



UNIVERSIDADE FEDERAL DO AMAZONAS  
FACULDADE DE TECNOLOGIA  
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CLAUDIO DUARTE SILVA FILHO

PROGNÓSTICO DE FALHAS EM ROLAMENTOS BASEADO EM MODELOS  
GRANULARES EVOLUTIVOS

Manaus  
Agosto de 2025



UNIVERSIDADE FEDERAL DO AMAZONAS  
FACULDADE DE TECNOLOGIA  
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CLAUDIO DUARTE SILVA FILHO

PROGNÓSTICO DE FALHAS EM ROLAMENTOS BASEADO EM MODELOS  
GRANULARES EVOLUTIVOS

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, PPGEE, da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientadores: Iury Valente de Bessa

Raoni de Paula Lourenço

Manaus  
Agosto de 2025

#### Ficha Catalográfica

Elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

---

S586p Silva Filho, Claudio Duarte  
Prognóstico de falhas em rolamentos baseado em modelos  
granulares evolutivos / Claudio Duarte Silva Filho. - 2025.  
114 f. : il., color. ; 31 cm.

Orientador(a): Iury Valente de Bessa.

Coorientador(a): Raoni de Paula Lourenço.

Dissertação (mestrado) - Universidade Federal do Amazonas,  
Programa de Pós-Graduação em Engenharia Elétrica, Manaus,  
2025.

1. Manutenção Preditiva. 2. Prognóstico. 3. Modelos evolutivos. 4.  
Granularidade. 5. Adaptação. I. Bessa, Iury Valente de. II.  
Lourenço, Raoni de Paula. III. Universidade Federal do Amazonas.  
Programa de Pós-Graduação em Engenharia Elétrica. IV. Título

---



Ministério da Educação  
Universidade Federal do Amazonas  
Coordenação do Programa de Pós-Graduação em Engenharia Elétrica

## **FOLHA DE APROVAÇÃO**

Poder Executivo Ministério da Educação  
Universidade Federal do Amazonas  
Faculdade de Tecnologia  
Programa de Pós-graduação em Engenharia Elétrica

Pós-Graduação em Engenharia Elétrica. Av. General Rodrigo Octávio Jordão Ramos, nº 3.000 - Campus Universitário, Setor Norte - Coroado, Pavilhão do CETELI. Fone/Fax (92) 99271-8954 Ramal:2607. E-mail: ppgee@ufam.edu.br

CLAUDIO DUARTE DA SILVA FILHO

### **PROGNÓSTICO DE FALHAS EM ROLAMENTOS BASEADO EM MODELOS GRANULARES EVOLUTIVOS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica na área de concentração Controle e Automação de Sistemas.

Aprovada em 12 de agosto de 2025.

**BANCA EXAMINADORA**  
Prof. Dr. Iury Valente de Bessa- Presidente  
Prof. Dr. Waldir Sabino da Silva Júnior - Membro Titular 1 - Interno  
Dr. Luiz Alberto Queiroz Cordovil Júnior - Membro Titular 2 - Externo

Manaus, 31 de julho de 2025.



Documento assinado eletronicamente por **Luiz Alberto Queiroz Cordovil Junior**, **Usuário Externo**, em 14/08/2025, às 09:52, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).

---



Documento assinado eletronicamente por **Iury Valente de Bessa**, **Professor do Magistério Superior**, em 14/08/2025, às 12:58, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).

---



Documento assinado eletronicamente por **Waldir Sabino da Silva Júnior**, **Coordenador**, em 14/08/2025, às 20:19, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).

---



A autenticidade deste documento pode ser conferida no site [https://sei.ufam.edu.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://sei.ufam.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **2716212** e o código CRC **F5F81802**.

---

Av. General Rodrigo Octávio Jordão Ramos, nº 3.000 - Bairro Coroado Campus Universitário, Setor Norte  
- Telefone: 99271-8954  
CEP 69080-900 Manaus/AM - Pavilhão do CETELI. E-mail: [ppgee@ufam.edu.br](mailto:ppgee@ufam.edu.br)

Referência: Processo nº 23105.032332/2025-47

SEI nº 2716212

*Dedico este trabalho de mestrado  
aos meus pais, Claudio Duarte  
Silva e Gioconda Berenice de  
Almeida Corrêa, por todo o  
apoio incondicional, amor e  
paciência. Cada passo desta  
jornada foi possível porque vocês  
sempre estiveram ao meu lado,  
celebrando minhas conquistas e  
me incentivando a cada novo  
degrau.*

# Agradecimentos

A elaboração e a conclusão desta dissertação foram possíveis graças ao apoio incondicional e à colaboração de inúmeras pessoas e instituições. A elas, expresso a minha mais profunda gratidão.

Agradeço, em primeiro lugar, aos meus orientadores, Professor Iury Valente de Bessa e Professor Raoni de Paula Lourenço, pela orientação precisa, paciência e pelos valiosos insights que moldaram este trabalho.

Um agradecimento especial ao meu pai e professor, Claudio Duarte Silva, e a minha mãe, Gioconda Berenice de Almeida Corrêa, pelo apoio constante e por terem me incentivado em cada etapa.

