspeção por meio de ensaios não destrutivos é importante para a análise de estruturas. A propagação de ondas ultrassônicas é uma aliada na avaliação da integridade de componentes estruturais, através do traçado de curvas de dispersão, que mostram o par frequência x comprimento de onda para um guia de ondas específico.

Uma estrutura de rede neural é utilizada para estimar a espessura nominal de dutos. As entradas da rede neural são curvas de dispersão obtidas por simulação, utilizando o método SAFE, que apresenta um custo computacional reduzido em comparação com uma simulação totalmente numérica.

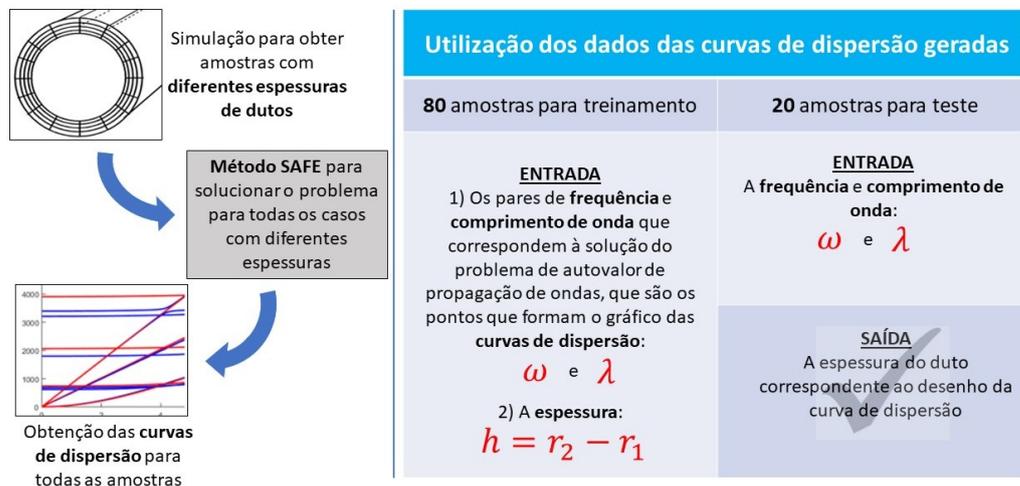


Figura 5.1: Resumo de como foi realizado este estudo para avaliar a espessura nominal de dutos utilizando redes neurais treinadas com curvas de dispersão obtidas pelo método SAFE.

#### 5.1

##### Problema Analisado

A estrutura analisada é a mesma utilizada no Capítulo 4. É um cilindro oco sem tensões como o mostrado na Fig 5.2 com comportamento linear elástico

e espessura inicial de 0,03 m. Como foram utilizadas 100 simulações, cada uma teve uma variação de  $2,0 \times 10^{-5}$  m na espessura inicialmente adotada. Para cada um destes cilindros, existe um conjunto associado de curvas de dispersão. Essas curvas de dispersão foram usadas como entrada para treinamento de aprendizado de máquina.

O método SAFE foi implementado em Matlab [40], utilizando elementos unidimensionais na direção radial da seção transversal do cilindro e a solução analítica nas direções longitudinal e circunferencial. As curvas de dispersão obtidas [40], são mostradas na Fig. 5.3, onde a razão  $h/\lambda$  é a frequência por comprimento de onda (valor adimensionalizado),  $h$  é a espessura do cilindro e  $\lambda$  é o comprimento de onda.

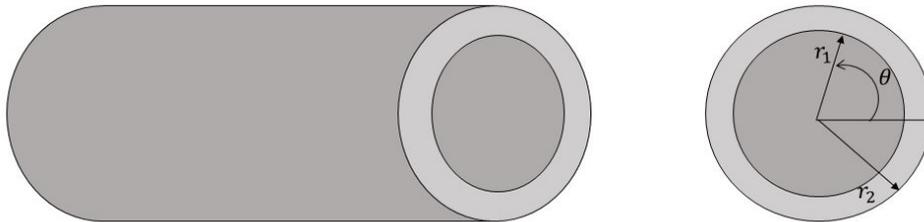


Figura 5.2: O modelo de cilindro livre de tensões que foi usado.

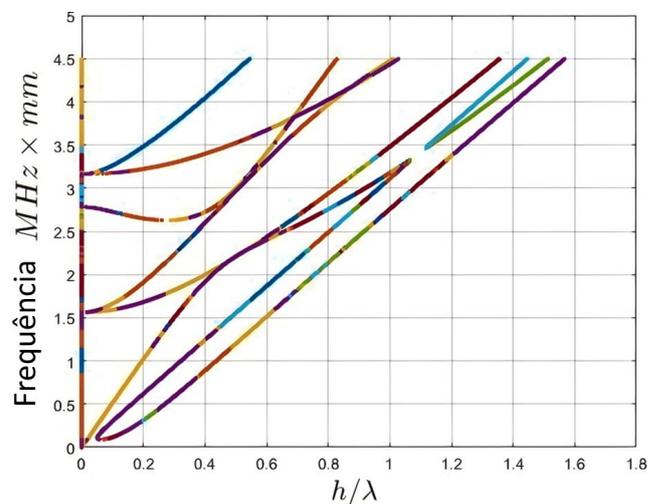


Figura 5.3: As curvas de dispersão para um cilindro foram calculadas usando o método SAFE.

## 5.2

### Metodologia

Em guias de onda muito longos, como trilhos de trem ou oleodutos de produção de poços de petróleo, o estudo da propagação de ondas elásticas

pode resultar em altos custos computacionais devido à discretização necessária para representar os comprimentos de onda na direção longitudinal [34, 37, 38]. Utilizando o método SAFE, conforme já exposto no Capítulo 4, a discretização da estrutura em elementos finitos é feita no sentido radial, obtendo-se um campo de deslocamentos aproximado. Tanto na direção circunferencial quanto na longitudinal, utiliza-se a solução analítica com a representação harmônica da onda no domínio do tempo [40].

### 5.2.1

#### Rede neural usada

A CNN é uma Rede Neural Artificial (em inglês Artificial Neural Network - ANN) avançada definida como uma estrutura representativa de aprendizagem profunda baseada em um mecanismo de percepção visual e computação convolucional [118]. Sendo diferente da ANN tradicional, a CNN pode capturar diretamente as características espaciais das imagens para melhorar a precisão e a eficiência da previsão [119].

A CNN tem sido amplamente empregada em diversos campos, como visão computacional, incluindo classificação de imagens [120], rastreamento de objetos, reconhecimento de ação; processamento de linguagem natural [121], classificação e previsão de séries temporais [49, 118, 122]. A estrutura principal da CNN inclui uma camada convolucional, uma camada de pooling e uma camada totalmente conectada.

Na camada convolucional, um neurônio está conectado a apenas uma parte dos neurônios vizinhos, o que reduz a complexidade da rede e o número de parâmetros [118]. Esta camada é usada para extrair as características das imagens de entrada aplicando um conjunto de filtros a esses dados de entrada, produzindo um conjunto de mapas de características de saída [122], conforme observado na Equação 5-1, que representa a equação da camada convolucional e os filtros que são aplicados à matriz de entrada.

$$\mathbf{y}_{ij}^{(l)} = \mathbf{f} \left( \sum_{m=1}^M \sum_{n=1}^N \mathbf{W}_{mn}^{(l)} \cdot \mathbf{x}_{(i+m)(j+n)}^{(l-1)} + \mathbf{b}^{(l)} \right) \quad (5-1)$$

onde,  $\mathbf{x}$  denota a matriz que captura os dados de entrada. Os pesos aprendidos  $\mathbf{W}$  são filtros convolucionais aplicados à matriz de entrada, com os tamanhos de filtro específicos e o número de filtros dependendo do problema em questão. O termo de polarização  $\mathbf{b}$  é adicionado à operação convolucional. A saída resultante  $\mathbf{y}$  é obtida aplicando a função de ativação  $\mathbf{f}$  elemento a elemento e o índice  $l$  representa a camada na rede neural convolucional na qual a operação está sendo executada.

A camada de *pooling* é empregada para reduzir as dimensões dos mapas

com as principais características, também chamados de *feature maps* e assim, manter informações relevantes dos frames transversais das imagens obtidas a partir da camada convolucional [123].

Neste problema de regressão, a camada de *pooling* pode ser aplicada após a camada convolucional. A entrada para a camada de *pooling* pode ser a saída da camada convolucional, resultando em uma representação de tamanho reduzido, aumentando a velocidade de computação. Esta operação ajuda a evitar ajuste excessivo [118]. O tipo de *pooling* usado e o tamanho da janela de *pooling* podem afetar a capacidade da rede de detectar recursos específicos.

Existem basicamente dois tipos de *pooling*: o máximo e o médio. O *pooling* máximo seleciona os valores máximos de um kernel específico aplicado à imagem, enquanto o pool médio calcula o valor médio dentro do kernel especificado na imagem [124]. Esses métodos de agrupamento desempenham um papel crucial na redução da resolução dos mapas de características, permitindo a extração de características dominantes e ao mesmo tempo reduzindo as dimensões espaciais [125].

A camada totalmente conectada classifica as características extraídas das camadas anteriores [49], se concentrando na transformação dos recursos agrupados resultantes das camadas de *pooling* em uma representação adequada para regressão. Os recursos agrupados são achatados em um vetor unidimensional e então vinculados a uma camada totalmente conectada [124]. A multiplicação de matrizes entre as camadas totalmente conectadas, está representada na Equação 5-2. A função de ativação ( $\sigma$ ) é aplicada a esta saída. Esta camada finaliza a transformação dos recursos extraídos em uma previsão final.

$$\mathbf{y}^{(l)} = \sigma \left( \mathbf{W}^{(l)} \cdot \mathbf{x}^{(l-1)} + b^{(l)} \right) \quad (5-2)$$

Nesta arquitetura, o número de neurônios nas camadas totalmente conectadas pode ser customizado com base na complexidade do problema de regressão. As funções de ativação nas camadas totalmente conectadas podem ser escolhidas com base nos requisitos do problema, como **ReLU** e **tanh**, permitindo mapeamentos não lineares entre as características e os valores alvo.

