

UNIVERSIDADE FEDERAL DE ALAGOAS
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

WILL JOSÉ DE LIMA JÚNIOR

**IDENTIFICAÇÃO DE DANOS EM ELEMENTOS ESTRUTURAIS LINEARES
UTILIZANDO METODOLOGIAS DE MODELO HÍBRIDO BASEADO NA FÍSICA E
ORIENTADO A DADOS, *MACHINE LEARNING* E *DIGITAL TWIN***

MACEIÓ

2024

WILL JOSÉ DE LIMA JÚNIOR

**IDENTIFICAÇÃO DE DANOS EM ELEMENTOS ESTRUTURAIS LINEARES
UTILIZANDO METODOLOGIAS DE MODELO HÍBRIDO BASEADO NA FÍSICA E
ORIENTADO A DADOS, MACHINE LEARNING E DIGITAL TWIN**

Tese apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas, como parte dos requisitos para a obtenção do título de Doutor em Engenharia Civil.

Área de concentração: *Estruturas e Materiais*

Orientador: Prof. Dr. Wayne Santos de Assis
Coorientador: Prof. Dr. André Luiz Lins de Aquino

MACEIÓ
2024

**Catalogação na Fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico**

Bibliotecário: Jone Sidney A. de Oliveira – CRB-4 – 1485

L732i Lima Júnior, Will José de.

Identificação de danos em elementos estruturais lineares utilizando metodologias de modelo híbrido baseado na física e orientado a dados, machine learning e digital twin / Will José de Lima Júnior. – 2024.

178 f. : il.

Orientador: Wayne Santos de Assis.

Coorientador: André Luiz Lins de Aquino.

Tese (Doutorado em Engenharia Civil) – Universidade Federal de Alagoas.
Programa de Pós-Graduação em Engenharia Civil. Maceió, 2024.

Bibliografia: f. 147-161.

Apêndices: f. 162-178.

1. Identificação de Dano. 2. Dinâmica das Estruturas. 3. Modelo Híbrido. 4. Machine Learning. 5. Digital Twin.. I. Título.

CDU: 624.01



**IDENTIFICAÇÃO DE DANOS EM ELEMENTOS ESTRUTURAIS LINEARES
UTILIZANDO METODOLOGIAS DE MODELO HÍBRIDO BASEADO NA FÍSICA E
ORIENTADO A DADOS, MACHINE LEARNING E DIGITAL TWIN**

WILL JOSÉ DE LIMA JÚNIOR

Tese de doutorado submetida à banca examinadora do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas e aprovada no dia 18 do mês de dezembro do ano de 2024.

Banca Examinadora:

ICP Brasil Documento assinado digitalmente
WAYNE SANTOS DE ASSIS
Data: 18/12/2024 16:55:49-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Wayne Santos de Assis
(Orientador – PPGEC/UFAL)

ANDRE LUIZ LINS DE Assinado de forma digital por
AQUINO:032350154 ANDRE LUIZ LINS DE
00 AQUINO:03235015400
Dados: 2024.12.26 12:05:24
-03'00'

Prof. Dr. André Luiz Lins de Aquino
(Coorientador – UFAL)

Documento assinado digitalmente
gov.br EDUARDO TOLEDO DE LIMA JUNIOR
Data: 26/12/2024 09:36:43-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Eduardo Toledo de Lima Junior
(Avaliador Interno – PPGEC/UFAL)

Documento assinado digitalmente
gov.br JOAO PAULO LIMA SANTOS
Data: 26/12/2024 10:12:45-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. João Paulo Lima Santos
(Avaliador Interno – PPGEC/UFAL)

Documento assinado digitalmente
gov.br FABIANE DA SILVA QUEIROZ
Data: 23/12/2024 16:17:33-0300
Verifique em <https://validar.iti.gov.br>

Profa. Dra. Fabiane da Silva Queiroz
(Avaliadora Externa ao Programa - UFAL)

Documento assinado digitalmente
gov.br LEANDRO MOUTA TRAUTWEIN
Data: 20/12/2024 10:24:00-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Leandro Mouta Trautwein
(Avaliador Externo à Instituição - UNICAMP)

RESUMO

Este trabalho apresenta um novo *framework* para monitoramento da integridade estrutural para identificação de danos em elementos estruturais lineares de aço (barras comprimidas e vigas), no contexto de sistemas dinâmicos. O *framework* integra um modelo híbrido baseado na física e orientado a dados (que pode resultar em um modelo generalizável, acurado, interpretável e computacionalmente eficiente) e métodos de *machine learning* supervisionado, para a construção de um *framework digital twin*. As equações governantes do movimento da estrutura íntegra, descobertas pela modelagem híbrida, são usadas para simular a resposta do sistema com dano em diferentes locais e intensidades. A partir dessas simulações, um conjunto de dados é construído para treinar os classificadores de *machine learning*, considerando os cenários da estrutura íntegra e com dano. O *framework digital twin* relaciona as entradas do *physical twin* a cenários de danos específicos, para avisar rapidamente se houver dano, onde está localizado e qual a sua intensidade, de forma que possa apoiar as decisões de engenharia. O *framework digital twin* foi avaliado em diferentes configurações de aplicações, demonstrando a potencial contribuição para o estabelecimento de sistemas monitoramento da integridade estrutural. O *Support Vector Machine* foi o classificador que apresentou melhor desempenho, com precisão de 93,37% para a barra engastada e 80,33% para a barra biengastada. Os danos nas barras engastada e biengastada, considerando a vibração axial, foram identificados e robustos para determinados níveis de ruído. A identificação de dano em vigas, considerando a vibração transversal, se mostrou promissora, com precisão de 84,49% para o classificador *Support Vector Machine*, usando atributo de deslocamento, e 99,98% para o classificador *Quadratic Discriminant Analysis*, usando atributos de deslocamento e aceleração.

Palavras-chave: Identificação de Dano; Dinâmica das Estruturas; Modelo Híbrido; *Machine Learning*; *Digital Twin*.

ABSTRACT

This work presents a novel structural health monitoring framework for damage identification in linear steel structural elements (compressed bars and beams), in the context of dynamical systems. The framework integrates a hybrid physics-based and data-driven model (that can result in a generalizable, accurate, interpretable and computationally efficient model) and supervised machine learning methods, to construct a digital twin. The governing equations of motion of the healthy structure, discovered by hybrid modeling, are used to simulate the response of the system with damage at different locations and intensities. From these simulations a dataset is constructed to train the machine learning classifiers, considering the beam scenarios with damage and undamaged. The constructed digital twin relates the inputs of the physical twin to specific damage scenarios to quickly warn if there is damage, where it is located and what is its severity, supporting engineering decisions. The DT framework was evaluated in different application configurations, demonstrating the potential contribution to the establishment of structural integrity monitoring systems. Support Vector Machine was the best performing classifier, with a precision of 93.37% for the cantilever bar and 80.33% for the fixed-end bar. The damage to the cantilever and fixed-end bars, considering axial vibration, was identified and robust for certain noise levels. The identification of damage in beams, considering transverse vibration, proved promising, with a precision of 84.49% for the Support Vector Machine classifier, using the displacement feature, and 99.98% for the Quadratic Discriminant Analysis classifier, using the displacement and acceleration features.

Keywords: Damage Identification; Structural Dynamics; Hybrid Model; Machine Learning, Digital Twin.

LISTA DE FIGURAS

Figura 1 – Procedimento metodológico proposto para a Tese.....	30
Figura 2 – Framework DT com quatro camadas.....	31
Figura 3 – Diagrama de Venn apresentando a relação entre os diferentes sistemas de IA.....	34
Figura 4 – Aplicação do método k-NN: (a) dados iniciais, (b) calcular distância e (c) encontrar vizinhos e votar em atributos.....	37
Figura 5 – (a) Conjunto de treinamento binário, (b) três hiperplanos de separação e (c) hiperplano de margem máxima.....	44
Figura 6 – Hiperplano de separação que maximiza a margem.....	45
Figura 7 – Um hiperplano de separação com amostra xi classificada incorretamente e amostra xj classificada corretamente, mas com margem menor que 1.....	49
Figura 8 – Caso não linearmente separável: (a) nenhum hiperplano pode separar as duas classes e (b) um mapeamento não linear pode ser usado.....	50
Figura 9 – Um exemplo de mapeamento não linear de 2 dimensões para 3 dimensões, onde as amostras se tornam linearmente separáveis.....	51
Figura 10 – Idealização do modelo de barra (a) engastada e (b) biengastada, utilizando descrição de parâmetros concentrados de n -DOF.....	66
Figura 11 – (a) Barra engastada discretizada em dois elementos finitos e (b) correspondente idealização.....	73
Figura 12 – Respostas de deslocamento no tempo, barra engastada discretizada em dois elementos finitos.....	73
Figura 13 – Determinação do (a) δ ótimo e (b) λ ótimo, barra engastada discretizada em dois elementos finitos.....	75
Figura 14 – (a) Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.....	76
Figura 15 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.....	76
Figura 16 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de	

20% na (a) primeira mola, (b) com correspondentes NRMSE, e (b) segunda mola, (d) com correspondentes NRMSE.....	77
Figura 17 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.....	79
Figura 18 – (a) Barra engastada discretizada em quatro elementos finitos e (b) correspondente idealização.....	82
Figura 19 – Respostas de deslocamento no tempo, barra engastada discretizada em quatro elementos finitos.....	82
Figura 20 – Determinação do (a) $\delta_{ótimo}$ e (b) $\lambda_{ótimo}$, barra engastada discretizada em quatro elementos finitos.....	83
Figura 21 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.....	84
Figura 22 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.....	84
Figura 23 – Comparação das respostas de deslocamento no tempo entre o <i>physical twin</i> (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.....	85
Figura 24 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.....	87
Figura 25 – (a) Barra engastada discretizada em seis elementos finitos e (b) correspondente idealização.....	89
Figura 26 – Respostas de deslocamento no tempo, barra engastada discretizada em seis elementos finitos.....	89
Figura 27 – Determinação do (a) $\delta_{ótimo}$ e (b) $\lambda_{ótimo}$, barra engastada discretizada em seis elementos finitos.....	90
Figura 28 – (a) Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.....	91
Figura 29 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS), para dano de	

5% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) quinta mola, (d) com correspondentes NRMSE.....	91
Figura 30 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quinta mola, (d) com correspondentes NRMSE.....	92
Figura 31 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.....	94
Figura 32 – (a) Barra biengastada discretizada em dois elementos finitos e (b) correspondente idealização.....	96
Figura 33 – Resposta de deslocamento no tempo, barra biengastada discretizada em dois elementos finitos	96
Figura 34 – Determinação do (a) $\delta_{ótimo}$ e (b) $\lambda_{ótimo}$, barra biengastada discretizada em dois elementos finitos.....	97
Figura 35 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em dois elementos finitos, e modelo SINDy (MS) considerando a taxa de amostragem de (a) 7 kHz e (b) 14 kHz, respectivamente.	97
Figura 36 –(a) Barra biengastada discretizada em quatro elementos finitos e (b) correspondente idealização.....	99
Figura 37 – Respostas de deslocamento no tempo, barra biengastada discretizada em quatro elementos finitos	100
Figura 38 – Determinação do (a) $\delta_{ótimo}$ e (b) $\lambda_{ótimo}$, barra biengastada discretizada em quatro elementos finitos.	100
Figura 39 – (a) Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.	101
Figura 40 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.	102
Figura 41 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.	102

Figura 42 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.....	104
Figura 43 – (a) Barra biengastada discretizada em seis elementos finitos e (b) correspondente idealização.....	106
Figura 44 – Respostas de deslocamento no tempo, barra biengastada discretizada em seis elementos finitos.....	106
Figura 45 – Determinação do (a) $\delta\text{ótimo}$ e (b) $\lambda\text{ótimo}$, barra biengastada discretizada em seis elementos finitos.....	107
Figura 46 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.....	108
Figura 47 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) quinta mola, (b) com correspondentes NRMSE, e (c) sexta mola, (d) com correspondentes NRMSE.....	108
Figura 48 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) sexta mola, (d) com correspondentes NRMSE.....	109
Figura 49 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.....	111
Figura 50 – Idealização do modelo de viga (a) engastada e (b) biengastada, utilizando descrição de parâmetros concentrados de n -DOF representado por um modelo SLS.....	114
Figura 51 – Idealização do sistema equivalente n -DOF do modelo de viga (a) engastada e (b) biengastada.....	114
Figura 52 – (a) Viga engastada discretizada em dois elementos finitos e (b) correspondente idealização.....	120
Figura 53 – Respostas de deslocamento no tempo, viga engastada discretizada em dois elementos finitos	120
Figura 54 – Determinação do (a) $\delta\text{ótimo}$ e (b) $\lambda\text{ótimo}$, viga engastada discretizada em dois elementos finitos.....	122
Figura 55 – (a) Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.....	123

Figura 56 – (a) Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para o DOF 2 e (b) correspondentes NRMSE.....	124
Figura 57 – (a) Índices de dano para o conjunto de dados do MEF e (b) índices de dano para os conjuntos de dados.	124
Figura 58 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.	125
Figura 59 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.	126
Figura 60 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 1, do modelo de viga engastada discretizada em dois elementos finitos.....	128
Figura 61 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em dois elementos finitos.....	129
Figura 62 – Matriz de confusão do classificador QDA, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em dois elementos finitos.....	129
Figura 63 – (a) Viga engastada discretizada em quatro elementos finitos e (b) correspondente idealização.	132
Figura 64 – Respostas de deslocamento no tempo, viga engastada discretizada em quatro elementos finitos	133
Figura 65 – Avaliação do erro (RMSE) para os cinco conjuntos de treinamento.	134
Figura 66 – Determinação do δ ótimo, viga engastada discretizada em quatro elementos finitos.	135
Figura 67 – Comparação de respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS).	136
Figura 68 – (a) Comparação de respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para o DOF 3 e (b) correspondentes NRMSE.	137
Figura 69 – (a) Índices de dano para o conjunto de dados do MEF e (b) índices de dano para os conjuntos de dados.	137

Figura 70 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.	138
Figura 71 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.	139
Figura 72 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 1, do modelo de viga engastada discretizada em quatro elementos finitos.	141
Figura 73 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em quatro elementos finitos.	141
Figura 74 – Matriz de confusão do classificador QDA, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em quatro elementos finitos.	142

LISTA DE QUADROS

Quadro 1 – Diferentes métricas de distância.....39

LISTA DE TABELAS

Tabela 1 – Hiperparâmetros dos classificadores de ML, modelo de barra.....	72
Tabela 2 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.	78
Tabela 3 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.	78
Tabela 4 – Respostas do <i>framework</i> DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em dois elementos finitos.	81
Tabela 5 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.....	86
Tabela 6 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.....	86
Tabela 7 – Respostas do <i>framework</i> DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em quatro elementos finitos.	88
Tabela 8 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.	93
Tabela 9 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.....	93
Tabela 10 – Respostas do <i>framework</i> DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em seis elementos finitos.....	95
Tabela 11 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em dois elementos finitos.	98
Tabela 12 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em dois elementos finitos.	99
Tabela 13 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.....	103
Tabela 14 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.....	104
Tabela 15 – Respostas do <i>framework</i> DT do classificador SVM para a primeira mola, considerando a barra biengastada discretizada em quatro elementos finitos.	105
Tabela 16 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.....	110

Tabela 17 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.....	110
Tabela 18 – Respostas do <i>framework</i> DT do classificador SVM para a primeira mola, considerando a barra biengastada discretizada em seis elementos finitos.....	112
Tabela 19 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.....	127
Tabela 20 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.....	127
Tabela 21 – Resultados do <i>framework</i> DT utilizando o método SVM com ID 1, modelo de viga engastada discretizada em dois elementos finitos.	130
Tabela 22 – Resultados do <i>framework</i> DT utilizando os métodos SVM e QDA com ID 2, modelo de viga engastada discretizada em dois elementos finitos.....	131
Tabela 23 – Hiperparâmetros ótimos para o classificador SVM, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.....	131
Tabela 24 – Métricas de desempenho para o classificador SVM, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.....	131
Tabela 25 – Resultados do <i>framework</i> DT para os dois primeiros níveis de investigação do dano estrutural, utilizando o método SVM com ID 1, modelo de viga engastada discretizada em dois elementos finitos.....	132
Tabela 26 – Avaliação das equações descobertas para os cinco conjuntos de treinamento, considerando o método	134
Tabela 27 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em quatro elementos finitos.	140
Tabela 28 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em quatro elementos finitos.	140
Tabela 29 – Resultados do <i>framework</i> DT utilizando o método SVM com ID 1, modelo de viga engastada discretizada em quatro elementos finitos.	143
Tabela 30 – Resultados do <i>framework</i> DT utilizando os métodos SVM e QDA com ID 2, modelo de viga engastada discretizada em quatro elementos finitos.	144
Tabela 31 – Respostas do <i>framework</i> DT do classificador SVM para a segunda mola, considerando a barra engastada discretizada em dois elementos finitos.	162
Tabela 32 – Respostas do <i>framework</i> DT do classificador SVM para a segunda mola, considerando a barra engastada discretizada em quatro elementos finitos.	163

Tabela 33 – Respostas do <i>framework</i> DT do classificador SVM para a terceira mola, considerando a barra engastada discretizada em quatro elementos finitos.	164
Tabela 34 – Respostas do <i>framework</i> DT do classificador SVM para a quarta mola, considerando a barra engastada discretizada em quatro elementos finitos.	165
Tabela 35 – Respostas do <i>framework</i> DT do classificador SVM para a segunda mola, considerando a barra engastada discretizada em seis elementos finitos.....	166
Tabela 36 – Respostas do <i>framework</i> DT do classificador SVM para a terceira mola, considerando a barra engastada discretizada em seis elementos finitos.....	167
Tabela 37 – Respostas do <i>framework</i> DT do classificador SVM para a quarta mola, considerando a barra engastada discretizada em seis elementos finitos.....	168
Tabela 38 – Respostas do <i>framework</i> DT do classificador SVM para a quinta mola, considerando a barra engastada discretizada em seis elementos finitos.....	169
Tabela 39 – Respostas do <i>framework</i> DT do classificador SVM para a sexta mola, considerando a barra engastada discretizada em seis elementos finitos.	170
Tabela 40 – Respostas do <i>framework</i> DT do classificador SVM para a segunda mola, considerando a barra biengastada discretizada em quatro elementos finitos.	171
Tabela 41 – Respostas do <i>framework</i> DT do classificador SVM para a terceira mola, considerando a barra biengastada discretizada em quatro elementos finitos.	172
Tabela 42 – Respostas do <i>framework</i> DT do classificador SVM para a quarta mola, considerando a barra biengastada discretizada em quatro elementos finitos.	173
Tabela 43 – Respostas do <i>framework</i> DT do classificador SVM para a segunda mola, considerando a barra biengastada discretizada em seis elementos finitos.....	174
Tabela 44 – Respostas do <i>framework</i> DT do classificador SVM para a terceira mola, considerando a barra biengastada discretizada em seis elementos finitos.....	175
Tabela 45 – Respostas do <i>framework</i> DT do classificador SVM para a quarta mola, considerando a barra biengastada discretizada em seis elementos finitos.....	176
Tabela 46 – Respostas do <i>framework</i> DT do classificador SVM para a quinta mola, considerando a barra biengastada discretizada em seis elementos finitos.....	177
Tabela 47 – Respostas do <i>framework</i> DT do classificador SVM para a sexta mola, considerando a barra biengastada discretizada em seis elementos finitos.....	178

LISTA DE ABREVIATURAS E SIGLAS

CD	Conjuntos de Dados
DA	<i>Discriminant Analysis</i>
DOF	<i>Degree of Freedom</i> (Grau de Liberdade)
DT	<i>Digital Twin</i>
DTs	<i>Decision Trees</i>
ET	<i>Extremely Randomized Trees</i>
IA	Inteligência Artificial
ID	Índice de Dano
IHM	Interface Homem-Máquina
IoT	<i>Internet of Things</i> (Internet das Coisas)
k-NN	<i>k-Nearest Neighbors</i>
LASSO	<i>Least Absolute Shrinkage and Selection Operator</i>
LDA	<i>Linear Discriminant Analysis</i>
ln	Logaritmo Natural
MAP	<i>Maximum A Posteriori</i> (Máximo a Posteriori)
MEF	Método dos Elementos Finitos
ML	<i>Machine Learning</i>
MS	Modelo SINDy
NB	<i>Naive Bayes</i>
NRMSE	<i>Normalized Root Mean Square Error</i> (Raiz do Erro Quadrático Médio Normalizado)
OVO	<i>One versus One</i> (Um contra Um)
OVR	<i>One versus Rest</i> (Um contra o Resto)
PT	<i>Physical Twin</i>
QDA	<i>Quadratic Discriminant Analysis</i>
RF	<i>Random Forest</i>
RMSE	<i>Root Mean Square Error</i> (Raiz do Erro Quadrático Médio)
SHM	<i>Structural Health Monitoring</i> (Monitoramento da Integridade Estrutural)

SINDy	<i>Sparse Identification of Nonlinear Dynamics</i> (Identificação Esparsa de Dinâmica Não Linear)
SLS	<i>Standard Linear Solid</i> (Sólido Linear Padrão)
SRM	<i>Structural Risk Minimization</i> (Minimização do Risco Estrutural)
STLSQ	<i>Sequential Thresholded Least Squares</i>
SVM	<i>Support Vector Machine</i>

LISTA DE SÍMBOLOS

A	Área da seção transversal
b	Largura da seção transversal
c	Amortecimento
C	Termo de regularização que estabelece o equilíbrio entre a complexidade do modelo e o erro de treinamento
$[C]$	Matriz de amortecimento proporcional
d	Porcentagem do dano que corresponde a sua intensidade
$d(x, y)$	Função de distância para as instâncias x e y
E	Módulo de elasticidade do material
$f(t)$	Função que especifica a dinâmica do sistema
$f(x, \dot{x}), g(x, v)$	Função que representa a aceleração em função do tempo
$g(x)$	Função de decisão
h	Altura da seção transversal
H	Hiperplano
I	Momento de inércia da seção transversal
k	Rigidez da mola
$[K]$	Matriz de rigidez
L	Comprimento da barra e da viga
M	Número de medidas observadas
m	Massa da mola
$[M]$	Matriz de massa
N	Número de atributos
n	Número de variáveis de estado
$P(c_i x)$	Probabilidade posterior ou <i>a posteriori</i>
$P(x c_i)$	Função de verossimilhança ou função de densidade de probabilidade condicional
$P(c_i)$	Probabilidade a priori da classe
$P(x)$	Evidência
p	Especifica uma única função candidata
$p(x, t)$	Força de excitação distribuída

$p(t)$	Forças externas em função do tempo
q	Número de classes
Q	Número total de funções distintas na biblioteca
r	Número de cenários de dano
$R(\cdot)$	Termo de regularização que promove soluções esparsas
s	Valor para corresponder a diferentes métricas
S	Conjunto de treinamento
t	Tempo
$u(x, t)$	Deslocamento longitudinal da barra
$v(x, t)$	Deflexão transversal
w	Vetor de pesos normal ao hiperplano
x e y	Instâncias
$x(t), u(t)$	Deslocamento em função do tempo
$\dot{x}(t), \dot{u}(t), v$	Velocidade em função do tempo
$\ddot{x}(t), \ddot{u}(t), \dot{v}$	Aceleração em função do tempo
$\mathbf{x}(t)$	Vetor de estados
$\dot{\mathbf{x}}$	Primeira derivada do vetor de estados
\mathbf{X}	Matriz de estados
$\dot{\mathbf{X}}$	Matriz da primeira derivada de estados
$X_{feature}$	Atributos
X_{data}	Conjunto de dados final
y_{label}	Rótulos
Ξ	Matriz de esparsidade
ξ	Vetor esparso
ξ_i	Variáveis de folga
$\mathcal{L}(\mathbf{w}, b, \alpha)$	Função lagrangiana
η	Escalar que modula a redução da rigidez da mola
$\Theta(\mathbf{X})$	Matriz de biblioteca
$\Phi(x)$	mapeamento não linear
$\phi_i(\mathbf{x})$	Funções candidatas
ρ	Massa específica do material
ρ_{lm}	Largura da margem
μ_i	Média da i -ésima classe

ω	Classe
Σ_i	Matriz de covariância da i -ésima classe
$ \Sigma_i $ e Σ_i^{-1}	Determinante e inverso da matriz de covariância, respectivamente
$\ \cdot\ _F$	Norma de Frobenius
$\ \cdot\ _2$	Norma-2 de um vetor
$\ \cdot\ _1$	Norma-1 de um vetor
λ	Hiperparâmetro que determina a força da regularização
δ	Limite de esparsidade
α, β	Constantes de proporcionalidade
α_i	Multiplicadores de Lagrange

SUMÁRIO

1 INTRODUÇÃO	22
1.1 Justificativa.....	25
1.2 Contribuições da Tese	27
1.3 Objetivos.....	27
1.4 Metodologia	27
1.5 Organização da Tese.....	31
2 DANO E MONITORAMENTO ESTRUTURAL, MACHINE LEARNING E DIGITAL TWIN.....	32
2.1 Dano e Monitoramento Estrutural.....	32
2.2 Machine Learning	33
2.2.1 <i>k-Nearest Neighbors</i>	36
2.2.2 <i>Naive Bayes</i>	40
2.2.3 <i>Discriminant Analysis</i>	42
2.2.4 <i>Support Vector Machine</i>	43
2.2.5 Métodos Baseados em Árvores de Decisão	53
2.2.5.1 <i>Decision Trees</i>	53
2.2.5.2 <i>Random Forest</i>	55
2.2.5.3 <i>Extremely Randomized Trees</i>	55
2.3 Digital Twin.....	56
3 IDENTIFICAÇÃO ESPARSA DE DINÂMICA NÃO LINEAR.....	59
3.1 Otimizadores	62
3.1.1 <i>Least Absolute Shrinkage and Selection Operator</i>	62
3.1.2 <i>Sequential Thresholded Least Squares</i>	63
3.2 Equações de Movimento.....	63
4 EXEMPLOS DE APLICAÇÃO: VIBRAÇÃO AXIAL.....	65
4.1 Procedimento para Aplicação Prática das Técnicas.....	65
4.1.1 <i>Physical Twin</i>	65
4.1.2 Modelo Baseado na Física.....	66
4.1.3 Respostas do SINDy.....	67
4.1.4 Parametrização de Dano	68
4.1.5 Construção do Conjunto de Dados.....	70
4.1.6 Construção dos Classificadores de <i>Machine Learning</i>	71

4.2 Barra Engastada	73
4.2.1 Barra Engastada Discretizada em Dois Elementos Finitos	73
4.2.1.1 <i>Respostas do SINDy</i>	74
4.2.1.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	77
4.2.1.3 <i>Aplicações do Framework Digital Twin</i>	80
4.2.2 Barra Engastada Discretizada em Quatro Elementos Finitos.....	81
4.2.2.1 <i>Respostas do SINDy</i>	82
4.2.2.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	85
4.2.2.3 <i>Aplicações do Framework Digital Twin</i>	87
4.2.3 Barra Engastada Discretizada em Seis Elementos Finitos	88
4.2.3.1 <i>Respostas do SINDy</i>	89
4.2.3.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	92
4.2.3.3 <i>Aplicações do Framework Digital Twin</i>	94
4.3 Barra Biengastada	96
4.3.1 Barra Biengastada Discretizada em Dois Elementos Finitos	96
4.3.1.1 <i>Respostas do SINDy</i>	96
4.3.1.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	98
4.3.2 Barra Biengastada Discretizada em Quatro Elementos Finitos	99
4.3.2.1 <i>Respostas do SINDy</i>	100
4.3.2.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	103
4.3.2.3 <i>Aplicações Framework Digital Twin</i>	104
4.3.3 Barra Biengastada Discretizada em Seis Elementos Finitos	106
4.3.3.1 <i>Respostas do SINDy</i>	106
4.3.3.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	109
4.3.3.3 <i>Aplicações do Framework Digital Twin</i>	111
5 EXEMPLOS DE APLICAÇÃO: VIBRAÇÃO TRANSVERSAL	113
5.1 Procedimento para Aplicação Prática das Técnicas.....	113
5.1.1 <i>Physical Twin</i>	113
5.1.2 Modelo Baseado na Física.....	114
5.1.3 Respostas do SINDy.....	115
5.1.4 Parametrização de Dano	116
5.1.5 Construção do Conjunto de Dados.....	119
5.1.6 Construção dos Classificadores de <i>Machine Learning</i>	119
5.2 Viga Engastada	120

5.2.1 Viga Engastada Discretizada em Dois Elementos Finitos	120
5.2.1.1 <i>Respostas do SINDy</i>	121
5.2.1.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	123
5.2.1.3 <i>Aplicações do Framework Digital Twin</i>	129
5.2.2 Viga Engastada Discretizada em Quatro Elementos Finitos.....	132
5.2.2.1 <i>Respostas do SINDy</i>	133
5.2.2.2 <i>Resultados e Discussões dos Métodos de Machine Learning</i>	136
5.2.2.3 <i>Aplicações do Framework Digital Twin</i>	142
6. CONCLUSÕES E SUGESTÕES DE TRABALHOS FUTUROS	145
6.1 Conclusões	145
6.2 Sugestões de Trabalhos Futuros	146
REFERÊNCIAS	147
APÊNDICE: RESPOSTAS DO <i>FRAMEWORK DT</i> DO CLASSIFICADOR SVM	162

1 INTRODUÇÃO

Ao longo da vida útil as estruturas se encontram sujeitas a diversas solicitações que podem causar dano estrutural, com potencial de reduzir significativamente tanto sua integridade quanto sua funcionalidade, podendo ainda originar e/ou propagar problemas estruturais relevantes, implicando em custo financeiro significativos e, no limite, acarretar a falha estrutural. Por vezes, torna-se necessário realizar o monitoramento da estrutura ao longo do tempo para avaliar a sua integridade, de forma que possibilite identificar o estado atual do dano, para evitar falhas inesperadas. Algumas técnicas tradicionais utilizadas para investigação da integridade estrutural dependem do julgamento humano, podendo produzir informações tendenciosas e, na maioria dos casos, é necessário avaliar toda a estrutura, ou pelo menos os locais que indicam maior probabilidade de dano, porém, existem situações em que o acesso a certos pontos da estrutura apresenta custo elevado e risco com relação à segurança da equipe técnica responsável pela avaliação. Para superar essas limitações, muitos pesquisadores se concentraram no desenvolvimento de sistemas de monitoramento baseados em vibração para avaliar a integridade estrutural (FUGATE, SOHN e FARRAR, 2001; MAGALHÃES, CUNHA e CAETANO, 2012).

Os métodos baseados em vibração têm atraído atenção significativa, permitindo avaliar o comportamento dinâmico global da estrutura a partir da aplicação de excitação em determinados pontos da estrutura. Numerosos estudos de identificação de danos baseados em vibração podem ser encontrados na literatura. Esses métodos consistem essencialmente em extrair um padrão de referência da resposta de vibração da estrutura íntegra, para comparação com os padrões de resposta de vibração da estrutura com dano.

O dano produz mudanças nas propriedades físicas da estrutura e, consequentemente, em suas características dinâmicas (que estão correlacionados com a massa e a rigidez). Assim, os danos presentes em uma estrutura provocam alterações dos seus parâmetros modais, que incluem as frequências naturais, os modos naturais de vibração e os fatores de amortecimento. Dessa forma, muitos trabalhos foram desenvolvidos utilizando métodos que consideram a mudança no comportamento dinâmico, incluindo variações nas frequências naturais (CAWLEY e ADAMS, 1979; PENNY, WILSON e FRISWELL, 1993; SALAWU, 1997; SAMMAN e BISWAS (1994a-b); YANG e WANG, 2010), dados de modos de vibração (LIEVEN e EWINS, 1988; PANDEY, BISWAS e SAMMAN, 1991; YAZDANPANAH, SEYEDPOOR e BENGAR, 2015), flexibilidade modal (PANDEY e BISWAS, 1994) e energia

de deformação modal (MORADIPOUR, CHAN e GALLAGE, 2015; SHI e LAW, 1998; SHI, LAW e ZHANG, 2020; WANG e LI, 2012; WU *et al.*, 2017).

Na abordagem baseada em vibração, apesar das vantagens notáveis, a avaliação precisa dos danos ainda requer a expertise de especialistas para realizar a análise modal e extrair as características dinâmica das estruturas. Além disso, os parâmetros modais podem ser facilmente afetados pela presença de ruído ou mudanças nas condições ambientais.

Alternativamente, nos últimos anos, o desenvolvimento de técnicas de reconhecimento de padrões, mais especificamente as abordagens baseadas em *machine learning* (ML), proporcionou ferramentas com aplicações importantes em vários campos, alcançando resultados promissores em problemas de identificação de danos. O *machine learning* corresponde à habilidade de uma máquina de adquirir seu próprio conhecimento automaticamente a partir dos dados (MÜLLER e GUIDO, 2017). Os algoritmos de ML têm sido bem-sucedidos ao aprender com processos passados, que dependem de diversos padrões de dados e métodos de reconhecimento. Esses algoritmos são particularmente eficazes no tratamento de problemas que envolvem relações não lineares e dados contaminados por ruído, que podem ser desafiadores para o discernimento humano (KHAN e YAIRI, 2018; YAGER e ZADEH, 2012), podendo permitir a identificação de dano mesmo com informações limitadas sobre a estrutura. Cada algoritmo tem sua própria técnica e hiperparâmetros que podem ser ajustados para conseguir um melhor desempenho na identificação de dano.

Nesse contexto, a integração das metodologias de *digital twin* (DT) e *machine learning* pode ser muito promissora, uma vez que possibilita a criação de melhores modelos de automação, que podem ser usados para exibir previsões de alta qualidade, fornecendo suporte à tomada de decisão. O *digital twin* é uma representação digital virtual de um ativo físico (*physical twin*) (GRIEVES, 2014), replicando-o digitalmente com determinados parâmetros, que consistem em informações geométricas, físicas e comportamentais (QI *et al.*, 2021), de modo que com o uso de *big data*, sinais de sensores, inteligência artificial e *machine learning* possibilita simular diferentes cenários para entender seu desempenho (MENDI, 2022; RATHORE *et al.*, 2021).

Uma representação virtual consiste tradicionalmente em um modelo computacional baseado na física ou modelo orientado a dados (com técnicas de *machine learning*) (LI *et al.*, 2022; VANDERHORN e MAHADEVAN, 2021). A modelagem baseada na física geralmente descreve o sistema a ser modelado usando um conjunto de equações governantes que representam fenômenos físicos conhecidos e compreendidos (GOMEZ-CABRERA e ESCAMILLA-AMBROSIO, 2022). A modelagem orientada a dados, usando técnicas de

machine learning, se baseia na suposição de que os dados são uma manifestação da física conhecida e desconhecida, de forma que podem ser usados para inferir modelos físicos interpretáveis, e desempenham um papel importante, uma vez que a modelagem baseada na física geralmente requer conhecimento detalhado do sistema, o que muitas vezes é difícil, pois apenas informações parciais do sistema são conhecidas (RASHEED, SAN e KVAMSDAL, 2020).

Alguns trabalhos foram desenvolvidos na tentativa de usar *machine learning* para inferir modelos físicos a partir de dados. Nos trabalhos de Bongard e Lipson (2007) e Schmidt e Lipson (2009), a regressão simbólica foi aplicada para inferir a dinâmica não linear do sistema. O método funciona notavelmente bem para descobrir modelos físicos interpretáveis, mas a regressão simbólica é computacionalmente cara e pode ser difícil de dimensionar para grandes problemas (MONTÁNS *et al.*, 2019). Uma abordagem alternativa baseada em processos (BRIDEWELL *et al.*, 2008; TANEVSKI, TODOROVSKI e DŽEROSKI, 2016; TANEVSKI *et al.*, 2017) permite especificar uma biblioteca de relacionamentos ou funções entre variáveis com base no conhecimento do domínio para produzir um conjunto interpretável de equações governantes. Um método que emprega regressão esparsa (*sparse identification of nonlinear dynamics* – SINDy) foi proposto por Brunton, Proctor e Kutz (2016) para encontrar o menor número de termos em uma biblioteca de funções candidatas necessárias para modelar a dinâmica de sistemas, o que permite que uma classe maior de funções seja considerada, quando comparado com o método baseado em processos.

O SINDy mostrou ser capaz de produzir modelos de sistemas dinâmicos interpretáveis e generalizáveis a partir de dados limitados (BRUNTON, PROCTOR e KUTZ, 2016) e tem sido amplamente aplicado para identificar modelos para sistemas ópticos (SOROKINA, SYGLETOS e TURITSYN, 2016), convecção de plasma (DAM *et al.*, 2017), fluxos de fluidos (LOISEAU e BRUNTON, 2018), controle preditivo de modelo (KAISER, KUTZ e BRUNTON, 2018), dinâmica de reação química (HOFFMANN, FRÖHNER e NOÉ, 2019), modelagem estrutural (LAI e NAGARAJAIAH, 2019; MOKHTARI e IMANPOUR, 2024) e modelagem do movimento de objetos em queda (DE SILVA *et al.*, 2020).

No entanto, o *digital twin* requer modelos que sejam generalizáveis, confiáveis, autoevolutivos e computacionalmente eficientes, mantendo boa acurácia (SAN, RASHEED e KVAMSDAL, 2021). É desafiador conseguir essas características de modelagem usando uma das duas abordagens de modelagem tradicionais. A modelagem baseada na física requer conhecimento para descrever adequadamente os fenômenos do sistema e geralmente é estática no sentido de que não é atualizado automaticamente para novos cenários encontrados após a

implantação do modelo. A modelagem orientada a dados pode apresentar limitação de generalização dos modelos, especialmente para cenários de extração, e pode ser um desafio estabelecer quais físicas são modeladas (BLAKSETH *et al.*, 2022).

As características de modelagem identificadas por San, Rasheed e Kvamsdal (2021) podem ser alcançadas por meio do modelo híbrido, que combina o modelo baseado na física e o modelo orientado a dados em uma arquitetura conjunta, de uma forma que potencialize seus pontos fortes (KURZ *et al.*, 2022), fornecendo um modelo aprimorado que combina interpretabilidade, robustez, acurácia e eficiência (RASHEED, SAN e KVAMSDAL, 2020).

1.1 Justificativa

O monitoramento contínuo da integridade estrutural com metodologias e técnicas adequadas pode permitir a identificação de danos em um estágio inicial para reduzir consideravelmente o custo de manutenção, aumentar a durabilidade e a vida útil da estrutura, garantir a segurança e evitar a falha do elemento estrutural.

Nesse contexto, com as melhorias no poder computacional e os avanços nas tecnologias de sensores, *hardware* e *software* de aquisição de dados, as técnicas de *machine learning* e *digital twin* apresentam grande potencial, uma vez que permitem a criação de modelos sob uma tendência mais digital e possibilitam simular diferentes cenários para o *physical twin* a ser monitorado. Dessa forma, o sistema de monitoramento da integridade estrutural (*structural health monitoring* - SHM) com as referidas técnicas acompanha o desempenho do *physical twin* em condições operacionais e fornece suporte de decisão para melhorar a gestão e confiabilidade das estruturas. No caso do monitoramento de *physical twin*, duas principais abordagens de modelagem podem ser implementadas, modelagem baseada na física e modelagem orientada a dados. A abordagem de modelo híbrido, que considera os princípios das duas abordagens, pode ser usada para superar as limitações das duas abordagens.

Bigoni e Hesthaven (2020) propuseram uma metodologia de identificação de anomalias baseada em simulação, que combina modelo baseado na física, dados de sinais sintéticos e técnicas de *machine learning* para construir classificadores separados de uma classe para distinguir os dados gerados a partir de uma estrutura íntegra dos dados gerados a partir de uma estrutura com dano. Ritto e Rochinha (2021) propuseram a integração de modelo baseado na física com *machine learning* para construir um DT para detecção de danos em uma barra engastada de aço. Svendsen *et al.* (2023) desenvolveram uma abordagem de SHM para identificação de danos em pontes, usando dados de um modelo numérico de elementos finitos

e dados experimentais com *machine learning*. Rhee *et al.* (2024) apresentaram uma estratégia que integra diferentes algoritmos de ML e um novo método de filtragem, desenvolvido para melhorar o processamento de sinais, para predizer o estado de dano em vigas de aço. Os dados para treinamento e teste foram obtidos por meio de ensaios de vibração em vigas de aço. Ruiz *et al.* (2024) apresentaram uma estratégia baseada em ML para detecção de danos em viga de alumínio, utilizando função de resposta em frequência (FRF), tanto para simulação numérica quanto para avaliação com dados experimentais. Torzoni *et al.* (2024) propuseram um *framework* DT para estruturas de engenharia, onde a estratégia apresentada depende de um modelo gráfico probabilístico. Os dados de treinamento foram gerados a partir de um modelo numérico de ordem reduzida e o diagnóstico da integridade estrutural foi indicado pela assimilação de dados detectados com modelos de aprendizado profundo. A estratégia foi avaliada em dois estudos de caso sintéticos, envolvendo uma viga e uma ponte, demonstrando a capacidade do DT. Em Siow *et al.* (2024) foi proposta uma estratégia de integração de um método baseado em parâmetros modais em um método híbrido baseado em ML (métodos não supervisionados e supervisionado) para detecção e localização de danos em estruturas semelhantes a vigas. Vu, Thom e Tran *et al.* (2024) utilizaram métodos de ML para detecção de dano em vigas de aços, usando as flutuações nas frequências naturais sob vários cenários de danos de um modelo dos elementos finitos (MEF).

Considerando a potencialidade existente nas técnicas supramencionadas, este trabalho propõe um novo *framework* para o monitoramento da integridade estrutural, mais especificamente para identificação de danos em elementos estruturais lineares de aço (barras comprimidas e vigas). O *framework* integra um modelo híbrido interpretável, que combina modelagem física e modelagem orientada a dados, e métodos de *machine learning* para classificação, para a construção de um *framework digital twin*, no contexto de sistemas dinâmicos. O *framework digital twin* construído relacionará as entradas do *physical twin* a cenários de danos específicos, para avisar rapidamente se houver dano, onde está localizado e qual a sua intensidade. Dessa forma, são abordados, parcial e implicitamente, três principais níveis de investigação do dano estrutural: detecção, localização e quantificação. Com essas informações, poderão ser estabelecidos critérios para a utilização da estrutura com segurança, fornecendo suporte para melhor identificar quando são necessárias intervenções e ações de manutenção, contribuindo para a extensão da vida útil e redução de custos financeiros.

1.2 Contribuições da Tese

Dentre as principais contribuições da tese, destacam-se: (i) construção de um *framework digital twin* para SHM, no contexto da dinâmica estrutural, mais especificamente para identificação de dano estrutural; (ii) proposição de um modelo híbrido que combina modelo físico e modelo orientado a dados, para resultar em modelos de maior interpretação física e mais robusto, e apresentação de uma estratégia para simular diferentes cenários de dano a partir desse modelo híbrido; (iii) avaliação de uma estratégia para combinar modelo híbrido com métodos de *machine learning*, com exploração de diferentes métodos e a determinação dos respectivos hiperparâmetros ótimos, no contexto da identificação de dano.

1.3 Objetivos

O objetivo geral deste trabalho consiste no desenvolvimento de um *framework* que integra modelo híbrido, baseado na física e orientado a dados, e *machine learning* para construção de um *framework digital twin* para monitoramento da integridade estrutural, avisando se o elemento estrutural de interesse apresenta dano, onde está localizado e qual a sua intensidade.

O objetivo geral é alcançado através dos seguintes objetivos específicos:

- a) Desenvolver um modelo híbrido que represente as equações governantes do sistema de interesse a partir da combinação de um modelo físico e modelo orientado a dados;
- b) Comparar a robustez de diferentes métodos de *machine learning* para classificação;
- c) Verificar a robustez dos métodos para diferentes locais de danos e intensidades correspondentes.

1.4 Metodologia

A metodologia empregada para o desenvolvimento deste trabalho foi realizada em quatro principais macroetapas: descoberta das equações governantes do movimento da estrutura, parametrização de danos, construção dos classificadores de *machine learning* e construção do *framework digital twin*.

A primeira macroetapa correspondeu ao desenvolvimento de modelos que representassem as equações governantes do movimento da estrutura, combinando equações diferenciais parciais, que descrevem a física do sistema, com modelagem orientada a dados,

para compensar os parâmetros desconhecidos. Dessa forma, o SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física. Os dados de deslocamento ($x_i(t)$), velocidade ($\dot{x}_i(t)$) e aceleração ($\ddot{x}_i(t)$) da estrutura, em diferentes tempos (t_1, t_2, \dots, t_M), foram usados para construir a matriz de estados (X), a primeira derivada da matriz de estados (\dot{X}) e a matriz de biblioteca $\Theta(X)$, de acordo com um conjunto de funções candidatas especificadas. Em seguida, foi determinada a matriz de esparsidade (Ξ) e, consequentemente, as equações governantes (Figura 1(a)).

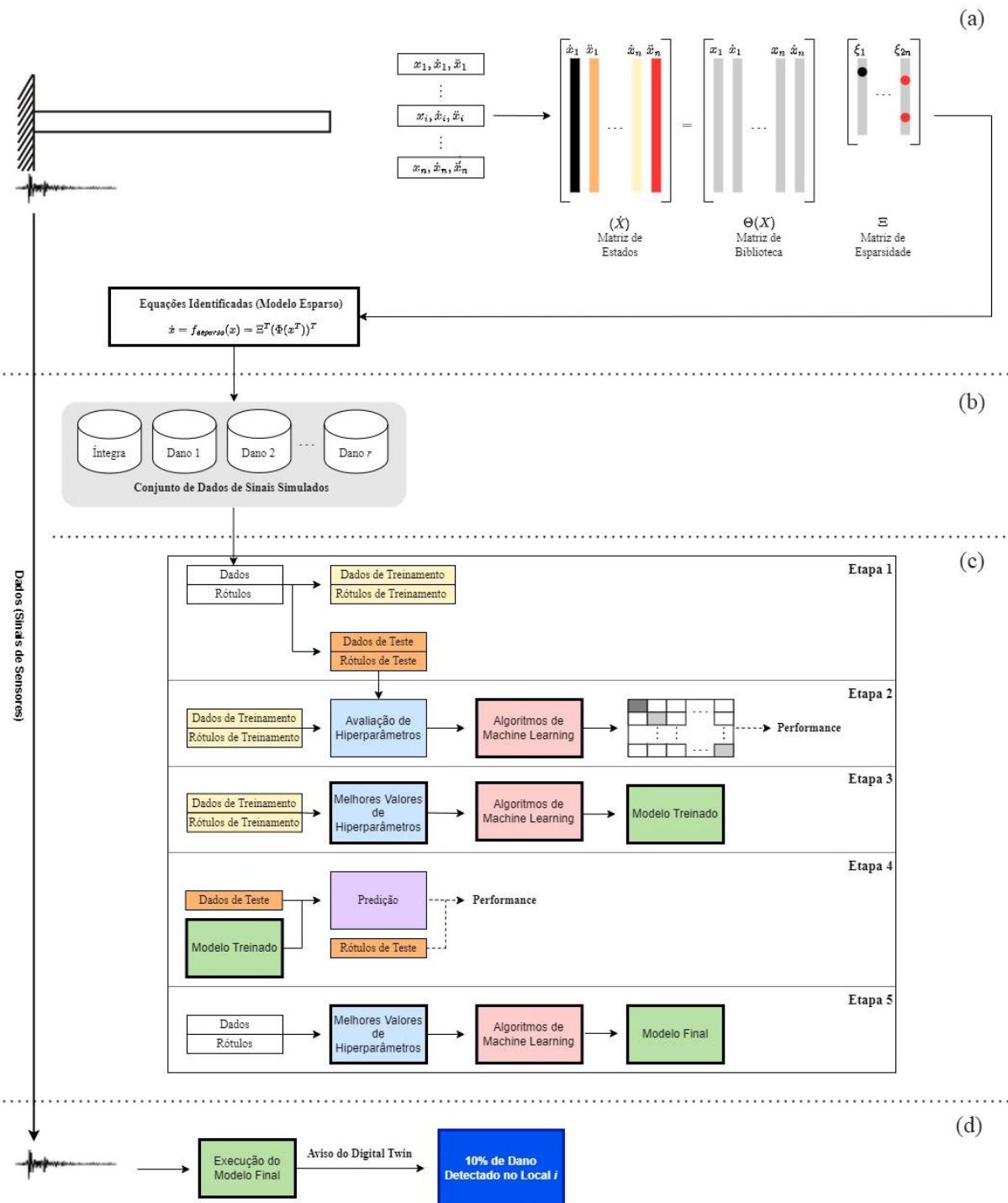
Na segunda macroetapa foram usadas as equações governantes descobertas, correspondentes às equações governantes da estrutura íntegra, para simular a resposta do sistema com diferentes locais de danos e intensidades correspondentes. Uma vez que foi usada uma estrutura equacional de um modelo físico, foi possível representar o dano pela redução da rigidez local (Figura 1(b)).

A terceira macroetapa correspondeu à construção do conjunto de dados, considerando os cenários da estrutura íntegra e com dano, para aplicação do classificador de *machine learning*. Os códigos para os classificadores foram desenvolvidos usando o pacote *scikit-learn* da linguagem Python 3.9. Foram consideradas as seguintes etapas: (1) os dados de entrada foram divididos em dois subconjuntos, de treinamento (90% dos dados) e de teste (10% dos dados); (2) os dados de treinamento foram usados para avaliar diferentes configurações de hiperparâmetros, para cada configuração de hiperparâmetro foi aplicado o método de validação cruzada *k-fold* (HASTIE, TIBSHIRANI e FRIEDMAN, 2017; KUHN e JOHNSON, 2013), resultando em vários modelos e estimativas de desempenho; (3) considerando as configurações de hiperparâmetros que apresentaram os melhores resultados no procedimento de validação cruzada *k-fold*, foi usado o conjunto de treinamento completo para ajustar o modelo com essas configurações; (4) o conjunto de teste, retido na etapa 1, foi usado para avaliar o modelo obtido na etapa 3; (5) após conclusão da etapa de avaliação, o modelo foi ajustado a todos os dados de entrada (conjuntos de dados de treinamento e teste) (RASCHKA, 2020). Quando os métodos não tinham hiperparâmetros para ajustar, as etapas consistiram, essencialmente, em: (1) utilizar o método de validação cruzada *k-fold* no conjunto de dados de entrada (dados de treinamento e teste) para avaliar o modelo; (2) ajustar o modelo a todos os dados de entrada (Figura 1(c)).

A última macroetapa correspondeu à construção do *framework digital twin* (Figura 1(d)). O *framework* DT possui uma arquitetura constituída por quatro camadas: parte física, armazenamento de dados, modelagem de DT e serviço (Figura 2). A parte física (*physical twin*) é representada por um modelo de alta fidelidade, correspondente a uma barra prismática e uma

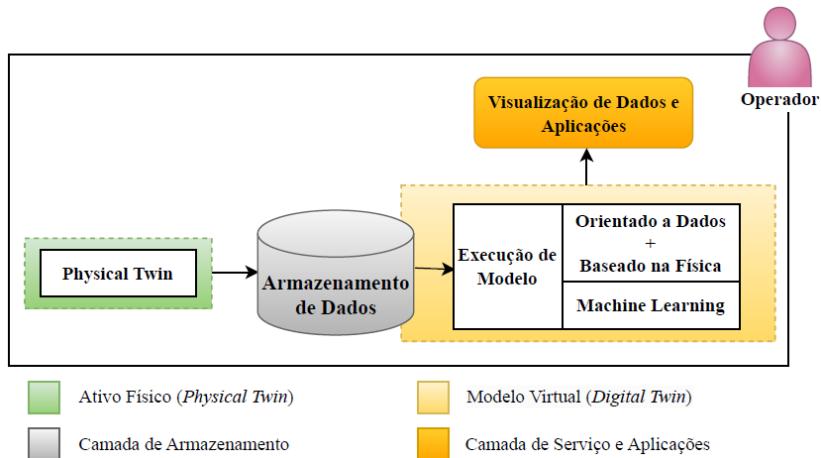
viga prismática governadas por equações diferenciais parciais, para gerar a resposta dinâmica da estrutura, produzindo dados sintéticos que seriam provenientes de sensores, sendo a base de construção da parte virtual; a segunda camada é o armazenamento de dados, condição prévia para a criação de novos conhecimentos; a terceira camada é a modelagem de DT, que suporta a simulação e controla a parte física; a quarta camada é a de serviço, responsável pelo desenvolvimento de aplicações orientada a dados ou funções de análise de dados padrão para fornecer informações ao usuário que possibilite a tomada de decisão, podendo aumentar a confiabilidade e produtividade do sistema de engenharia.

Figura 1 – Procedimento metodológico proposto para a Tese.



Fonte: Autor (2024).

Figura 2 – Framework DT com quatro camadas.