Agradeço também à Coordenação do Programa de Pós-graduação em Engenharia Elétrica e a todos os professores da UFAM que contribuíram para a minha formação. Aos meus colegas de curso, sou grato pela convivência intelectual e pelo ambiente colaborativo que compartilhamos.

Não poderia deixar de mencionar meu grande amigo Keven Soares, que foi fundamental no início desta jornada, desencadeando o processo que me trouxe ao mestrado.

Por fim, dedico minha alegria pela conclusão deste percurso à minha família. A vocês, que conviveram com as tensões, frustrações e o desânimo inerentes à vida acadêmica, o meu mais sincero obrigado. O apoio de vocês foi o alicerce para esta conquista.

Resumo da Dissertação apresentada à UFAM como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

## PROGNÓSTICO DE FALHAS EM ROLAMENTOS BASEADO EM MODELOS GRANULARES EVOLUTIVOS

Claudio Duarte Silva Filho

Orientadores: Iury Valente de Bessa

Raoni de Paula Lourenço

Programa: Pós-Graduação em Engenharia Elétrica

A manutenção preditiva é fundamental para garantir a continuidade operacional na indústria, sendo um dos pilares da Indústria 4.0, onde sensores e sistemas cibernéticos permitem o monitoramento contínuo de equipamentos. Nesse contexto, o Prognóstico e Gerenciamento de Saúde surge como uma abordagem para predição de falhas por meio da estimativa da vida útil restante de máquinas usando algoritmos baseados em modelos físicos, estatísticos e técnicas de inteligência computacional capazes de analisar parâmetros operacionais e calcular o tempo de vida até a falha. Contudo, os métodos tradicionais de predição enfrentam limitações, como a dependência de grandes volumes de dados para treinamento, estrutura e parâmetros fixos, alto custo computacional e baixa interpretabilidade. Essas restrições tornam tais métodos ineficazes em cenários reais, onde há escassez de dados históricos, mudanças operacionais bruscas e a necessidade de modelos computacionalmente eficientes e explicáveis. Para superar essas limitações, esta pesquisa propõe o Preditor AutoCloud, um modelo granular e evolutivo que aprende de forma contínua e autônoma, ajustando seus parâmetros e expandindo ou reduzindo sua estrutura conforme as características dos dados. Além disso, sua abordagem baseada em granularidade permite gerar predições mais interpretáveis e adaptáveis às variações do

sistema. Os testes preliminares do Preditor AutoCloud demonstram sua capacidade de contornar os desafios apresentados, oferecendo previsões mais adaptáveis e interpretáveis.

**Palavras-chave:** Manutenção Preditiva, prognóstico, modelos evolutivos, granularidade, adaptação, interpretabilidade.

Abstract of Dissertation presented to UFAM as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering

## FAILURE PROGNOSIS IN ROLLING BEARINGS WITH EVOLVING GRANULAR MODELS

Claudio Duarte Silva Filho

Advisors: Iury Valente de Bessa

Raoni de Paula Lourenço

Department: Postgraduate in Electrical Engineering

Predictive maintenance is essential to ensure operational continuity in the industry, being one of the pillars of Industry 4.0, where sensors and cyber-physical systems enable continuous equipment monitoring. In this context, Prognostics and Health Management (PHM) emerges as an approach for failure prediction through the estimation of the Remaining Useful Life (RUL) of machines using algorithms based on physical models, statistical methods, and computational intelligence techniques capable of analyzing operational parameters and calculating the time to failure. However, traditional prediction methods face limitations such as dependency on large volumes of training data, fixed structures and parameters, high computational cost, and low interpretability. These constraints make such methods ineffective in real-world scenarios, where there is a lack of historical data, abrupt operational changes, and the need for computationally efficient and explainable models. To overcome these limitations, this research proposes the AutoCloud Predictor, a granular and evolutionary model that learns continuously and autonomously, adjusting its parameters and expanding or reducing its structure according to the characteristics of the data. Additionally, its granularity-based approach enables the generation of more interpretable and adaptable predictions to system variations. Preliminary tests of the AutoCloud Predictor demonstrate its ability to address the challenges

presented, offering more adaptable and interpretable predictions. However, its accuracy has not yet reached the expected level, highlighting the need to explore potential improvements addressed throughout this study.