### 5.2.2 Arquitetura Proposta

Neste trabalho, é proposta uma arquitetura CNN baseada em LeNet-5 para estimar a espessura do duto. A arquitetura da CNN consiste em duas camadas convolucionais, duas camadas de *pooling* máximo, duas camadas totalmente conectadas e uma camada de saída, conforme ilustrado na Fig. 5.4.

Para este modelo, é usado um array 2D como dados de entrada. Portanto, como o resultado da simulação é um pré-processamento de dados de 2048x66, tem que ser feita uma seleção de características com base na variância para reduzir a quantidade de dados. Como resultado, uma matriz de 2048x32 foi obtida e utilizada como dado de entrada. Cada simulação foi considerada uma amostra para treinamento e validação do modelo. Na CNN, outra extração automática de características foi realizada por dois conjuntos de camadas convolucionais seguidas por camadas de *pooling* máximo.

Para definir a estrutura ideal da CNN, foi realizada uma busca em grade variando os filtros nas camadas convolucionais, o tamanho do kernel e o agrupamento máximo nas camadas de *pooling*. Para as camadas densas, foram utilizados nós de tamanhos 120 e 84. Apenas a função ReLU foi utilizada como função de ativação na camada densa.

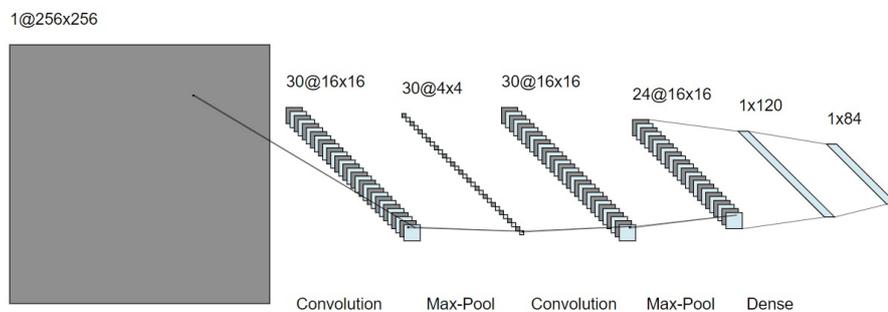


Figura 5.4: Representação esquemática ilustrando o procedimento de pré-processamento e a arquitetura da CNN proposta para avaliação da espessura nominal de dutos.

### 5.3 Resultados e Discussão

Esta seção foca na avaliação do desempenho da arquitetura CNN proposta. Para avaliar sua eficácia, diversas métricas de regressão foram empregadas, incluindo *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE) e *Maximum Error* (MAXE). As equações para essas métricas utilizadas são apresentadas abaixo.

Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (5-3)$$

Mean Absolute Error (MAE):

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (5-4)$$

Maximum Error (MAXE):

$$\text{MAXE} = \max_i |y_i - \hat{y}_i| \quad (5-5)$$

onde  $\mathbf{y}_i$  representa os valores verdadeiros,  $\hat{\mathbf{y}}_i$  os valores preditos, e  $\mathbf{n}$  é o número de amostras.

### 5.3.1

#### Seleção de Hiperparâmetros

Um método de busca em grade foi usado para selecionar os hiperparâmetros. Os conjuntos de dados foram divididos aleatoriamente em um conjunto de dados de treinamento contendo 80% das amostras e um conjunto de dados de teste contendo os 20% restantes.

A Tabela 5.1 resume os resultados obtidos ao longo da fase de teste. Tanto o RMSE quanto o MAE são métricas que medem a magnitude média do erro entre os valores reais e previstos. A diferença é que o RMSE é uma regra quadrática que dá alto peso aos erros mais elevados e o MAE dá uma medida da magnitude média do erro, ignorando a direção.

Tabela 5.1: Métricas de regressão obtidas com espessura prevista dos dutos

Conv. Filter	Kernel Size	Max Pool Size	RMSE	MAE	MAXE
5	8	2	0.015423	0.000238	0.01728
5	8	4	0.016153	0.000261	0.018012
5	16	2	0.014697	0.000216	0.016551
5	16	4	0.006617	0.000044	0.010102
5	32	2	0.015045	0.000226	0.0169
5	32	4	0.012906	0.000167	0.014754
15	8	2	0.017342	0.000301	0.019204
15	8	4	0.011582	0.000134	0.016487
15	16	2	0.011185	0.000125	0.013116
15	16	4	0.010304	0.000106	0.012862
15	32	2	0.014977	0.000224	0.016832
15	32	4	0.013527	0.000183	0.015378
30	8	2	0.015541	0.000242	0.017398
30	8	4	0.012125	0.000147	0.013272
30	16	2	0.014913	0.000222	0.016769
<b>30</b>	<b>16</b>	<b>4</b>	<b>0.002401</b>	<b>0.000006</b>	<b>0.007419</b>
30	32	2	0.011271	0.000127	0.013111
30	32	4	0.01225	0.00015	0.014096

Os resultados mostram que a média do RMSE foi superior à média do MAE indicando variações na magnitude dos erros. Apesar disso, ambas as métricas ficaram próximas do valor ideal de 0,0. O MAXE representa o erro máximo para um conjunto de amostras de teste. O fato de o valor médio de

MAXE ter sido superior a 1,0 indica que alguns dos modelos cometeram erros relativamente grandes.

Analisando a Fig. 5.5 e a Fig. 5.6, pode-se observar que o modelo converge rapidamente em algumas iterações. Não foi observado sinal de *overfitting* como perda de validação, devido à não ocorrência do aumento na previsão do erro. Esta é uma informação importante para avaliar a qualidade do modelo e pode ser usada para justificar a escolha de uma abordagem de treinamento específica ou configuração do modelo.

As Figuras 5.7 e 5.8 mostram que apesar de ter usado um pequeno conjunto de dados para treinamento, o modelo é capaz de fazer previsões razoavelmente boas na fase de teste. Como também pode ser visto na figura, o modelo forneceu uma boa aproximação entre os valores previstos e reais. O resultado do modelo com melhor desempenho é apresentado na tabela 5.1. Para este caso, o RMSE, MAE e MAXE foram iguais a 0,002401; 0,000006 e 0,007419; respectivamente.

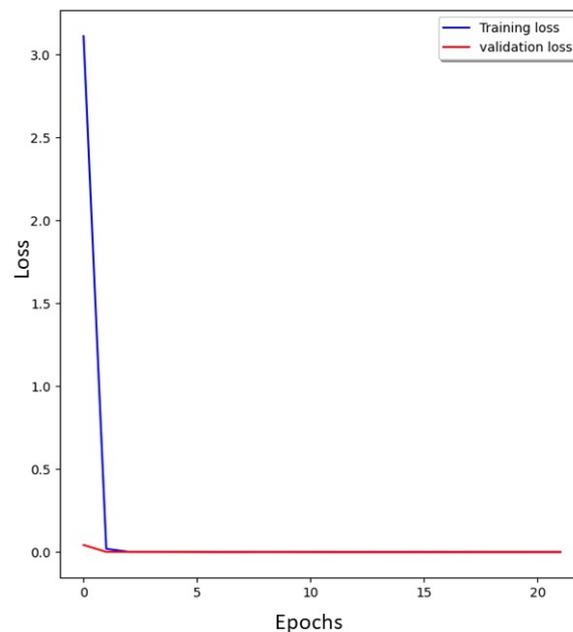


Figura 5.5: LOSS de treinamento e teste.

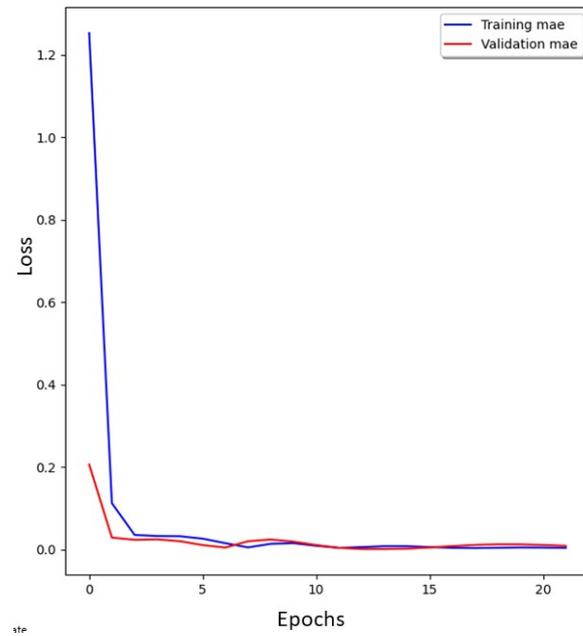


Figura 5.6: MAE para treinamento e teste.

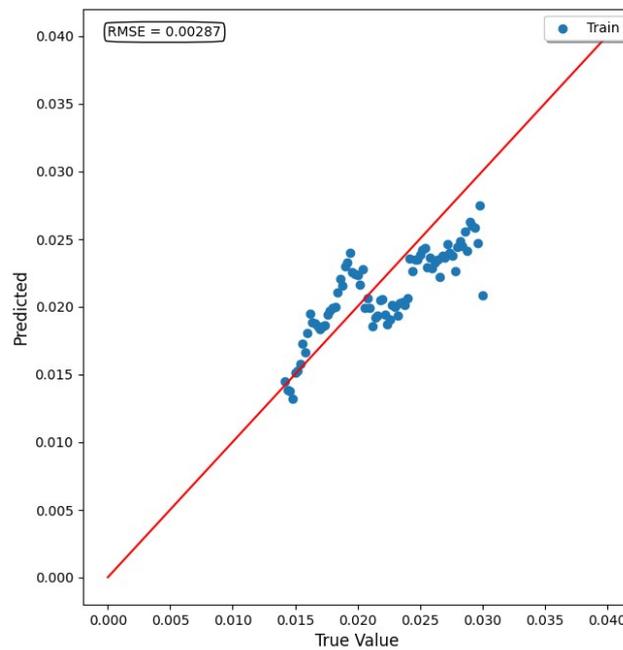


Figura 5.7: Diagrama de dispersão das previsões do treinamento

#### 5.4 Considerações finais desde capítulo

O método SAFE implementado no MATLAB foi utilizado para gerar curvas de dispersão referentes a guias de onda cilíndricos com diferentes espessuras. O par de frequências e números de onda, referentes às curvas obtidas, serviram como dados de entrada para aprendizado de máquina com a