Fonte: Adaptado de Singh, Weebera e Birke (2021).

1.5 Organização da Tese

O texto desta Tese está estruturado em seis capítulos. O primeiro capítulo consiste na parte introdutória ao tema da pesquisa, com apresentação da contextualização, justificativa, contribuições, objetivos e metodologia.

O segundo capítulo apresenta conceitos relativos a dano e monitoramento estrutural, *machine learning* e *digital twin*.

A descrição do método identificação esparsa de dinâmica não linear (*sparse identification of nonlinear dynamics* – SINDy) para descobrir equações governantes de sistema dinâmico a partir de dados é abordado no terceiro capítulo.

O quarto capítulo apresenta as aplicações das técnicas para um modelo de barra de aço, considerando a vibração axial.

O quinto capítulo apresenta as aplicações das técnicas para um modelo de viga em vibração transversal.

O sexto capítulo consiste nas conclusões do trabalho realizado e sugestões para trabalhos futuros científicos.

No Apêndice são apresentadas as respostas da aplicação do *framework digital twin*.

2 DANO E MONITORAMENTO ESTRUTURAL, MACHINE LEARNING E DIGITAL TWIN

2.1 Dano e Monitoramento Estrutural

Os danos estruturais são intrínsecos às estruturas de engenharia e são predominantemente propensos a se propagar devido às ações estáticas (peso próprio e ações com variação lenta), ações dinâmicas (por exemplo, vento, tráfego rodoviário e/ou ferroviário, carga de impacto, sismos e explosões) e outros fatores (mecânicos, físicos, biológicos, químicos e ambientais).

O dano pode ser definido como uma mudança na propriedade do material ou nas características geométricas dos elementos estruturais, que afeta adversamente seu desempenho, segurança e confiabilidade. A mecânica do dano investiga a degradação da estrutura, considerando que o dano se manifesta na escala macroscópica, como uma perda de rigidez local. Nesse contexto, o dano pode estar presente em toda a estrutura, caracterizando o dano global, ou apenas em alguns pontos específicos, originando o dano local, sendo a ocorrência desse último mais comum. Neste conceito, encontra-se inserida a comparação entre os estados inicial (denominado de estado íntegro ou intacto) e final (denominado de estado com dano).

Esses danos podem reduzir significativamente a durabilidade e a vida útil dos elementos estruturais e, com o decorrer do tempo, podem provocar fadiga em seus materiais e afetar o funcionamento da estrutura, no que se refere ao conforto e segurança de seus usuários. Pode, ainda, vir a se tornar o fator responsável pelo surgimento de problemas estruturais de maior relevância, gerando custos significativos e, no limite, acarretar a falha estrutural.

Dessa forma, é notória a necessidade da realização de monitoramento ao longo do tempo para avaliação da integridade estrutural. As abordagens iniciais referentes ao monitoramento de danos estruturais consistiam na inspeção visual para identificar e avaliar, se houver, as anomalias, falhas e/ou danos e evoluíram com o tempo, com o aperfeiçoamento nas tecnologias de sensores, *hardware* e *software* de aquisição de dados. O monitoramento da integridade estrutural é um campo de pesquisa amplo e interdisciplinar que pode envolver medição contínua das condições ambientais e operacionais, ensaios experimentais, identificação de sistema e aquisição e gerenciamento de dados. O componente mais crítico do SHM é a identificação de danos (FARRAR, DOEBLING e NIX, 2001), que se baseia na utilização de indicadores confiáveis e robustos que permitam detectar, localizar e quantificar

danos, além de, se possível, estimar a vida útil residual da estrutura após a caracterização do dano.

Durante décadas, os sistemas de SHM produziram grandes quantidades de dados de monitoramento para controlar o comportamento estrutural durante a operação (FARRAR e LIEVEN, 2007). Com as melhorias das tecnologias de sensores, poder computacional, *machine learning* e internet das coisas (*internet of things* - IoT), foi desenvolvido um novo paradigma de avaliação de condições que fundem informações de dados e os modelos sob uma tendência mais digital, denominado *digital twin*, que consiste essencialmente em um objeto físico, réplica digital e a conexão (ATTARAN e CELIK, 2023; CHAKRABORTY, ADHIKARI e GANGULI, 2021). O *digital twin* é uma das tecnologias digitais de rápida evolução que suportam a transformação digital da engenharia estrutural para permitir o suporte de decisão para melhorar o gerenciamento, confiabilidade e sustentabilidade das estruturas (CHIACHÍO *et al.*, 2022; LOVE e MATTHEWS, 2019).

Entre as inúmeras abordagens para SHM, o modelo DT está ganhando cada vez mais atenção. Esse modelo é uma reconstrução digital de um ativo físico (*physical twin*) e é atualizado frequente e automaticamente, usando dados provenientes de uma rede de sensores instalados no *physical twin*. Essa ferramenta pode fornecer funcionalidades para monitorá-lo e otimizá-lo, tomar decisões informadas e baseadas em dados, no contexto das condições operacionais cotidianas e após eventos extremos, e realizar predições comportamentais estruturais e, em certas condições, compensá-las automaticamente (BADO *et al.*, 2022). De um modo geral, a eficácia de uma abordagem de SHM é tão boa quanto sua capacidade de detectar em tempo hábil o aumento de criticidades e danos à estrutura (BROWNJOHN, 2007).

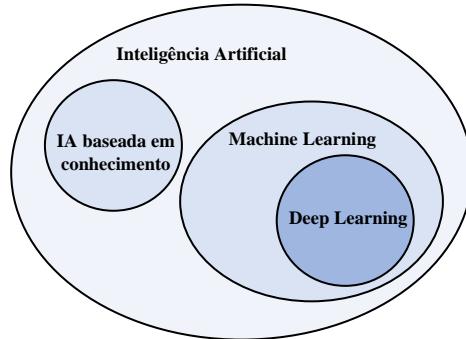
A identificação do dano, bem como a quantificação da sua intensidade, fornece importantes vantagens, por exemplo, o estabelecimento de critérios para a utilização da estrutura com segurança e a capacidade de melhor identificar quando são necessárias intervenções na estrutura, para que dessa forma se possa realizar convenientemente a manutenção na mesma, contribuindo para a extensão da vida útil e redução de custos.

2.2 Machine Learning

A inteligência artificial (IA) é um campo da ciência da computação que visa desenvolver atividades associadas ao pensamento humano, como tomada de decisão, resolução de problemas e aprendizagem (RUSSEL e NORVIG, 1995). As primeiras aplicações da IA foram as abordagens baseadas em conhecimentos, especialmente, direcionadas a problemas baseados

em regras (GOODFELLOW, BENGIO e COURVILLE, 2017). No entanto, os sistemas de IA baseado em conhecimento falharam em tarefas que parecem ser automáticas e diretas para um ser humano, como reconhecer rostos, detectar objetos e compreender a fala. Dessa forma, um grande desafio foi encontrar formas alternativas de ensinar para computadores conhecimento intuitivo e de senso comum, que não pode ser traduzido em uma lista explícita de regras (GHAHRAMANI, 2015). Para superar os desafios enfrentados pela abordagem baseada no conhecimento, o conceito de *machine learning* foi introduzido nos sistemas de IA (AVCI *et al.*, 2021). A IA baseada em conhecimento e o *machine learning* são um subconjunto da IA e o *deep learning* é um subconjunto especializado do *machine learning* (Figura 3).

Figura 3 – Diagrama de Venn apresentando a relação entre os diferentes sistemas de IA.



Fonte: Adaptado de Avci *et al.* (2021).

O *machine learning* é um campo de pesquisa na interseção de estatística, inteligência artificial e ciência da computação (MÜLLER e GUIDO, 2017). Como uma aplicação de IA, os algoritmos de ML fornecem aos sistemas a capacidade de aprender com um número suficiente de amostras de dados o conhecimento necessário para realizar uma tarefa específica, melhorar automaticamente o desempenho com a experiência e fazer previsões e/ou decisões acuradas sem ser explicitamente programado (LI *et al.*, 2018; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018).

Os algoritmos de ML geralmente podem ser classificados em algoritmos supervisionados e não supervisionados (SANTOS *et al.*, 2016). Os algoritmos supervisionados requerem um conjunto de dados com rótulos como alvos de aprendizado e seu principal objetivo é descobrir o mapeamento ideal das entradas para as saídas desejadas (KUBAT, 2017), sendo o tipo de ML mais utilizado (JORDAN e MITCHELL, 2015). Algoritmos de aprendizagem não supervisionados, por outro lado, requerem conjunto de dados não rotulados e dependem da compreensão dos próprios dados. As tarefas de aprendizado supervisionado mais comuns são

classificação e regressão, enquanto as tarefas de aprendizado não supervisionadas mais comuns são *clustering* e redução de dimensionalidade (ZHANG *et al.*, 2022).

A classificação corresponde a atribuir uma categoria a cada instância de dados, por exemplo, classificação de diagnóstico médico e imagem (AVCI *et al.*, 2021), e a regressão consiste em prever um valor real para cada instância de dados, por exemplo, predição de temperatura, idade e renda (ZHANG *et al.*, 2022). O *Clustering* visa dividir o conjunto de dados de entrada em grupos com características semelhantes, por exemplo, reconhecimento de padrões, e a redução de dimensão é o processo de transformar uma representação inicial de dados em uma representação de dimensão inferior, preservando algumas propriedades da representação inicial, por exemplo, pré-processamento de imagens digitais em tarefas de visão computacional (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018).

Este trabalho se concentra em técnicas de ML supervisionada de classificação, uma vez que os dados são rotulados e é de interesse realizar a classificação para novos dados de entrada. No aprendizado supervisionado o conjunto de dados é comumente dividido em dois subconjuntos. O primeiro é o conjunto de treinamento, usado para treinar um modelo, e o segundo é conjunto de teste, usado para avaliação de desempenho do modelo criado. Para dividir esses subconjuntos, os principais métodos utilizados na literatura são Holdout e validação cruzada *k-fold* (CAMPESATO, 2020).

No método Holdout os dados são divididos aleatoriamente em dois subconjuntos, sendo um de treinamento e o outro de teste. Comumente, é atribuído 2/3 dos dados ao conjunto de treinamento e 1/3 dos dados ao conjunto de teste. Se o conjunto de dados for relativamente grande, pode ser atribuído 90% dos dados para treinamento e 10% para teste. No entanto, essa abordagem utiliza apenas uma parte dos dados para treinamento, o que pode não conter dados mais representativos, comprometendo a confiabilidade dos resultados (RASCHKA, 2020).

Alternativamente, a validação cruzada *k-fold* consiste em dividir os dados em *k* subconjuntos mutuamente exclusivos de tamanhos aproximadamente iguais. O processo terá *k* iterações, onde, em cada iteração, um subconjunto é utilizado para teste e os restantes (*k* – 1) para treinamento. Como os subconjuntos são formados aleatoriamente, não se sabe se uma determinada classe estará em todos os subconjuntos, o que pode interferir tanto no treinamento quanto no teste do modelo. Dessa forma, a validação cruzada estratificada pode ser utilizada, de forma que todos os subconjuntos irão conter aproximadamente a mesma proporção de classes do conjunto de dados de entrada (RASCHKA, 2020).

Para avaliar o desempenho dos modelos, quatro métricas são amplamente utilizadas na literatura, acurácia, precisão, revocação (sensibilidade ou *recall*) e F-score (ALI, NEAGU e

TRUNDLE, 2019). A acurácia é a medida de desempenho mais intuitiva e definida como a razão entre o número de amostras classificadas corretamente e o número total de amostras avaliadas; enquanto a precisão é a razão entre o número de classificações corretas e o total de classificações de uma determinada classe; a revocação é definido pelo número de classificações corretas dividido pelo total de ocorrências de uma determinada classe; e, por fim, o F-score pode ser definido como uma média ponderada da precisão e da revocação, o seu valor igual a 1 representa um ótimo modelo, enquanto o valor igual a 0 é considera um modelo falho.

A seguir serão apresentados os algoritmos de classificação de *machine learning* que foram escolhidos para análise, com base em seu uso na literatura SHM e em sua interpretabilidade em termos de como usam atributos para predição de classes, a saber, *k-Nearest Neighbors* (k-NN), *Discriminant Analysis* (DA), *Naive Bayes* (NB), *Support Vector Machine* (SVM) e Métodos Baseados em Árvores de Decisão.

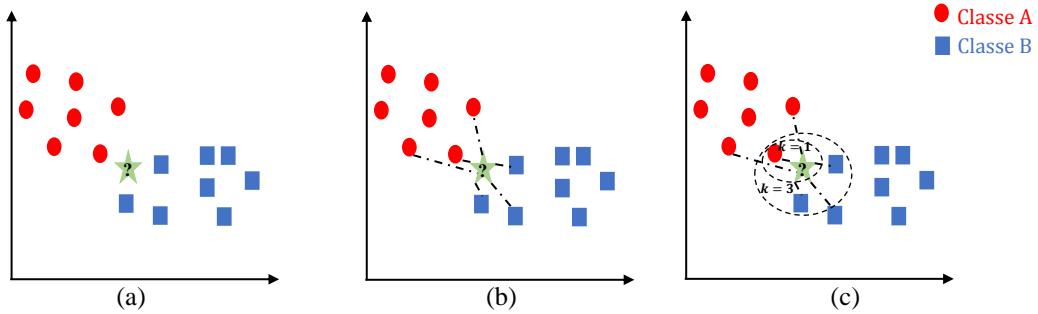
2.2.1 k-*Nearest Neighbors*

O *k-Nearest Neighbors* (k-NN) é um método não paramétrico usado para classificação de padrões e modelos de regressão (HASTIE, TIBSHIRANI e FRIEDMAN, 2017). Foi proposto por Fix e Hodges (FIX e HODGES, 1951) e modificado por Cover e Hart (COVER e HART, 1967). O método k-NN pertence à família de aprendizado baseado em instância, que é uma forma de aprendizado lento, em que nenhuma abstração é realizada nos dados de treinamento para criar um modelo generalizado. Dessa forma, todo o conjunto de dados é armazenado na memória, sendo esse conjunto a chamada instância do problema (WETTSCHERECK, AHA e MOHRI, 1997).

Para classificação, o método possui três componentes principais: um conjunto de observações rotuladas (conjunto de treino armazenado), uma métrica de distância (ou métrica de similaridade) e o valor de k (o número de vizinhos mais próximos). Nesse sentido, as principais etapas do método consistem, essencialmente, em: (1) determinar o número de vizinhos mais próximos (valor de k); (2) calcular a distância entre a nova amostra a ser classificada e as amostras de treinamento; (3) ordenar a distância e determinar os vizinhos mais próximos baseada na k-ésima distância mínima; (4) reunir os rótulos dos vizinhos mais próximos e classificar a amostra em análise como pertencente à classe predominante (Figura 4). Por exemplo, para a amostra em análise, se $k = 1$ será classificada como pertencente à classe da única amostra mais próxima e se $k > 1$ será classificada como pertencente a classe

predominante das amostras mais próximas (se $k = 3$, a classe predominante das três amostras mais próximas será atribuída) (Figura 4(c)).

Figura 4 – Aplicação do método k-NN: (a) dados iniciais, (b) calcular distância e (c) encontrar vizinhos e votar em atributos.



Fonte: Autor (2024).

Como a classificação é baseada no número de vizinhos, o valor de k é considerado um hiperparâmetro que determinará o desempenho do classificador. Diferentes valores podem ter um grande impacto na acurácia preditiva do método e escolher um bom valor geralmente não é intuitivo, olhando para o conjunto de dados (KANG, 2021). Ao escolher um valor pequeno para k , a classificação pode ser indevidamente afetada por *outliers* e sensíveis ao ruído. Com k pequeno, por exemplo, $k = 1$, a nova observação será classificada como pertencente à mesma classe da única observação mais próxima, um processo que pode levar o algoritmo a um *overfitting*, tendendo a memorizar o conjunto de dados de treinamento a custo da generalização. Por outro lado, escolher um valor de k que não seja muito pequeno tenderá a suavizar qualquer comportamento característico aprendido no conjunto de treinamento. No entanto, se for escolhido um valor de k muito grande, o comportamento localmente interessante pode ser ignorado. Então, o hiperparâmetro k é particular a cada problema (LAROSE e LAROSE, 2014). A otimização de k pode ser realizada por técnicas de reamostragem, como validação cruzada k -fold (HASTIE, TIBSHIRANI e FRIEDMAN, 2017; KUHN e JOHNSON, 2013), sendo escolhido o valor de k que minimiza o erro de classificação (HULETT, HALL e QU, 2012).

Para determinar os vizinhos mais próximos é utilizado o conceito de distância entre a instância a ser classificada e as instâncias do conjunto de treinamento. A função de distância $d(x, y)$ entre duas instâncias x e y , onde $x = (x_1, x_2, \dots, x_N)$ e $y = (y_1, y_2, \dots, y_N)$, onde N representa o número de atributos, é considerada uma métrica se satisfizer as seguintes condições (DEZA e DEZA, 2009):

1. *Não negatividade*: a distância entre x e y é sempre um valor maior ou igual a zero,

$$d(x, y) \geq 0. \quad (1)$$

2. *Identidade dos indiscerníveis*: a distância entre x e y é igual a zero se e somente se x é igual a y ,

$$d(x, y) = 0 \text{ se e somente se } x = y. \quad (2)$$

3. *Simetria*: A distância entre x e y é igual à distância entre y e x (comutatividade),

$$d(x, y) = d(y, x). \quad (3)$$

4. *Desigualdade triangular*: Considerando a presença de um terceiro ponto z , a distância entre x e y é sempre menor ou igual à soma da distância entre x e z e a distância entre y e z ,

$$d(x, y) \leq d(x, z) + d(z, y). \quad (4)$$

Dadas as instâncias x e y , com atributos numéricos, existem diferentes definições matemáticas de distâncias para medir a proximidade entre elas, como Minkowski, Manhattan (City Block), Euclidiana, Chebyshev e Cosseno (ALFEILAT *et al.*, 2019) (Quadro 1). Outras métricas de distância podem ser encontradas em Alfeilat *et al.* (2019). Dentre as métricas citadas, a Euclidiana é a função de distância mais utilizada com k-NN (WEINBERGER e SAUL, 2009; LAROSE e LAROSE, 2014). A distância de Minkowski é uma métrica generalizada, que inclui três métricas de distância como casos especiais, correspondendo a diferentes valores de s (do Quadro 1). Quando $s = 2$, a métrica se torna a distância Euclidiana; quando $s = 1$, se torna a distância de Manhattan; enquanto a distância de Chebyshev é uma variante da distância de Minkowski, onde $s = \infty$ (ALFEILAT *et al.*, 2019).

Quadro 1 – Diferentes métricas de distância.

Nome	Definição
Minkowski	$d(x, y) = \sqrt[s]{\sum_{i=1}^N x_i - y_i ^s}$
Manhattan (City Block)	$d(x, y) = \sum_{i=1}^N x_i - y_i $
Euclidiana	$d(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2}$
Chebyshev	$d(x, y) = \max_i x_i - y_i $
Cosseno	$d(x, y) = 1 - \frac{\sum_{i=1}^N x_i y_i}{\sqrt{\sum_{i=1}^N x_i^2} \sqrt{\sum_{i=1}^N y_i^2}}$

Fonte: Adaptado de Alfeilat *et al.* (2019).

Os k vizinhos mais próximos são selecionados com base em uma métrica de distância. Então, existe uma variedade de maneiras pelas quais os k vizinhos mais próximos podem ser usados para determinar a classe da amostra de interesse. A abordagem mais direta é atribuir a classe predominante dos vizinhos mais próximos (ALFEILAT *et al.*, 2019). No entanto, quando o número de amostras não é balanceado, o que ocorre, por exemplo, quando um volume significativo de dados é pertencente a uma classe, enquanto as outras classes são pequenas, pode acontecer erros de classificação, porque na predição de novas amostras a maioria dos vizinhos podem pertencer às classes de grandes dimensões (TANG e HE, 2015). Uma estratégia é atribuir maior peso aos vizinhos mais próximos na decisão da classe da observação. Uma técnica comumente empregada é ponderar a contribuição de cada vizinho de acordo com o inverso de sua distância até a nova amostra. Nesse caso, os vizinhos mais próximos da amostra a ser classificada terão uma influência maior do que os vizinhos mais distantes (CUNNINGHAM e DELANY, 2022).

Durante o processo de classificação é realizada uma comparação de distância entre a amostra em análise e cada amostra armazenada no conjunto de treinamento. Quando se tem grandes volumes de dados de treinamento disponíveis, essa busca pode introduzir um atraso notável no método de classificação e um custo computacional associado (SKIENA, 2017). Uma abordagem para acelerar a busca envolve o uso de estruturas de dados geométricas, como *kd-tree* e *ball tree* (CUNNINGHAM e DELANY, 2022).

2.2.2 Naive Bayes

O classificador *Naive Bayes* (NB) é um método estatístico fundamentado em modelos paramétricos (distribuições de probabilidade) baseado na aplicação do teorema de Bayes (da estatística Bayesiana) com suposições de independência condicional entre os atributos de cada classe, o que designa o termo “*Naive*” (ingênuo) (JOHN e LANGLEY, 1995).

O *Naive Bayes* pode ser utilizado para categorias binárias e multiclasse em muitas aplicações, como sistema de recomendação, análise de perfil, classificação de documento ou texto e filtragem de spam (SARKER, 2021). Geralmente, o método precisa de uma quantidade de dados de treinamento relativamente pequeno para estimar os parâmetros necessários, quando comparado com abordagens mais sofisticadas. No entanto, seu desempenho pode ser afetado devido as suas fortes premissas sobre independência de características, uma vez que em aplicações práticas é difícil encontrar problemas que sejam completamente independentes. As variantes comuns do classificador NB incluem, Gaussiano, multinomial, complementar, Bernoulli e categórico (SARKER, 2019).

Suponha que se deseje classificar o vetor $x = (x_1, \dots, x_N)$ em uma das q classes $\omega_1, \dots, \omega_q$, o teorema de Bayes (Equação 5) fornece uma maneira de calcular a probabilidade de cada classe possível dada x , para que se possa atribuir a x o rótulo da classe com maior probabilidade (SKIENA, 2017),

$$P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)}, \quad (5)$$

onde $P(\omega_i|x)$ corresponde a probabilidade posterior ou *a posteriori*, $P(x|\omega_i)$ é a função de verossimilhança, ou simplesmente a verossimilhança, ou a função de densidade de probabilidade condicional, $P(\omega_i)$ é a probabilidade *a priori* da classe (razão entre o número de amostras nessa classe e o número total de amostras em todas as classes) com $\sum_{i=1}^q P(\omega_i) = 1$, e $P(x)$ é a evidência, que é uma constante normalizadora igual para todas as classes, podendo ser desconsiderada (DUDA, HART e STORK, 2001).

Geralmente, $P(\omega_i|x)$ é calculado usando a verossimilhança ($P(x|\omega_i)$) e a probabilidade *a priori* ($P(\omega_i)$), uma vez que $P(x)$ corresponde a um termo constante normalizador que não depende da classe ω_i , é igual para todas as classes, não afetando os valores relativos de suas probabilidades (THARWAT, 2016).

Assumindo que os valores dos atributos de uma observação são condicionalmente independentes entre si dada a classe, a probabilidade de observar a conjunção de atributos $x_1, x_2, \dots, x_N | \omega_i$ é somente o produto das probabilidades para os atributos individuais:

$$P(x_1, x_2, \dots, x_N | \omega_i) = \prod_{j=1}^N P(x_j | \omega_i). \quad (6)$$

Com isso, a probabilidade de uma amostra pertencer à classe ω_i é proporcional a:

$$P(\omega_i | x) \propto P(\omega_i) \prod_{j=1}^N P(x_j | \omega_i). \quad (7)$$

Suponha que $P(\omega_i | x)$ denota a probabilidade de uma observação x pertencer à classe ω_i . A função de custo zero-um, que representa custo de associar x à classe incorreta, é minimizada se, e somente se, x é associada à classe ω_k para a qual $P(\omega_k | x)$ é máxima (DUDA, HART e STORK, 2001). Esse método é designado por estimativa Máximo a Posteriori (Maximum A Posteriori – MAP). Formalmente, a classe que deve ser associada a amostra x corresponde a:

$$y_{map} = argmax_i P(\omega_i) \prod_{j=1}^N P(x_j | \omega_i), \quad (8)$$

onde $argmax_i$ retorna a classe ω_i com maior probabilidade de estar associada a x , que é aquela que possui o valor máximo para $P(\omega_i | x)$. Os diferentes classificadores Naive Bayes diferem principalmente pelas suposições que fazem em relação à distribuição de $P(x | \omega_i)$.

O classificador obtido pelo uso da função discriminante dada pela Equação 7 e pela regra de decisão ilustrada na Equação 8 é conhecido como classificador de Naive Bayes.

As probabilidades de interesse para a obtenção do classificador Naive Bayes são computadas a partir dos dados de treinamento. Para calcular a probabilidade a priori de observar a classe ω_i , $P(\omega_i)$, é necessário manter um contador para cada classe. Para calcular a probabilidade condicional de observar um valor de um atributo dado que a amostra pertence a uma classe, é necessário distinguir entre atributos nominais e atributos contínuos. No caso de

atributos nominais, o conjunto de possíveis valores é um conjunto enumerável. Para calcular a probabilidade condicional, basta manter um contador para cada valor de atributo por classe. No caso de atributos contínuos, comumente é assumida uma distribuição particular para os valores dos atributos, sendo geralmente assumida como distribuição normal.

2.2.3 Discriminant Analysis

A *Discriminant Analysis* (DA) é usada para classificação, redução de dimensão e visualização de dados (ARAVEEPORN, 2022). O classificador análise discriminante foi introduzido por Fisher e foi usado em muitos problemas de classificação (ALTMAN, MARCO e VARETTO, 1994; GUO, HASTIE e TIBSHIRANI, 2007). Em análise discriminante, se destacam dois métodos, a saber, Linear Discriminant Analysis (LDA) e Quadratic Discriminant Analysis (QDA). No classificador LDA, a superfície de decisão é linear, enquanto no QDA a superfície de decisão é quadrática (THARWAT, 2016). Esses classificadores têm soluções de forma fechada que podem ser facilmente computadas, são inherentemente multiclasse, provaram funcionar bem na prática e não têm hiperparâmetros para ajustar (DILLEN *et al.*, 2022).

Os métodos LDA e QDA utilizam a função de distribuição de probabilidade normal multivariada de variáveis independentes como regra de classificação. O parâmetro de probabilidade a priori, média e matriz de covariância de cada classe também criam a função discriminante para o limite das classes. Os métodos são considerados como tendo a distribuição normal, que é a distribuição mais comum e padrão em aplicações práticas. Portanto, se as matrizes de covariância são assumidas iguais, o limite de decisão de classificação é na forma de uma função discriminante linear. Quando as matrizes de covariância são assumidas como desiguais, o limite de decisão da classificação está na forma de uma função discriminante quadrática (ARAVEEPORN, 2022).

Nesse contexto, os métodos DA podem ser derivados de modelos probabilísticos que modelam a distribuição condicional de classe dos dados $P(x|\omega_i)$ para cada classe i . As predições podem ser obtidas usando a regra de Bayes (Equação 5), para cada amostra de treinamento e sendo selecionada a classe i que maximiza essa probabilidade *a posteriori* (THARWAT, 2016).

Para análise discriminante linear e quadrática, a verossimilhança ($P(x|\omega_i)$) é modelado como uma distribuição gaussiana multivariada ($P(x|\omega_i) \sim \mathcal{N}(\mu_i, \Sigma_i)$) com densidade (BISHOP, 2006):

$$P(x|\omega_i) = \mathcal{N}(\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{N/2} |\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right\}, \quad (9)$$

onde μ_i representa a média da i -ésima classe e Σ_i é a matriz de covariância da i -ésima classe, $|\Sigma_i|$ e Σ_i^{-1} representam o determinante e o inverso da matriz de covariância, respectivamente. Geralmente, a amostra desconhecida será classificada para a classe, que maximiza a probabilidade posterior ou a verossimilhança, portanto, maximiza a função discriminante para o QDA da seguinte forma (THARWAT, 2016; JAMES *et al.*, 2021):

$$\ln(P(\omega_i|x)) = \ln(P(x|\omega_i)) + \ln(P(\omega_i)) \quad (10)$$

$$\ln(P(\omega_i|x)) = -\frac{1}{2} \ln|\Sigma_i| - \frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) + \ln(P(\omega_i)), \quad (11)$$

onde \ln representa o logaritmo natural.

LDA é um caso especial de QDA, onde assume-se que as Gaussianas para cada classe compartilham a mesma matriz de covariância, $\Sigma_i = \Sigma$ para todos i (GHOJOGH e CROWLEY, 2019):

$$\ln(P(\omega_i|x)) = -\frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) + \ln(P(\omega_i)). \quad (12)$$

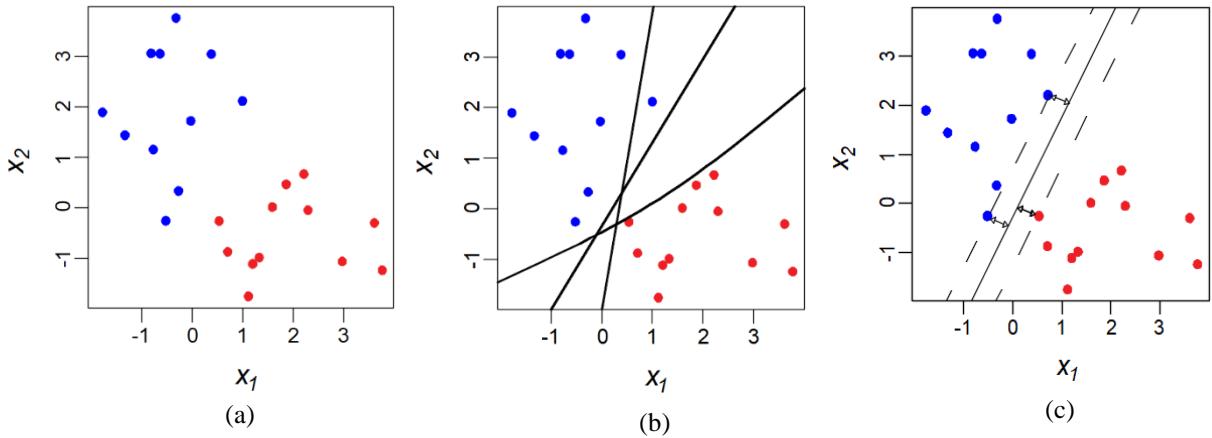
2.2.4 Support Vector Machine

O *Support Vector Machine* (SVM) constitui uma técnica de aprendizado embasada pela Teoria de Aprendizado Estatístico, fundamentado no princípio da minimização do Risco Estrutural (*Structural Risk Minimization* - SRM) (CORTES e VAPNIK, 1995). O SVM permite obter classificadores com boa generalização, correspondente a sua capacidade de predizer corretamente a classe de novos dados e pode ser também utilizado para realizar tarefas regressão e *clustering* (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018).

O objetivo da classificação do SVM é elaborar uma maneira computacionalmente eficiente de aprender hiperplanos de separação ótimo, em um espaço de características de alta dimensão, o qual maximiza a margem dos dados de treinamento (CRISTIANINI e SHAWE-TAYLOR, 2000). A Figura 5(a) apresenta um conjunto de treinamento, o objetivo do processo

de aprendizado é encontrar um classificador que separe os dados das classes azul e vermelha. Nesse contexto, se os dados puderem ser perfeitamente separados, existirá um número infinito de hiperplanos candidatos à fronteira de decisão, três possíveis hiperplanos de separação são mostrados Figura 5(b). A distância (perpendicular) de uma classe a um hiperplano de separação é a menor distância entre ele e as observações dessa classe, definida como margem. O hiperplano que será o escolhido para representar a fronteira de decisão é o que maximiza a margem de separação (menor distância perpendicular) entre as observações de treinamento (Figura 5(c)), denominado de hiperplano de margem máxima ou hiperplano de separação ótimo (JAMES *et al.*, 2014).

Figura 5 – (a) Conjunto de treinamento binário, (b) três hiperplanos de separação e (c) hiperplano de margem máxima.



Fonte: Adaptado de James *et al.* (2014).

Na Figura 5(c), três observações de treinamento são equidistantes do hiperplano de margem máxima e estão ao longo das linhas tracejadas, indicando a largura da margem. Essas três observações são conhecidas como vetores de suporte, uma vez que são vetores no espaço p -dimensional (nesse caso, $p = 2$), e eles suportam o hiperplano de margem máxima. Assim, o hiperplano de margem máxima depende diretamente dos vetores de suporte, mas não das outras observações.

Seja S um conjunto de treinamento, composto de M pares (x_i, y_i) , $x_i \in x$ e seus respectivos rótulos $y_i \in y_{label}$, onde x constitui o espaço dos dados e $y_{label} = \{-1, +1\}$. S é considerado linearmente separável, se for possível separar os dados das classes -1 e $+1$ por um hiperplano. Nesse caso, a solução do problema consiste em encontrar o hiperplano de margem máxima, que separe os dados de cada classe e a correspondente margem de separação seja máxima (Figura 6(a)), definido por:

$$\mathbf{w}^T \mathbf{x} + b = 0, \quad (13)$$

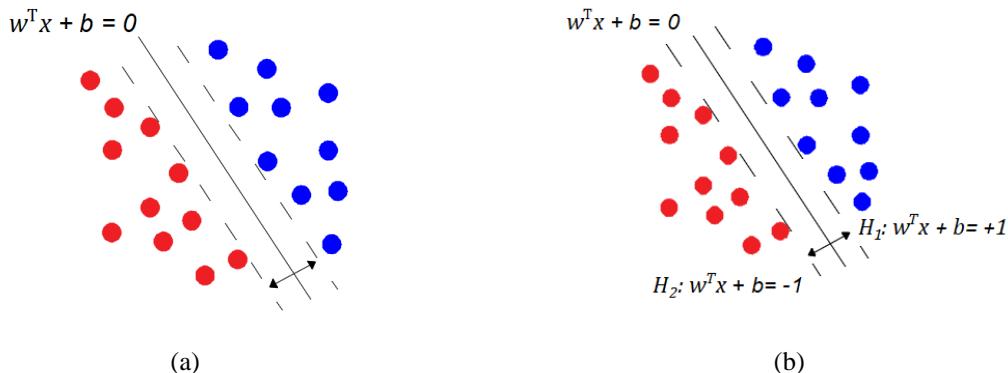
onde $\mathbf{w} \in \mathbb{R}^p$ é o vetor de pesos normal ao hiperplano e $b \in \mathbb{R}$ um escalar (bias).

A classificação de cada dado x do conjunto de treinamento é realizada de acordo a proximidade em relação às margens do hiperplano separador. Assim, será classificado como pertencente à classe -1 se estiver mais próximo da margem negativa e será pertencente a classe $+1$ se estiver mais próximo da margem positiva (Figura 6(b)). Portanto, as amostras mais próximas do hiperplano canônico (H) devem satisfazer as seguintes condições:

$$H_1: \mathbf{w}^T \mathbf{x}_i + b = 1 \quad \therefore \quad \mathbf{w}^T \mathbf{x}_i + (b - 1) = 0 \quad (14)$$

$$H_2: \mathbf{w}^T \mathbf{x}_i + b = -1 \quad \therefore \quad \mathbf{w}^T \mathbf{x}_i + (b + 1) = 0. \quad (15)$$

Figura 6 – Hiperplano de separação que maximiza a margem.



Fonte: Adaptado de Mohri, Rostamizadeh e Talwalkar (2018).

Uma amostra x é considerada classificada corretamente se estiver fora da margem de separação, ou seja, assume-se que todos os dados de treinamento satisfazem as seguintes restrições, garantindo que não existe dados de treinamento entre as margens de separação:

$$\mathbf{w}^T \mathbf{x}_i + b \geq 1 \text{ para } y_i \geq 1 \quad (16)$$

$$\mathbf{w}^T \mathbf{x}_i + b \leq -1 \text{ para } y_i \leq -1. \quad (17)$$

As equações (16) e (17) podem ser unificadas:

$$y_i(w^T x_i + b) - 1 \geq 0 \text{ para } i = 1, 2, \dots, M. \quad (18)$$

A largura da margem, distância entre os hiperplanos H_1 e H_2 , pode ser determinado pela projeção da diferença entre os vetores de suporte na direção ao vetor normal da superfície de decisão (Equação 19):

$$\rho_{lm} = \frac{2}{\|w\|}. \quad (19)$$

A maximização da margem de separação dos dados em relação ao hiperplano de separação pode ser obtida pela minimização de $\|w\|$ (BURGES, 1998). Dessa forma, pode ser definido o seguinte problema de otimização (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (20)$$

$$\text{sujeito a: } y_i(w^T x_i + b) - 1 \geq 0, \text{ para } i = 1, 2, \dots, M.$$

As restrições são impostas para assegurar que não existam dados de treinamento entre as margens de separação das classes. Por esse motivo, a SVM obtida possui também a nomenclatura de SVM com margens rígidas.

O problema de otimização que determina o hiperplano de margem máxima possui função objetivo convexa, sendo conveniente resolvê-lo com uso da Teoria Lagrangiana. Para isso, deve ser construída a função lagrangiana, que engloba as restrições à função objetivo, associadas a parâmetros denominados multiplicadores de Lagrange α_i , com $\alpha_i \geq 0$, a forma primal da função lagrangiana do problema corresponde a (CRISTIANINI e SHAWE-TAYLOR, 2000; SMOLA e SCHÖLKOPF, 2002; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^M \alpha_i [y_i(w^T x_i + b) - 1]. \quad (21)$$

A solução para esse problema de otimização é determinada minimizando $\mathcal{L}(w, b, \alpha)$ em relação a w , b e maximizando $\mathcal{L}(w, b, \alpha)$ com relação a α_i :

$$\frac{\partial \mathcal{L}(w, b, \alpha)}{\partial w} = w - \sum_{i=1}^M \alpha_i y_i x_i = 0 \quad \therefore w = \sum_{i=1}^M \alpha_i y_i x_i \quad (22)$$

$$\frac{\partial \mathcal{L}(w, b, \alpha)}{\partial b} = - \sum_{i=1}^M \alpha_i y_i = 0 \quad \therefore \sum_{i=1}^M \alpha_i y_i = 0 \quad (23)$$

com as condições: $\alpha_i [y_i(w^T x_i + b) - 1] = 0$, para $i = 1, 2, \dots, M$.

As Equações 21, 22 e 23 são conhecidas como as condições de Karush-Kuhn-Tucker (KKT). Substituindo as Equações (22) e (23) na Equação (21), obtém-se o seguinte problema de otimização:

$$\max_{\alpha} \left\{ \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j (x_i^T x_j) \right\} \quad (24)$$

sujeito a: $\alpha_i \geq 0$ e $\sum_{i=1}^M \alpha_i y_i = 0$, para $i = 1, 2, \dots, M$.

Essa formulação é denominada forma dual. Os problemas primal e dual são equivalentes, ou seja, a solução α do problema dual (Equação 24) pode ser usada diretamente para determinar a hipótese retornada por SVMs, usando a Equação 22 (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$g(x) = \text{sgn}\{f(x)\} = \text{sgn}(wx + b) = \text{sgn}\left(\sum_{i=1}^M \alpha_i y_i (x_i^T x) + b\right), \quad (25)$$

onde sgn denonta a função sinal:

se $g(x) < 0$ (sinal negativo): amostra x é classificada como classe 1;

se $g(x) > 0$ (sinal positivo): amostra x é classificada como classe 2.

Como os vetores de suporte estão nos hiperplanos de margem máxima, para qualquer vetor de suporte x_i , $w^T x_i + b = y_i$, e assim b pode ser obtido via:

$$b = y_i - \sum_{j=1}^M \alpha_j y_j (x_j^T x_i). \quad (26)$$

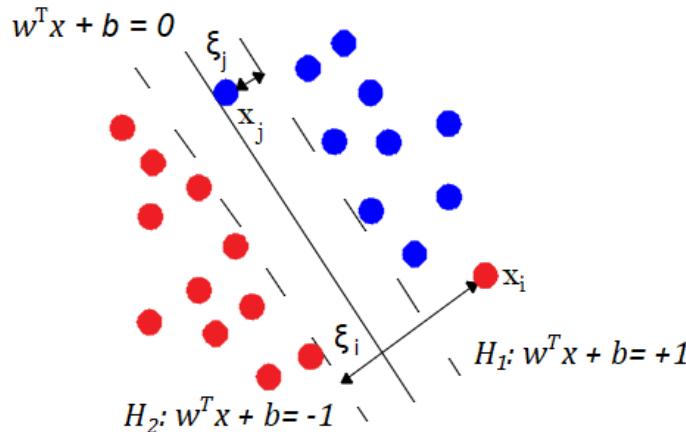
O problema de otimização dual (Equação 24) e as Equações (25) e (26) apresentam uma propriedade importante das SVMs, a solução de hipóteses depende apenas de produtos internos entre vetores e não diretamente dos vetores em si. Essa observação é fundamental para SVMs não linear.

Em aplicações práticas, é difícil que os dados sejam linearmente separáveis. Isso se deve a diversas condições, entre elas a presença de ruídos e *outliers* nos dados ou à própria natureza do problema, que pode ser não linear. Assim, para tornar o método descrito anteriormente capaz de manipular dados não linearmente separáveis é necessário relaxar as restrições das Equações (16) e (17), que utilizam critérios de margem rígida (*hard margin*), introduzindo variáveis de folga não negativas ξ_i ($i = 1, 2, \dots, M$), definida como margem suave (*soft margin*) (SMOLA *et al.*, 1999; SMOLA E SCHÖLKOPF, 2002; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$\begin{aligned} y_i(w^T x_i + b) &\geq 1 - \xi_i \text{ para } i = 1, 2, \dots, M \\ \xi_i &\geq 0, \text{ para } i = 1, 2, \dots, M. \end{aligned} \quad (27)$$

O procedimento de suavização da margem do classificador linear permite que alguns dados de treinamento possam violar a restrição e situem-se entre as margens, permitindo que alguns dados permaneçam entre os hiperplanos H_1 e H_2 (Figura 7), além de permitir a ocorrência de erros de classificação, por esse motivo, as SVMs obtidas nesse caso também podem ser referenciadas como SVMs com margens suaves.

Figura 7 – Um hiperplano de separação com amostra x_i classificada incorretamente e amostra x_j classificada corretamente, mas com margem menor que 1.



Fonte: Adaptado de Mohri, Rostamizadeh e Talwalkar (2018).

Um erro no conjunto de treinamento é indicado por um valor de ξ_i maior que 1. Assim, a soma dos ξ_i representa um limite no número de erros de treinamento (BURGES, 1998). Para considerar esse termo, minimizando assim o erro sobre os dados de treinamento, a função objetivo da Equação 20 é reformulada adicionado um termo de regularização (BURGES, 1998; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^M \xi_i \quad (28)$$

$$\text{sujeito a: } y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \text{ para } i = 1, 2, \dots, M,$$

onde c é um termo de regularização que estabelece o equilíbrio entre a complexidade do modelo e o erro de treinamento (PASSERINI, 2004). Esse parâmetro controla o peso do número de erros (que é limitado pelo somatório das variáveis de folga) e do tamanho da margem (que é inversamente proporcional à norma de w). A minimização $\|w\|^2$ resulta na maximização da margem, enquanto a minimização de $\sum_{i=1}^M \xi_i$ resulta na minimização do erro de classificação, uma vez que um valor de $\xi_i \in (0,1]$ indica um dado entre as margens.

Novamente, o problema de otimização gerado é quadrático, com as restrições lineares apresentadas na Equação 28. A sua solução é obtida de forma análoga ao caso separável, com a introdução de uma função Lagrangiana e tornando suas derivadas parciais nulas. Tem-se como resultado o seguinte problema dual (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$\max_{\alpha} \left\{ \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j (x_i^T x_j) \right\} \quad (29)$$

sujeito a:

$$0 \leq \alpha_i \leq C, \text{ para } i = 1, \dots, M$$

$$\sum_{i=1}^M \alpha_i y_i = 0.$$

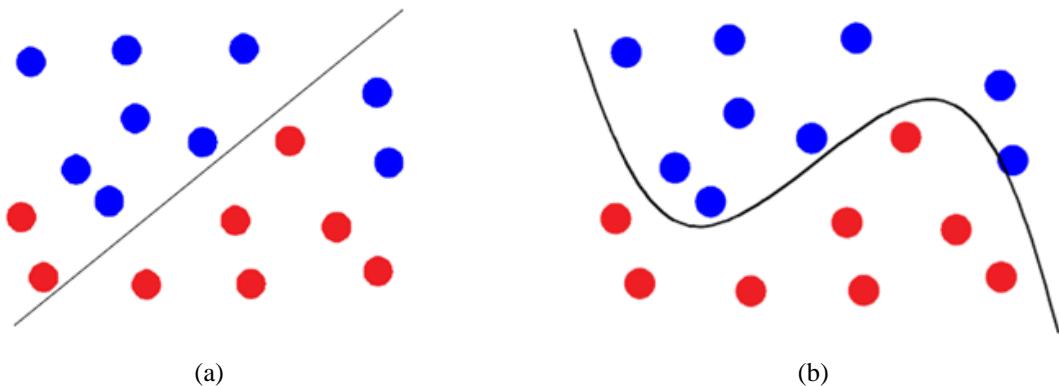
Essa formulação é semelhante à apresentada para as SVMs de margens rígidas, no entanto, a restrição nos α_i agora são limitados pelo valor de c .

A função de decisão é a mesma para o caso de margem rígida (Equação 25). Além disso, b pode ser obtido de qualquer vetor de suporte x_i situado em um hiperplano marginal, ou seja, qualquer vetor x_i com $0 < \alpha_i < C$ (Equação 26).

Como no caso separável, o problema de otimização dual (Equação 29) e as Equações (25) e (26) mostram uma propriedade importante de SVMs, a solução da hipótese depende apenas do produto interno entre vetores e não diretamente dos próprios vetores. Esse fato pode ser usado para estender SVMs para definir limites de decisão não lineares.

As SVMs lineares são eficazes na classificação de conjuntos de dados linearmente separáveis ou que possuam uma distribuição aproximadamente linear, sendo que a versão de margens suaves tolera a presença de alguns ruídos e outliers. No entanto, na prática, a separação linear geralmente não é possível. A Figura 8(a) mostra que qualquer hiperplano cruza ambas as classes. No entanto, pode-se usar funções mais complexas para separar os dois conjuntos, como na Figura 8(b).

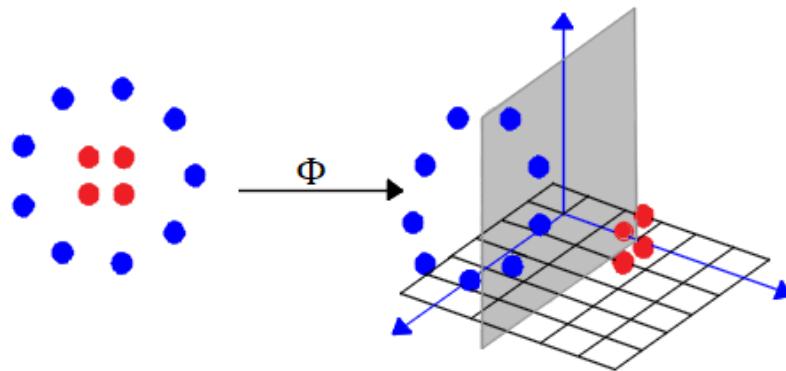
Figura 8 – Caso não linearmente separável: (a) nenhum hiperplano pode separar as duas classes e (b) um mapeamento não linear pode ser usado.



Fonte: Adaptado de Mohri, Rostamizadeh e Talwalkar (2018).

Quando se tem dados não linearmente separáveis, onde qualquer hiperplano cruza ambas as classes, pode-se usar funções mais complexas (função de separação não linear) para separar as duas classes. Uma estratégia para definir tal limite de decisão não linear é usar um mapeamento não linear Φ do espaço de entrada x para um espaço de dimensão superior H , chamado de espaço de características, sendo possível a separação linear através de um hiperplano (Figura 9) (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018).

Figura 9 – Um exemplo de mapeamento não linear de 2 dimensões para 3 dimensões, onde as amostras se tornam linearmente separáveis.



Fonte: Adaptado de Mohri, Rostamizadeh e Talwalkar (2018).

Os métodos kernel são amplamente usados para estender algoritmos como SVMs para definir limites de decisão não lineares. A ideia principal por trás desses métodos baseados em funções kernel é que, sob algumas condições de simetria e definição positiva, definem implicitamente um produto interno em um espaço de alta dimensão, a caracterização de quando uma função pode ser um kernel é dada pelo teorema de Mercer (HASTIE, TIBSHIRANI e FRIEDMAN, 2017; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018). Substituir o produto interno original no espaço de entrada por kernels definidos positivos estende algoritmos como SVMs para uma separação linear naquele espaço de alta dimensão ou, equivalentemente, para uma separação não linear no espaço de entrada.

Na prática, a modificação necessária para se implementar as SVMs não lineares em um espaço de característica maior é substituir na Equação (29) x por $\Phi(x)$. A formulação apresentada por SVMs não lineares tem uma característica singular, não é necessário conhecer mapeamento em si, apenas como realizar produtos escalares no novo espaço. Nesse sentido, o Truque de Kernel (Kernel Trick) é de interesse, que consiste em receber dois vetores no espaço

de entrada e retorna o valor do produto escalar entre eles no espaço de característica de maior dimensão (HERBRICH, 2001; MÜLLER e GUIDO, 2017),

$$K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j). \quad (30)$$

É comum empregar a função kernel sem conhecer o mapeamento, que é gerado implicitamente. A utilidade dos kernels está, portanto, na simplicidade de seu cálculo e em sua capacidade de representar espaços abstratos. A função de decisão assume a forma a seguir (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018):

$$g(x) = \operatorname{sgn} \left(\sum_{i=1}^N \alpha_i y_i K(x_i, x) + b \right) \quad (31)$$

com $b = y_i - \sum_{j=1}^N \alpha_j y_j K(x_j, x_i)$, para qualquer x_i com $0 < \alpha_i < C$.

Diferentes kernels podem ser encontrados na literatura (CRISTIANINI e SHAWE-TAYLOR, 2000; HASTIE, TIBSHIRANI e FRIEDMAN, 2017; MOHRI, ROSTAMIZADEH e TALWALKAR, 2018). O kernel radial basis function (RBF) está entre os kernels mais frequentemente usados em aplicações (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018) e possui dois hiperparametros para ajustar, C e γ . O parâmetro γ controla a flexibilidade da função de kernel, valores pequenos permitem ao classificador ajustar todos os rótulos havendo risco de *overfitting*, por outro lado, valores grandes de γ reduzem o kernel para uma função constante, tornando impossível o processo de aprendizagem (JAMES *et al.*, 2004; SHAWE-TAYLOR; CRISTIANINI, 2004).

SVMs foram desenvolvidas para classificação binária. No caso de classificação em múltiplas classes, é necessária a utilização de algum método para estender a SVM binária ou para combinar os resultados das SVMs binárias, as duas propostas mais populares são um contra um (one versus one - OVO) e um contra todos ou um contra o resto (one versus rest - OVR) (JAMES *et al.*, 2004).

2.2.5 Métodos Baseados em Árvores de Decisão

2.2.5.1 Decision Trees

O método *Decision Trees* (DTs) foi sugerido por Breiman *et al.* em 1984 (BREIMAN *et al.*, 1984) e pode ser aplicado para problemas de classificação, regressão e *clustering* (MOHRI, ROSTAMIZADEH e TALWALKAR, 2018). Correspondem em estratificar ou segmentar o espaço preditor em várias regiões simples. Como o conjunto de regras de divisão usadas para segmentar o espaço preditor pode ser resumido em uma árvore, esses tipos de abordagens são conhecidos como métodos de árvore de decisão (JAMES *et al.*, 2014). Nesse contexto, a construção de uma árvore de decisão corresponde a um conjunto de nós de decisão, conectados por ramificações, estendendo-se para baixo a partir do nó raiz até terminar em nós folha. Começando no nó raiz que, por convenção, é colocado no topo do diagrama da árvore de decisão, os atributos são testados nos nós de decisão, com cada resultado possível resultando em uma ramificação. Cada ramificação então leva a outro nó de decisão ou a um nó folha de terminação (LAROSE e LAROSE, 2014). Árvores de decisão buscam criar um conjunto de nós folha que sejam tão puros quanto possível, ou seja, onde cada um dos registros em um nó folha específico tenha a mesma classificação.

Os algoritmos de árvore de decisão mais utilizados são o ID3, C4.5 (QUINLAN, 1992), C5.0 e CART (BREINMAN *et al.*, 1984). Essencialmente, esses algoritmos diferem da forma como é construída a árvore e os critérios escolhidos para a divisão dos dados (LAROSE e LAROSE, 2014).