**Keywords:** Predictive maintenance, prognostics, evolving models, granularity, adaptation, interpretability.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Objetivos da Dissertação . . . . .	3
1.1.1	Objetivo geral . . . . .	3
1.1.2	Objetivos específicos . . . . .	3
1.2	Justificativa . . . . .	4
1.3	Organização do trabalho . . . . .	4
<b>2</b>	<b>Fundamentos Teóricos</b>	<b>6</b>
2.1	Prognóstico e Gerenciamento da Saúde . . . . .	6
2.2	Processamento de Sinais . . . . .	8
2.2.1	<i>Variational Mode Decomposition</i> . . . . .	10
2.3	<i>Recurrent Neural Networks</i> . . . . .	14
2.3.1	Rede Neural Recorrente de Aprendizado Recursivo . . . . .	14
2.4	Sistemas Inteligentes . . . . .	15
2.4.1	Sistemas Adaptativos . . . . .	16
2.4.2	Sistemas Evolutivos . . . . .	17
2.5	Sistemas Evolutivos e Computação Granular . . . . .	18
2.5.1	Computação Granular . . . . .	18
2.5.2	Sistemas Granulares Evolutivos . . . . .	20
2.6	Explicabilidade e Interpretabilidade de Modelos de Predição . . . . .	28
<b>3</b>	<b>Revisão da Literatura</b>	<b>30</b>
3.1	Bibliografias Analisadas . . . . .	30
3.2	Bases de Dados Utilizadas . . . . .	32
3.3	Pré-processamento de Dados . . . . .	34

3.4	Domínio de processamento . . . . .	35
3.5	Características e/ou Indicador de Saúde . . . . .	37
3.6	Modelo de Prognóstico . . . . .	38
3.7	Discussão Acerca dos Trabalhos Apresentados . . . . .	40
3.7.1	Objetivo Principal . . . . .	40
3.7.2	Dependência de Dados . . . . .	44
3.7.3	Adaptabilidade . . . . .	45
3.7.4	Evolutividade . . . . .	45
3.7.5	Interpretabilidade . . . . .	45
3.8	Conclusões Acerca da Revisão da Literatura . . . . .	46
3.9	Prognóstico Evolutivo de Falhas . . . . .	47
3.10	Modelo proposto . . . . .	48
<b>4</b>	<b>Metodologia Proposta</b>	<b>51</b>
4.1	Aquisição de Dados . . . . .	51
4.1.1	Rolamentos . . . . .	52
4.2	Pré-processamento de Dados . . . . .	54
4.2.1	Entropia da Banda de Frequência - <i>Frequency Band Entropy</i> (FBE) . . . . .	54
4.2.2	<i>Variational Mode Decomposition</i> (VMD) Otimizado pela FBE	57
4.3	Da Extração à Seleção de Características . . . . .	61
4.3.1	Características Extraídas . . . . .	61
4.3.2	Critério de Seleção de Características . . . . .	61
4.4	Criação do Indicador de Saúde . . . . .	63
4.4.1	Análise da Componente Principal . . . . .	64
4.4.2	Estágio de Saúde . . . . .	65
4.5	Criação do Preditor AutoCloud . . . . .	65
4.5.1	Preditor AutoCloud de Consequente Linear . . . . .	68
4.5.2	Preditor AutoCloud de Consequente Não-Linear . . . . .	68
4.6	Quantificação de Incerteza . . . . .	71
4.6.1	Incerteza por Parâmetros Granulares . . . . .	71
4.6.2	Incerteza por Saídas Granulares . . . . .	72
4.7	Prognóstico e Predição de <i>Remaining Useful Life</i> (RUL) . . . . .	72

4.7.1	Quantificação de Incerteza em Predição de RUL . . . . .	73
4.8	Quantificação dos Erros do Modelo . . . . .	75
<b>5</b>	<b>Resultados</b>	<b>76</b>
5.1	Base de Dados . . . . .	77
5.2	Pré-processamento de Dados . . . . .	79
5.3	Extração e Seleção de Características . . . . .	80
5.4	Criação dos Indicadores de Saúde e Estágio de Degradação . . . . .	82
5.5	Predição de RUL com o Preditor Autocloud . . . . .	84
5.5.1	Otimização de Hiperparâmetros . . . . .	85
5.5.2	Preditor AutoCloud (P.AutoCloud): <i>Recursive Last Squares with Variable-direction Forgetting</i> (RLSVDF) . . . . .	86
5.5.3	<i>Evolving Granular Neural Network with Radial Basis Function</i> (EGNN-RBF) . . . . .	89
5.5.4	<i>Evolving Granular Recurrent Neural Network</i> (EGRNN) . . . . .	92
5.6	Avaliação entre Modelos . . . . .	95
<b>6</b>	<b>Conclusão</b>	<b>99</b>
6.1	Publicações Derivadas . . . . .	100
6.1.1	Prognóstico de falhas em rolamentos baseado em modelos granulares evolutivos. . . . .	100
6.1.2	<i>Incremental learning-based rolling bearing prognostics with granular uncertainty quantification.</i> . . . . .	101
6.1.3	<i>Prognostics of Rolling Bearings based on Neuro-Fuzzy Evolving Models.</i> . . . . .	101
6.1.4	<i>Prognostics of Rolling Bearings based on Neuro-Fuzzy Evolving Models.</i> . . . . .	102