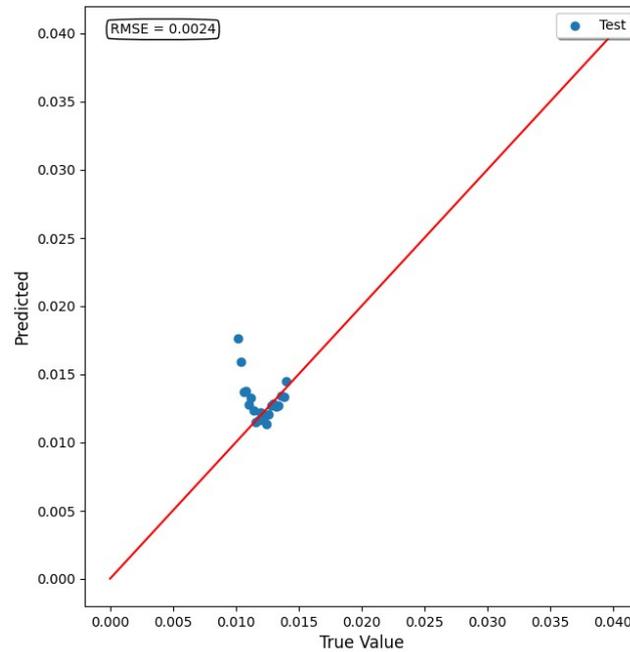


Figura 5.8: Diagrama de dispersão dos testes

técnica de redes neurais convolucionais.

Os resultados referentes ao treinamento indicam que o modelo converge rápido, ou seja, com poucas iterações. Além disso, não há sinais de *overfitting*, bem como a métrica de validação não piora, indicando uma boa generalização para estudos de guias de onda cilíndricos com diferentes espessuras.

Embora poucos dados tenham sido usados, o gráfico de predição mostra que o modelo é capaz de prever razoavelmente bem na fase de teste para uma condição de dados fora do envelope usado no treinamento. O RMSE é adequado para aplicações práticas e a previsão na fase de treinamento indica que ainda é possível melhorar este valor.

Com o que foi observado e relatado, pode-se destacar que a utilização dos resultados do método SAFE tem potencial para economizar tempo e reduzir a necessidade da coleta de dados *in situ*, o que pode levar a redução de custos financeiros e exposição dos trabalhadores a riscos, uma vez que a utilização de estudos com dados numéricos e computacionais a simulação proporcionou a obtenção de um número suficiente de dados. Isso evitaria a instalação obrigatória de bancadas de testes experimentais ou o deslocamento de equipes para trabalho de campo.

## 6

### Aprendizagem por transferência

Este capítulo explora a eficácia das redes neurais convolucionais (CNNs) e da Aprendizagem por Transferência (do inglês *Transfer Learning* ou TL) utilizando dois tipos de dados rotulados: os sinais de ondas ultrassônicas do experimento descrito no Capítulo 3 e os sinais sintéticos obtidos por interpolação linear a partir dos dados experimentais. O presente estudo avalia o desempenho de CNNs e modelos de transfer learning em termos de tempo de treinamento, precisão, incluindo ajuste fino dos dados.

*Transfer learning* é uma técnica de aprendizado de máquina na qual um modelo pré-treinado em uma tarefa é aproveitado como ponto de partida fundamental para resolver outra tarefa relacionada [64, 108, 126, 127]. Em vez de começar do zero e treinar um modelo desde o início, envolve a utilização de um modelo que já adquiriu conhecimentos gerais valiosos de uma tarefa anterior, que pode então ser ajustado para uma nova tarefa. Essa reutilização de informação e a transferência de conhecimentos previamente adquiridos têm o potencial de aumentar significativamente a eficiência do processo de aprendizagem [128]. Esta abordagem é particularmente vantajosa quando se lida com um conjunto de dados limitado para uma nova tarefa, uma vez que o modelo pré-treinado já capturou numerosos padrões subjacentes nas suas experiências de aprendizagem anteriores.

#### 6.1

##### Abordagens de aprendizagem por transferência

Redes Neurais Convolucionais (CNNs) são uma categoria de redes neurais projetadas para processar dados com uma estrutura de grade, como imagens [64]. Esses modelos são compostos por camadas de convolução que extraem características importantes das imagens, seguidas de camadas de *pool* para reduzir a dimensionalidade. Eles são amplamente utilizados em tarefas de classificação de imagens e também em detecção de objetos e segmentação semântica.

A capacidade de aprender características hierárquicas é uma de suas características mais marcantes, porém, pode ser computacionalmente cara e exigir grandes volumes de dados de treinamento [129]. Eles não são ideais para

tarefas de aprendizagem onde os dados não possuem uma estrutura de grade.

### 6.1.1

#### Visual Geometry Group

O modelo VGG é conhecido por sua simplicidade e uniformidade na arquitetura. É composto por várias camadas convolucionais empilhadas, seguidas por camadas de *pooling* e camadas totalmente conectadas. Foi projetado para classificação de imagens, mas suas arquiteturas podem ser adaptadas para outros tipos de tarefas e são amplamente utilizadas como modelos de aprendizagem por transferência. No entanto, pode não ser tão eficiente quanto modelos posteriores, como ResNets, em termos de utilização de recursos [130].

A série VGG inclui diversas arquiteturas de redes neurais convolucionais e neste trabalho foi utilizado o VGG16, que é uma variante da arquitetura VGG. Ele adiciona uma camada de especificidade à arquitetura VGG, destacando seu uso para tarefas de classificação de imagens. É conhecido por sua profundidade, composta por 16 camadas de convolução e *pooling*. A arquitetura VGG16 é caracterizada por camadas de convolução de tamanho pequeno (3x3) e camadas de *pooling* que reduzem a resolução da imagem [131].

### 6.1.2

#### Redes Residuais

As Redes Residuais, ou ResNets, introduziram um conceito inovador de “pular conexões” que ajudam a combater o problema do desaparecimento de gradientes. São conhecidos por suas arquiteturas profundas, permitindo a construção de redes com centenas de camadas. Por ser uma rede neural profunda, necessita de mais recursos computacionais para treinamento e inferência [132, 133].

### 6.1.3

#### MobileNet

O MobileNet foi projetado com foco na eficiência e na leveza, tornando-o ideal para aplicações em dispositivos móveis e sistemas com recursos limitados. Este modelo é notável pela sua capacidade de oferecer alto desempenho em um tamanho compacto. Sua arquitetura foi otimizada para economizar recursos computacionais, essencial para aplicações em tempo real. No entanto, pode ter desempenho um pouco inferior em comparação com modelos mais pesados, como ResNets [134, 135].

#### 6.1.4

##### MobileNetV2

MobileNetV2 é uma evolução do MobileNet original e continua focando em eficiência e leveza. Introduz mudanças na arquitetura que visam melhorar o desempenho em tarefas de visão computacional, por isso apresenta um desempenho geral superior quando comparado ao seu irmão MobileNet. Porém, pode ser facilmente superado em termos de desempenho por modelos mais pesados em tarefas mais complexas [136].

#### 6.1.5

##### Xception

Inspirado em Redes Neurais Convolucionais Profundas (ConvNets), o Xception adota uma abordagem de separação por convolução em duas etapas: convolução espacial e convolução de canal. Este modelo apresenta uma arquitetura que permite maior eficiência de cálculo, porém, esta arquitetura única pode resultar em maior complexidade de treinamento [137].

### 6.2

#### Método

Em contraste com os capítulos anteriores, onde os modelos foram treinados do zero, nessa parte foi utilizada a técnica de *transfer learning* para aproveitar o conhecimento prévio adquirido por modelos treinados em grandes conjuntos de dados de imagens. O objetivo principal foi explorar a eficácia de TL na tarefa de regressão, onde os modelos pré-treinados foram ajustados para prever valores numéricos a partir de imagens.

Da mesma forma como já foi feito anteriormente, os dados experimentais foram reservados apenas para teste e validação, enquanto para o treinamento, foi empregado um conjunto de dados sintéticos composto por 924 casos. Além disso, para a etapa de ajuste fino, empregamos um subconjunto de 462 casos. O ajuste fino, quando realizado, foi aplicado aos modelos para adaptá-lo aos dados específicos da tarefa de interesse.

*Fine-tuning*, ou ajuste fino, refere-se a uma técnica na qual um modelo de aprendizado de máquina pré-treinado é ajustado em um conjunto de dados específico relacionado ao problema em questão. Em outras palavras, o modelo é inicializado com pesos pré-treinados de uma tarefa relacionada e, em seguida, esses pesos são ajustados para a tarefa específica.

Em geral, durante o *fine-tuning*, é comum ajustar apenas as camadas finais do modelo, especialmente a camada de saída, para que sejam mais adequadas à tarefa específica em questão. Isso pode ser feito para todos os

modelos de TF mencionados, com o intuito de permitir um ajuste mais preciso aos dados de destino

O método foi implementado utilizando a biblioteca `TensorFlow` em Python, aproveitando arquiteturas de modelos pré-treinados, como VGG16, ResNet50, MobileNet, MobileNetV2 e Xception.

Os modelos de aprendizado profundo, como as redes neurais utilizadas neste trabalho, são mais adequados para lidar com dados estruturados, como imagens, que têm uma forma e padrões espaciais bem definidos. Dados sequenciais não têm uma estrutura espacial inerente [138]. Por isso, muitos modelos existentes, como VGG, Xception, MobileNet e ResNet, foram projetados para aceitar imagens de tamanho fixo e com um número específico de canais de cores. Além disso, o objetivo é aproveitar o aprendizado prévio realizado em um grande conjunto de dados de imagens, o ImageNet. Portanto, para utilizá-los com TF para a tarefa requerida, foi necessário pré-processar os dados e transformá-los em imagem.

### 6.2.1

#### Redimensionamento em Imagem

Muitos tipos de dados podem ser transformados em imagens usando técnicas de visualização adequadas. Além de sinais de ondas, dados como texto, áudio e dados tabulares podem ser convertidos em imagens para análise e processamento por modelos de aprendizado profundo. A transformação de um sinal de onda amostrado ao longo do tempo em uma imagem é um processo de representação visual que visa extrair informações úteis e padrões do sinal para análise ou processamento adicional.

