



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

Matheus Henrique Fessel

**USO DE REDES NEURAS ARTIFICIAIS PARA DETERMINAÇÃO DE
PROPRIEDADES EQUIVALENTES DE ESTRUTURAS PERIÓDICAS E PREDIÇÃO
DE RUÍDO DE CABINE DE AERONAVES**

Florianópolis

2023

Matheus Henrique Fessel

**USO DE REDES NEURAS ARTIFICIAIS PARA DETERMINAÇÃO DE
PROPRIEDADES EQUIVALENTES DE ESTRUTURAS PERIÓDICAS E PREDIÇÃO
DE RUÍDO DE CABINE DE AERONAVES**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Mecânica da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Mestre em Engenharia Mecânica.

Orientador(a): Prof. Arcanjo Lenzi, Ph.D.

Florianópolis

2023

Fessel, Matheus Henrique

Uso De Redes Neurais Artificiais Para Determinação De Propriedades Equivalentes De Estruturas Periódicas E Predição De Ruído De Cabine De Aeronaves / Matheus Henrique Fessel ; orientador, Arcanjo Lenzi, 2023.

106 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia Mecânica, Florianópolis, 2023.

Inclui referências.

1. Engenharia Mecânica. 2. Redes Neurais. 3. Estruturas Periódicas. 4. Ruido de Cabine de Aeronaves. I. Lenzi, Arcanjo. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Mecânica. III. Título.

Matheus Henrique Fessel

**USO DE REDES NEURAIAS ARTIFICIAIS PARA DETERMINAÇÃO DE
PROPRIEDADES EQUIVALENTES DE ESTRUTURAS PERIÓDICAS E PREDIÇÃO
DE RUÍDO DE CABINE DE AERONAVES**

O presente trabalho em nível de Mestrado foi avaliado e aprovado, em 15 de dezembro de 2023, pela banca examinadora composta pelos seguintes membros:

Prof. Júlio Apolinário Cordioli, Dr.
Universidade Federal de Santa Catarina

João Pedro Pinheiro Malere, Dr.
Embraer

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de Mestre em Engenharia Mecânica.

Insira neste espaço a
assinatura digital

Coordenação do Programa de Pós-Graduação

Insira neste espaço a
assinatura digital

Prof. Arcanjo Lenzi, Ph.D.
Orientador

Florianópolis, 2023.

AGRADECIMENTOS

Aos meus pais, que me conferiram todas as condições para que eu pudesse aproveitar e extrair o melhor das oportunidades que a vida oferece. A conclusão deste trabalho e desta etapa foi em grande parte do esforço deles.

À minha amada Tuana, com quem pude dividir meus dias, dos mais felizes aos mais desafiadores, com certeza a tua presença nessa jornada foi um aprendizado e uma motivação enorme. Esta dissertação tem a sua marca: resiliência.

Ao Professor Arcanjo Lenzi pela incrível competência de ser um docente excepcional e uma pessoa que inspira àqueles próximos. É uma honra termos trabalhado juntos.

Aos demais professores do LVA por contribuírem com o nobre papel de disseminar o conhecimento científico para aqueles que desejam aprender e adquirir conhecimento.

Aos colegas de longa data do LVA, com os quais tive o privilégio de participar de projetos juntos. Agradeço imensamente pelas conversas e aprendizados compartilhados durante esses anos.

A todos colegas da Embraer, que contribuíram com grande parte do conhecimento envolvido para o desenvolvimento desta dissertação e que apoiaram a conclusão desta etapa. À Embraer e suas lideranças por reconhecer a importância do ensino científico e acadêmico para desenvolvimento pessoal e profissional de seus colaboradores, disponibilizando tempo para dedicação neste trabalho.

“Pouco conhecimento faz com que as pessoas se sintam orgulhosas. Muito conhecimento, com que se sintam humildes.”

(Leonardo da Vinci)

RESUMO

Nesta dissertação de mestrado, o conceito de modelos substitutos é explorado por meio do uso de algoritmos de aprendizado de máquina que substituem análises determinísticas como, por exemplo, o método de elementos finitos. O ruído de cabine em aeronaves é um dos parâmetros mais importantes sob a ótica do conforto e da experiência do passageiro, principalmente para o setor da aviação executiva. Para cumprir com este requisito de mercado, materiais viscoelásticos são aplicados sobre a fuselagem em uma configuração de constrição, proporcionando uma redução da transmissão estrutural de energia mecânica oriunda da interação entre o escoamento de fluido e a superfície externa da aeronave. O objetivo deste trabalho é avaliar uma metodologia de modelo baseado em dados experimentais de caracterização dos materiais viscoelásticos para obtenção de propriedades vibroacústica da fuselagem como, por exemplo, Fator de Perda por Amortecimento, Densidade Modal e Fator de Perda por Radiação. Estas propriedades são obtidas primariamente pela solução de modelos de elementos finitos considerando uma estrutura periódica, a qual neste trabalho é uma partição da fuselagem com material viscoelástico aplicado, incluindo a presença de reforçadores de longarinas e cavernas. A partir da caracterização experimental dos materiais viscoelásticos e dos resultados obtidos pela solução de elementos finitos, constrói-se o conjunto de dados para treinamento e teste de uma Rede Neural Artificial, capaz de substituir o modelo físico. Os resultados das propriedades equivalentes obtidos em ambas as abordagens são comparados utilizando um modelo de SEA (*Statistical Energy Analysis*) de perda de transmissão de um painel duplo, composto por duas placas de alumínio espaçadas. As análises indicam boa concordância entre o método determinístico e o modelo substituto, sendo que a metodologia proposta é computacionalmente mais eficiente para obtenção dos parâmetros equivalentes e posterior avaliação do ruído de cabine.

Palavras-chave: Redes Neurais; Estruturas Periódicas; Ruído de Cabine de Aeronaves.

ABSTRACT

In this master's thesis, the concept of surrogate models is explored by using machine learning algorithms that substitutes deterministic analysis such as finite elements methods. Aircraft cabin noise is one of the most important features from the passenger's comfort and experience perspective, foremost for the executive aviation segment. To accomplish this market requirement, viscoelastic materials are applied over the fuselage in a constricted configuration, providing a reduction of the structural transmitted mechanical energy coming from the interaction between the fluid flow and the aircraft's externa surface. The objective of this work is to evaluate a methodology for surrogate models based on experimental data from the characterization of viscoelastic materials to further obtain vibroacoustic properties of the fuselage such as Damping Loss Factor, Modal Density and Radiation Loss Factor. These properties are primarily obtained by solving finite elements models considering a periodic structure, that for this work is a partition of fuselage with viscoelastic material, including the presence of reinforcers as stringers and frames. From the experimental characterization of viscoelastic materials and the results obtained by solving the finite elements model, the dataset is built to train and test and artificial neural network, capable of substituting the physical model. The results of the equivalent properties from both approaches are then compared within SEA (Statistical Energy Analysis) model for transmission loss, composed by two spaced aluminum plates. The analysis indicated great concordance between the deterministic method and the surrogate model, while the later shows more computational efficiency to obtain the equivalent parameters and posterior evaluation of aircraft cabin noise.

Keywords: Neural Networks; Periodic Structures; Aircraft Cabin Noise.

LISTA DE FIGURAS

Figura 1.1 - Esquema apresentando os elementos que compõem o projeto de controle de ruído típico de uma aeronave. Fonte: Elaborado pelo Autor.....	16
Figura 1.2 - Nível de pressão sonora no interior de uma aeronave para diferentes números de Mach. ■, M = 0,80, ♦, M = 0,65, ▲, M = 0,55. Fonte: (Wilby, 1995).....	17
Figura 1.3 – Evolução temporal do número de parâmetros utilizados para treinamento de alguns modelos de AI e ML. Fonte: (Bernstein, Hamerly <i>et al</i> , 2021).	19
Figura 2.1 - Representação de um sistema periódico da fuselagem em uma única direção. Fonte: Elaborado pelo Autor	22
Figura 2.2 – Construções típicas de painéis: a) painel uniforme simples; b) painel laminado; c) painel reforçado com longarinas; d) painel corrugado de composto; e) carenagem de veículo lançador treliçado; f) piso de alumínio extrudado. Fonte: (Cotoni <i>et al.</i> , 2008).....	23
Figura 2.3 - Modelo em elementos finitos de uma célula unitária de fuselagem com condições de contorno periódicas. Fonte: Elaborado pelo Autor.....	24
Figura 2.4 - Geometria de célula periódica da fuselagem. Em cinza: painel com viscoelástico; em laranja: longarina; em verde: caverna; em amarelo: conector; em azul: painel sem viscoelástico. Fonte: Elaborado pelo Autor.	28
Figura 2.5 - Fluxograma da metodologia proposta para obtenção do DLF Fonte: Adaptado (Pereira <i>et al.</i> , 2019).....	29
Figura 2.6 – a) DLF e b) módulo de elasticidade em uma viga com material viscoelástico. Fonte: (Pereira <i>et al.</i> , 2019).	29
Figura 2.7 - Fluxograma para o cálculo das propriedades equivalentes DLF, densidade modal e RLF. Fonte: Elaborado pelo Autor.	31
Figura 2.8 – a) DLF e b) módulo de elasticidade experimentais para 2 materiais distintos, considerando uma viga base de 0,8mm à -30°C.	32
Figura 2.9 - Resultado do DLF equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C.....	33
Figura 2.10 - Resultado da densidade modal equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C.....	33
Figura 2.11 - Resultado fator de perda por radiação equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C.....	34

Figura 3.1 – Relação hierárquica dos domínios de AI. Fonte: Elaborado pelo Autor.	35
Figura 3.2 – Modelo CRISP-DM. Fonte: (IBM, 2021).....	38
Figura 3.3 – Conjunto de dados quaisquer. Fonte: Elaborado pelo Autor.	39
Figura 3.4 – a) Modelo linear ajustado com base nos dados; b) Erro entre observações e modelo. Fonte: Elaborado pela Autor.	39
Figura 3.5 - a) Modelo ajustado com <i>overfitting</i> ; b) Nova predição com erro significativo; c) Modelo ajustado com <i>underfitting</i> ; d) Modelo com boa generalização. Fonte: (Portilla, 2022).....	41
Figura 3.6 – Evolução do erro ao longo do tempo para um modelo a) bom; b) ruim. Fonte: (Portilla, 2022).	42
Figura 3.7 – a) Evolução do erro ao longo do tempo para os conjuntos de treinamento e teste; b) modelo treinado com <i>overfitting</i> . Fonte: (Portilla, 2022).	42
Figura 3.8 – Exemplo de trajetória da otimização dos parâmetros w_0 e w_1 . Fonte: Adaptado de (Zhang, Lipton, Li, Smola, 2021).	44
Figura 3.9 - Exemplo de comparação da trajetória entre os algoritmos de Gradiente Descendente e Gradiente Descendente Estocástico. Fonte: Adaptado de (Zhang, Lipton, Li, Smola, 2021).....	46
Figura 3.10 – Representação gráfica de uma rede neural genérica. Fonte: Adaptado (Portilla, 2022).....	49
Figura 3.11 – Representação de um único neurônio com entradas x_1, x_2 e x_3 aplicando uma transformação resultando na saída y . Fonte: Adaptado (Portilla, 2022).....	49
Figura 3.12 – Inserção da função de ativação na rede neural. Fonte: (Da Silva, L. C., 2021).....	50
Figura 3.13 – Gráfico da função Sigmoid. Fonte: (Portilla, 2022).	50
Figura 3.14 – Gráfico da derivada da função Sigmoid. Fonte: Elaborado pelo Autor.	51
Figura 3.15 – Gráfico da função Tanh. Fonte: (Portilla, 2022).....	51
Figura 3.16 – Gráfico da derivada da função Tanh. Fonte: Elaborado pelo Autor.	52
Figura 3.17 – Gráfico da função ReLU. Fonte: (Portilla, 2022).	52
Figura 3.18 – Gráfico da derivada da função ReLU. Fonte: Elaborado pelo Autor. .	53
Figura 3.19 – Grafo que representa a função $f_{x_1, x_2} = \ln x_1 + x_1 x_2 - \sin(x_2)$. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).	55

Figura 3.20 – Grafo de uma rede neural genérica evidenciando a notação a): do peso $wjkl$; b) dos vieses bjl e funções de ativação ajl . Fonte (Nielsen, 2019).....	57
Figura 4.1 – Estrutura da rede neural para construção do modelo substituto. Fonte: Elaborado pelo Autor.....	63
Figura 4.2 – Visualização par a par do DLF Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.	66
Figura 4.3 – Histograma da distribuição dos valores de DLF Equivalente.....	66
Figura 4.4 – Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor.	67
Figura 4.5 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	68
Figura 4.6 - Modelo de regressão a) linear; b) polinomial de segundo grau para o DLF Equivalente. Fonte: Elaborado pelo Autor.....	68
Figura 4.7 - Visualização par a par da Densidade Modal Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.	70
Figura 4.8 – Histograma da distribuição dos valores de Densidade Modal Equivalente.	71
Figura 4.9- Comparação entre os valores de Densidade Modal Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor. ...	71
Figura 4.10 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	72
Figura 4.11 - Modelo de regressão a) linear; b) polinomial de segundo grau para a Densidade Modal Equivalente. Fonte: Elaborado pelo Autor.....	72
Figura 4.12 - Visualização par a par do RLF Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.	74
Figura 4.13 – Histograma da distribuição dos valores de RLF Equivalente.	74
Figura 4.14 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor.	75
Figura 4.15 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	75

Figura 4.16 - Modelo de regressão a) linear; b) polinomial de segundo grau para o RLF Equivalente. Fonte: Elaborado pelo Autor.....	76
Figura 4.17 - Representação da estratégia de transformação dos dados de RLF Equivalente. Fonte: Elaborado pelo Autor.	76
Figura 4.18 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados. Fonte: Elaborado pelo Autor.	78
Figura 4.19 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados. Fonte: Elaborado pelo Autor.....	78
Figura 4.20 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.....	79
Figura 4.21 - Comparação entre os valores da Densidade Modal Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.....	80
Figura 4.22 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.....	80
Figura 4.23 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.	81
Figura 4.24 - Comparação entre os valores de Densidade Modal Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.	81
Figura 4.25 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.	82
Figura 4.26 – Modelo de regressão linear polinomial de segundo grau para o DLF Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.....	83

Figura 4.27 - – Modelo de regressão linear polinomial de segundo grau para a Densidade Modal Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.	83
Figura 4.28 - – Modelo de regressão linear polinomial de segundo grau para o RLF Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.....	84
Figura 4.29 - Representação de uma simulação de TL em SEA. Fonte: Elaborado pelo Autor.	85
Figura 4.30 - Dispersão do DLF Equivalente do material A predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	86
Figura 4.31 - Dispersão da Densidade Modal Equivalente do material A predita pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	87
Figura 4.32 - Dispersão do RLF Equivalente do material A predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	87
Figura 4.33 – a) Dispersão da TL do material A considerando as propriedades preditas pelas redes neurais considerando funções de ativação ReLU; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.	88
Figura 4.34 – a) Dispersão da TL do material A considerando as propriedades preditas pelas redes neurais considerando três funções de ativação ReLU e uma função Sigmoid; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.	88
Figura 4.35 - Dispersão do DLF Equivalente do material B predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	90
Figura 4.36 - Dispersão da Densidade Modal Equivalente do material B predita pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	90
Figura 4.37 - Dispersão do RLF Equivalente do material B predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.	91

Figura 4.38 - a) Dispersão da TL do material B considerando as propriedades preditas pelas redes neurais considerando funções de ativação ReLU; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.91

Figura 4.39 - a) Dispersão da TL do material B considerando as propriedades preditas pelas redes neurais considerando três funções de ativação ReLU e uma função Sigmoid; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.92

LISTA DE TABELAS

Tabela 2-1 – Periodicidade, espessura e densidade de 2 materiais viscoelástico avaliados	31
Tabela 3-1 – Exemplo de diferenciação automática em Modo Adiante. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).	54
Tabela 3-2 – Exemplo de diferenciação automática em Modo Reverso. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).	56
Tabela 4-1 – Relação das condições para os ensaios de caracterização dos materiais viscoelásticos.	61
Tabela 4-2 – Densidade em [kg/m ³] de cada material para cada viga base.	61
Tabela 4-3 - Espessura total em [mm] de cada material para cada viga base.	62
Tabela 4-4 – Perioridicidade e frequência de análise para solução dos modelos de FE.	62
Tabela 4-5 - Exemplo de alguns valores de RLF Equivalente transformados.	77
Tabela 4-6 - Propriedades do material A utilizado para análise de sensibilidade.	85
Tabela 4-7 - Propriedades do material B utilizado para análise de sensibilidade.	89

LISTA DE ABREVIATURAS E SIGLAS

AI	Inteligência Artificial, do inglês <i>Artificial Intelligence</i>
ANN	Rede Neural Artificial, do inglês <i>Artificial Neural Network</i>
DL	Aprendizado Profundo, do inglês <i>Deep Learning</i>
DLF	Fator de perda por amortecimento, do inglês <i>Damping Loss Factor</i>
FE	Elementos Finitos, do inglês <i>Finite Elements</i>
FRF	Função Resposta em Frequência, do inglês <i>Frequency Response Function</i>
ML	Aprendizado de Máquina, do inglês <i>Machine Learning</i>
PINNs	Redes Neurais Informadas por Física, do inglês <i>Physics Informed Neural Networks</i>
ReLU	Unidade Linear Retificada, do inglês <i>Rectified Linear Unit</i>
RLF	Fator de perda por radiação, do inglês <i>Radiation Loss Factor</i>
SEA	Análise Estatística Energética, do inglês <i>Statistical Energy Analysis</i>
TBL	Camada limite turbulenta, do inglês <i>Turbulent Boudary Layer</i>
TL	Perda de transmissão, do inglês <i>Transmission Loss</i>

LISTA DE SÍMBOLOS

Capítulo 2

Alfabeto Latino

E	Energia total da célula periódica
F	Vetor completo de forças da célula periódica
F_{\bullet}	Vetor contendo as forças referentes às regiões \bullet da célula periódica
$G(\)$	Função de Green
$H(\)$	Função de Heaviside
$\text{Im}\{\bullet\}$	Parte imaginária de \bullet
j	Unidade imaginária igual a $\sqrt{-1}$
K	Matriz de rigidez da célula periódica
K_r	Matriz de rigidez da região r da célula periódica
K'	Matriz de rigidez simplificada da célula periódica
k_{\bullet}	Número de onda na direção \bullet
M	Matriz de massa da célula periódica
M'	Matriz de massa simplificada da célula periódica
n	Densidade modal
N	Número de modos
N_{\bullet}	Periodicidade da célula na direção \bullet
P_{diss}	Potência dissipada pela célula periódica
P_{rad}	Potência radiada pela célula periódica
q	Vetor completo contendo todos os graus de liberdade da célula periódica
q_{\bullet}	Vetor contendo os graus de liberdade referente às regiões \bullet da célula periódica
q'	Vetor simplificado dos graus de liberdade
R	Matriz de continuidade dos graus de liberdade
U_r	Energia potencial de deformação da região r da célula unitária

Alfabeto Grego

ϵ_{\bullet}	Constante de atraso de fase na direção \bullet
η	Fator de perda por amortecimento
η_n	Fator de perda por amortecimento do n -ésimo modo da célula periódica
η_r	Fator de perda por amortecimento da região r da célula periódica
η_{rad}	Fator de perda por radiação
μ_{\bullet}	Multiplicador de atraso de fase na direção \bullet
ρ	Densidade do meio
ϕ_n	Campo de deslocamento do n -ésimo modo da célula periódica
$\bar{\phi}_n$	Transformada de Fourier espacial do campo de deslocamento do n -ésimo modo da célula periódica
Ω_n	N -ésima frequência natural da célula periódica
ω	Frequência angular

Capítulo 3

Alfabeto Latino

a_j^l	Ativação da l -ésima camada do j -ésimo neurônio
b	Viés do modelo
b_j^l	Viés da l -ésima camada do j -ésimo neurônio
<i>EMA</i>	Erro médio absoluto
<i>EVS</i>	Pontuação de Variância Explicada, do inglês <i>Explained Variance Score</i>
e	Erro entre valor real e valor predito do rótulo
\bar{e}	Valor médio de e
$J(\)$	Função objetivo
K	Número de amostras de um subconjunto do conjunto de dados
M	Número de amostras do conjunto de dados
R^2	Coefficiente de correlação
<i>REQM</i>	Raiz do erro quadrático médio
$Var\{\bullet\}$	Variância de \bullet
v	Variável intermediária de uma função construída por meio de grafo
\dot{v}	Derivada de v
x	Valor dos atributos dos modelos ou parâmetro de entrada de uma

	função qualquer
\dot{x}	Derivada de x
z	Entrada da função de ativação
W	Vetor de parâmetros do modelo
w	Peso do modelo associado aos atributos
w_{jk}^l	Peso associado ao k -ésimo neurônio na camada $l - 1$ conectado ao j -ésimo neurônio na camada l
y	Valor real do rótulo
\bar{y}	Valor médio de y
\hat{y}	Valor predito para o rótulo
Alfabeto Grego	
α	Taxa de aprendizado do modelo
δ_j^l	Erro da l -ésima camada do j -ésimo neurônio
$\sigma(z)$	Função de ativação de z
Outros Símbolos	
$\frac{d(\bullet)}{d \circ}$	Derivada de \bullet em relação à \circ
$\frac{\partial(\bullet)}{\partial \circ}$	Derivada parcial de \bullet em relação à \circ
$\nabla(\bullet)$	Operador gradiente de \bullet
Capítulo 4	
Alfabeto Latino	
TL	Perda de transmissão sonora
Alfabeto Grego	
$\bar{\tau}$	Coefficiente de transmissão sonora ponderado pelo ângulo de incidência das ondas

SUMÁRIO

1	INTRODUÇÃO	16
1.1	OBJETIVOS	20
1.2	ESTRUTURA DO TRABALHO	20
2	ESTRUTURAS PERIÓDICAS EM MODELOS DE ELEMENTOS FINITOS	22
2.1	FUNDAMENTAÇÃO TEÓRICA	24
2.1.1	Cálculo das propriedades para os subsistemas SEA	26
2.2	MODELO DA CÉLULA PERIÓDICA DE UMA FUSELAGEM.....	28
2.2.1	Caracterização experimental das propriedades de materiais viscoelásticos.....	28
2.3	SOLUÇÃO DOS MODELOS CONSIDERANDO AS PROPRIEDADES VARIANDO EM FUNÇÃO DA FREQUÊNCIA	30
2.4	RESULTADOS	31
3	APRENDIZADO DE MÁQUINA.....	35
3.1	TIPOS DE APRENDIZADO DE MÁQUINAS	36
3.2	MODELO DE PROCESSO PARA APRENDIZADO DE MÁQUINA.....	37
3.2.1	Treinamento de modelos de ML.....	38
3.2.2	Algoritmos de otimização.....	42
3.2.3	Métricas de Avaliação	46
3.3	REDES NEURAS	47
3.3.1	Modelo de redes neurais artificiais	48
3.3.2	Diferenciação automática e Retropropagação	54
4	MODELO SUBSTITUTO	60
4.1	DADOS DE TREINAMENTO E TESTE E ESTRUTURA DA REDE NEURAL	60
4.1.1	Arquitetura da rede neural.....	63
4.2	REGRESSÃO DAS PROPRIEDADES EQUIVALENTES DE ESTRUTURAS PERIÓDICAS.....	64
4.2.1	Regressão do Fator de Amortecimento Equivalente	65
4.2.2	Regressão da Densidade Modal Equivalente	69
4.2.3	Regressão do Fator de Perda por Radiação Equivalente	72
4.2.4	Regressão simultânea das propriedades equivalentes.....	78

4.3	ANÁLISE DE SENSIBILIDADE EM UM MODELO DE PERDA DE TRANSMISSÃO EM SEA	84
5	CONCLUSÕES.....	93
5.1	SUGESTÕES PARA TRABALHOS FUTUROS.....	94
	REFERÊNCIAS.....	95

1 INTRODUÇÃO

O ruído de cabine em aeronaves é um aspecto fundamental no segmento aeronáutico, tendo impacto direto no conforto e experiência do passageiro, tornando o produto mais competitivo no mercado, além do fator de segurança e saúde relacionado à equipe de bordo, que pode se expor a níveis excessivos de ruído por um período significativo. Para cumprir com esses requisitos, o desenvolvimento de uma aeronave necessita de um projeto de controle de ruído eficaz, levando em consideração restrições de espaço e peso, que impactam diretamente na performance da aeronave. Usualmente, o controle de ruído é feito pelo uso de isoladores de vibração, que conectam estruturalmente a fuselagem e o painel interno da aeronave, materiais termoacústicos, como espumas e fibras de vidro, inseridos na cavidade formada entre as duas estruturas mencionadas (CAVALHEIRO, 2021; KULAKAUSKAS, 2016) e materiais viscoelásticos aplicados nas baias da fuselagem (PEREIRA *et al.*, 2019), conforme a Figura 1.1.

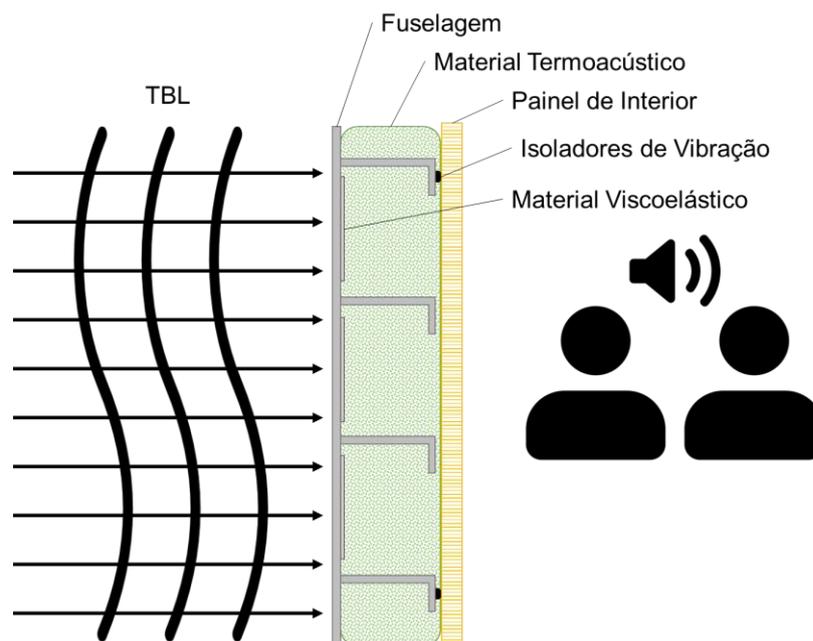


Figura 1.1 - Esquema apresentando os elementos que compõem o projeto de controle de ruído típico de uma aeronave. Fonte: Elaborado pelo Autor.

O desenvolvimento de projetos vibroacústicos, como o controle de ruído em cabines, é composto de duas partes, a primeira já apresentada anteriormente, que diz respeito às soluções e técnicas disponíveis, que podem ser aplicadas para redução dos níveis de ruído, e a segunda parte, que diz respeito à caracterização das fontes de excitação, como o ruído

transmitido estruturalmente pelos motores em baixas frequências, o ruído gerado pelas passagens de pás em aeronaves turbo hélices, que se caracteriza pelas frequências discretas das hélices e suas harmônicas, e o ruído gerado pela camada limite turbulenta (TBL, do inglês *Turbulent Boundary Layer*), induzido pela interação entre o escoamento de fluido e a superfície da fuselagem, dominando o espectro em médias e altas frequências (WILBY, 1995). A Figura 1.2 apresenta o espectro típico do ruído de cabine em aeronaves para diferentes condições de voo, variando o número de Mach.

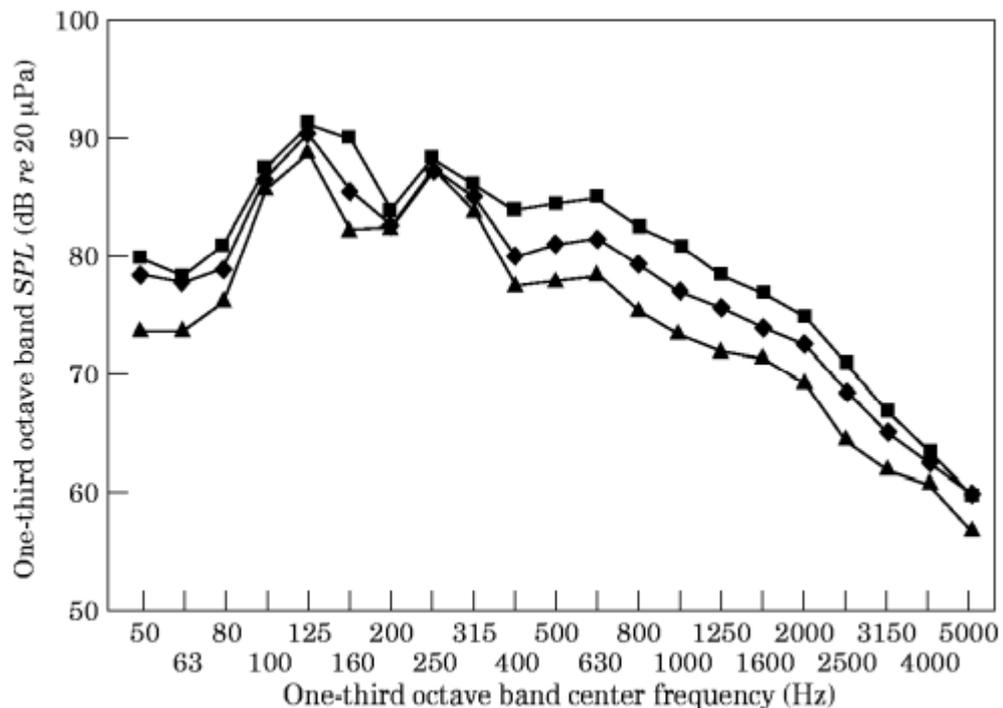


Figura 1.2 - Nível de pressão sonora no interior de uma aeronave para diferentes números de Mach. ■, M = 0,80, ◆, M = 0,65, ▲, M = 0,55. Fonte: (Wilby, 1995)

Para avaliar os níveis de pressão sonora na cabine e desenvolver o projeto de controle de ruído, a Análise Estatística Energética (SEA, do inglês *Statistical Energy Analysis*) é uma das principais metodologias utilizadas na indústria aeroespacial (CORDIOLI *et al.*, 2004). Uma das vantagens do uso deste método é seu equacionamento simples, baseado no balanço de energia e potências entre sistemas dinâmicos, que permite avaliar modelos complexos e em regiões de médias e altas frequências com custo computacional baixo. Por outro lado, existem algumas dificuldades inerentes a esta metodologia e duas delas são (LYON, DEJONG, 1995):

1) A construção do modelo a partir do particionamento em subsistemas de forma adequada, levando em consideração todas as características físicas para que as hipóteses de SEA sejam satisfeitas, por exemplo, alta densidade modal e alto fator de sobreposição modal.

2) Confiabilidade dos dados utilizados nas equações de SEA, como por exemplo, fator de perda por amortecimento (DLF, do inglês *Damping Loss Factor*), fator de perda por radiação (RLF, do inglês *Radiation Loss Factor*), densidade modal e excitações

A obtenção dos dados que caracterizam os subsistemas em SEA pode ser feita analiticamente, numericamente ou mesmo experimentalmente. É comum medir parâmetros como amortecimento e eficiência de radiação para serem inseridos em SEA e, alternativamente, utilizar modelos numéricos, como por exemplo o método de elementos finitos (FE, do inglês *Finite Element*), para calcular tais parâmetros. Porém, o uso deste método normalmente se restringe às análises em baixas frequências em função do custo computacional associado à resolução dos modelos para frequências mais altas. Diferentes abordagens podem ser adotadas para contornar essa limitação, como o uso de técnicas de redução de ordem (CAVALHEIRO, 2021) e o uso estruturas periódicas (BARDELL *et al.*, 1989; MEAD; PARTHAN, 1979; CORDIOLI; BRATI; STUMPF; LENZI; COTONI, 2010). Estes tipos de estruturas possuem a particularidade de exibir padrões ou simetrias geométricas, permitindo que sejam consideradas condições de contorno periódicas, simplificando modelos extensos em células unitárias e periódicas. Um exemplo de objeto que pode ser modelado por estruturas periódicas é a própria fuselagem, que possui repetições de uma mesma célula unitária ao longo do charuto da aeronave. Assim, reduz-se o tamanho do modelo em termos da quantidade de elementos e graus de liberdade, trazendo uma vantagem em relação ao custo computacional. Além disso, é possível calcular os parâmetros necessários inseridos em SEA (LANGLEY, 1994; SHORTER, 2004).