# Lista de Figuras

2.1	Panorama do PHM. . . . .	7
2.2	Etapas de realização do prognóstico. . . . .	7
2.3	Sinal ou Série temporal genérica. . . . .	9
2.4	Resultado da aplicação do VMD. . . . .	13
2.5	Esquema ilustrativo de uma <i>Recurrent Neural Networks</i> (RNN). . . . .	14
2.6	Classificação de alguns SEIs. Adaptado de: Lemos et al. (2013) . . . . .	17
2.7	Granulação de dados. . . . .	19
2.8	Granulação de dados baseado em TEDA. . . . .	22
3.1	Quantidade de citações por bases de dados utilizadas. . . . .	33
3.2	Frequência das técnicas de pré-processamento por classificação. . . . .	35
3.3	Frequência dos domínios de processamento por classificação. . . . .	37
3.4	Frequência das abordagens para extração de características por classificação. . . . .	38
3.5	Frequência dos modelos de prognóstico características por classificação. . . . .	39
3.6	Quantidade de abordagens de otimização e hibridização. . . . .	42
4.1	Etapas de realização do prognóstico. . . . .	51
4.2	Rolamento e seus componentes em vista explodida. . . . .	52
4.3	Motor elétrico em vista cortada. Adaptado de (Thomas, 2024) . . . . .	53
4.4	Falhas em: A - anel externo, B - anel interno, C - elemento rolante, D - gaiola. Adaptado de (Dario, 2024; James, 2024; NSK, 2024; NTN, 2024). . . . .	53
4.5	Fluxo de dados através do preditor AutoCloud de consequente <i>Radial Basis Function</i> (RBF). . . . .	69
4.6	Evolving recurrent neural network framework. . . . .	70

4.7	Incerteza da predição de RUL pela propagação da estimativa de erro. . . . .	75
5.1	Rolamento 2_4 - Teste do início ao fim da vida útil. . . . .	78
5.2	Rolamento 1_1 - Sinal bruto e sinal filtrado. . . . .	79
5.3	Rolamento 1_5 - Sinal bruto e sinal filtrado. . . . .	80
5.4	Rolamento 2_4 - Sinal bruto e sinal filtrado. . . . .	80
5.5	Características selecionadas dos Rolamentos 1_1, 1_5 e 2_4. . . . .	81
5.6	Métricas de avaliação de Características: Rolamento 1_5. . . . .	82
5.7	Indicadores de saúde e estágio de degradação dos Rolamentos 1_1, 1_5 e 2_4. . . . .	83
5.8	Granulação dos indicadores de saúde dos Rolamentos 1_4, 1_5 e 2_4. . . . .	87
5.9	Predição de RUL granular dos Rolamentos 1_4, 1_5 e 2_4. . . . .	88
5.10	RMSE da predição de RUL dos Rolamentos 1_4, 1_5 e 2_4. . . . .	89
5.11	Granulação dos indicadores de saúde dos Rolamentos 1_4, 1_5 e 2_4. . . . .	90
5.12	Predição de RUL granular dos Rolamentos 1_4, 1_5 e 2_4. . . . .	91
5.13	RMSE da predição de RUL dos Rolamentos 1_4, 1_5 e 2_4. . . . .	92
5.14	Granulação dos indicadores de estágio e saúde dos Rolamentos 1_4, 1_5 e 2_4. . . . .	93
5.15	Predição de RUL granular dos Rolamentos 1_4, 1_5 e 2_4. . . . .	94
5.16	RMSE da predição de RUL dos Rolamentos 1_4, 1_5 e 2_4. . . . .	95

# Lista de Tabelas

2.1	Sistemas Granulares Evolutivos e Métodos de Granulação. . . . .	21
3.1	Filtros para delimitação da busca na plataforma Scopus. . . . .	30
3.2	Referências analisadas e informações características - Parte 1. . . . .	31
3.3	Referências analisadas e informações características - Parte 2. . . . .	32
3.4	Frequência de amostragem de dados das bases utilizadas. . . . .	33
3.5	Condições operacionais das bases de dados utilizadas. . . . .	34
3.6	Classificação do método de pré-processamento de acordo com o autor	35
3.7	Classificação do domínio de processamento de acordo com o autor. . .	36
3.8	Classificação do método de extração de características de acordo com o autor. . . . .	38
3.9	Classificação do modelo de prognóstico de acordo com o autor. . . . .	39
3.10	Aspectos de discussão dos trabalhos pesquisados. . . . .	40
3.11	Objetivos dos trabalhos de acordo com o autor. . . . .	40
3.12	Publicações acerca do prognóstico evolutivo de falhas. . . . .	48
4.1	Características extraídas no domínio do tempo. . . . .	62
4.2	Características extraídas no domínio da frequência. . . . .	63
5.1	Base de dados de rolamentos XJTU. . . . .	77
5.2	Otimização de hiperparâmetros: Espaços de busca. . . . .	85
5.3	Preditor AutoCloud: RLSVDF - Otimização de hiperparâmetros. . .	86
5.4	EGNN-RBF - Otimização de hiperparâmetros. . . . .	89
5.5	EGRNN - Otimização de hiperparâmetros. . . . .	92
5.6	Comparação entre os valores de <i>Root Mean Square Error</i> (RMSE) para as previsões dos modelos. . . . .	96