Um sinal de onda é uma representação matemática de uma forma de onda que pode variar ao longo do tempo. Esse sinal é amostrado em intervalos regulares, capturando os valores da amplitude da onda em momentos discretos no tempo [139]. Após a amostragem do sinal, para a conversão em imagem, cada valor amostrado do sinal ao longo do tempo pode ser tratado como um ponto em uma grade bidimensional, onde uma dimensão representa o tempo e a outra dimensão representa a amplitude do sinal [65, 140]. A intensidade de cada ponto na grade pode ser codificada usando uma escala de cores, como tons de cinza ou padrão RGB.

Dependendo do objetivo da transformação, podem ser aplicadas técnicas adicionais de pré-processamento, como interpolação, normalização ou redimensionamento, para ajustar a representação visual resultante de acordo com os requisitos específicos da análise ou modelo de aprendizado profundo. Por exemplo, um sinal de onda com 1000 pontos de amostra, pode ser represen-

tado como uma imagem de 1000 pixels de largura (representando o tempo) por 100 pixels de altura (representando a amplitude). Cada pixel na imagem teria uma intensidade correspondente ao valor do sinal naquele ponto específico no tempo.

Neste trabalho, os modelos de aprendizagem por transferência, foram configurados para receber imagens quadradas de 128 por 128 pixels, por isso foi necessário recorrer a um redimensionamento das imagens, embora existam outras possibilidades, como recorte, combinação de múltiplas janelas ou preenchimento para adaptar o sinal de onda aos requisitos necessários [64, 141].

Os dados utilizados consistiam em sinais de ondas e foram carregados em duas versões diferentes: os sintéticos, representadas por conjuntos de 462 e 924 séries e os dados experimentais que são os sinais de ondas reais. Foram usados esses conjuntos, pois conforme visto no Capítulo 3 para esse tipo de dado, ao utilizar conjuntos maiores, a melhoria dos modelos de ML torna-se marginal, indicando um equilíbrio entre melhores resultados e custos computacionais.

Cada série temporal foi transformada em uma imagem em escala de cinza, que posteriormente é convertida em uma representação no padrão RGB, seguindo os seguintes passos:

- **Interpolação Linear:** Inicialmente, a série temporal é submetida à função `np.interp()` do NumPy para realizar uma interpolação linear. Isso é feito para dobrar o número de amostras e obter uma representação mais detalhada da variação do sinal.
- **Redimensionamento:** Após a interpolação, a série temporal é redimensionada usando a função `reshape()` do NumPy para corresponder às dimensões desejadas da imagem. Para este trabalho, as imagens resultantes têm tamanho de 128x128 pixels.
- **Normalização:** Os valores resultantes da interpolação são normalizados para o intervalo  $[0, 1]$  usando operações aritméticas e funções do NumPy (`np.min()` e `np.max()`). Isso é feito para garantir que os valores dos pixels estejam dentro do intervalo esperado para uma imagem RGB.
- **Empilhamento de Canais:** Por fim, os valores normalizados da série temporal são duplicados e empilhados em cada um dos três canais de cor (vermelho, verde e azul) da imagem RGB. Isso é feito usando a função `np.stack()` do NumPy, que empilha os *arrays* ao longo de um novo eixo. Os mesmos valores de intensidade de cinza são atribuídos a cada canal, resultando em uma imagem em tons de cinza representada em formato RGB.

### 6.3

#### Discussão e resultados

Os primeiros resultados mostrados aqui se referem ao processamento dos dados. As Figuras 6.1, 6.2 e 6.3 mostram o exemplo de como os sinais são remodelados em imagens para alimentar as redes neurais utilizadas.

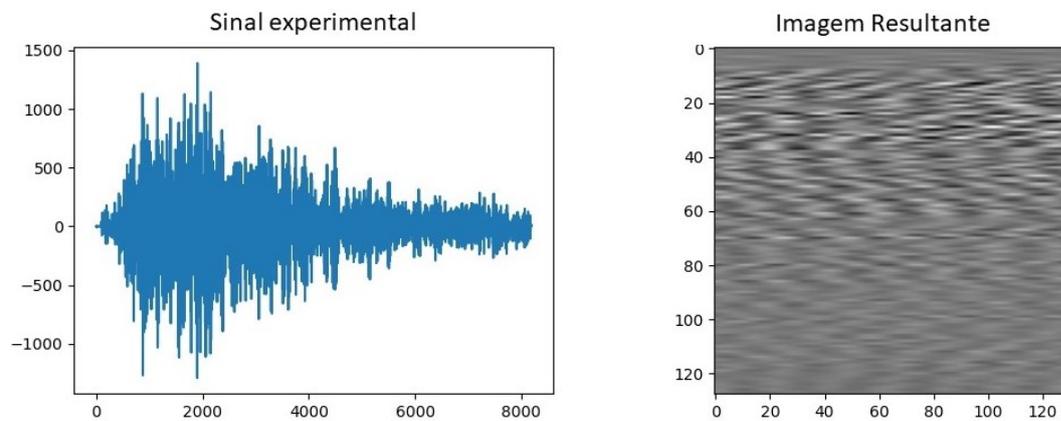


Figura 6.1: Sinal experimental e sua imagem correspondente.

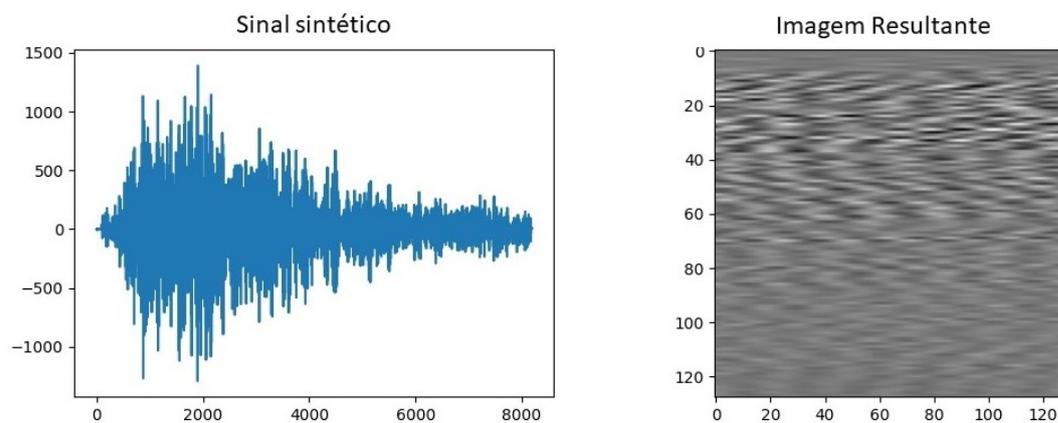


Figura 6.2: Sinal sintético e sua respectiva forma remodelada em imagem.

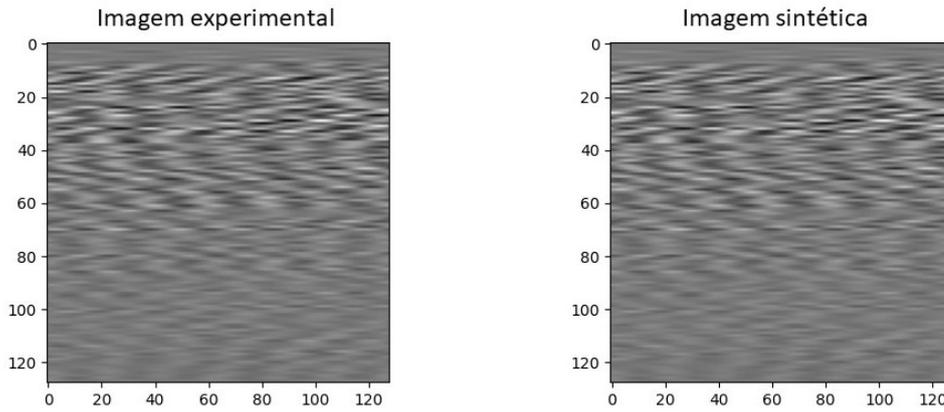


Figura 6.3: Sinais de onda remodelados.

Além disso, para avaliar a qualidade das imagens sintéticas em comparação com as experimentais, foram calculadas métricas de similaridade, como o Índice de Similaridade Estrutural (SSIM) e a Relação Sinal-Ruído de Pico (PSNR). Essas métricas foram calculadas para cada par de imagens (sintética e experimental).

O SSIM é uma medida que avalia a qualidade perceptual das imagens, levando em consideração a semelhança estrutural, o contraste e a luminância. Ele varia de -1 a 1, onde 1 indica uma correspondência perfeita entre as imagens. Por outro lado, o PSNR é uma medida que avalia a qualidade das imagens considerando a quantidade de ruído presente na imagem, em relação ao sinal útil. Quanto maior o valor do PSNR, melhor é a qualidade da imagem, sendo que valores mais altos indicam uma menor quantidade de ruído. Essas métricas foram calculadas para cada par de imagens (sintética e experimental), para verificar a correspondência visual entre elas.

Para a primeira imagem de exemplo, foram calculados os seguintes valores:  $SSIM = 1.0000$  e  $PSNR = \text{inf}$ . E as estatísticas resumidas para todas as amostras foram:

- Média SSIM: 0.9734
- Desvio Padrão SSIM: 0.0188
- Média PSNR: 34.5020
- Desvio Padrão PSNR: 4.6694

Os resultados forneceram uma visão que aponta para a boa qualidade da correspondência entre as imagens sintéticas e experimentais.

Com relação aos modelos de TF, os resultados das análises serão apresentados em termos de *Mean Squared Error* (MSE) uma métrica comum para avaliar a performance de modelos de regressão. O MSE é calculado como a

média dos quadrados das diferenças entre os valores preditos e os valores reais, [70, 87]. A fórmula para o cálculo dessa métrica é dada por:

Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (6-1)$$

onde  $y_i$  representa o valor real do  $i$ -ésimo exemplo,  $\hat{y}_i$  representa o valor predito pelo modelo e  $n$  é o número total de exemplos.