Uma outra abordagem para obtenção desses parâmetros é pelo uso de modelos de aprendizado de máquina (ML, do inglês *Machine Learning*), que podem ser capazes de identificar padrões, correlações e prever tendências em função dos dados coletados pelas análises de FE e utilizados para o processo de treinamento. O desenvolvimento e estudos de novas aplicações sobre Inteligência Artificial (AI, do inglês *Artificial Intelligence*) e ML tem crescido rapidamente. Estima-se que o número de publicações científicas ultrapassa 400.000 trabalhos em 2019, um aumento de 600% desde 2000, quando este número era de aproximadamente 52.000 (SAVAGE, 2020). Parte desse crescimento se dá à capacidade de processamento dos sistemas atuais, que conseguem ingerir quantidades massivas de dados e

com baixa latência, permitindo treinar modelos complexos e com um número significativo de parâmetros (BERNSTEIN, HAMERLY *et al*, 2021). A Figura 1.3 mostra algumas das principais aplicações de AI e ML em função da quantidade de parâmetros dos modelos e do ano de publicação delas.

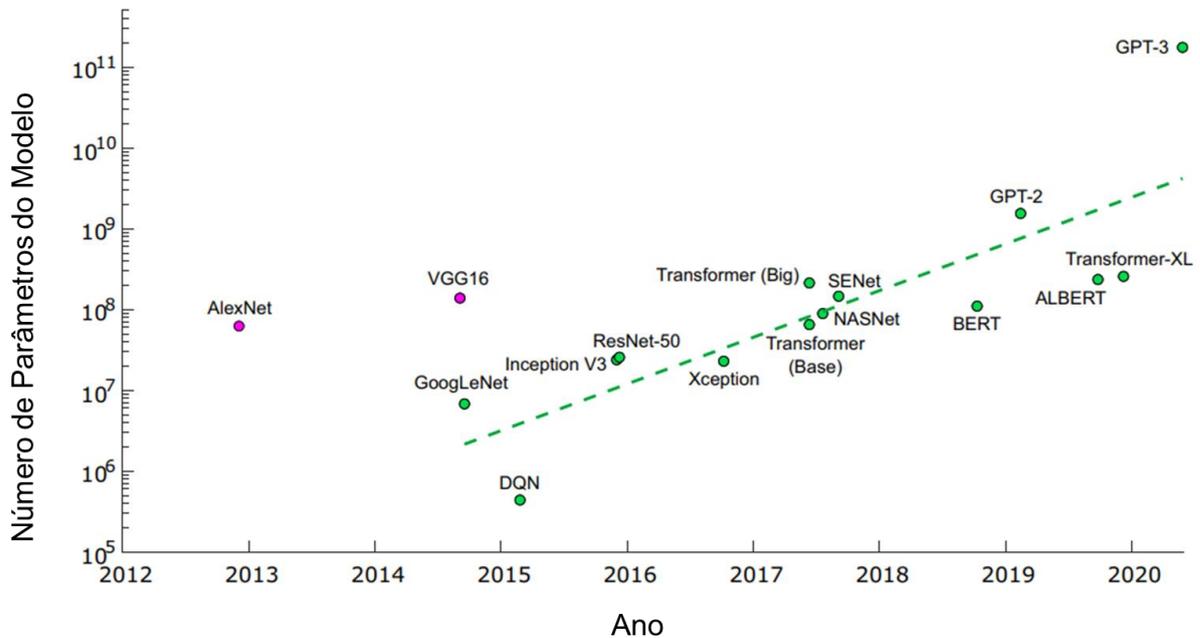


Figura 1.3 – Evolução temporal do número de parâmetros utilizados para treinamento de alguns modelos de AI e ML. Fonte: (Bernstein, Hamerly *et al*, 2021).

Dentre as aplicações de AI e ML, pode-se destacar os modelos substitutos ou metamodelos. O principal objetivo desta metodologia é substituir processos computacionalmente onerosos e repetitivos por um modelo estatístico, isto é, baseado em um conjunto de dados para treinamento e validação, que seja eficiente e consiga aproximar as soluções advindas de simulações físicas ou observações experimentais com nível de acurácia aceitável (HWANG *et al*, 2018; GUO, 2022). Neste contexto, alguns trabalhos recentes exploram a aplicação de Modelos Substitutos para problemas vibroacústicos (SACK, ÅBOM, 2020; YUCESAN *et al*, 2021; ALGUACIL *et al*, 2021) e outros exploram o conceito de Redes Neurais Informadas por Física (PINNs, do inglês *Physics Informed Neural Networks*) (RAISSI *et al*, 2019) para solução de equações diferenciais não-lineares. Portanto, foi identificada a oportunidade de avaliar o uso de ML para treinamento e validação de um Modelo Substituto das simulações em FE de estruturas periódicas para obtenção de parâmetros equivalentes utilizado em SEA para predição do ruído de cabine em aeronaves.

1.1 OBJETIVOS

Este trabalho tem como objetivo geral propor um processo baseado no conceito de modelos substitutos para determinação das propriedades equivalentes obtidas pela solução de modelos em elementos finitos com condições de contorno periódicas para a estrutura da fuselagem de uma aeronave, considerando as propriedades do material viscoelástico variando em função da frequência, especificamente DLF e o módulo de elasticidade.

Como objetivos específicos tem-se:

- Explorar e definir métodos de aprendizado de máquina para construir um modelo baseado em dados.
- Estruturar uma base de dados para treinamento e validação do modelo a partir dos resultados de caracterização dos materiais viscoelásticos e das soluções dos modelos em elementos finitos.
- Comparar e validar a abordagem proposta utilizando um modelo de previsão de ruído baseado em SEA.

1.2 ESTRUTURA DO TRABALHO

Este trabalho está organizado em 5 capítulos. No Capítulo 2 são introduzidos os fundamentos sobre estruturas periódicas aplicadas ao método de elementos finitos para problemas vibroacústico, apresentando a motivação pelo qual tal método foi utilizado no processo de previsão do ruído de cabine, pontuando algumas de suas limitações, evidenciando a necessidade de uma abordagem alternativa para obter os parâmetros equivalentes de interesse.

O Capítulo 3 apresenta conceitos gerais sobre o tema de aprendizado de máquina, explicando a diferença entre os tipos de problemas usualmente encontrados e como é o processo padrão de desenvolvimento de algoritmos de aprendizado de máquina, explicando as etapas de treinamento, teste e otimização do modelo baseado em dados e como é feita a avaliação de performance do modelo com base em métricas frequentemente utilizadas. Por fim, é apresentado o conceito de redes neurais e a formulação matemática, que compõe esse tipo de objeto.

No Capítulo 4 é introduzido o conceito de modelos substitutos e a motivação pela qual este método tem ganhado notoriedade em função dos avanços de técnicas de aprendizado

de máquina e inteligência artificial e da quantidade massiva de dados gerados em curtos períodos, que viabilizam uma abordagem estatística para diversos problemas das indústrias atuais e alavancam a tomada de decisão. Em seguida, são explicados a estrutura da base de dados utilizada para treinamento e teste da rede neural e o processo de regressão das propriedades equivalentes até a definição de um modelo para validação deste em uma simulação de perda de transmissão em SEA.

O Capítulo 5 sumariza as conclusões sobre o trabalho realizado e propõe sugestões de tópicos futuros.

2 ESTRUTURAS PERIÓDICAS EM MODELOS DE ELEMENTOS FINITOS

Uma estrutura periódica é caracterizada pela composição de um determinado número de elementos que estão conectados lado a lado e seguem um padrão de repetição espacial. Um exemplo de estrutura periódica é a fuselagem de aeronaves, que pode ser representada por um conjunto de elementos periódicos composto de um painel de alumínio uniforme reforçado com longarinas e cavernas, dispostas de forma ortogonal (MEAD, 1973), conforme a Figura 2.1.

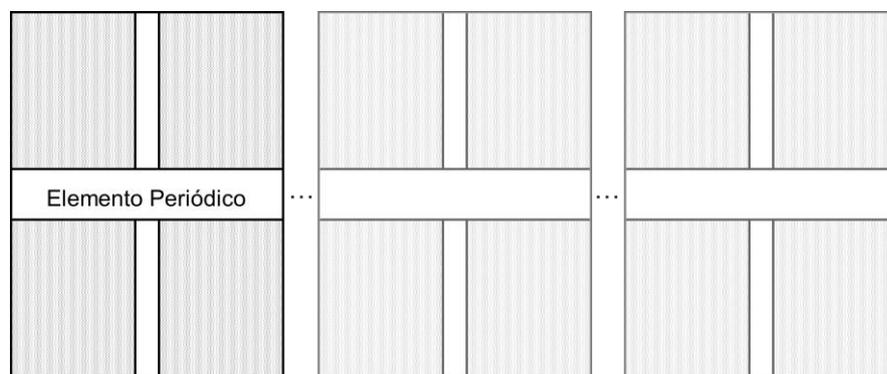


Figura 2.1 - Representação de um sistema periódico da fuselagem em uma única direção.
Fonte: Elaborado pelo Autor

Concomitantemente com o advento de aeronaves com motores a jato evidenciou-se a necessidade de projetar estruturas menos suscetíveis às excitações produzidas tanto pela TBL quanto pelos níveis elevados de ruído produzido pelos motores à época (WILBY, 1995; MEAD, 1996). Neste contexto, é fundamental compreender o mecanismo de propagação de ondas, prever o comportamento dinâmico dos elementos típicos de uma fuselagem de aeronave e para isso diversos trabalhos foram realizados, principalmente na Universidade de Southampton desde 1960 (MEAD, 1996; SILVA, 2015), para elucidar a modelagem desses componentes.

Sob a ótica de estruturas periódicas pode-se considerar estes sistemas sendo discretos ou contínuos e obter soluções analíticas para o campo de deslocamento e para o fluxo de energia, bem como soluções aproximadas por meio dos métodos de Rayleigh-Ritz e elementos finitos, comprovando a capacidade de avaliar sistemas complexos como placas e painéis curvos ambos com reforçadores (MEAD, 1973; MEAD, PARTHAN, 1979; MEAD, ZHU, BARDELL, 1988; BARDELL, MEAD, 1989).

Ainda, a possibilidade de reduzir grandes estruturas a uma célula unitária periódica flexibiliza o uso de métodos numéricos como FE para o domínio de frequências mais altas, pois é viável refinar a malha, ou seja, aumentar o número de elementos, em função do menor comprimento de onda analisado e resolver o modelo com eficiência computacional significativa quando comparada à resolução de um modelo explícito.

Embora seja possível obter expressões analíticas para calcular os parâmetros necessários para modelagem em SEA de diversos tipos de sistemas vibroacústicos, como por exemplo as estruturas mais simples mostradas nas Figuras 2.2a e 2.2b, algumas outras, como as demais apresentadas na mesma figura, possuem geometrias não convencionais que dificultam a formulação analítica para calcular os parâmetros de SEA (COTONI *et al.*, 2008). Neste contexto, o uso de estruturas periódicas em modelos de elementos finitos torna-se uma boa alternativa para calcular de forma acurada algumas propriedades importantes como DLF (CORDIOLI, BRATTI, STUMPF, LENZI, COTONI, 2010), densidade modal e fator de acoplamento.

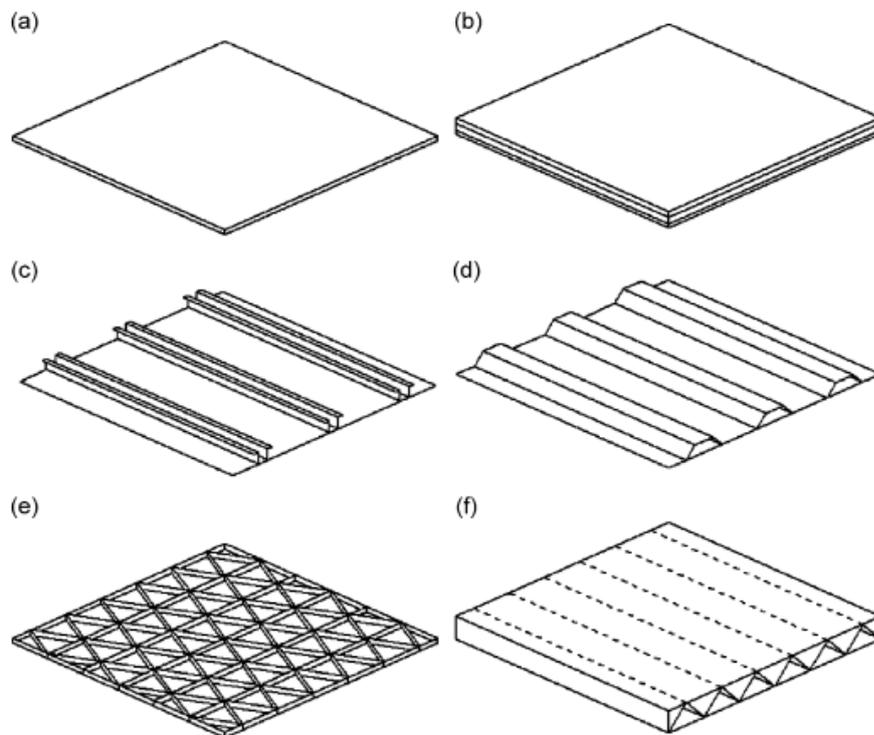


Figura 2.2 – Construções típicas de painéis: a) painel uniforme simples; b) painel laminado; c) painel reforçado com longarinas; d) painel corrugado de composto; e) carenagem de veículo lançador treliçado; f) piso de alumínio extrudado. Fonte: (Cotoni *et al.*, 2008).

2.1 FUNDAMENTAÇÃO TEÓRICA

A célula periódica que representa a fuselagem de uma aeronave é mostrada na Figura 2.3. A partir dela modela-se a estrutura em FE com graus de liberdade \mathbf{q} , particionada em regiões:

- I: Interior;
- INF, SUP, LD e LE: Inferior, Superior, Lateral Direita e Lateral Esquerda, respectivamente;
- CDS, CDI, CES, CEI: Canto Direito Superior, Canto Direito Inferior, Canto Esquerdo Superior e Canto Esquerdo Inferior, respectivamente;

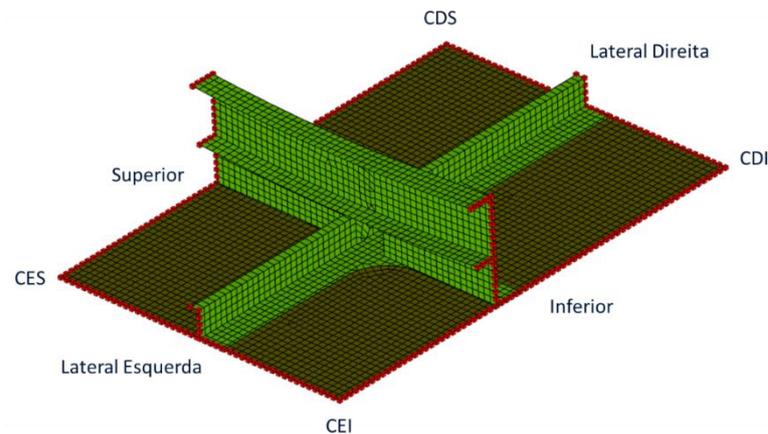


Figura 2.3 - Modelo em elementos finitos de uma célula unitária de fuselagem com condições de contorno periódicas. Fonte: Elaborado pelo Autor.

e a condição de contorno periódica é aplicada nas regiões INF, SUP, LD, LE, CDS, CDI, CES e CEI representando a conexão com células adjacentes, enquanto a região interna I não é conectada com demais células. O vetor completo \mathbf{q} é (MEAD, 1973. COTONI *et al.*, 2008):

$$\mathbf{q} = [\mathbf{q}_I \quad \mathbf{q}_{INF} \quad \mathbf{q}_{SUP} \quad \mathbf{q}_{LD} \quad \mathbf{q}_{LE} \quad \mathbf{q}_{CDS} \quad \mathbf{q}_{CDI} \quad \mathbf{q}_{CEI} \quad \mathbf{q}_{CES}]^T, \quad (2.1)$$

e para cada grau de liberdade há uma força correspondente e o vetor análogo é:

$$\mathbf{F} = [\mathbf{F}_I \quad \mathbf{F}_{INF} \quad \mathbf{F}_{SUP} \quad \mathbf{F}_{LD} \quad \mathbf{F}_{LE} \quad \mathbf{F}_{CDS} \quad \mathbf{F}_{CDI} \quad \mathbf{F}_{CEI} \quad \mathbf{F}_{CES}]^T, \quad (2.2)$$

Considerando uma excitação harmônica de frequência ω em estado estacionário, a equação de movimento para todos os graus de liberdade de um sistema não amortecido é:

$$[\mathbf{K} - \omega^2 \mathbf{M}]\{\mathbf{q}\} = \{\mathbf{F}\}, \quad (2.3)$$

na qual \mathbf{K} e \mathbf{M} são matrizes simétricas de rigidez e massa, respectivamente. Para o caso de movimento livre tem-se que as forças internas são nulas, $[\mathbf{F}_I] = \mathbf{0}$, e que o deslocamento de um nó das arestas ou cantos é igual ao deslocamento de seu correspondente oposto multiplicado por um fator e^{μ_x} ou e^{μ_y} dependendo da direção, sendo μ_x e μ_y as constantes de propagação, então:

$$\{\mathbf{q}_{LD}\} = e^{\mu_x}\{\mathbf{q}_{LE}\}, \{\mathbf{q}_{SUP}\} = e^{\mu_y}\{\mathbf{q}_{INF}\}, \quad (2.4)$$

$$\{\mathbf{q}_{CDI}\} = e^{\mu_x}\{\mathbf{q}_{CEI}\}, \{\mathbf{q}_{CDS}\} = e^{\mu_y}\{\mathbf{q}_{CEI}\}, \{\mathbf{q}_{CDS}\} = e^{\mu_x + \mu_y}\{\mathbf{q}_{CEI}\}, \quad (2.5)$$

Considerando as constantes na forma puramente imaginária $\mu_x = j\epsilon_x$ e $\mu_y = j\epsilon_y$, em que ϵ_x e ϵ_y são denominadas constantes de fase, é possível reescrever a Equação 2.1 como (COTONI *et al.*, 2008; CORDIOLI, BRATTI, STUMPF, LENZI, COTONI, 2010):

$$\{\mathbf{q}'\} = \mathbf{R}\{\mathbf{q}\}, \text{ com } \mathbf{R} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I}e^{-i\epsilon_y} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I}e^{-i\epsilon_x} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & 0 & 0 & \mathbf{I}e^{-i\epsilon_x} \\ 0 & 0 & 0 & \mathbf{I}e^{-i\epsilon_y} \\ 0 & 0 & 0 & \mathbf{I}e^{-i\epsilon_x - i\epsilon_y} \end{bmatrix}, \quad (2.6)$$

sendo $\{\mathbf{q}'\} = [\mathbf{q}_I \quad \mathbf{q}_{INF} \quad \mathbf{q}_{LE} \quad \mathbf{q}_{CEI}]^T$, pois é possível reduzir o vetor completo \mathbf{q} a partir das relações das Equações 2.4 e 2.5, e \mathbf{I} são matrizes identidade com dimensões equivalentes à quantidade de graus de liberdade nas regiões I, INF, LE e CEI. A nova equação do movimento homogênea pode ser escrita na forma:

$$[\mathbf{K}' - \omega^2 \mathbf{M}']\{\mathbf{q}'\} = 0 \quad (2.7)$$

com

$$\mathbf{K}' = \mathbf{R}^H(\epsilon_x, \epsilon_y)\mathbf{K}\mathbf{R}(\epsilon_x, \epsilon_y) \text{ e } \mathbf{M}' = \mathbf{R}^H(\epsilon_x, \epsilon_y)\mathbf{M}\mathbf{R}(\epsilon_x, \epsilon_y) \quad (2.8)$$

na qual o sobrescrito H indica a matriz transposta conjugada de \mathbf{R} . A Equação 2.7 pode ser resolvida como um problema de autovalor e autovetor fornecendo as constantes de fase, que variam de $-\pi$ a π . Os autovalores Ω_n são as frequências naturais em que há propagação pela estrutura para uma dada constante de fase e os autovetores ϕ_n representam o deslocamento correspondente.

2.1.1 Cálculo das propriedades para os subsistemas SEA

No contexto deste trabalho, tem-se o objetivo de calcular três propriedades importantes para subsistemas SEA:

1. Densidade modal $n(\omega)$
2. Fator de perda por amortecimento $\eta(\omega)$
3. Fator de perda por radiação $\eta_{rad}(\omega)$

A densidade modal pode ser calculada a partir da seguinte expressão (COTONI *et al.*, 2008):

$$n(\omega) = \frac{\partial N}{\partial \omega} \quad (2.9)$$

com (COTONI *et al.*, 2008; VAOne, 2020):

$$N(\omega) = \frac{N_x N_y}{4\pi^2} \sum_n \left(\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} H(\omega - \Omega_n(\epsilon_x, \epsilon_y)) d\epsilon_x d\epsilon_y \right) \quad (2.10)$$

na qual N_x e N_y é a quantidade de células periódicas em cada direção, H é a função degrau de Heaviside considerando $H(x < 0) = 0$ e $H(x > 0) = 1$, ω é a frequência angular e o somatório é feito para cada par de constante de fase (ϵ_x, ϵ_y) .

O DLF associado a uma determinada onda propagando na estrutura é dado por (SHORTER, 2004, COTONI *et al.*, 2008):

$$\eta_n(\epsilon_x, \epsilon_y) = P_{diss} / \omega E, \quad (2.11)$$

sendo P_{diss} a potência dissipada e E a energia total, ambas são função das constantes de fase e número de frequências naturais n . Considerando amortecimento estrutural, a potência dissipada ser expressa como:

$$P_{diss} = \sum_r 2\omega\eta_r U_r, \quad (2.12)$$

na qual η_r e U_r são o DLF e a energia de deformação, respectivamente, para uma determinada região r . A energia de deformação e energia total podem ser calculadas em termos das matrizes de massa \mathbf{M} e rigidez \mathbf{K} e do campo de deslocamento $\phi_n(\epsilon_x, \epsilon_y)$:

$$\eta_n(\epsilon_x, \epsilon_y) = \frac{\sum_r 2\eta_r \phi_n^H (\mathbf{R}^H \mathbf{K}_r \mathbf{R}) \phi_n}{\phi_n^H [\omega^2 \mathbf{M}' + \mathbf{K}'] \phi_n}, \quad (2.13)$$

em que \mathbf{K}_r é matriz de rigidez referente à região r . Então, considerando que há equipartição de energia para as ondas propagando na estrutura dentro de uma determinada banda de frequência $\Delta\omega$, o DLF da célula pode ser calculado como a média sobre todas as ondas propagando dentro desta banda de frequência (CORDIOLI, BRATTI, STUMPF, LENZI, COTONI, 2010):

$$\eta(\omega) = \langle \eta_n(\epsilon_x, \epsilon_y) \rangle_{\epsilon_x, \epsilon_y, \Omega_n \in \Delta\omega}. \quad (2.14)$$

E por fim, o RLF está relacionado com a potência radiada pela estrutura e a energia total da seguinte forma (VAOne, 2020):

$$\eta_{rad} = \frac{P_{rad}}{\omega E}, \quad (2.15)$$

e expressando em termos da frequência tem-se:

$$\eta_{rad} = \frac{1}{4\pi^2 \omega^2} \frac{1}{N_x N_y} \langle \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |\tilde{\phi}_n(k_x, k_y)|^2 \text{Im}\{G(k_x, k_y, \omega)\} dk_x dk_y \rangle_{\epsilon_x, \epsilon_y, \Omega_n \in \Delta\omega}, \quad (2.16)$$

na qual $\tilde{\phi}_n$ é a transformada de Fourier espacial do campo de deslocamento da célula periódica para cada par de constantes de fase, G é a função de Green para campo livre, $G =$

$i\rho\omega^2/(k^2 - k_x^2 - k_y^2)$, $\text{Im}\{ \}$ sua parte imaginária e k o número de onda e suas componentes k_x e k_y .

2.2 MODELO DA CÉLULA PERIÓDICA DE UMA FUSELAGEM

Neste trabalho foi considerada a célula periódica apresentada na Figura 2.4, a qual é um modelo representativo da fuselagem de uma aeronave, contendo todos os componentes reforçadores e o material viscoelástico aplicado sobre o painel.

Para todas as partes foi considerando um DLF constante de 1% e o módulo de elasticidade do alumínio, com exceção da região com material viscoelástico, que foi utilizado os valores obtidos experimentalmente (CORDIOLI, BRATTI, STUMPF, LENZI, COTONI, 2010; PEREIRA *et al.*, 2019), considerando uma estrutura equivalente composta pelo painel, material viscoelástico e camada constritora.

O modelo e as análises foram feitos com o *software* comercial VAOne, versão 2019.1.

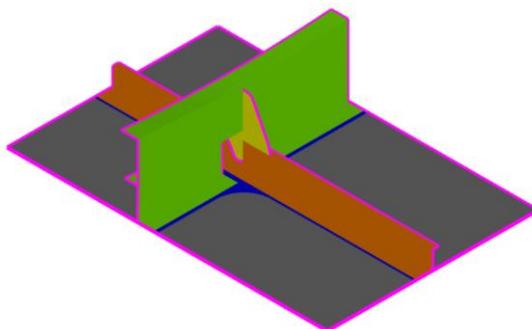


Figura 2.4 - Geometria de célula periódica da fuselagem. Em cinza: painel com viscoelástico; em laranja: longarina; em verde: caverna; em amarelo: conector; em azul: painel sem viscoelástico. Fonte: Elaborado pelo Autor.

2.2.1 Caracterização experimental das propriedades de materiais viscoelásticos

O procedimento para caracterização dos materiais viscoelásticos proposto por Pereira *et al.* (2019) visa contornar a dificuldade de extrair as propriedades de estruturas amortecidas até frequências mais altas, enquanto métodos tradicionais se restringem a frequências mais baixas. O procedimento consiste em medir a Função Resposta em Frequência (FRF, do inglês *Frequency Response Function*) de vigas com materiais viscoelásticos e reproduzir a curva de FRF com um modelo numérico em FE, que é representado por uma viga homogênea com

espessura e massa considerando todas as camadas. Primeiramente, o módulo de elasticidade é estimado comparando as frequências naturais da viga obtidos experimentalmente e numericamente e para cada modo de flexão, então para cada modo tem-se um valor de módulo de elasticidade. Considerando amortecimento estrutural na forma de rigidez complexa, é feito um ajuste de curva, minimizando o erro entre medição experimental e simulação em FE, considerando tanto o módulo de elasticidade quanto o DLF, uma vez que os modos de flexão da viga podem ser afetados em função do amortecimento. Este processo de otimização é feito separadamente para cada modo, então ao final tem-se um conjunto de pontos para o DLF e para o módulo de elasticidade, ambos em função da frequência, que são utilizados para interpolar uma curva contínua para as duas propriedades. A Figura 2.5 mostra o fluxograma do processo de otimização e a Figura 2.6 mostra o resultado obtido.

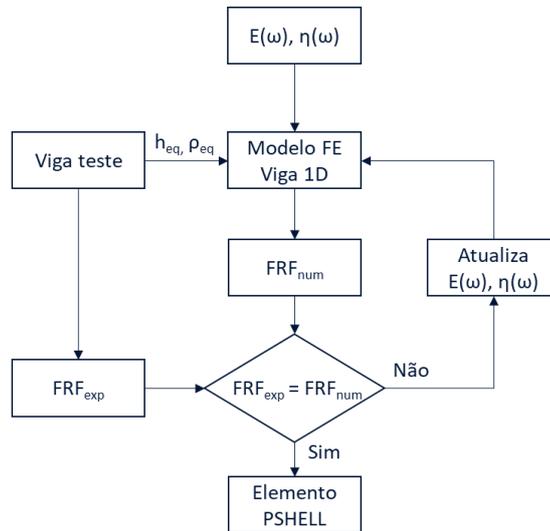


Figura 2.5 - Fluxograma da metodologia proposta para obtenção do DLF Fonte: Adaptado (Pereira *et al.*, 2019).

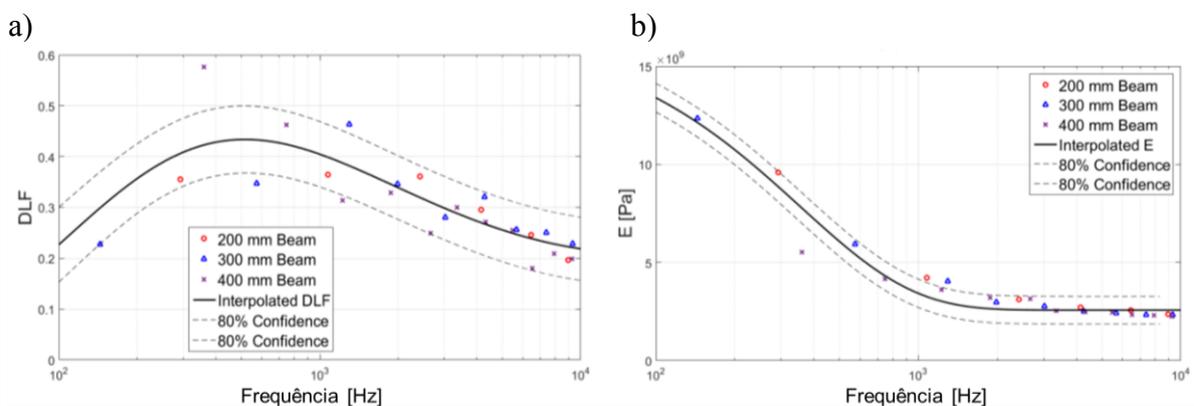


Figura 2.6 – a) DLF e b) módulo de elasticidade em uma viga com material viscoelástico. Fonte: (Pereira *et al.*, 2019).

2.3 SOLUÇÃO DOS MODELOS CONSIDERANDO AS PROPRIEDADES VARIANDO EM FUNÇÃO DA FREQUÊNCIA

Com os materiais viscoelásticos caracterizados é possível utilizar as suas propriedades na célula periódica para calcular as demais propriedades de interesse para os modelos SEA. Contudo, o *software* VAOne possui uma limitação em seu módulo de FE, o qual não permite considerar a variação do módulo de elasticidade em função da frequência, como é caso dos materiais viscoelásticos. Embora existam outros métodos para contornar esta limitação, como o proposto por Alimonti *et al.* (2018), que utiliza um método integral para resolução de problemas de autovalor não-linear gerado pela inclusão de parâmetros dependentes da frequência, notou-se que seria mais simples adaptar a solução dos modelos FE no VAOne para considerar essa dependência, visto que o uso do *software* se dá somente para a construção da base de dados para o posterior treinamento da rede neural.

Portanto, um método alternativo foi desenvolvido em que as matrizes de massa [M] e rigidez [K] são iterativamente recalculadas para cada banda de frequência e o modelo de FE é resolvido frequência por frequência. A Figura 2.7 mostra o fluxograma da abordagem adotada. Primeiramente é definida a geometria da célula unitária, conforme a Figura 2.4, e as propriedades de cada componente são atualizadas, de acordo com a caracterização dos materiais viscoelásticos. Aplica-se a condição de contorno periódica na célula unitária, informando os valores de N_x e N_y , e determina-se a faixa de frequência a ser analisada, neste caso de 500 Hz a 2000 Hz em bandas de um terço de oitava, região em que há a maior contribuição para o nível de pressão sonora global na cabine. Então, inicia-se o processo de solução do modelo FE e para cada frequência analisada as propriedades de DLF e módulo de elasticidade são atualizadas nos respectivos componentes da célula unitária, sendo necessário recalculas as matrizes de massa e rigidez para cada iteração. Apesar desta etapa gerar uma ineficiência ao processo, a solução dos modelos por essa abordagem ainda é viável, levando em torno de 25 a 30 minutos em uma máquina com processador Intel Core i7-8565U CPU @ 1.80GHz 1.99 GHz e 16 GB de memória RAM.

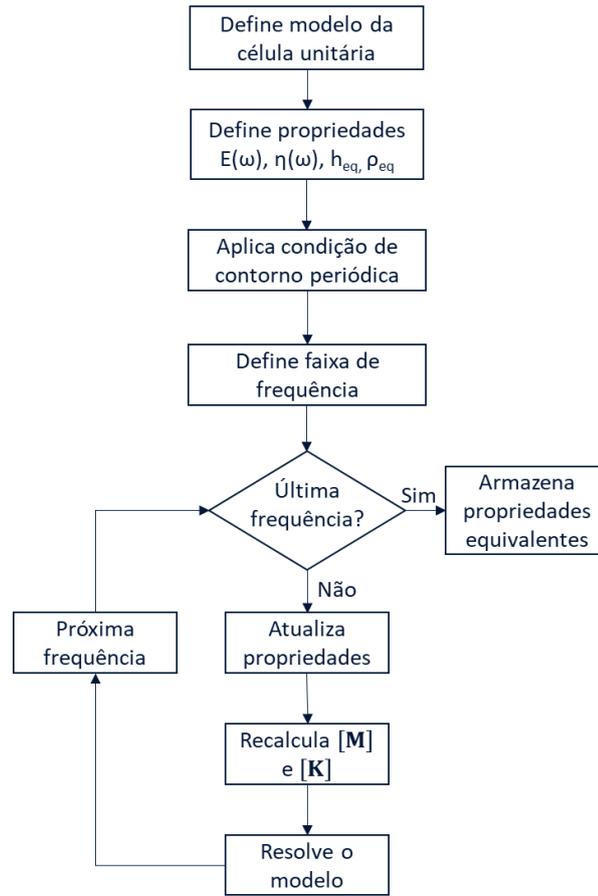


Figura 2.7 - Fluxograma para o cálculo das propriedades equivalentes DLF, densidade modal e RLF. Fonte: Elaborado pelo Autor.