A construção de uma árvore de classificação pode ser baseada no critério da taxa de erro de classificação. Uma vez que o objetivo é atribuir uma observação em uma determinada região à classe de observações de treinamento que ocorre mais frequentemente naquela região, a taxa de erro de classificação é a razão das observações de treinamento naquela região que não pertencem à classe mais comum (JAMES *et al.*, 2014):

$$E = 1 - \max_k(\hat{p}_{mq}), \quad (32)$$

onde \hat{p}_{mq} representa a proporção de observações de treinamento na região m – ésima que são da classe q – ésima.

No entanto, verifica-se que o erro de classificação não é suficientemente sensível para o crescimento de árvores e, na prática, duas outras medidas são preferíveis, o índice de Gini e a entropia (JAMES *et al.*, 2014).

O índice de Gini é definido por:

$$G = \sum_{q=1}^{\omega} \hat{p}_{mq}(1 - \hat{p}_{mq}), \quad (33)$$

uma medida da variância total entre as classes ω . O índice de Gini assume um valor pequeno se todos os \hat{p}_{mk} estiverem próximos de zero ou um. Por esse motivo, o índice de Gini é chamado de medida da pureza do nó, um valor pequeno indica que um nó contém predominantemente observações de uma única classe.

Uma alternativa ao índice de Gini é a entropia, expressa por:

$$D = - \sum_{q=1}^{\omega} \hat{p}_{mq} \log(\hat{p}_{mq}), \quad (34)$$

como $0 \leq \hat{p}_{mq} \leq 1$, segue-se que $0 \leq -\hat{p}_{mq} \log(\hat{p}_{mq})$. Pode-se mostrar que a entropia assumirá um valor próximo de zero se os \hat{p}_{mk} 's estiverem todos próximos de zero ou próximos de um. Portanto, como o índice de Gini, a entropia assumirá um valor pequeno se o m -ésimo nó for puro.

Ao construir uma árvore de classificação, o índice de Gini ou a entropia são normalmente usados para avaliar a qualidade de uma divisão específica, uma vez que essas duas abordagens são mais sensíveis à pureza do nó do que a taxa de erro de classificação. Qualquer uma dessas três abordagens pode ser usada ao podar a árvore, mas a taxa de erro de classificação é preferível se a acurácia da predição da árvore podada final for o objetivo (JAMES *et al.*, 2014).

Os modelos preditivos resultantes de árvores de decisão são simples e úteis para interpretação. No entanto, um importante problema do método é a alta variância. Uma pequena alteração nos dados pode causar uma grande alteração na estrutura da árvore e, consequentemente, alterar a interpretação do modelo ajustado. No entanto, ao agregar muitas árvores de decisão, o desempenho preditivo das árvores pode ser substancialmente melhorado, reduzindo essa variância (HASTIE, TIBSHIRANI e FRIEDMAN, 2017; JAMES *et al.*, 2014).

2.2.5.2 Random Forest

O *Random Forest* (RF) (BREIMAN, 2001) é uma modificação do método *bagging* que constrói uma grande coleção de árvores descorrelacionadas (HASTIE, TIBSHIRANI e FRIEDMAN, 2014). Para implementar essa estratégia, é preciso construir muitas árvores de decisão. Cada árvore deve fazer um trabalho aceitável de predizer o alvo e também deve ser diferente das outras árvores. *Random Forest* recebe esse nome por atribuir aleatoriedade na construção da árvore para garantir que cada árvore seja diferente. Há duas formas pelas quais as árvores em uma floresta aleatória são aleatorizadas: selecionando os pontos de dados usados para construir uma árvore e selecionando os atributos em cada teste de divisão (MÜLLER e GUIDO, 2017). Para classificação, cada árvore na floresta atribui um voto para a classificação de uma nova amostra, e a proporção de votos em cada classe em todo o conjunto é o vetor de probabilidade previsto, enquanto para regressão, é feita uma média (KUHN e JOHNSON, 2013).

Como o algoritmo seleciona aleatoriamente os preditores em cada divisão, a correlação da árvore será necessariamente reduzida. Ao construir essas árvores de decisão, cada vez que uma divisão em uma árvore é considerada, uma amostra aleatória de m preditores é escolhida como candidata à divisão do conjunto completo de p preditores, a divisão pode usar apenas um desses m preditores (JAMES *et al.*, 2014). Uma nova amostra de m preditores é obtida em cada divisão, sendo recomendado definir um terço do número de preditores em problemas de regressão e a raiz quadrada do número de preditores em problemas de classificação (BREIMAN, 2001).

O número de árvores para a floresta também deve ser escolhido. Na prática, quanto maior a floresta, pode melhorar o modelo, mas também maior será o custo computacional. Breiman (2001) mostrou que as florestas aleatórias são protegidas de *overfitting*, portanto, o modelo não será afetado negativamente se um grande número de árvores for construído para a floresta.

2.2.5.3 Extremely Randomized Trees

O método *Extremely Randomized Trees* (Extra-Trees - ET) (GEURTS, ERNST e WEHENKEL, 2006) foi proposto para problemas de classificação e regressão supervisionados e consiste essencialmente em randomizar fortemente a escolha de atributos e pontos de corte enquanto divide um nó de árvore. Como em *Random Forest*, um subconjunto aleatório de

características candidatas é usado, mas os limites discriminantes são feitos aleatoriamente para cada característica candidata e o melhor desses limites gerados aleatoriamente é escolhido como a regra de divisão. O aumento da aleatoriedade geralmente permite reduzir um pouco mais a variância do modelo, às custas de um aumento no viés.

2.3 Digital Twin

O *Digital Twin* (DT) está na vanguarda da quarta revolução industrial, potencializado por meio de análise avançada de dados, *big data*, inteligência artificial, *machine learning* e conectividade de internet das coisas (FULLER *et al.*, 2020; RATHORE *et al.*, 2021). O interesse no DT tem crescido significativamente na academia e na indústria, no entanto, não existe na literatura uma definição unificada sobre o que é o DT e como o conceito está evoluindo para atender às diferentes necessidades de aplicações (JONES *et al.*, 2020). Entretanto, é consensual que um DT é uma tecnologia emergente que permite uma representação virtual de um sistema físico (cada vez mais, esse sistema físico é chamado de *physical twin*), que usa dados coletados desse sistema para conectar partes digitais e físicas, o que pode ser definido como um processo de coletar informações sobre ativos físicos e convertê-las em uma representação digital que pode ser processada automaticamente (CALLCUT *et al.*, 2021). No âmbito deste trabalho, DT é definido como um conceito engenhoso que articula modelos computacionais baseado na física e orientado a dados, sinais de sensores e *machine learning*, com o objetivo de apoiar decisões de engenharia relacionadas a um *physical twin*.

Grieves (2014) definiu o *framework* DT como consistindo em três componentes: um objeto físico, uma representação virtual desse objeto e a conexão para troca bidirecional de dados entre eles, que alimentam dados da representação física para a virtual e informações e processos da representação virtual para a física.

Tao e Zhang (2017) propuseram um modelo para DT com cinco componentes: parte física, parte virtual, conexão, dados e serviços. A parte física é a base da construção da parte virtual; a parte virtual suporta a simulação, tomada de decisão e controle da parte física; os dados constituem a parte central, sendo fundamentais para a criação de novos conhecimentos; o que levam a novos serviços que podem aumentar a conveniência, confiabilidade e produtividade de um sistema projetado; a conexão conecta a parte física, a parte virtual, os dados e o serviço (TAO *et al.*, 2019).

Singh, Weebera e Birke (2021) propuseram um *framework* DT com sete componentes, o que inclui: o *physical twin* composto por uma unidade de controle, sensores e atuadores; a

parte de comunicação responsável pela aquisição e processamento de dados; a parte de segurança responsável pelo manuseio seguro do fluxo de dados; a parte de armazenamento de dados; a parte de modelagem e otimização de DT; a parte de serviço responsável pelo desenvolvimento de aplicações avançadas orientadas a dados ou funções de análise de dados padrão; e, por fim, a última parte responsável por fornecer as informações à parte interessada usando a visualização de dados, o que também inclui dispositivos que permitem a tomada de decisão para o usuário e o *feedback* de informações para o dispositivo físico (por exemplo, por meio de uma interface homem-máquina (IHM) ou *feedback* direto para a unidade de controle do *physical twin*). Essa arquitetura coloca o processo físico no centro e constrói o DT em torno das funcionalidades do mundo real. Também pode ser criado de forma incremental, com adição de novos componentes em diferentes ciclos de implementação.

Em um DT, os elementos físicos e virtuais trocam informações para monitorar, simular, prever, otimizar, diagnosticar e controlar o estado e o comportamento do objeto físico dentro do espaço virtual. Assim, espera-se que os DTs prevejam a evolução do estado do objeto físico usando os dados trocados. Um uso comum de DTs para essa finalidade é evitar a interrupção do serviço em situações de manutenção. Um DT também pode ser usado para realizar monitoramento contínuo por meio de aquisição de dados em tempo real. Outro uso comum do DT é melhorar a segurança e a resiliência, devido à capacidade do DT de detectar ações maliciosas em um sistema. Além disso, permite uma melhor avaliação de risco para testar vários casos hipotéticos que podem afetar os objetos físicos (SEGOVIA e GARCIA-ALFARO, 2022).

Nesse sentido, um DT se atualiza para rastrear o *physical twin* por meio do uso de sensores, análise de dados, *machine learning* e IoT (CHAKRABORTY, ADHIKARI e GANGULI, 2021).

No contexto da engenharia estrutural, a parte física compreende os componentes e subsistemas estruturais do ativo físico (*physical twin*) em questão, que interage com o meio ambiente por meio de processos físicos ou químicos, o ambiente circundante é responsável pelas ações externas na estrutura, como carregamentos, temperatura e umidade. Por outro lado, a parte virtual fornece uma idealização do *physical twin* sob um nível específico de abstração por meio de modelos matemáticos baseados em leis da física, dados ou ambos. A conexão realiza a interação do objeto físico e do objeto virtual, por meio de sensores e atuadores, que coletam e processam dados sobre o comportamento estrutural e as condições ambientais. O desempenho estrutural e sua evolução ao longo do tempo são descritos pelos modelos usando estados e parâmetros. O modelo virtual é atualizado de forma adaptativa com base no desempenho do *physical twin* medido pelos sensores ao longo de parte ou de todo o ciclo de

vida estrutural. Ao mesmo tempo, fornece uma ferramenta para inferir o estado da integridade estrutural, realizar avaliações de confiabilidade e risco, prever sua vida útil remanescente e obter autonomamente planejamento de cenários e ações de tomada de decisão (VANDERHORN e MAHADEVAN, 2021; KAN e ANUMBA, 2019).

É importante entender que um DT estrutural pode não modelar perfeitamente o desempenho real da estrutura. Uma vez que determinados estados estruturais e variáveis ambientais podem não ser medidos diretamente com os métodos de detecção disponíveis e todo o desempenho estrutural pode não ser idealizado através de todos os níveis possíveis de abstração dentro de um *framework* de modelagem, devido à limitação de conhecimento, tecnologia ou economia (CHIACHÍO *et al.*, 2022). Nesse sentido, um *framework* DT estrutural deve ser definido usando *twins* contextuais (WORDEN *et al.*, 2020), onde o contexto é dado por um subconjunto de parâmetros estruturais, estados de integridade ou danos e variáveis ambientais de interesse (CHIACHÍO *et al.*, 2022).

3 IDENTIFICAÇÃO ESPARSA DE DINÂMICA NÃO LINEAR

Neste capítulo são apresentados o método utilizado neste trabalho, o qual emprega regressão esparsa para descobrir equações governantes do sistema dinâmico a partir de dados, os conhecimentos matemáticos necessários para aprender a dinâmica através de regressão esparsa e a configuração do problema de descoberta de modelo que será de interesse resolver.

A identificação esparsa de dinâmica não linear (*sparse identification of nonlinear dynamics* – SINDy) é um método desenvolvido por Brunton, Proctor e Kutz (2016) que associa técnicas de esparsidade e *machine learning* para descobrir equações governantes de sistemas dinâmicos a partir de dados medidos.

Considere o sistema dinâmico não linear para o vetor de estado $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T \in \mathbb{R}^n$ definido por:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t)). \quad (35)$$

Dado um conjunto de medições de $\mathbf{x}(t)$, o método SINDy busca identificar $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$. Para muitos sistemas dinâmicos, quando representados na base de funções candidatas apropriada, a função que especifica a dinâmica, \mathbf{f} , consiste em apenas alguns termos importantes, de forma que as equações são esparsas no espaço de funções possíveis. Então, usando técnicas de regressão esparsa pode-se determinar o número suficiente de termos para representar \mathbf{f} , sendo esperados modelos parcimoniosos que equilibram a acurácia com a complexidade do modelo para evitar *overfitting*.

Para determinar a função \mathbf{f} , é coletado um conjunto de dados, provavelmente ruidosos, do estado $\mathbf{x}(t)$ e sua primeira derivada $\dot{\mathbf{x}}(t)$ ou do estado $\mathbf{x}(t)$ e da aproximação numérica de sua primeira derivada $\dot{\mathbf{x}}(t)$, amostrados no tempo, t_1, t_2, \dots, t_M . Essas medidas são concatenadas em duas matrizes, a matriz de estados \mathbf{X} ($\mathbf{X} \in \mathbb{R}^{M \times n}$) (Equação 36) e a matriz da derivada de estados $\dot{\mathbf{X}}$ ($\dot{\mathbf{X}} \in \mathbb{R}^{M \times n}$) (Equação 37), onde as colunas correspondem a diferentes variáveis de estado e as linhas correspondem a diferentes pontos no tempo:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(t_1)^T \\ \mathbf{x}(t_2)^T \\ \vdots \\ \mathbf{x}(t_M)^T \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \ddots & x_n(t_2) \\ \vdots & \vdots & \cdots & \vdots \\ x_1(t_M) & x_2(t_M) & \cdots & x_n(t_M) \end{bmatrix} \quad (36)$$

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{x}}(t_1)^T \\ \dot{\mathbf{x}}(t_2)^T \\ \vdots \\ \dot{\mathbf{x}}(t_M)^T \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \cdots & \dot{x}_n(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \ddots & \dot{x}_n(t_2) \\ \vdots & \vdots & \cdots & \vdots \\ \dot{x}_1(t_M) & \dot{x}_2(t_M) & \cdots & \dot{x}_n(t_M) \end{bmatrix}. \quad (37)$$

Em seguida, é especificada uma base de funções candidatas, $\{\phi_i(\mathbf{x}), i = 1, 2, \dots, p\}$, com a qual se pretende representar \mathbf{f} . As funções candidatas são avaliadas em \mathbf{X} para construir uma matriz de biblioteca de dados:

$$\Phi(\mathbf{X}) = \begin{bmatrix} | & | & & | \\ \phi_1(\mathbf{X}) & \phi_2(\mathbf{X}) & \cdots & \phi_p(\mathbf{X}) \\ | & | & & | \end{bmatrix}. \quad (38)$$

Na matriz $\Phi(\mathbf{X})$ ($\Phi(\mathbf{X}) \in \mathbb{R}^{M \times Q}$), onde Q é número total de funções distintas na biblioteca, cada coluna corresponde a uma única função candidata, que podem incluir, por exemplo, termos constantes, monômios até certo grau finito, funções trigonométricas e funções racionais. Por exemplo, se $\phi_1(X)$ for um polinômio de grau 2, $\phi_1(X)$ denota não linearidade quadrática que pode ser expressa como:

$$\phi_1(X) = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & \cdots & x_2^2(t_1) & \cdots & x_n^2(t_1) \\ x_1^2(t_2) & x_1(t_2)x_2(t_2) & \cdots & x_2^2(t_2) & \cdots & x_n^2(t_2) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_1^2(t_M) & x_1(t_M)x_2(t_M) & \cdots & x_2^2(t_M) & \cdots & x_n^2(t_M) \end{bmatrix}. \quad (39)$$

Na prática, a seleção dessas funções candidatas pode ser informada pelo conhecimento prévio do profissional sobre o sistema dinâmico que está sendo medido.

Cada componente de \mathbf{f} pode ser representado como uma combinação linear esparsa da base de funções candidatas, o que permite apresentar um problema de regressão esparsa a ser resolvido para os coeficientes usados nessas combinações lineares:

$$\dot{\mathbf{X}} = \Phi(\mathbf{X})\Xi, \quad (40)$$

onde Ξ ($\Xi \in \mathbb{R}^{Q \times n}$) é a matriz esparsa, onde cada coluna corresponde a um vetor esparso (ξ_k) de coeficientes que determina quais termos estão ativos (diferentes de zero) no lado direito para cada componente de \mathbf{f} . Uma vez que Ξ tenha sido determinada, um modelo de cada componente das equações governantes pode ser construído da seguinte forma:

$$\dot{x}_j = f_j(x) = \Phi(x^T)\xi_k, \quad (41)$$

onde $\Phi(x^T)$ deve ser interpretado como um vetor (linha) de funções simbólicas de componentes de x . O sistema completo de equações diferenciais é então dado por:

$$\dot{x} = f(x) = \Xi^T(\Phi(x^T))^T. \quad (42)$$

Para ilustração do método SINDy é apresentado um exemplo usando as equações do sistema caótico de Lorenz (LORENZ, 1963), desenvolvidas para descrever alguns dos comportamentos imprevisíveis característicos do tempo meteorológico (SPARROW, 1982),

$$\begin{cases} \dot{x} = -\sigma x + \sigma y, \\ \dot{y} = \rho x - y - xz, \\ \dot{z} = xy - \beta z. \end{cases} \quad (43)$$

Essas equações originam a dinâmica caótica que evolui em um atrator e apenas alguns termos estão ativos no lado direito. Considerando as funções candidatas $[1, x, y, z, x^2, xy, xz, y^2, yz, z^2]$, a dinâmica esparsa é identificada de forma que o sistema pode ser expresso como:

$$\dot{x} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \Xi^T(\Phi(x^T))^T = \begin{bmatrix} 0 & -\sigma & \sigma & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \rho & -1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -\beta & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ z \\ x^2 \\ xy \\ xz \\ y^2 \\ yz \\ z^2 \end{bmatrix}. \quad (44)$$

Na prática, as medições são contaminadas por ruído, sendo observada uma versão perturbada de $x(t)$, e, em muitos casos, $\dot{x}(t)$ não é observado diretamente, sendo aproximado de $x(t)$, estabelecendo outra fonte de erro. A Equação (40) anteriormente exata, a ser resolvida para Ξ , é suplantada pelo problema de aproximação. Além disso, na maioria das aplicações, $M \gg Q$, dessa forma, a Equação (40) é sobredeterminada. Assim, é buscada uma solução esparsa para um sistema sobredeterminado com ruído (BRUNTON, PROCTOR e KUTZ, 2016; DE SILVA *et al.*, 2020):

$$\dot{\mathbf{X}} \approx \Phi(\mathbf{X})\Xi. \quad (45)$$

3.1 Otimizadores

Para determinar Ξ , o SINDy busca resolver o seguinte problema de otimização (CHAMPION *et al.*, 2020):

$$\min_{\Xi} \frac{1}{2} \left\| \dot{\mathbf{X}} - \Phi(\mathbf{X})\Xi \right\|_F^2 + \lambda R(\Xi), \quad (46)$$

onde $\|\cdot\|_F$ é a norma de Frobenius, $R(\cdot)$ é um termo de regularização que promove soluções esparsas e λ é um hiperparâmetro que determina a força da regularização. Cada coluna de Ξ codifica uma equação diferencial para um único componente de \mathbf{x} . Então, resolver a Equação (46) consiste em resolver n problemas desacoplados.

3.1.1 Least Absolute Shrinkage and Selection Operator

Uma abordagem comum é escolher $R(\cdot)$ para ser a norma ℓ_1 promotora de esparsidade, que é a relaxação convexa da norma ℓ_0 . Nesse caso, SINDy é resolvido via *Least Absolute Shrinkage and Selection Operator* (LASSO) (BRUNTON, PROCTOR e KUTZ, 2016; CHAMPION *et al.*, 2020). O método LASSO (TIBSHIRANI, 1996; HASTIE, TIBSHIRANI e WAINWRIGHT, 2015) (Equação 47) executa tanto a seleção de coeficientes, ao ser capaz de tornar alguns coeficientes nulos, quanto a regularização com base na penalidade ℓ_1 :

$$\min_{\Xi} \frac{1}{2} \left\| \dot{\mathbf{X}} - \Phi(\mathbf{X})\Xi \right\|_2^2 + \lambda \|\Xi\|_1, \quad (47)$$

onde $\|\cdot\|_2$ é norma-2 de um vetor e $\|\cdot\|_1$ é a norma-1 de um vetor. Quando $\lambda=0$, a Equação (47) é reduzida a mínimos quadrados, à medida que λ aumenta, Ξ_{lasso} fica mais esparso. Ao selecionar λ adequado, a solução pode equilibrar a compensação entre a acurácia e a complexidade do modelo. O método LASSO normalmente é resolvido por um algoritmo de otimização.

3.1.2 Sequential Thresholded Least Squares

O método LASSO pode se tornar computacionalmente caro para grandes conjuntos de dados e não identificar com sucesso modelos esparsos, apresentando padrões de esparsidade incorretos (BRUNTON, PROCTOR e KUTZ, 2016; SU, BOGDAN e CANDÈS, 2017). Mesmo com dados sem ruídos, os modelos LASSO para SINDy normalmente têm muitos coeficientes que são pequenos em magnitude, mas diferentes de zero (CHAMPION *et al.*, 2020). Nesse contexto, Brunton, Proctor e Kutz (2016) propuseram o método *sequential thresholded least squares* (STLSQ) como um algoritmo computacionalmente eficiente, notavelmente robusto ao ruído e que converge rapidamente para uma solução esparsa em um pequeno número de iterações. O método STLSQ possui um único parâmetro δ necessário para determinar o grau de esparsidade em Ξ .

Dado um limite de esparsidade δ , que especifica a magnitude mínima para um coeficiente em Ξ , o algoritmo de STLSQ alterna em (a) resolver sucessivamente o problema de mínimos quadrados para cada coluna de Ξ e (b) remover funções candidatas de consideração cujos coeficientes correspondentes em Ξ estão com magnitude abaixo do limite especificado.

3.2 Equações de Movimento

As equações de movimento do sistema dinâmico, $x(t)$, envolvem um termo de aceleração. Consequentemente, é imposta a restrição de que as equações governantes do modelo sejam equações diferenciais de segunda ordem:

$$\ddot{x} = f(x, \dot{x}). \quad (48)$$

A estrutura SINDy é projetada para trabalhar com sistemas de equações diferenciais de primeira ordem, então a Equação (48) é convertida em tal sistema:

$$\begin{cases} \dot{x} = v \\ \dot{v} = g(x, v). \end{cases} \quad (49)$$

Em seguida, o SINDy pode ser aplicado, com $x = [x \ v]^T$ e $f(x) = [v \ g(x)]^T$, e é tentado aprender a função g . De fato, como já é conhecida a função correta do lado direito para \dot{x} , é preciso encontrar uma expressão para \dot{v} .

Caso se tenha conhecimento do modelo físico da estrutura, o SINDy pode ser estruturado considerando a estrutura das equações de um modelo baseado na física.

4 EXEMPLOS DE APLICAÇÃO: VIBRAÇÃO AXIAL

Este capítulo apresenta as aplicações das técnicas supramencionadas para um modelo de barra de aço, considerando a vibração axial. Inicialmente, são apresentadas as etapas para aplicação das técnicas. E em seguida, demonstra-se a aplicabilidade a dois modelos de barra de aço: engastada e biengastada. Cada modelo foi discretizado em dois, quatro e seis elementos finitos, correspondendo a seis configurações de análises. As propriedades do material e informações geométricas dos modelos são: comprimento $L = 1,00$ m, altura e largura da seção transversal retangular $h \times b = 0,025$ m \times 0,05 m, massa específica $\rho = 7850$ kg/m³ e módulo de elasticidade $E = 210 \times 10^9$ N/m².

4.1 Procedimento para Aplicação Prática das Técnicas

4.1.1 Physical Twin

Nestas aplicações, o *physical twin* consistiu em uma barra uniforme sob excitação axial, considerado como um modelo de alta fidelidade da estrutura em análise, e sua equação de movimento longitudinal pode ser expressa como:

$$\rho A \frac{\partial^2 u(x, t)}{\partial t^2} - EA \frac{\partial^2 u(x, t)}{\partial x^2} = p(x, t), \quad (50)$$

onde $u(x, t)$ é o deslocamento longitudinal da barra, x é a posição espacial ao longo de seu comprimento, t é o tempo, E é o módulo de elasticidade, A é a área da seção transversal, ρ é a massa específica e $p(x, t)$ é a força axial distribuída. Essa equação diferencial parcial foi discretizada através do método dos elementos finitos, sendo empregadas funções de forma linear, de modo que foi obtido o seguinte sistema discreto (CRAIG e KURDILA, 2006):

$$[M_{pt}] \ddot{u}_{pt}(t) + [C_{pt}] \dot{u}_{pt}(t) + [K_{pt}] u_{pt}(t) = p_{pt}(t), \quad (51)$$

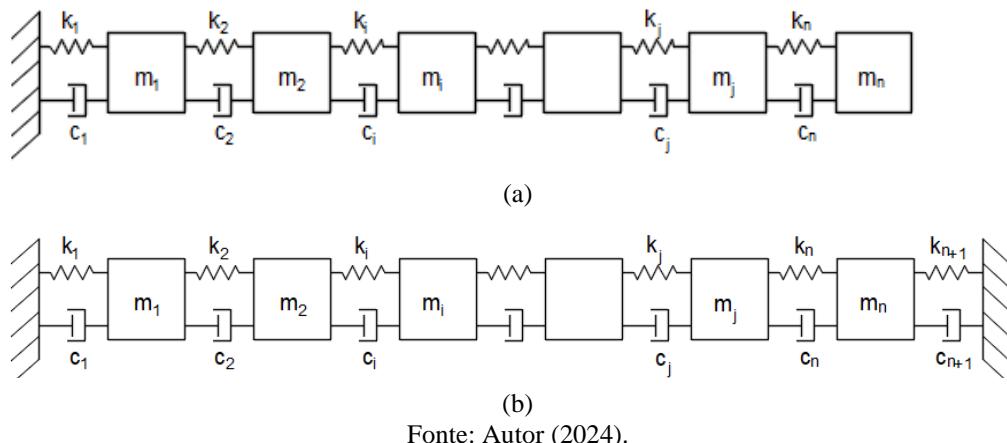
onde o subscrito *pt* representa o *physical twin*, $[M_{pt}]$ é a matriz de massa, $[K_{pt}]$ é a matriz de rigidez e $[C_{pt}]$ é a matriz de amortecimento proporcional ($[C_{pt}] = \alpha[M_{pt}] + \beta[K_{pt}]$), onde α e β são constantes de proporcionalidade reais e positivas (amortecimento de Rayleigh), estabelecidas partindo da especificação das razões de amortecimento para o modo fundamental

e para um dos modos de frequência mais alta que contribui significativamente para a resposta dinâmica (CLOUGH e PENZIEN, 2003). O sistema foi discretizado em dois, quatro e seis elementos finitos e foram aplicadas as condições de contorno adequadas para o modelo de barra engastada e para o modelo de barra biengastada.

4.1.2 Modelo Baseado na Física

A idealização do modelo de barra (*physical twin*) foi baseada na física e construída utilizando uma descrição de parâmetros concentrados de n graus de liberdade (*degrees of freedom – DOF*) (n -DOF), para representação do modelo de barra engastada (Figura 10(a)) e biengastada (Figura 10(b)):

Figura 10 – Idealização do modelo de barra (a) engastada e (b) biengastada, utilizando descrição de parâmetros concentrados de n -DOF.



Fonte: Autor (2024).

As equações de movimento do sistema n -DOF são:

$$[M_{mf}] \ddot{x}_{mf}(t) + [C_{mf}] \dot{x}_{mf}(t) + [K_{mf}] x_{mf}(t) = f_{mf}(t), \quad (52)$$

onde o subscrito mf representa o modelo físico, $[C_{mf}]$ é a matriz de amortecimento proporcional ($[C_{mf}] = \alpha[M_{mf}] + \beta[K_{mf}]$).

A estrutura SINDy é projetada para trabalhar com sistemas de equações diferenciais de primeira ordem, então as equações de movimento (Equação 52) foram convertidas em tal sistema (Equação 53), para o modelo de barra engastada, $k_{n+1} = 0$:

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(-\frac{k_1 + k_2}{m_1} \right) x_1 + \left(-\frac{\alpha(m_1) + \beta(k_1 + k_2)}{m_1} \right) x_2 + \left(\frac{k_2}{m_1} \right) x_3 \\ \quad + \left(\frac{\beta(k_2)}{m_1} \right) x_4 \\ \dot{x}_{2i-1} = x_{2i} \\ \dot{x}_{2i} = \left(\frac{f_i(t)}{m_i} \right) + \left(\frac{k_i}{m_i} \right) x_{2i-3} + \left(\frac{\beta(k_i)}{m_i} \right) x_{2i-2} + \left(-\frac{k_i + k_{i+1}}{m_i} \right) x_{2i-1} \\ \quad + \left(-\frac{\alpha(m_i) + \beta(k_i + k_{i+1})}{m_i} \right) x_{2i} + \left(\frac{k_{i+1}}{m_i} \right) x_{2i+1} + \left(\frac{\beta(k_{i+1})}{m_i} \right) x_{2i+2} \\ \quad \text{para } i = 2, 3, \dots, n-1 \\ \dot{x}_{2n-1} = x_{2n} \\ \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left(\frac{k_n + k_{n+1}}{m_n} \right) x_{2n-3} + \left(\frac{\beta(k_n)}{m_n} \right) x_{2n-2} + \left(-\frac{k_n}{m_n} \right) x_{2n-1} \\ \quad + \left(-\frac{\alpha(m_n) + \beta(k_n + k_{n+1})}{m_n} \right) x_{2n}. \end{array} \right. \quad (53)$$

4.1.3 Respostas do SINDy

Para gerar os dados de treinamento para o SINDy, foi aplicada uma força de compressão com magnitude de 10×10^3 N na extremidade direita do modelo da barra engastada e de 50×10^3 N no último DOF do modelo da barra biengastada (modelo do *physical twin*). As forças foram escolhidas de modo a mobilizar a estrutura e atender ao seu critério de estabilidade estática. As respectivas respostas de deslocamento no tempo foram então registradas. As respostas mostraram que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para os DOFs que estão mais próximos do ponto de aplicação da força.

A taxa de amostragem utilizada foi determinada usando o teorema da amostragem de Nyquist–Shannon¹ (SHANNON, 1949), considerada uma taxa de amostragem de aproximadamente duas vezes e meia a maior frequência do modelo de alta fidelidade.

¹ Um sinal analógico que foi amostrado pode ser recuperado a partir de uma sequência de amostras, se a taxa de amostragem for superior a $2F_s$ amostras por segundo, onde F_s é a maior componente de frequência contida em um sinal original (LATHI, 1998).

Os dados de velocidade ($\dot{x}(t)$) e aceleração ($\ddot{x}(t)$) foram obtidos através da diferenciação numérica dos dados de deslocamento ($x(t)$), via método das diferenças finitas centrais de segunda ordem (KUTZ, 2013).

Foram comparados os algoritmos STLSQ e LASSO, usando o pacote PySINDy da linguagem Python 3.9 (DE SILVA *et al.*, 2020). Os modelos LASSO foram ajustados usando o pacote *scikit-learn* (PEDREGOSA *et al.*, 2011). A matriz da biblioteca SINDy $\Phi(X)$ foi construída usando termos polinomiais de primeira ordem. O SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física (Equação 53).

O parâmetro δ , para o algoritmo STLSQ, e o hiperparâmetro λ , para o algoritmo LASSO, foram variados considerando um determinado intervalo, de modo que aumentar δ ou λ produziam modelos com poucos ou nenhum termo e diminuí-los produziam modelos convergentes. À medida que os resultados foram gerados pelo SINDy, para cada parâmetro δ e hiperparâmetro λ variado, foi avaliada a raiz do erro quadrático médio (*root mean square error* – RMSE) (WILLMOTT e MATSUURA, 2005) entre os sinais de entrada de aceleração e a predição das equações descobertas para aceleração. O parâmetro ótimo δ ($\delta_{óptimo}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo}$) foram escolhidos de forma que minimizasse a medida de erro.

4.1.4 Parametrização de Dano

O dano foi representado pela redução da rigidez da mola (como uma perda de rigidez local), parametrizado por meio de um escalar $\eta \in [0, 1]$. O valor da rigidez foi multiplicado por η , onde $d = 1 - \eta$, com d a porcentagem do dano que corresponde a sua intensidade. A parametrização de dano para os termos das equações que contêm uma única mola correspondeu à Equação (54), para cada i -ésima mola de interesse:

$$\frac{k_i}{m} = \frac{\eta_i k_i}{m} = \frac{k_i}{m} (1 - d_i). \quad (54)$$

Para os termos das equações que contêm duas molas,

$$\frac{k_i + k_j}{m} = \frac{\eta_i k_i + \eta_j k_j}{m}. \quad (55)$$

Uma vez que a estrutura está sem dano, $k_i = k_j = k$. Então,

$$\begin{aligned}
\frac{k}{m}(\eta_i + \eta_j) &= \frac{k}{m}((1 - d_i) + (1 - d_j)) = \frac{k}{m}(2 - d_i - d_j) = \frac{2k}{m}\left(1 - \frac{d_i}{2} - \frac{d_j}{2}\right) \\
&= \frac{k_i + k_j}{m}\left(1 - \frac{d_i}{2} - \frac{d_j}{2}\right).
\end{aligned} \tag{56}$$

Portanto, a parametrização de dano para os termos das equações em que contêm duas molas correspondeu à Equação (57), para cada i -ésima e j -ésima mola de interesse:

$$\frac{k_i + k_j}{m} = \frac{k_i + k_j}{m}\left(1 - \frac{d_i}{2} - \frac{d_j}{2}\right). \tag{57}$$

Uma estrutura íntegra é representada por $d = 0$ em todas as molas. A representação do dano corresponde à redução na matriz de rigidez, enquanto a matriz de amortecimento permanece constante.

As melhores equações de movimento descobertas pelo SINDy foram reescritas para parametrização de dano, modelo de barra engastada (Equação 58) e modelo de barra biengastada (Equação 59):

$$\left\{
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \left(\frac{f_1(t)}{m_1}\right) + \left(\left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right)\left(-\frac{k_1 + k_2}{m_1}\right)\right)x_1 + \left(-\frac{\alpha(m_1) + \beta(k_1 + k_2)}{m_1}\right)x_2 \\
&\quad + \left((1 - d_2)\left(\frac{k_2}{m_1}\right)\right)x_3 + \left(\frac{\beta(k_2)}{m_1}\right)x_4 \\
\dot{x}_{2i-1} &= x_{2i} \\
\dot{x}_{2i} &= \left(\frac{f_i(t)}{m_i}\right) + \left((1 - d_i)\left(\frac{k_i}{m_i}\right)\right)x_{2i-3} + \left(\frac{\beta(k_i)}{m_i}\right)x_{2i-2} + \left(-\frac{k_i + k_{i+1}}{m_i}\right)x_{2i-1} \\
&\quad + \left(-\frac{\alpha(m_i) + \beta(k_i + k_{i+1})}{m_i}\right)x_{2i} + \left(\frac{k_{i+1}}{m_i}\right)x_{2i+1} + \left(\frac{\beta(k_{i+1})}{m_i}\right)x_{2i+2} \\
&\quad \text{para } i = 2, 3, \dots, n-1 \\
\dot{x}_{2n-1} &= x_{2n} \\
\dot{x}_{2n} &= \left(\frac{f_n(t)}{m_n}\right) + \left((1 - d_n)\left(\frac{k_n}{m_n}\right)\right)x_{2n-3} + \left(\frac{\beta(k_n)}{m_n}\right)x_{2n-2} \\
&\quad + \left((1 - d_n)\left(-\frac{k_n}{m_n}\right)\right)x_{2n-1} + \left(-\frac{\alpha(m_n) + \beta(k_n)}{m_n}\right)x_{2n}
\end{aligned} \right. \tag{58}$$

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(\left(1 - \frac{d_1}{2} - \frac{d_2}{2} \right) \left(-\frac{k_1 + k_2}{m_1} \right) \right) x_1 + \left(-\frac{\alpha(m_1) + \beta(k_1 + k_2)}{m_1} \right) x_2 \\ \quad + \left((1 - d_2) \left(\frac{k_2}{m_1} \right) \right) x_3 + \left(\frac{\beta(k_2)}{m_1} \right) x_4 \\ \dot{x}_{2i-1} = x_{2i} \\ \dot{x}_{2i} = \left(\frac{f_i(t)}{m_i} \right) + \left((1 - d_i) \left(\frac{k_i}{m_i} \right) \right) x_{2i-3} + \left(\frac{\beta(k_i)}{m_i} \right) x_{2i-2} + \left(-\frac{k_i + k_{i+1}}{m_i} \right) x_{2i-1} \\ \quad + \left(-\frac{\alpha(m_i) + \beta(k_i + k_{i+1})}{m_i} \right) x_{2i} + \left(\frac{k_{i+1}}{m_i} \right) x_{2i+1} + \left(\frac{\beta(k_{i+1})}{m_i} \right) x_{2i+2} \\ \text{para } i = 2, 3, \dots, n-1 \\ \dot{x}_{2n-1} = x_{2n} \\ \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left(\left(1 - \frac{d_n}{2} - \frac{d_{n+1}}{2} \right) \left(\frac{k_n + k_{n+1}}{m_n} \right) \right) x_{2n-3} + \left(\frac{\beta(k_n)}{m_n} \right) x_{2n-2} \\ \quad + \left((1 - d_n) \left(-\frac{k_n}{m_n} \right) \right) x_{2n-1} + \left(-\frac{\alpha(m_n) + \beta(k_n + k_{n+1})}{m_n} \right) x_{2n}. \end{array} \right. \quad (59)$$

4.1.5 Construção do Conjunto de Dados

A estrutura do conjunto de dados X_{dados} empregada para o aprendizado supervisionado de classificação foi construída usando M amostras de deslocamentos do sistema (U_1, U_2, \dots, U_N), equações descobertas através do SINDy. O conjunto de dados X_{dados} (atributos) e o cenário de dano associado $y_{rótulo}$ (rótulos) são o par de entrada para o classificador.

A estrutura do conjunto de dados compreende M amostras de deslocamentos para cada um dos r cenários de dano, que são gerados usando as equações descobertas. A resposta de cada amostra de deslocamento corresponde a uma linha da matriz X_{dados} . Por exemplo, o cenário de dano d_1 pode ser de dano nulo em todas as molas (que caracteriza a estrutura íntegra), que está associada ao rótulo d_1 (íntegra), correspondendo a classe íntegra ($d_1 X_{dados}$). Portanto,

$$d_1 X_{dados} = \begin{bmatrix} U_1^{(1)} & U_2^{(1)} & \cdots & U_N^{(1)} & y_{d1} \\ U_1^{(M)} & U_2^{(M)} & \cdots & U_N^{(M)} & y_{d1} \end{bmatrix}.$$

O conjunto de dados final X_{dados} é uma matriz de dimensão $((M \times r) \times N)$, composta das M amostras de deslocamentos de cada r cenários de dano, incluindo a estrutura íntegra (classes), com N deslocamentos medidos (atributos):

$$X_{dadosfinal} = \begin{bmatrix} d_1 X_{dados} \\ d_2 X_{dados} \\ \dots \\ d_r X_{dados} \end{bmatrix}.$$

4.1.6 Construção dos Classificadores de *Machine Learning*

Na construção do conjunto de dados usado para treinar os classificadores, as equações do modelo SINDy foram usadas para simular a resposta de deslocamento no tempo do sistema com diferentes locais de danos e diferentes intensidades correspondentes. O modelo SINDy foi integrado no tempo, de forma que 300 amostras foram coletadas para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, \dots, n$), onde n corresponde ao número de molas. São intensidades de dano amplamente investigado na literatura (PARVIZ, CHAN e GALLAGE, 2020; RATCLIFFE, 1997; SHI, LAW e ZHANG, 2000; WANG e LI, 2012; WU *et al.*, 2017). Isso resultou em um conjunto de dados de treinamento na forma $((300 \times (\text{íntegra} + n^o \text{ de molas} \times 4 \text{ intensidades de dano}) \times (n^o \text{ de DOFs}))$, onde uma validação cruzada estratificada de 5 *folds* foi empregada. Para melhorar as condições de treinamento, os dados foram normalizados subtraindo a média e dividindo pelo desvio padrão das amostras.

Um conjunto de algoritmos de classificação de *machine learning* foi escolhido para análises, com base em seu uso na literatura SHM e em sua interpretabilidade em termos de como usam atributos para predição de classes. Os algoritmos de classificação escolhidos foram: *Support Vector Machine* (SVM), *k-Nearest Neighbors* (k-NN), *Naive Bayes* (NB), *Linear Discriminant Analysis* (LDA), *Quadratic Discriminant Analysis* (QDA), *Random Forest* (RF), *Decision Trees* (DTs) e *Extremely Randomized Trees* (ET). Os códigos para os referidos classificadores foram desenvolvidos usando o pacote *scikit-learn* da linguagem Python 3.9 (PEDREGOSA *et al.*, 2011).

O procedimento de busca em grade, utilizando o método de validação cruzada *k-fold*, foi empregado para identificar os valores dos hiperparâmetros para cada modelo de classificação. A busca em grade, com validação cruzada, é o método mais amplamente usado

para otimizar os parâmetros de um classificador de *machine learning* (PEDREGOSA *et al.*, 2011). Esse método gera uma lista de modelos candidatos a partir de uma grade de valores de parâmetros. O tamanho da grade de hiperparâmetros é determinado pelo intervalo e combinação de hiperparâmetros escolhidos para cada modelo. Os hiperparâmetros ótimos são então escolhidos a partir do modelo com melhor desempenho de classificação. O intervalo e a combinação de hiperparâmetros, para cada um dos algoritmos de classificação, são escolhidos como o espaço de hiperparâmetros dentro do qual se espera que estejam os parâmetros do modelo ótimo para o problema de classificação.

O intervalo de hiperparâmetros foi escolhido com base nos intervalos de hiperparâmetros recomendados na literatura de SHM e *machine learning* (BUCKLEY, GHOSH e PAKRASHI, 2023; HASTIE, TIBSHIRANI e FRIEDMAN, 2017; KUHN e JOHNSON, 2013) e documentação do *scikit-learn* (PEDREGOSA *et al.*, 2011). Os hiperparâmetros do modelo para cada algoritmo de classificação são mostrados na Tabela 1. Para os métodos que possuem hiperparâmetros, 90% dos dados foram usados para treinamento e 10% para teste, e o modelo final foi ajustado com todos os dados.

Tabela 1 – Hiperparâmetros dos classificadores de ML, modelo de barra.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	$k = 3; 4; \dots; 25$
	Peso	Uniforme; distância
	Métrica	Cosseno; Euclidiana
SVM	C	0,1; 1; 10; 100; 1000
	Kernel	Linear; RBF; sigmoide; polinomial
	Grau	2; 3
	Gama (kernel não linear)	1; 0,1; 0,01; 0,001
	Função de decisão	Um contra um (OVO); um contra o resto (OVR)
RF	Número de estimadores	100; 150; 200; 300; 500
	Profundidade máxima da árvore	6; 8; 10; 12; 14
	Crítério	Gini; entropia
ET	Número de estimadores	100; 150; 200; 300; 500
	Profundidade máxima da árvore	12; 14; 16; 18; 20
	Crítério	Gini; entropia
DTs	Profundidade máxima da árvore	8; 10; 12; 14; 16
	Crítério	Gini; entropia

Fonte: Autor (2024).

4.2 Barra Engastada

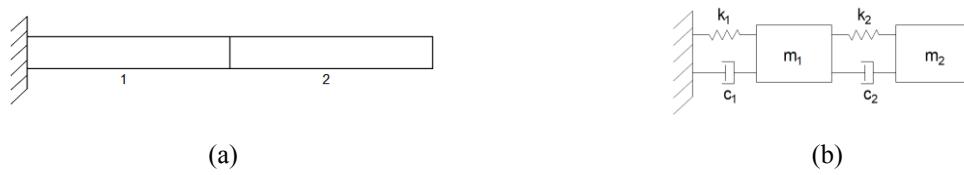
4.2.1 Barra Engastada Discretizada em Dois Elementos Finitos

O sistema foi discretizado em dois elementos finitos (Figura 11(a)) e foram aplicadas as condições de contorno fixa-livre (Equação 51) e a idealização da barra foi construída como um modelo 2-DOF (Figura 11(b)).

As duas frequências naturais, obtidas a partir das matrizes de massa e rigidez do sistema (problema de autovalor generalizado), foram $\{1326; 4634\}$ Hz e as duas razões de amortecimento obtidas foram $\{2,0; 2,0\}$ %, com $\alpha = 259,1883$ e $\beta = 1,0681 \times 10^{-6}$, estabelecidas para o modo fundamental e o segundo modo.

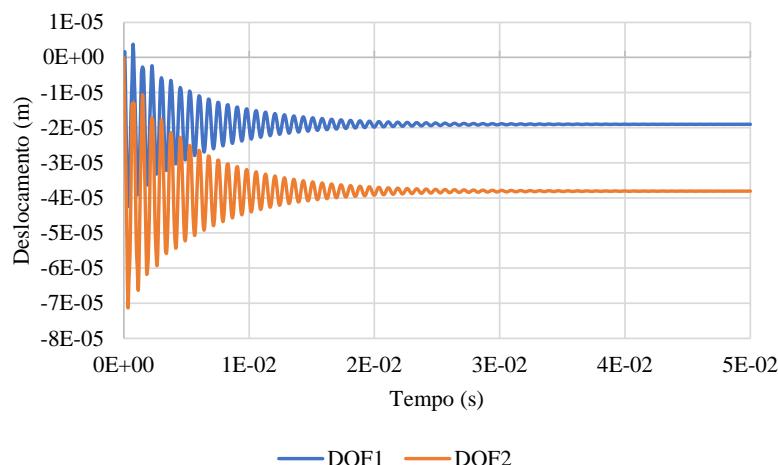
A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorre para o DOF 2, que está mais próximo do ponto de aplicação da força (Figura 12).

Figura 11 – (a) Barra engastada discretizada em dois elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 12 – Respostas de deslocamento no tempo, barra engastada discretizada em dois elementos finitos



Fonte: Autor (2024).

4.2.1.1 Respostas do SINDy

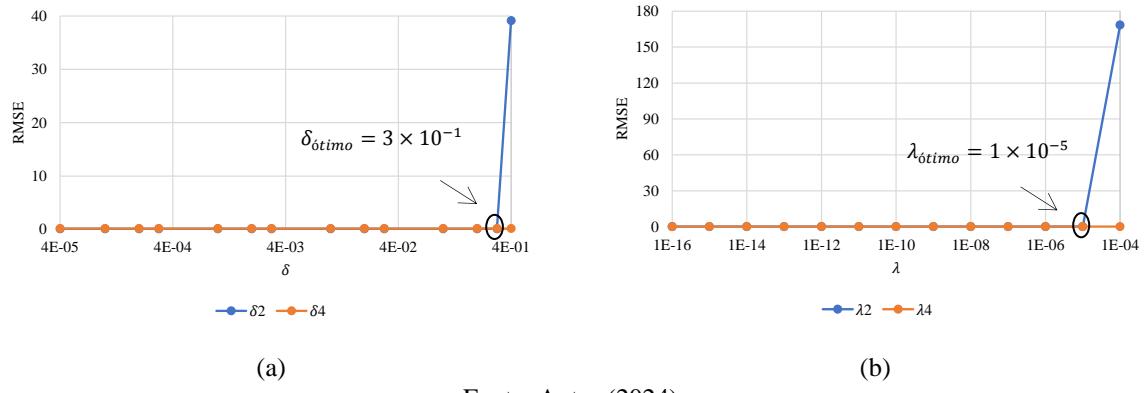
Para gerar os dados de treinamento para o SINDy, o sistema (Equação 51) foi simulado com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0]^T$, no tempo de 0 a 0,04 s, com período de amostragem de $8,3333 \times 10^{-5}$ s, que correspondem a sensores com taxa de amostragem de 12 kHz. Isso resultou em um conjunto de dados com 480 amostras.

As equações de movimento do modelo 2-DOF foram convertidas em um sistema de equações diferenciais de primeira ordem utilizando a Equação (53):

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left(\frac{f_1(t)}{m_1}\right) + \left(-\frac{k_1 + k_2}{m_1}\right)x_1 + \left(-\frac{\alpha(m_1) + \beta(k_1 + k_2)}{m_1}\right)x_2 + \left(\frac{k_2}{m_1}\right)x_3 \\ \quad + \left(\frac{\beta(k_2)}{m_1}\right)x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \left(\frac{f_2(t)}{m_2}\right) + \left(\frac{k_2}{m_2}\right)x_1 + \left(\frac{\beta(k_2)}{m_2}\right)x_2 + \left(-\frac{k_2}{m_2}\right)x_3 + \left(-\frac{\alpha(m_2) + \beta(k_2)}{m_2}\right)x_4. \end{cases} \quad (60)$$

O SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física (Equação 60). À medida que os resultados foram gerados pelo SINDy, para cada parâmetro δ (algoritmo STLSQ) e hiperparâmetro λ (algoritmo LASSO) variado, foram avaliados a raiz do erro quadrático médio (RMSE) entre os sinais de entrada de aceleração e a predição das equações descobertas para aceleração. A Figura 13 apresenta no eixo das ordenadas os valores do RMSE para os correspondentes valores de parâmetros δ (Figura 13(a)) e hiperparâmetros λ (Figura 13(b)), que estão no eixo das abscissas. Os valores de parâmetros δ_2 correspondem ao DOF 1, enquanto os valores de parâmetros δ_4 correspondem ao DOF 2, uma vez que as equações diferenciais foram convertidas em um sistema de primeira ordem. As mesmas ideias são realizadas para o hiperparâmetro λ . O parâmetro ótimo δ ($\delta_{óptimo} = 3 \times 10^{-1}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-5}$) foram escolhidos de forma que minimizasse a medida de erro (Figura 13). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 61).

Figura 13 – Determinação do (a) $\delta_{ótimo}$ e (b) $\lambda_{ótimo}$, barra engastada discretizada em dois elementos finitos.

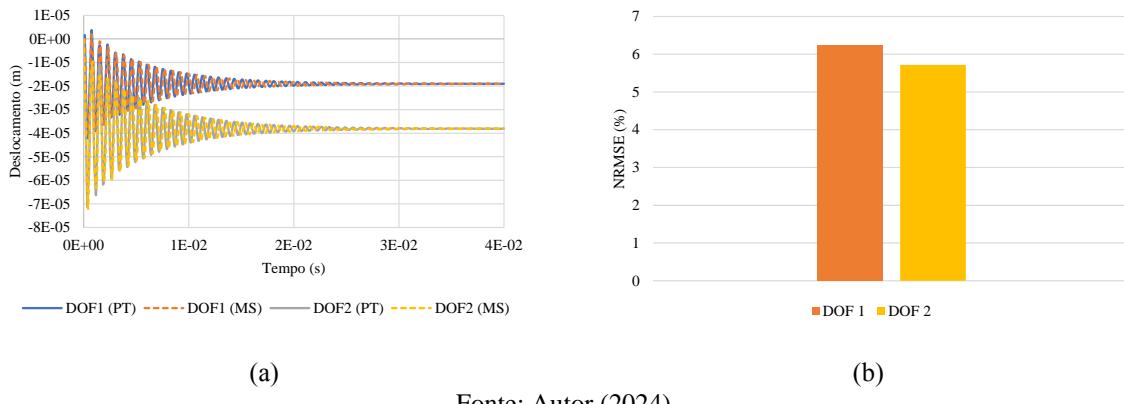


Fonte: Autor (2024).