Em termos simples, o MSE é uma medida de quão bem o modelo se ajusta aos dados. Quanto menor o valor do MSE, melhor é a qualidade do modelo, pois indica que os valores previstos estão mais próximos dos valores reais. É possível ver na Figura 6.4 e na Tabela 6.1 que o modelo CNN teve um MSE maior do que os modelos MobileNet, MobileNetV2 e Xception. Isso sugere que, em geral, os modelos com transferência de aprendizado tiveram um desempenho melhor em relação ao modelo CNN.

É importante ressaltar que a arquitetura do modelo CNN utilizado neste estudo já foi previamente empregada em um trabalho publicado por Villares Holguin *et. al.* (Improved Stress Estimation with Machine Learning and Ultrasonic Guided Waves) [43]. Nesse estudo que está sendo citado, o modelo CNN foi treinado exclusivamente com os dados experimentais e foi realizada uma comparação com outros modelos de regressão, conforme pode ser visto na Tabela 6.2, que mostra informações integralmente reproduzidas da publicação [43].

Verifica-se que resultados obtidos nesta tese apresentam uma consistência com os resultados anteriores, uma vez que os valores de MSE estão na mesma ordem de grandeza. Enquanto aqui o MSE para o modelo CNN foi de 10.46, no trabalho citado foi relatado um valor de 11.32. Reiterando que no estudo anterior, o modelo CNN foi treinado e testado com os dados experimentais, enquanto neste trabalho foi utilizada uma combinação de dados experimentais e sintéticos, ou seja, uma base de dados maior, portanto, a diferença pode estar relacionada a esse fato.

Nota-se também que modelos ResNet e VGG apresentam valores muito maiores que os demais modelos. A escala poderia ser uma justificativa, já que as vezes o MSE pode ser difícil de interpretar diretamente, especialmente se os valores dos dados forem muito diferentes em escala. Mas estão sendo utilizados os mesmos dados para todos os modelos.

É possível que o problema esteja relacionado a complexidade do modelo. Modelos mais complexos podem ser mais propensos ao superajuste (*overfitting*) [142], o que pode levar a valores de MSE mais altos nos dados de teste, resultando em valores de MSE mais altos.

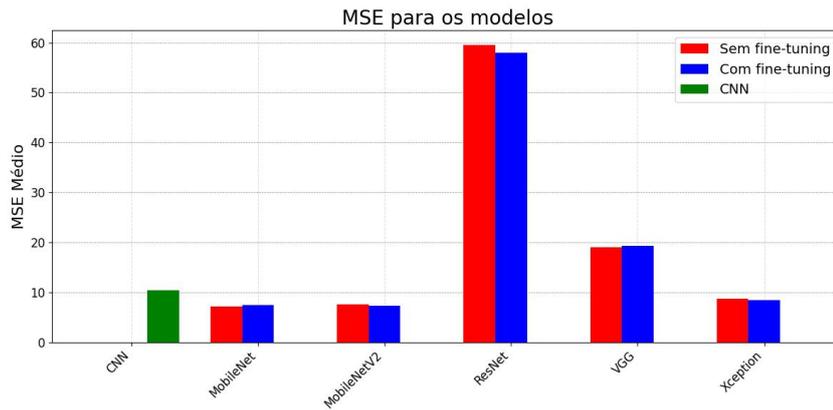


Figura 6.4: Comparação entre os modelos em termos de MSE.

modelo	MSE	Sem FT	Com FT
CNN	10.463132	-	-
MobileNet	-	7.22733	7.50732
MobileNetV2	-	7.60537	7.33395
ResNet	-	59.4543	57.8887
VGG	-	18.9757	19.231
Xception	-	8.69653	8.49203

Tabela 6.1: Valores de MSE para os diferentes modelos, onde FT corresponde a abreviatua de *fine-tuning*.

Nas tabelas, a coluna “Sem FT” se refere aos resultados dos modelos sem o ajuste fino, ou seja, os modelos baseados apenas no treinamento inicial com os dados do conjunto principal. “Com FT” representa os resultados dos modelos após o ajuste fino utilizando um conjunto adicional de dados. Podemos comparar os resultados de “Sem FT” com “Com FT” para avaliar o impacto do fine-tuning nos modelos. Se os resultados melhorarem após o fine-tuning, isso sugere que a técnica foi benéfica para o desempenho dos modelos.

Sem fine-tuning, os modelos MobileNet, MobileNetV2 e Xception têm MSE mais baixo em comparação com ResNet e VGG, sugerindo um desempenho potencialmente melhor desses modelos. Não parece haver uma diferença significativa nos valores de MSE com e sem *fine-tuning* para os modelos MobileNet, MobileNetV2 e Xception. Isso indica que o *fine-tuning* não teve um impacto substancial nos resultados de MSE para esses modelos.

O modelo ResNet tem um MSE mais alto do que o VGG sem *fine-tuning*, mas essa diferença diminuiu com o *fine-tuning*. Isso pode sugerir que o *fine-tuning* teve um efeito positivo na redução do MSE para o modelo ResNet. O modelo VGG mostrou uma ligeira melhoria no MSE com o *fine-tuning*, mas essa melhoria foi mínima.

Pelo ponto de vista das métricas de avaliação consideradas, os modelos

MobileNet e MobileNetV2 são os que possuem o melhor desempenho. No entanto, o processamento deles é mais demorado. Enquanto o modelo CNN tem um processamento de aproximadamente 20 minutos, o modelo MobileNet demorou 1 hora e 20 minutos e o modelo MobileNetV2 levou 1 hora e 50 minutos. Ressaltando que foi utilizado um computador de 256 GB de RAM com dois processadores de 2.30GHz.

Tanto ResNet quanto VGG são modelos mais profundos e complexos em comparação com os outros modelos mencionados. Isso significa que eles podem ser mais suscetíveis ao sobreajuste, especialmente se o conjunto de dados de treinamento não for grande o suficiente ou não estiver bem representado [64, 70]. O sobreajuste pode levar a uma pior generalização do modelo. Enquanto modelos como MobileNet e Xception são projetados para facilitar a transferência de aprendizado [143–145], ResNet e VGG podem exigir mais ajustes para se adaptarem aos novos dados.

Modelo	MSE	Std	Min.	1st	Median	3rd	Max.	% MSE
SVR	10.33	57.77	0.00	0.11	0.55	2.32	922.72	
CNN	<b>11.32</b>	62.14	0.00	0.22	1.05	3.46	908.05	9.56
RFR	11.72	61.79	0.00	0.04	0.22	1.12	1103.29	13.47
KNN	11.94	67.05	0.00	0.04	0.21	1.00	1079.43	15.56
LRG	12.62	61.96	0.00	0.34	1.62	5.74	902.45	22.13
DTR	18.01	144.68	0.00	0.02	0.12	0.43	8988.69	74.32

Tabela 6.2: As informações desta tabela foram integralmente reproduzidas do trabalho citado, mostrando as estatísticas de erro quadrático médio para diferentes modelos testados pelos autores. O valor de MSE do modelo CNN está destacado em negrito.

## 6.4

### Considerações finais desde capítulo

As análises realizadas incluíram as métricas de similaridade SSIM e a PSNR para comparar a qualidade das imagens sintéticas com as experimentais. Os resultados indicaram uma boa correspondência visual entre essas imagens, fornecendo uma validação adicional para a utilização dos dados sintéticos na análise.

A técnica de TL foi explorada como uma estratégia para aproveitar o conhecimento prévio de modelos treinados em grandes conjuntos de dados de imagens. Essa abordagem se mostrou vantajosa em termos de precisão, principalmente se compararmos o modelo MobileNet com a CNN. Porém, em termos de custo computacional, a CNN desempenha melhor. Os dados sintéticos obtidos por meio de interpolação linear continuaram mostrando que funcionam bem para treinar os modelos.

## 7

### Conclusão

O presente trabalho abordou a avaliação estrutural por meio de ondas ultrassônicas guiadas, integrando métodos semi-analíticos e técnicas de aprendizado de máquina (ML) para oferecer uma abordagem abrangente e inovadora. A utilização do Semi-Analytical Finite Element Method (SAFE) permitiu a análise eficiente da propagação de ondas elásticas em guias cilíndricos, otimizando recursos computacionais e garantindo resultados precisos. Paralelamente, a implementação de modelos de ML, como Random Forest Regression (RFR) e Gradient Boosting Regression (GBR), proporcionou estimativas robustas da tensão em placas sólidas.

Os resultados obtidos destacam diversas contribuições significativas para o campo da avaliação estrutural. Primeiramente, a abordagem de interpolação linear para geração de dados sintéticos emergiu como uma solução eficiente, reduzindo a dependência de experimentos dispendiosos e simulações numéricas para expansão da base de dados. Essa estratégia não apenas aumentou a precisão dos modelos de ML, mas também mitigou os custos computacionais associados.

A eficácia dos modelos de ML, especialmente RFR e GBR, foi evidenciada em conjuntos de dados maiores, alcançando bons níveis de precisão. Para conjuntos menores, modelos como Linear Regression (LRG) e K-Nearest Neighbors (KNN) demonstraram superioridade, sublinhando a flexibilidade e adaptabilidade do *pipeline* proposto.

O método SAFE, por sua vez, provou ser uma ferramenta vantajosa para a análise de guias de onda cilíndricos, proporcionando uma alternativa eficiente e precisa em comparação com abordagens numéricas convencionais. Além disso, a combinação de ML com os resultados do SAFE revelou-se promissora, oferecendo novas perspectivas para análises futuras.

A análise por meio de técnicas de transfer learning trouxe uma nova dimensão à pesquisa. Os resultados obtidos mostraram que modelos como MobileNet e MobileNetV2, com ou sem fine-tuning, superaram outros modelos mais complexos em termos de Mean Squared Error (MSE) e Mean Absolute Error (MAE). Essa abordagem representa uma extensão significativa das capacidades analíticas do projeto, demonstrando a aplicabilidade de métodos

de ML em contextos de avaliação estrutural.

## 7.1

### **Implicações e Futuras Direções**

Os resultados desta pesquisa são significativos para a área de avaliação estrutural e métodos não destrutivos. A abordagem eficiente para geração de dados sintéticos pode ser amplamente adotada, especialmente em contextos onde a disponibilidade de dados experimentais é limitada. A aplicação bem-sucedida no aprendizado de máquina denota seu potencial para otimizar a precisão das análises, ao mesmo tempo que reduz os custos computacionais.