2.4 RESULTADOS

Nesta seção são apresentados os resultados obtidos pela implementação da solução dos modelos de FE considerando a variação do DLF e módulo de elasticidade em função da frequência para a geometria de fuselagem mostrada na Figura 2.4. A Tabela 2-1 apresenta as propriedades de dois materiais utilizados para análise e a Figura 2.8 mostra as curvas de DLF e módulo de elasticidade obtidos experimentalmente pela caracterização do material viscoelástico conforme detalhado na seção 2.2.1 deste capítulo.

Tabela 2-1 – Periodicidade, espessura e densidade de 2 materiais viscoelástico avaliados.

Amostras	Periodicidade	Espessura Total [mm]	Densidade [kg/m ³]
Material 1	2x2	3,8	1044
Material 2	2x2	3,0	1830

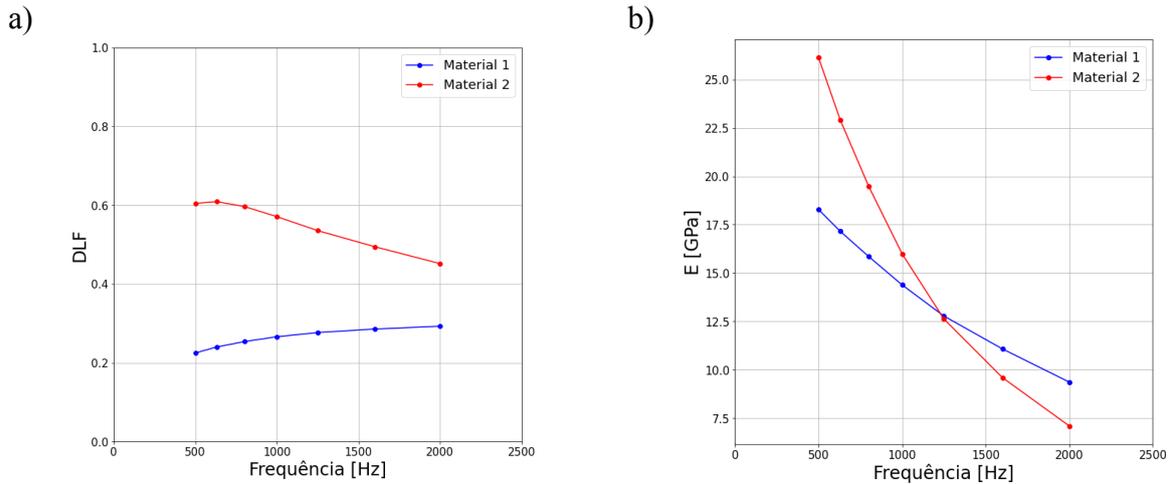


Figura 2.8 – a) DLF e b) módulo de elasticidade experimentais para 2 materiais distintos, considerando uma viga base de 0,8mm à -30°C.

Anteriormente à implementação da solução considerando a variação da frequência, utilizava-se do módulo de elasticidade referente à banda de 1000 Hz para resolver o modelo e obter as propriedades equivalentes. As Figuras 2.9, 2.10 e 2.11 apresentam os resultados obtidos para o DLF, Densidade Modal e RLF Equivalentes, respectivamente, comparando-os às soluções com o módulo de elasticidade constante.

É possível notar a influência da variação da rigidez na resposta das propriedades equivalentes. Para o DLF equivalente em algumas bandas de frequência observa-se um desvio de aproximadamente 5% na banda de 1250 Hz para o material 1 e na banda de 800 Hz para o material 2 e observa-se um desvio ainda maior na banda de 1600 Hz para o material 2, com um desvio de aproximadamente 8%. Essas diferenças são atribuídas à influência da rigidez sobre a atuação dos materiais viscoelásticos em dissipar energia proveniente da deformação da fuselagem em função da excitação produzida pela TBL por meio do cisalhamento do material viscoelástico. O máximo da dissipação de energia ocorre na região em que a taxa de variação do módulo de elasticidade em função da frequência também é máxima. Para a Densidade Modal Equivalente nota-se um comportamento crescente em função da frequência, o que é esperado, visto que a massa do sistema se mantém constante e a rigidez diminui conforme aumenta-se a frequência. Conforme a Figura 2.8, os modos da estrutura são deslocados para frequência mais baixas portanto, a quantidade de modos para a mesma faixa de frequência aumenta. Por fim, para o RLF Equivalente novamente a influência da rigidez é observada, conforme a estrutura torna-se mais flexível a capacidade de transmitir potência para o meio adjacente torna-se maior e o RLF diminui em função da frequência conforme a Equação 2.15.

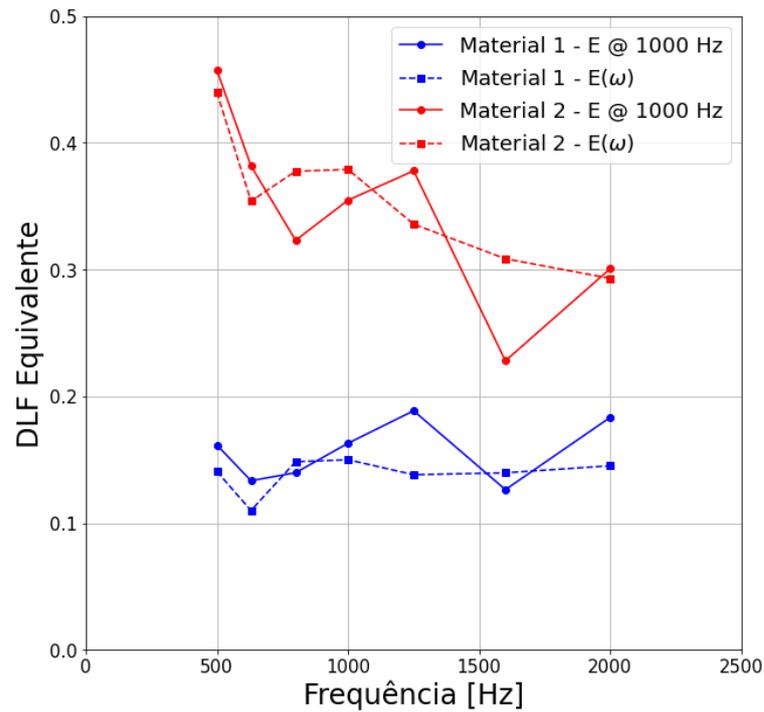


Figura 2.9 - Resultado do DLF equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C .

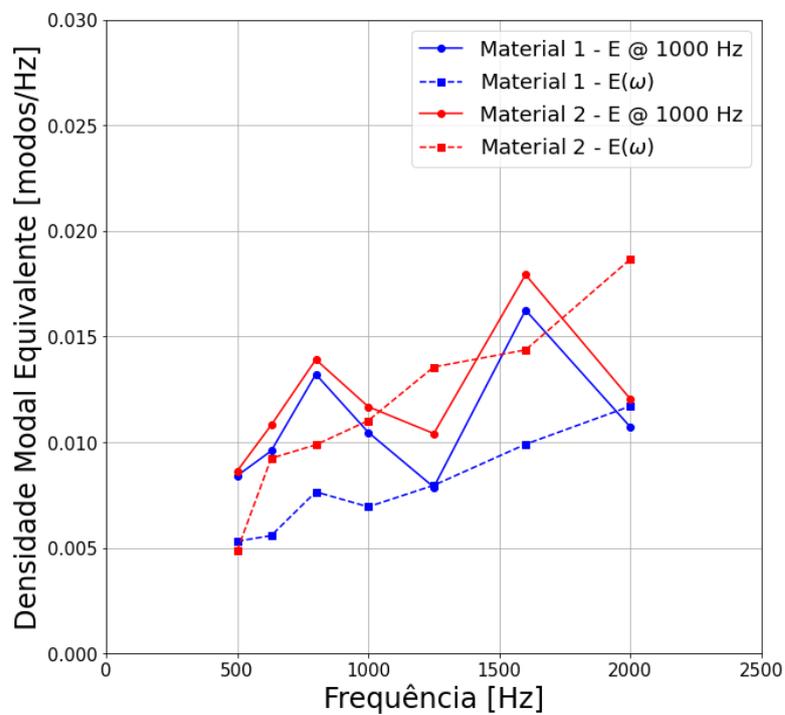


Figura 2.10 - Resultado da densidade modal equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C .

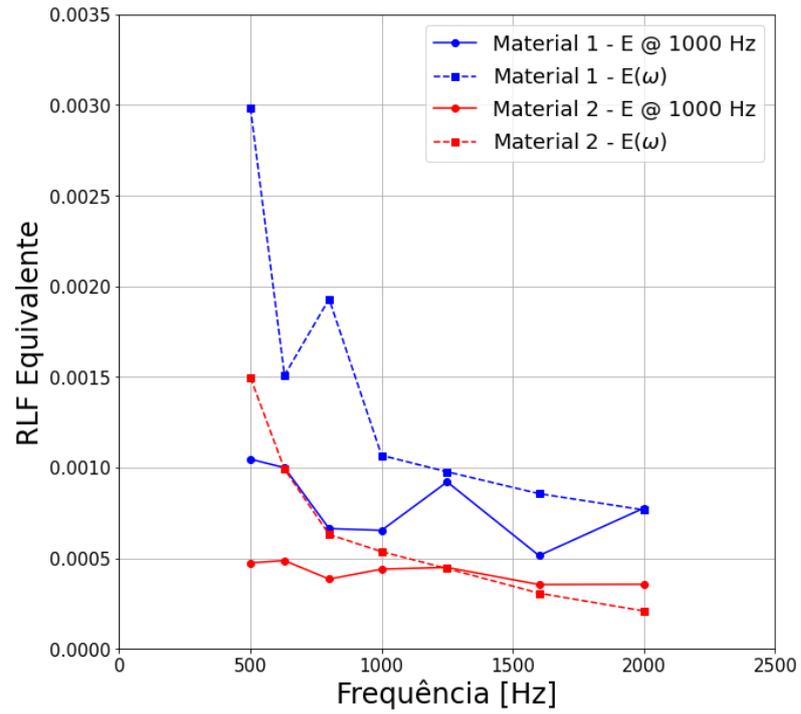


Figura 2.11 - Resultado fator de perda por radiação equivalente da célula unitária, considerando $N_x N_y = 4$, para 2 materiais distintos à -30°C .

3 APRENDIZADO DE MÁQUINA

Com o rápido crescimento do campo de Inteligência Artificial e *Big Data*, métodos de aprendizagem estatística tornaram-se um destaque para diversas áreas da indústria e do mercado em geral trazendo competitividade para novos produtos e serviços. O termo “aprendizagem estatística” refere-se ao grupo de ferramentas e métodos que permite modelar e compreender conjuntos de dados, extraíndo informações relevantes e identificando padrões (JAMES, WITTEN, HASTIE, TIBSHIRANI, 2013). Dentro do domínio de Inteligência Artificial, do inglês *Artificial Intelligence* (AI), existe a categoria denominada Aprendizado de Máquina, do inglês *Machine Learning* (ML), que consiste em algoritmos capazes de aprender automaticamente conforme a experiência, observação em dados ou interações com o ambiente (MITCHEL, 1997) e um subconjunto de ML são as redes neurais, que dão origem a uma gama de outras técnicas, como por exemplo redes neurais artificiais, recorrentes, convolucionais, generativas e dependendo da quantidade de camadas dessas redes neurais há ainda uma subcategoria denominada Aprendizado Profundo, do inglês *Deep Learning* (KAVLAKOGLU, 2020). A Figura 3.1 ilustra a relação entre as diferentes classes de AI.

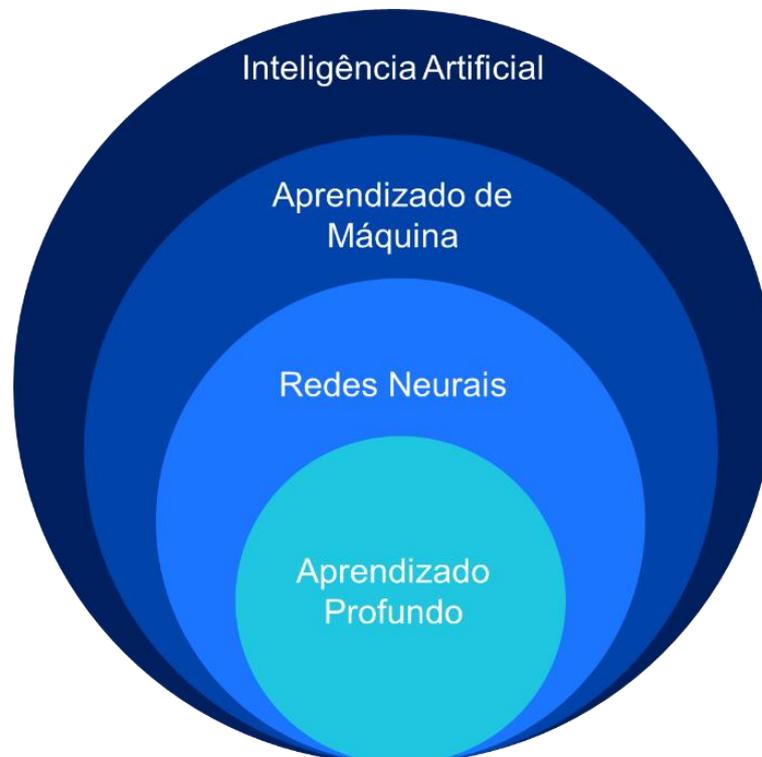


Figura 3.1 – Relação hierárquica dos domínios de AI. Fonte: Elaborado pelo Autor.

3.1 TIPOS DE APRENDIZADO DE MÁQUINAS

Existem três tipos de ML, supervisionado, não supervisionado, aprendizado por reforço e há ainda um quarto denominado semi-supervisionado, que é uma interseção das categorias mencionadas anteriormente (JAMES, WITTEN, HASTIE, TIBSHIRANI, 2013).

Aprendizado supervisionado consiste em aprender uma função a partir de um conjunto de dados que estabelece a relação entre atributos, que são as variáveis independentes, e rótulos, que são as variáveis dependentes. Portanto conhecendo esta relação o algoritmo é capaz de prever valores contínuos sobre um determinado fenômeno observado, sendo este um problema de regressão, ou prever valores discretos, que representam diferentes classes, identificado como um problema de classificação.

Para o aprendizado não supervisionado os rótulos não são conhecidos e somente um conjunto de dados com atributos é fornecido. Neste tipo de situação, o objetivo é treinar um algoritmo que seja capaz de identificar padrões nos dados e que permita analisá-los e agrupá-los conforme suas características. Quando se combina uma pequena porção de dados rotulados com uma porção grande dados não rotulados em um mesmo conjunto, origina-se o aprendizado semi-supervisionado e o objetivo é treinar um modelo capaz de realizar previsões assim como o aprendizado supervisionado.

Por fim, o aprendizado por reforço consiste na interação entre um agente com o ambiente em uma determinada janela de tempo. A cada instante, o agente recebe um conjunto de observações do ambiente e precisa realizar uma ação ou tomar uma decisão. Uma recompensa ou penalidade é dada ao agente conforme a ação tenha sido correta ou incorreta e a pontuação acumulada do agente mede a evolução do aprendizado dele. Este tipo de algoritmo é comum para aplicações em robótica e jogos.

Neste trabalho, deseja-se explorar a aplicabilidade de redes neurais artificiais para o aprendizado supervisionado das propriedades equivalentes apresentadas na seção 2.4, combinando a abordagem determinística do método de FE com a abordagem estatística de ML. O modelo estatístico resultante dessa combinação é chamado modelos substitutos e será explicado em mais detalhes no capítulo 4.

3.2 MODELO DE PROCESSO PARA APRENDIZADO DE MÁQUINA

De maneira geral a área de ML tem como objetivo automatizar o processo de aprendizado com base em análise de dados utilizando algoritmos que geram modelos estatísticos capazes de realizar previsões ou inferências (JAMES, WITTEN, HASTIE, TIBSHIRANI, 2013):

- **Predição:** o objetivo é estimar uma função capaz de prever valores de rótulos com base nos atributos e com acurácia. Neste sentido, a função pode ser vista como uma caixa preta que gera uma previsão dado um valor de entrada.

- **Inferência:** o objetivo não é necessariamente gerar previsões e sim entender a relação ou mesmo a sensibilidade entre atributos e rótulos. Por exemplo, quais atributos são os mais importantes e como interferem no comportamento dos rótulos.

Em ambos os casos, o desenvolvimento desses modelos consiste em componentes fundamentais (ZHANG, LIPTON, LI, SMOLA, 2021):

- **Dados** são conjuntos de observações representadas de forma numérica, compatível para ingestão do modelo para treinamento, validação e teste. Quanto mais dados disponíveis, melhor para a construção do modelo, porém vale reforçar a máxima: lixo entra, lixo sai, do inglês *garbage in, garbage out*, ou seja, são necessários dados certos, limpos e preparados para o processo de treinamento, validação e teste.

- **Modelo** pode ser entendido como um objeto que ingere os dados e os transforma de alguma forma para realizar uma previsão como saída deste objeto, ou seja, é aprendida uma função que relaciona o mapeamento entre atributos e rótulos.

- **Função objetivo** é a função pela qual mede-se a performance de aprendizado do modelo. Normalmente, busca-se minimizar a função objetivo, isto é, minimizar o erro entre a observação real e a previsão realizada pelo modelo. Existem diversas métricas para medir o quão bom ou ruim está sendo o aprendizado do modelo, sendo o erro quadrático médio a medida mais comum. Neste sentido, vale ressaltar que um valor baixo da função objetivo não significa necessariamente um bom modelo, pois é possível ocorrer o problema de sobre-ajuste, do inglês *overfitting*, ou seja, o modelo não é capaz de generalizar as previsões para o conjunto de teste, que são dados não utilizados durante o processo de aprendizagem, e fica ajustado somente para o conjunto de treinamento e validação.

- **Algoritmo de otimização** é a ferramenta pela qual se minimiza a função objetivo definida para o problema. Existem diversos algoritmos disponíveis para realizar essa

tarefa, o algoritmo de Gradiente Descendente é o mais popular e dele derivam-se variações como Gradiente Descendente Estocástico, Gradiente Descendente Estocástico em Pequenos Lotes, *AdaGrad* (Gradiente Adaptativo), *RMSprop* e *Adam* (Momento Adaptativo). A seção 3.2.2 apresentará mais detalhes de alguns algoritmos e a escolha daquele utilizado neste trabalho.

A Figura 3.2 mostra um modelo de processo padrão utilizado em ciência de dados e ML denominado CRISP-DM, do inglês *CRoss Industry Standard Process for Data Mining*, que compreende todas as etapas do desenvolvimento de soluções baseadas em ML, que parte do entendimento do problema e dos dados disponíveis, passando pela parte de modelagem e construção do modelo, sua posterior avaliação e então a implantação, considerando que todo esse ciclo deve ser gerido continuamente para garantir a robustez do modelo.

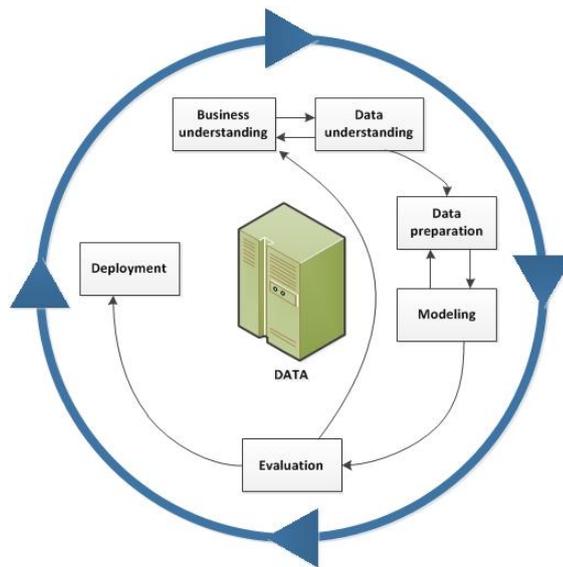


Figura 3.2 – Modelo CRISP-DM. Fonte: (IBM, 2021).

3.2.1 Treinamento de modelos de ML

O processo de treinamento é a etapa em que o modelo aprende e ajusta seus parâmetros internos com base nos dados fornecidos utilizando um algoritmo de otimização. O trabalho desenvolvido por Da Silva (2021) apresenta os conceitos e o processo de treinamento de modelos de ML de forma clara e lógica, portanto seguindo a mesma abordagem e utilizando um modelo simples de regressão linear unidimensional, busca-se detalhar a etapa de treinamento e cálculo dos parâmetros do modelo para fins de compreensão.

Considerando um conjunto de observações quaisquer, conforme Figura 3.3, tem-se o objetivo de determinar uma função que estabeleça a relação entre a variável independente x e a variável dependente y .

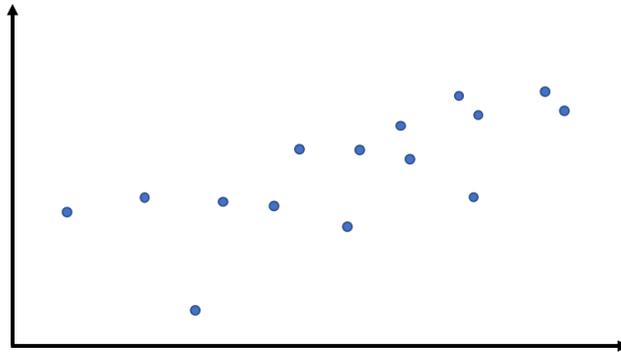


Figura 3.3 – Conjunto de dados quaisquer. Fonte: Elaborado pelo Autor.

A função mais simples que pode ser usada para descrever a relação observada é uma reta, como ilustra a Figura 3.4, dada pela Equação 3.1 e este processo é denominado regressão:

$$\hat{y} = b + wx, \quad (3.1)$$

na qual, \hat{y} é o valor predito, b é chamado de viés, do inglês *bias*, e w é o peso da variável x . Portanto, conhecendo os parâmetros da reta \hat{y} , ou seja, o viés e o peso, é possível encontrar uma função contínua que prediz y dada uma entrada x e o objetivo principal em ML é justamente determinar os melhores desses parâmetros de forma a minimizar a diferença entre o valor real observado e o valor predito, conforme a Figura 3.4.

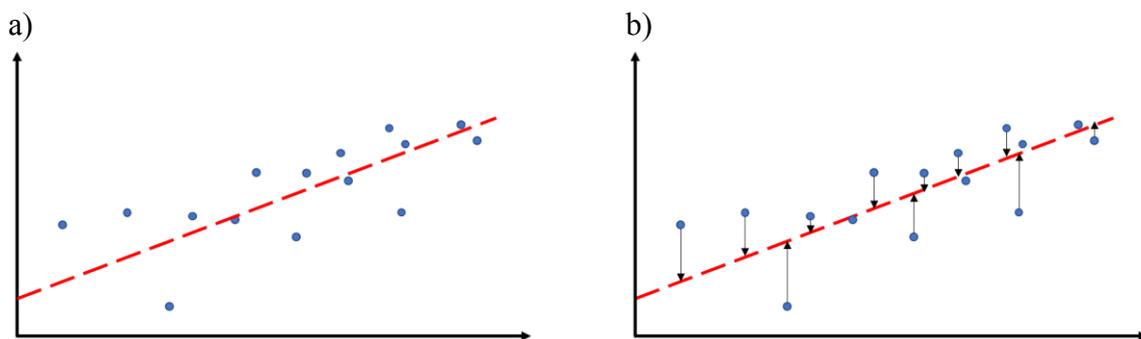


Figura 3.4 – a) Modelo linear ajustado com base nos dados; b) Erro entre observações e modelo. Fonte: Elaborado pela Autor.

De forma geral, os problemas de ML possuem inúmeros atributos que caracterizam a relação com os rótulos e a Equação 3.1 pode ser generalizada como:

$$\hat{y} = b + w_1x_1 + w_2x_2 + w_3x_3 + \dots = b + \sum_{i=1}^n w_ix_i \quad (3.2)$$

sendo w_i e x_i os n -ésimos pesos e atributos. Para se obter o viés e todos os pesos que minimizam o erro entre y e \hat{y} é necessário definir uma função objetivo, chamada também de função de perda, que quantifica a performance de aprendizado e um algoritmo de otimização, que minimiza o valor da função objetivo a cada iteração até convergir para o conjunto de melhores parâmetros. A seção 3.2.2 a seguir apresentará em mais detalhes o processo de otimização.

Ao final da etapa de treinamento é necessário validar a capacidade do modelo de realizar previsões em conjunto de dados distinto, chamado conjunto de teste. O objetivo é avaliar a capacidade de generalização do modelo para quaisquer novos dados coletados e fornecidos para realizar previsões. Como mencionado anteriormente, é possível que durante o processo de treinamento ocorra o *overfitting* e o modelo tenha uma performance boa somente com os dados de treinamento e uma vez que dados não vistos pelo modelo são usados para gerar novas previsões a performance cai significativamente. Considerando o caso do treinamento de um modelo unidimensional, a Figura 3.5 mostra visualmente o efeito de *overfitting* e fica evidente que o modelo é ajustado ao máximo para aproximar uma função que passa por todos os pontos. Quando uma nova observação é fornecida ao modelo, conforme a Figura 3.5, nota-se um erro significativo entre o valor real e o valor predito. De forma análoga, é possível ocorrer o sob-ajuste, do inglês *underfitting*, em que o modelo não é capaz de identificar padrões ou tendências com base no conjunto de treinamento, resultando em um modelo muito simples, que não relaciona os atributos com os rótulos de forma acurada, conforme a Figura 3.5. Então, um bom modelo é aquele que consegue capturar os padrões e tendências e generalizar para um novo conjunto de observações, realizando previsões com acurácia, Figura 3.5.

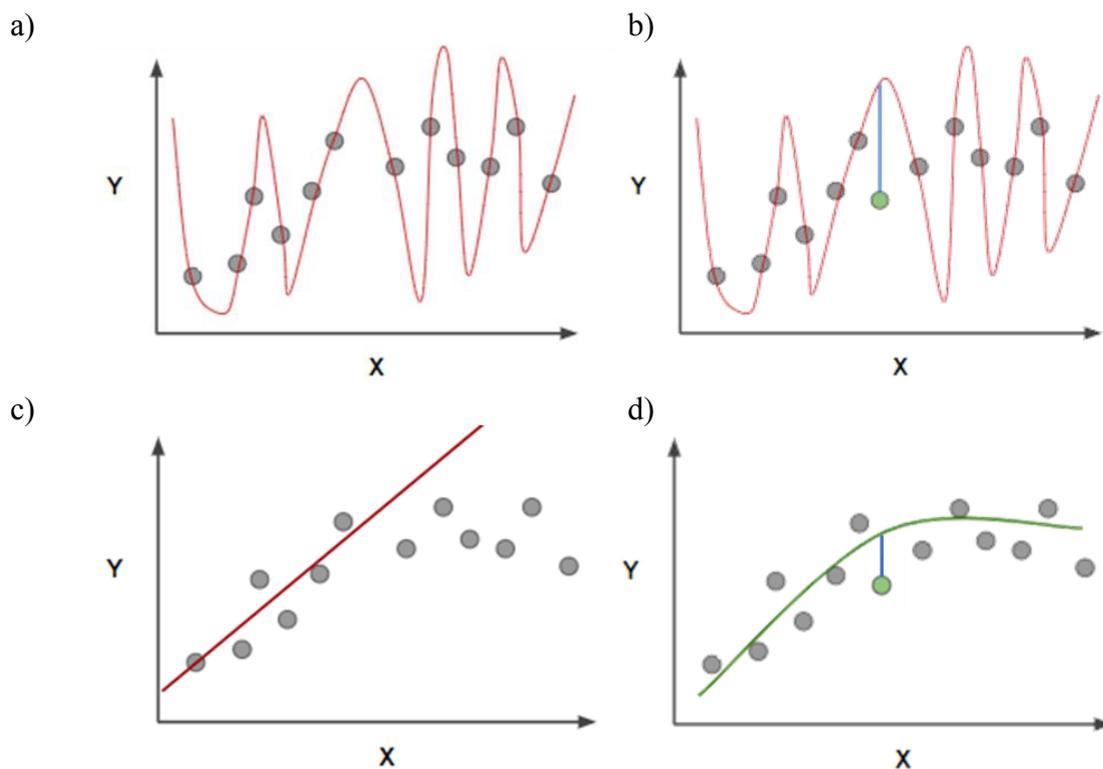


Figura 3.5 - a) Modelo ajustado com *overfitting*; b) Nova predição com erro significativo; c) Modelo ajustado com *underfitting*; d) Modelo com boa generalização. Fonte: (Portilla, 2022).

Embora seja possível visualizar graficamente a relação entre atributos e rótulos e a função obtida pelo ajuste de modelo para problemas de poucas dimensões, permitindo inferir a capacidade de generalização do modelo, como mencionado anteriormente, normalmente os problemas possuem inúmeros atributos. Portanto, é necessária uma forma mais adequada de aferir a performance dos modelos e isto é feito monitorando a evolução do erro ao longo do processo de treinamento, conforme a Figura 3.6. Para um bom modelo é esperado que o erro diminua a cada época até atingir um valor de convergência e por outro lado um modelo ruim apresenta um crescimento do erro, indicando que a dificuldade em extrair e identificar os padrões nos dados de treinamento. O termo época se refere a cada iteração do processo de otimização para estimar os parâmetros internos do modelo.

A mesma avaliação pode ser feita considerando a evolução do erro para o conjunto de teste, conforme Figura 3.7. Um bom modelo generalista deve apresentar uma evolução decrescente do erro em função das épocas para ambos os conjuntos, enquanto um modelo com *overfitting* tem um comportamento crescente do erro após um determinado tempo de treinamento, como mostra a Figura 3.7. A intersecção entre as curvas de erro indica o momento de cessar o treinamento para evitar o *overfitting*.

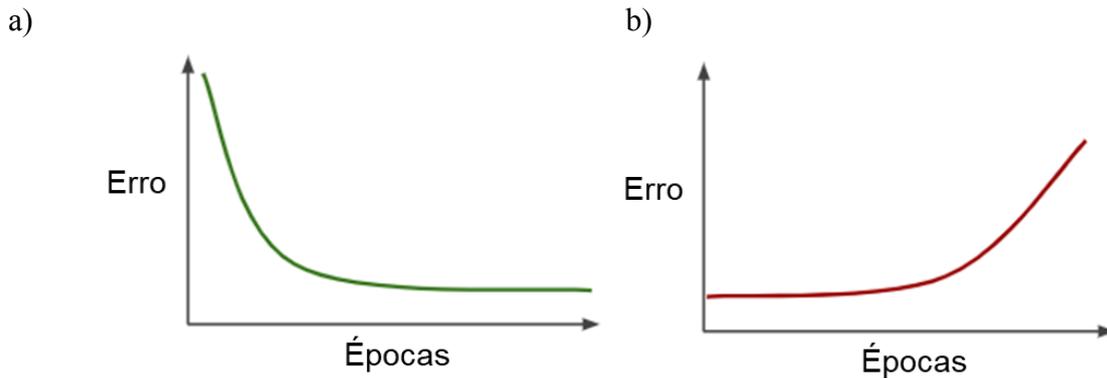


Figura 3.6 – Evolução do erro ao longo do tempo para um modelo a) bom; b) ruim. Fonte: (Portilla, 2022).