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = 459,428 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 286002077,029x_1 - 747,538x_2 + (1-d_2)155061225,545x_3 \\ \quad + 292,208x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -4988,241 + (1-d_2)310122079,663x_1 + 585,687x_2 - (1-d_2)286002641,293x_3 \\ \quad - 746,631x_4. \end{cases} \quad (61)$$

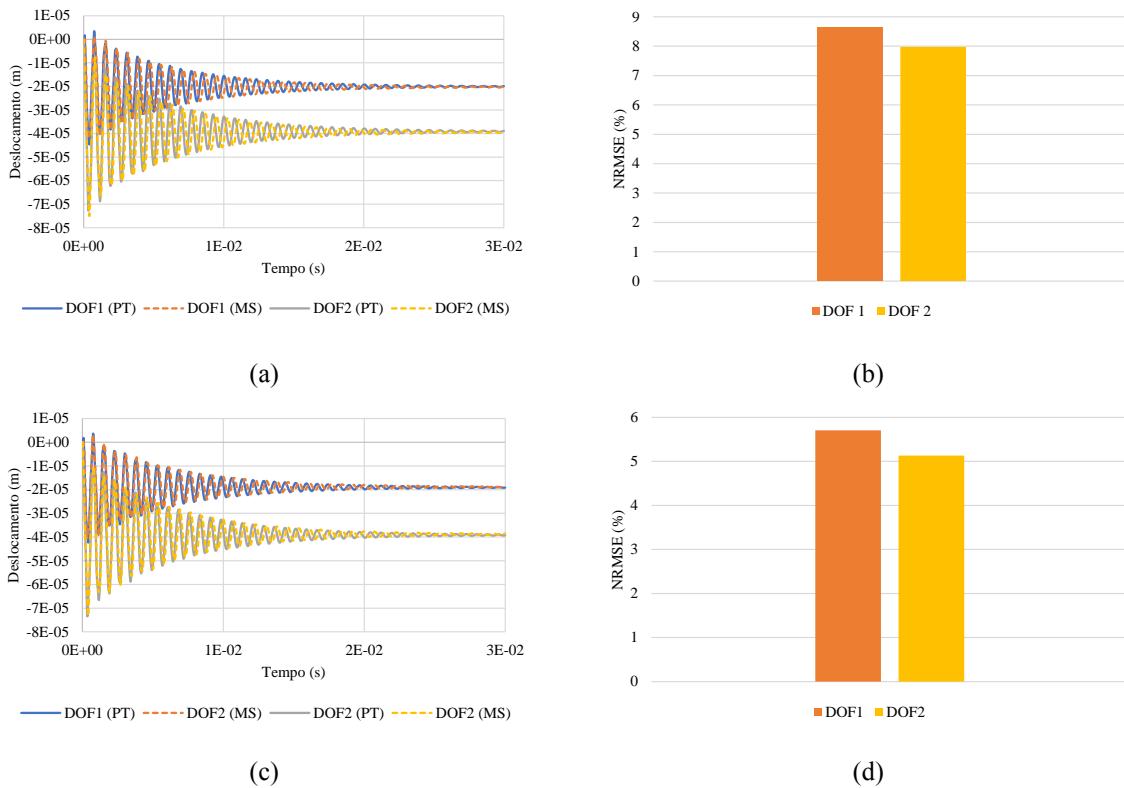
O modelo SINDy (Equação 61) foi integrado no tempo e comparado com a resposta de entrada (*physical twin*) (Figura 14(a)), sendo observada uma boa correspondência das respostas de deslocamento no tempo, com a raiz do erro quadrático médio normalizado (*normalized root mean square error* – NRMSE) para os DOFs 1 e 2 inferiores a 6,5% (Figura 14(b)). Também foram comparadas as respostas de deslocamento no tempo, do modelo SINDy e do *physical twin*, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 15 e 16), sendo apresentadas as respostas que apresentam menor e maior NRMSE. A resposta muda dependendo da intensidade do dano, bem como da mola com dano observada. Para a intensidade de dano de 5%, quando o dano é introduzido na mola 1 verifica-se uma correspondência bem representativa e maior NRMSE (inferior a 8,75%) (Figura 15(a-b)) e quando o dano está na mola 2 é observada uma boa correspondência, com NRMSE menor que 5,75% (Figura 15(c-d)). Para a intensidade de dano de 20%, quando o dano está na mola 1 é observada uma concordância satisfatória (com NRMSE inferior a 12,75%) (Figura 16(a-b)) e quando o dano está na mola 2 constata-se uma boa correspondência, com NRMSE menor que 2,75% (Figura 16(c-d)).

Figura 14 – (a) Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



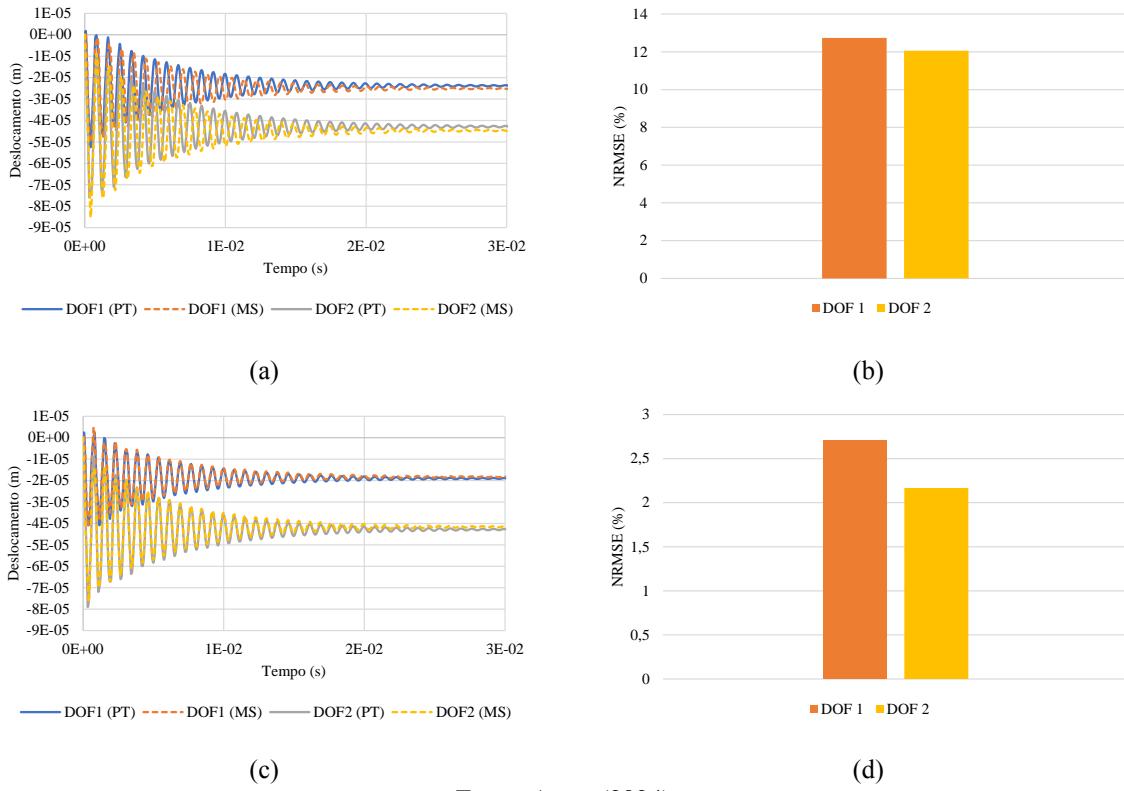
Fonte: Autor (2024).

Figura 15 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 16 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) primeira mola, (b) com correspondentes NRMSE, e (b) segunda mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

4.2.1.2 Resultados e Discussões dos Métodos de Machine Learning

Para construção do conjunto de dados usado para treinar os classificadores de *machine learning*, o modelo SINDy (Equação 61) foi integrado no tempo de 0 a 0,03 s, usando um período de amostragem de 1×10^{-4} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, 2$). Isso resultou em um conjunto de dados de treinamento de 2700 amostras (300×9), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 9) \times 2)$.

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, aplicada usando o conjunto de dados, são mostrados na Tabela 2. A Tabela 3 apresenta as métricas de desempenho dos classificadores. A precisão mostra que os métodos SVM e k-NN apresentaram os melhores resultados, seguido dos métodos baseados em árvores, método LDA e do método QDA, para os quais ocorre uma diminuição da precisão, e, por fim, o método NB, que apresenta desempenho bem inferior em relação aos outros métodos. Como é de interesse identificar danos no sistema, o falso negativo é de grande relevância, uma vez que pode ser um problema crítico

a estrutura estar com dano e o método indicar que a estrutura está íntegra. Portanto, um sistema construído para esse propósito deve levar em consideração uma taxa de falsos negativos próxima a zero. Uma métrica que pode ser utilizada para comparar sistemas diferentes é a revocação, valores altos de revocação indicam altos valores de verdadeiros positivos mesmo quando se leva em conta o total de falsos negativos, ou seja, um bom modelo deve ter alta revocação. Avaliando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 2 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	4
	Peso	Distância
	Métrica	Euclidiana
SVM	C	1000
	Kernel	RBF
	Gama (kernel não linear)	0,1
	Função de decisão	Um contra um (OVO)
RF	Número de estimadores	200
	Profundidade máxima da árvore	14
	Critério	Entropia
ET	Número de estimadores	200
	Profundidade máxima da árvore	20
	Critério	Gini
DTs	Profundidade máxima da árvore	16
	Critério	Entropia

Fonte: Autor (2024).

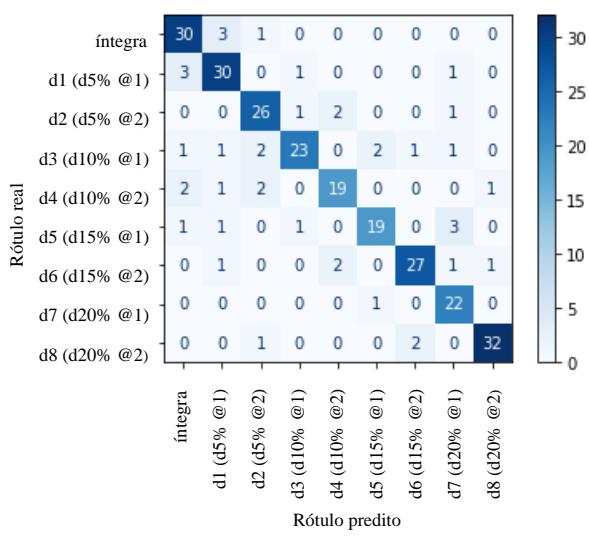
Tabela 3 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.

Classificador	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	84,44	84,54	84,25	0,84
k-NN	80,37	80,57	80,46	0,80
ET	78,15	78,11	78,68	0,78
RF	77,41	77,74	77,65	0,77
DTs	74,44	74,83	74,44	0,74
LDA	74,96	74,48	74,96	0,74
QDA	69,22	69,70	69,22	0,69
NB	18,52	21,78	18,52	0,15

Fonte: Autor (2024).

É analisada a matriz de confusão para o método SVM (Figura 17), que apresentou melhor desempenho. Cada linha representa o cenário de dano real, e as colunas exibem os resultados do classificador. A primeira linha mostra que 30 das 34 amostras (soma da linha) para a condição da estrutura íntegra estão classificadas corretamente. Três amostras foram classificadas incorretamente como d1 (dano de 5% na primeira mola) e 1 amostra classificada incorretamente como d2 (dano de 5% na segunda mola). De forma análoga, para as linhas sucessivas, as estatísticas para identificação de danos podem ser obtidas. Portanto, os termos diagonais estão relacionados às amostras classificadas corretamente, e os termos fora da diagonal estão relacionados às amostras classificadas incorretamente. Há muitas informações nesta matriz, como a quantificação de verdadeiros positivos e falsos negativos. A probabilidade de classificar corretamente um dano é muito alta na primeira mola com dano de 20% (95,7%) (22/23) e na segunda mola com dano de 20% (91,4%) (32/35), e alta na segunda mola com dano de 5% (86,7%) (26/30), na primeira mola com dano de 5% (85,7%) (30/35), na segunda mola com dano de 15% (84,4%) (27/32), na segunda mola com dano de 10% (76%) (19/25), na primeira mola com dano de 15% (76%) (19/25) e na primeira mola com dano de 10% (74,2%) (23/31). Ainda, a probabilidade de indicar uma estrutura íntegra quando o sistema está com dano (falso negativo) na primeira mola com dano de 5% é 8,6% (3/35), na primeira mola com dano de 10% é 3,2% (1/31), na segunda mola com dano de 10% é 8% (2/25) e na primeira mola com dano de 15% é 4% (1/25). Outra informação que pode ser obtida dessa matriz é a probabilidade de considerar um dano quando o sistema está íntegro (falso positivo), que corresponde a 11,8% (4/34).

Figura 17 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em dois elementos finitos.



Fonte: Autor (2024).

4.2.1.3 Aplicações do Framework Digital Twin

Para ilustração do *framework* DT do classificador SVM, o *physical twin* foi simulado no tempo de 0 a 0,03 s, usando um período de amostragem de 2×10^{-5} s, que correspondeu a 1500 amostras para cada um dos cenários de interesse (dano investigado – DI), condição íntegra e de 1% até 25% de dano em cada elemento, para verificar qual cenário de dano treinado será associado (dano associado – DA), podendo ser íntegra, 5% de dano, 10% de dano, 15% de dano e 20% de dano. Esses dados de entrada (sinais simulados) não são rotulados, uma vez que na prática não é conhecida a classe à qual pertence o sinal medido. Portanto, o *framework* DT apresenta a probabilidade de os dados pertencerem a determinada classe, sendo apresentada a maior probabilidade (P). Várias quantidades de ruído foram adicionadas aos dados de inferência (correspondentes a 1500 amostras) para avaliar a tolerância do modelo de classificação ao ruído. Um ruído gaussiano aleatório foi adicionado, com média zero e diferentes níveis de ruído (1%, 3%, 5% e 10%), obtidos a partir de um desvio padrão que compreende o fenômeno em análise, correspondente a 5×10^{-6} m.

A aplicação mostra os seguintes resultados para a primeira mola (Tabela 4), nas condições analisadas sem ruído, 1% de ruído e 3% de ruído: de 0% a 2% de dano, indica que a estrutura está íntegra; de 3% a 7% de dano, indica que a estrutura está com dano de 5%; de 8% a 12% de dano, indica que a estrutura está com dano de 10%; de 13% a 16% de dano, indica que a estrutura está com dano de 15% e, por fim, de 17% a 25% de dano, indica que estrutura está com dano de 20%. Para 5% de ruído, ocorre uma diminuição da probabilidade de identificação de dano nos cenários investigados, no entanto, as maiores probabilidades são constatadas para dano de 20%. Para 10% de ruído, as maiores probabilidades ocorrem para dano de 20%. As respostas para a segunda mola estão no Apêndice.

Tabela 4 – Respostas do *framework* DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em dois elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	85,07	84,67	64,93		51,40		29,87	
1%	84,27	78,33	58,27	Íntegra	49,73		30,07	
2%	80,33	59,73	50,47		43,67		28,53	
3%	73,40	54,53	41,07		40,47		26,27	
4%	78,93	73,67	46,73		33,40		28,13	
5%	80,00	77,80	48,87		31,93		23,67	
6%	79,80	73,53	47,80		33,13		22,47	
7%	75,20	54,80	42,27		31,47		22,07	
8%	73,67	58,67	44,60		35,53		20,47	
9%	78,73	75,67	50,47		36,47		22,87	
10%	81,00	79,47	53,67		37,40		23,27	
11%	80,40	73,60	50,73		35,53		27,27	
12%	73,67	54,60	44,13		34,27		28,67	
13%	70,60	58,13	41,40		30,47		31,93	
14%	75,33	72,87	47,87		31,20		34,53	
15%	76,73	74,53	48,87		34,07		38,60	
16%	75,87	65,40	42,87		37,67		43,47	
17%	67,53	48,00	47,60		46,87		46,67	
18%	83,27	75,87	57,87		56,20		52,33	
19%	85,07	84,13	70,27		61,93		58,27	
20%	86,87	86,47	77,87		68,47		59,53	
21%	88,13	88,07	83,93		75,80		63,73	
22%	88,60	88,47	87,13		79,53		68,33	
23%	88,73	88,80	88,40		83,07		70,40	
24%	89,53	89,47	89,07		85,87		76,73	
25%	90,80	90,73	90,53		88,40		78,00	

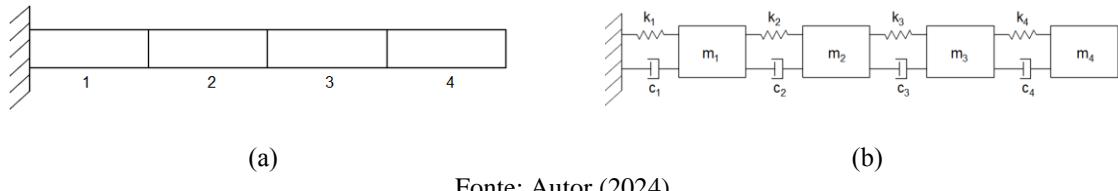
Fonte: Autor (2024).

4.2.2 Barra Engastada Discretizada em Quatro Elementos Finitos

O sistema foi discretizado em quatro elementos finitos (Figura 18(a)) e idealizado como modelo de 4-DOF (Figura 18(b)). As três primeiras frequências naturais obtidas foram {1301; 4105; 7458} Hz e as três primeiras razões de amortecimento obtidas foram {2,0; 1,2; 1,5} %, com $\alpha = 291,8517$ e $\beta = 5,2676 \times 10^{-7}$, estabelecidas para o modo fundamental e o quarto modo.

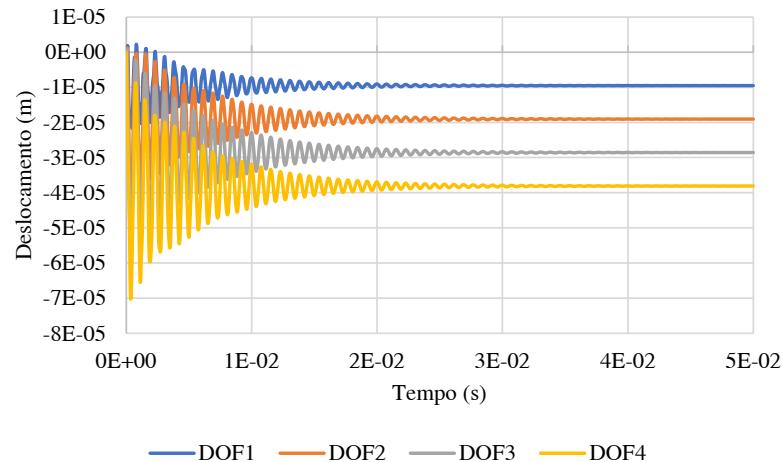
A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para os DOFs que estão mais próximos do ponto de aplicação da força (Figura 19).

Figura 18 – (a) Barra engastada discretizada em quatro elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 19 – Respostas de deslocamento no tempo, barra engastada discretizada em quatro elementos finitos.



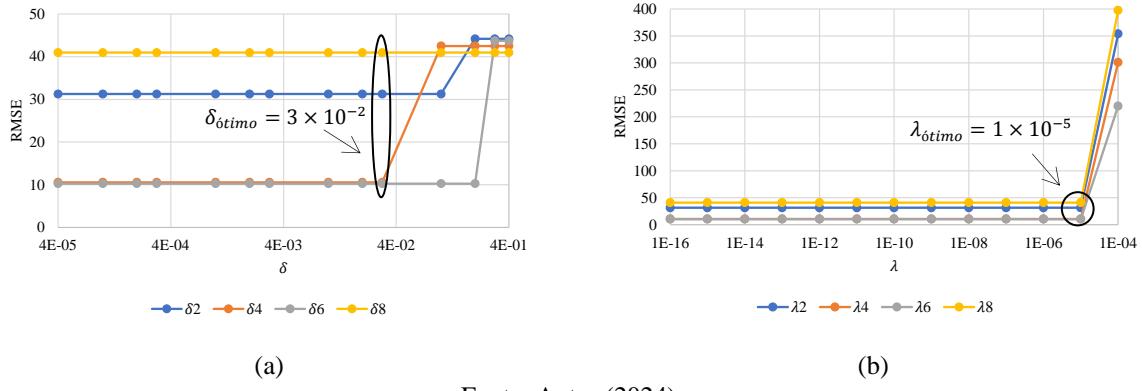
Fonte: Autor (2024).

4.2.2.1 Respostas do SINDy

O sistema foi simulado com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0, 0, 0]^T$, no tempo de 0 a $0,04\text{ s}$, com período de amostragem de $4 \times 10^{-5}\text{ s}$, que corresponde a sensores com taxa de amostragem de 25 kHz. Isso resultou em um conjunto de dados com 1000 amostras.

O parâmetro ótimo δ ($\delta_{óptimo} = 3 \times 10^{-2}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-5}$) foram determinados de forma que minimizasse a medida de RMSE (Figura 20). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 62).

Figura 20 – Determinação do (a) $\delta_{\text{óptimo}}$ e (b) $\lambda_{\text{óptimo}}$, barra engastada discretizada em quatro elementos finitos.



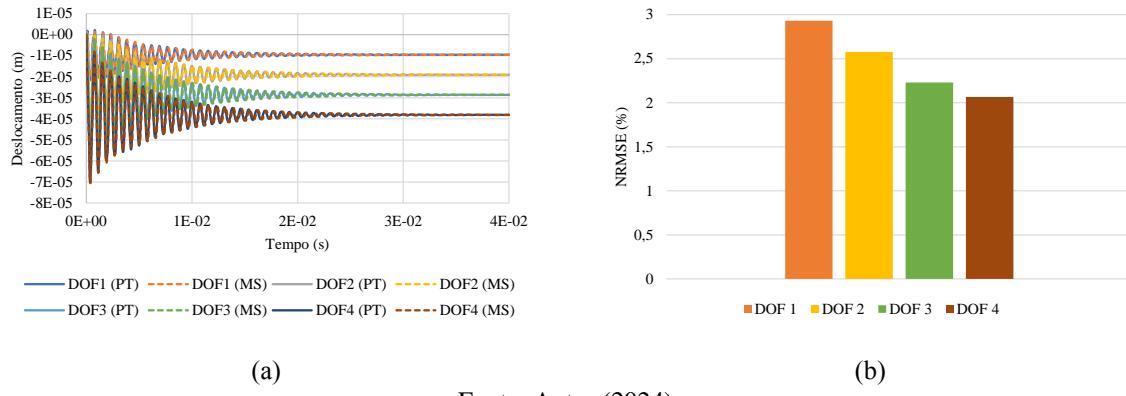
Fonte: Autor (2024).

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = 160,041 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 1057912720,767x_1 - 848,171x_2 + (1 - d_2)537352655,210x_3 \\ \quad + 277,279x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = 305,440 + (1 - d_2)637429664,452x_1 + 360,989x_2 - \left(1 - \frac{d_2}{2} - \frac{d_3}{2}\right) 1143327449,657x_3 \\ \quad - 915,973x_4 + (1 - d_3)560428386,284x_5 + 300,041x_6 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = 953,504 + (1 - d_3)577427586,222x_3 + 600,552x_4 - \left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right) 1183763540,630x_5 \\ \quad - 1600,078x_6 + (1 - d_4)624138668,116x_7 + 752,041x_8 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = -9714,180 + (1 - d_4)1087683486,632x_5 + 1377,734x_6 - (1 - d_4)1070764725,287x_7 \\ \quad - 1588,453x_8. \end{array} \right. \quad (62)$$

O modelo SINDy (Equação 62) foi integrado no tempo e comparado com a resposta de entrada (Figura 21(a)), sendo observada uma ótima correspondência das respostas, e os valores de NRMSE para os DOFs são inferiores a 3% (Figura 21(b)). Também foram comparadas as respostas de deslocamento no tempo, do modelo SINDy e do *physical twin*, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 22 e 23), sendo apresentadas as respostas que apresentam menor e maior erro (NRMSE). A resposta muda dependendo da intensidade do dano, bem como da mola com dano observada. Quando o dano de 5% está na mola 1 verifica-se uma boa correspondência e maior NRMSE (NRMSE inferior a 5%) (Figura 22(a-b)), e quando o dano de 5% está na mola 4 é observada uma ótima correspondência e menor erro, os resultados de NRMSE são inferiores a 2,5% (Figura 22(c-d)). Para o dano de 20%, quando o dano é introduzido na mola 1 constata-se uma concordância satisfatória (com maior NRMSE entre 9 e 11,5%) (Figura 23(a-b)) e

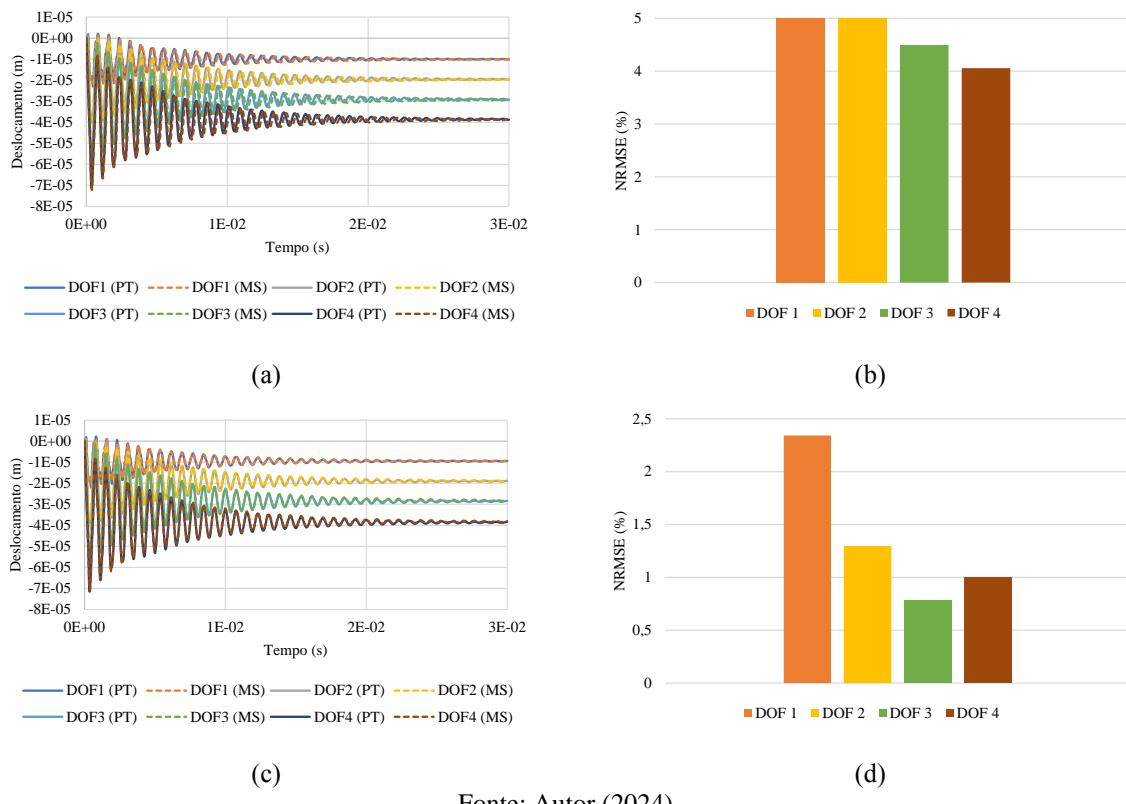
quando o dano de 20% está na mola 2 é observada uma ótima correspondência e menor erro (NRMSE inferior a 2,5%) (Figura 23(c-d)).

Figura 21 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



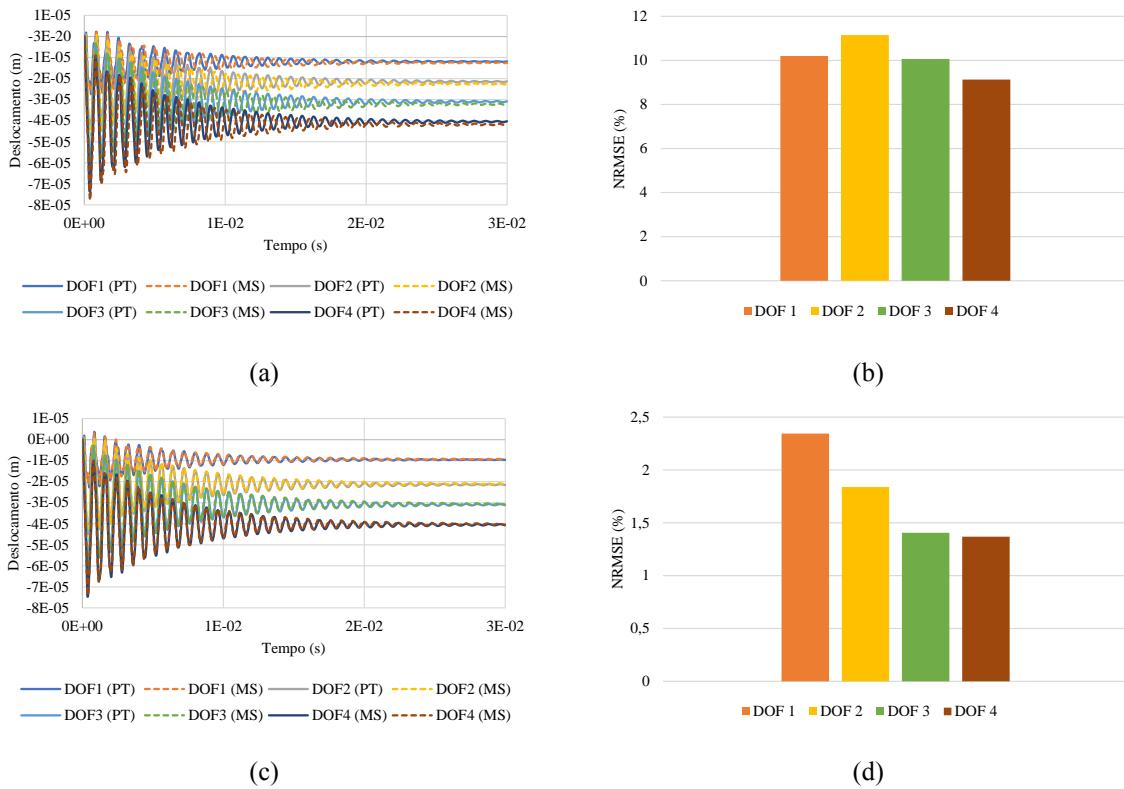
Fonte: Autor (2024).

Figura 22 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 23 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

4.2.2.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 62) foi integrado no tempo de 0 a 0,03 s, usando um período de amostragem de 1×10^{-4} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, \dots, 4$). Isso resultou em um conjunto de dados de treinamento de 5100 amostras (300×17), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 17) \times 4)$.

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, aplicada usando o conjunto de dados são mostrados na Tabela 5. A Tabela 6 apresenta as métricas de desempenho para os classificadores. A precisão indica que os métodos SVM e QDA apresentaram os melhores desempenhos, seguido dos métodos LDA, k-NN e os baseados em árvores, novamente, o método NB apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 5 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	3
	Peso	Distância
	Métrica	Euclidiana
SVM	C	1000
	Kernel	Linear
	Função de decisão	Um contra um (OVO)
RF	Número de estimadores	500
	Profundidade máxima da árvore	14
	Critério	Entropia
ET	Número de estimadores	150
	Profundidade máxima da árvore	18
	Critério	Gini
DTs	Profundidade máxima da árvore	16
	Critério	Entropia

Fonte: Autor (2024).

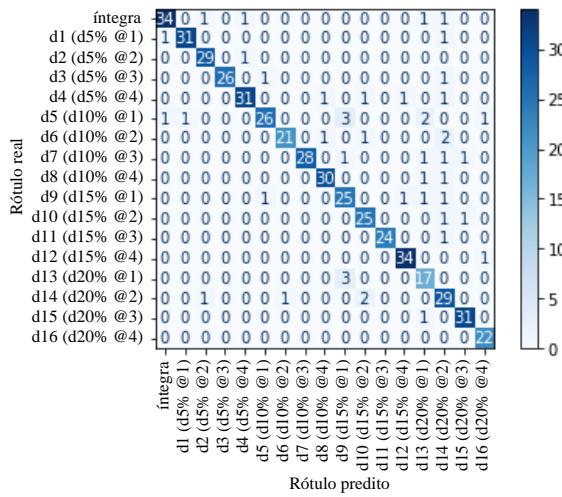
Tabela 6 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.

Classificador	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	90,78	91,09	90,88	0,91
QDA	90,25	90,86	90,25	0,90
LDA	83,96	83,71	83,96	0,84
k-NN	76,07	76,63	76,49	0,76
ET	72,35	72,66	72,74	0,72
RF	71,18	71,45	71,69	0,71
DTs	68,43	70,16	68,85	0,69
NB	10,88	15,03	10,88	0,09

Fonte: Autor (2024).

A matriz de confusão do método SVM mostra que a probabilidade de considerar um dano quando o sistema está íntegro é 10,5% (4/38) e a probabilidade de indicar uma estrutura íntegra quando a primeira mola está com dano de 5% é 3% (1/33) e quando a primeira mola está com dano de 10% é 2,9% (1/34) (Figura 24).

Figura 24 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

4.2.2.3 Aplicações do Framework Digital Twin

O *physical twin* foi simulado no tempo de 0 a 0,03 s, usando um período de amostragem de 2×10^{-5} s, que correspondeu a 1500 amostras para cada um dos cenários de interesse, condição íntegra e de 1% até 25% de dano em cada elemento. Um ruído gaussiano aleatório foi adicionado aos dados (1500 amostras), com média zero e diferentes níveis de ruído, a partir de um desvio padrão de $2,60 \times 10^{-6}$ m, que compreende o desvio padrão do fenômeno em análise.

A aplicação do framework DT indica os seguintes resultados para a primeira mola (Tabela 7), nas condições sem ruído, 1% de ruído e 3% de ruído: de 0% a 2% de dano, a estrutura está íntegra; de 3% a 7% de dano, a estrutura está com dano de 5%; de 8% a 12% de dano, a estrutura está com dano de 10%; de 13% a 17%, a estrutura está com dano de 15% e, por fim, de 18% a 25% de dano, a estrutura está com dano de 20%. Para 5% de ruído, ocorre uma diminuição da probabilidade de identificação de dano, no entanto, as maiores probabilidades são apresentadas para dano de 20%. Para 10% de ruído, as maiores probabilidades ocorrem para dano de 20%. As respostas para as demais molas estão no Apêndice.

Tabela 7 – Respostas do *framework* DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		DA
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	
0%	85,87	83,67	52,80	Íntegra	29,40	14,73		
1%	83,40	77,93	46,47		28,73	16,40		
2%	77,80	57,80	37,00		24,00	13,33		
3%	74,60	56,73	34,93		22,20	14,47		
4%	79,60	74,60	40,00		22,87	15,40		
5%	80,20	78,60	45,67		23,07	16,13		
6%	78,87	74,47	41,40		24,87	15,73		
7%	71,47	51,47	40,13		25,80	14,80		
8%	75,73	63,20	44,87		31,40	15,20		
9%	79,13	76,20	55,00		34,27	15,47		
10%	80,60	79,20	56,27		34,60	17,13		
11%	78,33	75,40	52,40		33,47	19,60		
12%	70,13	50,80	41,20		34,80	22,67		
13%	77,00	68,73	53,07		42,33	23,67		
14%	79,80	79,07	62,80		45,00	26,67		
15%	80,13	79,80	66,33		45,27	32,47		
16%	78,73	76,00	59,13		45,47	38,20		
17%	70,13	53,07	45,40		45,67	41,60		
18%	82,80	78,13	63,20		58,20	49,33		
19%	87,33	86,73	77,67		66,67	54,13		
20%	89,60	89,40	84,93		76,13	63,13		
21%	91,13	90,93	89,40		83,47	66,47		
22%	92,60	92,67	91,60		88,67	69,20		
23%	93,60	93,47	93,33		90,27	77,13		
24%	94,20	94,20	93,47		92,60	80,87		
25%	94,47	94,33	94,20		93,47	84,07		

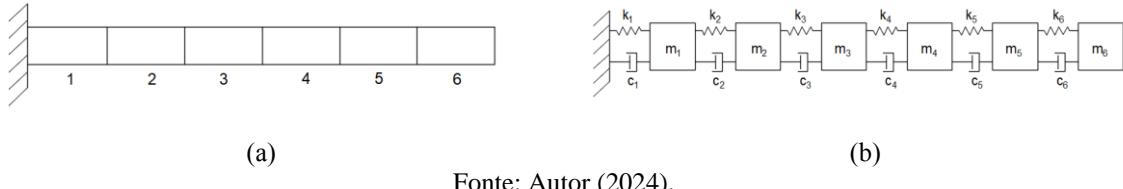
Fonte: Autor (2024).

4.2.3 Barra Engastada Discretizada em Seis Elementos Finitos

O sistema foi discretizado em seis elementos finitos (Figura 25(a)) e representado como modelo de 6-DOF (Figura 25(b)). As três primeiras frequências naturais obtidas foram {1297; 3979; 6930} Hz e as duas razões de amortecimento obtidas foram {2,0; 1,0; 1,1} %, com $\alpha = 302,3999$ e $\beta = 3,5411 \times 10^{-7}$, estabelecidas para o modo fundamental e o sexto modo.

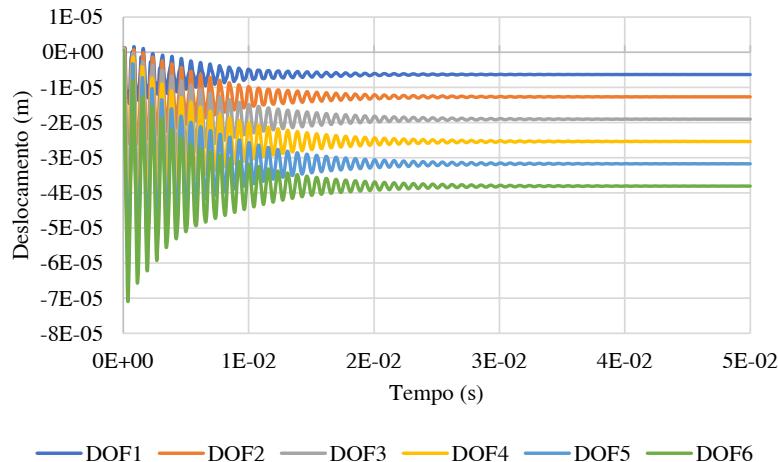
As respostas de deslocamento no tempo são apresentadas na Figura 26, mostrando que o sistema oscila de forma rápida e as maiores amplitudes ocorrem para os DOFs que estão mais próximos do ponto de aplicação da força.

Figura 25 – (a) Barra engastada discretizada em seis elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 26 – Respostas de deslocamento no tempo, barra engastada discretizada em seis elementos finitos



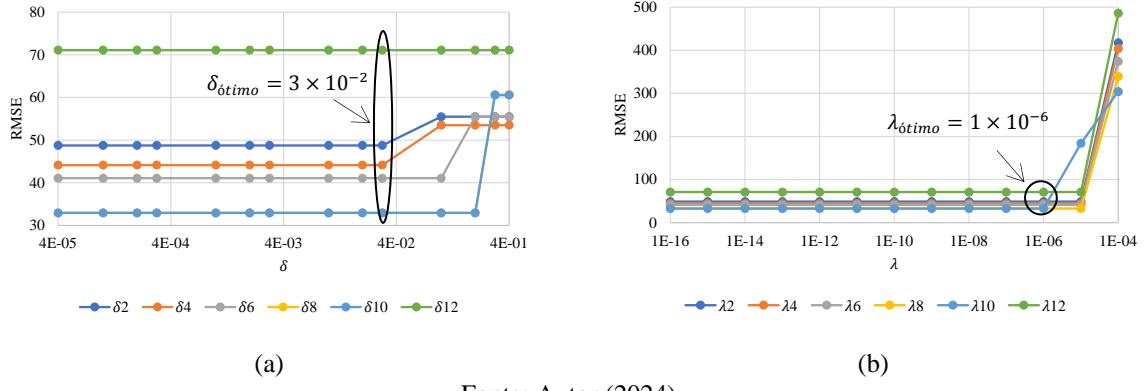
Fonte: Autor (2024).

4.2.3.1 Respostas do SINDy

O sistema foi simulado com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$, no tempo de 0 a 0,04 s, com período de amostragem de $2,5 \times 10^{-5}$ s, que corresponde a sensores com taxa de amostragem de 40 kHz. Isso resultou em um conjunto de dados com 1600 amostras.

O parâmetro ótimo δ ($\delta_{óptimo} = 3 \times 10^{-2}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-6}$) foram escolhidos de forma que minimizasse a medida de RMSE (Figura 27). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 63).

Figura 27 – Determinação do (a) $\delta_{\text{óptimo}}$ e (b) $\lambda_{\text{óptimo}}$, barra engastada discretizada em seis elementos finitos.



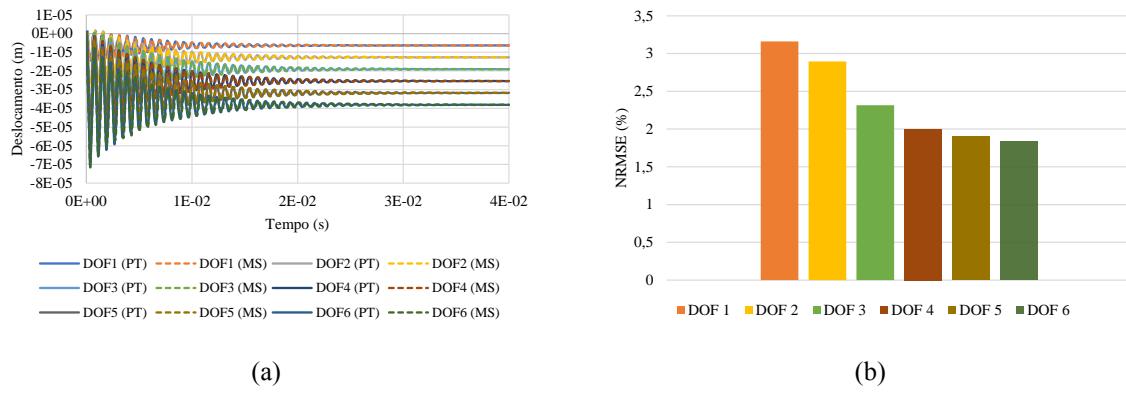
Fonte: Autor (2024).

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = 120,710 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 2363127733,487x_1 - 1077,953x_2 + (1 - d_2)1191067236,552x_3 \\ \quad + 381,955x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = 160,591 + (1 - d_2)1456560637,342x_1 + 184,494x_2 - \left(1 - \frac{d_2}{2} - \frac{d_3}{2}\right) 2525378734,397x_3 \\ \quad - 776,547x_4 + (1 - d_3)1206495186,072x_5 + 240,567x_6 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = 248,318 + (1 - d_3)1316271294,278x_3 + 335,930x_4 - \left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right) 2486409542,476x_5 \\ \quad - 943,039x_6 + (1 - d_4)1216445368,318x_7 + 305,661x_8 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = 507,755 + (1 - d_4)1281851519,765x_5 + 585,608x_6 - \left(1 - \frac{d_4}{2} - \frac{d_5}{2}\right) 2530536531,606x_7 \\ \quad - 1410,958x_8 + (1 - d_5)1271307335,705x_9 + 544,542x_{10} \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = 1675,871 + (1 - d_5)1257346166,867x_7 + 1493,529x_8 - \left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right) 2703517402,506x_9 \\ \quad - 4026,713x_{10} + (1 - d_6)1458693097,918x_{11} + 2288,672x_{12} \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = -14340,963 + (1 - d_6)2346362864,977x_9 + 2897,921x_{10} - (1 - d_6)2331762418,059x_{11} \\ \quad - 3094,836x_{12}. \end{array} \right. \quad (63)$$

O modelo SINDy (Equação 63) foi integrado no tempo e comparado com a resposta de entrada (Figura 28), sendo observada uma ótima correspondência das respostas (Figura 28(a)) e o NRMSE para os DOFs são inferiores a 3,25% (Figura 28(b)). Também foram comparadas as respostas de deslocamento no tempo, do modelo SINDy e do *physical twin*, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 29 e 30), sendo apresentadas as respostas que apresentam menor e maior erro (NRMSE). A resposta muda dependendo da intensidade do dano, bem como da mola com dano observada. Quando o dano de 5% está na mola 2 é apresentada uma ótima concordância e menor erro, com NRMSE inferior a 3,25% (Figura 29(a-b)) e quando o dano de 5% está na

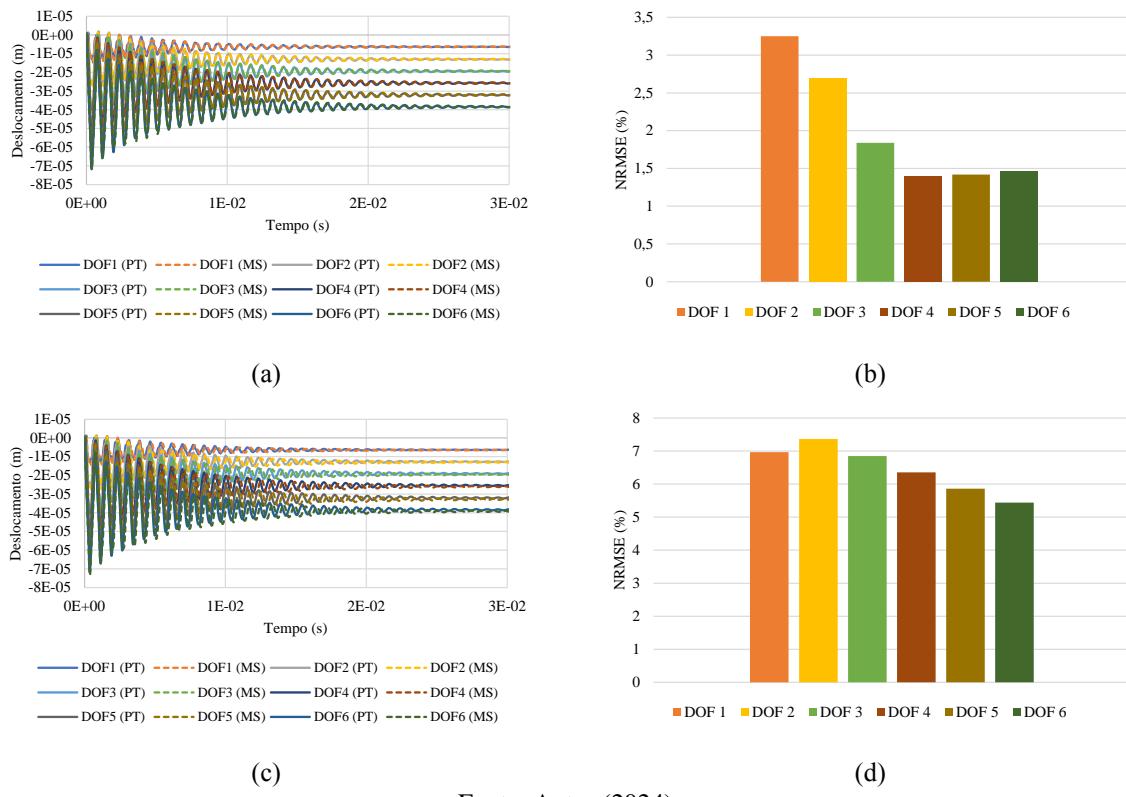
mola 5 é observada uma correspondência bem representativa, com NRMSE entre 5,5 a 7,5% (Figura 29(c-d)). Para o dano de 20%, quando o dano de 20% está na mola 3 é observada uma ótima correspondência e menor erro (NRMSE inferior a 3,25%) (Figura 30(a-b)) e quando o dano é introduzido na mola 1 é apresentada uma concordância satisfatória (com maior NRMSE entre 12,5 a 16,25%) (Figura 30(c-d)).

Figura 28 – (a) Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



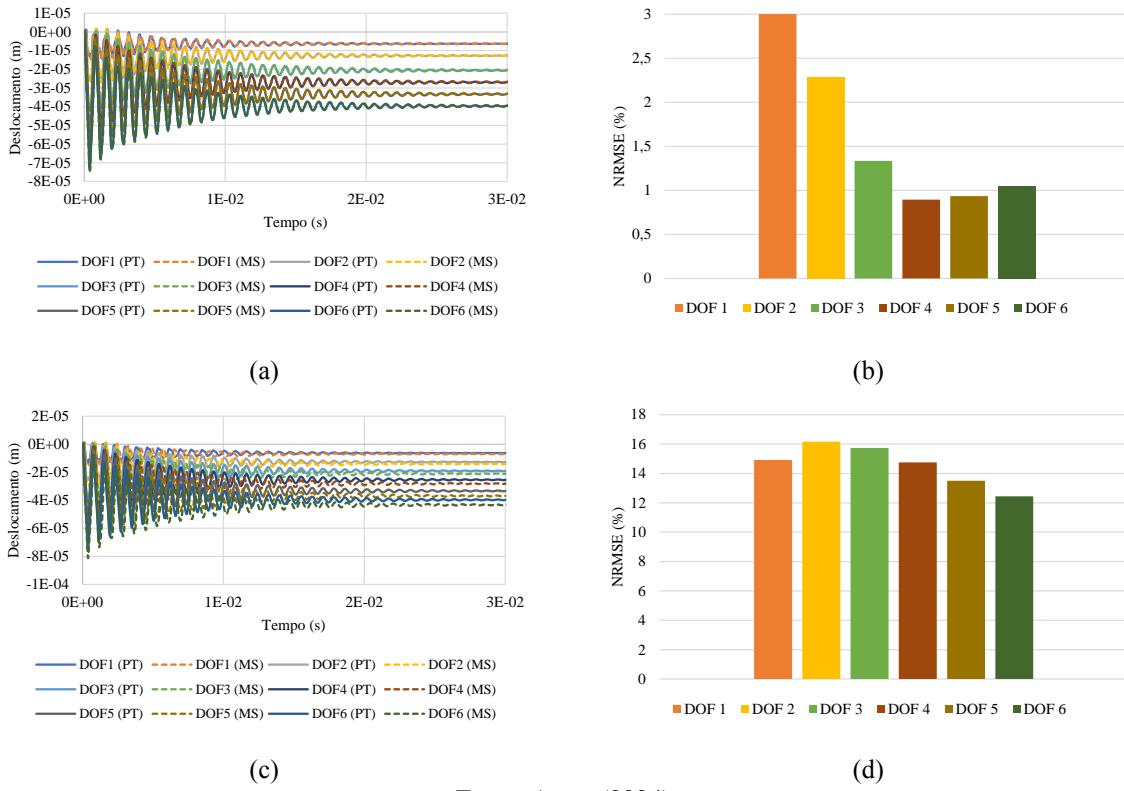
Fonte: Autor (2024).

Figura 29 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS), para dano de 5% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) quinta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 30 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra engastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quinta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

4.2.3.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 63) foi integrado no tempo de 0 a 0,03 s, usando um período de amostragem de 1×10^{-4} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, \dots, 6$). Isso resultou em um conjunto de dados de treinamento de 7500 amostras (300×25), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 25) \times 6)$.

Foram obtidos os hiperparâmetros ótimos da busca em grade, com validação cruzada, usando o conjunto de dados (Tabela 8) e as respectivas métricas de desempenho dos classificadores (Tabela 9). A precisão indica que os métodos SVM e QDA apresentaram os melhores resultados, seguido dos métodos LDA, k-NN e os baseados em árvores, mais uma vez, o método NB apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 8 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	3
	Peso	Distância
	Métrica	Euclidiana
SVM	C	1000
	Kernel	Linear
	Função de decisão	Um contra um (OVO)
RF	Número de estimadores	200
	Profundidade máxima da árvore	14
	Critério	Entropia
ET	Número de estimadores	300
	Profundidade máxima da árvore	20
	Critério	Gini
DTs	Profundidade máxima da árvore	16
	Critério	Entropia

Fonte: Autor (2024).