Como direções futuras, sugerimos explorar a integração de métodos semi-analíticos e ML em contextos mais específicos, como a análise de juntas adesivas em estruturas complexas, por exemplo. Além disso, a expansão para técnicas de aprendizado profundo pode aprimorar ainda mais a capacidade de modelagem, especialmente em cenários com grande dimensionalidade. Adicionalmente, a investigação de estratégias para aprimorar a interpretabilidade dos modelos de ML pode aprofundar o entendimento sobre os processos.

## 7.2

### **Considerações Finais**

Em conclusão, a presente pesquisa contribui para o campo da avaliação estrutural, apresentando uma abordagem integrativa. Ao combinar um método semi-analítico com ML e expandir a base de dados disponível para além de dados experimentais, foi possível alcançar resultados animadores que não apenas otimizam os recursos computacionais, mas também elevam a precisão das análises.

## Referências bibliográficas

- [1] THE ECONOMIST. **Tensions will linger over a chinese balloon downed by america**, Feb. 2023.
- [2] THE WASHINGTON POST. **Chinese spy balloon flying over u.s. ‘right now,’ pentagon says**, 2024.
- [3] G1. **Balão chinês misterioso que sobrevoa eua é equipamento meteorológico, diz pequim**, 2024.
- [4] WEINER, T.. **Legacy of ashes: The history of the CIA**. Vintage, 2008.
- [5] KEAN, S.. **O último suspiro de César: A história épica do ar à nossa volta**. Editora Schwarcz-Companhia das Letras, 2019.
- [6] BARRY, R. G.; CHORLEY, R. J.. **Atmosfera, tempo e clima**. Bookman Editora, 2009.
- [7] NERY, J. T.. **Glossário de termos técnicos em Meteorologia e Climatologia**. Paco Editorial, 2013.
- [8] ZILIO, S. C.. **Óptica moderna**. São Carlos-SP: Instituto de Física de São Carlos, 93:217–233, 2009.
- [9] BORGES, A. N.; RODRIGUES, C. G.. **Introdução à física acústica**. São Paulo: Editora Livraria Física, 2017.
- [10] HELLIER, C.; DRURY, J. ; SHAKINOVSKY, M.. **Handbook of Non-destructive Evaluation**. McGraw-Hill New York, 2003.
- [11] MIAN, M. A.. **Petroleum Engineering Handbook for the Practicing Engineer**, volume 1. PennWell Books, 1992.
- [12] FREIRE, J. L. D. F.. **Engenharia de dutos**. ABCM–Associação Brasileira de Engenharia e Ciências Mecânicas. Rio de Janeiro, p. 15–1, 2009.
- [13] CUNHA, D. J. S.; FERRAZ, C.; NETTO, T. A.; ROSA, D. G. G. ; FREIRE, J. L. F.. **Hydrostatic collapse tests of full-scale pipeline specimens with thickness metal loss**. International Pipeline Conference - IPC2020, 2(13), 2020.

- [14] ROSA, D. G. G.; CUNHA, D. J. S.; DINIZ, J. C.; FERRAZ, C.; NETTO, T. A.; VIEIRA, R. D. ; FREIRE, J. L. D. F.. **Previsão de pressão de colapso em dutos sob pressão externa**. Conferencia Internacional sobre Evaluación de Integridad y Extensión de Vida de Equipos Industriales - CONAENDI-IEV, 1(22), 2020.
- [15] XIN, J.; LI, R.; CHEN, J.; LU, R.-K.; LIU, C.; SU, Z.; HE, R. ; ZHU, H.. **A crack characterization model for subsea pipeline based on spatial magnetic signals features**. Ocean Engineering, 274:114112, 2023.
- [16] ZHANG, Y.; TAN, T.; XIAO, Z.; ZHANG, W. ; ARIFFIN, M.. **Failure assessment on offshore girth welded pipelines due to corrosion defects**. Fatigue & Fracture of Engineering Materials & Structures, 39(4):453–466, 2016.
- [17] CHEN, B.-Q.; ZHANG, X. ; SOARES, C. G.. **The effect of general and localized corrosions on the collapse pressure of subsea pipelines**. Ocean Engineering, 247:110719, 2022.
- [18] ALOBAIDI, W. M.; ALKUAM, E. A.; AL-RIZZO, H. M.; SANDGREN, E. ; OTHERS. **Applications of ultrasonic techniques in oil and gas pipeline industries: A review**. American Journal of Operations Research, 5(04):274, 2015.
- [19] CARVALHO, A.; REBELLO, J.; SOUZA, M.; SAGRILLO, L. ; SOARES, S.. **Reliability of non-destructive test techniques in the inspection of pipelines used in the oil industry**. International journal of pressure vessels and piping, 85(11):745–751, 2008.
- [20] PAN, E.; ROGERS, J.; DATTA, S. ; SHAH, A.. **Mode selection of guided waves for ultrasonic inspection of gas pipelines with thick coating**. Mechanics of Materials, 31(3):165–174, 1999.
- [21] MEGSON, T. H. G.. **Introduction to aircraft structural analysis**. Butterworth-Heinemann, 2013.
- [22] JAWAD, M. H.; FARR, J. R.. **Structural analysis and design of process equipment**. John Wiley & Sons, 2018.
- [23] TIMOSHENKO, S. P.; GERE, J. M.. **Theory of elastic stability**. Courier Corporation, 2009.

- [24] OTEGUI, J. L.; OTEGUI, J. L.. **Failure mitigation and extension of service life**. *Failure Analysis: Fundamentals and Applications in Mechanical Components*, p. 251–281, 2014.
- [25] BENASCIUTTI, D.; WHITTAKER, M. T. ; DIRLIK, T.. **Fracture, fatigue, and structural integrity of metallic materials and components undergoing random or variable amplitude loadings**, 2022.
- [26] RAMBERG, W.; OSGOOD, W. R.. **Description of stress-strain curves by three parameters**. Technical report, 1943.
- [27] PARK, S.-H.; YI, K.; LIU, P.; CHOI, G.; JHANG, K.-Y. ; SOHN, H.. **In situ and layer-by-layer grain size estimation in additively manufactured metal components using femtosecond laser ultrasonics**. *Journal of Laser Applications*, 35(2), 2023.
- [28] YANG, C.; KIM, Y.; RYU, S. ; GU, G. X.. **Prediction of composite microstructure stress-strain curves using convolutional neural networks**. *Materials & Design*, 189:108509, 2020.
- [29] YAMANAKA, A.; KAMIJYO, R.; KOENUMA, K.; WATANABE, I. ; KUWABARA, T.. **Deep neural network approach to estimate biaxial stress-strain curves of sheet metals**. *Materials & Design*, 195:108970, 2020.
- [30] PARK, S.-H.; CHUNG, J.; YI, K.; SOHN, H. ; JHANG, K.-Y.. **Full-range stress-strain curve estimation of aluminum alloys using machine learning-aided ultrasound**. *Ultrasonics*, p. 107146, 2023.
- [31] PIERCE, A. D.. **Acoustics: an introduction to its physical principles and applications**. Springer, 2019.
- [32] CHEN, C.; LI, C.; RENIERS, G. ; YANG, F.. **Safety and security of oil and gas pipeline transportation: A systematic analysis of research trends and future needs using wos**. *Journal of Cleaner Production*, 279:123583, 2021.
- [33] SU, Y.; LI, J.; YU, B.; ZHAO, Y. ; YAO, J.. **Fast and accurate prediction of failure pressure of oil and gas defective pipelines using the deep learning model**. *Reliability Engineering & System Safety*, 216:108016, 2021.
- [34] KYRIAKIDES, S.; CORONA, E.. **Mechanics of offshore pipelines: volume 1 buckling and collapse, volume 1**. Elsevier, 2007.

- [35] WILCOX, P.; EVANS, M.; DILIGENT, O.; LOWE, M. ; CAWLEY, P.. **Dispersion and excitability of guided acoustic waves in isotropic beams with arbitrary cross section**. In: AIP CONFERENCE PROCEEDINGS, volume 615, p. 203–210. American Institute of Physics, 2002.
- [36] ELISHAKOFF, I.. **Mechanical vibration: where do we stand?**, volume 488. Springer Science & Business Media, 2007.
- [37] HAYASHI, T.; SONG, W.-J. ; ROSE, J. L.. **Guided wave dispersion curves for a bar with an arbitrary cross-section, a rod and rail example**. Ultrasonics, 41(3):175–183, 2003.
- [38] ROSE, J. L.. **Ultrasonic waves in solid media**, 2000.
- [39] BARTOLI, I.; MARZANI, A.; DI SCALEA, F. L. ; VIOLA, E.. **Modeling wave propagation in damped waveguides of arbitrary cross-section**. Journal of sound and vibration, 295(3-5):685–707, 2006.
- [40] LAVOR, B.; SESINI, P. ; BRAGA, A.. **A comparative study of dispersion curves in cylindrical waveguide using the semi-analytical finite element method**. DINAME 2023. Pirinópolis, 2023.
- [41] PEI, N.; BOND, L. J.. **Higher order acoustoelastic lamb wave propagation in stressed plates**. The Journal of the Acoustical Society of America, 140(5):3834–3843, 2016.
- [42] ROSE, J.. **Ultrasonic waves in solid media**. cambridge university press, cambridge, 2014.
- [43] VILLARES HOLGUIN, C.; HULTMANN AYALA, H. ; KUBRUSLY, A.. **Improved stress estimation with machine learning and ultrasonic guided waves**. Experimental Mechanics, p. 1–15, 2022.
- [44] LIU, Y.; BAO, Y.. **Review on automated condition assessment of pipelines with machine learning**. Advanced Engineering Informatics, 53:101687, 2022.
- [45] ALAMRI, A. H.. **Application of machine learning to stress corrosion cracking risk assessment**. Egyptian Journal of Petroleum, 31(4):11–21, 2022.
- [46] SOOMRO, A. A.; MOKHTAR, A. A.; KURNIA, J. C.; LASHARI, N.; LU, H. ; SAMBO, C.. **Integrity assessment of corroded oil and gas pipelines using machine learning: A systematic review**. Engineering Failure Analysis, 131:105810, 2022.