5.7	Comparação entre as quantidades de grânulos formadas para cada modelo. . . . .	97
-----	--	----

# Lista de Acrônimos

- ADMM** *Alternate Direction Method of Multipliers*. 11
- ANFIS** *Adaptive Neuro-Fuzzy Inference System*. 16
- ANN** *Artificial Neural Networks*. 2
- BPTT** *Back-propagation Through Time*. 14
- CG** *Computação Granular*. 18, 20
- CNN** *Convolutional Neural Networks*. 2, 37, 43, 44, 47, 100
- DBSCAN** *Density-based spatial clustering of applications with noise*. 18
- DE** *Differential Evolution*. 44
- DENFIS** *Dynamic Evolving Neuro-Fuzzy Inference System*. 18
- DL** *Deep Learning*. 8, 37–39, 43, 44, 46, 47, 64
- DRL** *Deep Reinforcement Learning*. 43
- DTR** *Decision Tree Regressor*. 2
- EBeTS** *Error Based Evolving Takagi-Sugeno Model*. 48
- ED** *Estágio de Degradação*. 65, 83, 84, 95, 99, 102
- EEFIG** *Evolving Ellipsoidal Fuzzy Information Granules*. 48
- EGNN-RBF** *Evolving Granular Neural Network with Radial Basis Function*. 69, 96, 97

**EGNN-RBF** *Evolving Granular Neural Network with Radial Basis Function*. xiii, xvi, 69, 89, 92, 97–99

**EGRNN** *Evolving Granular Recurrent Neural Network*. 70

**EGRNN** *Evolving Granular Recurrent Neural Network*. xiii, xvi, 70, 92, 96–100

**EM** *Evolving Models*. 47, 48, 76

**EMD** *Empirical Mode Decomposition*. 42, 44, 46

**ES** Estágio de Saúde. 65, 66

**eTS** *Evolving Takagi-Sugeno*. 18

**FBE** *Frequency Band Entropy*. xii, 54, 56, 57, 59, 60, 78

**FFT** *Fast Fourier Transform*. 8, 12

**FMEA** *Failure Modes and Effects Analysis*. 7

**FP** Filtro de Partículas. 47

**FV** Fim de Vida. 72, 73, 84, 86, 87, 90, 92, 93, 96, 97

**GI** Grânulos de Informação. 18, 19, 21, 25, 29, 48, 49, 70, 71, 76, 84, 85, 88, 90, 91, 93, 94, 99

**GRU** *Gated Recurrent Unit*. 100

**IA** Inteligência Artificial. 2, 3, 8, 28

**IC** Intervalo de Confiança. 73

**IED** Indicador de Estágio de Degradação. 65, 66, 77, 82–84, 92–94, 97–99, 102

**IMF** *Intrinsic Mode Functions*. 12, 54, 59

**IS** Indicador de Saúde. 8, 31, 37, 40, 43, 44, 46, 63–66, 69, 72, 73, 77, 83, 84, 86–88, 90, 92–94, 97–99, 101, 102

**ISOMAP** *Isometric Feature Mapping*. 44

**LSTM** *Long Short Term Memory*. 43, 47, 100

**MAPE** *Mean Absolute Percentage Error*. 75, 85, 86, 89, 95

**MGE** *Modelo Granular Evolutivo*. 20

**ML** *Machine Learning*. 2, 38

**MSCATN** *Multi-feature Spaces Cross Adaption Transformer Network*. 43

**MSTFN** *Multi-state Temporal Frequency Network*. 43

**ND** *Nível de Degradação*. 8

**NF** *Neuro Fuzzy*. 2

**NN** *Neural Networks*. 14, 18, 76

**P.AutoCloud** *Preditor AutoCloud*. 66, 68–70, 73, 74, 100, 101

**P.AutoCloud** *Preditor AutoCloud*. xiii, 69, 76, 84, 86, 98, 99

**PCA** *Principal Component Analysis*. 64, 65, 83

**PDRI** *Physics Degradation Rate-informed*. 43

**PDS** *Processamento Digital de Sinais*. 9, 10

**PF-DE** *Particle Filter and Differential Evolution*. 47

**PHM** *Prognostics and Health Management*. 1, 5–8

**PI-BDDN** *Physics-Informed Bayes Deep Dual network PI-BDDN*. 42

**PJG** *Principle of Justifiable Granularity*. 19, 71, 76, 85, 98, 101

**PSO** *Particle Swarm Optimization*. 42

**QRNN** *Quasi-Recurrent Neural Network*. 43

**RBF** *Radial Basis Function*. xiv, 49, 68–70, 76, 84, 89, 90, 97, 99, 101

**RFLO** *Random Feedback Local Online*. 14, 15

**RLRNN** *Recursive Learning Recurrent Neural Networks*. 15, 70, 84, 92, 96, 99, 100

**RLS** *Recursive Least Squares*. 27, 48, 49, 66, 72, 100

**RLSVDF** *Recursive Last Squares with Variable-direction Forgetting*. xiii, xvi, 49, 68, 69, 76, 84, 86, 88, 89, 92, 96, 99, 101

**RMSE** *Root Mean Square Error*. xvi, 75, 88, 92, 95, 96

**RNN** *Recurrent Neural Networks*. 14

**RNN** *Recurrent Neural Networks*. xiv, 14, 15, 49, 68, 70, 76, 94, 95, 102

**RUL** *Remaining Useful Life*. 1

**RUL** *Remaining Useful Life*. xii, xiii, xv, 1–4, 7, 8, 16, 32, 36–39, 43, 45–49, 61, 64–66, 72–77, 84–92, 94–96, 98, 99, 102