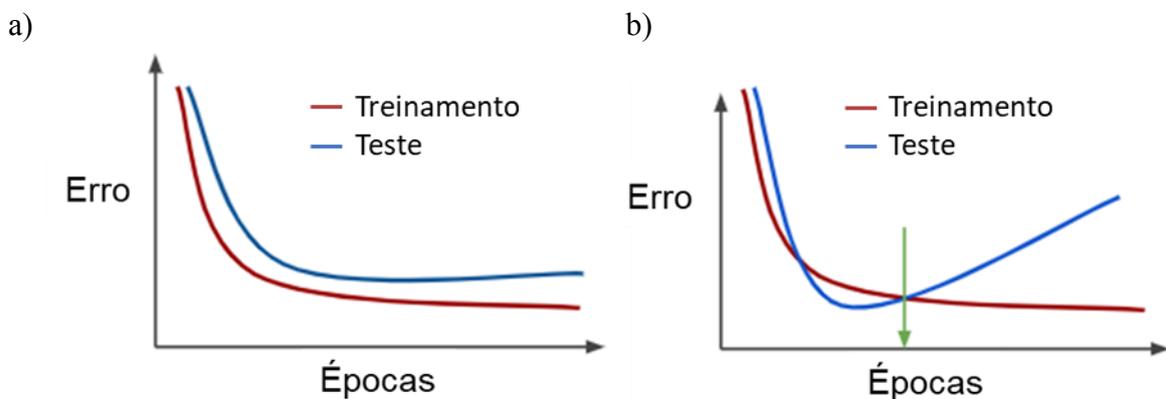


Figura 3.7 – a) Evolução do erro ao longo do tempo para os conjuntos de treinamento e teste; b) modelo treinado com *overfitting*. Fonte: (Portilla, 2022).

3.2.2 Algoritmos de otimização

Para atualizar os parâmetros internos do modelo em treinamento os algoritmos de otimização são essenciais para encontrar o melhor conjunto de pesos e vieses que minimizam a função objetivo definida para o problema. Existem diversos tipos de funções objetivo e a escolha de qual utilizar depende do tipo de modelo de ML. Para problemas de classificação a função de Entropia-Cruzada é a mais utilizada e para problemas de regressão as funções Erro Quadrático, Erro Quadrático Médio, Erro Absoluto Médio e Raiz do Erro Quadrático Médio são as mais comuns.

No domínio de ML e ciência de dados, o algoritmo de Gradiente Descendente é fundamental para realizar a tarefa de minimizar a função objetivo e extrair o melhor conjunto de parâmetros do modelo. Desse algoritmo, originam-se outras variações como Gradiente Descendente Estocástico, Gradiente Descendente Estocástico em Pequenos Lotes, Gradiente

Descendente Baseado em Momento e outros como *AdaGrad* (Gradiente Adaptativo), *RMSprop* e *Adam* (Momento Adaptativo). Para fins de compreensão, a aplicação do algoritmo de Gradiente Descendente é explicada para o caso de regressão linear a partir da Equação 3.2 considerando $N = 1$, em que o objetivo final é encontrar os parâmetros da reta, conforme a Figura 3.4, que minimiza o erro entre as previsões e os valores reais dos rótulos. O conceito deste algoritmo consiste em avançar iterativamente na direção oposta ao gradiente da função objetivo até convergir para o mínimo de acordo com a seguinte regra:

$$\mathbf{W} \leftarrow \mathbf{W} - \alpha \nabla J(\mathbf{W}), \quad (3.3)$$

na qual \mathbf{W} é o vetor de parâmetros do modelo a serem ajustados, isto é, o viés b e o peso w_1 , e α é uma constante chamada Taxa de Aprendizagem, que define o tamanho dos passos de cada iteração em direção ao mínimo da função, J é a função objetivo e ∇ é o operador gradiente. Como exemplo, usando o Erro Quadrático Médio, a função objetivo é dada por:

$$J = \frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2 \quad (3.4)$$

em que M é o número de amostras do conjunto de dados. Substituindo a Equação 3.1 na Equação 3.3 tem-se:

$$J(b, w_1) = \frac{1}{M} \sum_{i=1}^M (y_i - (b + w_1 x_i))^2 = \frac{1}{M} \sum_{i=1}^M (y_i - b - w_1 x_i)^2. \quad (3.5)$$

Em seguida, é necessário calcular as derivadas parciais de J em função de b e w_1 :

$$\frac{\partial J(b, w_1)}{\partial b} = \frac{\partial}{\partial b} \frac{1}{M} \sum_{i=1}^M (y_i - b - w_1 x_i)^2, \quad (3.6)$$

e

$$\frac{\partial J(b, w_1)}{\partial w_1} = \frac{\partial}{\partial w_1} \frac{1}{M} \sum_{i=1}^M (y_i - b - w_1 x_i)^2, \quad (3.7)$$

aplicando a regra da cadeia conclui-se que:

$$\frac{\partial J(b, w_1)}{\partial b} = -\frac{2}{M} \sum_{i=1}^M (y_i - (b + w_1 x_i)), \quad (3.8)$$

e

$$\frac{\partial J(b, w_1)}{\partial w_1} = -\frac{2}{M} \sum_{i=1}^M (y_i - (b + w_1 x_i)) x_i, \quad (3.9)$$

Portanto, como y_i e x_i são valores conhecidos do conjunto de dados, basta informar valores iniciais para b e w_1 , calcular $\nabla J(\mathbf{W})$ com as Equações 3.8 e 3.9 e substituir iterativamente estes valores na Equação 3.3 até atingir o critério de convergência ou número de passos definido. A Figura 3.8 apresenta visualmente um exemplo do processo de otimização do algoritmo de Gradiente Descendente.

A implementação deste método é simples e direta, porém o custo computacional está diretamente associado ao número de atributos n , número de amostras do conjunto de dados M , conforme as Equações 3.2 e 3.4, respectivamente, e da própria forma da função \hat{y} , uma vez que o cálculo do gradiente é parte fundamental do algoritmo e funções mais complexas tendem a levar mais tempo para computar suas derivadas. Portanto, o número total de cálculos necessários para cada passo do algoritmo é de $n \times M$ e dependendo do problema em questão torna-se custoso o uso de Gradiente Descendente.

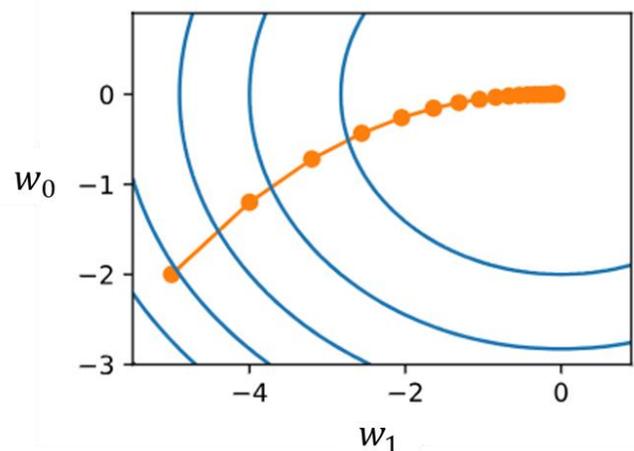


Figura 3.8 – Exemplo de trajetória da otimização dos parâmetros w_0 e w_1 . Fonte: Adaptado de (Zhang, Lipton, Li, Smola, 2021).

Para contornar esta limitação, o algoritmo de Gradiente Descendente Estocástico difere sua estratégia de avaliar a função objetivo considerando somente um ponto do conjunto de dados, assim a Equação 3.4 pode ser simplificada:

$$J = (y_i - \hat{y}_i)^2, \quad (3.10)$$

e desta forma evita-se calcular o Erro Quadrático Médio com todas as observações do conjunto de dados. Embora seja possível melhorar o algoritmo em termos de custo computacional para processamento dos dados em cada passo da otimização, pode se observar na Figura 3.9 que a escolha aleatória do ponto do conjunto de dados para calcular os novos parâmetros gera uma trajetória aleatória até convergir para o mínimo da função. Uma alternativa que combina ambos os cenários é o algoritmo de Gradiente Descendente em Pequenos Lotes. Neste caso, uma parte do conjunto de dados é utilizada para calcular a função objetivo, e a Equação 3.4 é modificada para:

$$J = \frac{1}{K} \sum_{i=1}^K (y_i - \hat{y}_i)^2, \quad (3.11)$$

sendo K um subconjunto de observações utilizadas para calcular cada passo do processo de otimização. Por exemplo, escolhendo $K = 100$, a cada iteração 100 observações são usadas para calcular os novos parâmetros. Portando o uso de pequenos lotes representa uma combinação que captura as vantagens de ambos os métodos anteriores, sendo computacionalmente e estatisticamente eficiente, isto é, baixo tempo de processamento e maior garantia de convergência da solução.

Os outros algoritmos mencionados no início desta seção incorporam outras técnicas como o uso de Taxa de Aprendizagem dinâmica, médias móveis, capacidade de tratar matrizes esparsas para assegurar a convergência e acelerar o tempo de processamento. O algoritmo *Adam (Adaptive Momentum)* concatena características de diversos outros algoritmos resultando em um algoritmo robusto e aplicável para diversos tipos de problemas de ML (KINGMA; BA, 2015), logo para o design da arquitetura da rede neural foi utilizado este algoritmo, conforme será mostrado na seção 4.2.2.

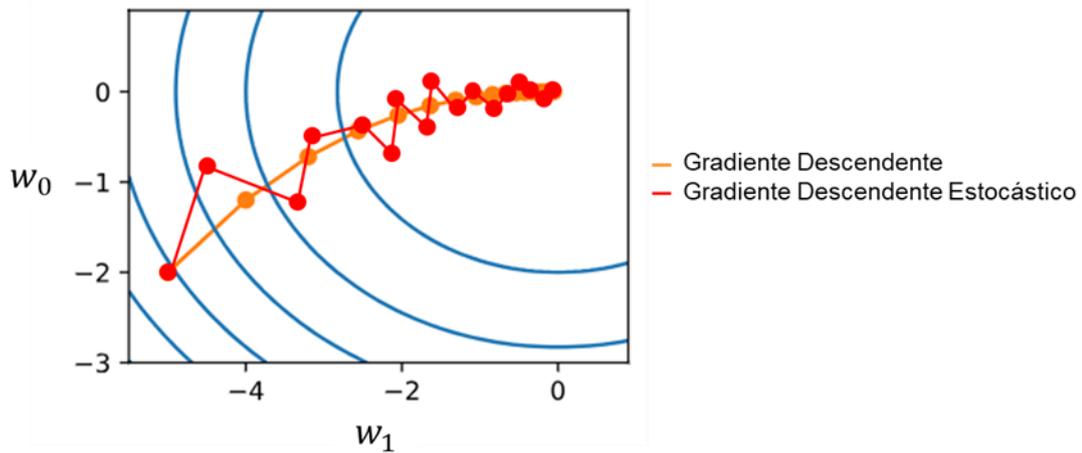


Figura 3.9 - Exemplo de comparação da trajetória entre os algoritmos de Gradiente Descendente e Gradiente Descendente Estocástico. Fonte: Adaptado de (Zhang, Lipton, Li, Smola, 2021).

3.2.3 Métricas de Avaliação

Após treinar e ajustar os parâmetros internos a última etapa do processo de construção do modelo de ML e avaliar a performance através do conjunto de teste. Dependendo do tipo de problema existem métricas diferentes que quantificam o quão bom ou ruim o modelo é. Para problemas de classificação a matriz de confusão é a base para o cálculo de métricas como acurácia, precisão, *recall* e *F1-score*. Já para problemas de regressão as métricas utilizadas são o Erro Quadrático Médio, Raiz do Erro Quadrático Médio, Erro Absoluto Médio, coeficiente R^2 , *Explained Variance Score* e somente estes serão explicados nesta seção. As métricas de classificação estão fora do escopo deste trabalho.

Como visto anteriormente sob a perspectiva de função objetivo, o Erro Quadrático Médio é dado pela Equação 3.4 e pode ser utilizado como métrica de avaliação. Porém, é uma métrica que dependendo da ordem de grandeza das variáveis y e \hat{y} pode gerar valores muito pequenos ou muito grandes, sendo de difícil interpretação. Alternativamente, é possível calcular a Raiz do Erro Quadrático Médio:

$$REQM(y, \hat{y}) = \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2}, \quad (3.12)$$

Porém, essa métrica não tangibiliza de forma clara o quão bom ou ruim o modelo é. Uma métrica mais simples de interpretar é o Erro Médio Absoluto:

$$EMA(y, \hat{y}) = \frac{1}{M} \sum_{i=1}^M |y_i - \hat{y}_i|, \quad (3.13)$$

pois mede a diferença em magnitude entre a média dos valores reais dos rótulos e os valores preditos pelo modelo. Outra métrica comumente utilizada é o coeficiente R^2 :

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^M (y_i - \hat{y}_i)^2}{\sum_{i=1}^M (y_i - \bar{y})^2}, \quad (3.13)$$

na qual $\bar{y} = \frac{1}{M} \sum_{i=1}^M y_i$. A interpretação é simples, o numerador representa a soma quadrática do erro residual e o denominador contabiliza a soma quadrática total, que quantifica a variância dos dados observados. Quanto menor o erro residual, o valor de R^2 tende a 1 e pode indicar o quão acurado é o modelo. No caso ideal de R^2 igual a 1 as predições \hat{y}_i teriam exatamente os mesmos valores dos rótulos y_i . Por fim, a métrica *Explained Variance Score* é similar ao coeficiente R^2 e é dado por:

$$EVS(y, \hat{y}) = 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}}, \quad (3.14)$$

sendo $Var\{ \}$ a variância e reescrevendo a equação acima:

$$EVS(y, \hat{y}) = 1 - \frac{\sum_{i=1}^M (y_i - \hat{y}_i - \bar{e})^2}{\sum_{i=1}^M (y_i - \bar{y})^2}, \quad (3.15)$$

em que \bar{e} é a média do erro $e = y_i - \hat{y}_i$. Portanto, se $\bar{e} = 0$, conclui-se que $R^2 = EVS$ e a interpretação é similar.

3.3 REDES NEURAIAS

Redes neurais são uma categoria de modelo de ML baseada na estrutura do cérebro humano e imita seu comportamento funcional. O cérebro é um órgão complexo composto de

inúmeros neurônios interconectados que transmitem impulsos elétricos que podem gerar ações motoras do corpo e capacidade cognitiva de reconhecer padrões, aprender e tomar decisões. De forma análoga, as redes neurais são compostas por elementos unitários chamados de nós, que representam os neurônios. O arranjo de diferentes combinações de conexões entre nós e a quantidade de camadas constituem o que é a rede neural e por meio de conceitos matemáticos esses algoritmos buscam reproduzir as capacidades do cérebro.

Considerando a Equação 3.1, como exemplo, é possível visualizá-la como uma rede neural com uma camada de entrada e uma camada de saída. Portanto uma simples regressão linear pode ser entendida como um aprendizado raso, oposto ao aprendizado profundo, pois o princípio matemático é mesmo e a aplicação deste tipo de método data de mais de dois séculos atrás (SCHMIDHUBER, 2022). Um grande marco na história do desenvolvimento de algoritmos de ML foi o trabalho de Rosenblatt (1958), que introduziu o *perceptron*, um modelo probabilístico que combinava o neurônio com uma função sendo capaz de gerar classificações binárias. Além disso, o conceito de modelo de percepção de múltiplas camadas, do inglês *Multilayer Perceptrons*, foi apresentado e consistia em uma estrutura de neurônios conectados com pesos randomizados e uma camada de saída adaptativa. Diversos trabalhos foram elaborados para estudar as capacidades e limitações desse tipo de modelo e os algoritmos evoluíram, originando vários métodos mais recentes como redes neurais recorrentes, redes neurais convolucionais, redes neurais generativas, entre outros, inclusive viabilizando o aprendizado profundo.

3.3.1 Modelo de redes neurais artificiais

O modelo de redes neurais pode ser visualizado na Figura 3.10. Ele consiste em três tipos de camadas distintas: primeiro a camada de entrada, referente ao número de observações que compõe o conjunto de atributos, segundo as camadas ocultas, que contém as transformações lineares mais a adição dos vieses para ajustar o modelo, e então a camada de saída, que é de fato a predição da variável dependente \hat{y} . Portanto, de forma semelhante ao neurônio biológico, a informação da camada de entrada é transmitida para as camadas ocultas adjacentes e assim sucessivamente até o final da rede produzindo uma saída qualquer, seja uma regressão, uma classificação ou mesmo uma ação, e cada neurônio atua aplicando uma função que transforma a informação recebida e repassa como uma nova informação. Este

processo é mostrado na Figura 3.11, considerando somente um neurônio para facilitar a visualização. Pode-se representar matematicamente o neurônio com a seguinte equação:

$$\hat{y} = \sum_{i=1}^n x_i w_i + b, \quad (3.16)$$

e agora há um viés b_i para cada atributo.

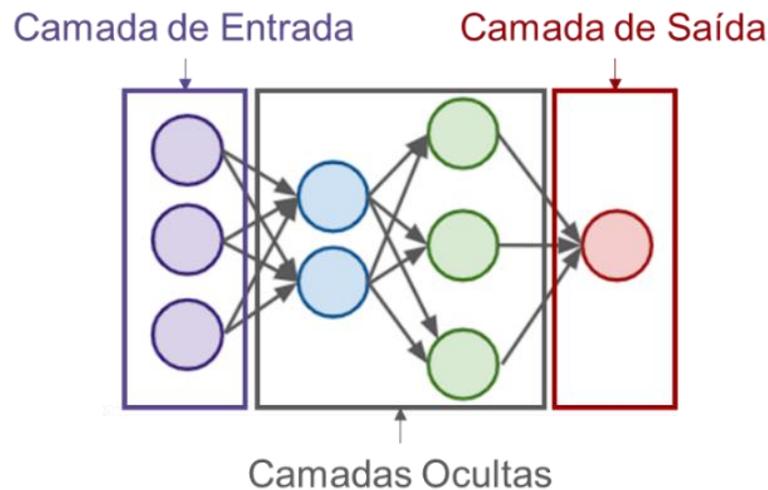


Figura 3.10 – Representação gráfica de uma rede neural genérica. Fonte: Adaptado (Portilla, 2022).

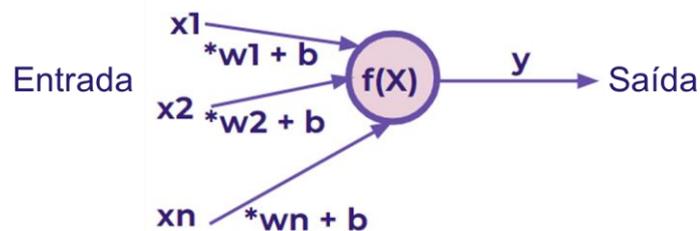


Figura 3.11 – Representação de um único neurônio com entradas x_1, x_2 e x_3 aplicando uma transformação resultando na saída y . Fonte: Adaptado (Portilla, 2022).

Entendendo o conceito base das redes neurais é possível diferenciar o aprendizado raso do aprendizado profundo. No primeiro caso, a quantidade de camadas ocultas se limita a somente uma, independente de quantidade de neurônios, que representam a largura da rede neural. Já no segundo caso, se a quantidade de camadas ocultas for igual ou superior a duas, considera-se aprendizado profundo e teoricamente não há restrições para o tamanho da rede neural. Entretanto, nota-se que pela Equação 3.6 o modelo só é capaz de realizar transformações lineares com os pesos, atributos e vieses e a combinação de infinitas funções

lineares continua resultando em uma função linear. Portanto há a necessidade de incorporar um elemento que extrapole para um comportamento não linear e a rede neural consiga identificar padrões complexos, que pode ser feito com o uso de funções de ativações. Essas funções são operadores diferenciáveis, normalmente, não lineares, e definem se um neurônio deve ser ativado ou não com base no resultado da soma ponderada mais a adição do viés, emulando o comportamento biológico das sinapses. A Figura 3.12 ilustra como a função ativação é incorporada na estrutura da rede neural.

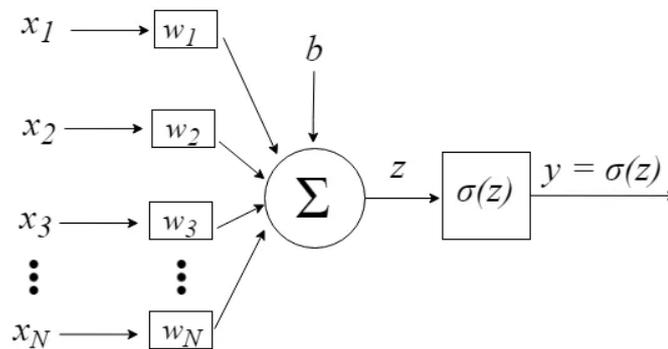


Figura 3.12 – Inserção da função de ativação na rede neural. Fonte: (Da Silva, L. C., 2021).

Existem várias funções de ativação, dentre as quais pode se citar as principais:

- Sigmoid: a função Sigmoid é chamada também de função logística ou curva S e é dada pela seguinte expressão

$$\text{Sigmoid}(z) = \frac{1}{1 + e^{-z}}, \quad (3.17)$$

na qual z é a soma ponderada mais os vieses e a Figura 3.13 apresenta sua forma.

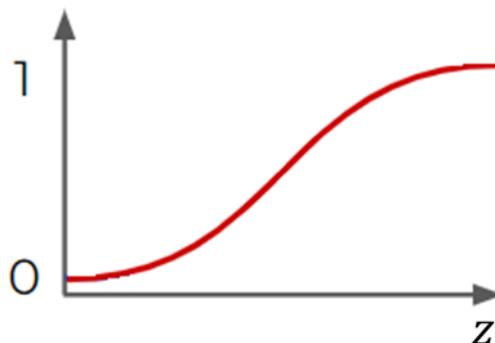


Figura 3.13 – Gráfico da função Sigmoid. Fonte: (Portilla, 2022).

A derivada da função Sigmoid é dada por:

$$\frac{d}{dz} \text{Sigmoid}(z) = \text{Sigmoid}(z)(1 - \text{Sigmoid}(z)) \quad (3.18)$$

e a Figura 3.14 mostra o gráfico da Equação 3.18.

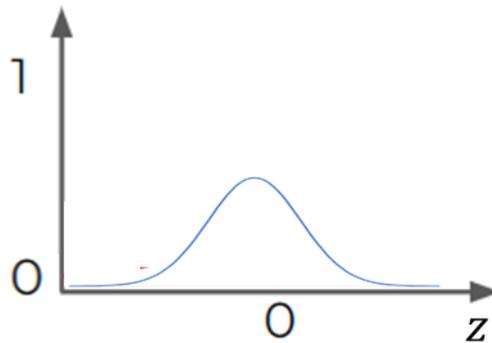


Figura 3.14 – Gráfico da derivada da função Sigmoid. Fonte: Elaborado pelo Autor.

- Tanh: a função Tanh é a tangente hiperbólica dada pela expressão:

$$\text{Tanh}(z) = \frac{\text{senh}(z)}{\text{cosh}(z)} = \frac{1 - e^{-2z}}{1 + e^{-2z}} \quad (3.19)$$

e a Figura 3.15 mostra graficamente sua forma. A função Tanh é semelhante à função Sigmoid, porém pode assumir valores entre -1 e 1 , enquanto a função Sigmoid se restringe a valores positivos.

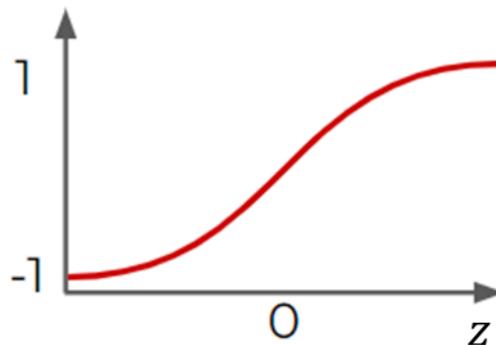


Figura 3.15 – Gráfico da função Tanh. Fonte: (Portilla, 2022).

A derivada da função Tanh é dada por:

$$\frac{d}{dz} \text{Tanh}(z) = 1 - \text{Tanh}^2(z) \quad (3.20)$$

e a Figura 3.16 mostra graficamente a Equação 3.20. Novamente, o formato da derivada da função Tanh é semelhante à derivada da função Sigmoid, porém com diferença nas imagens dessas funções.

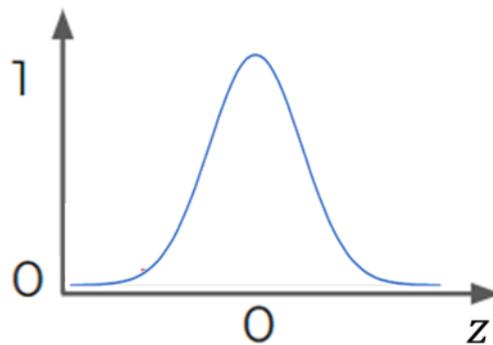


Figura 3.16 – Gráfico da derivada da função Tanh. Fonte: Elaborado pelo Autor.

- ReLU: Unidade Linear Retificada, do inglês *Rectified Linear Unit*, dada pela seguinte expressão:

$$\text{ReLU}(z) = \max(z, 0), \quad (3.21)$$

ou seja, a função é igual a 0 para qualquer $z \leq 0$ e igual a z para $z > 0$, conforme a Figura 3.17.

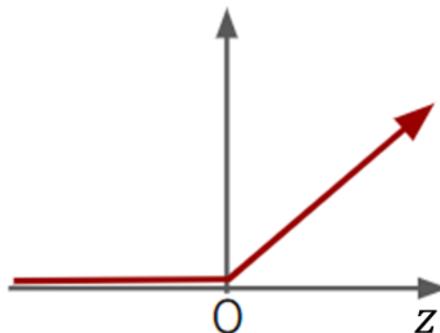


Figura 3.17 – Gráfico da função ReLU. Fonte: (Portilla, 2022).

A derivada da função ReLU é dada por:

$$\frac{d}{dz} \text{ReLU}(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0 \end{cases} \quad (3.22)$$

e a Figura 3.18 mostra graficamente a Equação 3.22 e é simplesmente a função degrau unitário.

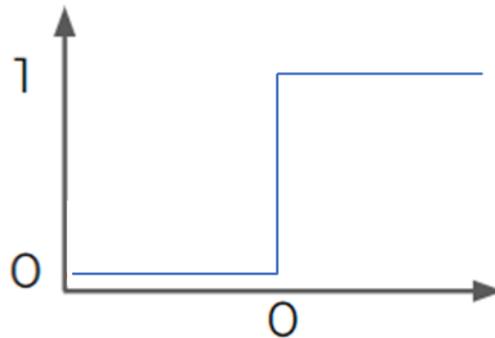


Figura 3.18 – Gráfico da derivada da função ReLU. Fonte: Elaborado pelo Autor.

Como visto na seção 3.2.2, calcular derivadas é essencial para otimizar os parâmetros dos modelos de ML, portanto a função ReLU é muito utilizada visto que é de fácil computação, assim como a sua derivada. Além disso, a função ReLU mitiga dois problemas que podem ocorrer durante a etapa da otimização, que é a Dissipação do Gradiente e a Explosão do Gradiente. Como a inicialização dos parâmetros da otimização é feita de forma aleatória, existe a possibilidade de que o gradiente calculado seja muito pequeno e o algoritmo encontre dificuldades em minimizar a função objetivo. Neste caso, caracteriza-se a Dissipação do Gradiente e por outro lado é possível que o gradiente calculado seja muito grande e prejudique a minimização da função objetivo.

Com a incorporação de funções não lineares nas redes neurais, elas tornam-se capazes de aproximar quaisquer funções reais com n parâmetros de entrada, mesmo para o caso de redes com somente uma camada oculta dependendo, portanto, do número de neurônios desta camada, sendo então consideradas aproximadores universais (CYBENKO, 1989). Entretanto, para muitos casos a modelagem e aproximação de funções complexas é mais viável utilizando redes profundas do que redes largas (ZHANG, LIPTON, LI, SMOLA, 2021).

3.3.2 Diferenciação automática e Retropropagação

Diferenciação automática é uma técnica eficiente para computar derivadas, principalmente no contexto de redes neurais profundas em que as quantidades de parâmetros de entrada, neurônios e camadas ocultas podem gerar matrizes de grandes dimensões. A diferenciação automática se baseia na representação de uma função como um grafo, no qual em cada nó é aplicada uma das seguintes operações elementares: adição, multiplicação e composição. Utilizando a regra da cadeia em cada nó, computa-se a derivada da função completa em relação aos seus parâmetros de entrada. Existem dois tipos de diferenciação automática (BAYDIN; PEARLMUTTER; RADUL; SISKIND, 2018):

- Modo Adiante, do inglês *Forward Mode*: as derivadas são calculadas no sentido da entrada até a saída aplicando a regra da cadeia em cada operação elementar tal que:

$$\dot{v}_i = \frac{\partial v_i}{\partial x_i}, \quad (3.23)$$

em que v_i é a variável intermediária e x_i é a variável de entrada. Para exemplificar o funcionamento da diferenciação automática, considera-se uma função da seguinte forma $f(x_1, x_2) = \ln(x_1) + x_1x_2 - \text{sen}(x_2)$. Essa função pode ser representada pelo grafo da Figura 3.19 e o algoritmo do Modo Adiante parte da esquerda para a direita, informando os valores de x_1 e x_2 e definindo $\dot{x}_1 = 1$ é possível calcular $\frac{\partial y}{\partial x_1}$ a partir da Equação 3.23:

Tabela 3-1 – Exemplo de diferenciação automática em Modo Adiante. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).

Avaliação de $f(\mathbf{x}_1, \mathbf{x}_2)$:	Avaliação de $\partial f(\mathbf{x}_1, \mathbf{x}_2)/\partial \mathbf{x}_1$:
$\mathbf{v}_{-1} = \mathbf{x}_1$	$\dot{\mathbf{v}}_{-1} = \dot{\mathbf{x}}_1 = \mathbf{1}$
$\mathbf{v}_0 = \mathbf{x}_2$	$\dot{\mathbf{v}}_0 = \dot{\mathbf{x}}_2 = \mathbf{0}$
$\mathbf{v}_1 = \ln(\mathbf{v}_{-1}) = \ln(\mathbf{x}_1)$	$\dot{v}_1 = \dot{v}_{-1}/\dot{v}_0 = 1/v_{-1} = 1/x_1$
$\mathbf{v}_2 = \mathbf{v}_{-1}\mathbf{v}_0 = \mathbf{x}_1\mathbf{x}_2$	$\dot{v}_2 = \dot{v}_{-1}v_0 + \dot{v}_0v_{-1} = \dot{x}_1x_2 + x_1\dot{x}_2 = x_2$
$\mathbf{v}_3 = \text{sen}(\mathbf{v}_0) = \text{sen}(\mathbf{x}_2)$	$\dot{v}_3 = \dot{v}_0 \cos(v_0) = 0$
$\mathbf{v}_4 = \mathbf{v}_1 + \mathbf{v}_2 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2$	$\dot{v}_4 = \dot{v}_1 + \dot{v}_2 = 1/x_1 + x_2$
$\mathbf{v}_5 = \mathbf{v}_4 - \mathbf{v}_3 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2 - \text{sen}(\mathbf{x}_2)$	$\dot{v}_5 = \dot{v}_4 - \dot{v}_3 = 1/x_1 + x_2$
$\mathbf{y} = \mathbf{v}_5 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2 - \text{sen}(\mathbf{x}_2)$	$\dot{y} = \dot{v}_5 = 1/x_1 + x_2$

E para computar $\partial y/\partial x_2$ basta definir $\dot{x}_2 = 1$ e repetir o processo apresentado na Tabela 3-1.

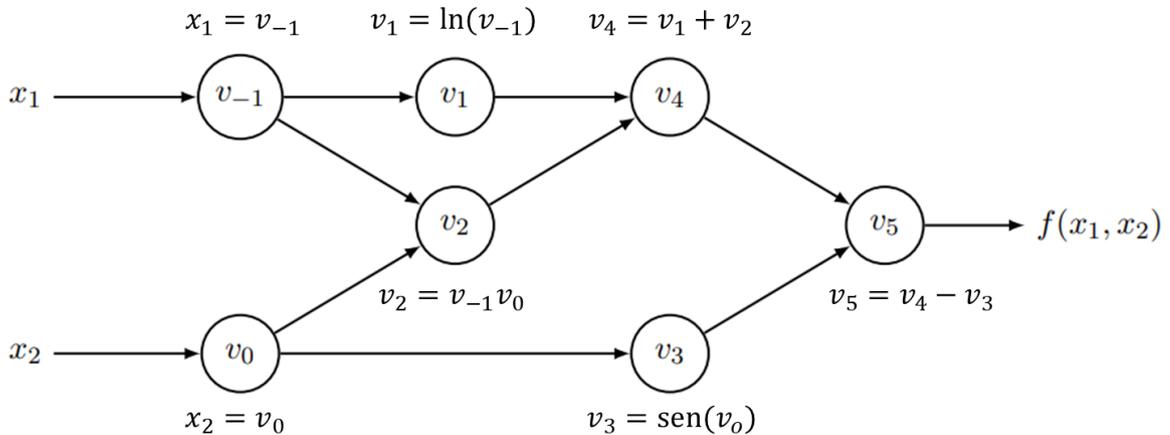


Figura 3.19 – Grafo que representa a função $f(x_1, x_2) = \ln(x_1) + x_1x_2 - \text{sen}(x_2)$. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).