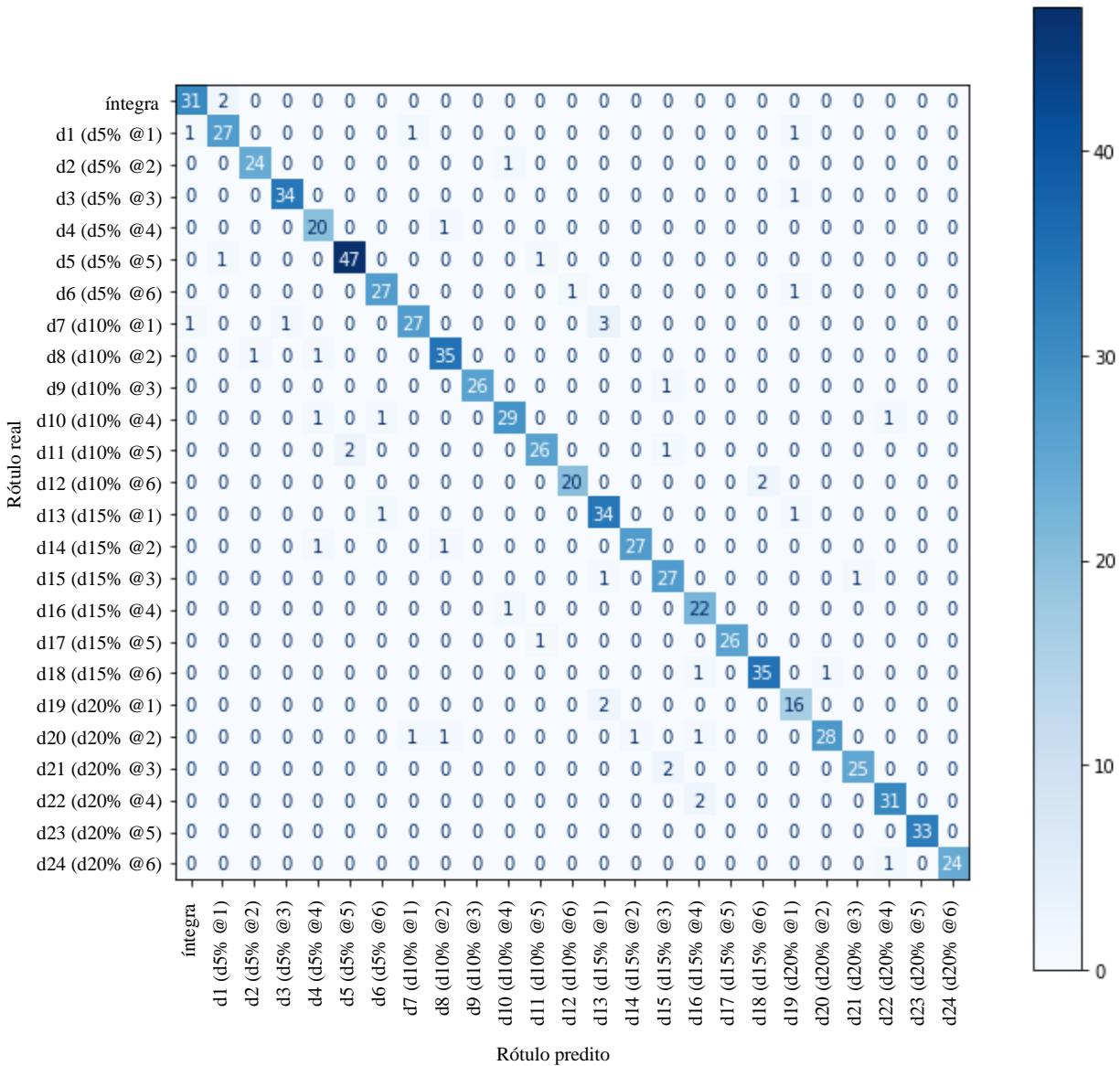
Tabela 9 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.

Classificador	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	93,47	93,37	93,36	0,93
QDA	92,48	93,21	92,48	0,93
LDA	84,71	84,49	84,71	0,84
k-NN	70,40	71,13	71,40	0,70
ET	69,47	69,66	70,51	0,69
RF	67,60	68,38	68,79	0,68
DTs	63,20	68,01	64,25	0,65
NB	7,55	11,55	7,55	0,05

Fonte: Autor (2024).

A matriz de confusão do método SVM apresenta que a probabilidade de considerar um dano quando o sistema está íntegro é 6,06% (2/33), enquanto a probabilidade de indicar uma estrutura íntegra quando a primeira mola está com dano de 5% é de 3,4% (1/29) e quando a primeira mola está com dano de 10% é 3,1% (1/32) (Figura 31).

Figura 31 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra engastada discretizada em seis elementos finitos.



Fonte: Autor (2024).

4.2.3.3 Aplicações do Framework Digital Twin

O *physical twin* foi simulado no tempo de 0 a 0,03 s, usando um período de amostragem de 2×10^{-5} s, que correspondeu a 1500 amostras para cada um dos cenários de interesse, condição íntegra e de 1% até 25% de dano em cada elemento. Um ruído gaussiano aleatório foi adicionado aos dados (1500 amostras), com média zero e diferentes níveis de ruído, a partir de um desvio padrão de $1,70 \times 10^{-6}$ m, que compreende o desvio padrão do fenômeno em análise.

A aplicação do *framework DT* apresenta para a primeira mola (Tabela 10) os seguintes resultados nas condições sem ruído e 1% de ruído: de 0% a 2% de dano, a estrutura está íntegra;

de 3% a 7% de dano, a estrutura está com dano de 5%; de 8% a 12% de dano, a estrutura está com dano de 10%; de 13% a 16% de dano, a estrutura está com dano de 15% e, por fim, de 17% a 25% de dano, a estrutura está com dano de 20%. Para 3% de ruído, as maiores probabilidades de identificação de dano são apresentadas a partir de 10% de dano. Para 5% e 10% de ruído, as maiores probabilidades ocorrem para dano de 20%. As respostas para as demais molas estão no Apêndice.

Tabela 10 – Respostas do *framework* DT do classificador SVM para a primeira mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído	10% de ruído			
	P (%)	P (%)	DA	P (%)	P (%)	DA	P (%)	DA
0%	88,53	85,73		38,73	17,13		11,20	
1%	87,20	81,20		36,47	17,13		12,60	
2%	81,60	60,93		30,40	16,80		12,00	
3%	81,40	62,80		31,27	16,73		12,33	
4%	85,07	82,20		37,53	19,20		12,33	
5%	85,73	83,53		41,27	17,20		12,53	
6%	84,67	79,73		38,20	17,80		13,80	
7%	76,73	55,53		33,40	21,07		12,80	
8%	82,73	71,27		45,53	24,33		14,40	
9%	85,73	84,20		51,20	24,00		13,13	
10%	86,47	85,20		51,87	28,13		15,33	
11%	84,93	76,67		48,67	27,87		15,60	
12%	58,53	46,13		40,27	29,20		16,13	
13%	84,67	77,20		50,47	32,27		17,67	
14%	86,53	85,27		56,80	33,87		18,67	
15%	87,40	85,53		56,93	36,60		21,47	
16%	84,40	69,13		49,60	34,20		26,40	
17%	78,80	53,33		50,80	40,53		27,87	
18%	88,33	82,27		61,27	50,13		33,53	
19%	90,87	90,20		74,27	59,20		35,27	
20%	92,13	91,93		81,53	66,53		40,00	
21%	92,93	92,80		87,20	71,93		41,87	
22%	93,33	93,20		89,47	77,67		46,93	
23%	93,67	93,60		92,53	80,53		47,87	
24%	93,87	93,73		92,73	83,60		50,67	
25%	94,47	94,00		93,20	85,33		57,73	

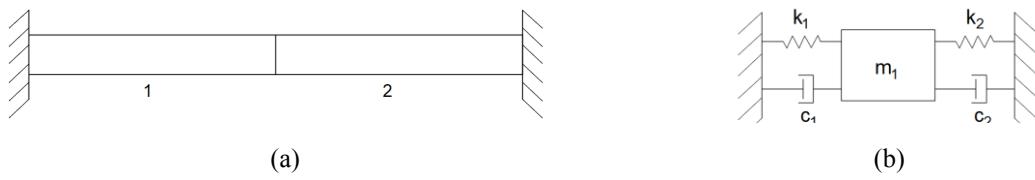
Fonte: Autor (2024).

4.3 Barra Biengastada

4.3.1 Barra Biengastada Discretizada em Dois Elementos Finitos

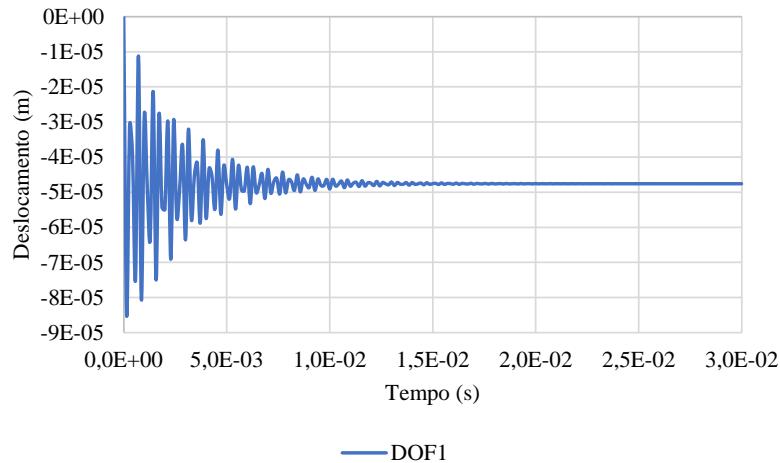
O sistema foi discretizado em dois elementos finitos (Figura 32(a)) e idealizado como modelo de 1-DOF (Figura 32(b)). A frequência natural obtida foi $\{2851\}$ Hz. Foram adotadas as constantes de proporcionalidade $\alpha = 518,3765$ e $\beta = 5,3404 \times 10^{-7}$, oriundas do modelo de barra biengastada discretizada em quatro elementos finitos. A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada (Figura 33).

Figura 32 – (a) Barra biengastada discretizada em dois elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 33 – Resposta de deslocamento no tempo, barra biengastada discretizada em dois elementos finitos



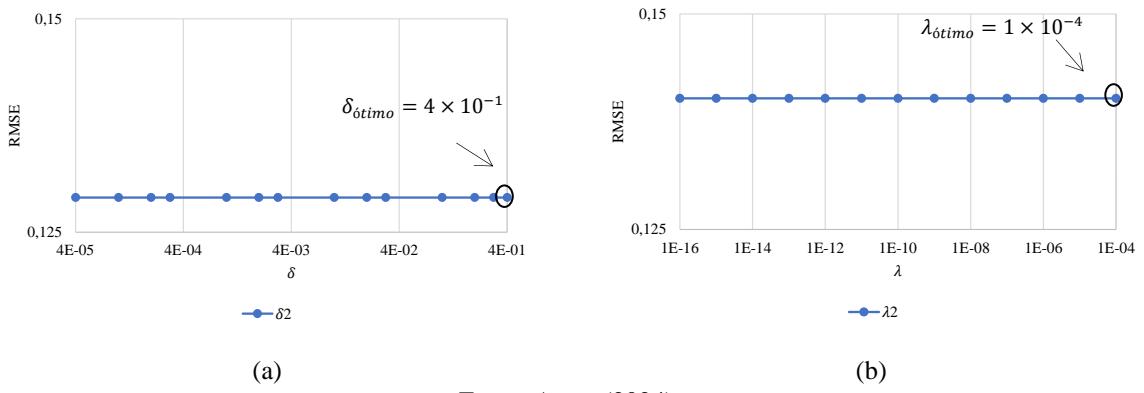
Fonte: Autor (2024).

4.3.1.1 Respostas do SINDy

O sistema foi simulado com as condições iniciais $x(t = 0) = [0,0]^T$, no tempo de 0 a $0,02$ s, com período de amostragem de $1,4286 \times 10^{-4}$ s, que corresponde a sensores com taxa de amostragem de 7 kHz. Isso resultou em um conjunto de dados com 140 amostras.

O parâmetro ótimo δ ($\delta_{óptimo} = 2 \times 10^{-2}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-4}$) foram determinados de forma que minimizasse a medida de RMSE (Figura 34). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 64).

Figura 34 – Determinação do (a) $\delta_{óptimo}$ e (b) $\lambda_{óptimo}$, barra biengastada discretizada em dois elementos finitos.

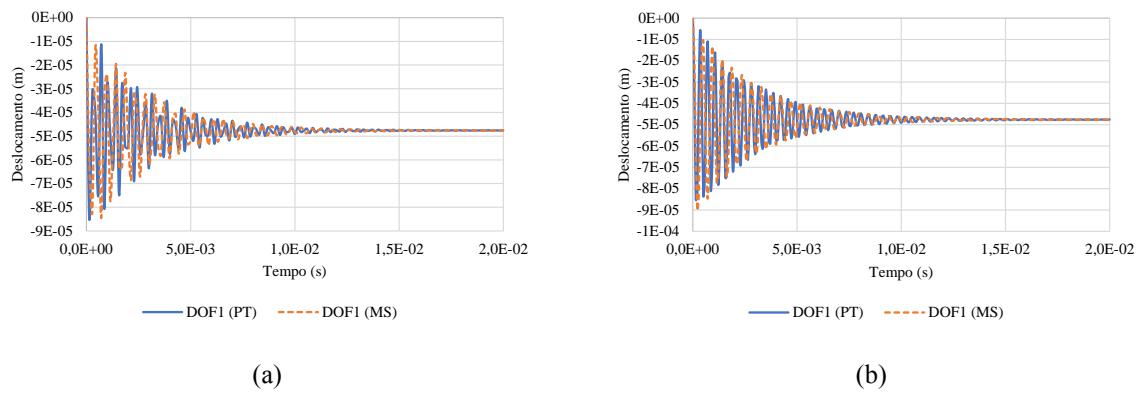


Fonte: Autor (2024).

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -8560,396 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 179768334,214x_1 - 690,652x_2. \end{cases} \quad (64)$$

O modelo SINDy (Equação 64) foi integrado no tempo e comparado com a resposta de entrada (Figura 35(a)), sendo observada uma comparação satisfatória e NRMSE de 14,85%. Para melhor visualização da resposta, o sistema foi amostrando com uma taxa de amostragem de 14 kHz (Figura 35(b)), o NRMSE correspondente foi de 13,02%.

Figura 35 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em dois elementos finitos, e modelo SINDy (MS) considerando a taxa de amostragem de (a) 7 kHz e (b) 14 kHz, respectivamente.



Fonte: Autor (2024).

4.3.1.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 64) foi integrado no tempo de 0 a 0,015 s, usando um período de amostragem de 5×10^{-5} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, 2$). Isso resultou em um conjunto de dados de treinamento de 2700 amostras (300×9), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 9) \times 1)$.

Como os resultados dos classificadores não foram satisfatórios, quando considerada uma taxa de amostragem de 7 kHz, que corresponde a aproximadamente duas vezes e meia a frequência do modelo de alta fidelidade, foi realizada outra análise usando uma taxa de amostragem de 14 kHz, correspondente a cinco vezes a frequência do modelo. O procedimento foi repetido e obtido um novo conjunto de dados final na forma $((300 \times 9) \times 1)$.

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, aplicada usando o conjunto de dados de 7 kHz e 14 kHz são apresentadas na Tabela 11. A Tabela 12 apresenta as métricas de desempenho dos classificadores para as duas situações, indicando desempenho insatisfatório.

Tabela 11 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em dois elementos finitos.

Classificador	Parâmetro	Valor	
		7 kHz	14 kHz
QDA	-	-	-
LDA	-	-	-
NB	Modelo	Gaussiano	Gaussiano
k-NN	Número de vizinhos	24	24
	Peso	Uniforme	Uniforme
	Métrica	Euclidiana	Euclidiana
SVM	C	1000	1000
	Kernel	RBF	RBF
	Gama (kernel não linear)	1	1
	Função de decisão	Um contra um (OVO)	Um contra um (OVO)
RF	Número de estimadores	300	300
	Profundidade máxima da árvore	2	2
	Critério	Gini	Gini
ET	Número de estimadores	150	150
	Profundidade máxima da árvore	3	3
	Critério	Gini	Gini
DTs	Profundidade máxima da árvore	3	3
	Critério	Gini	Gini

Fonte: Autor (2024).

Tabela 12 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em dois elementos finitos.

Classificador	Acurácia (%)		Precisão (%)		Revocação (%)		F-score	
	7 kHz	14 kHz	7 kHz	14 kHz	7 kHz	14 kHz	7 kHz	14 kHz
SVM	20,74	21,85	14,97	14,54	22,41	23,46	0,16	0,17
LDA	24,26	24,19	13,01	13,23	24,26	24,19	0,16	0,16
ET	21,11	21,11	11,45	11,75	22,87	22,95	0,14	0,15
RF	22,96	21,85	13,44	16,43	24,39	22,57	0,15	0,16
DTs	22,59	21,85	17,12	13,30	22,91	22,27	0,18	0,15
k-NN	13,33	14,04	10,53	13,09	13,44	13,69	0,11	0,13
QDA	13,19	13,26	5,34	5,69	13,19	13,26	0,06	0,07
NB	13,19	13,26	5,37	5,70	13,19	13,26	0,06	0,07

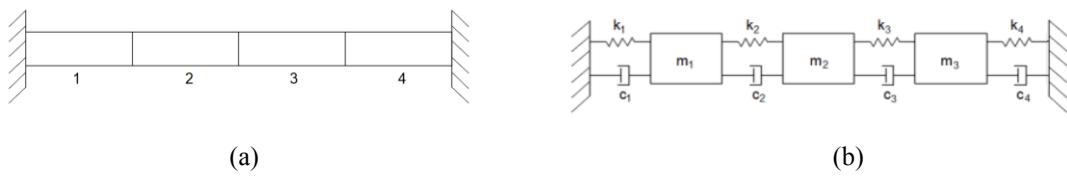
Fonte: Autor (2024).

4.3.2 Barra Biengastada Discretizada em Quatro Elementos Finitos

O sistema foi discretizado em quatro elementos finitos (Figura 36(a)) e idealizado como modelo de 3-DOF (Figura 36(b)). As três frequências naturais obtidas foram $\{2653; 5703; 9268\}$ Hz e as três primeiras razões de amortecimento obtidas foram $\{2,0; 1,7; 2,0\}$ %, com $\alpha = 518,3765$ e $\beta = 5,3404 \times 10^{-7}$, estabelecidas para o modo fundamental e o terceiro modo.

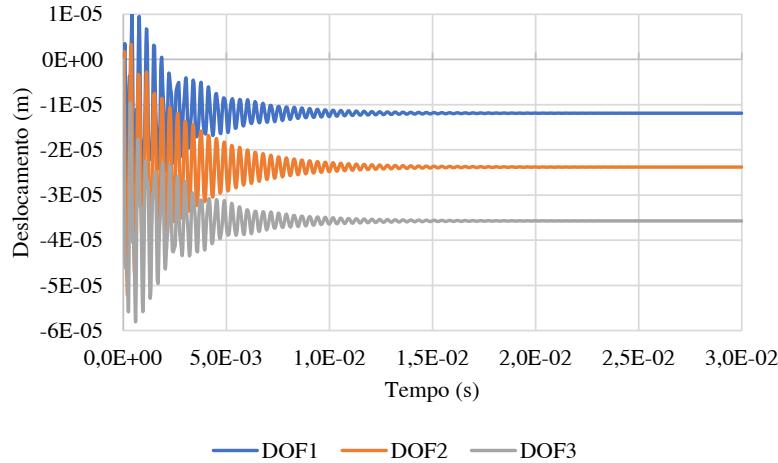
A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para os DOFs que estão mais próximos do ponto de aplicação da força (Figura 37).

Figura 36 – (a) Barra biengastada discretizada em quatro elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 37 – Respostas de deslocamento no tempo, barra biengastada discretizada em quatro elementos finitos



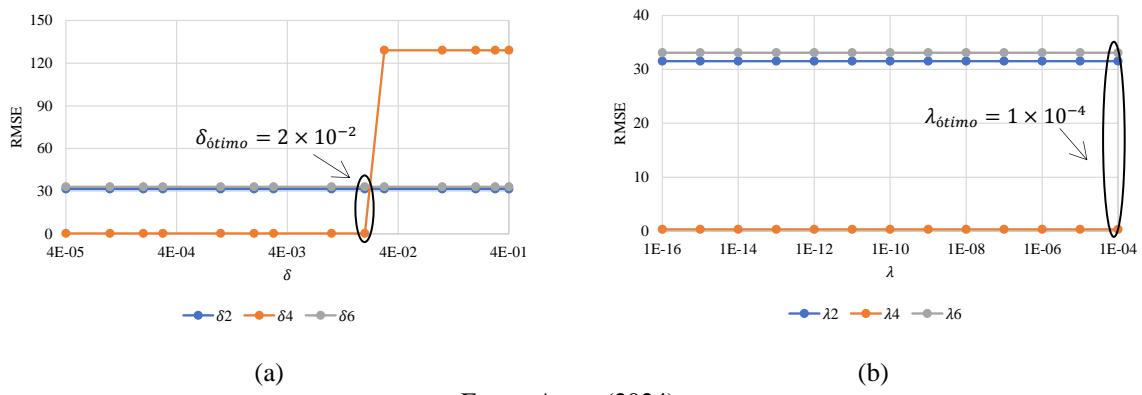
Fonte: Autor (2024).

4.3.2.1 Respostas do SINDy

O sistema foi simulado com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0]^T$, no tempo de 0 a 0,02 s, com período de amostragem de $4,3478 \times 10^{-5}$ s, que corresponde a sensores com taxa de amostragem de 23 kHz. Isso resultou em um conjunto de dados com 460 amostras.

O parâmetro ótimo δ ($\delta_{\text{óptimo}} = 2 \times 10^{-2}$) e o hiperparâmetro ótimo λ ($\lambda_{\text{óptimo}} = 1 \times 10^{-4}$) foram determinados de forma que minimizasse a medida de RMSE (Figura 38). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 65).

Figura 38 – Determinação do (a) $\delta_{\text{óptimo}}$ e (b) $\lambda_{\text{óptimo}}$, barra biengastada discretizada em quatro elementos finitos.

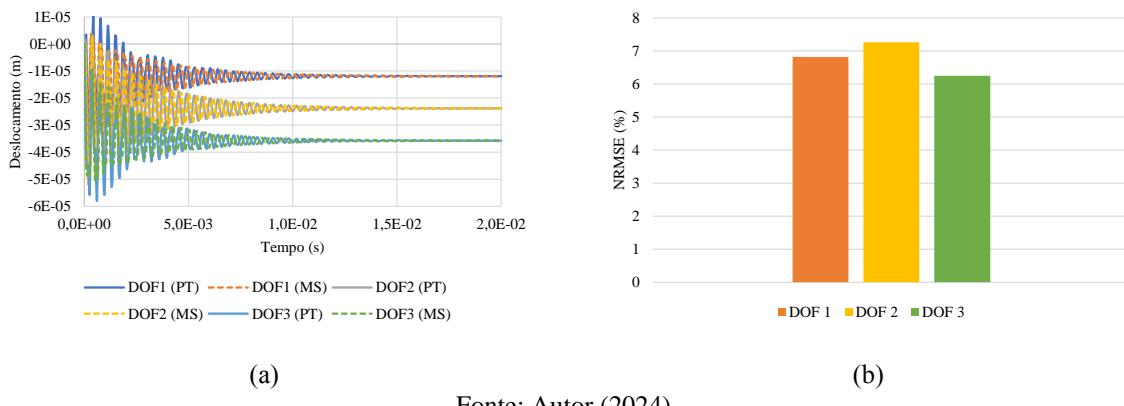


Fonte: Autor (2024).

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = 734,191 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 1052602976,874x_1 - 1182,783x_2 + (1 - d_2)557119326,092x_3 \\ \quad + 359,321x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = 1847,961 + (1 - d_2)586372155,950x_1 + 586,427x_2 - \left(1 - \frac{d_2}{2} - \frac{d_3}{2}\right) 1095127926,492x_3 \\ \quad - 1497,707x_4 + (1 - d_3)586370681,297x_5 + 587,312x_6 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = -24290,724 + (1 - d_3)555457705,429x_3 + 459,502x_4 - \left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right) 1050452131,947x_5 \\ \quad - 1310,705x_6. \end{cases} \quad (65)$$

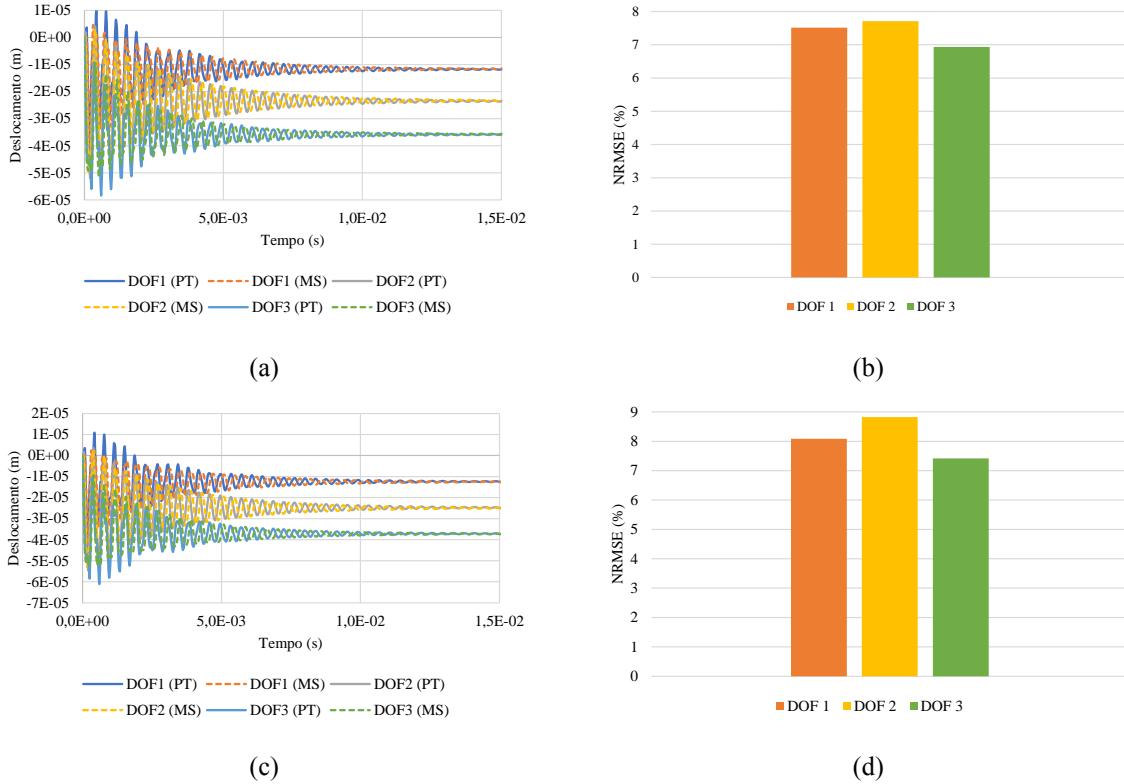
O modelo SINDy (Equação 65) foi integrado no tempo e comparado com a resposta de entrada (Figura 39(a)), sendo observada uma correspondência bem representativa das respostas, e o NRMSE para os DOFs são inferiores a 7,5% (Figura 39(b)). Também foram comparadas as respostas de deslocamento no tempo, do modelo SINDy e do *physical twin*, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 40 e 41), sendo apresentadas as respostas que apresentam maior e menor erro (NRMSE). Para o dano de 5% é observada uma correspondência bem representativa, com menor erro na mola 3 (NRMSE inferior a 8%) (Figura 40(a-b)), enquanto o maior erro ocorre quando o dano está na mola 4, com NRMSE inferior a 9% (Figura 40(c-d)). Para o dano de 20%, quando o dano é introduzido na mola 2 verifica-se uma concordância representativa, com menor NRMSE (sendo inferior a 7%) (Figura 41(a-b)) e quando o dano de 20% está na mola 4 é observada uma resposta bem representativa (NRMSE inferior a 10%) (Figura 41(c-d)).

Figura 39 – (a) Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



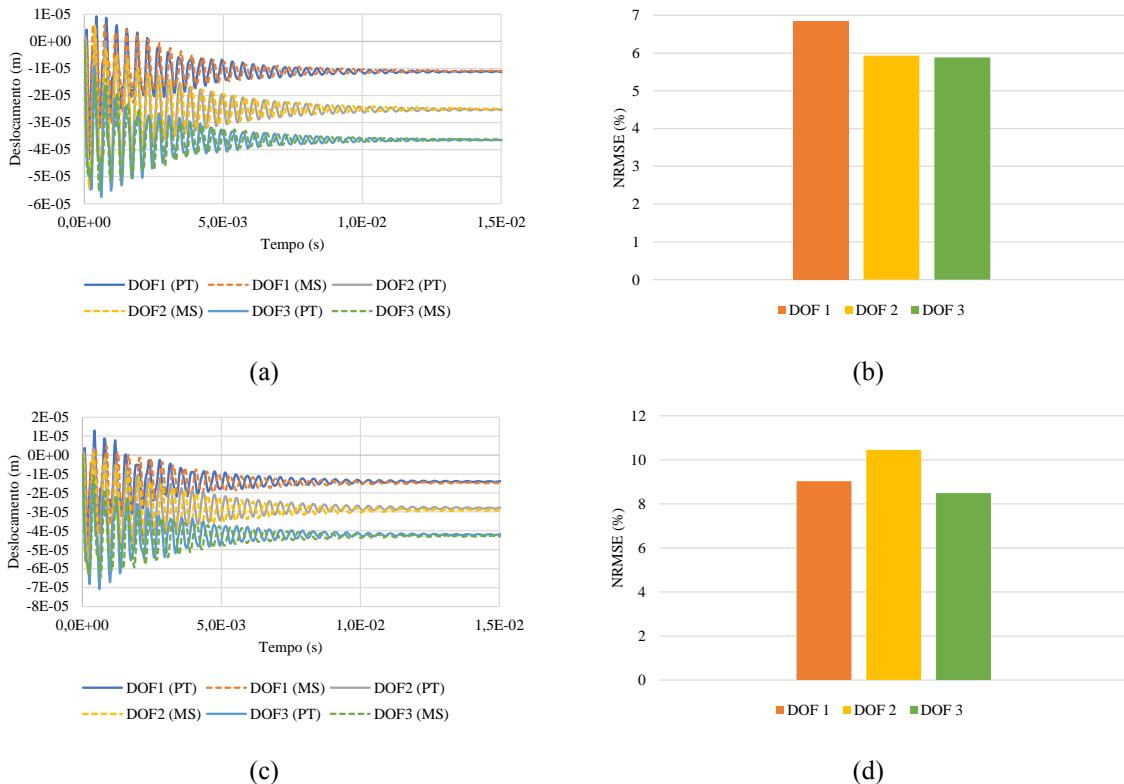
Fonte: Autor (2024).

Figura 40 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 41 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

4.3.2.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 65) foi integrado no tempo de 0 a 0,015 s, usando um período de amostragem de 5×10^{-5} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, \dots, 4$). Isso resultou em um conjunto de dados de treinamento de 5100 amostras (300×17), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 17) \times 3)$.

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, usando o conjunto de dados são mostrados na Tabela 13. A Tabela 14 apresenta as métricas de desempenho para os classificadores. A precisão indica que os métodos SVM apresentou melhor desempenho, seguido dos métodos k-NN, LDA e baseados em árvores, o método QDA apresentou uma diminuição considerável de precisão e, por fim, o método NB apresentou desempenho inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 13 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	6
	Peso	Distância
	Métrica	Euclidiana
SVM	C	1000
	Kernel	RBF
	Gama (kernel não linear)	1
	Função de decisão	Um contra um (OVO)
RF	Número de estimadores	150
	Profundidade máxima da árvore	14
	Critério	Gini
ET	Número de estimadores	500
	Profundidade máxima da árvore	20
	Critério	Entropia
DTs	Profundidade máxima da árvore	14
	Critério	Entropia

Fonte: Autor (2024).

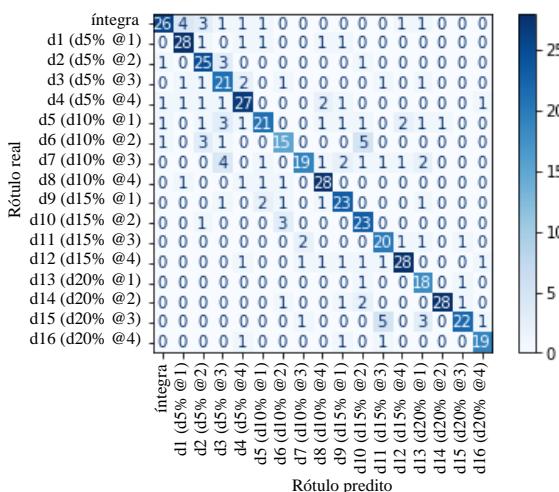
Tabela 14 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.

Classificador	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	76,66	77,10	77,17	0,76
k-NN	65,29	66,20	65,71	0,65
LDA	63,29	62,84	63,29	0,62
RF	62,55	63,02	63,06	0,62
ET	60,59	61,06	61,30	0,60
DTs	57,45	59,05	57,86	0,58
QDA	37,78	37,59	37,78	0,34
NB	10,49	14,88	10,49	0,08

Fonte: Autor (2024).

Na matriz de confusão do método SVM é constatado que a probabilidade de considerar um dano quando o sistema está íntegro é 31,6% (12/38) e a probabilidade de indicar uma estrutura íntegra quando a segunda mola está com dano de 5% é 3,3% (1/30), quando a quarta mola está com dano de 5% é 2,9% (1/35), quando a primeira mola está com dano de 10% é 2,9% (1/34) e quando a segunda mola está com dano de 10% é 4% (1/25) (Figura 42).

Figura 42 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra biengastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

4.3.2.3 Aplicações Framework Digital Twin

O *physical twin* foi simulado no tempo de 0 a 0,02 s, usando um período de amostragem de $1,3333 \times 10^{-5}$ s, que correspondeu a 1500 amostras para cada um dos cenários de interesse, condição íntegra e de 1% até 25% de dano em cada elemento. Um ruído gaussiano aleatório foi

adicionado aos dados (1500 amostras), com média zero e diferentes níveis de ruído, a partir de um desvio padrão de $4,50 \times 10^{-6}$ m, que compreende o desvio padrão do fenômeno em análise.

A aplicação do *framework* DT apresenta para a primeira mola (Tabela 15) os seguintes resultados para as condições sem ruído e 1% de ruído: de 0% a 2% de dano, a estrutura está intacta; de 3% a 6% de dano, a estrutura está com dano de 5%; de 7% a 12% de dano, a estrutura está com dano de 10%; de 13% a 16% de dano, a estrutura está com dano de 15% e, por fim, de 17% a 25% de dano a estrutura está com dano de 20%. Para 3% e 5% de ruído, maiores probabilidades de identificação de dano são apresentadas para dano de 20%. Quando utilizado 10% de ruído, verifica-se uma baixa probabilidade, não identificando o elemento com dano. As respostas para as demais molas estão no Apêndice.

Tabela 15 – Respostas do *framework* DT do classificador SVM para a primeira mola, considerando a barra biengastada discretizada em quatro elementos finitos.

DI	sem ruído		1% de ruído		3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA	Dano 5% @2						
0%	79,53	63,13		29,07		20,73		14,27		
1%	79,00	55,80		26,93		18,40		14,07		
2%	73,53	44,13		25,07		17,27		13,47		
3%	74,27	49,53		24,13		17,20		13,40		
4%	76,87	55,73		26,33		15,93		14,80		
5%	77,67	58,07		26,40		17,13		11,13		
6%	77,47	51,27		28,47		17,53		12,20		
7%	53,47	38,73		25,67		18,07		12,13		
8%	74,80	50,33		24,80		17,53		12,27		
9%	76,20	60,07		24,87		16,67		12,60		
10%	75,93	64,80		28,60		15,53		11,20		
11%	75,40	59,13		29,00		20,07		13,07		
12%	69,00	49,07		29,93		21,00		13,93		
13%	71,67	39,80		27,27		20,80		15,40		
14%	75,27	49,73		26,93		20,00		14,27		
15%	75,73	52,40		29,60		21,33		16,93		
16%	73,67	42,27		33,60		22,67		15,20		
17%	74,40	46,80		35,27		24,27		17,60		
18%	78,27	59,07		40,07		29,27		17,27		
19%	79,27	68,00		41,00		29,53		17,87		
20%	80,07	74,87		42,07		31,47		17,27		
21%	80,87	77,67		44,53		31,60		17,20		
22%	81,47	78,67		46,60		33,53		19,53		
23%	80,87	76,20		46,07		33,73		20,07		
24%	81,33	72,73		46,73		35,20		21,00		
25%	81,20	72,07		48,53		35,33		21,93		

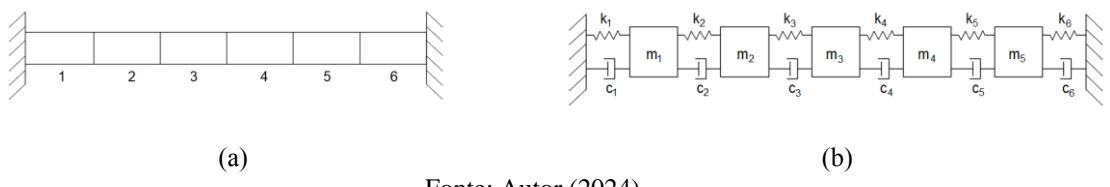
Fonte: Autor (2024).

4.3.3 Barra Biengastada Discretizada em Seis Elementos Finitos

O sistema foi discretizado em seis elementos finitos (Figura 43(a)) e idealizado como modelo de 5-DOF (Figura 43(b)). As três primeiras frequências naturais obtidas foram {2616; 5410; 8555} Hz e as três primeiras razões de amortecimento obtidas foram {2,0; 1,4; 1,5} %, com $\alpha = 562,5844$ e $\beta = 3,5104 \times 10^{-7}$, estabelecidas para o modo fundamental e para o quinto modo.

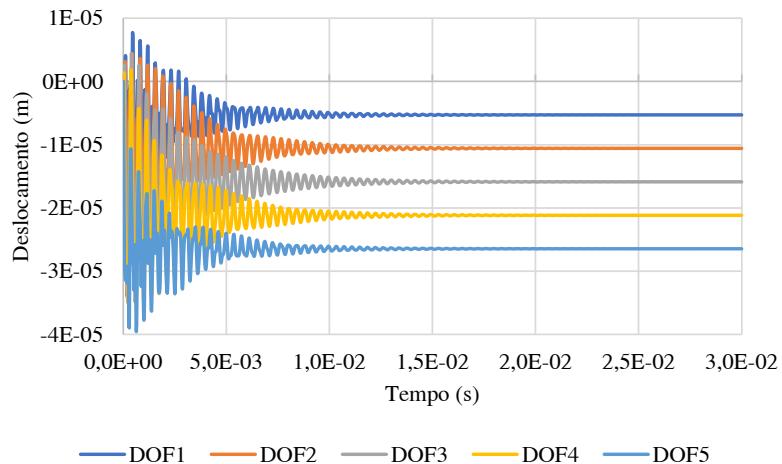
A resposta dinâmica indica que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para os DOFs que estão mais próximos do ponto de aplicação da força (Figura 44).

Figura 43 – (a) Barra biengastada discretizada em seis elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 44 – Respostas de deslocamento no tempo, barra biengastada discretizada em seis elementos finitos



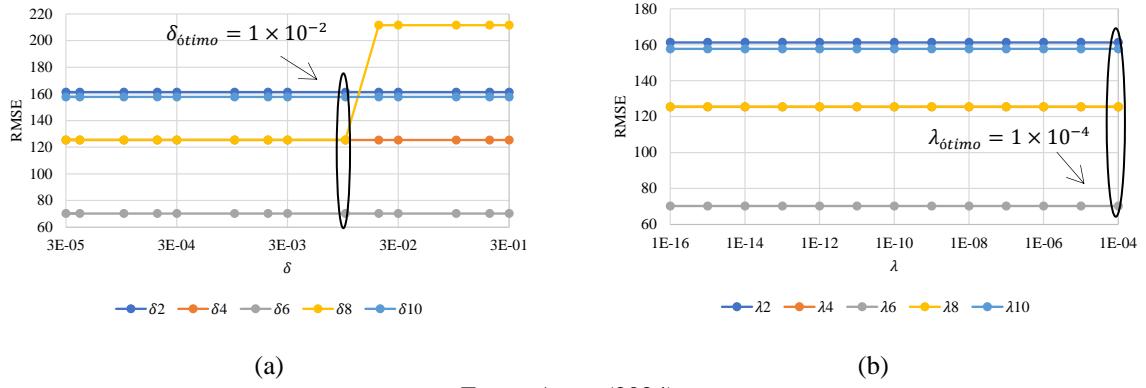
Fonte: Autor (2024).

4.3.3.1 Respostas do SINDy

O sistema foi simulado com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0, 0, 0, 0, 0]^T$, no tempo de 0 a 0,02 s, com período de amostragem de $2,6316 \times 10^{-5}$ s, que corresponde a sensores com taxa de amostragem de 38 kHz. Isso resultou em um conjunto de dados com 760

amostras. O parâmetro ótimo δ ($\delta_{óptimo} = 1 \times 10^{-2}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-4}$) foram determinados de forma que minimizasse a medida de RMSE (Figura 45). Nesses cenários, os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 66).

Figura 45 – Determinação do (a) $\delta_{óptimo}$ e (b) $\lambda_{óptimo}$, barra biengastada discretizada em seis elementos finitos.



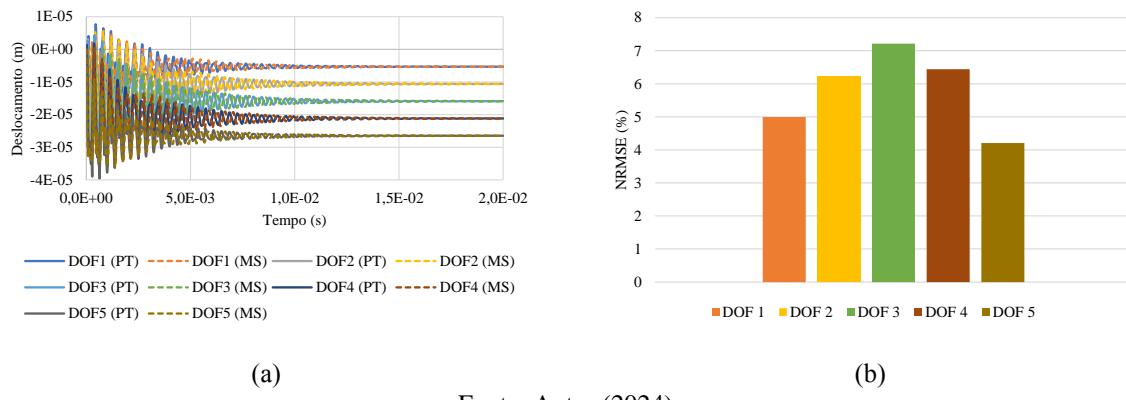
Fonte: Autor (2024).

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = 513,506 - \left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right) 2420461894,276x_1 - 1359,173x_2 + (1 - d_2)1258738094,121x_3 \\ \quad + 360,321x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = 705,514 + (1 - d_2)1386824387,400x_1 + 296,762x_2 - \left(1 - \frac{d_2}{2} - \frac{d_3}{2}\right) 2513099364,858x_3 \\ \quad - 1171,136x_4 + (1 - d_3)1257537034,695x_5 + 252,477x_6 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = 1250,169 + (1 - d_3)1276991365,542x_3 + 500,905x_4 - \left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right) 2478312704,101x_5 \\ \quad - 1519,659x_6 + (1 - d_4)1279275620,141x_7 + 485,705x_8 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = 3287,514 + (1 - d_4)1251052355,486x_5 + 995,646x_6 - \left(1 - \frac{d_4}{2} - \frac{d_5}{2}\right) 2499171261,202x_7 \\ \quad - 2547,485x_8 + (1 - d_5)1372990599,153x_9 + 1426,658x_{10} \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = -37206,103 + (1 - d_5)1251987379,441x_7 + 1041,324x_8 - \left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right) 2408011390,353x_9 \\ \quad - 2244,117x_{10}. \end{array} \right. \quad (66)$$

O modelo SINDy (Equação 66) foi integrado no tempo e comparado com a resposta de entrada (Figura 46), sendo observado uma boa correspondência das respostas, e o NRMSE para os DOFs são inferiores a 7,25% (Figura 46(b)). Também foram comparadas as respostas de deslocamento no tempo, do modelo SINDy e do *physical twin*, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 47 e 48), sendo apresentadas as respostas que apresentam menor e maior erro (NRMSE). Quando o dano é de

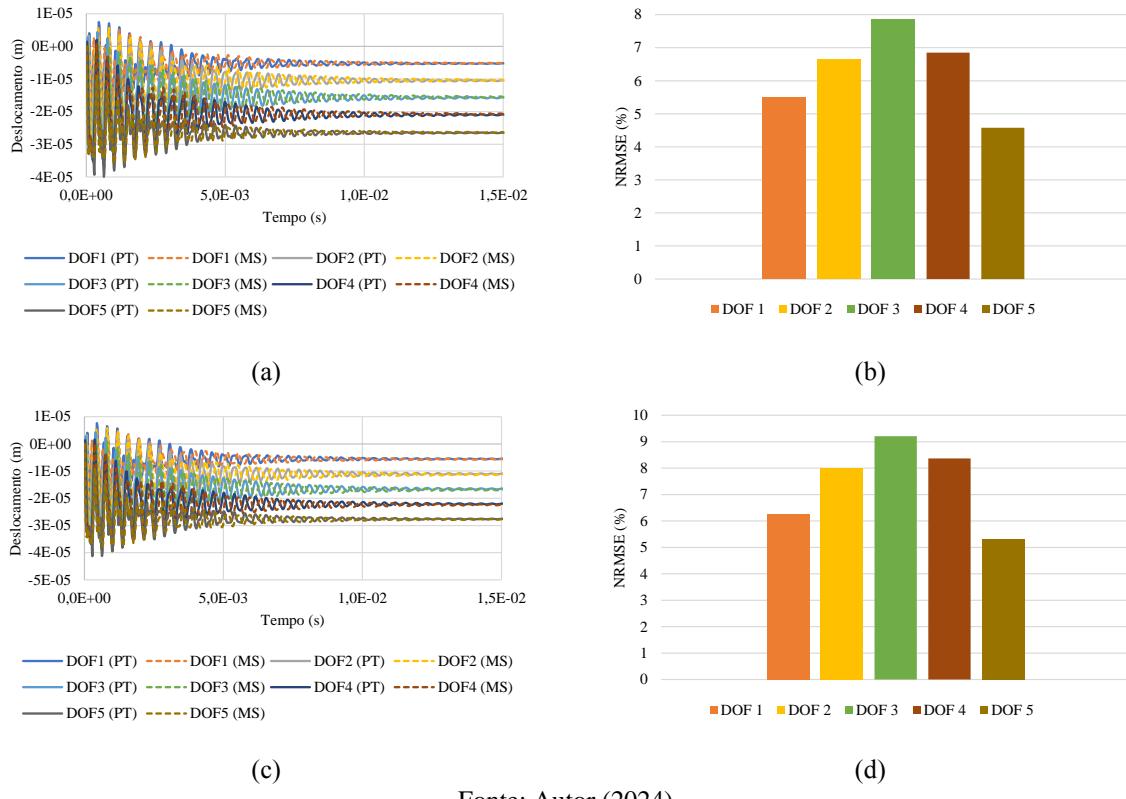
5%, constata-se uma resposta bem representativa, com menor erro na mola 5 (NRMSE inferior a 8%) (Figura 47(a-b)) e maior erro na mola 6 (NRMSE inferior a 9,5%) (Figura 47(c-d)). Para o dano de 20%, quando o dano é introduzido na mola 2 verifica-se uma concordância bem representativa, com menor erro (NRMS inferior a 6,5%) (Figura 48(a-b)), e quando o dano está na mola 6 é observada uma correspondência satisfatória, com maior erro (NRMSE entre 7% a 12,5%) (Figura 48(c-d)).

Figura 46 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



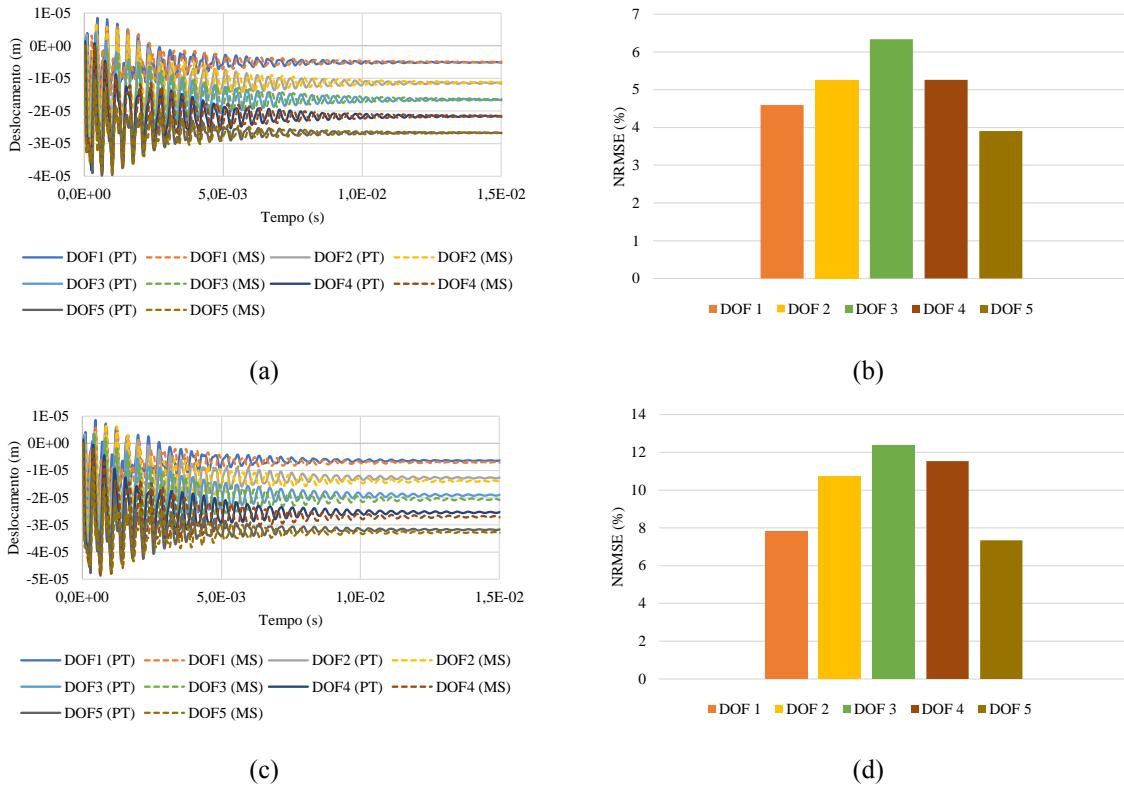
Fonte: Autor (2024).

Figura 47 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) quinta mola, (b) com correspondentes NRMSE, e (c) sexta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 48 – Comparação das respostas de deslocamento no tempo entre o physical twin (PT), barra biengastada discretizada em seis elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) segunda mola, (b) com correspondentes NRMSE, e (c) sexta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

4.3.3.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 66) foi integrado no tempo de 0 a 0,015 s, usando um período de amostragem de 5×10^{-5} s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada i -ésima mola ($i = 1, \dots, 6$). Isso resultou em um conjunto de dados de treinamento de 7500 amostras (300×25), sendo o conjunto de dados final uma matriz de dimensão $((300 \times 25) \times 5)$,

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, aplicada usando o conjunto de dados são mostrados na Tabela 16. A Tabela 17 apresenta as métricas de desempenho para os classificadores. A precisão indica que o método SVM apresentou melhor desempenho, seguido dos métodos LDA, k-NN e baseados em árvores, mais uma vez, os métodos QDA e NB apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 16 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.

Classificador	Parâmetro	Valor
QDA	-	-
LDA	-	-
NB	Modelo	Gaussiano
k-NN	Número de vizinhos	3
	Peso	Distância
	Métrica	Euclidiana
SVM	C	1000
	Kernel	RBF
	Gama (kernel não linear)	0,1
	Função de decisão	Um contra um (OVO)
RF	Número de estimadores	500
	Profundidade máxima da árvore	14
	Critério	Gini
ET	Número de estimadores	300
	Profundidade máxima da árvore	20
	Critério	Gini
DTs	Número de estimadores	14
	Critério	Entropia

Fonte: Autor (2024).