**SA** *Sistemas Adaptativos*. 16, 17

**SE** *Sistemas Evolutivos*. 16–18, 20

**SGE** *Sistema Granular Evolutivo*. 20, 21

**SHAP** *SHapley Additive exPlanations*. 46

**SI** *Sistemas Inteligentes*. 16

**STFT** *Short Time Fourier Transform*. 8, 54–56, 59, 60

**SVM** *Support Vector Machine*. 2

**TDNN** *Time Delay Neural Network*. 43

**TEDA** *Typicality and Eccentricity Data Analytics*. 21, 23, 24, 26, 49, 65, 66, 68, 76, 99, 100

**TL-TSRF** *Transfer Learning-based Time Series Recursive Forecasting*. 43

**TW** *Transformada Wavelet*. 8

**VMD** *Variational Mode Decomposition.* 9

**VMD** *Variational Mode Decomposition.* xii, xiv, 10, 12, 13, 46, 54, 57–59, 78

# Capítulo 1

## Introdução

A manutenção de equipamentos é crucial para promover a continuidade operacional e segurança industrial. No contexto da Indústria 4.0, a integração entre dispositivos físicos e digitais permite a comunicação direta entre máquinas e sistemas cibernéticos criando um ecossistema interconectado capaz de centralizar informações acerca de cada ativo em um parque fabril, proporcionando o monitoramento e a tomada de decisão em tempo real, e contribuindo para uma manutenção dinâmica e proativa.

Assim, as tecnologias contemporâneas viabilizam a abordagem preditiva de manutenção, caracterizada pela antecipação das falhas de um equipamento baseada na análise contínua de seus parâmetros de funcionamento. Nesse âmbito, o conceito de Prognóstico e Gerenciamento de Saúde, ou *Prognostics and Health Management* (PHM) em inglês, surge como uma abordagem sistemática de monitoramento do ciclo de vida de um equipamento, objetivando reduzir/eliminar inspeções e manutenções periódicas mediante monitoramento preciso, detecção de falhas incipientes e predição de falhas iminentes, (Kalgren et al., 2006).

O prognóstico é a etapa do PHM responsável por estimar de forma confiável e precisa a Vida Útil Restante, ou *Remaining Useful Life* (RUL) em inglês, de um equipamento/dispositivo, acompanhando e analisando a progressão de seu estado de saúde/degradação, dentro de um intervalo de confiança associado, (Kalgren et al., 2006). Nesse sentido, o prognóstico pode potencialmente prevenir danos críticos, otimizar o tempo médio entre falhas, além reduzir custos de manutenção.

A RUL é estimada por algoritmos fundamentados em modelos físicos, esta-

tísticos ou inteligências computacionais através da predição do instante de tempo imediatamente anterior à ocorrência de uma falha em um equipamento ou um de seus componentes, a partir de seus dados operacionais. Contudo, a precisão do valor calculado pode ser afetada pelo aumento da complexidade nos dados sequenciais (normalmente não-lineares), variações e incertezas no tempo, além da dificuldade de criar um indicador que meça o nível de saúde/degradação a partir dos dados.

Diante disso, diversas ferramentas computacionais como *Machine Learning* (ML) e Inteligência Artificial (IA) vêm sendo desenvolvidas e/ou otimizadas em pesquisas científicas acerca do prognósticos de falhas em máquinas/equipamentos como turbinas a gás, fresadoras, baterias, rolamentos entre outros, (Lei et al., 2018; Ramezani et al., 2019).

Em todo caso, as ferramentas de inteligência computacional sugeridas objetivam aprender o padrão de desgaste de um equipamento, estabelecendo uma aproximação funcional entre o tempo e respectivo nível de saúde/degradação. *Artificial Neural Networks* (ANN), *Support Vector Machine* (SVM) e *Decision Tree Regressor* (DTR) são apenas alguns exemplos de técnicas utilizadas para aprender a relação entre nível de degradação e "tempo de vida", conforme os registros de (Polverino et al., 2023).

Entretanto, mesmo sendo soluções potencialmente viáveis para cálculos de prognóstico, os algoritmos apresentam diversas limitações e empecilhos num cenário fabril real. Autores como (Bagri et al., 2024; Kan et al., 2015; Ramezani et al., 2019) apresentam revisões da literatura acerca dos algoritmos utilizados para cálculo de RUL pontuando suas respectivas limitações. Por exemplo, a precisão de SVMs e DTRs dependem de hiperprâmetros especificamente, além de demandar longos treinamentos, e ANNs como *Convolutional Neural Networks* (CNN), demandam grandes quantidades de dados pré-existentes, além prover dados de opacos de difícil explicação.