- [47] RACHMAN, A.; ZHANG, T. ; RATNAYAKE, R. C.. **Applications of machine learning in pipeline integrity management: A state-of-the-art review**. *International journal of pressure vessels and piping*, 193:104471, 2021.
- [48] SIAVASHI, J.; NAJAFI, A.; EBADI, M. ; SHARIFI, M.. **A cnn-based approach for upscaling multiphase flow in digital sandstones**. *Fuel*, 308:122047, 2022.
- [49] REZENDE BESSA FERREIRA, G.; AIDA SESINI, P.; PAULO BRASIL DE SOUZA, L.; CONCI KUBRUSLY, A. ; VICENTE HULTMANN AYALA, H.. **Corrosion-like defect severity estimation in pipelines using convolutional neural networks**. In: 2021 IEEE SYMPOSIUM SERIES ON COMPUTATIONAL INTELLIGENCE (SSCI), p. 01–07, 2021.
- [50] SPANDONIDIS, C.; THEODOROPOULOS, P.; GIANNOPOULOS, F.; GALIATSATOS, N. ; PETSAS, A.. **Evaluation of deep learning approaches for oil & gas pipeline leak detection using wireless sensor networks**. *Engineering Applications of Artificial Intelligence*, 113:104890, 2022.
- [51] WANG, B.; GUO, Y.; WANG, D.; ZHANG, Y.; HE, R. ; CHEN, J.. **Prediction model of natural gas pipeline crack evolution based on optimized dcnn-lstm**. *Mechanical Systems and Signal Processing*, 181:109557, 2022.
- [52] CHEN, L.; YAO, X.; TAN, C.; HE, W.; SU, J.; WENG, F.; CHEW, Y.; NG, N. P. H. ; MOON, S. K.. **In-situ crack and keyhole pore detection in laser directed energy deposition through acoustic signal and deep learning**. *Additive Manufacturing*, 69:103547, 2023.
- [53] DE MASI, G.; VICHI, R.; GENTILE, M.; BRUSCHI, R. ; GABETTA, G.. **A neural network predictive model of pipeline internal corrosion profile**. In: PROCEEDINGS OF THE 1ST INTERNATIONAL CONFERENCE ON SYSTEMS INFORMATICS, MODELING AND SIMULATION, p. 18–23, 2014.
- [54] KRIZHEVSKY, A.; SUTSKEVER, I. ; HINTON, G. E.. **Imagenet classification with deep convolutional neural networks**. *Communications of the ACM*, 60(6):84–90, 2017.
- [55] YASNIY, O.; PASTUKH, O.; DIDYCH, I.; YATSYSHYN, V. ; CHYKHIRA, I.. **Application of machine learning for modeling of 6061-t651**

- aluminum alloy stress- strain diagram. *Procedia Structural Integrity*, 48:183–189, 2023.
- [56] UHLIG, S.; ALKHASLI, I.; SCHUBERT, F.; TSCHÖPE, C. ; WOLFF, M.. **A review of synthetic and augmented training data for machine learning in ultrasonic non-destructive evaluation.** *Ultrasonics*, p. 107041, 2023.
- [57] ALKHALIFAH, T.; WANG, H. ; OVCHARENKO, O.. **Mlreal: Bridging the gap between training on synthetic data and real data applications in machine learning.** *Artificial Intelligence in Geosciences*, 3:101–114, 2022.
- [58] FERNANDO, B.; HABRARD, A.; SEBBAN, M. ; TUYTELAARS, T.. **Un-supervised visual domain adaptation using subspace alignment.** In: *PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION*, p. 2960–2967, 2013.
- [59] LAKSHMANAN, K.; TESSICINI, F.; GIL, A. J. ; AURICCHIO, F.. **A fault prognosis strategy for an external gear pump using machine learning algorithms and synthetic data generation methods.** *Applied Mathematical Modelling*, 123:348–372, 2023.
- [60] KAHAR, K.; DHEKEKAR, R.; BHAIYYA, M.; SRIVASTAVA, S. K.; REWATKAR, P.; BALPANDE, S. ; GOEL, S.. **Optimization of mems-based energy scavengers and output prediction with machine learning and synthetic data approach.** *Sensors and Actuators A: Physical*, 358:114429, 2023.
- [61] KONG, X.-S.; GAO, H.; JIN, Z.; ZHENG, C. ; WANG, Y.. **Predictions of the responses of stiffened plates subjected to underwater explosion based on machine learning.** *Ocean Engineering*, 283:115216, 2023.
- [62] REN, S.-F.; ZHAO, P.-F.; WANG, S.-P. ; LIU, Y.-Z.. **Damage prediction of stiffened plates subjected to underwater contact explosion using the machine learning-based method.** *Ocean Engineering*, 266:112839, 2022.
- [63] LECUN, Y.; BENGIO, Y. ; HINTON, G.. **Deep learning.** *nature*, 521(7553):436–444, 2015.
- [64] GOODFELLOW, I.; BENGIO, Y. ; COURVILLE, A.. **Deep learning.** MIT press, 2016.

- [65] BISHOP, C. M.; NASRABADI, N. M.. **Pattern recognition and machine learning**, volume 4. Springer, 2006.
- [66] YAN, F.; SONG, K.; LIU, Y.; CHEN, S. ; CHEN, J.. **Predictions and mechanism analyses of the fatigue strength of steel based on machine learning**. *Journal of Materials Science*, 55:15334–15349, 2020.
- [67] JORDAN, M. I.; MITCHELL, T. M.. **Machine learning: Trends, perspectives, and prospects**. *Science*, 349(6245):255–260, 2015.
- [68] SOMMER, D.; HAUFE, A. ; MIDDENDORF, P.. **A machine learning material model for structural adhesives in finite element analysis**. *International Journal of Adhesion and Adhesives*, 117:103160, 2022.
- [69] BERK, R. A.; OTHERS. **Statistical learning from a regression perspective**, volume 14. Springer, 2008.
- [70] HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H. ; FRIEDMAN, J. H.. **The elements of statistical learning: data mining, inference, and prediction**, volume 2. Springer, 2009.
- [71] MÜLLER, A. C.; GUIDO, S.. **Introduction to machine learning with Python: a guide for data scientists**. "O'Reilly Media, Inc.", 2016.
- [72] SAMAITIS, V.; YILMAZ, B. ; JASIUNIENE, E.. **Adhesive bond quality classification using machine learning algorithms based on ultrasonic pulse-echo immersion data**. *Journal of Sound and Vibration*, 546:117457, 2023.
- [73] HALLIDAY, D.; RESNICK, R. ; WALKER, J.. **Fundamentals of physics**. John Wiley & Sons, 2013.
- [74] TIMOSHENKO, S. P.; GOODIER, J. N.. **Theory of Elasticity**. McGraw Hill, 1951.
- [75] GERE, J. M.; TIMOSHENKO, S.. **Mechanics of materials 2e**. Brooks/Cole Engineering, 198(4), 1984.
- [76] SPENCER, A. J. M.. **Continuum mechanics**. Courier Corporation, 2004.
- [77] MIKLOWITZ, J.. **Theory of elastic waves and waveguides**. North-Holland Publishing Company, 1978.
- [78] MORSE, P. M.; FESHBACH, H.. **Methods of theoretical physics**. McGraw-Hill, 1954.

- [79] MITCHELL, T. M.. **Machine learning**, 1997.
- [80] ALPAYDIN, E.. **Introduction to machine learning**. MIT press, 2020.
- [81] SARAVANAN, R.; SUJATHA, P.. **A state of art techniques on machine learning algorithms: a perspective of supervised learning approaches in data classification**. In: 2018 SECOND INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING AND CONTROL SYSTEMS (ICICCS), p. 945–949. IEEE, 2018.
- [82] RUSSELL, S. J.; NORVIG, P.. **Artificial intelligence: a modern approach**. Pearson, 2016.
- [83] RASCHKLA, S.; MIRJALILI, V.. **Python machine learning, packt publishing**. Birmingham, United Kingdom.[Google Scholar], 2017.
- [84] CERVANTES, J.; GARCIA-LAMONT, F.; RODRÍGUEZ-MAZAHUA, L. ; LOPEZ, A.. **A comprehensive survey on support vector machine classification: Applications, challenges and trends**. *Neurocomputing*, 408:189–215, 2020.
- [85] CHAUDHURI, A.; DE, K.. **Fuzzy support vector machine for bankruptcy prediction**. *Applied Soft Computing*, 11(2):2472–2486, 2011.
- [86] WU, X.; KUMAR, V.; ROSS QUINLAN, J.; GHOSH, J.; YANG, Q.; MOTODA, H.; MCLACHLAN, G. J.; NG, A.; LIU, B.; YU, P. S. ; OTHERS. **Top 10 algorithms in data mining**. *Knowledge and information systems*, 14:1–37, 2008.
- [87] JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. ; OTHERS. **An introduction to statistical learning**, volume 112. Springer, 2013.
- [88] COVER, T.; HART, P.. **Nearest neighbor pattern classification**. *IEEE transactions on information theory*, 13(1):21–27, 1967.
- [89] ZHANG, S.. **Cost-sensitive knn classification**. *Neurocomputing*, 391:234–242, 2020.
- [90] FRIEDMAN, J. H.. **Greedy function approximation: a gradient boosting machine**. *Annals of statistics*, p. 1189–1232, 2001.
- [91] RUMELHART, D. E.; HINTON, G. E. ; WILLIAMS, R. J.. **Learning representations by back-propagating errors**. *nature*, 323(6088):533–536, 1986.