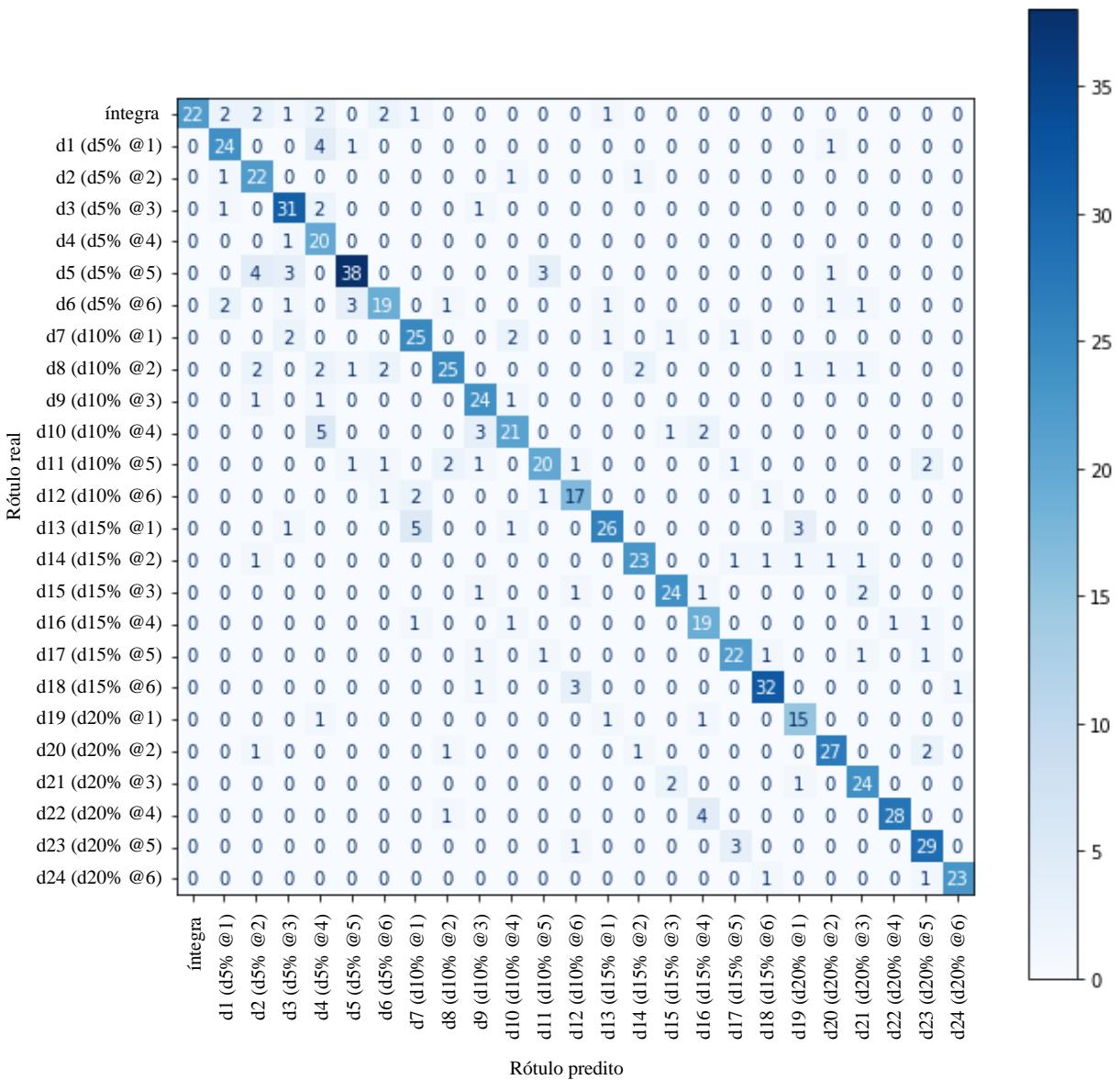
Tabela 17 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.

Classificador	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	80,00	80,33	80,57	0,80
LDA	69,11	69,70	69,11	0,69
k-NN	56,80	57,65	57,66	0,57
ET	54,93	55,49	55,98	0,55
RF	54,27	55,15	55,22	0,54
DTs	51,07	53,38	51,96	0,52
QDA	27,11	27,78	27,11	0,22
NB	7,83	9,78	7,83	0,05

Fonte: Autor (2024).

A matriz de confusão do método SVM mostra que a probabilidade de considerar um dano quando o sistema está íntegro é 33,3% (11/33) (Figura 49).

Figura 49 – Matriz de confusão do classificador SVM, conjunto de dados do modelo de barra biengastada discretizada em seis elementos finitos.



Fonte: Autor (2024).

4.3.3.3 Aplicações do Framework Digital Twin

O *physical twin* foi simulado no tempo de 0 a 0,02 s, usando um período de amostragem de $1,3333 \times 10^{-5}$ s, que correspondeu a 1500 amostras para cada um dos cenários de interesse, condição íntegra e de 1% até 25% de dano em cada elemento. Um ruído gaussiano aleatório foi adicionado aos dados (1500 amostras), com média zero e diferentes níveis de ruído, a partir de um desvio padrão de $2,30 \times 10^{-6}$ m, que compreende o desvio padrão do fenômeno em análise.

A aplicação do *framework* DT apresenta os seguintes resultados para a primeira mola (Tabela 18) nas condições sem ruído e 1% de ruído: de 0% a 2% de dano, a estrutura está intacta; de 3% a 7% de dano, a estrutura está com dano de 5%; de 8% a 12% de dano, a estrutura está com dano de 10%; de 13% a 16% de dano, a estrutura está com dano de 15% e, por fim, de 17% a 25% de dano a estrutura está com dano de 20%. Para 3% e 5% de ruído, maiores probabilidades de identificação de dano são apresentadas para dano de 20%. Quando utilizado 10% de ruído, constata-se uma baixa probabilidade, não identificando o elemento com dano. As respostas para as demais molas estão no Apêndice.

Tabela 18 – Respostas do *framework* DT do classificador SVM para a primeira mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA	P (%)	DA	P (%)	DA	P (%)
0%	81,13	70,40		22,27		14,60		8,20
1%	80,20	63,53		21,53		14,33		8,53
2%	75,67	48,13	Íntegra	18,93		14,27		9,53
3%	74,20	47,20		16,93		15,07		9,47
4%	77,87	58,60		16,53		14,07		8,87
5%	78,40	63,07		16,13		16,13		9,20
6%	76,80	55,33		15,93		14,87		8,93
7%	68,73	40,53		14,73		15,20		9,73
8%	74,27	51,80		19,13		14,27		12,27
9%	79,00	62,87		19,33		13,67		11,27
10%	78,80	64,53		20,93		14,40		11,87
11%	77,40	55,87		20,13		16,13		12,67
12%	51,20	41,07		19,87		15,60		14,13
13%	77,87	59,13		24,27		16,80		12,33
14%	80,47	66,60		24,73		20,00		15,40
15%	80,13	66,40		25,27		20,67		17,87
16%	78,00	54,87		30,87		24,60		18,87
17%	72,07	47,27		37,53		28,33		19,53
18%	80,27	66,87		45,00		30,27		20,40
19%	82,67	77,47		49,27		35,47		19,20
20%	83,40	82,00		53,40		38,33		21,13
21%	84,20	84,33		60,00		41,00		23,87
22%	84,60	84,07		61,33		44,67		27,00
23%	85,00	84,80		66,27		45,00		27,40
24%	85,07	84,93		70,00		51,20		25,53
25%	85,80	85,87		72,00		52,20		29,27

Fonte: Autor (2024).

5 EXEMPLOS DE APLICAÇÃO: VIBRAÇÃO TRANSVERSAL

Este capítulo apresenta as etapas para aplicação das técnicas supramencionadas para um modelo de viga em vibração transversal. Em seguida, as técnicas são aplicadas a um modelo de viga de aço engastada, discretizado em dois e quatro elementos finitos. As propriedades do material e informações geométricas dos modelos são as mesmas apresentadas no capítulo quatro.

5.1 Procedimento para Aplicação Prática das Técnicas

5.1.1 Physical Twin

O *physical twin* correspondeu a uma viga prismática em vibração transversal. A equação diferencial de movimento que governa a sua vibração, considerando a teoria de viga de Euler-Bernoulli, consiste em:

$$\frac{\partial^2}{\partial x^2} \left(EI \frac{\partial^2 v(x, t)}{\partial x^2} \right) + \rho A \frac{\partial^2 v(x, t)}{\partial t^2} = p(x, t), \quad (67)$$

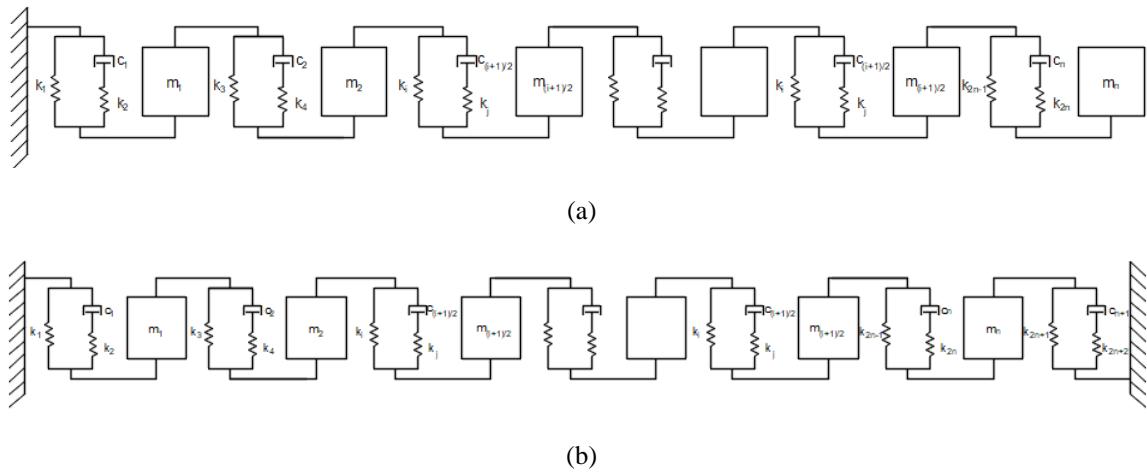
onde $v(x, t)$ é a deflexão transversal, x é a posição espacial ao longo de seu comprimento, t é o tempo, E é o módulo de elasticidade, I é o momento de inércia da seção transversal, A é a área da seção transversal, ρ é a massa específica e $p(x, t)$ é a força de excitação transversal distribuída. Essa equação diferencial parcial foi discretizada através do método dos elementos finitos, empregando-se funções de forma polinomial cúbica, sendo obtido um sistema discreto (Equação 51) (CRAIG e KURDILA, 2006).

As constantes de proporcionalidade (α e β) foram estabelecidas para o modo fundamental e para um dos modos de frequência mais alta que contribui significativamente para a resposta dinâmica (CLOUGH e PENZIEN, 2003). O sistema foi discretizado em dois e quatro elementos finitos e foram aplicadas as condições de contorno adequada para o modelo de viga engastada.

5.1.2 Modelo Baseado na Física

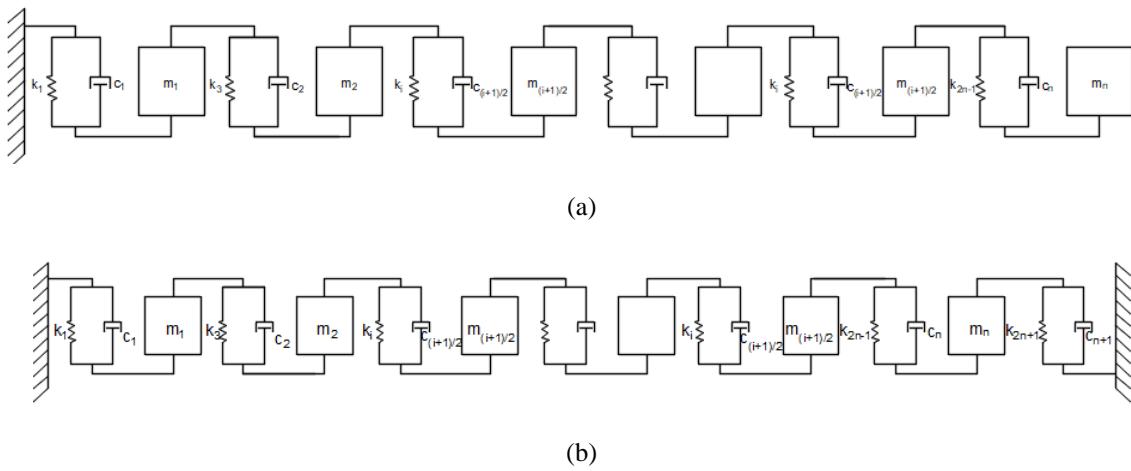
O modelo de viga (*physical twin*) foi idealizado baseado na física e construído utilizando uma descrição de parâmetros concentrados de n graus de liberdade (n -DOF), cada DOF foi representado por um modelo sólido linear padrão (*standard linear solid – SLS*), obtidos pela associação de uma mola em paralelo ao modelo de Maxwell (Figura 50). O sistema equivalente para uma representação n -DOF é apresentado na Figura 51.

Figura 50 – Idealização do modelo de viga (a) engastada e (b) biengastada, utilizando descrição de parâmetros concentrados de n -DOF representado por um modelo SLS.



Fonte: Autor (2024)

Figura 51 – Idealização do sistema equivalente n -DOF do modelo de viga (a) engastada e (b) biengastada.



Fonte: Autor (2024).

As equações de movimento do sistema equivalente n -DOF (Figura 51) (Equação 52) foram convertidas em um sistema de equações diferenciais de primeira ordem (Equação 68), para o modelo de viga engastada, não haverá as parcelas correspondentes a k_{n+1} , k_{2n+1} e k_{2n+2} :

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(-\frac{k_1 + k_3}{m_1} \right) x_1 + \left(-\frac{1}{m_1} \left(\frac{c_1(k_1 + k_2)}{k_2} \right) \right) x_2 + \left(-\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) x_2 \\ \quad + \left(-\frac{k_3}{m_1} \right) x_3 + \left(\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) x_4 \\ \dot{x}_{2i-1} = x_{2i} \\ \dot{x}_{2i} = \left(\frac{f_i(t)}{m_i} \right) + \left(\frac{k_{2i-1}}{m_i} \right) x_{2i-3} + \left(\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) x_{2i-2} + \left(-\frac{k_{2i-1} + k_{2i+1}}{m_i} \right) x_{2i-1} \\ \quad + \left(-\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) x_{2i} + \left(-\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) x_{2i} + \left(\frac{k_{2i+1}}{m_i} \right) x_{2i+1} \\ \quad + \left(\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) x_{2i+2} \\ \quad \text{para } i = 2, 3, \dots, n-1 \\ \dot{x}_{2n-1} = x_{2n} \\ \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left(\frac{k_{2n-1} + k_{2n+1}}{m_n} \right) x_{2n-3} + \left(\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) x_{2n-2} + \left(-\frac{k_{2n-1}}{m_n} \right) x_{2n-1} \\ \quad + \left(-\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) x_{2n} + \left(-\frac{1}{m_n} \left(\frac{c_{n+1}(k_{2n+1} + k_{2n+2})}{k_{2n+2}} \right) \right) x_{2n}. \end{array} \right. \quad (68)$$

5.1.3 Respostas do SINDy

Os dados de treinamento para o SINDy foram gerados com a aplicação de uma força vertical com magnitude de 100 N na extremidade direita do modelo de viga engastada (*physical twin*), escolhida de forma que mobilizasse a estrutura e atendesse ao seu critério de estabilidade estática. As respectivas respostas de deslocamento no tempo foram então registradas.

Foi considerada uma taxa de amostragem de aproximadamente duas vezes e meia a maior frequência do modelo de alta fidelidade.

Os dados de velocidade ($\dot{x}(t)$) e aceleração ($\ddot{x}(t)$) foram determinados através da diferenciação numérica dos dados de deslocamento ($x(t)$), via método das diferenças finitas centrais de segunda ordem.

Foram comparados os algoritmos STLSQ e LASSO, utilizando o pacote PySINDy, sendo os modelos LASSO ajustados usando o pacote *scikit-learn*. O parâmetro δ para o algoritmo STLSQ e o hiperparâmetro λ para o algoritmo LASSO foram variados considerando um determinado intervalo, de modo que aumentar δ ou λ produziam modelos com poucos ou

nenhum termo e diminuí-los produziam modelos convergentes, e escolhidos de forma que minimizasse a medida de erro (RMSE) entre os sinais de entrada de aceleração e a predição das equações descobertas para aceleração.

A matriz da biblioteca SINDy $\Phi(X)$ foi construída usando termos polinomiais de primeira ordem. O SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física (Equação 68).

5.1.4 Parametrização de Dano

A parametrização de dano para os termos das equações que contêm uma única mola correspondeu à Equação (54), enquanto para os termos em que contêm duas molas correspondeu à Equação (57) e os termos correspondentes a equivalência do amortecimento foram parametrizados usando as Equações (54) e (57), resultando em:

$$\left(-\frac{1}{m_i} \left(\frac{c_i(k_i + k_j)}{k_j} \right) \right) = \frac{\left(1 - \frac{d_i}{2} - \frac{d_j}{2} \right)}{(1 - d_i)} \left(-\frac{1}{m_i} \left(\frac{c_i(k_i + k_j)}{k_j} \right) \right). \quad (69)$$

As melhores equações de movimento descobertas pelo SINDy foram reescritas para parametrização de dano, modelo de barra engastada (Equação 70) e modelo de barra biengastada (Equação 71):

$$\left\{
\begin{aligned}
& \dot{x}_1 = x_2 \\
& \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(\left(1 - \frac{d_1}{2} - \frac{d_3}{2} \right) \left(-\frac{k_1 + k_3}{m_1} \right) x_1 + \left(\frac{\left(1 - \frac{d_1}{2} - \frac{d_2}{2} \right)}{(1 - d_2)} \left(-\frac{1}{m_1} \left(\frac{c_1(k_1 + k_2)}{k_2} \right) \right) \right) x_2 \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2} \right)}{(1 - d_4)} \left(-\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) \right) x_2 + \left((1 - d_3) \left(\frac{k_3}{m_i} \right) \right) x_3 \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2} \right)}{(1 - d_4)} \left(\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) \right) x_4 \right) \\
& \dot{x}_{2i-1} = x_{2i} \\
& \dot{x}_{2i} = \left(\frac{f_i(t)}{m_i} \right) + \left((1 - d_{2i-1}) \left(\frac{k_{2i-1}}{m_i} \right) \right) x_{2i-3} + \left(\frac{\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i}}{2} \right)}{(1 - d_{2i})} \left(\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) \right) x_{2i-2} \\
& \quad + \left(\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i+1}}{2} \right) \left(-\frac{k_{2i-1} + k_{2i+1}}{m_i} \right) \right) x_{2i-1} + \left(\frac{\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i}}{2} \right)}{(1 - d_{2i})} \left(-\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) \right) x_{2i} \\
& \quad + \left(\frac{\left(1 - \frac{d_{2i+1}}{2} - \frac{d_{2i+2}}{2} \right)}{(1 - d_{2i+2})} \left(-\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) \right) x_{2i} + \left((1 - d_{2i+1}) \left(\frac{k_{2i+1}}{m_i} \right) \right) x_{2i+1} \\
& \quad + \left(\frac{\left(1 - \frac{d_{2i+1}}{2} - \frac{d_{2i+2}}{2} \right)}{(1 - d_{2i+2})} \left(\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) \right) x_{2i+2} \\
& \text{para } i = 2, 3, \dots, n-1 \\
& \dot{x}_{2n-1} = x_{2n} \\
& \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left((1 - d_{2n-1}) \left(\frac{k_{2n-1}}{m_n} \right) \right) x_{2n-3} \\
& \quad + \left(\frac{\left(1 - \frac{d_{2n-1}}{2} - \frac{d_{2n}}{2} \right)}{(1 - d_{2n})} \left(\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) \right) x_{2n-2} + \left((1 - d_{2n-1}) \left(-\frac{k_{2n-1}}{m_n} \right) \right) x_{2n-1} \\
& \quad + \left(\frac{\left(1 - \frac{d_{2n-1}}{2} - \frac{d_{2n}}{2} \right)}{(1 - d_{2n})} \left(-\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) \right) x_{2n}
\end{aligned} \right. \tag{70}$$

$$\left\{
\begin{aligned}
& \dot{x}_1 = x_2 \\
& \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(\left(1 - \frac{d_1}{2} - \frac{d_3}{2} \right) \left(-\frac{k_1 + k_3}{m_1} \right) x_1 + \left(\frac{\left(1 - \frac{d_1}{2} - \frac{d_2}{2} \right)}{(1 - d_2)} \left(-\frac{1}{m_1} \left(\frac{c_1(k_1 + k_2)}{k_2} \right) \right) \right) x_2 \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2} \right)}{(1 - d_4)} \left(-\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) \right) x_2 + \left((1 - d_3) \left(\frac{k_3}{m_i} \right) \right) x_3 \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2} \right)}{(1 - d_4)} \left(\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) \right) x_4 \right. \\
& \quad \left. \dot{x}_{2i-1} = x_{2i} \right. \\
& \quad \left. \dot{x}_{2i} = \left(\frac{f_i(t)}{m_i} \right) + \left((1 - d_{2i-1}) \left(\frac{k_{2i-1}}{m_i} \right) \right) x_{2i-3} + \left(\frac{\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i}}{2} \right)}{(1 - d_{2i})} \left(\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) \right) x_{2i-2} \right. \\
& \quad \left. + \left(\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i+1}}{2} \right) \left(-\frac{k_{2i-1} + k_{2i+1}}{m_i} \right) \right) x_{2i-1} + \left(\frac{\left(1 - \frac{d_{2i-1}}{2} - \frac{d_{2i}}{2} \right)}{(1 - d_{2i})} \left(-\frac{1}{m_i} \left(\frac{c_i(k_{2i-1} + k_{2i})}{k_{2i}} \right) \right) \right) x_{2i} \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_{2i+1}}{2} - \frac{d_{2i+2}}{2} \right)}{(1 - d_{2i+2})} \left(-\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) \right) x_{2i} + \left((1 - d_{2i+1}) \left(\frac{k_{2i+1}}{m_i} \right) \right) x_{2i+1} \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_{2i+1}}{2} - \frac{d_{2i+2}}{2} \right)}{(1 - d_{2i+2})} \left(\frac{1}{m_i} \left(\frac{c_{i+1}(k_{2i+1} + k_{2i+2})}{k_{2i+2}} \right) \right) \right) x_{2i+2} \right. \\
& \quad \left. \text{para } i = 2, 3, \dots, n-1 \right. \\
& \quad \left. \dot{x}_{2n-1} = x_{2n} \right. \\
& \quad \left. \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left(\left(1 - \frac{d_{2n-1}}{2} - \frac{d_{2n+1}}{2} \right) \left(\frac{k_{2n-1} + k_{2n+1}}{m_n} \right) \right) x_{2n-3} \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_{2n-1}}{2} - \frac{d_{2n}}{2} \right)}{(1 - d_{2n})} \left(\frac{1}{m_n} \left(\frac{c_i(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) \right) x_{2n-2} + \left((1 - d_{2n-1}) \left(-\frac{k_{2n-1}}{m_n} \right) \right) x_{2n-1} \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_{2n-1}}{2} - \frac{d_{2n}}{2} \right)}{(1 - d_{2n})} \left(-\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) \right) x_{2n} \right. \\
& \quad \left. + \left(\frac{\left(1 - \frac{d_{2n+1}}{2} - \frac{d_{2n+2}}{2} \right)}{(1 - d_{2n+2})} \left(-\frac{1}{m_n} \left(\frac{c_{n+1}(k_{2n+1} + k_{2n+2})}{k_{2n+2}} \right) \right) \right) x_{2n} \right).
\end{aligned} \tag{71}
\right.$$

5.1.5 Construção do Conjunto de Dados

O conjunto de dados final X_{dados} é uma matriz de dimensão $((M \times r) \times N)$, formada das M amostras de deslocamentos de cada r cenários de dano, incluindo a estrutura íntegra (classes), com N deslocamentos medidos (atributos):

$$X_{dadosfinal} = \begin{bmatrix} d_1 X_{dados} \\ d_2 X_{dados} \\ \dots \\ d_r X_{dados} \end{bmatrix}.$$

5.1.6 Construção dos Classificadores de *Machine Learning*

O conjunto de dados, usado para treinar os classificadores de *machine learning*, foi construído simulando a resposta de deslocamento no tempo do sistema (equações do modelo SINDy) com diferentes locais de danos e diferentes intensidades correspondentes. Dessa forma, 300 amostras foram obtidas para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%), distribuído para a i -ésima mola ($i = 1, \dots, 2n - 1$) e j -ésima mola ($j = 2, \dots, 2n$), que representam a rigidez elementar. Uma validação cruzada estratificada de 5 *folds* foi empregada. Para melhorar as condições de treinamento, os dados foram normalizados subtraindo a média e dividindo pelo desvio padrão de todas as amostras.

Um conjunto de classificadores de *machine learning*, desenvolvido usando o pacote *scikit-learn* da linguagem Python 3.9, foi selecionado para análises, considerando seu uso na literatura SHM e sua interpretabilidade. O procedimento de busca em grade, utilizando o método de validação cruzada k -fold, foi usado para identificar os hiperparâmetros ótimos de cada modelo de classificação, com base em intervalos recomendados na literatura de SHM e *machine learning* e documentação do *scikit-learn*. Os modelos para cada algoritmo de classificação e os respectivos hiperparâmetros são correspondentes os mesmos apresentados na Tabela 1. Para os métodos que possuem hiperparâmetros, 90% dos dados foram usados para treinamento e 10% para teste, e o modelo final foi ajustado com todos os dados.

5.2 Viga Engastada

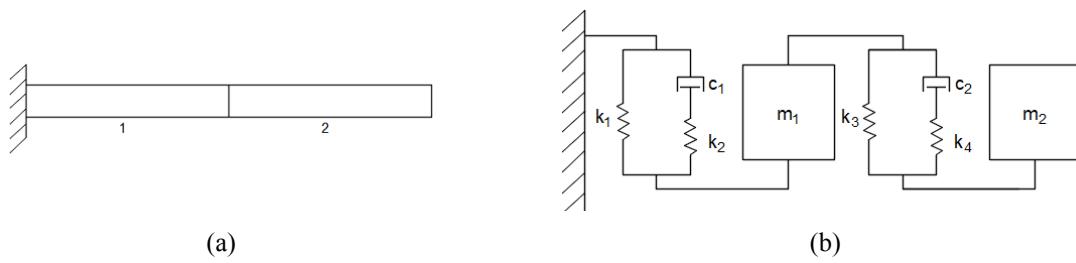
5.2.1 Viga Engastada Discretizada em Dois Elementos Finitos

O sistema foi discretizado em dois elementos finitos (Figura 52(a)), aplicando as condições de contorno fixa-livre (Equação 51), e o modelo de viga foi idealizado como um modelo 2-DOF (Figura 52(b)).

As três primeiras frequências naturais, obtidas a partir das matrizes de massa e rigidez do sistema (problema de autovalor generalizado), foram $\{20,898; 132,013; 446,493\}$ Hz e as três primeiras razões de amortecimento obtidas foram $\{2,0; 0,5; 0,8\}$ %, com $\alpha = 5,1689$ e $\beta = 4,8346 \times 10^{-6}$, estabelecidas para o modo fundamental e para o quarto modo.

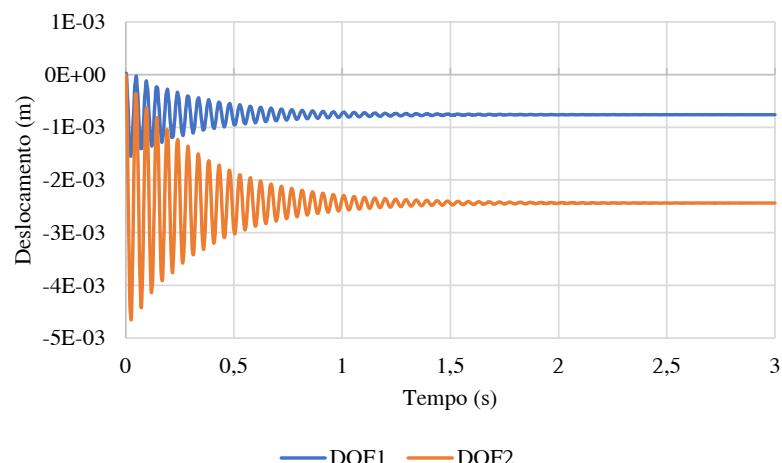
A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para o DOF que está mais próximos do ponto de aplicação da força (Figura 53).

Figura 52 – (a) Viga engastada discretizada em dois elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 53 – Respostas de deslocamento no tempo, viga engastada discretizada em dois elementos finitos



Fonte: Autor (2024).

5.2.1.1 Respostas do SINDy

Os dados de treinamento para o SINDy foram gerados simulando o sistema (Equação 51) com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0, 0]^T$, no tempo de 0 a 2,5 s, com período de amostragem de $2,8571 \times 10^{-4}$ s, que corresponde a sensores com taxa de amostragem de 3,5 kHz. Isso resultou em um conjunto de dados com 8750 amostras.

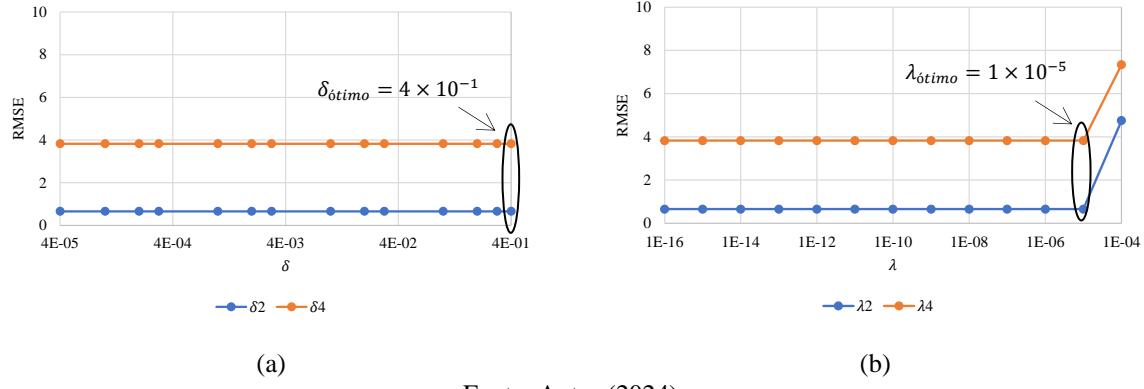
As equações de movimento do modelo 2-DOF foram convertidas em sistemas de equações diferenciais de primeira ordem, para uso do SINDy:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left(\frac{f_1(t)}{m_1} \right) + \left(-\frac{k_1 + k_3}{m_1} \right) x_1 + \left(-\frac{1}{m_1} \left(\frac{c_1(k_1 + k_2)}{k_2} \right) \right) x_2 + \left(-\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) x_2 \\ \quad + \left(\frac{k_3}{m_1} \right) x_3 + \left(\frac{1}{m_1} \left(\frac{c_2(k_3 + k_4)}{k_4} \right) \right) x_4 \\ \dot{x}_{2n-1} = x_{2n} \\ \dot{x}_{2n} = \left(\frac{f_n(t)}{m_n} \right) + \left(\frac{k_{2n-1}}{m_n} \right) x_{2n-3} + \left(\frac{1}{m_n} \left(\frac{c_i(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) x_{2n-2} + \left(-\frac{k_{2n-1}}{m_n} \right) x_{2n-1} \\ \quad + \left(-\frac{1}{m_n} \left(\frac{c_n(k_{2n-1} + k_{2n})}{k_{2n}} \right) \right) x_{2n}. \end{cases} \quad (72)$$

O SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física (Equação 72).

O parâmetro δ (algoritmo STLSQ) e hiperparâmetro λ (algoritmo LASSO) foram variados à medida que os valores de RMSE foram computados. Então, o parâmetro ótimo δ ($\delta_{óptimo} = 4 \times 10^{-1}$) e o hiperparâmetro ótimo λ ($\lambda_{óptimo} = 1 \times 10^{-5}$) foram escolhidos de forma que minimizasse a medida de erro (Figura 54). Os dois algoritmos descobriram as mesmas equações de movimento, que foram reescritas para parametrização de dano (Equação 73). O tempo de execução do algoritmo STLSQ foi de 4,1942 s, enquanto para o do algoritmo LASSO foi 20 minutos e 1534 segundos.

Figura 54 – Determinação do (a) $\delta_{\text{óptimo}}$ e (b) $\lambda_{\text{óptimo}}$, viga engastada discretizada em dois elementos finitos.

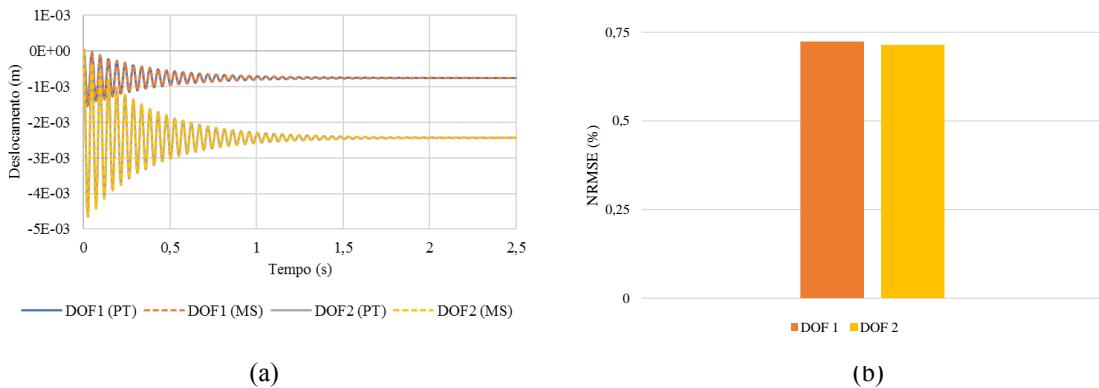


Fonte: Autor (2024).

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = 16,855 - \left(1 - \frac{d_1}{2} - \frac{d_3}{2}\right) 471838,773 x_1 - \left(\frac{\left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right)}{(1 - d_2)}\right) 3,636 x_2 \\ \quad - \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 3,636 x_2 + (1 - d_3) 154362,151 x_3 + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 0,626 x_4 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -83,158 + ((1 - d_3)) 627583,891 x_1 + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 3,941 x_2 \\ \quad - (1 - d_3) 230230,665 x_3 - \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 7,014 x_4. \end{array} \right. \quad (73)$$

Comparando as respostas de deslocamento no tempo do modelo SINDy (Equação 73) e a resposta de entrada (*physical twin*) (Figura 55(a)), constata-se uma ótima concordância das respostas, e o NRMSE para os DOFs 1 e 2 são inferiores a 0,60% (Figura 55(b)).

Figura 55 – (a) Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) e (b) correspondentes NRMSE.



Fonte: Autor (2024).

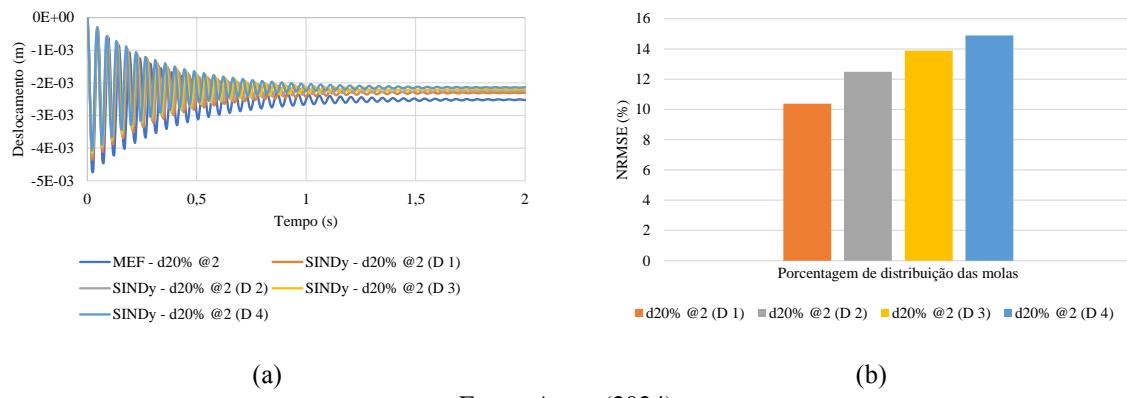
5.2.1.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 73) foi integrado no tempo de 0 a 2 s, usando um período de amostragem de $6,6667 \times 10^{-4}$ s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%). Cada intensidade de dano foi distribuída para cada i -ésima mola ($i = 1, 3$) e j -ésima mola ($i = 2, 4$) (Figura 52). Foi considerada uma porcentagem de distribuição (D), para cada intensidade de dano, de 10% e 90% (D 1), 15% e 85% (D 2), 20% e 80% (D 3), 25% e 75% (D 4), em cada i -ésima mola e j -ésima mola, respectivamente. Isso resultou em quatro conjuntos de dados (CD) de treinamento diferentes, cada um com 2700 amostras (300×9), sendo cada conjunto de dados final uma matriz de dimensão $((300 \times 9) \times 2)$. Para as quatro porcentagens de distribuição das molas foram calculados o NRMSE em relação as respostas do MEF, considerando um dano com intensidade de 20% em cada mola, uma vez que é esperado que ocorra o maior erro entre as respostas. Quando comparado as respostas, constata-se que o maior erro ocorre para o DOF2, quando o dano é introduzido na mola 2, e o erro aumento à medida que a intensidade de dano na i -ésima aumenta (Figura 56).

Foi escolhido o método SVM para avaliação do desempenho dos quatro conjuntos de dados, sendo comparados com respostas de um conjunto de dados gerado do modelo MEF. O método SVM foi escolhido devido ter apresentado o melhor desempenho para identificação de dano nas investigações anteriores (Capítulo 4). Para cada conjunto de dados, foi realizada uma busca de grade com base nos hiperparâmetros da Tabela 1. Dada a constatação de desempenho insatisfatório ao utilizar apenas deslocamento como atributo, foram investigados 7 diferentes

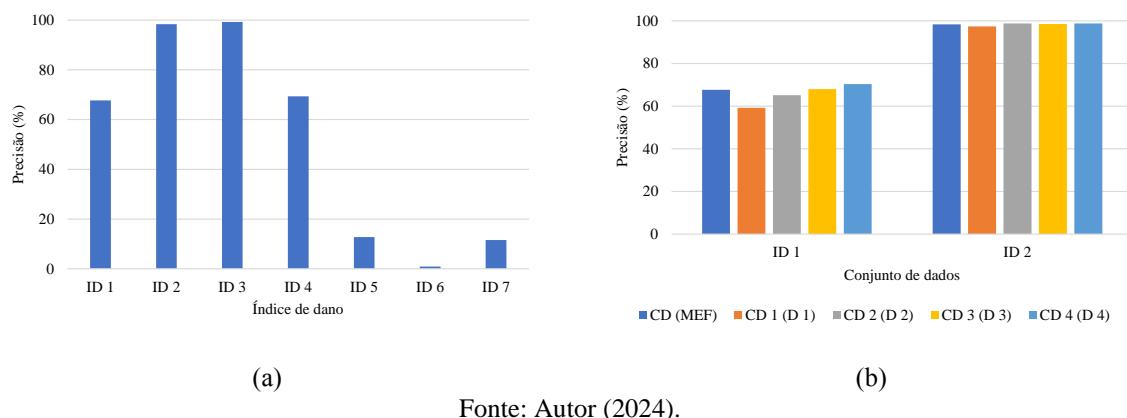
índices de dano (ID), atributos de deslocamento (ID 1), deslocamento e aceleração (ID 2), deslocamento, velocidade e aceleração (ID 3), deslocamento e velocidade (ID 4), velocidade e aceleração (ID 5), velocidade (ID 6) e aceleração (ID 7), usando conjunto de dados do MEF (Figura 57(a)). Desses, foi escolhido o índice de dano de deslocamento e aceleração (ID 2), visto que apresentou melhor desempenho na comparação com os demais, considerando-se a precisão e o número de atributos dos cenários investigados. Dessa forma, foram comparados os índices de dano com atributo de deslocamento (ID 1) e com atributos de deslocamento e aceleração (ID 2) com dados do MEF (Figura 57(b)). Foi constatado que, à medida que o erro aumenta (Figura 56(b)), a precisão do modelo de ML também aumenta (Figura 57(b)) e, para o conjunto de dados de 25% e 75%, apresenta um desempenho superior ao do modelo MEF. Em seguida, foi escolhida a porcentagem de distribuição 20% e 80%, adotada considerando o equilíbrio entre o erro, discrepância entre as respostas MEF e SINDy, e o desempenho do modelo.

Figura 56 – (a) Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para o DOF 2 e (b) correspondentes NRMSE.



Fonte: Autor (2024).

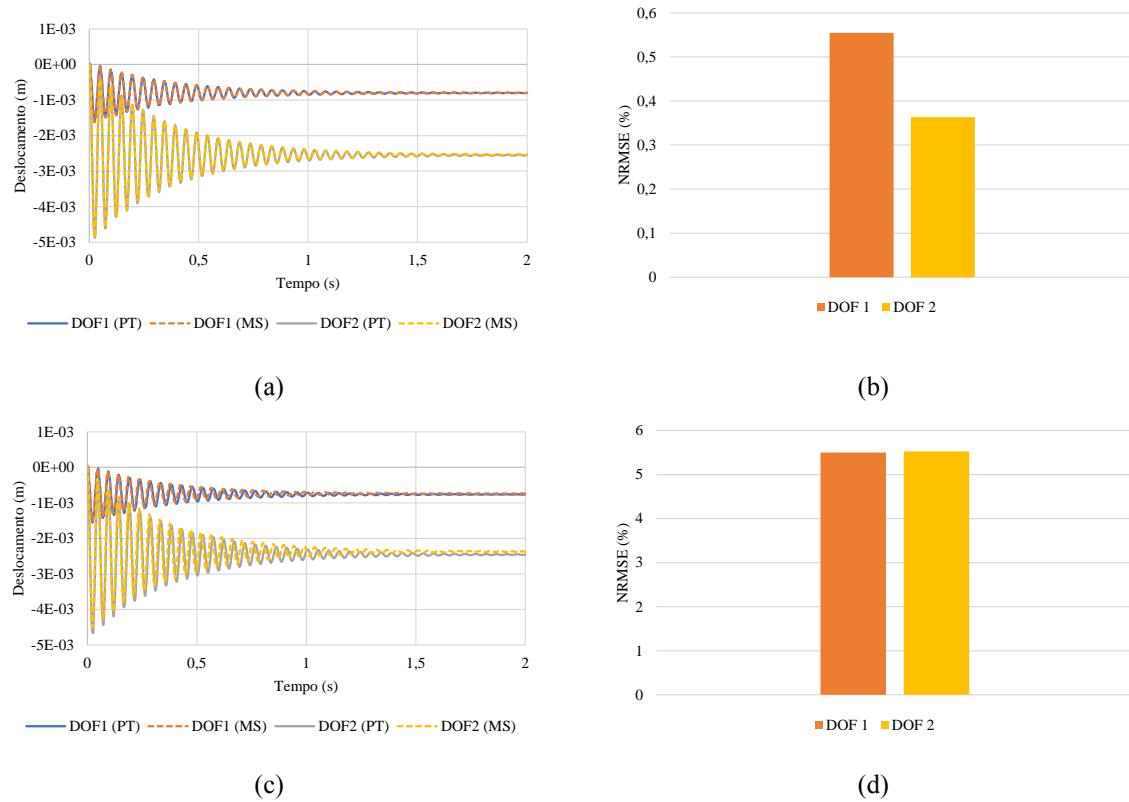
Figura 57 – (a) Índices de dano para o conjunto de dados do MEF e (b) índices de dano para os conjuntos de dados.



Fonte: Autor (2024).

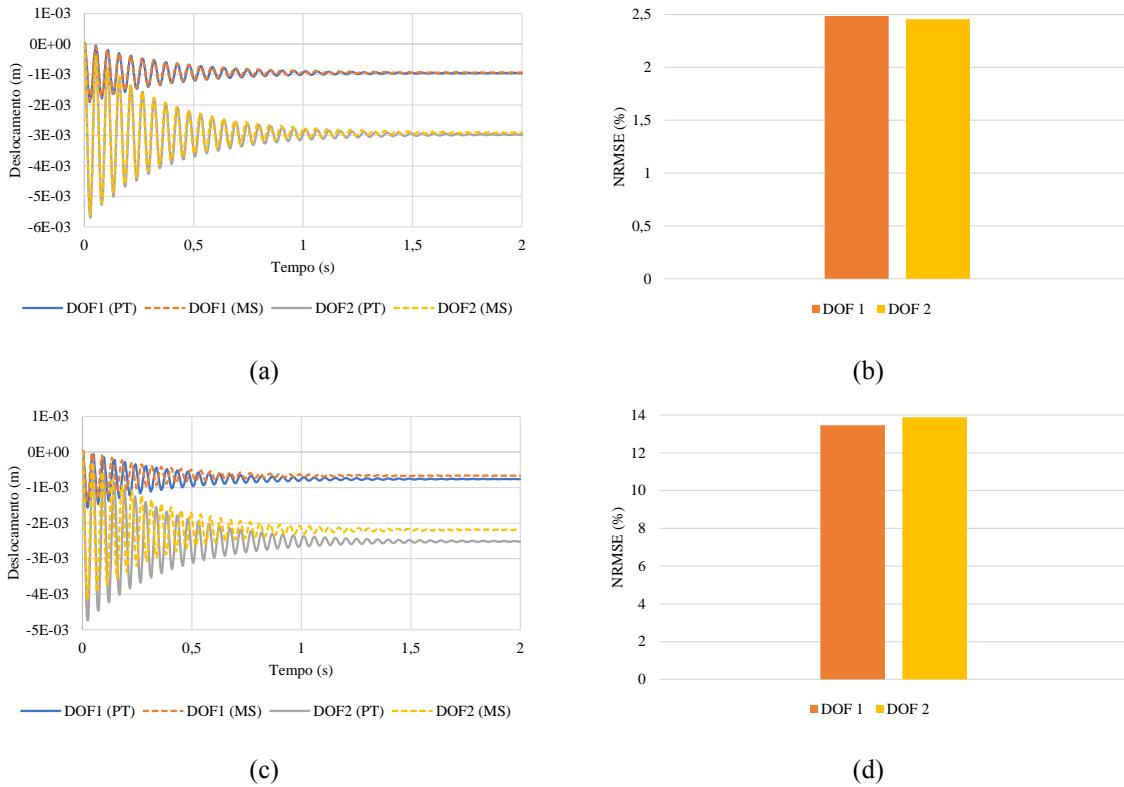
Foram comparadas as respostas de deslocamento no tempo, do *physical twin* e do modelo SINDy, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 58 e 59). Para a intensidade de dano de 5%, quando o dano é introduzido na mola 1 é apresentada uma ótima correspondência, com NRMSE inferior a 0,6% (Figura 58(a-b)) e quando o dano está na mola 2 é observada uma boa correspondência, com NRMSE inferior a 5,75% (Figura 58(c-d)). Quando o dano de 20% está na mola 1 verifica-se uma ótima correspondência (NRMSE inferior a 2,5%) (Figura 59(a-b)) e quando o dano é introduzido na mola 2 constata-se uma concordância satisfatória (com NRMSE inferior a 14%) (Figura 59(c-d)).

Figura 58 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 59 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em dois elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) primeira mola, (b) com correspondentes NRMSE, e (c) segunda mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, usando o conjunto de dados com porcentagem de distribuição 20% e 80% (CD 3) para os dois índices de dano, com atributos de deslocamento (ID 1) e deslocamento e aceleração (ID 2) são mostrados na Tabela 19. A Tabela 20 apresenta as métricas de desempenho dos classificadores. Considerando o índice de dano 1, a precisão indica que os métodos SVM e LDA apresentaram os melhores desempenhos, seguido dos métodos k-NN, QDA e os baseados em árvores, e o método NB apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão. A análise por meio do índice de dano 2 promoveu o aumento significativo de desempenho, com valores de precisão indicando que os métodos QDA e SVM apresentaram os melhores resultados, seguido dos métodos LDA, k-NN e os baseados em árvores, e o método NB novamente apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 19 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.

Classificador	Parâmetro	Valor	
		ID 1	ID 2
QDA	-	-	-
LDA	-	-	-
NB	Modelo	Gaussiano	Gaussiano
k-NN	Número de vizinhos	3	4
	Peso	Distância	Distância
	Métrica	Euclidiana	Euclidiana
SVM	C	1000	1000
	Kernel	RBF	Linear
	Gama (kernel não linear)	1	-
	Função de decisão	Um contra um (OVO)	Um contra um (OVO)
RF	Número de estimadores	500	200
	Profundidade máxima da árvore	14	14
	Critério	Gini	Entropia
ET	Número de estimadores	500	300
	Profundidade máxima da árvore	20	20
	Critério	Gini	Entropia
DTs	Profundidade máxima da árvore	14	12
	Critério	Entropia	Entropia

Fonte: Autor (2024).

Tabela 20 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.

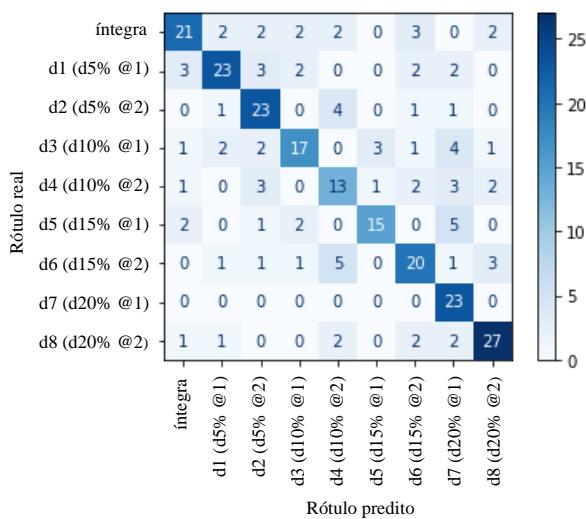
Classificador	Acurácia (%)		Precisão (%)		Revocação (%)		F-score	
	ID 1	ID 2	ID1	ID 2	ID 1	ID 2	ID 1	ID 2
SVM	67,41	98,52	68,04	98,54	67,85	98,58	0,69	0,99
LDA	66,00	88,59	64,66	88,31	66,00	88,59	0,65	0,88
k-NN	59,63	75,55	59,89	75,91	60,28	75,63	0,59	0,76
QDA	56,07	99,56	55,84	99,57	56,07	99,56	0,54	0,96
ET	54,44	73,70	54,15	73,39	55,05	73,69	0,54	0,74
DTs	52,96	71,85	53,08	71,85	52,65	72,24	0,53	71,82
RF	52,22	73,70	51,84	73,50	52,34	73,87	0,52	0,74
NB	17,67	21,00	16,90	12,19	17,67	21,00	0,14	0,13

Fonte: Autor (2024).

Analizando a matriz de confusão do método SVM (Figura 60), conjunto de dados com atributos de deslocamento, a probabilidade de classificar corretamente um dano na segunda mola com dano de 20% corresponde a 77,1% (27/35), na segunda mola com dano de 5% corresponde a 76,7% (23/30), na primeira mola com dano de 5% corresponde a 65,7% (23/35), na segunda mola com dano de 15% corresponde a 62,5% (20/32), na primeira mola com dano de 15% corresponde a 60% (15/25), na primeira mola com dano de 10% de corresponde a 54,8%

(17/31) e na segunda mola com dano de 10% corresponde a 52% (13/25). Outra informação que pode ser obtida dessa matriz é a probabilidade de considerar um dano quando o sistema está íntegro, que corresponde a 38,2% (13/34). Já a probabilidade de indicar uma estrutura íntegra quando o sistema está com dano (falso negativo) na primeira mola com dano de 5% é 8,6% (3/35), na primeira mola com dano de 10% é 3,2% (1/31), na segunda mola com dano de 10% é 4% (1/25), na primeira mola com dano de 15% é 8% (2/25) e na segunda mola com dano de 20% é 2,9% (1/35).

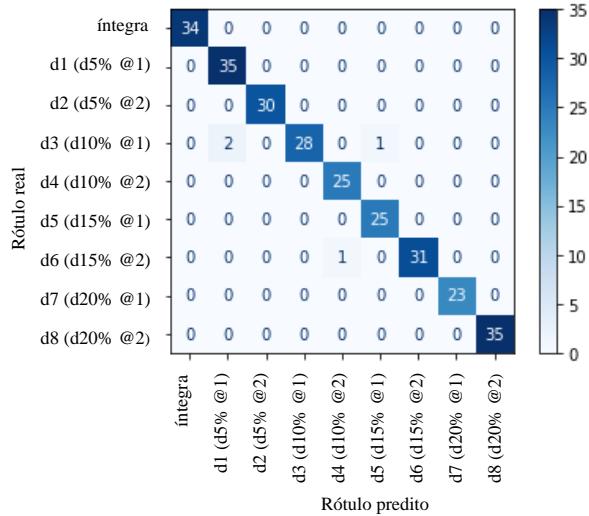
Figura 60 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 1, do modelo de viga engastada discretizada em dois elementos finitos.



Fonte: Autor (2024).

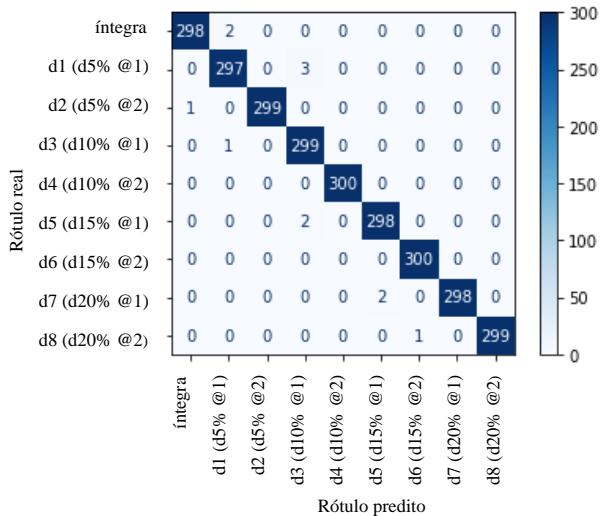
Na matriz de confusão do método SVM (Figura 61), conjunto de dados com atributos de deslocamento e aceleração, a primeira linha mostra que 34 das 34 amostras (soma da linha) para a condição da estrutura íntegra estão classificadas corretamente e, de forma análoga, para as linhas sucessivas, as estatísticas para identificação de danos podem ser obtidas. A probabilidade de classificar corretamente um dano é excelente na segunda mola com dano de 15% (96,9%) (31/32) e na primeira mola com dano de 10% (90,3%) (28/31), enquanto para os demais cenários de danos, todas as amostras são classificadas corretamente. Na matriz de confusão do método QDA (Figura 62), a primeira linha mostra que 298 das 300 amostras para a condição da estrutura íntegra estão classificadas corretamente e, de forma análoga, para as linhas sucessivas, as estatísticas para identificação de danos podem ser obtidas. A probabilidade de considerar um dano quando o sistema está íntegro é 0,7% (2/300) e a probabilidade de indicar uma estrutura íntegra quando a segunda mola está com dano de 5% é 0,3% (1/300).