Simultaneamente, algoritmos de abordagens mais sofisticadas, como modelos híbridos, buscam mitigar algumas das limitações apresentadas ao combinar modelos diferentes. Redes Neuro Fuzzy (NF), por exemplo, aliam a capacidade de aprendizado das ANNs à capacidade de lidar com incertezas da lógica Fuzzy, criando modelos adaptáveis às variações dos dados e explicáveis graças às regras linguísticas

(Kan et al., 2015), porém ainda pecam por demandarem uma base de dados para treinamento extensa e por exigirem um custo computacional elevado.

No geral, as desvantagens dos algoritmos de predição de abordagem convencional relacionam-se à modelagem dependente de grandes quantidade de dados, às estacionariedades estrutural e paramétrica, e à pouca ou nenhuma interpretabilidade dos resultados. A primeira implica na necessidade de um histórico de dados pré-existente suficientemente abrangente para o treinamento de um modelo prévio. A segunda desvantagem implica na incapacidade de lidar com perturbações pontuais e/ou mudanças operacionais, demanda por hiperparametrização manual, custo computacional fixo e demanda por retreinamentos periódicos. E a terceira implica na ausência de lastro explicativo capaz de descrever as relações entre os dados operacionais fornecidos e predições de RUL realizadas.

Considerando um cenário industrial marcado pela falta de histórico de dados, por constantes mudanças/perturbações nas condições operacionais e pela tomada de decisões devidamente fundamentadas, nota-se a necessidade de um modelo alternativo fundamentados na independência de dados, aprendizado e mutação contínuas e interpretabilidade. O primeiro fundamento mitiga a necessidade de dados e etapa de treinamento de modelo. O segundo promove a assimilação de novas informações e padrões contidos nos dados consecutivos, desencadeando o ajuste de parâmetros e alterações estruturais para contemplar novas circunstâncias operacionais. E o terceiro garante predições de RUL transparentes e explicáveis. Dessa forma, modelos de IA Evolutivos se destacam por suprir as demandas citadas.

## **1.1 Objetivos da Dissertação**

### **1.1.1 Objetivo geral**

Desenvolver um modelo de inteligência artificial evolutivo capaz de realizar o prognóstico de falhas em equipamentos com predições de RUL precisas e confiáveis.

### **1.1.2 Objetivos específicos**

Este trabalho apresenta os seguintes objetivos específicos:

1. Desenvolver um modelo de inteligência artificial:
  - que não dependa de dados para treinamento de um modelo prévio.
  - que exclua a necessidade de armazenamento de dados.
  - versátil e mutável capaz de se adequar às mudanças operacionais e perturbações nos dados.
  - com mecanismo de autoajuste de parâmetros e estruturas desencadeado pela assimilação contínua de informações contidas nos dados.
  - com mecanismo de associação de estágio de degradação à predição de RUL
  - com mecanismo de transparência capaz de explicar relações entre dados e predições de RUL
2. Analisar a capacidade de aprendizado a partir do zero do modelo.
3. Analisar e comparar a performance do modelo com a de outros modelos diante da mesma base de dados.

## 1.2 Justificativa

A justificativa para o desenvolvimento deste trabalho baseia-se na demanda por soluções que integrem a capacidade preditiva de RUL em equipamentos industriais no âmbito da Indústria 4.0, promovendo a segurança e a continuidade operacional num cenário caracterizado pela ausência e escassez de histórico de dados acerca do funcionamento de equipamentos e pela mudança das suas circunstâncias operacionais.

Além disso, dada a baixa disponibilidade de publicações científicas que utilizem Modelos Evolutivos para predição de prognóstico de falhas em equipamentos, vê-se a oportunidade de explorá-los nesse contexto e verificar sua viabilidade.

## 1.3 Organização do trabalho

Este trabalho está organizado segundo a seguinte divisão de capítulos:

- O Capítulo 1 contextualiza o tema pesquisado, prognóstico de falhas em equipamentos com modelos computacionais para predição de vida útil restante, e define os objetivos gerais e específicos almejados.
- O Capítulo 2 apresenta a fundamentação teórica das principais ferramentas e dispositivos/equipamentos usados no desenvolvimento deste trabalho: PHM, rolamentos, processamento de sinais, redes neurais artificiais, sistemas evolutivos, computação granular e sistemas granulares evolutivos.
- O Capítulo 3 discorre sobre as principais publicações recentes acerca do prognóstico de falhas em equipamentos, pontuando peculiaridades de cada etapa, desde a aquisição de dados ao cálculo da vida útil restante por meio de modelos computacionais.
- O Capítulo 4 define a metodologia de desenvolvimento desta pesquisa, apresentando a base de dados utilizada, o método de mapeamento de falhas, as ferramentas de processamento de sinal, o processo de extração e definição características, a criação de um indicador de saúde e o algoritmo de predição proposto.
- O Capítulo 5 expõe os resultados alcançados pelo algoritmo proposto, comparando-os aos resultados atingidos por outros algoritmos.
- O Capítulo 6 conclui esta obra, apontando oportunidades de melhoria e sugestões de trabalhos futuros.