- [92] HAYKIN, S.. **Redes neurais: princípios e prática**. Bookman Editora, 2001.
- [93] HORNIK, K.; STINCHCOMBE, M. ; WHITE, H.. **Multilayer feed-forward networks are universal approximators**. *Neural networks*, 2(5):359–366, 1989.
- [94] FUNAHASHI, K.-I.. **On the approximate realization of continuous mappings by neural networks**. *Neural networks*, 2(3):183–192, 1989.
- [95] CYBENKO, G.. **Approximation by superpositions of a sigmoidal function**. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- [96] HAN, J.; PEI, J. ; TONG, H.. **Data mining: concepts and techniques**. Morgan kaufmann, 2022.
- [97] JOLLIFFE, I.. **Principal component analysis**. *Encyclopedia of statistics in behavioral science*, 2005.
- [98] TREFETHEN, L. N.; BAU, D.. **Numerical linear algebra**, volume 181. Siam, 2022.
- [99] VARMA, S.; SIMON, R.. **Bias in error estimation when using cross-validation for model selection**. *BMC bioinformatics*, 7(1):1–8, 2006.
- [100] ARLOT, S.; CELISSE, A.. **A survey of cross-validation procedures for model selection**. 2010.
- [101] KOHAVI, R.; OTHERS. **A study of cross-validation and bootstrap for accuracy estimation and model selection**. In: *IJCAI*, volume 14, p. 1137–1145. Montreal, Canada, 1995.
- [102] KUBRUSLY, A. C.; PÉREZ, N.; DE OLIVEIRA, T. F.; ADAMOWSKI, J. C.; BRAGA, A. M. ; VON DER WEID, J. P.. **Mechanical strain sensing by broadband time reversal in plates**. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 63(5):746–756, 2016.
- [103] RIZZO, P.; LANZA DI SCALEA, F.. **Effect of frequency on the acoustoelastic response of steel bars**. *Experimental Techniques*, 27:40–43, 2003.
- [104] RIZZO, P.; PALMER, M. D. ; DI SCALEA, F. L.. **Ultrasonic characterization of steel rods for health monitoring of civil structures**. In: *SMART STRUCTURES AND MATERIALS 2003: SMART SYSTEMS AND*

- NONDESTRUCTIVE EVALUATION FOR CIVIL INFRASTRUCTURES, volume 5057, p. 75–84. SPIE, 2003.
- [105] CHAKI, S.; BOURSE, G.. **Guided ultrasonic waves for non-destructive monitoring of the stress levels in prestressed steel strands.** *Ultrasonics*, 49(2):162–171, 2009.
- [106] PRESS WILLIAM, H.; TEUKOLSKY SAUL, A.; VETTERLING WILLIAM, T. ; FLANNERY BRIAN, P.. **Numerical recipes: the art of scientific computing**, 1992.
- [107] BURDEN, R. L.; FAIRES, J. D. ; BURDEN, A. M.. **Numerical analysis.** Cengage learning, 2015.
- [108] GÉRON, A.. **Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow.** "O'Reilly Media, Inc.", 2022.
- [109] BERGSTRA, J.; BARDENET, R.; BENGIO, Y. ; KÉGL, B.. **Algorithms for hyper-parameter optimization.** *Advances in neural information processing systems*, 24, 2011.
- [110] NOCEDAL, J.; WRIGHT, S. J.. **Numerical optimization.** Springer, 1999.
- [111] BOYD, S. P.; VANDENBERGHE, L.. **Convex optimization.** Cambridge university press, 2004.
- [112] BERGSTRA, J.; BENGIO, Y.. **Random search for hyper-parameter optimization.** *Journal of machine learning research*, 13(2), 2012.
- [113] BERGSTRA, J.; YAMINS, D. ; COX, D.. **Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures.** In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, p. 115–123. PMLR, 2013.
- [114] PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M. ; DUCHESNAY, E.. **Scikit-learn: Machine learning in Python.** *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [115] STODDEN, V.; GUO, P. ; MA, Z.. **Toward reproducible computational research: an empirical analysis of data and code policy adoption by journals.** *PloS one*, 8(6):e67111, 2013.

- [116] DE MAGALHÃES CORREIA, T.; DE SOUZA, L. P. B.; HIDALGO, J. A. S.; DE CASTRO RIBEIRO, M. G.; CAMERINI, I. G.; DE SOUSA, B. G.; KUBRUSLY, A. C.; AYALA, H. V. H.; BATISTA, J. H. G. ; DE ALMEIDA, R. V.. **Analytical and numerical modeling of through-tubing acoustic logging**. TEST, 2020.
- [117] BRAGA, A. M.; BARBONE, P. E. ; HERRMANN, G.. **Wave propagation in fluid-loaded laminated cylindrical shells**. 1990.
- [118] ZHA, W.; LIU, Y.; WAN, Y.; LUO, R.; LI, D.; YANG, S. ; XU, Y.. **Forecasting monthly gas field production based on the cnn-lstm model**. Energy, 260:124889, 2022.
- [119] ZHU, B.; NECHAK, L. ; BAREILLE, O.. **Kriging metamodeling approach for predicting the dispersion curves for wave propagating in complex waveguide**. Journal of Sound and Vibration, p. 117595, 2023.
- [120] LIU, F.; WANG, X.; LIU, Z.; TIAN, F.; ZHAO, Y.; PAN, G.; PENG, C.; LIU, T.; ZHAO, L.; ZHANG, K.; ZHANG, S.; LIU, X. ; ZHAO, R.. **Identification of tight sandstone reservoir lithofacies based on cnn image recognition technology: A case study of fuyu reservoir of sanzha sag in songliao basin**. Geoenergy Science and Engineering, 222:211459, 2023.
- [121] GU, J.; WANG, Z.; KUEN, J.; MA, L.; SHAHROUDY, A.; SHUAI, B.; LIU, T.; WANG, X.; WANG, G.; CAI, J. ; CHEN, T.. **Recent advances in convolutional neural networks**. Pattern Recognition, 77:354–377, 2018.
- [122] KWON, S.; PARK, G.; JANG, Y.; CHO, J.; GON CHU, M. ; MIN, B.. **Determination of oil well placement using convolutional neural network coupled with robust optimization under geological uncertainty**. Journal of Petroleum Science and Engineering, 201:108118, 2021.
- [123] BRANTSON, E. T.; ABDULKADIR, M.; AKWENSI, P. H.; OSEI, H.; APPIAH, T. F.; ASSIE, K. R. ; SAMUEL, S.. **Gas-liquid vertical pipe flow patterns convolutional neural network classification using experimental advanced wire mesh sensor images**. Journal of Natural Gas Science and Engineering, 99:104406, 2022.
- [124] RAJPUT, D. S.; MEENA, G.; ACHARYA, M. ; MOHBHEY, K. K.. **Fault prediction using fuzzy convolution neural network on iot envi-**

- ronment with heterogeneous sensing data fusion. *Measurement: Sensors*, 26:100701, 2023.
- [125] PAUL, E.; R.S., S.. **Modified convolutional neural network with pseudo-cnn for removing nonlinear noise in digital images.** *Displays*, 74:102258, 2022.
- [126] WEST, J.; VENTURA, D. ; WARNICK, S.. **Spring research presentation: A theoretical foundation for inductive transfer.** Brigham Young University, College of Physical and Mathematical Sciences, 1(08), 2007.
- [127] VASILEV, I.; SLATER, D.; SPACAGNA, G.; ROELANTS, P. ; ZOCCA, V.. **Python Deep Learning: Exploring deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow.** Packt Publishing Ltd, 2019.
- [128] GEORGE KARIMPANAL, T.; BOUFFANAIS, R.. **Self-organizing maps for storage and transfer of knowledge in reinforcement learning.** *Adaptive Behavior*, 27(2):111–126, 2019.
- [129] KRIZHEVSKY, A.; SUTSKEVER, I. ; HINTON, G. E.. **Imagenet classification with deep convolutional neural networks.** *Advances in neural information processing systems*, 25, 2012.
- [130] SIMONYAN, K.; ZISSERMAN, A.. **Very deep convolutional networks for large-scale image recognition.** arXiv preprint arXiv:1409.1556, 2014.
- [131] VISUAL GEOMETRY GROUP (VGG), U. O. O.. **Vgg very deep networks for visual recognition**, 2023.
- [132] HE, K.; ZHANG, X.; REN, S. ; SUN, J.. **Deep residual learning for image recognition.** In: *PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION*, p. 770–778, 2016.
- [133] SCHROFF, F.; KALENICHENKO, D. ; PHILBIN, J.. **Facenet: A unified embedding for face recognition and clustering.** In: *PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION*, p. 815–823, 2015.
- [134] HOWARD, A. G.; ZHU, M.; CHEN, B.; KALENICHENKO, D.; WANG, W.; WEYAND, T.; ANDREETTO, M. ; ADAM, H.. **Mobilenets: Efficient**

- convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.
- [135] MOBILENET. **Projeto mobilenet**, 2023.
- [136] SANDLER, M.; HOWARD, A.; ZHU, M.; ZHMOGINOV, A. ; CHEN, L.-C.. **Mobilenetv2: Inverted residuals and linear bottlenecks**. In: PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, p. 4510–4520, 2018.
- [137] CHOLLET, F.. **Xception: Deep learning with depthwise separable convolutions**. In: PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, p. 1251–1258, 2017.
- [138] WANG, Z.; YAN, W. ; OATES, T.. **Time series classification from scratch with deep neural networks: A strong baseline**. In: 2017 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN), p. 1578–1585. IEEE, 2017.
- [139] OPPENHEIM, A. V.. **Discrete-time signal processing**. Pearson Education India, 1999.
- [140] GONZALEZ, R. C.. **Digital image processing**. Pearson education india, 2009.
- [141] RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATY, A.; KHOSLA, A.; BERNSTEIN, M. ; OTHERS. **Imagenet large scale visual recognition challenge**. International journal of computer vision, 115:211–252, 2015.
- [142] GARETH, J.; DANIELA, W.; TREVOR, H. ; ROBERT, T.. **An introduction to statistical learning: with applications in R**. Springer, 2013.
- [143] PAN, S. J.; YANG, Q.. **A survey on transfer learning**. IEEE Transactions on knowledge and data engineering, 22(10):1345–1359, 2009.
- [144] BENGIO, Y.. **Deep learning of representations for unsupervised and transfer learning**. In: PROCEEDINGS OF ICML WORKSHOP ON UNSUPERVISED AND TRANSFER LEARNING, p. 17–36. JMLR Workshop and Conference Proceedings, 2012.

- [145] YOSINSKI, J.; CLUNE, J.; BENGIO, Y. ; LIPSON, H.. **How transferable are features in deep neural networks?** Advances in neural information processing systems, 27, 2014.