Figura 61 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em dois elementos finitos.



Fonte: Autor (2024).

Figura 62 – Matriz de confusão do classificador QDA, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em dois elementos finitos.



Fonte: Autor (2024).

5.2.1.3 Aplicações do Framework Digital Twin

Para ilustração do *framework DT*, o *physical twin* foi simulado no tempo de 0 a 2 s, usando um período de amostragem de $1,3333 \times 10^{-3}$ s, que correspondeu a 1500 amostras para cada um dos nove cenários de interesse (dano investigado – DI): nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada elemento, de forma que tivesse um

número representativo de novos dados. Esses dados de entrada (sinais simulados) não são rotulados, uma vez que na prática não é conhecida a classe à qual pertencerá o sinal medido. Portanto, o *framework* DT apresenta a probabilidade de os dados pertencerem a determinada classe.

Para o índice de dano 1 (ID 1), atributo de deslocamento, foi avaliado o método SVM que apresentou melhor desempenho (Tabela 21) e para o índice de dano 2 (ID 2), atributos de deslocamento e aceleração, foram avaliados os métodos SVM e QDA, que apresentaram melhores resultados (Tabela 22). A Tabela 21 mostra que o método SVM identifica o elemento íntegro e com dano em todos os cenários investigados. No entanto, 3 dos 4 cenários investigados identificam corretamente a intensidade de dano na primeira mola e 1 dos 4 cenários investigados identifica corretamente a intensidade de dano na segunda mola. Quando os dados foram analisados com atributos de deslocamento e aceleração (ID 2), o método SVM identifica e quantifica corretamente todos os cenários investigados no elemento 1 e o método QDA identifica corretamente a localização do dano nos elementos 1 e 2.

Tabela 21 – Resultados do *framework* DT utilizando o método SVM com ID 1, modelo de viga engastada discretizada em dois elementos finitos.

DI	DA	P (%)
íntegra	íntegra	62,80
d5% @1	d5% @1	60,66
d10% @1	d10% @1	57,73
d15% @1	d15% @1	38,00
d20% @1	d15% @1	51,07
d5% @2	d10% @2	57,87
d10% @2	d20% @2	59,33
d15% @2	d20% @2	70,80
d20% @2	d20% @2	71,87

Fonte: Autor (2024).

Tabela 22 – Resultados do *framework* DT utilizando os métodos SVM e QDA com ID 2, modelo de viga engastada discretizada em dois elementos finitos.

DI	SVM		QDA	
	DA	P (%)	DA	P (%)
íntegra	íntegra	93,53	íntegra	96,07
d5% @ 1	d5% @ 1	91,00	d5% @ 1	91,73
d10% @ 1	d10% @ 1	89,80	d5% @ 1	82,19
d15% @ 1	d15% @ 1	87,93	d10% @ 1	87,87
d20% @ 1	d20% @ 1	93,47	d15% @ 1	80,33
d5% @ 2	d5% @ 2	85,27	d10% @ 2	91,73
d10% @ 2	d5% @ 2	76,07	d15% @ 2	87,60
d15% @ 2	d5% @ 1	66,87	d20% @ 2	99,33
d20% @ 2	d10% @ 1	72,67	d20% @ 2	99,67

Fonte: Autor (2024).

Foi utilizado o conjunto de dados com atributo de deslocamento (ID 1) para construir um *framework* DT do classificador SVM para detecção e localização de dano. Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, são mostrados na Tabela 23. A Tabela 24 apresenta as métricas de desempenho para o classificador, mostrando que ocorreu um aumento na precisão (Tabelas 20 e 24). É constatado que o método SVM detecta, localiza e quantifica corretamente todos os cenários de dano investigados, e com exceção da viga com condição íntegra, é apresentada ótima probabilidade (Tabela 25).

Tabela 23 – Hiperparâmetros ótimos para o classificador SVM, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.

Classificador	Parâmetro	Valor
SVM	C	1000
	Kernel	RBF
	Gama (kernel não linear)	1
	Função de decisão	Um contra um (OVO)

Fonte: Autor (2024).

Tabela 24 – Métricas de desempenho para o classificador SVM, conjunto de dados do modelo de viga engastada discretizada em dois elementos finitos.

Método	Acurácia (%)	Precisão (%)	Revocação (%)	F-score
SVM	85,19	81,96	79,61	80,64

Fonte: Autor (2024).

Tabela 25 – Resultados do *framework* DT para os dois primeiros níveis de investigação do dano estrutural, utilizando o método SVM com ID 1, modelo de viga engastada discretizada em dois elementos finitos.

DI	DA	P (%)
íntegra	íntegra	56,60
d5% @1	d @1	79,53
d10% @1	d @1	88,13
d15% @1	d @1	91,00
d20% @1	d @1	92,53
d5% @2	d @2	86,93
d10% @2	d @2	91,00
d15% @2	d @2	93,53
d20% @2	d @2	95,19

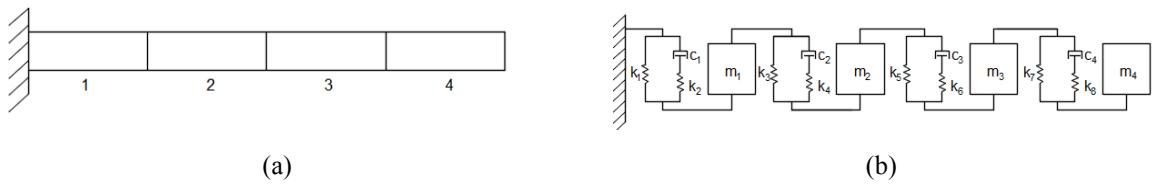
Fonte: Autor (2024).

5.2.2 Viga Engastada Discretizada em Quatro Elementos Finitos

O sistema foi discretizado em quatro elementos finitos (Figura 63(a)) e a idealização do modelo de viga foi construída com um modelo 4-DOF (Figura 63(b)). As três primeiras frequências naturais obtidas foram $\{20,889; 131,055; 369,368\}$ Hz e as três primeiras razões de amortecimento obtidas foram $\{2,0; 0,4; 0,5\}$ %, com $\alpha = 5,2000$ e $\beta = 2,8970 \times 10^{-6}$, estabelecidas para o modo fundamental e para o sexto modo.

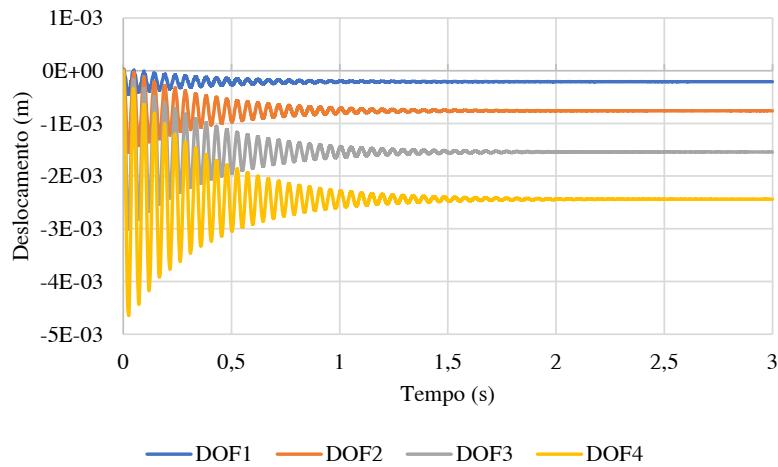
A resposta dinâmica mostra que o sistema oscila de forma rápida até decair para a resposta referente a força aplicada e as maiores amplitudes de deslocamento ocorrem para o DOF que está mais próximo do ponto de aplicação da força (Figura 64).

Figura 63 – (a) Viga engastada discretizada em quatro elementos finitos e (b) correspondente idealização.



Fonte: Autor (2024).

Figura 64 – Respostas de deslocamento no tempo, viga engastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

5.2.2.1 Respostas do SINDy

Os dados de treinamento para o SINDy foram gerados simulando o sistema (Equação 51) com as condições iniciais $\mathbf{x}(t = 0) = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$. Foi usado um período de amostragem de $7,1429 \times 10^{-5}$ s, que corresponde a sensores com taxa de amostragem de 14 kHz. Foram avaliados cinco diferentes conjuntos de treinamento para verificar as respostas geradas, o parâmetro δ , para o algoritmo STLSQ, e o hiperparâmetro λ , para o algoritmo LASSO, foram variados considerando um determinado intervalo, sendo escolhidos os $\delta_{\text{óptimo}}$ e $\lambda_{\text{óptimo}}$ de forma que minimizasse a medida de erro (RMSE) (Tabela 26 e Figura 65). O SINDy foi estruturado considerando a estrutura das equações de um modelo baseado na física (Equação 68).

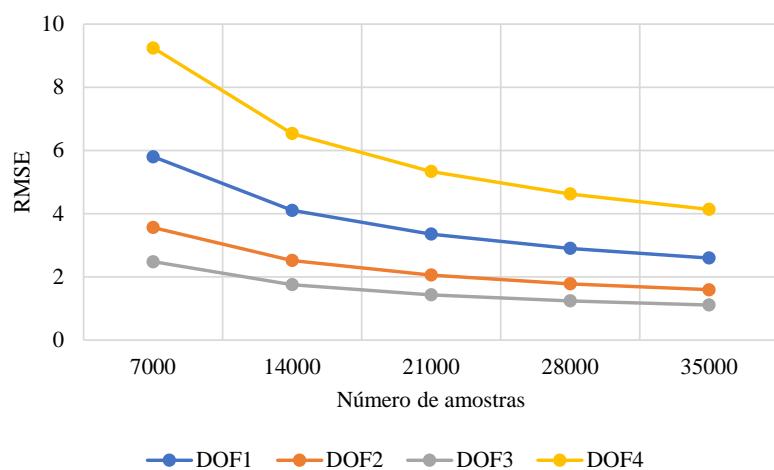
O método LASSO para o primeiro conjunto de treinamento, com 7000 amostras, teve um tempo de execução de 71 minutos e 10 segundos, mesmo reduzindo o intervalo em que o algoritmo foi variado, considerando a convergência. À medida que o número de amostras aumenta, o tempo de execução consequentemente aumenta, mostrando que o método LASSO tem um custo computacional maior, quando comparado com o método STLSQ. Dessa forma, foi escolhido para as análises o método STLSQ.

Tabela 26 – Avaliação das equações descobertas para os cinco conjuntos de treinamento, considerando o método STLSQ.

Tempo de integração (s)	Número de amostras	Método STLSQ	
		Tempo de execução (s)	$\delta_{óptimo}$
0,5	7000	4,0524	3×10^{-2}
1	14000	6,8191	3×10^{-2}
1,5	21000	10,0179	3×10^{-2}
2	28000	13,1456	3×10^{-2}
2,5	35000	16,4387	3×10^{-2}

Fonte: Autor (2024).

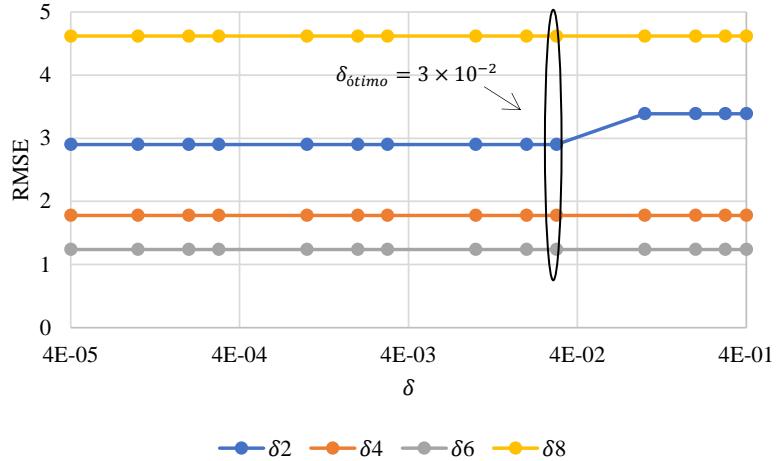
Figura 65 – Avaliação do erro (RMSE) para os cinco conjuntos de treinamento.



Fonte: Autor (2024).

Foi escolhido o conjunto de dados com 28000 amostras, uma vez o RMSE não apresentou diferença significativa quando comparado com o conjunto de dados com 35000 amostras. O parâmetro ótimo δ ($\delta_{óptimo} = 3 \times 10^{-2}$) foi determinado de forma que minimizasse a medida de RMSE (Figura 66). As equações de movimento descoberta pelo método STLSQ foram reescritas para parametrização de dano (Equação 74).

Figura 66 – Determinação do $\delta_{\text{o}timo}$, viga engastada discretizada em quatro elementos finitos.

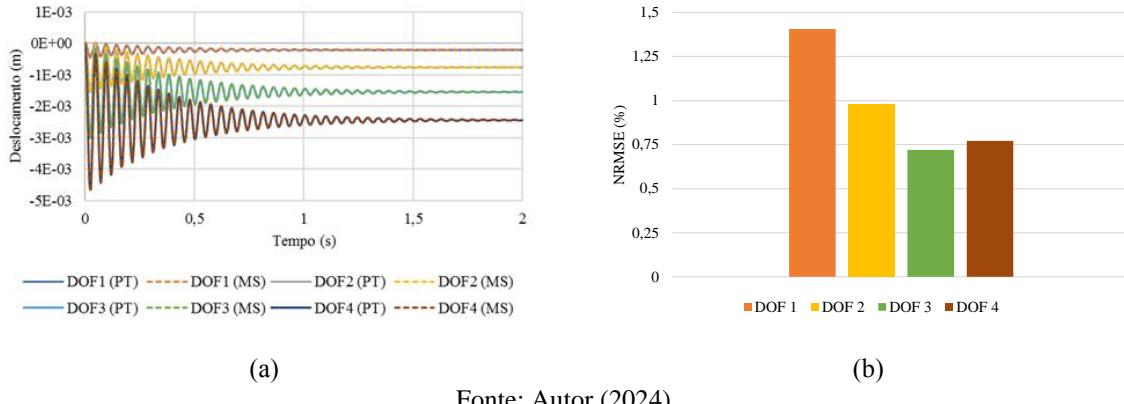


Fonte: Autor (2024).

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ x_2 = 10,543 - \left(1 - \frac{d_1}{2} - \frac{d_3}{2}\right) 1609040,873 x_1 - \left(\frac{\left(1 - \frac{d_1}{2} - \frac{d_2}{2}\right)}{(1 - d_2)}\right) 3,409 x_2 \\ \quad - \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 3,409 x_2 + (1 - d_3) 456323,345 x_3 + \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 0,227 x_4 \\ \dot{x}_3 = x_4 \\ x_4 = 15,443 + (1 - d_3) 1184585,461 x_1 - \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 10,570 x_2 - \left(1 - \frac{d_3}{2} - \frac{d_5}{2}\right) 1486420,179 x_3 \\ \quad - \left(\frac{\left(1 - \frac{d_3}{2} - \frac{d_4}{2}\right)}{(1 - d_4)}\right) 0,630 x_4 - \left(\frac{\left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right)}{(1 - d_6)}\right) 0,630 x_4 + (1 - d_5) 583174,314 x_5 - \left(\frac{\left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right)}{(1 - d_6)}\right) 0,878 x_6 \\ \dot{x}_5 = x_6 \\ x_6 = 16,373 + (1 - d_5) 2459060,851 x_3 + \left(\frac{\left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right)}{(1 - d_6)}\right) 10,675 x_4 - \left(1 - \frac{d_5}{2} - \frac{d_7}{2}\right) 3390054,622 x_5 \\ \quad - \left(\frac{\left(1 - \frac{d_5}{2} - \frac{d_6}{2}\right)}{(1 - d_6)}\right) 10,131 x_6 - \left(\frac{\left(1 - \frac{d_7}{2} - \frac{d_8}{2}\right)}{(1 - d_8)}\right) 10,131 x_6 + (1 - d_7) 1383527,450 x_7 + \left(\frac{\left(1 - \frac{d_7}{2} - \frac{d_8}{2}\right)}{(1 - d_8)}\right) 6,036 x_8 \\ \dot{x}_7 = x_8 \\ x_8 = -95,663 + (1 - d_7) 884930,899 x_5 + \left(\frac{\left(1 - \frac{d_7}{2} - \frac{d_8}{2}\right)}{(1 - d_8)}\right) 7,742 x_6 - (1 - d_7) 599234,578 x_7 \\ \quad - \left(\frac{\left(1 - \frac{d_7}{2} - \frac{d_8}{2}\right)}{(1 - d_8)}\right) 10,259 x_8. \end{array} \right. \quad (74)$$

Comparando as respostas de deslocamento no tempo do modelo SINDy (Equação 74) com a resposta de entrada (Figura 67(a)), uma ótima concordância das respostas pode ser observada, e o NRMSE para os DOFs são inferiores a 1,5% (Figura 67(b)).

Figura 67 – Comparação de respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS).



Fonte: Autor (2024).

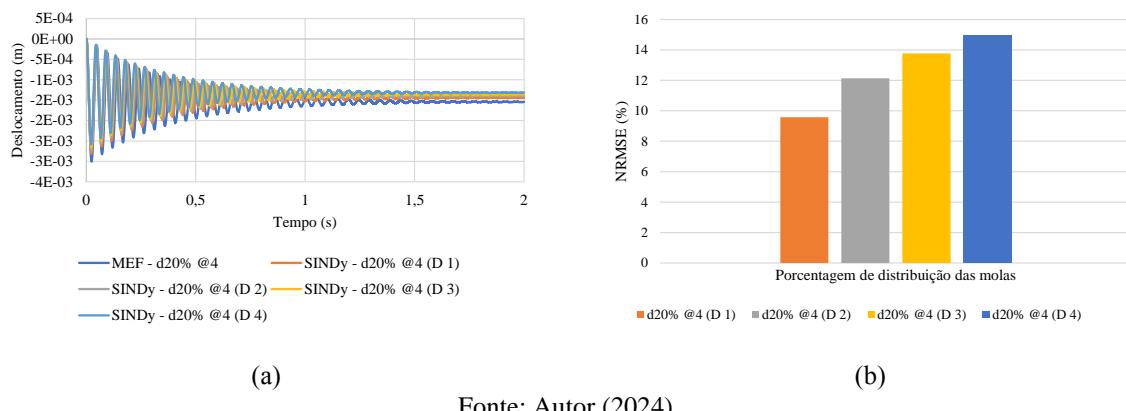
5.2.2.2 Resultados e Discussões dos Métodos de Machine Learning

O modelo SINDy (Equação 74) foi integrado no tempo de 0 a 2 s, usando um período de amostragem de $6,6667 \times 10^{-4}$ s, para cada um dos cenários de interesse: nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%). Cada intensidade de dano foi distribuída para cada i -ésima mola ($i = 1, 3, 5, 7$) e j -ésima mola ($i = 2, 4, 6, 8$) (Figura 63). Foi considerada uma porcentagem de distribuição (D), para cada intensidade de dano, de 10% e 90% (D 1), 15% e 85% (D 2), 20% e 80% (D 3), 25% e 75% (D 4), em cada i -ésima mola e j -ésima mola, respectivamente. Isso resultou em quatro conjuntos de dados (CD) de treinamento diferentes, cada um com 5100 amostras (300×17), sendo cada conjunto de dados final uma matriz de dimensão $((300 \times 17) \times 4)$. Para as quatro porcentagens de distribuição das molas foram calculados o NRMSE em relação as respostas do MEF, considerando um dano com intensidade de 20% em cada mola, uma vez que é esperado que ocorra o maior erro entre as respostas. Comparando as respostas, é observado que o maior erro ocorre para o DOF3 quando o dano é introduzido na mola 4 e o erro aumenta à medida que a intensidade de dano na i -ésima aumenta (Figura 68).

Foi escolhido o método SVM para avaliação do desempenho dos quatro conjuntos de dados, considerando como comparativo respostas de um conjunto de dados gerado do modelo MEF. Para cada conjunto de dados, foi realizada uma busca de grade com base nos hiperparâmetros da Tabela 19. Foram investigados 7 diferentes índices de dano (ID), com conjunto de dados do MEF, com atributo de deslocamento (ID 1), deslocamento e aceleração (ID 2), deslocamento, velocidade e aceleração (ID 3), deslocamento e velocidade (ID 4), velocidade e aceleração (ID 5), velocidade (ID 6) e aceleração (ID 7) (Figura 69(a)). Foi

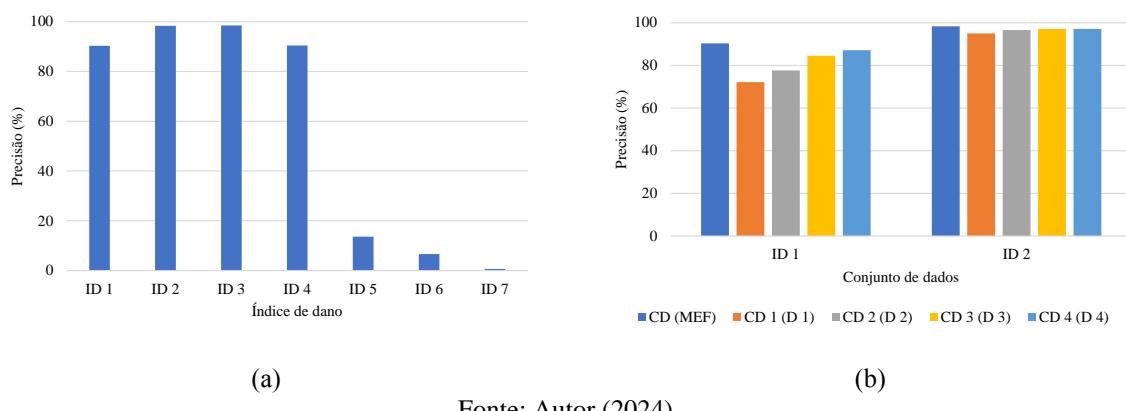
escolhido o índice de dano de deslocamento e aceleração (ID 2), que apresentou desempenho significativo quando comparado a precisão e o número de atributos dos cenários investigados. Dessa forma, foram comparados os índices de dano com atributo de deslocamento (ID 1) e com atributos de deslocamento e aceleração (ID 2) com dados do MEF (Figura 69(b)). Foi constado que à medida que o erro aumenta (Figura 68(b)) aumenta a precisão do modelo de ML (Figura 69(b)). Dessa forma, deve ser escolhida uma porcentagem de distribuição que considere o equilíbrio entre o erro, discrepância entre as respostas SINDy e MEF, e o desempenho do modelo. Portanto, foi escolhido como porcentagem de distribuição 20% e 80%.

Figura 68 – (a) Comparação de respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para o DOF 3 e (b) correspondentes NRMSE.



Fonte: Autor (2024).

Figura 69 – (a) Índices de dano para o conjunto de dados do MEF e (b) índices de dano para os conjuntos de dados.

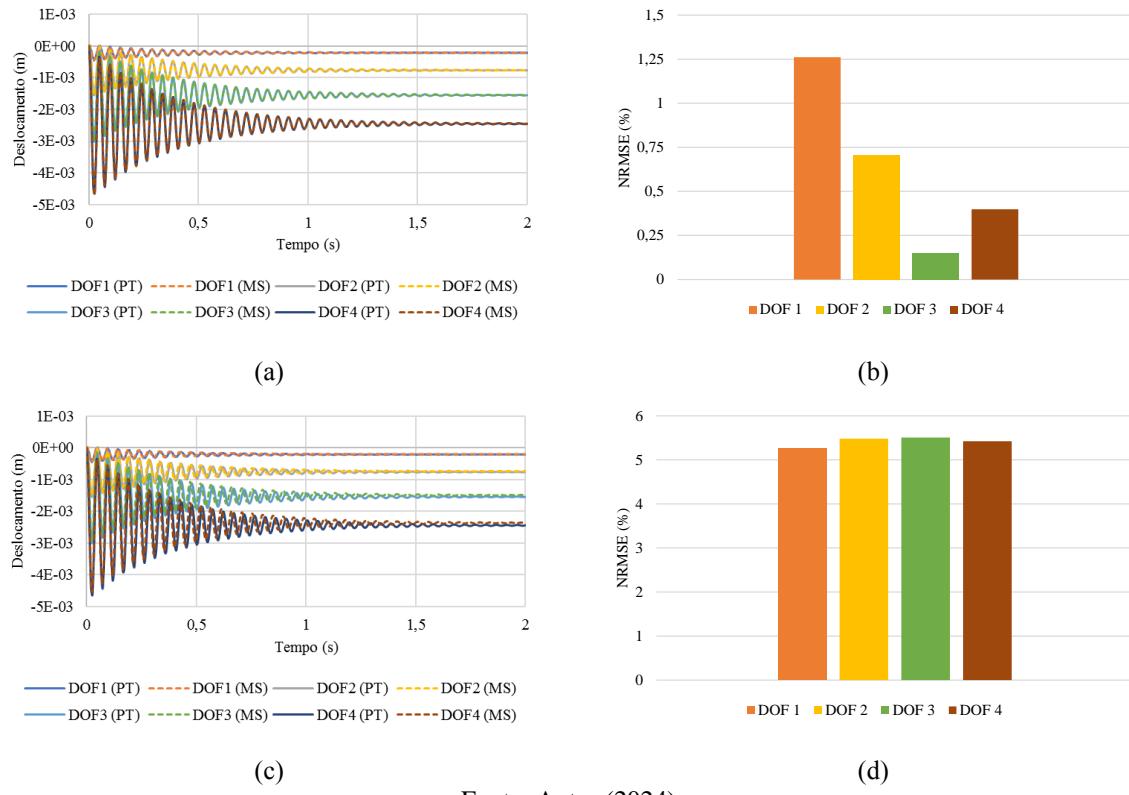


Fonte: Autor (2024).

Foram comparadas as respostas de deslocamento no tempo, do *physical twin* e do modelo SINDy, para dois cenários: dano de 5% em cada uma das molas e dano de 20% em cada uma das molas (Figuras 70 e 71), sendo apresentadas as respostas que apresentam menor e maior erro (NRMSE). A resposta muda dependendo da intensidade do dano, bem como da mola

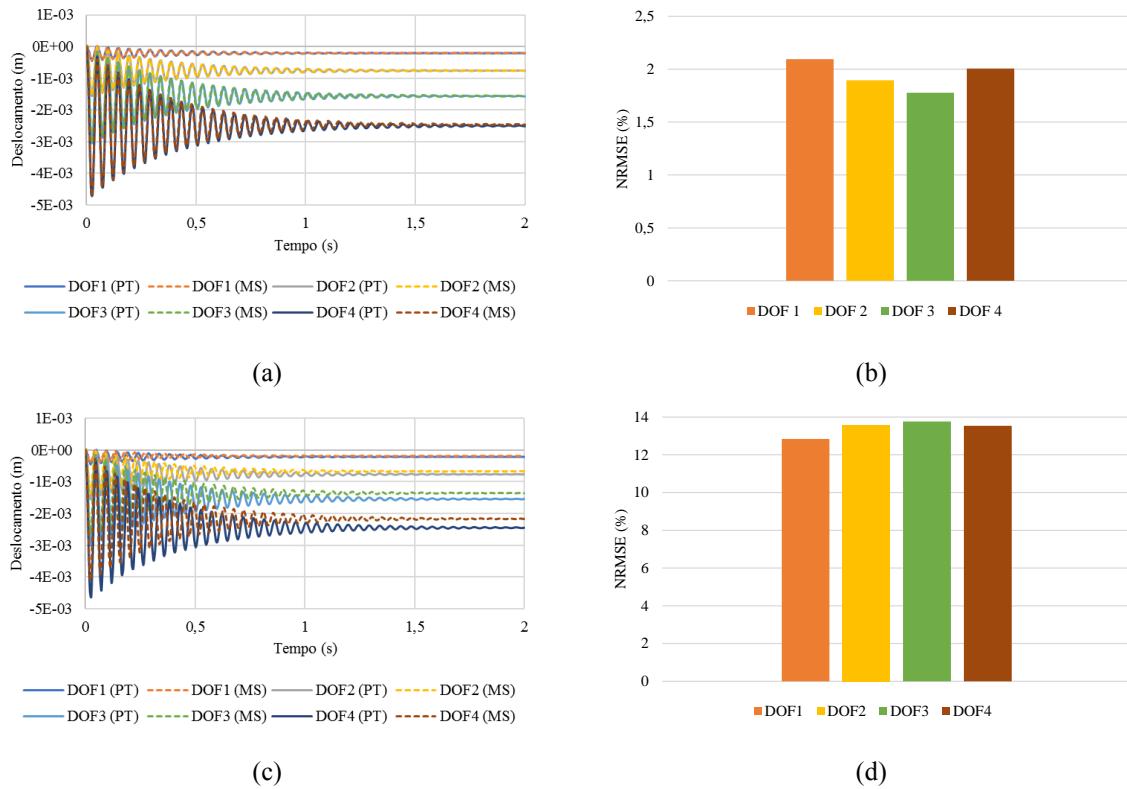
com dano observada. Quando o dano de 5% está na mola 3 é apresentada uma ótima correspondência e menor NRMSE (NRMSE inferior a 1,3%) (Figura 70(a-b)) e quando o dano está na mola 4 é observada uma boa correspondência e maior NRMSE (NRMSE inferior a 5,6%) (Figura 70(c-d)). Para o dano de 20%, quando o dano está na mola 3 é observada uma ótima correspondência e menor erro (NRMSE inferior 2,1%) (Figura 71(a-b)) e quando o dano é introduzido na mola 4 verifica-se uma concordância satisfatória com maior NRMSE (NRMSE inferior a 14%) (Figura 71(c-d)).

Figura 70 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 5% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Figura 71 – Comparação das respostas de deslocamento no tempo entre o *physical twin* (PT), viga engastada discretizada em quatro elementos finitos, e modelo SINDy (MS) para dano de 20% na (a) terceira mola, (b) com correspondentes NRMSE, e (c) quarta mola, (d) com correspondentes NRMSE.



Fonte: Autor (2024).

Os hiperparâmetros ótimos obtidos da busca em grade, com validação cruzada, usando o conjunto de dados com porcentagem de distribuição 20% e 80% (CD 3) para os dois índices de dano, com atributos de deslocamento (ID 1) e deslocamento e aceleração (ID 2) são mostrados na Tabela 27. A Tabela 28 apresenta as métricas de desempenho para dos classificadores. Para o índice de dano 1 (ID 1), a precisão indica que os métodos SVM e QDA apresentaram os melhores desempenhos, seguido dos métodos LDA, k-NN e os baseados em árvores, e o método NB apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão. Quando analisado o índice de dano 2 (ID 2) a precisão indica que os métodos QDA e SVM apresentaram os melhores desempenhos, seguido dos métodos LDA, os baseados em árvore e o k-NN. O método NB novamente apresentou desempenho bem inferior em relação aos outros métodos. Analisando a métrica de desempenho revocação, as conclusões são semelhantes às constatadas na precisão.

Tabela 27 – Hiperparâmetros ótimos para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em quatro elementos finitos.

Classificador	Parâmetro	Valor	
		ID 1	ID 2
QDA	-	-	-
LDA	-	-	-
NB	Modelo	Gaussiano	Gaussiano
k-NN	Número de vizinhos	3	3
	Peso	Distância	Distância
	Métrica	Euclidiana	Euclidiana
SVM	C	1000	1000
	Kernel	Linear	Linear
	Função de decisão	Um contra um (OVO)	Um contra um (OVO)
RF	Número de estimadores	100	500
	Profundidade máxima da árvore	14	14
	Critério	Entropia	Entropia
ET	Número de estimadores	150	500
	Profundidade máxima da árvore	20	20
	Critério	Gini	Entropia
DTs	Profundidade máxima da árvore	16	14
	Critério	Entropia	Entropia

Fonte: Autor (2024).

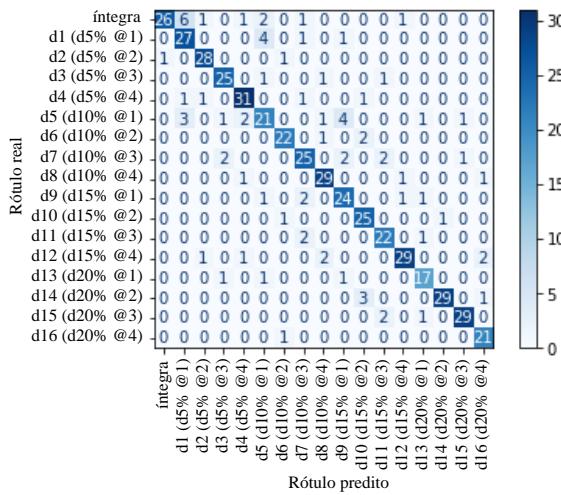
Tabela 28 – Métricas de desempenho para os classificadores de ML, conjunto de dados do modelo de viga engastada discretizada em quatro elementos finitos.

Classificador	Acurácia (%)		Precisão (%)		Revocação (%)		F-score	
	ID 1	ID 2	ID1	ID 2	ID 1	ID 2	ID 1	ID 2
SVM	84,31	96,86	84,49	97,10	85,01	97,08	0,84	0,97
QDA	81,25	99,98	81,34	99,98	81,25	99,98	0,81	0,99
LDA	78,94	88,39	78,92	87,94	78,94	88,39	78,73	0,88
k-NN	55,69	50,39	57,05	51,78	56,43	50,83	55,95	0,51
ET	54,12	53,73	55,67	54,13	54,86	53,95	0,54	0,54
RF	51,76	54,51	52,37	55,53	52,22	54,75	0,52	0,55
DTs	47,65	48,23	48,89	48,88	48,01	48,65	0,47	0,48
NB	8,92	8,22	7,19	4,76	8,92	8,22	0,06	0,05

Fonte: Autor (2024).

A matriz de confusão do método SVM (Figura 72), conjunto de dados com atributos de deslocamento, mostra que a probabilidade de considerar um dano quando o sistema está íntegro é 31,6% (12/38) e a probabilidade de indicar uma estrutura íntegra quando a segunda mola está com dano de 5% é 3,3% (1/30).

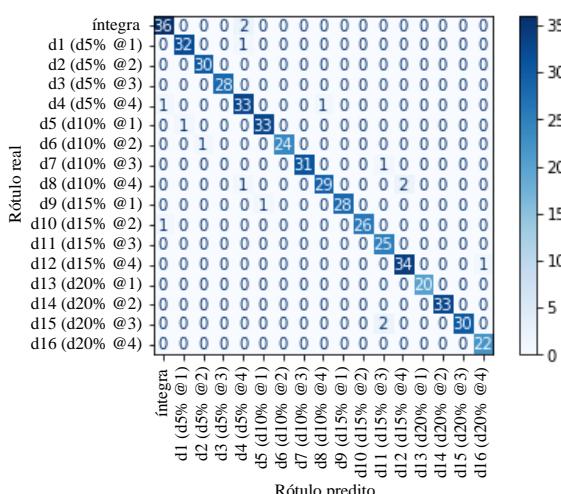
Figura 72 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 1, do modelo de viga engastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

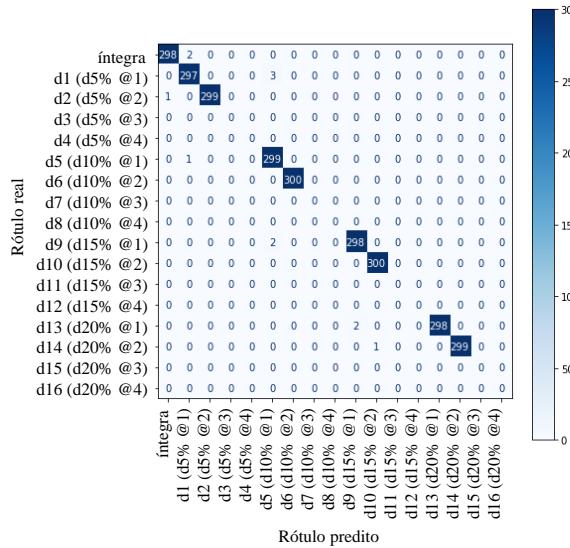
A matriz de confusão do método SVM (Figura 73), conjunto de dados com atributos de deslocamento e aceleração, mostra que a probabilidade de considerar um dano quando o sistema está íntegro é 5,3% (2/38) e a probabilidade de indicar uma estrutura íntegra quando a quarta mola está com dano de 5% é 2,9% (1/35) e quando a segunda mola está com dano de 15% é 3,7% (1/27). A matriz de confusão do método QDA (Figura 74) indica que a probabilidade de considerar um dano quando o sistema está íntegro é 0,7% (2/300) e a probabilidade de indicar uma estrutura íntegra quando a segunda mola está com dano de 5% é 0,3% (1/300).

Figura 73 – Matriz de confusão do classificador SVM, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

Figura 74 – Matriz de confusão do classificador QDA, conjunto de dados, índice de dano 2, do modelo de viga engastada discretizada em quatro elementos finitos.



Fonte: Autor (2024).

5.2.2.3 Aplicações do Framework Digital Twin

Para ilustração do framework DT, o *physical twin* foi simulado no tempo de 0 a 2 s, usando um período de amostragem de $1,3333 \times 10^{-3}$ s, que correspondeu a 1500 amostras para cada um dos cenários de interesse (dano investigado – DI): nenhum dano (íntegra) e 4 intensidades de dano (5%, 10%, 15% e 20%) em cada elemento, de forma que tivesse um número representativo de dados que o modelo não conhecesse. Esses dados de entrada (sinais simulados) não são rotulados, uma vez que na prática não é conhecida a classe ao qual pertence o sinal medido. Portanto, o framework DT apresenta a probabilidade de os dados pertencer a determinada classe.

Para o índice de dano 1 (ID 1), atributo de deslocamento, foi avaliado o método SVM, uma vez que apresentou melhor desempenho (Tabela 29) e para o índice de dano 2 (ID 2), atributos de deslocamento e aceleração, foram avaliados os métodos SVM e QDA, que apresentaram melhores resultados (Tabela 30). Considerando o índice de dano 1, o classificador SVM identifica e quantifica corretamente os danos nos elementos 1 e 3 dos 4 cenários investigados no elemento 2. No elemento 3 não são identificados os danos, embora acerte a quantificação, e no elemento 4, 3 dos 4 cenários investigados detecta e localiza o dano (Tabela 29). Para o índice de dano 2, o classificador SVM detecta e localiza os danos nos elementos 1 e 3, com alguns acertos quanto à quantificação. No elemento 2, não são identificados os danos e no elemento 4, 1 dos 4 cenários investigados detecta e localiza o dano (Tabela 30). O

classificador QDA detecta e localiza os danos nos elementos 1, com alguns acertos quanto à quantificação. No elemento 2, os danos são identificados e quantificados corretamente e no elemento 3 os danos não são identificados. E, por fim, no elemento 4, 3 dos 4 cenários investigados detecta e localiza o dano.

Tabela 29 – Resultados do *framework* DT utilizando o método SVM com ID 1, modelo de viga engastada discretizada em quatro elementos finitos.

DI	DA	P (%)
íntegra	íntegra	85,73
d5% @ 1	d5% @ 1	80,07
d10% @ 1	d10% @ 1	80,40
d15% @ 1	d15% @ 1	80,33
d20% @ 1	d20% @ 1	89,47
d5% @ 2	d5% @ 2	91,13
d10% @ 2	d10% @ 2	89,20
d15% @ 2	d15% @ 2	77,80
d20% @ 2	d15% @ 2	91,07
d5% @ 3	d5% @ 1	80,27
d10% @ 3	d10% @ 1	81,47
d15% @ 3	d15% @ 1	81,60
d20% @ 3	d20% @ 1	91,73
d5% @ 4	íntegra	74,93
d10% @ 4	d5% @ 4	86,27
d15% @ 4	d5% @ 4	78,47
d20% @ 4	d10% @ 4	87,33

Fonte: Autor (2024).

Tabela 30 – Resultados do *framework* DT utilizando os métodos SVM e QDA com ID 2, modelo de viga engastada discretizada em quatro elementos finitos.

DI	SVM		QDA	
	DA	P (%)	DA	P (%)
íntegra	íntegra	84,60	íntegra	83,00
d5% @1	d10% @1	75,33	d5% @1	85,87
d10% @1	d20% @1	93,40	d10% @1	79,13
d15% @1	d20% @1	95,40	d10% @1	80,80
d20% @1	d20% @1	95,6	d15% @1	86,13
d5% @2	d15% @3	76,87	d5% @2	90,27
d10% @2	d20% @3	91,67	d10% @2	89,80
d15% @2	d20% @3	93,47	d15% @2	91,47
d20% @2	d20% @3	94,93	d20% @2	90,27
d5% @3	d5% @3	82,87	d5% @1	85,47
d10% @3	d15% @3	75,87	d10% @1	82,07
d15% @3	d20% @3	87,93	d15% @1	78,73
d20% @3	d20% @3	92,60	d20% @1	87,47
d5% @4	íntegra	83,53	íntegra	76,93
d10% @4	íntegra	81,40	d5% @4	75,93
d15% @4	íntegra	65,53	d5% @4	77,33
d20% @4	d5% @4	73,13	d10% @4	71,20

Fonte: Autor (2024).

6. CONCLUSÕES E SUGESTÕES DE TRABALHOS FUTUROS

6.1 Conclusões

Neste trabalho foi construído e apresentado um novo *framework* para SHM, mais especificamente para identificação de danos em elementos estruturais lineares de aço (barras comprimidas e vigas). O *framework* construído integra um modelo híbrido baseado na física e orientado a dados e métodos de *machine learning*, para a construção de um *framework digital twin*. O *framework digital twin* construído relaciona as entradas do *physical twin* a cenários de danos específicos, avisando se a estrutura apresenta dano, onde está localizado e qual a sua intensidade, dessa forma, foram abordados três principais níveis de investigação do dano estrutural: detecção, localização e quantificação. A estratégia foi avaliada em diferentes configurações de aplicações, considerando a vibração axial e vibração transversal da estrutura.

Considerando o modelo de barra engastada, com vibração axial, os métodos SVM, LDA e QDA apresentaram melhor desempenho à medida que aumentou o número de classes e, consequentemente, o número de amostras e o número de atributos, enquanto os métodos k-NN e baseados em árvores apresentam menor desempenho. Esses resultados podem ser atribuídos devido os métodos SVM, LDA e QDA criarem funções de decisão.

A identificação do dano na barra biengastada apresentou menor desempenho, quando comparada com a condição engastada, sendo constatado que identificar dano em estruturas hiperestáticas é mais desafiador. A discretização da estrutura em dois elementos finitos, resultando em um único atributo, não foi suficiente para detectar, localizar e quantificar os danos estruturais. Os métodos SVM e LDA apresentaram melhor desempenho à medida que o número de classes foi aumentado e, consequentemente, o número de amostras e o número de atributos, enquanto os demais métodos apresentaram menor desempenho.

Na vibração transversal foram investigados dois índices de dano, utilizando atributo de deslocamento e atributos de deslocamento e aceleração. Considerando o atributo de deslocamento, os métodos SVM, LDA e QDA apresentaram melhor desempenho à medida que o número de classes foi aumentado e, consequentemente, o número de amostras e o número de atributos, enquanto os métodos k-NN e baseados em árvores apresentam menor desempenho ou desempenho semelhante. Para os atributos de deslocamento e aceleração, os métodos SVM, LDA e QDA apresentaram desempenho semelhante à medida que aumentou o número de classes e, consequentemente, o número de amostras e o número de atributos, enquanto os métodos k-NN e baseados em árvores apresentam menor desempenho. Apesar do aumento de

desempenho considerável quando do uso do atributo de deslocamento e aceleração, apenas o uso de recurso de deslocamento se mostrou promissor para detectar e localizar o dano.

O método SVM apresentou melhor desempenho para identificação de danos estruturais, o que pode ser atribuído à sua capacidade de formar um limite preciso, com possibilidade de funções não lineares, a partir de uma pequena quantidade de dados de treinamento. O método *Naive Bayes* apresentou desempenho muito inferior em relação aos demais métodos, sendo um método ineficaz para identificação de dano, devido a suposições de independência condicional entre as características de cada classe.

A aplicação do *framework digital twin*, considerando as condições íntegra e de 1% até 25% de dano em cada elemento discretizado do modelo de barra, mostrou que o *framework DT* associa os cenários de dano investigados com cenários de danos específicos esperados e se mostrou robusto para determinados níveis de ruído.

Neste estudo, foram considerados para o treinamento cenários da estrutura íntegra e quatro intensidades de dano (5%, 10%, 15% e 20%) em cada elemento, considerando uma investigação mais desafiadora, no entanto, que considera o mapeamento de toda estrutura. Reduzir os cenários de intensidade de dano ou considerar os dois primeiros níveis de investigação do dano estrutural, detecção e localização, pode aumentar a precisão do *framework digital twin*.

6.2 Sugestões de Trabalhos Futuros

Considerando os diferentes cenários avaliados, as técnicas propostas apresentaram contribuições significativas para o estabelecimento de um sistema de SHM, podendo ser expandidas para várias aplicações de engenharia.

Para aplicações futuras, sugere-se investigar o dano múltiplo, considerando mais de um elemento com dano simultaneamente, bem como utilizar dados de sensores físicos e adicionar ao DT uma camada de comunicação. Outra vertente promissora consiste em investigar modelos baseados na física e construídos utilizando uma descrição de parâmetros concentrados para idealização do modelo de viga, além da adoção de condições de vinculações e carregamentos distintas das utilizadas pelo autor.

REFERÊNCIAS

- ADAMS, R. D.; CAWLEY, P. **The Location of Defects in Structures From Measurements of Natural Frequencies.** Journal of Strain Analysis. v. 14, n. 2, p. 49-57, 1979.
- ALFEILAT, H. A. A.; HASSANAT, A. B. A.; LASASSMEH, O.; TARAWNEH, A. S.; ALHASANAT, M. B.; SALMAN, H. S. E.; PRASATH V. B. S. **Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review.** Big Data v. 7, n. 4, p. 221-248, 2019.
- ALI, N., NEAGU, D.; TRUNDLE, P. **Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets.** SN Applied Sciences. v. 1: 1559, 2019.
- ALTMAN, E. I.; MARCO, G.; VARETTO, F. **Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience).** Journal of Banking & Finance, v. 18, n. 3, p. 505–529, 1994.
- ARAVEEPORN, A. **Comparing the Linear and Quadratic Discriminant Analysis of Diabetes Disease Classification Based on Data Multicollinearity.** International Journal of Mathematics and Mathematical Sciences. v. 2022: 7829795, 2022.
- ATTARAN, M.; CELIK, B. G. **Digital Twin: Benefits, use cases, challenges, and opportunities.** Decision Analytics Journal. v. 6: 100165, 2023.
- AVCI, O.; ABDELJABER, O.; KIRANYAZ, S.; HUSSEIN, M.; GABBOUJ, M. INMAN, D. J. **A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications.** Mechanical Systems and Signal Processing. v. 147: 107077, 2021.
- AYDIN, A; BASU, A. **The Schmidt hammer in rock material characterization.** Engineering Geology. v. 81, n. 1, p. 1-14, 2005.

BADO, M. F.; TONELLI, D.; POLI, F.; ZONTA, D.; CASAS, J. R. **Digital twin for civil engineering systems: an exploratory review for distributed sensing updating.** Sensors. v. 22, n. 9: 3168, 2022.

BIGONI, C.; HESTHAVEN, J. S. **Simulation-based anomaly detection and damage localization: an application to structural health monitoring.** Computer Methods in Applied Mechanics and Engineering. v. 363: 112896, 2020.

BISHOP, C. M. **Pattern Recognition and Machine Learning.** 1. ed. New York: Springer, 2006.

BLAKSETH, S. S.; RASHEED, A.; KVAMSDAL, T.; SAN, O. **Combining physics-based and data-driven techniques for reliable hybrid analysis and modeling using the corrective source term approach.** Applied Soft Computing. v. 28: 109533, 2022.

BONGARD, J.; LIPSON, H. **Automated reverse engineering of nonlinear dynamical systems.** Proceedings of the National Academy of Sciences. v. 104, n. 24, p. 9943–9948, 2017.

BREIMAN, L. **Random Forests.** Machine Learning. v. 45, n. 1, p. 5-32, 2001.

BREIMAN, L.; FRIEDMAN, J.; OLSHEN, R.; STONE, C. **Classification and Regression Trees.** 1. ed. Boca Raton, FL: Chapman & Hall/CRC, 1984.

BRIDEWELL, W.; LANGLEY, P.; TODOROVSKI, L.; DŽEROSKI , S. **Inductive process modeling.** Machine learning. v. 71, n. 1, p. 1-32, 2008.

BROWNJOHN, J. M. W. **Structural health monitoring of civil infrastructure.** Philos Trans A Math Phys Eng Sci. v. 365, n. 1851, p. 589–622, 2007.

BRUNTON, S. L.; PROCTOR, J. L.; KUTZ, J. N. **Discovering governing equations from data by sparse identification of nonlinear dynamical systems.** Proceedings of the National Academy of Sciences. v. 113, n. 15, p. 3932-3937, 2016.

- BURGES, C. J. C. **A Tutorial on Support Vector Machines for Pattern Recognition.** Data Mining and Knowledge Discovery. v. 2, p. 121–167, 1998.
- CALLCUT, M.; AGLIOZZO, J.- P. C.; VARGA, L.; MCMILLA, L. **Digital Twins in Civil Infrastructure Systems.** Sustainability. v. 13, n. 20: 11549, 2021.
- CAMPESATO, O. **Artificial Intelligence Machine Learning and Deep Learning.** 1. ed. Dulles, VA: Mercury Learning and Information, 2020.
- CHAKRABORTY, S.; ADHIKARI, S.; GANGULI, R. **The role of surrogate models in the development of digital twins of dynamic systems.** Applied Mathematical Modelling. v. 90, p. 662-681, 2021.
- CHAMPION, K.; ZHENG, P.; ARAVKIN, A. Y.; BRUNTON, S. L.; KUTZ, J. N. **A Unified Sparse Optimization Framework to Learn Parsimonious Physics-Informed Models From Data.** IEEE Access. v. 8, p. 169259-169271, 2020.
- CHIACHÍO, M.; MEGÍA, M.; CHIACHÍO, J.; FERNANDEZ, J.; JALÓN, M. L. **Structural digital twin framework: Formulation and technology integration.** Automation in Construction. v. 140: 104333, 2022.
- CLOUGH, R. W.; PENZIEN, J. **Dynamics of Structures.** 3. ed. Berkeley, CA: Computers & Structures, Inc., 2003.
- CORTES, C.; VAPNIK, V. **Support vector networks.** Machine Learning. v. 20, n. 3, p. 273–296, 1995.
- COVER, T.; HART, P. **Nearest neighbor pattern classification.** IEEE Transactions on Information Theory. v. 13, n. 1, p. 21-27, 1967.
- CRAIG JR., R. R.; KURDILA, A. J. **Fundamentals of Structural Dynamics.** 2. ed. Hoboken, NJ: John Wiley & Sons, Inc., 2006.

CRISTIANINI, N.; SHawe-Taylor, J. **An Introduction to Support Vector Machines and other kernel-based learning methods**. Cambridge University Press, 2000.

CUNNINGHAM, P.; DELANY, S. J. **k-Nearest Neighbour Classifiers - A Tutorial**. ACM Computing Surveys. v. 54, n. 6: 128, 2022.

DAM, M.; BRØNS, M.; RASMUSSEN, J. J.; NAULIN, V.; HESTHAVEN, J. S. **Sparse identification of a predator-prey system from simulation data of a convection model**. Physics of Plasmas. v. 24, n. 2: 022310, 2017.

DE SILVA, B. M.; HIGDON, D. M.; BRUNTON S. L.; KUTZ, J. N. **Discovery of Physics From Data: Universal Laws and Discrepancies**. Frontiers in Artificial Intelligence. v. 3: 25, 2020.