- **Modo Reverso**, do inglês *Reverse Mode*: as derivadas são calculadas no sentido contrário, da saída para a entrada. O Modo Reverso é uma generalização do algoritmo de Retropropagação, no qual se calcula também a variável intermediária v_i com a seguinte expressão:

$$\bar{v}_i = \frac{\partial y_j}{\partial v_i}, \quad (3.24)$$

sendo y_j o parâmetro de saída. A Equação 3.24 representa a sensibilidade de y_j em função da variação de v_i . O Modo Reverso é composto de duas etapas, a primeira delas consiste em computar e armazenar todos os nós de $f(x_1, x_2)$, conforme o lado esquerdo da Tabela 3-1, identificando todas as relações que compõem a função. Na sequência, as derivadas \bar{v}_i são calculadas da saída para a entrada. A Tabela 3-2 exemplifica o método do Modo Reverso para a função da Figura 3.19, considerando que $\bar{v}_5 = \bar{y} = \frac{\partial y}{\partial y} = 1$. Calculando todas as variáveis intermediárias até chegar nas variáveis de entrada, nota-se que os resultados obtidos para $\partial y/\partial x_1$ são iguais comparando as Tabelas 3-1 e 3-2.

A grande vantagem desta abordagem é devido à eficiência em calcular os gradientes de y em função de cada parâmetro x , pois se obtém todas as derivadas em uma única propagação reversa. Portanto, como problemas de ML normalmente possuem um número

muito maior de parâmetros de entrada do que de saída, o método de diferenciação automático em Modo Reverso é mais adequado do que o Modo Adiante.

Tabela 3-2 – Exemplo de diferenciação automática em Modo Reverso. Fonte: (Baydin; Pearlmutter; Radul; Siskind, 2018).

Avaliação de $f(\mathbf{x}_1, \mathbf{x}_2)$:	Avaliação de $\partial f(\mathbf{x}_1, \mathbf{x}_2)/\partial \mathbf{x}$:
$\mathbf{v}_{-1} = \mathbf{x}_1$ $\mathbf{v}_0 = \mathbf{x}_2$	$\bar{\mathbf{v}}_5 = \bar{\mathbf{y}} = \mathbf{1}$
$\mathbf{v}_1 = \ln(\mathbf{v}_{-1}) = \ln(\mathbf{x}_1)$	$\bar{v}_4 = \bar{v}_5 \frac{\partial v_5}{\partial v_4} = \bar{v}_5(1) = 1$
$\mathbf{v}_2 = \mathbf{v}_{-1}\mathbf{v}_0 = \mathbf{x}_1\mathbf{x}_2$	$\bar{v}_3 = \bar{v}_5 \frac{\partial v_5}{\partial v_3} = \bar{v}_5(-1) = -1$
$\mathbf{v}_3 = \text{sen}(\mathbf{v}_0) = \text{sen}(\mathbf{x}_2)$	$\bar{v}_2 = \bar{v}_4 \frac{\partial v_4}{\partial v_2} = \bar{v}_4(1) = 1$
$\mathbf{v}_4 = \mathbf{v}_1 + \mathbf{v}_2 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2$	$\bar{v}_1 = \bar{v}_4 \frac{\partial v_4}{\partial v_1} = \bar{v}_4(1) = 1$
$\mathbf{v}_5 = \mathbf{v}_4 - \mathbf{v}_3 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2 - \text{sen}(\mathbf{x}_2)$	$\bar{v}_0 = \bar{v}_3 \frac{\partial v_3}{\partial v_0} + \bar{v}_2 \frac{\partial v_2}{\partial v_0} = -\cos(v_0) + v_{-1}$
	$\bar{v}_{-1} = \bar{v}_1 \frac{\partial v_1}{\partial v_{-1}} + \bar{v}_2 \frac{\partial v_2}{\partial v_{-1}} = 1/v_{-1} + v_0$
$\mathbf{y} = \mathbf{v}_5 = \ln(\mathbf{x}_1) + \mathbf{x}_1\mathbf{x}_2 - \text{sen}(\mathbf{x}_2)$	$\bar{v}_{-1} = \bar{x}_1 = 1/x_1 + x_2$ $\bar{v}_0 = \bar{x}_2 = x_1 - \cos(x_2)$

Como mencionado, o algoritmo de Retropropagação pode ser entendido como um caso especial da Diferenciação Automático em Modo Reverso. Neste sentido, será explicado como o algoritmo de Retropropagação é utilizado em ML para minimizar a função objetivo J ajustando os pesos w e vieses b de cada neurônio em cada camada da rede neural a partir da Equação 3.24, calculando as derivadas parciais $\partial J/\partial w$ e $\partial J/\partial b$. Para facilitar a compreensão, algumas notações serão padronizadas para referenciar os pesos, vieses e ativações na rede neural. O peso w_{jk}^l está associado ao k -ésimo neurônio na camada $l - 1$ conectado ao j -ésimo neurônio na camada l , conforme é exemplificado na Figura 3.20 com o peso w_{24}^3 que denota o peso do quarto neurônio na segunda camada conectado aos neurônios da terceira camada. Para os vieses b_j^l e ativações a_j^l os índices l e j denotam j -ésimo neurônio na l -ésima camada, conforme a Figura 3.20.

Considerando as notações utilizadas, é possível escrever uma expressão geral para as ativações da rede neural:

$$a_j^l = \sigma \left(\sum_k w_{jk}^l a_k^{l-1} + b_j^l \right) = \sigma(z_j^l), \quad (3.25)$$

na qual σ é a função de ativação, conforme as Equações 3.17, 3.19 e 3.21, $z_j^l = \sum_k w_{jk}^l a_k^{l-1} + b_j^l$ e a somatória é feita para todos os k neurônios até a camada $l - 1$. É possível vetorizar a Equação 3.25 e expressá-la em sua forma matricial, uma vez que a função de ativação σ é aplicada elemento a elemento:

$$a^l = \sigma(w^l * a^{l-1} + b^l) = \sigma(z^l), \quad (3.26)$$

com $z^l = w^l a^{l-1} + b^l$ e z^l representando as variáveis intermediárias da rede neural, que são os pesos ponderados dos neurônios na camada l . O símbolo $*$ denota multiplicação elemento a elemento.

Como o objetivo do algoritmo de Retropropagação é compreender como a variação dos pesos e vieses interferem na função objetivo, uma nova variável δ_j^l é introduzida de forma a relacionar esses parâmetros, que representa o erro no neurônio j e camada l , e é dado pela seguinte expressão:

$$\delta_j^l = \frac{\partial J}{\partial z_j^l}. \quad (3.27)$$

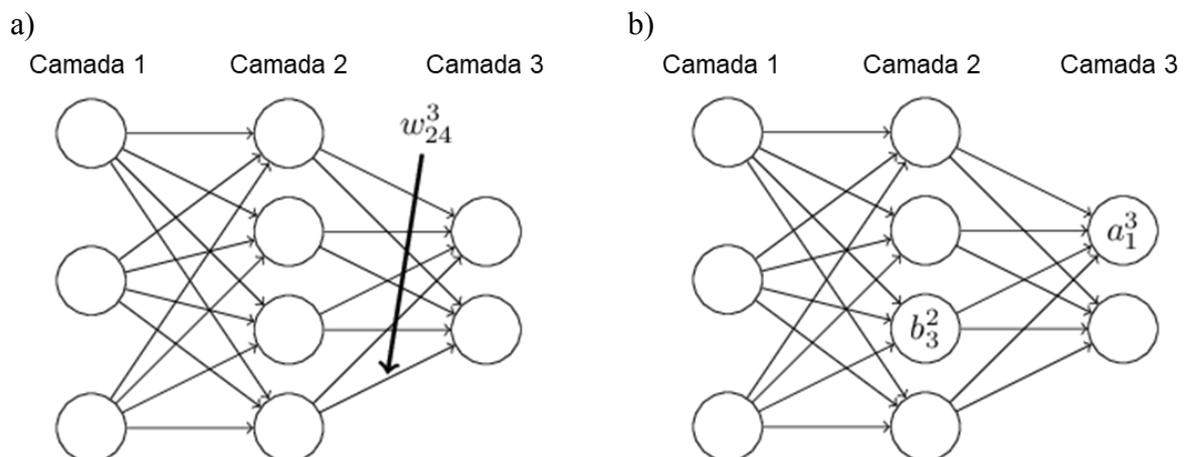


Figura 3.20 – Grafo de uma rede neural genérica evidenciando a notação a) do peso w_{jk}^l ; b) dos vieses b_j^l e funções de ativação a_j^l . Fonte (Nielsen, 2019).

O erro associado à camada de saída da rede neural pode ser calculado como:

$$\delta_j^l = \frac{\partial J}{\partial a_j^l} \sigma'(z_j^l), \quad (3.28)$$

em que $\sigma'(z_j^l)$ é a derivada da função de ativação, conforme as Equações 3.18, 3.20 e 3.22. O termo $\partial J / \partial a_j^l$ também é facilmente calculado, considerando por exemplo a Equação 3.4 e que \hat{y} nada mais é que o valor do neurônio da camada de saída, ou seja, a_j^l , a Equação 3.28 pode ser determinada e também pode ser escrita em sua forma matricial:

$$\delta^l = \nabla_a J * \sigma'(z^l). \quad (3.29)$$

na qual, o operador gradiente é aplicado em relação à a . Para calcular o erro das demais camadas da rede neural utiliza-se a seguinte expressão:

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) * \sigma'(z^l), \quad (3.30)$$

e a partir da qual a intuição do algoritmo de Retropropagação fica mais evidente, pois uma vez que o erro da camada l é calculado com a Equação 3.29, é possível calcular os demais erros das camadas precedentes com a Equação 3.30, determinando o erro δ^{l-1} , δ^{l-2} e assim sucessivamente até chegar no início da rede neural.

Sendo possível calcular o erro δ^l em qualquer camada, é possível então calcular a taxa de variação da função objetivo em função da variação dos vieses e pesos com as respectivas expressões:

$$\frac{\partial J}{\partial b_j^l} = \delta_j^l, \quad (3.31)$$

e

$$\frac{\partial J}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l, \quad (3.32)$$

que podem ser vetorizadas nas seguintes formas:

$$\frac{\partial J}{\partial b^l} = \delta^l, \quad (3.33)$$

e

$$\frac{\partial J}{\partial w^l} = a^{l-1} \delta^l. \quad (3.34)$$

As Equações 3.29, 3.30, 3.33 e 3.34 constituem o algoritmo de Retropropagação. Primeiramente, os pesos e vieses são iniciados de forma aleatória e é feita uma propagação em Modo Adiante para se determinar uma primeira estimativa, que é usada para determinar o erro da camada de saída da rede neural, Equação 3.29. Em seguida, este erro é retropropagado com a Equação 3.30, para calcular a sensibilidade dos pesos e vieses a partir das Equações 3.33 e 3.34, permitindo atualizar estes parâmetros na Equação 3.3, bastando informar a taxa de aprendizagem α . Dessa forma a função objetivo é iterativamente minimizada com os métodos já apresentados na seção 3.2.2.

4 MODELO SUBSTITUTO

O desenvolvimento de produtos de engenharia normalmente requer diversas análises e simulações para que o projeto seja bem-sucedido. Para tanto, são feitas análises de sensibilidade, que explora o comportamento do produto variando os parâmetros de projeto, de otimização, que buscam as características ótimas para maior eficiência e performance do produto, e de quantificação de incertezas, que visam calcular a confiabilidade do produto em função de algumas incertezas dos parâmetros de projeto (HWANG, MARTINS, 2018; GUO, 2020). Conduzir todas essas análises pode levar um tempo considerável, visto que uma grande quantidade de simulações deve ser realizada, combinando todas as variações de parâmetros de projeto, e dependendo da simulação pode ser computacionalmente custoso para resolver os modelos. Modelos substitutos, também conhecidos como metamodelos, tornam-se uma abordagem que viabiliza essa grande quantidade de análises por meio de um modelo estatístico, que aproxima as soluções obtidas pelas simulações de forma acurada. Portanto, este modelo com base nos dados pode substituir a necessidade de realizar simulações computacionais.

Os modelos substitutos funcionam exatamente como o aprendizado supervisionado, logo os mesmos algoritmos e métodos podem ser utilizados para construir esses modelos, que aproximam a forma de uma função qualquer y com base em dados gerando uma nova função \hat{y} , capaz de gerar previsões acuradas. Como mencionado nos capítulos anteriores, o objetivo deste trabalho é construir um modelo substituto baseado em redes neurais profundas capaz de prever as propriedades equivalentes de estruturas periódicas, DLF, densidade modal e RLF, obtidas pelo método de FE.

Embora a escolha das redes neurais como modelo de ML neste trabalho seja arbitrária, os estudos relacionados a esse tema e aplicados à problemas vibroacústicos tendem também a utilizar modelos baseados em redes neurais (SACK, ÅBOM, 2020; YUCESAN *et al*, 2021; ALGUACIL *et al*, 2021; THADA *et al*, 2021; MITUSCH *et al*, 2021), apesar destes trabalhos também não apresentarem uma justificativa técnica para essa adoção.

4.1 DADOS DE TREINAMENTO E TESTE E ESTRUTURA DA REDE NEURAL

Os dados utilizados para o processo de treinamento e teste da rede neural foram obtidos pela caracterização dos materiais viscoelástico e pelas soluções dos modelos de FE

apresentados no capítulo 2. A Tabela 4-1 mostra a relação de todas as condições consideradas durante a etapa de caracterização, na qual os ensaios foram feitos para oito materiais aplicados em vigas base uniformes de alumínio de três espessuras distintas, 0,8 mm, 1,2 mm e 2,0 mm, em diferentes temperaturas de operação, variando de -30°C a 20°C, para a faixa de frequência de 200 a 8000 Hz.

Tabela 4-1 – Relação das condições para os ensaios de caracterização dos materiais viscoelásticos.

Amostras	Viga base [mm]	Temperatura [°C]	Frequência [Hz]
Material 1	0,8	-30	200
Material 2	1,2	-20	8000
Material 3	2,0	-10	
Material 4		0	
Material 5		10	
Material 6		20	
Material 7			
Material 8			

Em se tratando de conforto acústico dentro da cabine da aeronave, considera-se a fase de altitude de cruzeiro, uma vez que é a mais longa durante um trajeto de voo, expondo passageiros e tripulação a níveis de ruído por um período significativo. A altitude de voo pode variar de 33.000 a 42.000 pés e a temperatura externa pode chegar a -40°C. Portanto foi considerada a condição de teste em -30°C para treinamento da rede neural.

As Tabela 4-2 e Tabela 4-3 apresentam as densidades e espessuras totais de cada material para cada viga base em que material viscoelástico foi aplicado, respectivamente.

Tabela 4-2 – Densidade em [kg/m³] de cada material para cada viga base.

Amostras	Viga base 0,8 mm	Viga base 1,2 mm	Viga base 2,0 mm
Material 1	2050	2079	2210
Material 2	2054	2086	2277
Material 3	1044	1155	1417
Material 4	1830	1897	2026
Material 5	1030	1165	1405
Material 6	2133	2165	2292
Material 7	1392	1496	1761
Material 8	1979	2044	2234

Ao final do processo de caracterização, foi obtida uma curva de DLF e outra de módulo de elasticidade para cada material considerando as variações de espessura da viga base.

Tabela 4-3 - Espessura total em [mm] de cada material para cada viga base.

Amostras	Viga base 0,8 mm	Viga base 1,2 mm	Viga base 2,0 mm
Material 1	2,4	2,8	3,6
Material 2	1,8	2,2	3,0
Material 3	3,8	4,2	5,0
Material 4	3,0	3,4	4,2
Material 5	3,8	5,2	5,0
Material 6	2,4	2,8	3,6
Material 7	2,7	3,1	3,9
Material 8	2,0	2,4	3,2

Conforme visto na seção 2.3 e 2.4, as propriedades dos materiais e as curvas de DLF e módulo de elasticidade foram utilizadas para obter as propriedades equivalentes da estrutura periódica, considerando as periodicidades e faixa de frequência, conforme a Tabela 4-4.

Tabela 4-4 – Periodicidade e frequência de análise para solução dos modelos de FE.

Amostras	Periodicidade	Frequência [Hz]
Material 1	2x2	500
Material 2	2x3	630
Material 3	2x4	800
Material 4	2x5	1000
Material 5		1250
Material 6		1600
Material 7		2000
Material 8		

O conjunto de dados é composto por 672 observações, as quais foram separadas em conjunto de treinamento, conjunto de validação e conjunto de teste, na proporção 80/20, ou seja, 80% dos dados foram utilizados para treinamento e 20% para teste, e 15% dos dados de treinamento foram separados para o conjunto de validação, logo tem-se 446 amostras para treinamento, 86 para validação e 140 para teste. A separação dos conjuntos de treinamento e teste foi feita por material e periodicidade, de forma a garantir uma distribuição uniforme da frequência durante o processo de treinamento da rede neural. Além disso, um material foi separado para realizar uma análise de sensibilidade, que será detalhada na seção 4.3 deste capítulo, ou seja, este material não compõe nenhum dos conjuntos de treinamento, validação ou teste.

4.1.1 Arquitetura da rede neural

A estrutura da rede neural tem seis neurônios na camada de entrada, que são as propriedades físicas e mecânicas, a periodicidade da célula unitária e as bandas de frequência, e três neurônios na camada de saída, que são as propriedades equivalentes da estrutura periódica, conforme a Figura 4.1.

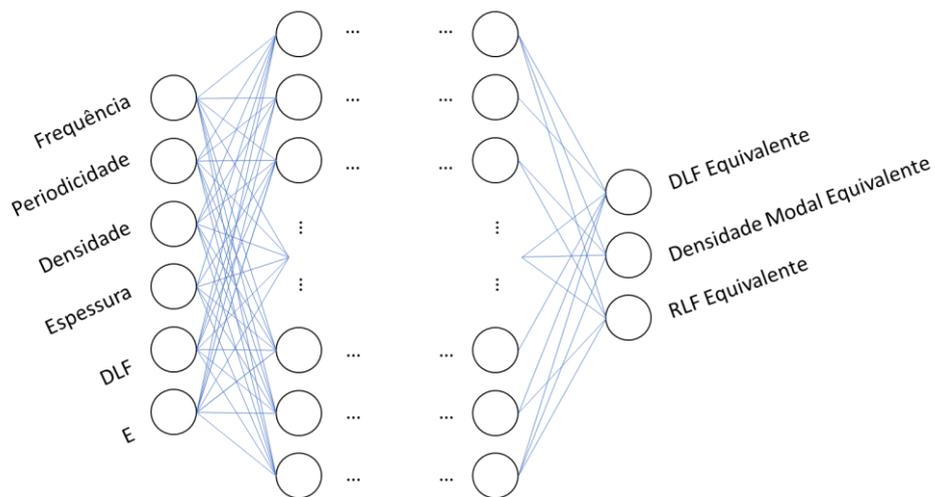


Figura 4.1 – Estrutura da rede neural para construção do modelo substituto. Fonte: Elaborado pelo Autor

É necessário também definir alguns outros parâmetros, denominados hiperparâmetros, antes de realizar o treinamento das redes neurais. Os hiperparâmetros são aqueles relacionados ao processo de aprendizado, como por exemplo a taxa de aprendizagem, o próprio algoritmo de otimização e até mesmo a quantidade de camadas ocultas e neurônios em cada uma delas e as próprias funções de ativação. O ajuste do hiperparâmetros pode ser realizado utilizando uma técnica chamada *Grid Search*, na qual define-se todos os parâmetros de interesse a serem ajustados e os valores a serem testados durante o processo de treinamento do modelo ML. Nota-se, no entanto, que o tamanho do espaço de busca dos hiperparâmetros ótimos é grande, logo o custo computacional para treinar o modelo pode aumentar significativamente. Portanto, inicialmente foi considerada uma rede neural com quatro camadas ocultas com vinte neurônios em cada uma delas.

Para as funções de ativação optou-se por utilizar a função ReLU em todas as camadas por dois motivos principais, a capacidade de evitar os problemas de Dissipação e Explosão e Gradiente e funcionar como um método de regularização da rede neural, evitando

o *overfitting*. Existe uma técnica de regularização denominada *Dropout*, na qual se associa uma probabilidade de desconectar alguns neurônios da rede, minimizando a chance de ocorrer o *overfitting* que pode ser gerado pela dependência criada entre os padrões de conexão dos neurônios. A função ReLU, dada pela Equação 3.21, emula o efeito do *Dropout*, uma vez que seu valor é 0 para $z \leq 0$, logo é o mesmo que desfazer a conexão entre os neurônios em que a função ReLU é igual 0.

A função objetivo utilizada para calcular o erro entre os valores reais das propriedades equivalentes e os valores preditos pelo modelo substituto foi o Erro Quadrático Médio, Equação 3.4, e o algoritmo escolhido para realizar a etapa de otimização foi o *Adam* com uma taxa de aprendizagem padrão de 10^{-3} .

Além disso, alguns outros parâmetros são informados para definir o processo de treinamento da rede neural. O primeiro deles é a quantidade de modelos que serão treinados, pois uma vez que se trata de uma otimização estocástica é natural que cada modelo seja ligeiramente diferente e posteriormente é possível determinar uma dispersão e uma confiabilidade para os modelos testados. O segundo parâmetro é o número de amostras do conjunto de treinamento que é utilizado para calcular o gradiente em cada época, ou seja, o tamanho do lote. O terceiro parâmetro informado é o número de épocas da etapa de treinamento, ou seja, quantas iterações serão realizadas para minimizar a função objetivo. E por fim, o último parâmetro definido é chamado de paciência, do inglês *patience*, que é uma forma de aferir a convergência do modelo e cessar o treinamento antes de atingir o número de épocas estipuladas. O uso do parâmetro paciência é interessante, pois é possível deixar um número de épocas elevado para evitar o *underfitting* e cessar o treinamento, caso o erro se estabilize ao longo das épocas, evitando a possibilidade de *overfitting*. Portanto, foi definido que o número de modelos a serem treinados era 30, com lote de 64 amostras por época, 10.000 épocas e um *patience* de 100 épocas.

4.2 REGRESSÃO DAS PROPRIEDADES EQUIVALENTES DE ESTRUTURAS PERIÓDICAS

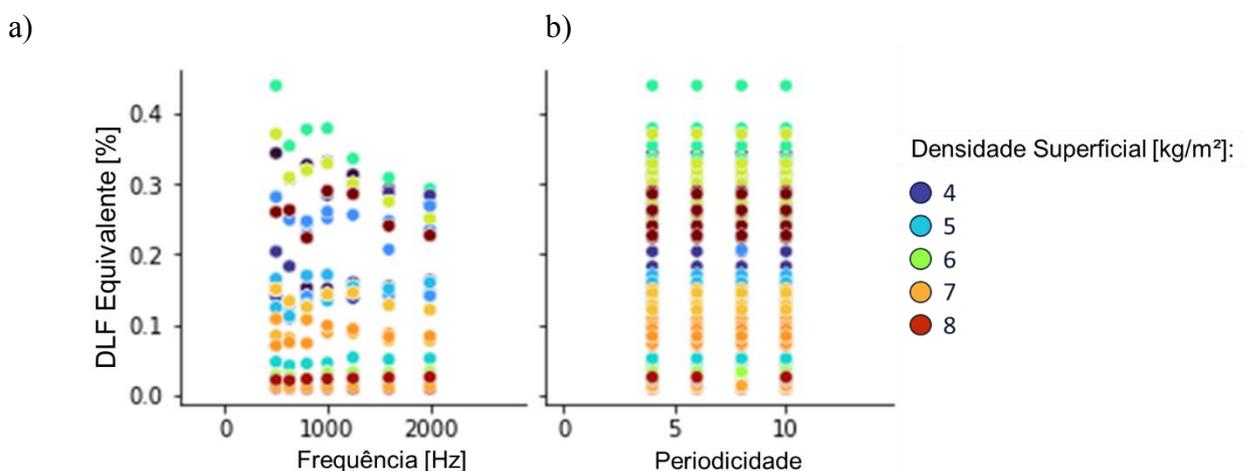
Após consolidar e estruturar a base de dados, foram separados os conjuntos de treinamento e teste e é uma boa prática no desenvolvimento de modelos de ML aplicar um método para escalar os atributos. Neste trabalho, foi utilizado método *MinMaxScaler*, que transforma os dados para um determinado intervalo, neste caso entre 0 e 1. O método de

escala extrai algumas características, como valores mínimo e máximo do conjunto de treinamento, transforma os dados e utiliza as mesmas características para escalar o conjunto de teste. Essa transformação é importante para facilitar o aprendizado e determinação dos pesos e vieses, pois grandes diferenças em termos de ordens de grandeza entre os atributos podem gerar instabilidades para encontrar os pesos e vieses ótimos.

Inicialmente, as propriedades equivalentes da célula unitária foram regredidas individualmente para compreender o comportamento da rede neural e então, posteriormente, um novo modelo capaz de fazer a regressão simultânea dos três parâmetros foi treinado, conforme é apresentado nas próximas seções deste capítulo.

4.2.1 Regressão do Fator de Amortecimento Equivalente

Primeiramente, uma visualização do DLF Equivalente em função dos parâmetros de entrada da rede neural foi gerada, conforme a Figura 4.2, para realizar uma análise exploratória dos dados de forma gráfica. Nota-se que há uma alta correlação entre o DLF do material viscoelástico e o DLF equivalente da célula unitária, o que é esperado conforme a Equação 2.13 e a área de cobertura do material viscoelástico, mostrada na Figura 2.4. Também é possível identificar que de fato não há dependência da periodicidade para o DLF Equivalente da célula unitária, Equação 2.10 e que a visualização em função da frequência é um reflexo do DLF experimental, ou seja, os materiais com maiores amortecimentos encontram-se na parte superior da Figura 4.2. Para o módulo de elasticidade nenhuma correlação evidente pôde ser identificada.



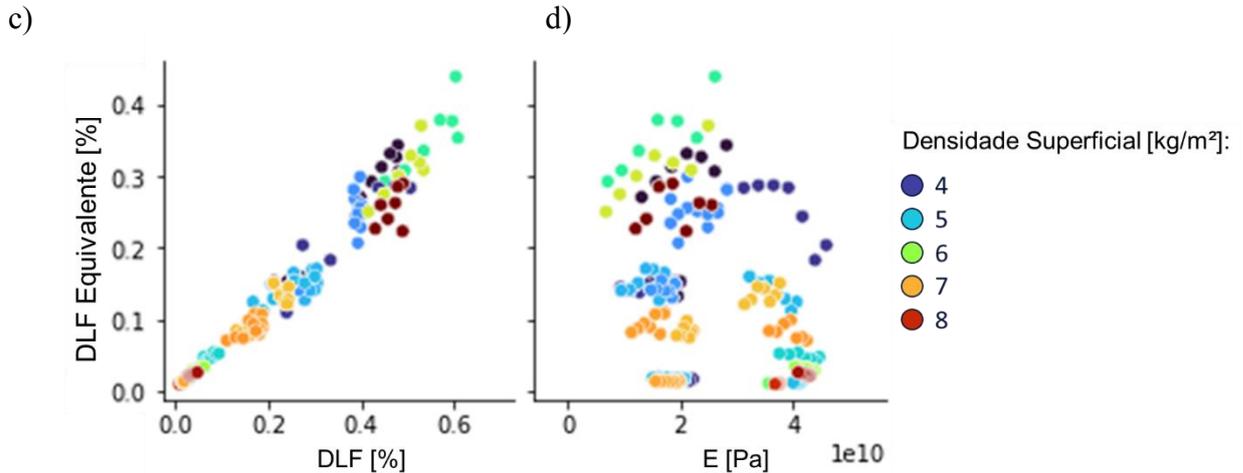


Figura 4.2 – Visualização par a par do DLF Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.

Além da visualização da figura anterior, é interessante avaliar também a distribuição dos dados de DLF Equivalente utilizando um histograma, conforme a Figura 4.3. Nota-se uma grande concentração de amostras com amortecimento baixo e poucas amostras com amortecimento alto, logo há um desbalanço considerável na distribuição dos dados, o que pode impactar na capacidade de predição da rede neural para casos de poucas ocorrências.

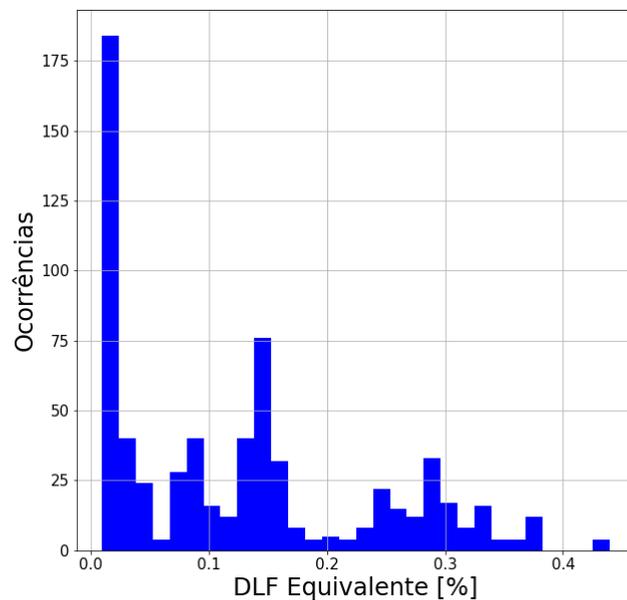


Figura 4.3 – Histograma da distribuição dos valores de DLF Equivalente.

Na sequência, as redes neurais foram treinadas considerando a arquitetura e parâmetros apresentados na seção 4.1.1. O tempo médio para treinar os 30 modelos foi de aproximadamente 30 minutos e os resultados das redes neurais são mostrados na Figura 4.4,

na qual são comparados os valores de DLF Equivalentes reais, ou seja, obtidos pelas soluções dos modelos de FE, e os valores preditos pelas redes neurais. Lembrando que os valores apresentados na Figura 4.4 são referentes ao conjunto de teste, dados não “vistos” pelas redes neurais durante a etapa de treinamento. Para mensurar a performance dos modelos resultantes, utilizou-se a métrica *Explained Variance Score*, por ser a mais intuitiva em termos de interpretação dos resultados. O valor obtido para essa métrica foi de $99,01 \pm 0,38\%$, indicando uma performance excelente das redes neurais em prever os valores de DLF Equivalente.

Mesmo com a alta acurácia dos modelos, uma arquitetura diferente foi testada considerando que a função de ativação entre a última camada oculta e a camada de saída foi alterada de ReLU para Sigmoid para avaliar o comportamento das redes neurais. A Figura 4.5 apresenta os resultados obtidos e nota-se o mesmo comportamento, indicando boa performance dos modelos e em termos do *Explained Variance Score* foi de em média $99,15 \pm 0,30\%$, indicando uma pequena melhora na acurácia e um leve decréscimo da dispersão dos modelos.

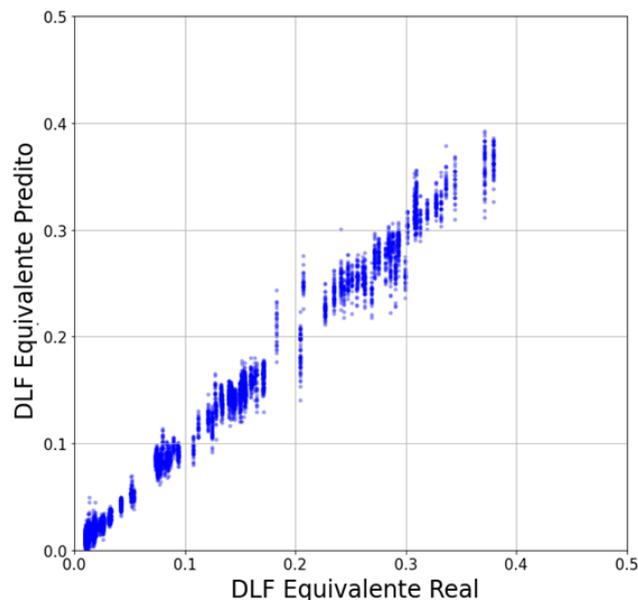


Figura 4.4 – Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor.

Para fins de comparação, dois modelos mais simples de regressão linear e regressão polinomial de segundo grau foram treinados com o intuito de avaliar a capacidade de prever o DLF Equivalente, conforme a Figura 4.6. Para o modelo de regressão linear, o *Explained Variance Score* foi de $98,27\%$ e para o modelo polinomial de segundo grau o *Explained*

Variance Score foi de 99,03%. Para ambos os modelos os resultados foram comparáveis à performance das redes neurais, indicando que um modelo mais simples poderia prever com acurácia os valores de DLF Equivalente e ainda inferir a significância de cada parâmetro de entrada, enquanto essa interpretabilidade torna-se uma “caixa preta” para as redes neurais.