DEZA, M. M.; DEZA, E. **Encyclopedia of distances**. 2. ed. Springer Berlin, Heidelberg, 2013

DILLEN, A.; LATHOUWERS, E.; MILADINOVIC, A.; MARUSIC, U.; GHAFFARI, F.; ROMAIN, O.; MEEUSEN, R.; DE PAUW, K. **A data-driven machine learning approach for brain-computer interfaces targeting lower limb neuroprosthetics**. Frontiers in Human Neuroscience. v. 16: 949224, 2022.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern Classification**. 2. ed. New York: John Wiley & Sons, 2001.

FARRAR, C. R; DOEBLING, S. W.; NIX, D. A. **Vibration-based structural damage identification**. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. v. 359, n. 1778, p. 131-149, 2001.

FARRAR, C. R. LIEVEN, N. A. **Damage prognosis: the future of structural health monitoring**. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. v. 365, n. 1851, p. 623–632, 2007.

FIX, E.; HODGES, J. L. **Discriminatory analysis. Nonparametric discrimination; consistency properties.** Technical Report 4, USAF School of Aviation Medicine, Randolph Field, TX, USA, 1951.

FUGATE, M. L.; SOHN, H.; FARRAR, C. R. **Vibration-based damage detection using statistical process control.** Mechanical Systems and Signal Processing. v. 15, n. 4, p. 707-721, 2001.

FULLER, A.; FAN, Z.; DAY, C.; BARLOW, C. **Digital Twin: enabling technologies, challenges and open research.** IEEE ACCESS. v. 8, p. 108952–108971, 2020.

GEURTS, P.; ERNST, D.; WEHENKEL, L. **Extremely randomized trees.** Machine Learning. v. 63, n. 1, p. 3-42, 2006.

GHAHRAMANI, Z. **Probabilistic machine learning and artificial intelligence.** Nature. v. 521, p. 452-459, 2015.

GHOJOGH, B.; CROWLEY, M. **Linear and Quadratic Discriminant Analysis: Tutorial.** ArXiv: 1906.02590, 2019.

GOMEZ-CABRERA, A.; ESCAMILLA-AMBROSIO, P. J. **Review of Machine-Learning Techniques Applied to Structural Health Monitoring Systems for Building and Bridge Structures.** Applied Sciences. v. 12, n. 21: 10754, 2022.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning.** Cambridge, MA: The MIT Press, 2017.

GRIEVES, M. **Digital twin: manufacturing excellence through virtual factory replication.** White paper, p. 1–7, 2014.

GUO, Y.; HASTIE, T.; TIBSHIRANI, R. **Regularized linear discriminant analysis and its application in microarrays.** Biostatistics. v. 8, n. 1, p. 86–100, 2007.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction.** 2. ed. Springer Series in Statistics. New York: Springer, 2017.

HASTIE, T.; TIBSHIRANI, R.; WAINWRIGHT, M. **Statistical Learning With Sparsity: The Lasso and Generalizations.** Boca Raton, FL: CRC Press, 2015.

HERBRICH, R. **Learning Kernel Classifiers: Theory and Algorithms.** Cambridge, MA: MIT Press, 2001.

HOFFMANN, M.; FRÖHNER, C.; NOÉ, F. Reactive SINDy: **Discovering governing reactions from concentration data.** The Journal of Chemical Physics. v. 150, n. 2: 025101, 2019.

HULETT, C.; HALL, A.; QU, G. **Dynamic selection of k nearest neighbors in instance-based learning.** 2012 IEEE 13th International Conference on Information Reuse & Integration (IRI), p. 85-92, 2012.

JAMES , G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. **An Introduction to Statistical Learning with Applications in R.** 2. ed. New York: Springer, 2021.

JOHN, G. H.; LANGLEY, P. **Estimating continuous distributions in bayesian classifiers.** Proceedings of the Eleventh conference on Uncertainty in artificial intelligence (UAI'95). San Francisco, CA: Morgan Kaufmann Publishers Inc. p. 338–345, 1995.

JONES, D.; SNIDER, C.; NASSEHI, A.; YON, J.; HICKS, B. **Characterising the digital twin: A systematic literature review.** CIRP Journal of Manufacturing Science and Technology. v. 29, parte A, p. 36-52, 2020.

JORDAN, M. I.; MITCHELL, T. M. **Machine learning: Trends, perspectives, and prospects.** Science. v. 349, n.6245, p. 255–260, 2015.

- KAN, C.; ANUMBA, C. J. **Digital Twins as the Next Phase of Cyber-Physical Systems in Construction.** In: CHO, Y.; LEITE, F.; BEHZADAN, A.; WANG, C. (eds.). Computing in Civil Engineering 2019: Data, Sensing, and Analytics. Reston, VA: American Society of Civil Engineers, 2019. p. 256–264.
- KANG, S. **k-Nearest Neighbor Learning with Graph Neural Networks.** Mathematics. v. 9, n. 8: 830, 2021.
- KAISER, E.; KUTZ, J. N.; BRUNTON, S. L. **Sparse identification of nonlinear dynamics for model predictive control in the low-data limit.** Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, v. 474, n. 2219: 20180335, 2018.
- KHAN, S.; YAIRI, T. **A review on the application of deep learning in system health management.** Mechanical Systems and Signal Processing. v. 107, p. 241-265, 2018.
- KUBAT, M. **An Introduction to Machine Learning.** 2. ed. Cham: Springer, 2017.
- KUHN, M.; JOHNSON, K. **Applied Predictive Modeling.** 1. ed. New York: Springer, 2013.
- KURZ, S.; DE GERSEM, H.; GALETZK, A.; KLAEDTKE, A; LIEBSCH, M; LOUKREZIS, D.; RUSSENSCHUCK, S.; SCHMIDT, M. **Hybrid modeling: towards the next level of scientific computing in engineering.** Journal of Mathematics in Industry. v. 12: 8, 2022.
- KUTZ, J. N. **Data-Driven Modeling & Scientific Computation: Methods for Complex Systems & Big Data.** 1. ed. New York: Oxford University Press, 2013.
- LAI, Z.; NAGARAJAIAH, S. **Sparse structural system identification method for nonlinear dynamic systems with hysteresis/inelastic behavior.** Mechanical Systems and Signal Processing. v. 117, p. 813-842, 2019.
- LAROSE, D. T.; LAROSE, C. D. **Discovering knowledge in data: an introduction to data mining.** 2. ed. Hoboken, NJ: John Wiley & Sons, Inc., 2014.

LATHI, B. P. **Signal Processing & Linear Systems.** 1. ed. Carmichael CA; Berkeley Cambridge Press, 1998.

LI, L.; ASLAM, S.; WILEMAN, A.; PERINPANAYAGAM, S. **Digital Twin in Aerospace Industry: A Gentle Introduction.** *IEEE Access.* v. 10, p. 9543-9562, 2022.

LI, Y. **Deep Reinforcement Learning: An Overvie.** ArXiv:1810.06339, 2018.

LIEVEN, N.; EWINS, D. J. **Spatial correlation of mode shapes, the Coordinate Modal Assurance Criterion (COMAC).** Proc. 6th International Modal Analysis Conference, 1988.

LOISEAU, J.-C.; BRUNTON, S. L. **Constrained sparse Galerkin regression.** *Journal of Fluid Mechanics.* v. 838, p. 42-67, 2018.

LORENZ, E. N. **Deterministic nonperiodic flow.** *Journal of Atmospheric Sciences,* v. 20, n. 2, p. 130–141, 1963.

LOVE, P. E. D.; MATTHEWS, J. **The ‘how’ of benefits management for digital technology: from engineering to asset management.** *Automation in Construction.* v. 107: 102930, 2019.

MAGALHÃES, F.; CUNHA, A.; CAETANO, E. **Vibration based structural health monitoring of an arch bridge: from automated OMA to damage detection.** *Mechanical Systems and Signal Processing,* v. 28, p. 212-228, 2012.

McEVILY, A. J. **Failures in inspection procedures: case studies.** *Engineering Failure Analysis.* v. 11, n. 2, p. 167-176, 2004

MENDI, A. F. **A Digital Twin Case Study on Automotive Production Line.** *Sensors.* v. 22, n. 18: 6963, 2022.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. **Foundations of machine learning.** 2. ed. Cambridge, MA: The MIT Press, 2018.

MOKHTARI, F.; IMANPOUR, A. **Hybrid data-driven and physics-based simulation technique for seismic analysis of steel structural systems.** Computers & Structures. v. 295, 2024.

MONTÁNS, F. J.; CHINESTA, F.; GÓMEZ-BOMBARELLI, R.; J. KUTZ, J. N. **Data-driven modeling and learning in science and engineering.** Comptes Rendus Mécanique. v. 347, n. 11, p. 845-855, 2019.

MORADIPOUR, P.; CHAN, T. H. T.; GALLAGE, C. **An improved modal strain energy method for structural damage detection, 2D simulation.** Structural Engineering and Mechanics. v. 54, n. 1, p. 105-119, 2015.

MÜLLER, A. C.; GUIDO, S. **Introduction to Machine Learning with Python: A Guide for Data Scientists.** 1. ed. Sebastopol, CA: O'Reilly Media, 2017.

PANDEY, A. K.; BISWAS, M. **Damage detection in structures using changes in flexibility.** Journal of Sound and Vibration. v. 169, n. 4, p. 3-17, 1994.

PANDEY, A. K.; BISWAS, M.; SAMMAN, M. **Damage detection from changes in curvature mode shape.** Journal of Sound and Vibration. v. 145, n. 2, p. 321 –332, 1991.

PASSERINI, A. **Kernel Methods, multiclass classification and applications to computational molecular biology.** PhD Thesis, Università Degli Studi di Firenze, 2004.

PEDREGOSA, F.; VAROQUAUX, G; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG V; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M. DUCHESNAY, E. **Scikit-learn: machine learning in python.** The Journal of Machine Learning Research. v. 12, p. 2825–2830, 2011.

PENNY, J. E. T.; WILSON, D. A. L.; FRISWELL, M. I. **Damage Location in Structures Using Vibration Data.** in Proc. of the 11th International Modal Analysis Conference, p. 861–867, 1993.

QUINLAN, J. R. **C4.5: Programs for Machine Learning.** 1. ed. San Francisco, CA: Morgan Kaufmann, 1992.

RATCLIFFE, C. P. **Damage detection using a modified laplacian operator on mode shape data.** Journal of Sound and Vibration. v. 204, n. 3, p. 505-517, 1997.

RHEE, J. H.; NGUYEN, H. D.; KIM, M. K.; LIM, Y. M.; KIM, G. **An integrated machine-learning platform for assessing various dynamic responses of steel beams.** Structures. v. 61: 106125, 2024.

RUIZ, D. V.; BRAGANÇA, C. S. C.; PONCETTI, B. L.; BITTENCOURT, T. N.; FUTAI, M. M. **Vibration-based structural damage detection strategy using FRFs and machine learning classifiers.** Structures. v. 59: 105753, 2024.

QI, Q.; TAO, F.; HU, T.; ANWER, N.; LIU, A.; WEI, Y.; WANG, L.; NEE, A. **Enabling technologies and tools for DT.** Journal of Manufacturing Systems. v. 58, parte B, p. 3-21, 2021.

RASCHKA, S. **Model evaluation, model selection, and algorithm selection in machine learning.** ArXiv:1811.12808v3, 2020.

RASHEED, A; SAN, O; KVAMSDAL, T. **Digital Twin: Values, Challenges and Enablers From a Modeling Perspective.** IEEE Access. v. 8, p. 21980- 22012, 2020.

RATHORE, M. M.; SHAH, S. A.; SHUKLA, D.; BENTAFAT, E.; BAKIRAS, S. **The Role of AI, Machine Learning, and Big Data in Digital Twinning: A Systematic Literature Review, Challenges, and Opportunities.** IEEE Access, v. 9, p. 32030–32052, 2021.

RITTO, T. G.; ROCHINHA, F. A. **Digital twin, physics-based model, and machine learning applied to damage detection in structures.** Mechanical Systems and Signal Processing. v. 155: 107614. 2021.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach.** 1 ed. Englewood Cliffs, NJ: Prentice Hall, 1995.

SALAWU, O.S. **Detection of structural damage through changes in frequency: a review.** Engineering Structures. v. 19, n. 9, p. 718-723, 1997.

SAMMAN, M. M.; BISWAS, M. **Vibration testing for nondestructive evaluation of bridges I: Theory.** Journal of Structural Engineering. v. 120, n. 1, p. 269-289, 1994-a.

SAMMAN, M. M.; BISWAS, M. **Vibration testing for nondestructive evaluation of bridges II: Results.** Journal of Structural Engineering. v. 120, n. 1, p. 290-306, 1994-b.

SAN, O.; RASHEED, A.; KVAMSDAL, T. **Hybrid analysis and modeling, eclecticism, and multifidelity computing toward digital twin revolution.** GAMM-Mitteilungen. v. 44, n. 2: e202100007, 2021.

SANTOS, A.; FIGUEIREDO, E.; SILVA, M. F. M.; SALES, C. S.; COSTA, J. C. W. A. **Machine learning algorithms for damage detection: Kernel-based approaches.** Journal of Sound and Vibration. v. 363, p. 584–599, 2016.

SARKER, I. H. **A machine learning based robust prediction model for real-life mobile phone data.** Internet Things. v. 5, p. 180–193, 2019.

SARKER, I. H. **Machine Learning: Algorithms, Real-World Applications and Research Directions.** SN Computer Science. v. 2: 160, 2021.

SCHMIDT, M.; LIPSON, H. **Distilling free-form natural laws from experimental data.** Science. v. 324, n. 5923, p. 81-85, 2009.

SEGOVIA, M.; GARCIA-ALFARO, J. **Design, Modeling and Implementation of Digital Twins.** Sensors. v. 22, n. 14: 5396, 2022.

SHANNON, C. E. **Communication in the presence of noise.** Proceedings of the IRE. v. 37, n. 1, p. 10-21, 1949.

SHawe-Taylor, J.; Cristianini, N. **Kernel Methods for Pattern Analysis.** New York, NY: Cambridge University Press, 2004.

- SHI, Z. Y.; LAW, S. S. **Structural damage localization from modal strain energy change.** Journal of Sound and Vibration. v. 218, n. 5, p. 825-844, 1998.
- SHI, Z. Y.; LAW, S. S.; ZHANG, L. M. **Structural damage detection from modal strain energy change.** Journal of Engineering Mechanics, v. 126, n. 12, p. 1216-1223, 2000.
- SINGH, S.; WEEBER, M; BIRKE, K.-P. **Advancing digital twin implementation: a toolbox for modelling and Simulation.** Procedia CIRP. v. 99, p. 567-572, 2021.
- SIOW, P. Y.; ONG, Z. C.; KHOO, S. Y.; LIM, K.-S. **Hybrid modal-machine learning damage identification approach for beam-like structures.** Journal of Vibration and Control. v. 30, n. 19-20, p. 4286–4303, 2024.
- SKIENA, S. S. **The data science design manual.** 1. ed. Cham, Switzerland: Springer, 2017.
- SMOLA, A. J.; BARLETT, P.; SCHÖLKOPF, B.; SCHUURMANS, D. **Introduction to large margin classifiers.** In SMOLA, A. J.; BARLETT, P.; SCHÖLKOPF, B.; SCHUURMANS, D. (Ed.), *Advances in Large Margin Classifiers*, v. 1, p. 1–28. MIT Press, 1999.
- SMOLA, A. J.; SCHÖLKOPF, B. **Learning with Kernels.** Cambridge, MA: The MIT Press, 2002.
- SOROKINA, M.; SYGLETOS, S.; TURITSYN, S. **Sparse identification for nonlinear optical communication systems: SINO method.** Optics express. v. 24, n. 26, p. 30433-30443, 2016.
- SPARROW, C. **The Lorenz equations: bifurcations, chaos and strange attractors.** 1. ed. New York: Springer, 1982
- SU, W.; BOGDAN, M.; CANDÈS, E. **False discoveries occur early on the lasso path.** The Annals of Statistics, v. 45, n. 5, p. 2133–2150, 2017
- SVENDSEN, B. T; ØISETH, O.; FRØSETH, G. T.; RØNNQUIST, A. **A hybrid structural health monitoring approach for damage detection in steel bridges under simulated**

environmental conditions using numerical and experimental data. Structural Health Monitoring. v. 22, n.1, p. 540-561, 2023.

TANEVSKI, J.; TODOROVSKI, L.; DŽEROSKI, S. **Learning stochastic process-based models of dynamical systems from knowledge and data.** BMC Systems Biology. v. 10: 30, 2016.

TANEVSKI, J.; SIMIDJIEVSKI, N.; TODOROVSKI, L.; DŽEROSKI, S. **Process-Based Modeling and Design of Dynamical Systems.** In: ALTUN, Y.; DAS, K.; MIELIKÄINEN, T.; MALERBA, D.; STEFANOWSKI, J.; READ, J.; ŽITNIK, M.; CECI, M.; DŽEROSKI, S. (eds.). Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2017. Lecture Notes in Computer Science. v. 10536, p. 378-382. Cham, Switzerland: Springer, 2017.

TANG, B; HE, H. **ENN: Extended Nearest Neighbor Method for Pattern Recognition.** IEEE Computational Intelligence Magazine. v.10, n. 3, p. 52-60, 2015.

TAO, F.; ZHANG, M. **Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing.** IEEE Access. v. 5, p. 20418–20427, 2017.

TAO, F.; ZHANG, H.; LIU, A.; NEE, A. Y. C. **Digital Twin in Industry: State-of-the-Art.** IEEE Transactions on Industrial Informatic v. 15, n. 4, p. 2405 – 2415, 2019.

THARWAT, A. **Linear vs. quadratic discriminant analysis classifier: a tutorial.** International Journal of Applied Pattern Recognition. v. 3, n. 2, p. 145-180, 2016

TIBSHIRANI, R. **Regression shrinkage and selection via the lasso.** Journal of the Royal Statistical Society: Series B (Methodological). v. 58, n. 1, p. 267–288, 1996.

TORZONI, M.; TEZZELE, M.; MARIANI, S.; MANZONI, A.; WILLCOX, K. E. **A digital twin framework for civil engineering structures.** Computer Methods in Applied Mechanics and Engineering. v. 418, Part B: 116584, 2024.

VANDERHORN, E.; MAHADEVAN, S. **Digital Twin: Generalization, characterization and implementation.** Decision Support Systems. v. 145: 113524, 2021.

VU, V. T.; THOM, D. V.; TRAN, T. D. **Identification of damage in steel beam by natural frequency using machine learning algorithms.** Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science. v. 238, n. 19, p. 9644-9659, 2024.

WANG, S. Q.; LI, H. J. **Assessment of structural damage using natural frequency changes.** Acta Mechanica Sinica. v. 28, n. 1, p. 118-127, 2012.

WEINBERGER, K. Q.; SAUL, L. K. **Distance metric learning for large margin nearest neighbor classification.** The Journal of Machine Learning Research. v. 10, p. 207-244, 2009.

WETTSCHERECK, D.; AHA, D. W.; MOHRI, T. **A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms.** In: AHA, D. W. (ed.). Lazy learning. Dordrecht: Springer, 1997. p. 273-314.

WICKERT, T. **Theory of Vibration | Equations of Motion.** In: BRAUN, S. G.; EWINS, D. J.; RAO, S. (eds.). Encyclopedia of Vibration. Elsevier, 2001. p. 1324-1332.

WILLMOTT, C. J.; MATSUURA, K. **Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance.** Climate Research. v. 30, n. 1, p. 79–82, 2005.

WORDEN, K.; CROSS, E. J.; BARTHORPE, R. J.; WAGG, D. J. **On digital twins, mirrors, and virtualizations: frameworks for model verification and validation.** ASCE-ASME Journal of Risk and Uncertain in Engineering Systems, Part B: Mechanical Engineering. v. 6, n.3: 030902, 2020.

WU, S.; ZHOU, J.; RUI, S; FEI, Q. **Reformulation of elemental modal strain energy method based on strain modes for structural damage Struct.** Journal: Advances in Structural Engineering. v. 20, n. 6, p. 896–905, 2017.

- YAGER, R. R.; ZADEH, L. A. **An introduction to fuzzy logic applications in intelligent systems.** v. 165. Springer Science & Business Media, 2012.
- YANG, Z.; WANG, L. **Structural damage detection by changes in natural frequencies.** Journal of Intelligent Material Systems and Structures. v. 21, n. 3, p. 309-319, 2010.
- YAZDANPANAH, O.; SEYEDPOOR, S. M.; BENGAR, H. A. **A new damage detection indicator for beams based on mode shape data.** Structural Engineering and Mechanics. v. 53, n. 4, p. 725-744, 2015.
- ZHANG, M. J.; HARMAN, M.; MA, L.; LIU, Y. **Machine Learning Testing: Survey, Landscapes and Horizons.** IEEE Transactions on Software Engineering. v. 48, n.1, p. 1-36, 2022.

APÊNDICE: RESPOSTAS DO *FRAMEWORK DT* DO CLASSIFICADOR SVM

Barra Engastada Discretizada em Dois Elementos Finitos

Tabela 31 – Respostas do *framework DT* do classificador SVM para a segunda mola, considerando a barra engastada discretizada em dois elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído	10% de ruído	
	P (%)	P (%)	P (%)	P (%)	P (%)	DA
0%	85,07	84,67	64,93	51,40	29,87	
1%	85,07	84,87	68,93	51,47	27,20	
2%	82,33	73,93	55,80	48,20	31,33	
3%	80,80	69,00	54,87	49,27	36,07	
4%	84,20	83,67	72,27	60,87	36,60	
5%	84,60	84,27	79,47	65,20	39,40	
6%	83,87	83,27	77,13	60,00	37,13	
7%	80,33	73,60	57,60	49,60	36,87	
8%	83,47	75,07	58,87	52,27	39,40	
9%	86,53	86,27	77,53	65,93	43,47	
10%	87,20	87,00	85,13	70,33	45,60	
11%	86,53	86,40	78,80	67,07	43,27	
12%	82,60	77,27	61,20	53,33	40,87	
13%	81,60	76,60	57,93	52,53	40,20	
14%	84,53	84,33	78,93	68,27	43,33	
15%	84,87	84,80	82,93	71,60	47,00	
16%	84,87	84,47	78,60	68,87	43,80	
17%	81,27	72,53	58,27	52,47	42,87	
18%	85,80	82,73	66,53	60,40	53,13	
19%	89,20	89,33	85,40	75,73	64,07	
20%	91,07	90,93	90,13	85,93	73,40	
21%	92,07	92,00	91,60	90,47	82,40	
22%	92,93	92,87	93,20	92,73	87,67	
23%	93,73	93,73	93,67	93,40	89,87	
24%	94,20	94,20	94,20	93,87	91,87	
25%	94,73	94,53	94,47	94,47	93,93	

Fonte: Autor (2024).

Barra Engastada Discretizada em Quatro Elementos Finitos

Tabela 32 – Respostas do framework DT do classificador SVM para a segunda mola, considerando a barra engastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	85,87	83,67	52,80		29,40		14,73	Dano 5% @4
1%	87,07	82,93	51,80		31,67		15,47	
2%	83,67	62,40	43,33	Íntegra	28,33	Íntegra	14,93	Dano 5% @3
3%	87,20	69,20	50,00		32,93		14,60	
4%	88,80	88,00	61,47		38,07		16,27	
5%	89,20	88,80	63,73		40,20		18,13	
6%	87,73	83,53	60,80		41,60		21,47	
7%	74,73	47,80	45,20		40,60		24,67	
8%	87,27	81,60	61,20		50,73		26,13	
9%	88,53	87,60	72,73		55,27		27,60	
10%	87,67	87,47	76,73		56,80		28,33	
11%	85,67	83,80	67,93		49,87		29,33	
12%	75,73	55,47	50,33		42,67		30,47	
13%	83,07	80,80	63,27		54,07		33,93	
14%	86,67	86,53	76,73		61,33		37,40	
15%	86,20	85,80	78,13		62,53		37,60	
16%	83,87	82,87	70,47		55,20		39,67	
17%	67,33	48,13	49,73		47,67		48,73	
18%	88,40	86,87	71,27		65,13		58,67	
19%	91,67	91,33	86,73		78,53		65,60	
20%	93,87	93,73	91,93		86,73		73,87	
21%	94,47	94,47	94,00		91,87		79,07	
22%	95,00	94,87	94,73		94,33		85,93	
23%	95,53	95,60	95,27		95,20		89,13	
24%	96,13	95,87	95,87		95,80		91,93	
25%	96,33	96,27	96,27		96,13		94,13	

Fonte: Autor (2024).

Barra Engastada Discretizada em Quatro Elementos Finitos

Tabela 33 – Respostas do framework DT do classificador SVM para a terceira mola, considerando a barra engastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	85,87	83,67	52,80	Íntegra	29,40	Íntegra	14,73	Dano 5% @4
1%	83,87	77,47	48,00		27,00		15,87	
2%	78,93	58,20	39,13		29,67		15,13	
3%	80,40	56,53	45,87		33,67		16,73	
4%	89,33	82,87	58,07		40,33		18,47	
5%	90,13	89,07	64,07		40,00		20,13	
6%	90,07	88,87	62,33	Dano 5%		Dano 5%		
7%	90,20	75,07	55,33		43,07		20,47	
8%	84,47	57,07	49,47		40,47		22,73	
9%	91,20	86,33	61,27		39,80		23,87	
10%	91,87	91,40	71,40	Dano 10%		Dano 10%		
11%	92,60	91,80	70,60		44,60		26,27	
12%	91,47	83,33	61,07		48,00		26,87	
13%	58,93	48,53	48,00		46,87		25,33	
14%	91,87	85,73	63,00		45,87		26,13	
15%	92,93	92,47	72,27	Dano 15%		Dano 15%		
16%	92,67	92,40	73,20		40,60		27,07	
17%	92,80	87,87	65,87		49,20		29,60	
18%	78,33	52,60	50,67		54,67		31,40	
19%	91,53	84,33	64,13		53,60		31,33	
20%	92,53	92,13	79,47	Dano 20%		Dano 20%		
21%	92,67	92,53	87,40		49,00		38,13	
22%	92,80	92,87	90,47		46,40		42,20	
23%	93,67	93,53	92,53		58,53		49,27	
24%	93,80	93,73	93,60		66,40		57,13	
25%	94,33	94,20	94,00	Dano 20%		Dano 20%		
					77,73		62,93	
					83,53		67,20	
					87,47		70,67	
					91,27		78,33	
					92,13		81,13	

Fonte: Autor (2024).

Barra Engastada Discretizada em Quatro Elementos Finitos

Tabela 34 – Respostas do framework DT do classificador SVM para a quarta mola, considerando a barra engastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	85,87	83,67	52,80	Íntegra	29,40	Íntegra	14,73	
1%	86,00	84,07	49,13		27,20		18,80	
2%	85,13	70,53	40,13		32,60		22,33	
3%	88,00	67,93	51,33		42,80		23,47	
4%	90,73	89,00	68,47		50,73		26,53	
5%	91,20	91,07	79,33		54,80		27,20	
6%	91,33	91,13	77,27		54,67		26,73	
7%	91,07	86,80	63,60		49,93		30,93	
8%	79,33	51,47	48,07		47,47		32,33	
9%	92,53	91,07	69,80		55,93		37,67	
10%	93,07	93,00	81,47		63,07		38,27	
11%	92,93	92,67	83,07		65,60		40,80	
12%	91,73	90,27	74,07		61,47		39,47	
13%	85,73	60,87	52,20		47,40		35,93	
14%	90,47	85,00	62,27		54,67		35,67	
15%	91,27	91,13	80,20		63,80		37,40	
16%	91,40	91,33	83,87		66,00		36,80	
17%	90,60	90,47	78,60		62,67		38,27	
18%	87,47	72,87	56,33		51,27		47,27	
19%	92,27	84,53	65,47		58,00		54,33	
20%	95,13	94,53	84,27		74,20		61,20	
21%	95,60	95,53	93,40		85,67		70,07	
22%	96,07	95,87	95,53		91,13		74,53	
23%	96,33	96,27	96,07		94,07		85,20	
24%	96,67	96,87	96,87		96,33		89,20	
25%	97,47	97,33	97,07		96,87		92,27	
	Dano 20%			Dano 15%		Dano 10%		Dano 5%
								Dano 15%
								Dano 20%

Fonte: Autor (2024).

Barra Engastada Discretizada em Seis Elementos Finitos

Tabela 35 – Respostas do *framework* DT do classificador SVM para a segunda mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	DA	P (%)	DA	P (%)	DA
	P (%)	P (%)	P (%)					
0%	88,53	85,73	38,73			17,13		11,20
1%	89,40	86,40	37,40			16,93		12,47
2%	87,20	59,27	32,00			18,60		12,87
3%	89,73	75,93	44,40			24,07		13,80
4%	91,53	90,00	52,87			27,80		13,87
5%	91,27	90,60	56,73			28,40		13,47
6%	89,60	85,13	55,20			28,27		12,27
7%	57,00	48,20	38,40			27,47		13,13
8%	91,20	85,87	57,73			37,07		13,00
9%	91,73	90,60	69,13			40,80		12,93
10%	90,93	90,47	69,87			40,87		13,47
11%	88,13	81,33	57,33			37,47		15,27
12%	85,07	63,87	49,47			34,40		16,73
13%	92,53	91,40	67,00			47,20		19,33
14%	93,13	92,33	77,80			48,27		24,93
15%	92,67	89,40	66,67			49,93		30,40
16%	76,40	49,87	49,13			45,13		36,27
17%	94,40	92,40	69,53			59,20		43,87
18%	95,47	95,40	85,20			70,67		49,93
19%	96,13	96,00	93,07			82,13		56,67
20%	96,73	96,60	95,93			88,53		64,53
21%	97,20	96,93	96,73			92,60		69,60
22%	97,53	97,40	96,93			94,53		75,53
23%	97,60	97,73	97,20			96,47		79,27
24%	97,60	97,80	97,40			96,40		83,40
25%	97,80	98,00	97,73			97,27		88,80

Fonte: Autor (2024).

Barra Engastada Discretizada em Seis Elementos Finitos

Tabela 36 – Respostas do *framework* DT do classificador SVM para a terceira mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	88,53	85,73	38,73		17,13		11,20	
1%	88,00	74,53	34,60	Íntegra	16,93	Íntegra	11,93	
2%	68,33	46,27	30,27		19,27		12,67	
3%	88,80	76,33	44,20		23,53		9,53	
4%	91,47	86,67	45,60		25,27		9,93	
5%	91,93	88,00	47,87		24,87		10,07	
6%	88,13	64,87	43,00		26,93		11,60	
7%	87,53	65,40	42,07	Dano 5%	27,07	Dano 5%	12,60	
8%	92,53	84,67	51,67		29,13		13,87	
9%	93,33	90,27	51,80		31,67		13,67	
10%	92,07	81,20	50,60	Dano 10%	31,27		14,67	
11%	49,07	47,73	42,80		33,20		16,07	
12%	91,73	80,47	51,87	Dano 15%	38,67	Dano 15%	17,27	
13%	93,20	90,33	56,33		39,53		18,87	
14%	92,53	88,53	57,00		39,13		19,33	
15%	89,00	66,33	48,60	Dano 20%	37,33	Dano 20%	23,73	
16%	89,13	67,13	54,80		46,13		29,93	
17%	93,80	89,40	69,07		54,33		31,80	
18%	94,80	94,20	80,07		62,80		35,93	
19%	95,00	94,87	86,00		69,67		37,00	
20%	95,47	95,33	89,53	Dano 20%	71,47	Dano 20%	40,13	
21%	95,87	95,87	89,80		74,93		42,53	
22%	96,07	95,93	89,33		76,27		44,20	
23%	95,87	95,60	88,00		76,53		46,73	
24%	95,60	95,40	87,67		73,73		47,07	
25%	95,33	94,67	83,00	Dano 20%	70,60	Dano 20%	46,87	

Fonte: Autor (2024).

Barra Engastada Discretizada em Seis Elementos Finitos

Tabela 37 – Respostas do *framework* DT do classificador SVM para a quarta mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	88,53	85,73	38,73	Íntegra	17,13	Íntegra	11,20	5% @5
1%	87,87	83,80	38,80		22,13		12,47	5% @4
2%	85,20	62,73	33,26		28,20		16,00	
3%	89,33	67,67	49,47		32,27		16,93	
4%	92,00	89,27	58,87		36,67		20,80	
5%	92,80	92,07	64,00	Dano 5%	38,60		23,07	
6%	91,73	84,73	58,00		36,33		22,47	
7%	70,47	49,40	48,80		41,87		26,73	
8%	92,07	86,07	63,53		48,00		25,33	
9%	93,20	92,73	72,60		52,13		28,20	
10%	92,73	93,07	70,73	Dano 10%	50,13		28,60	
11%	90,73	78,87	61,20		44,00		28,53	
12%	83,07	63,87	53,60		46,47		29,20	
13%	92,87	90,27	70,20		52,53		29,67	
14%	94,20	93,87	78,53		57,93		34,80	
15%	94,00	93,67	76,60	Dano 15%	57,20		36,33	
16%	92,87	87,13	64,73		52,07		43,93	
17%	81,80	56,53	52,00		51,93		48,93	
18%	95,60	91,93	70,87		63,07		57,53	
19%	96,60	96,47	84,60		74,53		63,27	
20%	97,07	97,07	93,47	Dano 20%	84,27		70,00	
21%	97,47	97,40	96,20		89,93		73,73	
22%	97,80	97,73	97,27		93,20		82,27	
23%	98,00	98,00	97,80		95,60		84,67	
24%	98,07	98,13	97,93		97,27		88,27	
25%	98,13	98,20	98,13	Dano 20%	97,73		90,47	

Fonte: Autor (2024).

Barra Engastada Discretizada em Seis Elementos Finitos

Tabela 38 – Respostas do *framework* DT do classificador SVM para a quinta mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	DA	5% de ruído	DA	10% de ruído	DA
	P (%)	P (%)	P (%)		P (%)		P (%)	
0%	88,53	85,73	38,73	Íntegra	17,13	Íntegra	11,20	Dano 5%
1%	88,73	80,27	34,60		20,67		14,67	
2%	84,93	58,00	30,53		25,27		15,60	
3%	46,87	47,87	40,33		32,40		16,20	
4%	91,73	82,93	56,40		38,67		17,53	
5%	93,53	91,93	66,00		43,47		19,73	
6%	94,40	93,80	70,60		48,67		21,40	
7%	94,40	92,93	69,53		46,27		24,40	
8%	92,73	78,20	56,27		44,27		24,60	
9%	84,53	56,60	50,20		41,07		24,07	
10%	93,87	88,93	63,53		48,80		27,00	
11%	95,00	94,20	73,80		52,93		30,33	
12%	95,40	94,87	78,13		55,80		29,93	
13%	95,07	93,87	70,80		53,67		29,60	
14%	93,07	76,67	58,53		46,47		29,07	
15%	90,07	64,13	53,07		46,13		30,73	
16%	95,20	92,73	68,73		51,73		30,60	
17%	95,73	95,53	78,87		57,33		33,40	
18%	95,60	95,87	79,27		60,27		35,07	
19%	95,07	93,00	70,93		55,33		41,73	
20%	89,47	73,47	59,07		49,40		47,60	
21%	92,80	76,93	59,53		56,13		53,47	
22%	96,60	95,40	76,73		68,53		59,07	
23%	97,07	97,13	89,07		76,73		66,53	
24%	97,67	97,53	95,00		85,33		69,47	
25%	98,00	98,00	97,00		91,93		76,40	

Fonte: Autor (2024).

Barra Engastada Discretizada em Seis Elementos Finitos

Tabela 39 – Respostas do framework DT do classificador SVM para a sexta mola, considerando a barra engastada discretizada em seis elementos finitos.

DI	sem ruído		1% de ruído		3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA	Dano 5% @5						
0%	88,53	85,73		38,73		17,13		11,20		
1%	87,67	85,27		37,47		15,87		10,67		
2%	86,67	77,73		31,53		17,53		12,33		
3%	77,47	48,33		31,73		26,07		12,87		
4%	90,33	77,93		50,40		33,93		14,33		
5%	94,00	92,60		65,27		41,87		14,20		
6%	94,60	94,53		73,67		45,53		17,87		
7%	94,40	93,27		70,67		47,80		20,47		
8%	92,20	83,67		62,53		47,27		21,73		
9%	66,67	49,53		50,53		44,33		25,80		
10%	92,87	87,07		65,93		54,73		27,80		
11%	95,07	94,40		76,27		57,07		29,20		
12%	95,20	94,87		82,93		58,20		29,80		
13%	94,47	92,93		72,87		55,87		30,67		
14%	89,00	75,67		59,00		49,47		35,20		
15%	90,27	75,60		60,33		53,20		40,07		
16%	94,60	94,13		78,93		64,13		43,73		
17%	95,33	95,20		87,20		71,73		43,07		
18%	94,80	94,27		87,33		71,67		44,23		
19%	92,13	91,07		73,53		60,47		42,00		
20%	65,33	52,13		50,00		48,47		48,80		
21%	93,67	91,87		74,53		66,13		55,87		
22%	95,80	95,67		90,40		78,53		65,33		
23%	96,40	96,53		95,60		88,27		75,40		
24%	96,73	96,80		96,73		94,67		81,80		
25%	97,07	97,07		97,00		96,47		87,60		

Fonte: Autor (2024).

Barra Biengastada Discretizada em Quatro Elementos Finitos

Tabela 40 – Respostas do *framework* DT do classificador SVM para a segunda mola, considerando a barra biengastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	79,53	63,13	29,07		20,73		14,27	
1%	80,73	60,13	29,40	Íntegra	24,40		14,20	
2%	67,47	44,93	38,00		29,40		18,07	
3%	79,20	69,00	44,87		30,13		16,27	
4%	81,80	78,80	50,33		32,73		17,00	
5%	81,53	80,80	51,73		33,20		17,80	
6%	81,60	74,53	49,27		33,80		18,20	
7%	78,20	56,93	46,67		31,07		19,80	
8%	78,67	54,07	41,60		30,87		19,07	
9%	81,33	73,53	48,27		31,27		20,00	
10%	81,87	80,00	50,93		31,93		20,60	
11%	81,73	76,73	47,33		31,20		21,13	
12%	76,47	54,33	40,40		34,93		22,67	
13%	81,20	63,00	49,53		37,47		24,33	
14%	82,73	80,53	59,00		39,33		24,93	
15%	82,40	82,00	60,00		43,73		29,60	
16%	82,27	80,33	57,13		42,07		35,20	
17%	78,13	66,73	50,80	Dano 15%	38,47		38,60	
18%	76,67	56,27	48,60	Dano 10%	45,87		42,33	
19%	82,47	80,33	61,13	Dano 15%	54,27		45,60	
20%	83,67	83,47	73,07	Dano 20%	63,80		53,73	
21%	84,87	84,53	80,27	Dano 20%	72,13		58,67	
22%	85,27	85,13	83,47	Dano 20%	74,87		62,33	
23%	86,27	86,13	85,33	Dano 20%	79,67		66,33	
24%	86,40	86,33	85,80	Dano 20%	83,07		68,93	
25%	86,67	86,33	86,13	Dano 20%	84,53		72,60	

Fonte: Autor (2024).

Barra Biengastada Discretizada em Quatro Elementos Finitos

Tabela 41 – Respostas do framework DT do classificador SVM para a terceira mola, considerando a barra biengastada discretizada em quatro elementos finitos.

DI	sem ruído		1% de ruído		3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA	P (%)	DA	P (%)	DA	P (%)	DA	
0%	79,53	63,13		29,07		20,73		Dano 5% @2	14,27	
1%	41,93	44,40		24,27		19,20			13,00	
2%	77,73	57,60	Dano 5% @1	28,07		18,27		Íntegra	12,93	
3%	73,93	49,27		26,00		17,80			11,13	
4%	71,93	47,40		30,47		18,60			15,53	
5%	75,40	68,07		32,40		19,60			15,27	
6%	73,07	67,80		31,07		21,80			15,13	
7%	62,13	44,13		31,93		26,53			15,00	
8%	68,33	56,20		39,87		29,00			16,07	
9%	75,40	71,33		47,07		31,80			18,40	
10%	76,53	75,33		47,33		33,47			19,00	
11%	77,20	74,87		46,87		28,33			19,53	
12%	76,73	70,40		38,60		27,27			20,07	
13%	73,67	45,73		34,60		31,00			22,67	
14%	74,07	54,80		43,07		31,80			23,27	
15%	80,87	73,73		48,33		33,73			29,47	
16%	81,13	79,47		53,53		34,93			31,27	
17%	80,60	76,80		50,60		33,47			34,67	
18%	78,87	57,20		43,20		36,40			38,27	
19%	77,53	59,07		47,33		45,53			40,73	
20%	79,60	77,20		58,60		52,40			45,20	
21%	80,27	79,93		67,87		58,93			47,40	
22%	80,27	80,07		73,87		65,07			53,60	
23%	80,13	80,07		77,13		68,20			55,33	
24%	80,13	80,00		78,60		72,07			57,40	
25%	79,73	79,53		79,40		74,33			60,80	

Fonte: Autor (2024).

Barra Biengastada Discretizada em Quatro Elementos Finitos

Tabela 42 – Respostas do *framework* DT do classificador SVM para a quarta mola, considerando a barra biengastada discretizada em quatro elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)	DA
0%	79,53	63,13	29,07	Íntegra	20,73	5% @2	14,27	
1%	70,53	41,87	27,20	Dano 5% @1	18,53	Íntegra	14,67	
2%	77,47	57,13	24,80		18,40		12,93	
3%	62,40	36,60	22,60	Dano 5% @3	16,07	Íntegra	14,27	
4%	67,40	48,73	28,67	Dano 5% @2	19,00	Dano 5% @3	11,13	
5%	75,33	72,60	40,33		25,73		11,73	
6%	75,13	72,47	46,47	Dano 5% @4	26,47	Dano 5% @5	14,93	
7%	66,87	50,40	39,80		30,27		14,53	
8%	72,07	64,33	38,93	Dano 10% @1	26,73	Dano 10% @2	13,73	
9%	77,53	76,93	51,20		31,00		15,87	
10%	79,73	79,33	55,47	Dano 10% @3	36,20	Dano 10% @4	18,80	
11%	80,60	74,07	50,67		36,60		18,40	
12%	64,13	50,47	41,47	Dano 15% @1	33,00	Dano 15% @2	21,27	
13%	76,80	66,20	42,20		29,67		24,93	
14%	79,87	77,67	46,07	Dano 15% @3	31,60	Dano 15% @4	31,60	
15%	80,07	70,73	43,20		32,80		35,93	
16%	57,00	40,47	36,13	Dano 20% @1	36,73	Dano 20% @2	38,87	
17%	76,40	55,27	48,20		45,00		43,53	
18%	81,47	76,07	58,67	Dano 20% @3	52,40	Dano 20% @4	48,73	
19%	82,40	82,07	68,00		58,93		51,53	
20%	83,13	83,07	75,60	65,87		56,47		
21%	82,87	82,73	79,87	Dano 20% @5	72,87	Dano 20% @6	58,40	
22%	83,33	83,20	81,93		75,93		59,93	
23%	82,60	82,53	82,00	Dano 20% @7	77,20	Dano 20% @8	62,60	
24%	82,20	81,93	81,80		76,60		64,27	
25%	80,40	80,27	79,27	77,40		62,53		

Fonte: Autor (2024).

Barra Biengastada Discretizada em Seis Elementos Finitos

Tabela 43 – Respostas do *framework* DT do classificador SVM para a segunda mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído	
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)
	81,13	70,40	22,27	Íntegra	14,60	Dano 5% @4	8,20
0%	81,13	70,40	22,27	Íntegra	14,60	Dano 5% @4	8,20
1%	80,93	72,00	22,00		12,20		8,20
2%	79,27	53,87	21,86		13,60		7,87
3%	80,27	57,40	30,07		14,60		8,67
4%	82,27	75,13	34,27		15,27		8,93
5%	83,20	80,93	39,47		17,67		9,67
6%	82,80	74,60	37,53	Dano 5%	20,33	Dano 5% @4	11,00
7%	78,13	52,53	32,27		21,67		12,00
8%	82,67	64,93	40,93		26,00		15,33
9%	84,13	81,00	45,53		26,93		16,53
10%	84,13	81,47	47,80	Dano 10%	29,13	Dano 10% @4	17,67
11%	82,40	69,20	42,73		28,13		20,67
12%	76,93	45,87	39,13		26,20		24,27
13%	83,73	72,40	45,87		28,87		25,80
14%	84,67	82,07	49,27		31,27		27,27
15%	84,13	80,27	50,13	Dano 15%	31,87	Dano 15% @4	31,80
16%	82,47	65,13	42,60		38,93		35,20
17%	83,07	56,33	48,20		46,40		40,27
18%	86,67	78,53	61,93		55,93		44,33
19%	88,47	87,00	69,73		60,40		47,87
20%	88,87	88,80	77,13	Dano 20%	67,20	Dano 20% @4	48,93
21%	89,73	89,67	82,47		72,20		52,47
22%	90,47	90,13	86,27		78,20		58,67
23%	90,73	90,60	88,67		82,80		60,80
24%	90,60	90,53	89,53		84,07		64,53
25%	91,47	91,33	91,20		86,80		68,13

Fonte: Autor (2024).

Barra Biengastada Discretizada em Seis Elementos Finitos

Tabela 44 – Respostas do framework DT do classificador SVM para a terceira mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído		5% de ruído		10% de ruído		
	P (%)	P (%)	DA	P (%)	DA	P (%)	DA	P (%)	DA
0%	81,13	70,40		22,27		14,60		8,20	
1%	79,67	60,87	Íntegra	22,12	Íntegra	14,00	Dano 5% @ 4	7,93	
2%	75,87	41,93		19,33		12,20	Dano 5% @ 4	7,67	
3%	71,27	45,20		23,60		11,60		8,87	
4%	82,07	66,60		27,33		12,13		10,67	
5%	83,40	75,40		28,40		12,87		11,87	
6%	83,93	72,47	Dano 5%	28,67	Dano 5%	15,60		13,80	
7%	80,33	52,20		29,27		17,20		14,93	
8%	82,80	57,07		33,13		18,87		15,13	
9%	84,33	74,27		36,47		19,87		16,60	
10%	85,40	78,93		38,13		22,07		20,80	
11%	85,00	71,20	Dano 10%	38,00	Dano 10%	21,87	Dano 10%	23,07	
12%	77,67	51,13		37,47		24,80	Dano 15%	23,47	
13%	82,47	61,53		38,27		24,27		27,07	
14%	83,33	76,33		38,40		26,93	Dano 15%	28,80	
15%	83,27	77,47		41,67		28,80		31,47	
16%	82,73	67,27	Dano 15%	38,73	Dano 15%	37,40	Dano 15%	34,07	
17%	65,80	45,13		42,60		41,73		36,93	
18%	83,47	67,33		53,87		48,00		40,27	
19%	85,27	80,20		61,93		53,87		42,40	
20%	85,67	85,13		68,27		60,60		46,00	
21%	86,53	86,33	Dano 20%	74,20	Dano 20%	64,40	Dano 20%	49,60	
22%	87,13	86,93		79,47		68,87		53,27	
23%	87,33	87,27		83,20		71,93		56,47	
24%	87,87	87,67		85,47		77,63		57,53	
25%	88,60	88,40		87,33		79,93		59,73	

Fonte: Autor (2024).

Barra Biengastada Discretizada em Seis Elementos Finitos

Tabela 45 – Respostas do *framework* DT do classificador SVM para a quarta mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído		1% de ruído		3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA	Dano 5%						
0%	81,13	70,40		22,27		14,60		8,20		
1%	80,47	65,80		22,20		15,40		8,80		
2%	78,53	51,33	Íntegra	25,33		17,80		9,20		
3%	81,40	47,73		31,80		18,67		10,93		
4%	85,60	71,00		34,53		20,87		12,53		
5%	85,67	80,80		39,00		20,40		12,40		
6%	83,87	78,07		37,87		22,00		13,73		
7%	81,27	65,40		37,07		23,87		15,80		
8%	65,13	44,43		38,87		25,80		17,73		
9%	86,00	69,33		41,20		28,20		20,07		
10%	86,07	80,87		46,20		28,20		20,60		
11%	86,00	81,87		43,47		27,67		25,53		
12%	84,60	71,27		44,47		29,13		26,07		
13%	75,80	49,40		40,13		28,87		28,73		
14%	84,27	61,47		42,60		26,47		30,87		
15%	85,60	78,73		46,80		30,33		34,60		
16%	85,27	82,73		46,60		33,00		37,00		
17%	84,80	75,40		45,47		37,27		40,87		
18%	81,87	52,80		43,73		47,00		44,80		
19%	86,73	66,87		53,13		50,67		47,13		
20%	88,87	83,80		64,80		57,47		50,20		
21%	89,53	88,93		72,20		61,27		53,20		
22%	90,00	89,67		79,67		69,20		54,87		
23%	90,67	90,53		84,07		74,60		59,33		
24%	90,67	90,53		88,53		77,53		63,47		
25%	91,27	91,13		89,60		81,73		64,27		

Fonte: Autor (2024).

Barra Biengastada Discretizada em Seis Elementos Finitos

Tabela 46 – Respostas do *framework* DT do classificador SVM para a quinta mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído	1% de ruído	3% de ruído	5% de ruído		10% de ruído		Dano 5% @4	Dano 5% @4
	P (%)	P (%)	P (%)	DA	P (%)	DA	P (%)		
0%	81,13	70,40	22,27		14,60		8,20		
1%	80,80	70,13	20,53		12,27		7,87		
2%	79,53	58,33	20,00		15,93		7,53		
3%	73,60	42,20	28,20		16,00		7,53		
4%	81,40	67,13	35,60		17,73		6,93		
5%	83,67	80,07	39,47		18,47		8,00		
6%	83,60	82,93	39,67		18,00		9,33		
7%	82,67	77,20	36,87		19,00		11,20		
8%	80,40	59,33	33,00		21,47		9,67		
9%	73,00	48,27	37,33		24,13		10,47		
10%	80,07	71,13	42,53		26,13		10,20		
11%	82,53	77,53	45,27		26,53		14,13		
12%	82,87	80,80	47,20		26,40		14,00		
13%	82,60	78,20	44,33		26,00		18,20		
14%	78,93	62,67	37,93		24,27		18,73		
15%	76,93	51,93	39,47		26,07		21,40		
16%	82,73	76,00	45,87		29,93		21,87		
17%	84,00	82,67	48,67		27,53		25,20		
18%	84,53	83,53	49,07		27,33		29,67		
19%	82,93	75,93	42,73		33,47		32,80		
20%	78,47	56,20	37,47		39,40		34,07		
21%	80,93	60,60	49,60		45,00		39,67		
22%	84,40	79,47	57,67		49,20		41,87		
23%	84,87	84,00	67,87		53,40		39,73		
24%	86,13	85,80	73,20		60,33		44,33		
25%	85,87	85,47	77,33		63,40		44,73		

Fonte: Autor (2024).

Barra Biengastada Discretizada em Seis Elementos Finitos

Tabela 47 – Respostas do *framework* DT do classificador SVM para a sexta mola, considerando a barra biengastada discretizada em seis elementos finitos.

DI	sem ruído		1% de ruído		3% de ruído		5% de ruído		10% de ruído	
	P (%)	P (%)	DA							
0%	81,13	70,40	Íntegra	22,27	Íntegra	14,60	Dano 5% @4	8,20		
1%	78,80	53,13		21,93		14,00		6,87		
2%	82,27	74,73	Dano 5% @5	36,00	Dano 5% @5	17,07		7,67		
3%	81,73	71,33		35,47		18,40		7,27		
4%	81,80	72,00	Dano 5%	38,47		23,33		10,13		
5%	83,60	82,73		52,53	Dano 5%	30,73		12,67		
6%	83,20	77,53		50,87		31,80	Dano 5%		16,07	Dano 5% @4
7%	54,80	34,93		34,13		27,67		18,67		
8%	82,20	78,87		53,87		38,13		19,87		
9%	85,47	85,27		68,73		48,80		24,73		
10%	86,20	85,87		75,27		55,27		26,27		
11%	85,80	85,13		68,80		53,80		29,60		
12%	76,47	58,20		51,93		45,07		27,73		
13%	84,60	82,87		64,07		52,73		30,40		
14%	86,13	86,27		79,73		64,40		35,60		
15%	86,73	86,47		83,73		70,33		36,40		
16%	85,67	85,20		74,93		63,00		36,07		
17%	74,60	58,80		50,67		46,73		44,60	Dano 15%	
18%	87,80	86,07		72,00		62,40		54,93		
19%	89,80	89,33		86,53		78,00		64,87		
20%	90,87	90,53		90,53		85,20		74,60		
21%	91,33	91,33		91,33		90,53		80,73		
22%	92,20	92,00		92,07		91,93		85,73		
23%	92,53	92,60		92,80		92,40		89,20		
24%	93,00	92,87		92,87		92,53		91,20		
25%	93,13	93,07		93,20		93,13		91,67	Dano 20%	

Fonte: Autor (2024).