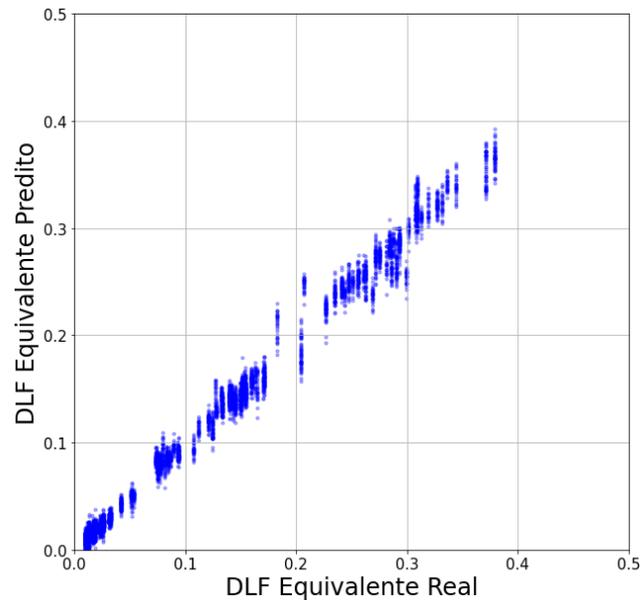


Figura 4.5 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

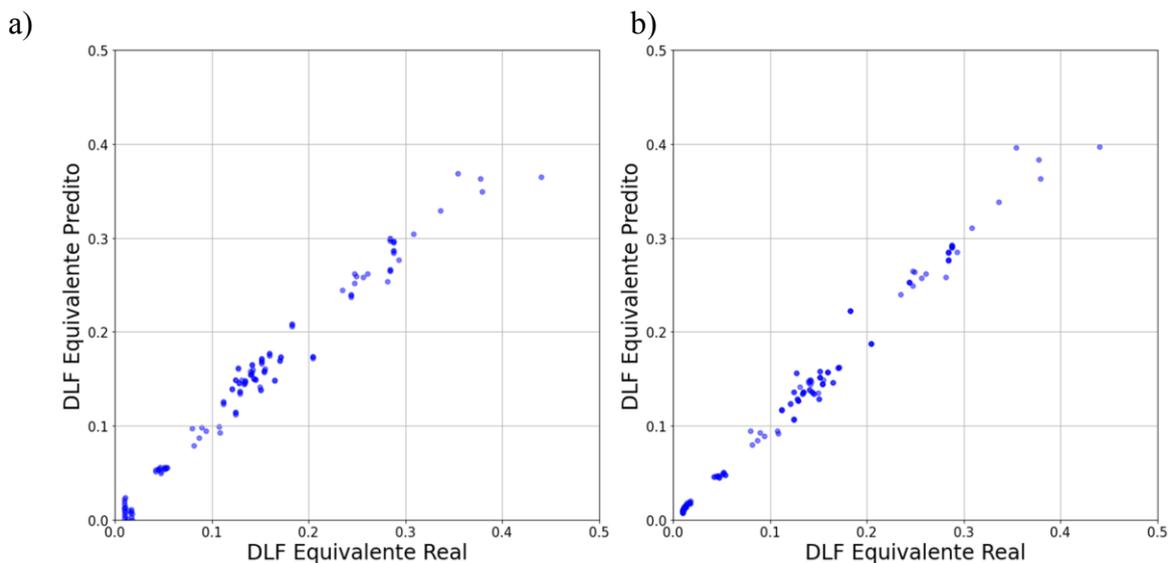


Figura 4.6 - Modelo de regressão a) linear; b) polinomial de segundo grau para o DLF Equivalente. Fonte: Elaborado pelo Autor.

Para todos os casos, seja com o modelo baseado em redes neurais ou mesmo a regressão polinomial, o desbalanço dos dados de DLF Equivalente não prejudicaram a performance do modelo. Nota-se que para algumas faixas de valores em torno de 0,2 e 0,4 em que há poucas amostras também há um desvio ligeiramente maior entre os valores reais e os valores preditos.

4.2.2 Regressão da Densidade Modal Equivalente

Novamente, a análise exploratória dos dados é feita visualizando a Densidade Modal Equivalente em função dos parâmetros de entrada, como é mostrado na Figura 4.7. Naturalmente, a quantidade de modos presentes em uma determinada banda de frequência aumenta conforme se avança para faixas de altas frequências. A periodicidade também é um fator que contribui para a Densidade Modal Equivalente, conforme a Equação 2.10 e a Figura 4.7, ou seja, para células unitárias com maiores periodicidades, que representam áreas efetivas maiores, a quantidade de modos por banda de frequência aumenta em um fator proporcional à $N_x \times N_y$. Outra observação que é identificada é a relação entre o módulo de elasticidade e a Densidade Modal Equivalente. Para valores menores do módulo de elasticidade, ou seja, para estruturas mais flexíveis a densidade modal é maior, pois os modos são deslocados para região de baixas frequências, e para estruturas mais rígidas a densidade modal é menor, uma vez que os modos são deslocados para região de altas frequências. De maneira simples, é possível entender esse comportamento pela expressão $\omega = \sqrt{k/m}$, na qual k é a rigidez do sistema e m sua massa. Já para o DLF experimental nenhum padrão evidente pôde ser identificado. Por fim, a Figura 4.8 apresenta a distribuição dos dados de Densidade Modal Equivalente e é possível notar uma concentração assimétrica em torno de 0,01 modos/Hz.

Em seguida, um novo conjunto de redes neurais foi treinado para prever os valores de Densidade Modal Equivalente, considerando a arquitetura inicial de quatro camadas ocultas e funções de ativação ReLU e os resultados obtidos são apresentados na Figura 4.9. Como os padrões e tendências dos parâmetros de entrada das redes neurais são menos evidentes para o caso da Densidade Modal Equivalente, nota-se uma dispersão maior dos resultados das redes neurais se comparadas àquelas que predisseram o DLF Equivalente, em que havia uma alta correlação com o DLF experimental. Em termos da métrica de performance, o *Explained Variance Score* foi de em média $78,17 \pm 11,85\%$. Para modelos de

ML em geral uma acurácia de 78% ainda se considera um bom modelo, apesar da queda significativa em comparação ao DLF.

Considerando a segunda arquitetura de rede neural, com a função de ativação Sigmoid da última camada oculta para a camada de saída, uma nova rodada de treinamentos foi efetuada, os resultados obtidos são mostrados na Figura 4.10 e são similares àqueles da Figura 4.9. O *Explained Variance Score* foi de em média $85,72 \pm 10,41\%$.

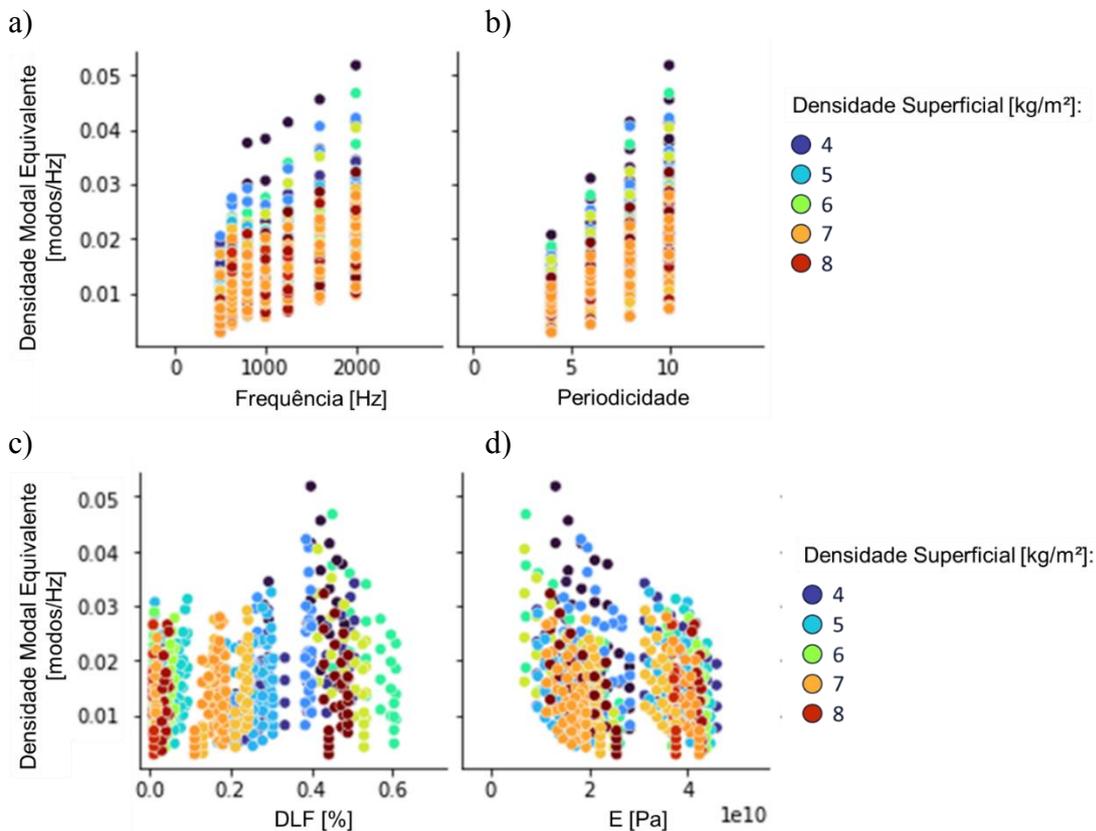


Figura 4.7 - Visualização par a par da Densidade Modal Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.

Por fim, novamente dois modelos simples de regressão linear e polinomial de segundo grau foram treinados e os resultados são apresentados na Figura 4.11. Para o modelo de regressão linear, o *Explained Variance Score* foi de 83,04%, acurácia bem abaixo se comparada às redes neurais. Já para o modelo de regressão polinomial de segundo grau, o *Explained Variance Score* foi de 91,10%, logo um modelo um pouco mais complexo foi capaz de identificar os padrões para prever com boa acurácia os valores de Densidade Modal Equivalente.

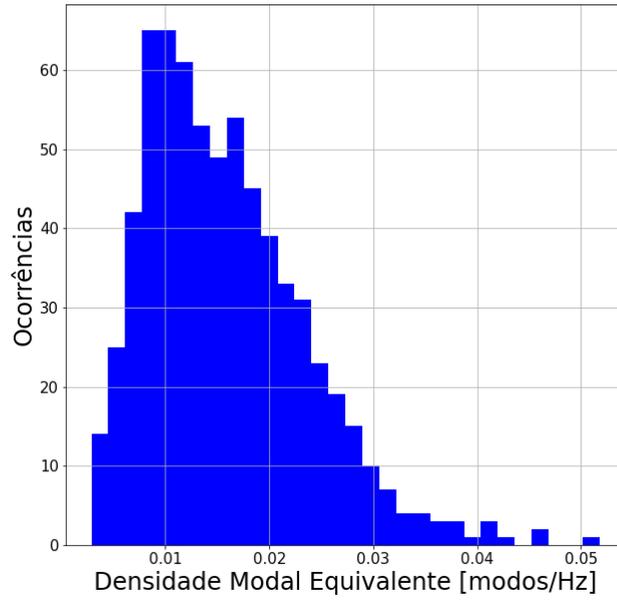


Figura 4.8 – Histograma da distribuição dos valores de Densidade Modal Equivalente.

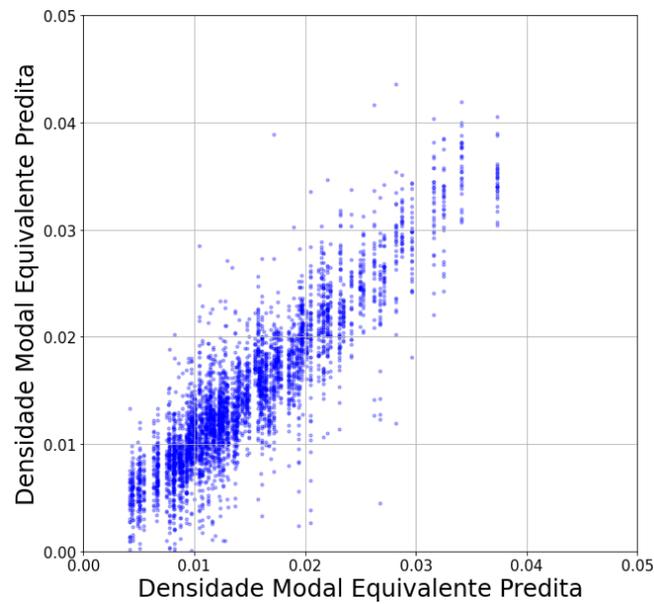


Figura 4.9- Comparação entre os valores de Densidade Modal Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor.

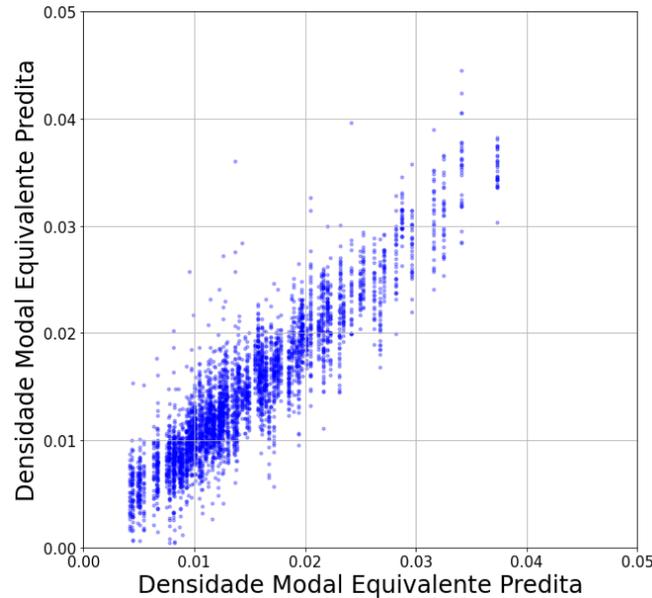


Figura 4.10 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

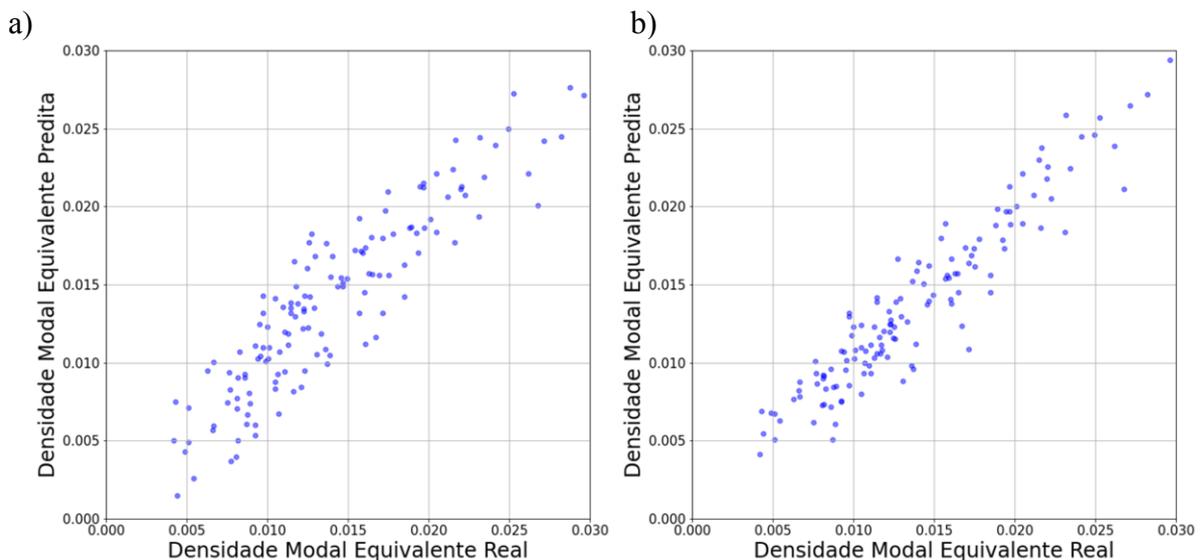


Figura 4.11 - Modelo de regressão a) linear; b) polinomial de segundo grau para a Densidade Modal Equivalente. Fonte: Elaborado pelo Autor.

4.2.3 Regressão do Fator de Perda por Radiação Equivalente

A última propriedade avaliada foi o RLF Equivalente e, da mesma forma, a análise exploratória dos dados foi feita par a par, como mostra a Figura 4.12. Conforme a Equação 2.16, verifica-se que o RLF Equivalente é inversamente proporcional à frequência e à periodicidade e isso pode ser facilmente visualizado nas Figuras 4.12a e 4.12b. A Equação

2.16 indica que o RLF Equivalente é função do campo de deslocamento da célula unitária, logo o amortecimento e o módulo de elasticidade diretamente na resposta da estrutura, porém analisando as Figuras 4.12c e 4.12d é difícil identificar um padrão evidente de influência desses parâmetros sobre o RLF Equivalente. Um possível padrão observado é que o aumento do DLF experimental reduz a amplitude de deslocamento da estrutura e que uma estrutura mais rígida tende a ter uma perda por radiação maior, porém identificam-se amostras nas Figuras 4.12c e 4.12d que não sustentam tal padrão. Esta avaliação pode indicar uma dificuldade maior da rede neural reconhecer as relações entre os parâmetros de entrada e saída e com isso prever de forma acurada os valores de RLF Equivalente. Além disso, a distribuição dos dados de RLF Equivalente é apresentada na Figura 4.13 e nota-se novamente uma concentração assimétrica, neste caso em torno de 0,0005.

As Figuras 4.14 e 4.15 apresentam os resultados obtidos para predição do RLF Equivalente considerando a arquitetura inicial com somente funções de ativação ReLU e a segunda arquitetura com a inclusão da função *Sigmoid*, respectivamente. Em ambos os casos as redes neurais não foram capazes de prever o RLF Equivalente, havendo praticamente uma desconexão entre os valores reais e preditos, visto a grande dispersão de pontos. Em termos da métrica de performance, os modelos apresentaram valores negativos para o *Explained Variance Score*, pois diversas predições resultaram em valores menores que 0, o que não está de acordo com a própria formulação da Equação 2.16.

Seguindo a mesma lógica das regressões das demais propriedades equivalentes, um modelo de regressão linear e outro polinomial de segundo grau foram treinados e a Figura 4.16 apresenta os resultados obtidos. A regressão linear, mesmo com baixa acurácia com *Explained Variance Score* de 52,96%, revela um comportamento não linear dos dados e o uso de um modelo polinomial apresentou uma melhora da acurácia, com *Explained Variance Score* de 62,03%. Apesar das duas métricas ainda indicarem uma performance ruim do modelo, nota-se que foi bem superior se comparada às redes neurais e como visto para casos anteriores as redes neurais se sobressaíram em relação aos modelos mais simples ou pelo menos a mesma acurácia foi obtida, como foi o caso do modelo polinomial de segundo grau para a Densidade Modal Equivalente. Logo, espera-se uma performance melhor das redes neurais para prever o RLF Equivalente.

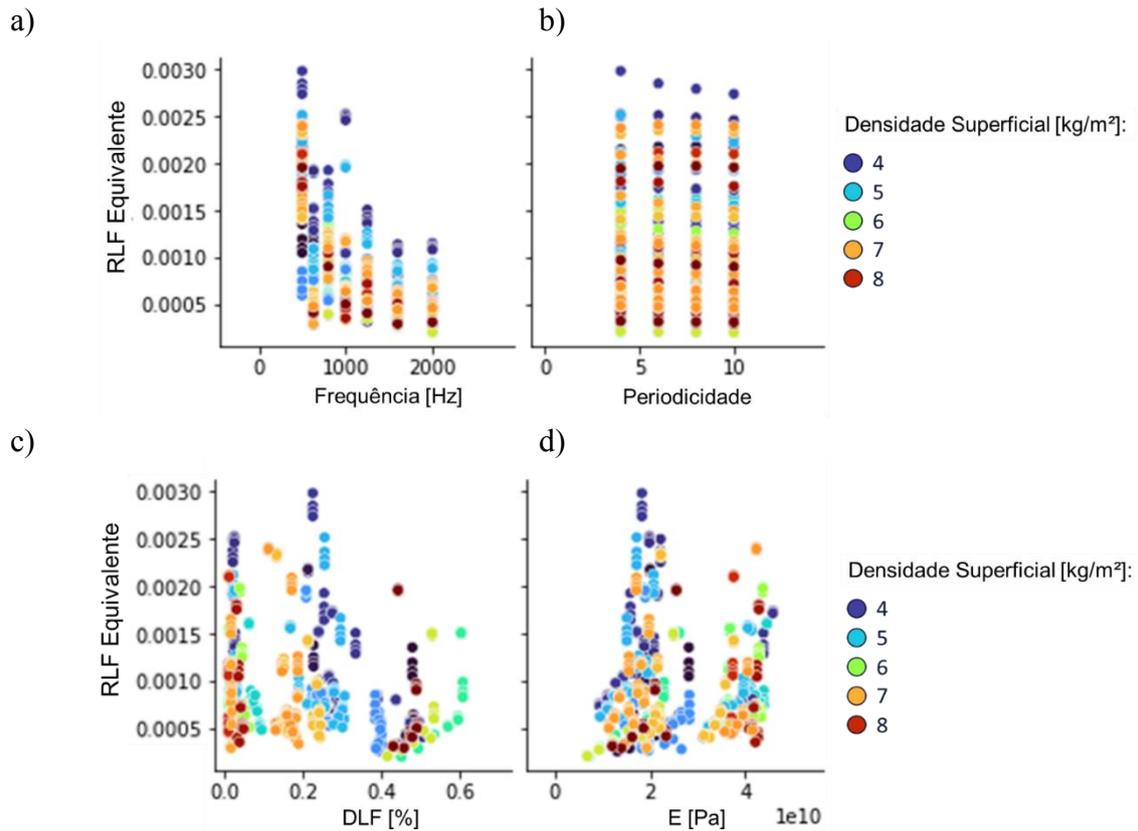


Figura 4.12 - Visualização par a par do RLF Equivalente em função dos parâmetros de entrada da rede neural. Fonte: Elaborado pelo Autor.

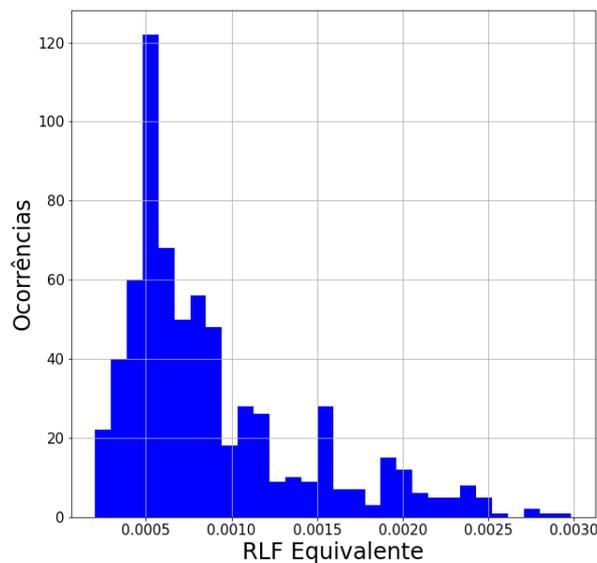


Figura 4.13 – Histograma da distribuição dos valores de RLF Equivalente.

O motivo exato pelo qual as redes neurais apresentaram uma acurácia ruim não é claro, porém uma possível hipótese está relacionada aos valores do RLF Equivalente e a função objetivo, que para todos os casos é o Erro Quadrático Médio. Como numericamente os

valores do RLF Equivalente são pequenos e possuem uma variação de uma ordem de grandeza, entre 2×10^{-4} a 3×10^{-3} , aproximadamente, ao calcular a função objetivo, em que é necessário elevar ao quadrado a diferença entre o valor predito e o valor real, o resultado é um valor muito pequeno, que impacta no processo de minimização da função objetivo, calculando valores de pesos e vieses quase aleatórios para estimar o RLF Equivalente.

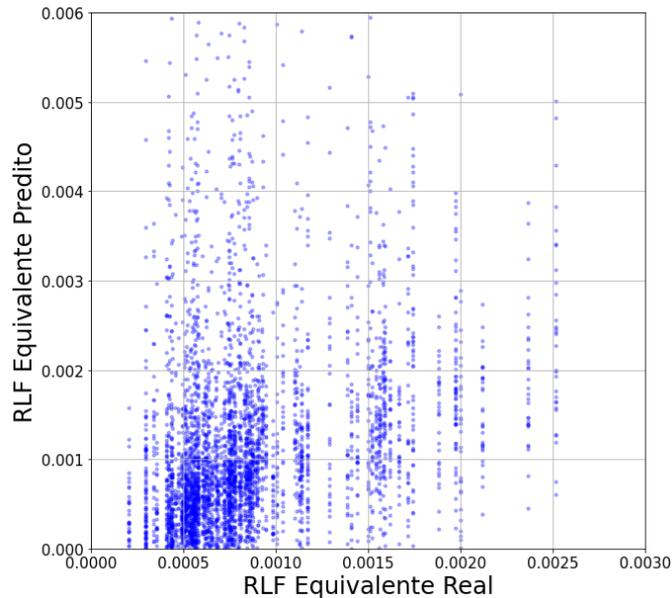


Figura 4.14 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU. Fonte: Elaborado pelo Autor.

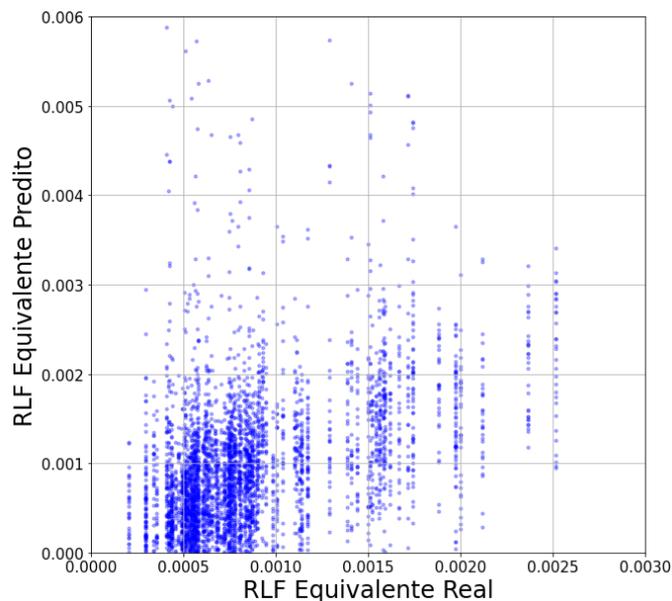


Figura 4.15 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

Para testar a hipótese mencionada, adotou-se a estratégia de aplicar o mesmo método de escala, que é utilizado sobre os atributos do conjunto de dados, conforme mencionado no início da seção 4.2, e aplicar também sobre os rótulos, neste caso o RLF Equivalente. Esta abordagem permite transformar os valores da propriedade equivalente e utilizar esses novos valores durante o processo de treinamento e posteriormente aplicar a transformação inversa para obter o valor original do RLF Equivalente predito, como é mostrado na Figura 4.17. A Tabela 4-5 apresenta alguns valores de exemplo do RLF Equivalente transformados.

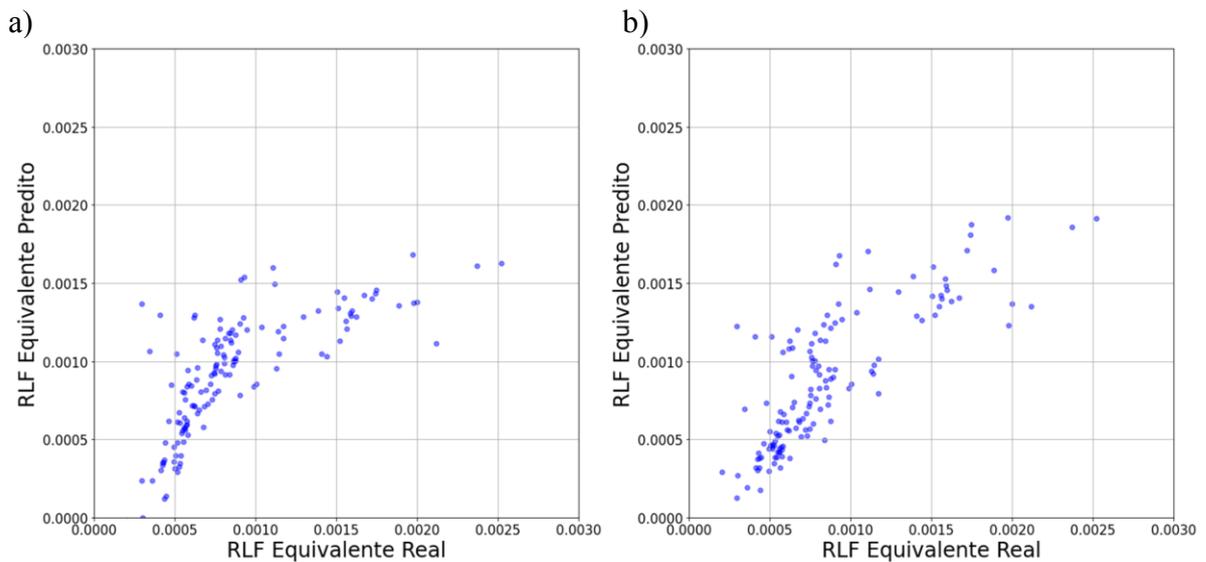


Figura 4.16 - Modelo de regressão a) linear; b) polinomial de segundo grau para o RLF Equivalente. Fonte: Elaborado pelo Autor.

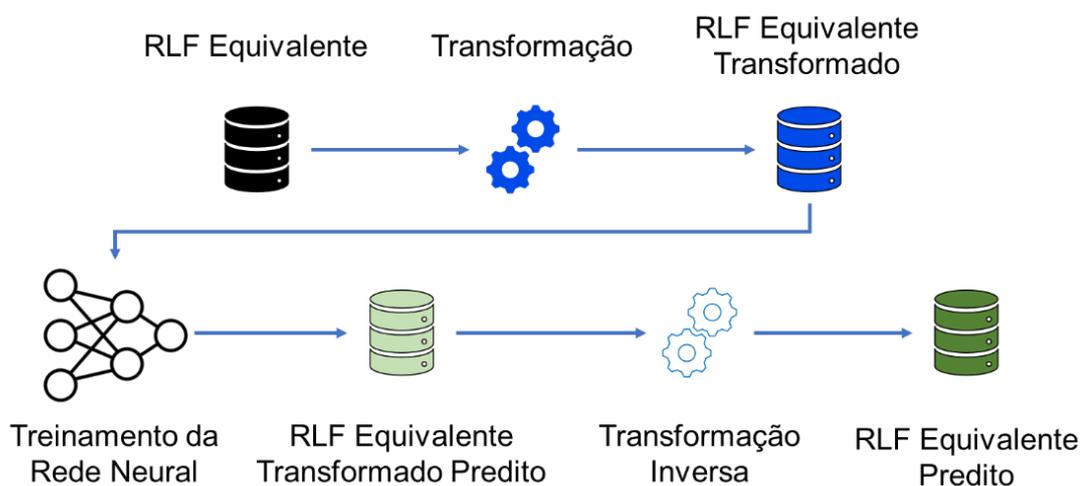


Figura 4.17 - Representação da estratégia de transformação dos dados de RLF Equivalente. Fonte: Elaborado pelo Autor.

Tabela 4-5 - Exemplo de alguns valores de RLF Equivalente transformados.

RLF Equivalente	RLF Equivalente Transformado
0,00083065	0,225909
0,00125216	0,377365
0,000419854	0,0783025
0,000472689	0,0972868
0,000910158	0,254478

Analisando os dados da Tabela 4-5, observa-se que valores transformados estão 3 ordens de grandeza maiores e a diferença dos valores de RLF Equivalente transformado entre si também são maiores, essas características podem facilitar o aprendizado das redes neurais. A Figura 4.18 apresenta os resultados obtidos considerando a etapa de transformação do RLF Equivalente e a arquitetura inicial, e a melhora da performance da rede neural é significativa, obtendo-se um *Explained Variance Score* de $93,18 \pm 5,50\%$, muito superior àquele obtido na Figura 4.14 de $47,96\%$, indicando evidências que reforçam a hipótese formulada para a dificuldade de aprendizado da rede neural com os dados brutos de RLF Equivalente.

Em seguida, a segunda arquitetura de rede neural, com a função de ativação *Sigmoid* da última camada oculta para a camada de saída, foi avaliada e os resultados obtidos são mostrados na Figura 4.19. Novamente, uma grande melhora em comparação aos resultados da Figura 4.15, com um *Explained Variance Score* de $95,52 \pm 4,01\%$.

Com as análises realizadas para a regressão individual de cada propriedade equivalente, alguns aprendizados puderam ser adquiridos para construir um modelo capaz de prever as três propriedades simultaneamente. Primeiramente, em relação à arquitetura da rede neural e suas funções de ativação os resultados foram bem similares e o uso da função *Sigmoid* mostrou um pequeno decréscimo da dispersão das predições se comparada à arquitetura com funções ReLU somente. Dada a boa performance dos modelos testados, não houve a necessidade de construir uma rede neural mais profunda, logo poupou-se tempo de treinamento dos modelos. E a estratégia de transformar os dados dos rótulos se mostrou importante para gerar um modelo mais acurado, portanto essa transformação será utilizada para todas as propriedades equivalentes para predizê-las simultaneamente.

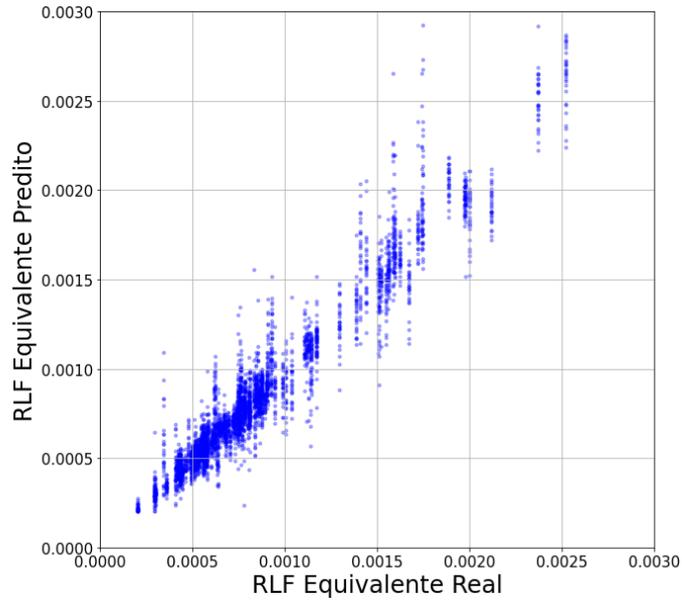


Figura 4.18 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados. Fonte: Elaborado pelo Autor.

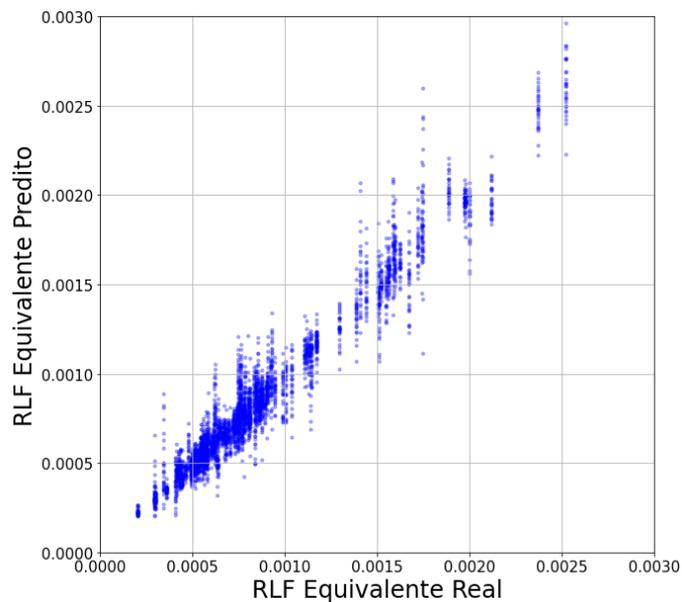


Figura 4.19 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados. Fonte: Elaborado pelo Autor.

4.2.4 Regressão simultânea das propriedades equivalentes

Levando em consideração todas as análises feitas, estruturou-se um modelo de redes neurais que fosse capaz de prever as propriedades equivalentes de forma simultânea,

conforme é mostrado na Figura 4.1. Os resultados obtidos para a predição das propriedades equivalentes, respectivamente para o DLF, Densidade Modal e RLF, considerando a arquitetura composta por funções de ativação ReLU, são apresentados nas Figuras 4.20, 4.21 e 4.22. Nota-se que as redes neurais obtiveram boa performance em prever as três propriedades, com um comportamento similar à regressão individual de cada uma delas, conforme as seções anteriores. Em termos da métrica de performance, o *Explained Variance Score* médio obtido para esses modelos foi de $95,40 \pm 2,14\%$.

Considerando a segunda arquitetura das redes neurais, novamente bons resultados puderam ser observados, conforme as Figuras 4.23, 4.24 e 4.25, seguindo os mesmos padrões obtidos na regressão individual das propriedades equivalentes e o *Explained Variance Score* neste caso foi de em média $95,21 \pm 1,95\%$.

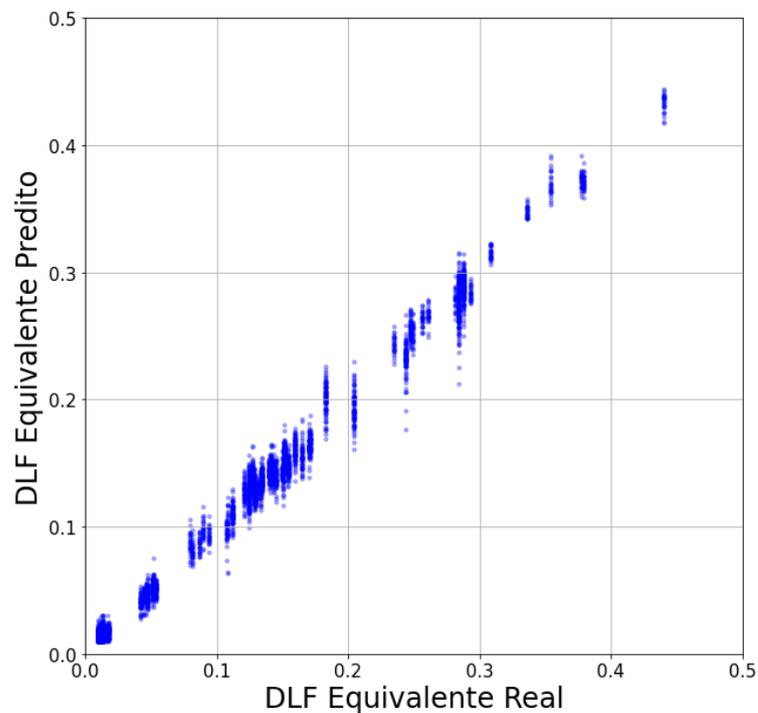


Figura 4.20 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

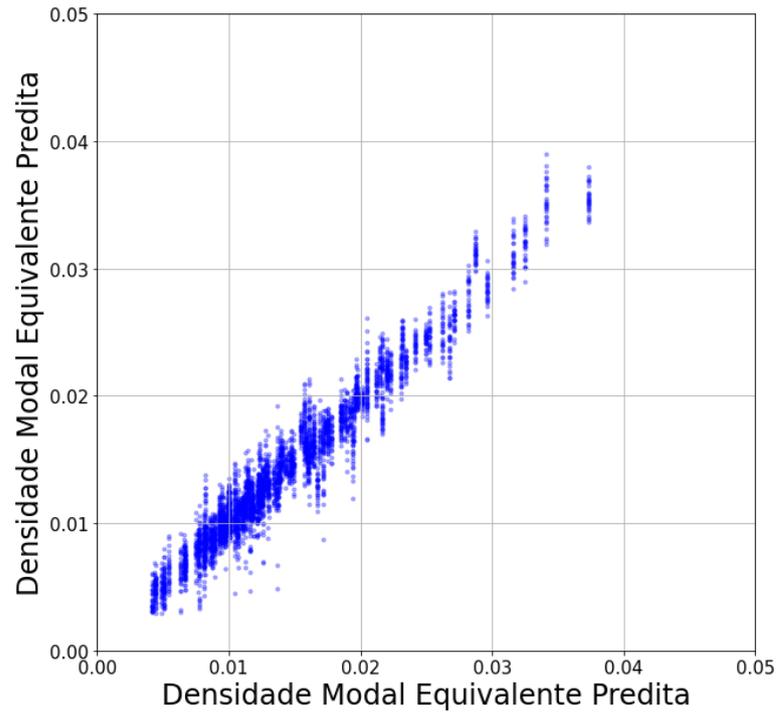


Figura 4.21 - Comparação entre os valores da Densidade Modal Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

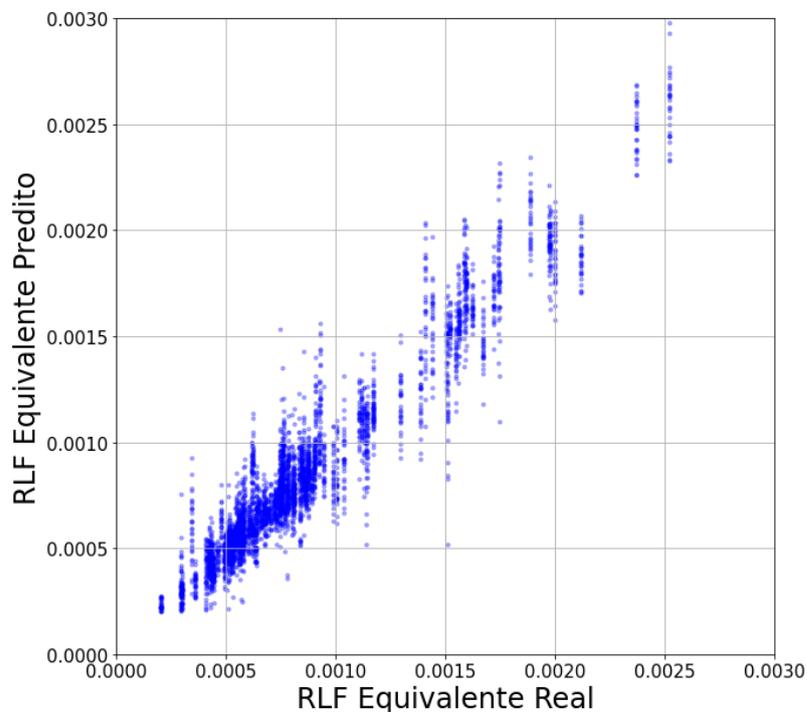


Figura 4.22 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com funções de ativação ReLU, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

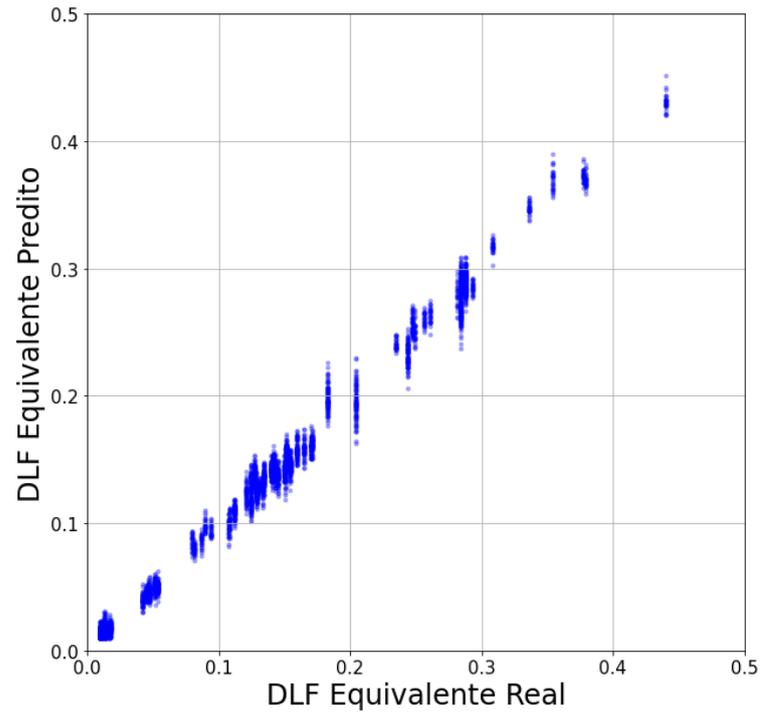


Figura 4.23 - Comparação entre os valores de DLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

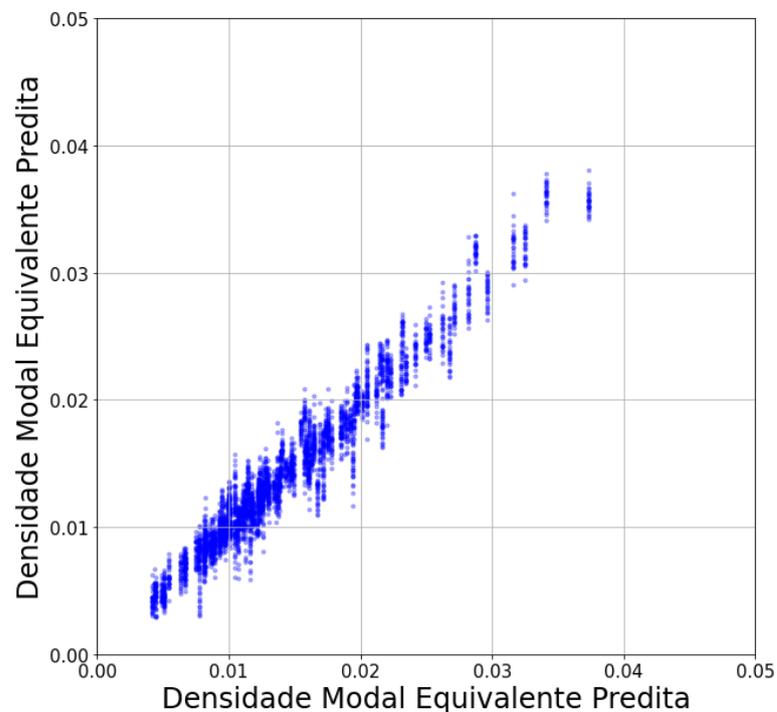


Figura 4.24 - Comparação entre os valores de Densidade Modal Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

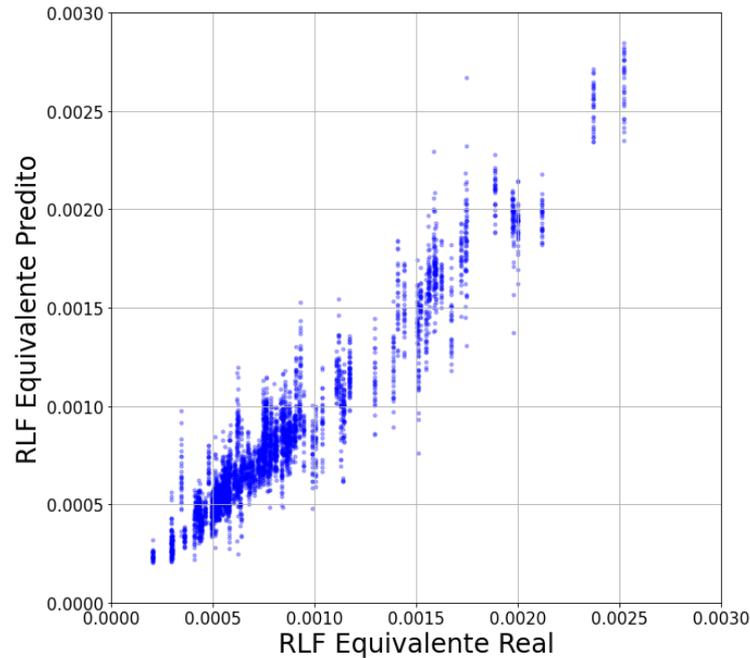


Figura 4.25 - Comparação entre os valores de RLF Equivalente Real e Predito pelas redes neurais com três funções de ativação ReLU e uma função Sigmoid, considerando a transformação dos dados e regressão simultânea. Fonte: Elaborado pelo Autor.

E para fins comparativos, um modelo de regressão linear polinomial de segundo grau foi treinado para estimar as três propriedades equivalentes simultaneamente e os resultados obtidos são apresentados nas Figuras 4.24, 4.25 e 4.26. As previsões realizadas por esse modelo são idênticas às previsões obtidas nas Figura 4.6, Figura 4.11 e Figura 4.16b, logo utilizar um método de transformação dos rótulos não apresentou melhoras para o modelo de regressão linear e acurácia deste modelo foi de 84,05%, o que representa a média das acurácias dos modelos de regressão individuais. Dessa forma, evidencia-se a necessidade de um modelo mais complexo que seja capaz de prever todos os parâmetros com uma acurácia mais elevada e uniforme, conforme os resultados obtidos para as redes neurais.

Até o momento, as análises exploratórias foram feitas para aferir a capacidade das redes neurais em prever as propriedades equivalentes e com isso determinar a acurácia e a dispersão destes modelos. Como essas propriedades são utilizadas posteriormente nos modelos de avaliação do ruído de cabine, é necessário conhecer a relação entre a dispersão dos valores das propriedades equivalentes previstas pelas redes neurais e a atenuação de ruído proporcionada pelos materiais viscoelásticos. Portanto, foi simulado um modelo de perda de transmissão sonora em SEA para determinar um intervalo de confiança para as redes neurais, que quantifica a variação da atenuação de ruído em função da dispersão dos valores das propriedades equivalentes previstas.

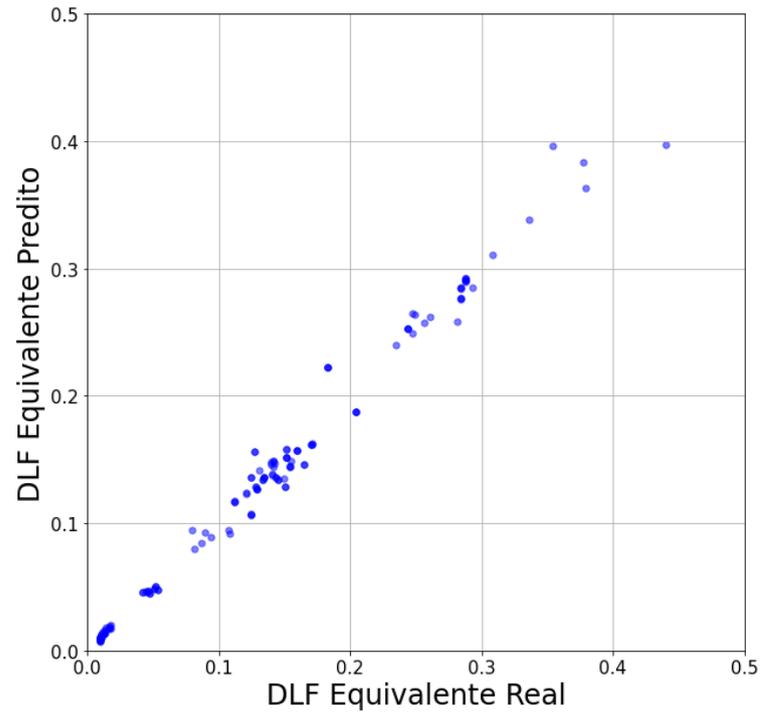


Figura 4.26 – Modelo de regressão linear polinomial de segundo grau para o DLF Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.

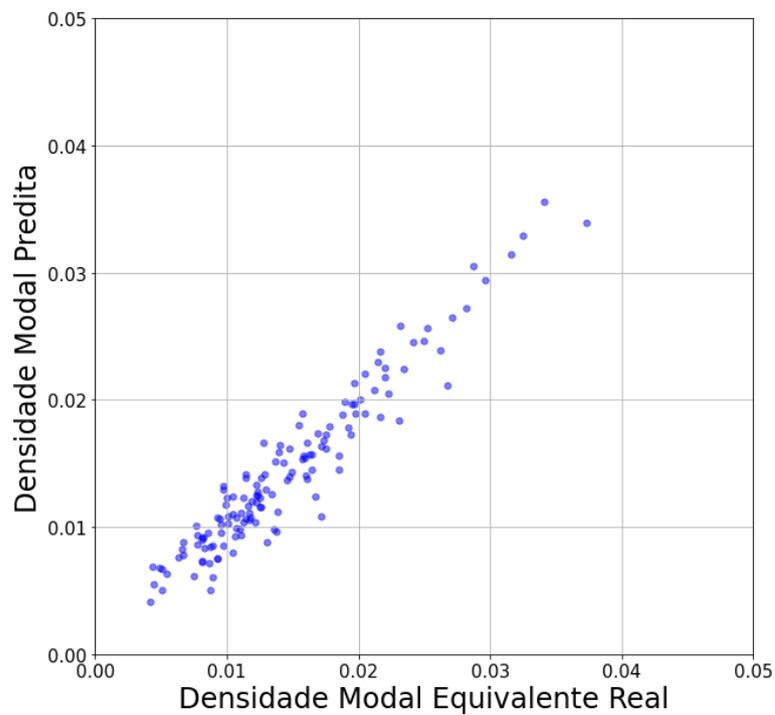


Figura 4.27 - – Modelo de regressão linear polinomial de segundo grau para a Densidade Modal Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.

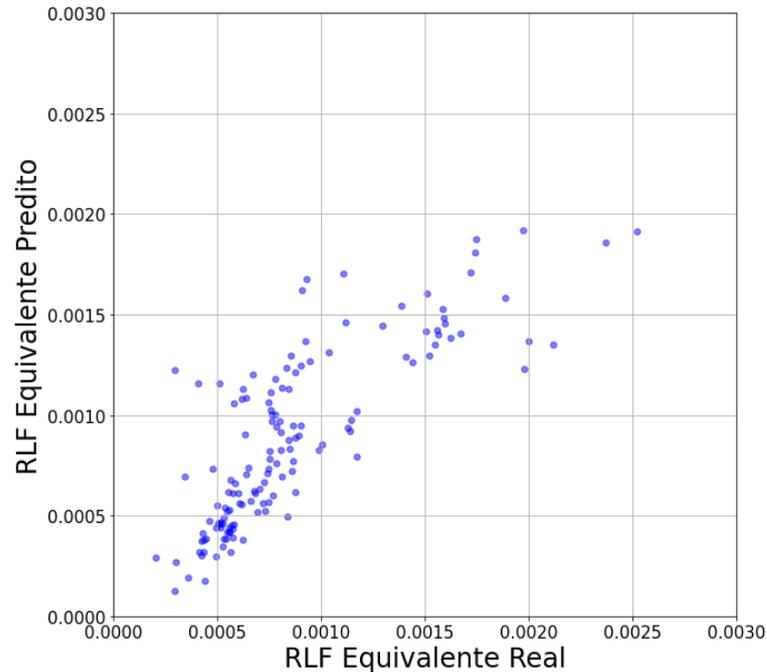


Figura 4.28 - – Modelo de regressão linear polinomial de segundo grau para o RLF Equivalente considerando a regressão simultânea das propriedades equivalentes. Fonte: Elaborado pelo Autor.

4.3 ANÁLISE DE SENSIBILIDADE EM UM MODELO DE PERDA DE TRANSMISSÃO EM SEA

Em função do caráter estocástico do algoritmo de otimização é esperado que cada rede neural tenha pesos e vieses distintos para cada treinamento. Logo, é necessário entender o impacto da variação das redes neurais em comparação aos resultados obtidos por FE em um caso mais prático de avaliação desses materiais, como por exemplo em uma simulação de perda de transmissão (TL, do inglês *Transmission Loss*) em SEA. A TL é definida como:

$$TL = -10 \log(\bar{\tau}), \quad (4.1)$$

sendo $\bar{\tau}$ o coeficiente de transmissão ponderado, definido como a fração de energia que é transmitida do meio fonte para o meio receptor na presença de uma partição entre eles. A ponderação do coeficiente de transmissão se dá pela dependência do coeficiente de transmissão função do ângulo de incidência das ondas sobre a partição. A Figura 4.29 mostra uma representação de uma simulação de TL em SEA, a qual consiste de uma cavidade acústica fonte, denominada neste caso de câmara fonte, com uma excitação acústica unitária

na faixa de frequência de interesse, conectada a uma partição, sendo este um painel de alumínio que representa uma parte da fuselagem e possui as propriedades equivalentes obtidas pela análise de estruturas periódicas e dimensões iguais à célula periódica. E do outro lado há uma segunda cavidade acústica, que é denominada câmara receptora. Portanto, conhecendo as energias em cada uma das cavidades é possível calcular a TL conforme a Equação 4.1.

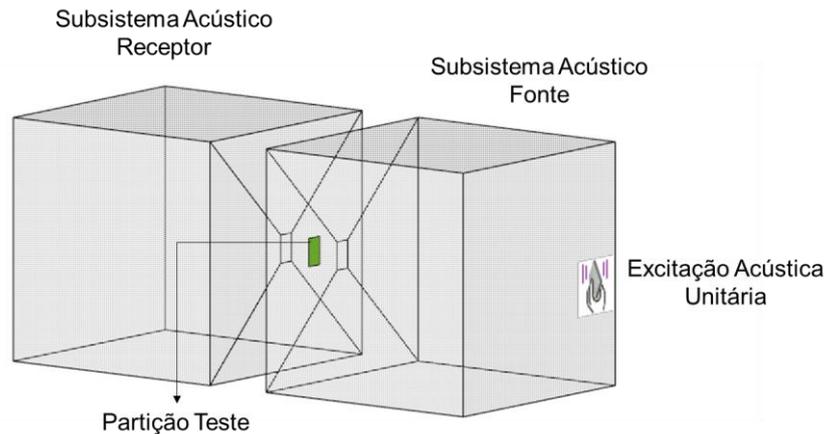


Figura 4.29 - Representação de uma simulação de TL em SEA. Fonte: Elaborado pelo Autor.

Conforme mencionado na seção 4.1 deste capítulo, um material viscoelástico foi separado para realizar a análise de sensibilidade comparando os valores preditos pelas redes neurais e o valor de referência obtido pelo método de FE. A Tabela 4-6 apresenta as propriedades do material viscoelástico A utilizado no modelo de TL para uma primeira análise de sensibilidade.

Tabela 4-6 - Propriedades do material A utilizado para análise de sensibilidade.

Periodicidade	Espessura Total [mm]	Densidade [kg/m ³]
2x5	3,8	1044

Frequência [Hz]	DLF	Módulo de Elasticidade [Pa]
500	0,2247	1,83E+10
630	0,2398	1,72E+10
800	0,2536	1,59E+10
1000	0,2659	1,44E+10
1250	0,2766	1,28E+10
1600	0,2855	1,11E+10
2000	0,2926	9,36E+09

Frequência [Hz]	DLF Equivalente	Densidade Modal Equivalente [modos/Hz]	RLF Equivalente
-----------------	-----------------	--	-----------------

500	0,140675	0,013286	0,00274037
630	0,109959	0,0139533	0,00152215
800	0,148439	0,0191334	0,00164452
1000	0,149998	0,0173624	0,00104526
1250	0,13818	0,0199081	0,000980726
1600	0,13968	0,0247671	0,000807767
2000	0,145246	0,0292763	0,000735633

As Figuras 4.30, 4.31 e 4.32 apresentam os resultados obtidos para a predição das propriedades equivalentes de DLF, Densidade Modal e RLF considerando 30 treinamentos das redes neurais para as duas arquiteturas utilizadas nas seções anteriores. Apesar da dispersão das propriedades equivalentes preditas pelas redes neurais oriundas do processo de otimização estocástico, verifica-se que o comportamento de todas as curvas se assemelha à curva de referência obtida pela solução do modelo de FE para as duas arquiteturas de redes neurais. As Figuras 4.33a e 4.34a apresentam os resultados de TL obtidos utilizando os valores das propriedades equivalentes preditas pelas redes neurais e as Figuras 4.33b e 4.34b apresentam o erro em função da frequência entre a TL referência considerando os valores das propriedades equivalentes advindas da solução de FE e a TL considerando as propriedades equivalentes preditas pelas redes neurais. Com exceção de um modelo, que apresentou um erro de aproximadamente 0,13 dB a 0,30 dB nas bandas de 800 Hz e 500 Hz, respectivamente, o erro médio ficou abaixo de 0,1 dB para toda faixa de frequência analisa, o que evidencia a capacidade das redes neurais em predizer as propriedades equivalentes com acurácia suficiente de forma a não introduzir significativos erros no comportamento da TL.

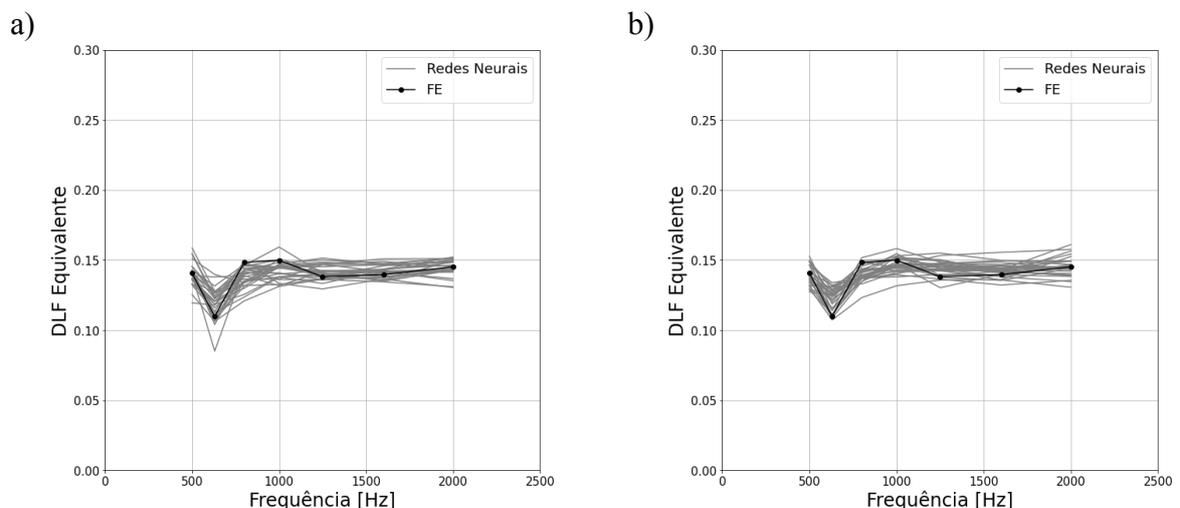


Figura 4.30 - Dispersão do DLF Equivalente do material A predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid.

Fonte: Elaborado pelo Autor.

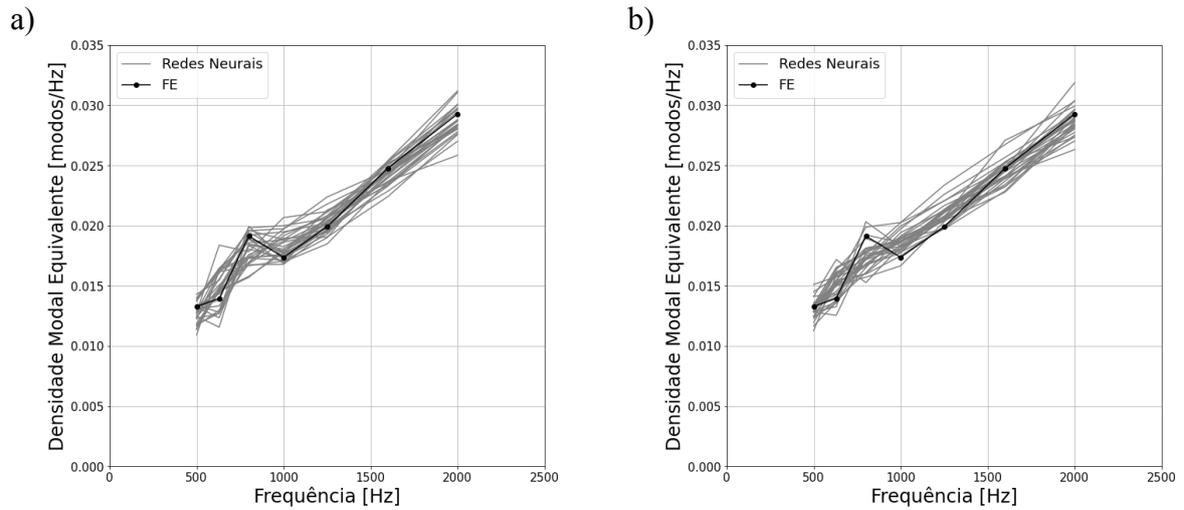


Figura 4.31 - Dispersão da Densidade Modal Equivalente do material A predita pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

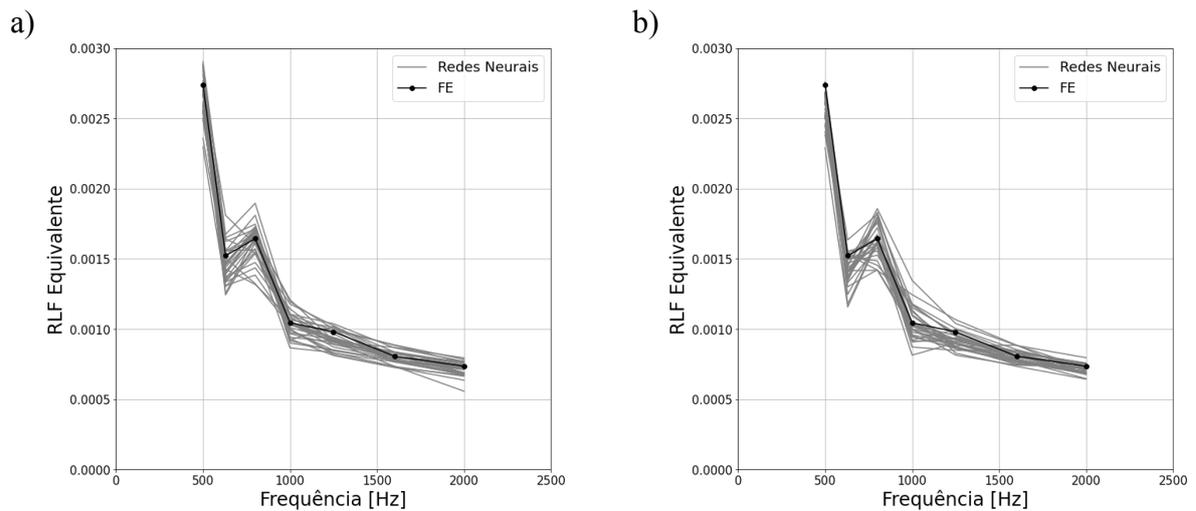


Figura 4.32 - Dispersão do RLF Equivalente do material A predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

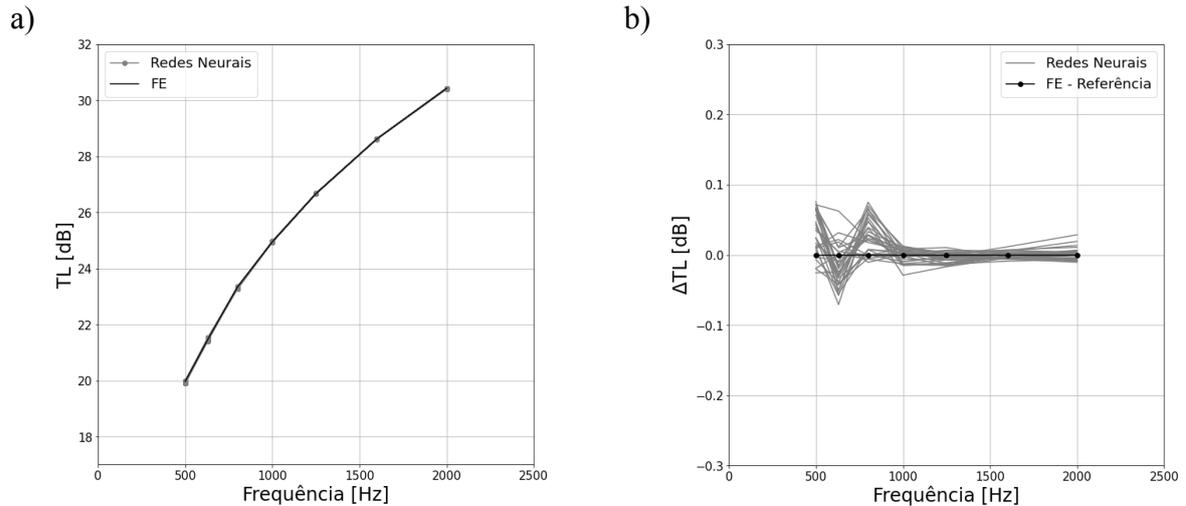


Figura 4.33 – a) Dispersão da TL do material A considerando as propriedades previstas pelas redes neurais considerando funções de ativação ReLU; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.

Além do material analisado, uma nova rodada de treinamentos foi realizada mantendo as proporções entre conjunto de treinamento, validação e teste e separando um segundo material para avaliar a capacidade de generalização das redes neurais considerando outros valores das propriedades do material viscoelástico. A Tabela 4-7 apresenta as propriedades do material B utilizado nesta análise de sensibilidade em um modelo TL em SEA.

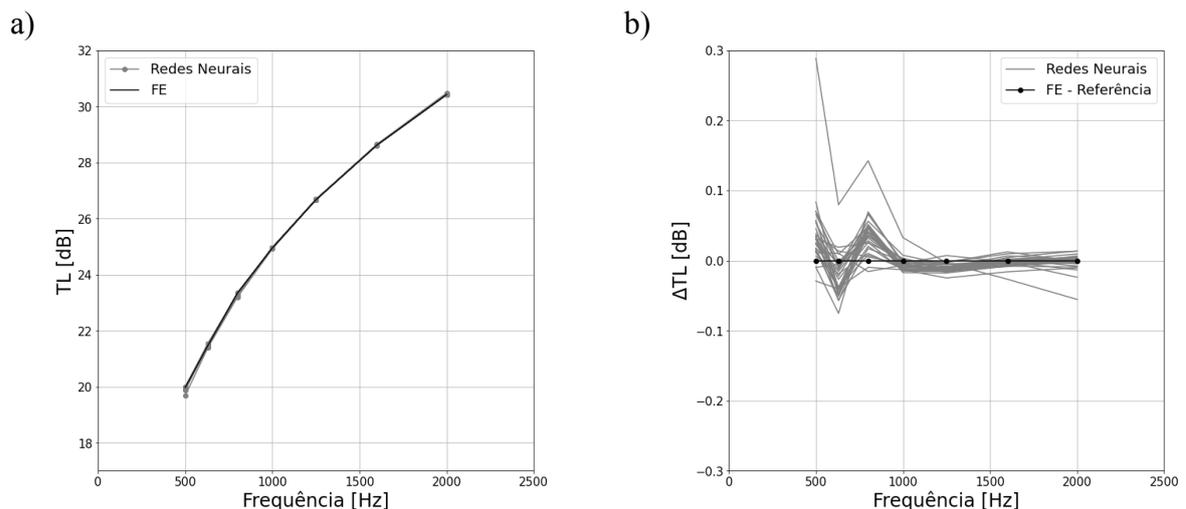


Figura 4.34 – a) Dispersão da TL do material A considerando as propriedades previstas pelas redes neurais considerando três funções de ativação ReLU e uma função Sigmoid; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.

Tabela 4-7 - Propriedades do material B utilizado para análise de sensibilidade.

Periodicidade	Espessura Total [mm]	Densidade [kg/m³]
2x4	2,0	1979

Frequência [Hz]	DLF	Módulo de Elasticidade [Pa]
500	0,274471	4,58E+10
630	0,334324	4,39E+10
800	0,390105	4,16E+10
1000	0,437536	3,91E+10
1250	0,473604	3,65E+10
1600	0,496531	3,38E+10
2000	0,506208	3,12E+10

Frequência [Hz]	DLF Equivalente	Densidade Modal Equivalente [modos/Hz]	RLF Equivalente
500	0,2040946	0,01550031	0,001732433
630	0,1830454	0,01639513	0,001333788
800	0,24415	0,02142937	0,000820352
1000	0,2843845	0,02094507	0,000805353
1250	0,2877539	0,02257705	0,000565804
1600	0,2875746	0,02525543	0,000568472
2000	0,2840836	0,02728774	0,000415317

As Figuras 4.35, 4.36 e 4.37 apresentam os resultados obtidos para a predição das propriedades equivalentes de DLF, Densidade Modal e RLF considerando 30 treinamentos das redes neurais. Observa-se um nível de dispersão similar ao caso do material A e o mesmo comportamento das curvas preditas pelas redes neurais para cada propriedade que na média tende à curva de FE.

As Figuras 4.38a e 4.39a mostram os resultados obtidos para a TL considerando os valores das propriedades equivalentes preditas pelas redes neurais em comparação com as propriedades obtidas pela solução do modelo de FE e as Figuras 4.38b e 4.39b apresentam o erro em função da frequência entre a TL referência e a TL para as propriedades obtidas pelas redes neurais. Novamente, os erros observados são baixos, com o maior desvio na banda de 500 Hz de aproximadamente 0,1 dB, com uma média próxima de 0,025 dB, ao longo da faixa de frequência analisada, reforçando mais uma vez a capacidade das redes neurais de predizerem as propriedades equivalentes com confiança significativa.

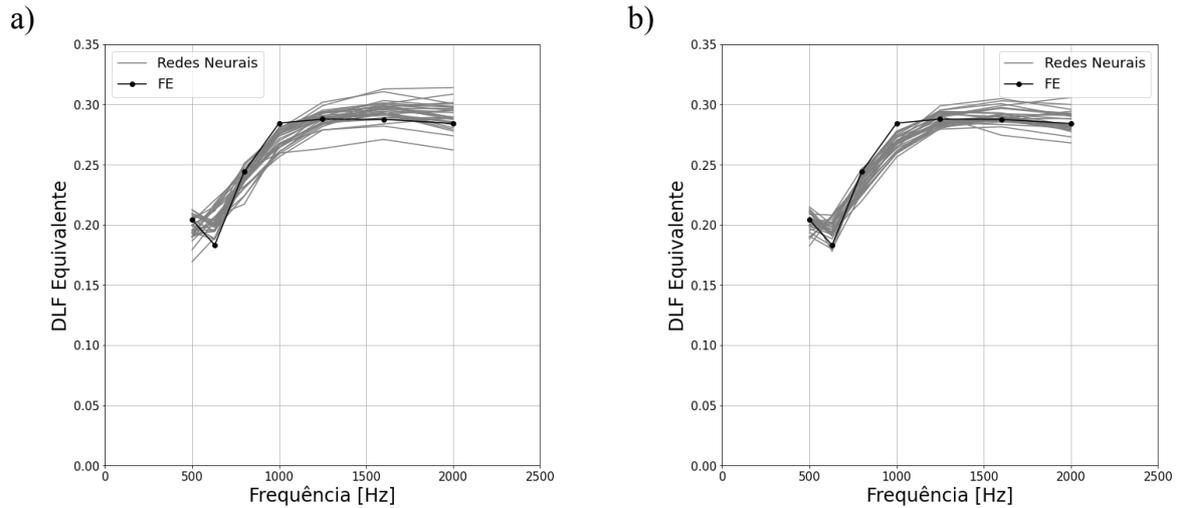


Figura 4.35 - Dispersão do DLF Equivalente do material B predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid.

Fonte: Elaborado pelo Autor.

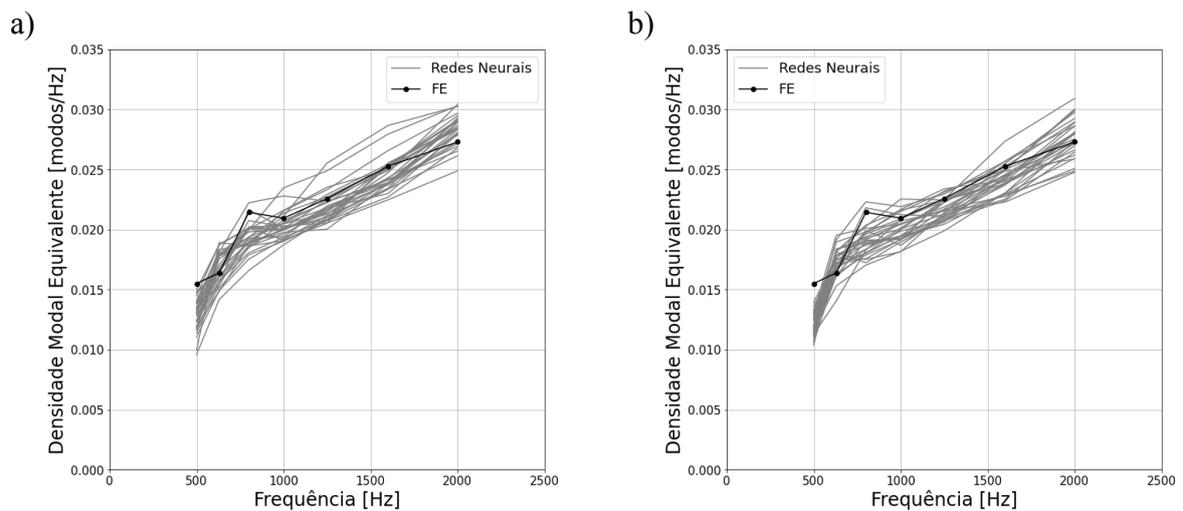


Figura 4.36 - Dispersão da Densidade Modal Equivalente do material B predita pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid. Fonte: Elaborado pelo Autor.

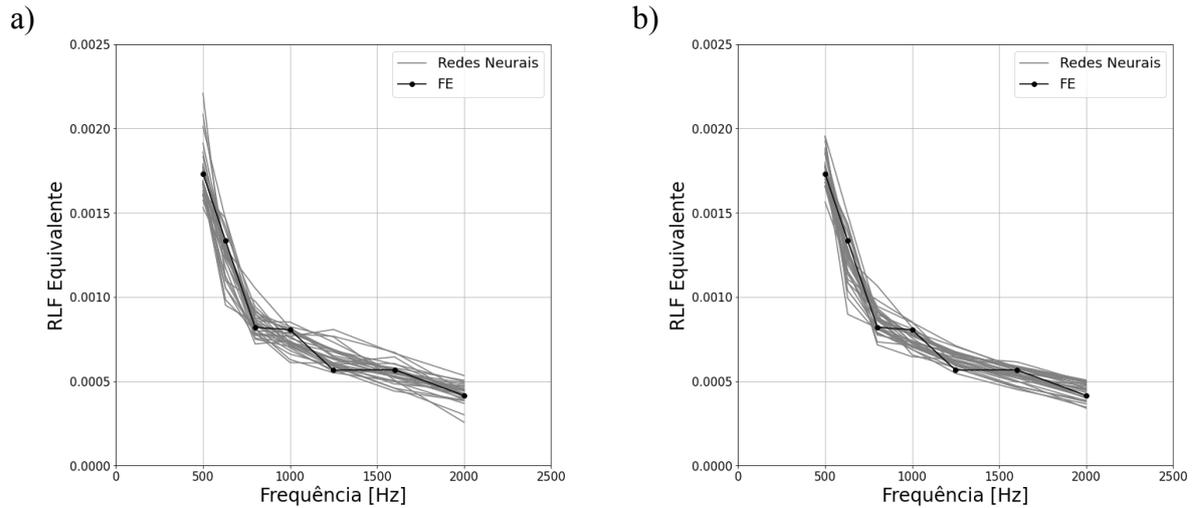


Figura 4.37 - Dispersão do RLF Equivalente do material B predito pelas redes neurais a) com funções de ativação ReLU; b) com três funções de ativação ReLU e uma função Sigmoid.
Fonte: Elaborado pelo Autor.

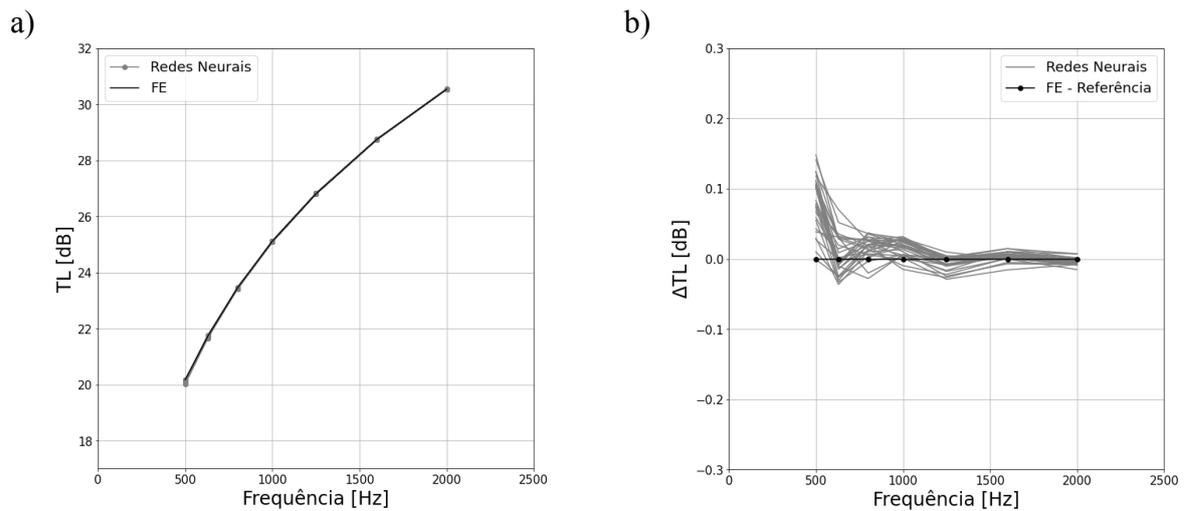


Figura 4.38 - a) Dispersão da TL do material B considerando as propriedades preditas pelas redes neurais considerando funções de ativação ReLU; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.

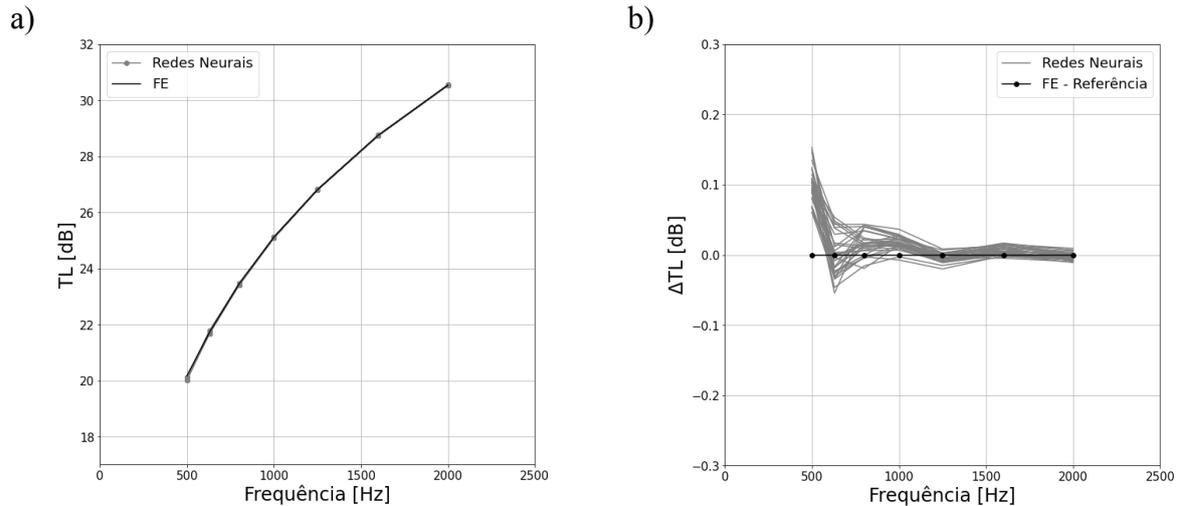


Figura 4.39 - a) Dispersão da TL do material B considerando as propriedades previstas pelas redes neurais considerando três funções de ativação ReLU e uma função Sigmoid; b) dispersão do erro da TL. Fonte: Elaborado pelo Autor.

A avaliação dos dois materiais viscoelásticos visou aferir a robustez e a capacidade de generalização das redes neurais para materiais com propriedades mecânicas e geométricas distintas. Neste sentido, foi possível avaliar a performance das redes neurais em relação aos materiais bem amortecidos, com rigidezes e características diferentes, assegurando a confiabilidade das redes neurais para a geometria da célula unitária definida na seção 2.2.

Considerando as ordens de grandeza dos erros associadas às previsões, conclui-se que são ínfimas e não impactam a tomada de decisão sobre a escolha do material viscoelástico sob a ótica de performance vibroacústica. Portanto, a escolha se dará por outros critérios, como por exemplo peso e custo, que influenciam diretamente o projeto de uma aeronave.

5 CONCLUSÕES

Este trabalho propôs estudar a aplicação de redes neurais para construir um modelo substituto das análises de FE e prever as propriedades equivalentes de estruturas periódicas, DLF, Densidade Modal e RLF, utilizadas nos modelos de SEA de ruído de cabine de aeronaves.

O Capítulo 2 apresentou a teoria de estruturas periódicas para solução de modelos de FE, como as propriedades equivalentes são calculadas, o método de caracterização dos materiais viscoelástico e como as propriedades destes materiais foram incorporadas nos modelos de FE considerando a dependência em função da frequência. Os resultados das propriedades equivalentes de dois materiais viscoelásticos foram apresentados e pôde-se verificar a influência da variação da rigidez na frequência, evidenciando a importância de considerar essa dependência no cálculo destas propriedades.

O Capítulo 3 visou consolidar o embasamento teórico acerca do campo de inteligência artificial, aprendizado de máquina e redes neurais, permitindo a compreensão do estudo desenvolvido no Capítulo 4. A combinação da caracterização dos materiais viscoelástico e a solução dos modelos de FE para obtenção das propriedades equivalentes da célula periódica compuseram o conjunto de dados utilizados para treinamento, validação, teste e análise de sensibilidade. Para tanto, foi definida inicialmente uma arquitetura de rede neural, a qual foi comparada a uma segunda arquitetura considerando diferentes funções de ativação. Em ambos os casos, o tamanho, a largura, parâmetros das redes neurais e algoritmos de otimização foram iguais.

Inicialmente, as propriedades equivalentes foram previstas de maneira individual de forma a compreender o comportamento das redes neurais. Para o DLF e para a Densidade Modal Equivalentes bons resultados foram observados com acurácias em torno de 99% e 85%, respectivamente. Já para o RLF Equivalente as redes neurais apresentaram um erro muito elevado, gerando previsões praticamente aleatórias. Porém, notou-se que um modelo mais simples de regressão linear polinomial gerou previsões muito mais acuradas do que as redes neurais, o que motivou a tentativa de escalar e transformar os dados de RLF Equivalente. Com essa abordagem os resultados obtidos foram muito superiores, atingindo uma acurácia de aproximadamente 94% e, portanto, essa estratégia foi utilizada para realizar a regressão simultânea das propriedades equivalentes, na qual se atingiu uma acurácia próxima a 95%

Por fim, os valores preditos pelas redes neurais para as propriedades equivalentes foram utilizados em um modelo de perda de transmissão em SEA para determinar a dispersão da curva de atenuação sonora e o erro entre a abordagem pelo uso do modelo substituto e o modelo de FE. Observou-se que a dispersão das propriedades preditas pelas redes neurais influenciou pouco na perda de transmissão dos dois painéis com material viscoelástico avaliados, com erros abaixo de 0,3 dB em toda a faixa de frequência analisada. Portanto, a utilização das redes neurais para predição das propriedades equivalentes de estruturas periódicas restritas à geometria da célula unitária adotada se mostrou um método acurado, com baixo erro associado à perda de transmissão e eficiente, pois com as redes neurais treinadas a avaliação de novos materiais torna-se instantânea.

5.1 SUGESTÕES PARA TRABALHOS FUTUROS

Como recomendações para trabalhos futuros, sugere-se:

- Estender o treinamento das redes neurais para outras temperaturas de operação dos materiais viscoelásticos. Neste trabalho, o foco se deu na fase de altitude de cruzeiro, na qual a temperatura do meio é próxima à -40°C , porém é válido considerar outras fases de voo, como subida, descida e até mesmo com a aeronave em solo, em que o meio se encontra à temperatura ambiente, resultando em um modelo substituto dependente da temperatura de operação do material viscoelástico.
- Utilizou-se uma única célula periódica para calcular as propriedades equivalentes, porém a variação de geometrias ao longo do comprimento da aeronave é grande, com diferentes espessuras de fuselagem, reforçadores e área de aplicação do material viscoelástico. Neste sentido, propõe-se estudar um método que consiga inferir o comportamento dinâmico de diferentes células periódicas, como por exemplo frequências naturais, formas modais ou momentos de inércia, e a partir disso predizer as propriedades equivalentes de diferentes geometrias.

REFERÊNCIAS

- ALGUACIL, A.; BAUERHEIM, M.; JACOB, M. C.; MOREAU, S. Predicting the propagation of acoustic waves using deep convolutional neural networks. **Journal of Sound and Vibration**, 512, (2021), 116285.
- ALIMONTI, L. ATALLA, N. GARDNER, B. Assessment of a contour integral method for the nonlinear eigenvalue problem arising from the wave finite element description of two-dimensional periodic waveguides involving dissipative and frequency dependent properties media. **6th Noise and vibration emerging methods**, Ibiza.
- BARDELL, N. S.; MEAD, D. J.; Free vibration of an orthogonally stiffened cylindrical shell, part I: Discrete line simple supports. **Journal of Sound and Vibration**, 134, (1989), 29-54.
- BARDELL, N. S.; MEAD, D. J.; Free vibration of an orthogonally stiffened cylindrical shell, part II: Discrete line simple supports. **Journal of Sound and Vibration**, 134, (1989), 55-72.
- BAYDIN, A. G.; PEARLMUTTER, B. A.; RADUL, A. A.; SISKIND, J. M. Automatic Differentiation in Machine Learning: a Survey. **Journal of Machine Learning Research**, 18, (2018), 1-43.
- BERNSTEIN, L., HAMERLY, R. *et al.* Freely scalable and reconfigurable optical hardware for deep learning. **Sci Rep** 11, (2021),3144.
- BIANCO, M. J.; GERSTOFT, P.; TRAER, J.; OZANICH, E.; ROCH, M. A.; GANNOT, S.; DELEDALLE, C-A. Machine learning in acoustics: Theory and applications. **Journal of the Acoustical Society of America**, 146, (2019), 3590-3628.
- CAVALHEIRO, T. **Estudo numérico do comportamento vibroacústico de painéis duplos aeronáuticos com enfoque em redução de ordem**. Tese de Doutorado. Universidade Federal de Santa Catarina, 2021.
- CORDIOLI, J. A.; GERGES, N. Y.; PEREIRA, A. K.; CARMO, M; GRANDI, C. **Vibro-acoustic modeling of aircrafts using statistical energy analysis**. SAE Technical Paper, 2004.
- CORDIOLI, J. A.; BRATTI, G.; STUMPF, C.; LENZI, A.; COTONI, V. On the prediction of damping loss factor of fuselage panels with viscoelastic materials using Periodic Structure Theory and Finite Element method. **International Conference on Noise and Vibration Engineering**, Leuven, 2257-2266.
- COTONI, V.; LANGLEY, R.; SHORTER, P. J. A statistical energy analysis subsystem formulation using finite element and periodic structure theory. **Journal of Sound and Vibration**, 318, (2008), 1077-1108.
- CYBENKO, G. Approximation by Superpositions of a Sigmoidal Function. **Math. Control Signals Systems**, 2, (1989), 303-314.
- DA SILVA, L. C. **Detecção e classificação de padrões sonoros utilizando inteligência artificial**. Dissertação de Mestrado. Universidade Federal de Santa Catarina, 2021.

EREMENKO, K.; DE PONTEVES, H. **Machine Learning A-Z™: Python & R in Data Science [2022]**. Udemy, <https://www.udemy.com/course/machinelearning/>.

FAHY, F. J. Statistical energy analysis: a critical overview. **Philosophical Transactions of the Royal Society of London**. Series A: Physical and Engineering Sciences, 346, (1994), 431–447.

GUO, S. **An introduction to Surrogate Modeling, Part I: fundamentals**. [s.l.: s.n.] <https://towardsdatascience.com/an-introduction-to-surrogate-modeling-part-i-fundamentals-84697ce4d241>. Acessado em 20 de dezembro de 2022.

HWANG, J. T.; MARTINS, J. R. R. A. A fast-prediction surrogate model for large datasets. **Aerospace Science and Technology**, 75, (2018), 74-87.

IBM. **Introduction to CRIPS-DM**. [s.l.: s.n.] <https://www.ibm.com/docs/en/spss-modeler/saas?topic=dm-crisp-help-overview>. Acessado em 20 de dezembro de 2022.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. **An Introduction to Statistical Learning**. Springer, 2013.

KAGIYAMA, N.; SHRESTHA, S.; FARJO, P. D.; SENGUPTA, P. P. Artificial Intelligence: Practical Primer for Clinical Research in Cardiovascular Disease. **Journal of the American Heart Association**, 8, (2019), 17.

KAVLAKOGLU, E. **AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the difference?** [s.l.: s.n.] <https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks>. Acessado em 20 de dezembro de 2022.

KINGMA, D. P.; BA, J. B. ADAM: A method for stochastic optimization. **Proceedings of the 3rd International Conference on Learning Representation**, San Diego, CA.

KULAKAUSKAS, F. **Avaliação do uso de materiais porosos na perda de transmissão de painéis duplos**. Dissertação de mestrado: Universidade Federal de Santa Catarina, Florianópolis, 2016.

LANGLEY, R. S. A note on the force boundary conditions for two-dimensional periodic structures with corner freedoms. **Journal of Sound and Vibration**, 167, (1993), 377-381.

LANGLEY, R. S. On the modal density and energy flow characteristics of periodic structures. **Journal of Sound and Vibration**, 172, (1994), 491-511.

LYON, R., H.; DeJong, R., G. Theory and application of Statistical Energy Analysis. **Journal of the Acoustical Society of America**, 98, (1995), 3021.

MEAD, D. J.; A general theory of harmonic wave propagation in linear periodic systems with multiple coupling. **Journal of Sound and Vibration**, 27, (1973), 235-260.

MEAD, D. J. Wave propagation in continuous periodic structures: Research contributions from Southampton. **Journal of Sound and Vibration**, 190, (1996), 495-524.

MEAD, D. J.; PARTHAN, S. Free wave propagation in two-dimensional periodic plates. **Journal of Sound and Vibration**, 64, (1979), 325-348.

MEAD, D. J.; ZHU, D. C.; BARDELL, N. S. Free vibration of an orthogonally stiffened flat plate. **Journal of Sound and Vibration**, 127, (1988), 19-48.

MITCHEL, T. M. **Machine Learning**. McGraw-Hill, 1997.

MITUSCH, S. K.; FUNKE, S. W.; KUTCHA, M. Hybrid FEM-NN models: Combining artificial neural networks with the finite element method. **Journal of Computational Physics**, 446, (2021), 110651.

NIELSEN, M. **Neural Networks and Deep Learning**. [s.l.: s.n.], <http://neuralnetworksanddeeplearning.com/index.html>. Acessado em 20 de dezembro de 2022.

PEREIRA, I.; NETO, B. G.; FUSTATSUGI, S.; SANTOS, G.; MONTEIRO, J. V.; CORDIOLI, J. A. Characterization Methodology for Highly Damped Aeronautic Structures Using Equivalent Properties. **NoiseCon19**, San Diego, CA, 240-250.

PORTILLA, J. **Complete Tensorflow 2 and Keras Deep Learning Bootcamp**. Udemy, <https://www.udemy.com/course/complete-tensorflow-2-and-keras-deep-learning-bootcamp/>.

PORTILLA, J. **Python for Data Science and Machine Learning Bootcamp**. Udemy, <https://www.udemy.com/course/python-for-data-science-and-machine-learning-bootcamp/>.

RAISSI, M.; PERDIKARIS, P.; KARNIADAKIS, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. **Journal of Computational Physics**, 378, (2019), 686-707.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. **Psychological Review**, 65, (1958), 386-408, 1958.

RUMELHARTM, D. E.; HINTON G. E.; WILLIAMS, R. J. Representations by Back-propagating Errors. **Nature**, 323, (1986), 533-536.

SAVAGE, N. **The race to the top among the world's leaders in artificial intelligence**. [s.l.: s.n.], <https://www.nature.com/articles/d41586-020-03409-8>. Acessado em 20 de dezembro de 2022.

SACK, S; ÅBOM, M. Acoustic plane-wave decomposition by means of multilayer perceptron neural networks. **Journal of Sound and Vibration**, 486, (2020), 115518.

SCHIMDHUBER, J. **Annotated History of Modern AI and Deep Learning**. [s.l.: s.n.], <https://people.idsia.ch/~juergen/deep-learning-history.html>. Acessado em 15 de janeiro de 2023.

SHORTER, P. J. Wave propagation and damping in linear viscoelastic laminates. **Journal of the Acoustical Society of America**, 115, (2004), 1917-1925.

SILVA, P. B. **Dynamic Analysis of periodic structures via wave-based numerical approaches and substructuring techniques**. Tese de Doutorado. Universidade Estadual de Campinas, 2015.

THADA, A. PANCHAL, S.; DUBEY, A.; RAO, L. B. Machine learning based frequency modelling. **Mechanical Systems and Signal Processing**, 160, (2021), 107915.

VA One 2019 User's Guide, ESI Group, San Diego, EUA, Fevereiro 2009.

WILBY, J. F.; Aircraft Interior Noise. **Journal of Sound and Vibration**, 190, (1996), 545-564.

YUCESAN, Y. A.; VIANA, F. A. C.; MANIN, L.; MAHFOUD, J. Adjusting a torsional vibration damper model with physics-informed neural networks. **Mechanical System and Signal Processing**, 154, (2021), 107552.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. **Dive into Deep Learning**. [s.l.: s.n.], <https://d2l.ai/index.html>. Acessado em 20 de dezembro de 2022.