# Capítulo 2

## Fundamentos Teóricos

Este capítulo fornece fundamentos e definições acerca dos conhecimentos teóricos utilizados para o desenvolvimento desta pesquisa. A Seção 2.1 expõe os conceitos, etapas e metodologias acerca do Prognóstico e Gerenciamento de Saúde de equipamentos. A Seção 4.1.1 apresenta detalhadamente aspectos construtivos, funcionamento e possíveis falhas em rolamentos. A Seção 2.2 introduz o conceito de processamento de sinais orientados ao prognóstico de falhas e apresenta uma de suas abordagens. A Seção 2.4 fala acerca de sistemas inteligentes e das características que os classificam em adaptativos ou evolutivos. Finalmente, Seção 2.5 discorre acerca de sistemas inteligentes de características simultaneamente evolutivas e e granulares pontuando sobre um dos métodos de granulação de dados.

### 2.1 Prognóstico e Gerenciamento da Saúde

O PHM é um paradigma baseado em computação fundamentado no conhecimento físico, informações e dados sobre a operação e conservação de máquinas e equipamentos cujo objetivo detectar é o desgaste de componentes, detectar e diagnosticar falhas e prever a vida útil, viabilizando o gerenciamento proativo da manutenção, (Zio, 2016, 2022)

Seu funcionamento se dá a partir da simbiose entre o meio físico e o virtual, onde sensores para medição de parâmetros de funcionamento, aliados a uma interface computacional, viabilizam a coleta e análise de dados acerca da operação de um equipamento, permitindo o monitoramento de condição, o acompanhamento do

desgaste e o prognóstico de falhas através de cálculos de RUL. A Figura 2.1 ilustra um panorama acerca do PHM.

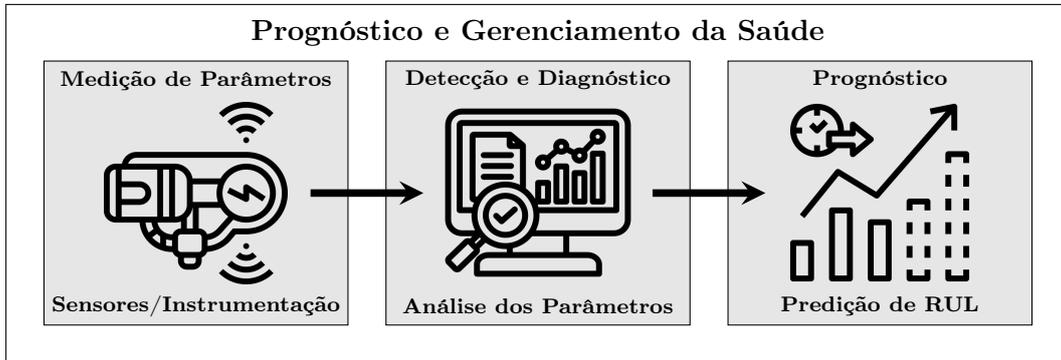


Figura 2.1: Panorama do PHM.

Os resultados da elaboração do PHM são utilizados para apoiar decisões de manutenção baseada no monitoramento da condição, antevendo falhas e promovendo operações eficientes, confiáveis e seguras. Para isso, normas como a ISO 13381-1:2015 *Condition monitoring and diagnostics of machines - Prognostics* definem direcionamentos para a implementação do modelo. Resumidamente, o PHM pode ser alcançado seguindo as três etapas expostas na Figura 2.2.



Figura 2.2: Etapas de realização do prognóstico.

A detecção, primeira etapa, consiste na identificação de anomalias operacionais de um equipamento/componente. A partir do estabelecimento dos valores normais de referência para os parâmetros de funcionamento, é possível identificar detectar anomalias mediante discrepância observadas entre os valores observados e os de referência. Para isso, usam-se metodologias como a IEC 60812-2018 *Análise dos Modos de Falha e Seus Efeitos*, ou *Failure Modes and Effects Analysis* (FMEA) em inglês, capazes de auxiliar no processo de mapeamento de falhas, definições de valores operacionais e criticidade.

O diagnóstico, segunda etapa, é a classificação da falha observada no equipamento/componente. Verificam-se os valores de seus parâmetros e então determina-se a natureza da falha e seu nível de severidade. Ferramentas de processamento de sinal como a *Fast Fourier Transform* (FFT), *Short Time Fourier Transform* (STFT) e Transformada Wavelet (TW) são comumente usadas em sinais de vibração para diagnosticar falhas em máquinas rotativas (Strömbergsson et al., 2020).

O prognóstico, terceira etapa, consiste no cálculo de um tempo de vida útil restante e a confiabilidade associada. Dado um modo de falha, seus respectivos parâmetros de medição são usados para calcular a progressão da degradação em instantes futuros até um limite de tolerância. O período entre um instante de referência e o instante em que se atinge o limite de degradação o cálculo da RUL e respectivo intervalo de confiança (Kalgren et al., 2006; Kan et al., 2015).

Entre as abordagens de prognóstico, para o cálculo do Indicador de Saúde (IS), ou Nível de Degradação (ND), a literatura registra métodos baseados em indicadores estatísticos, baseados no comportamento físico dos dados, e indicadores virtuais, criados a partir da fusão de múltiplos indicadores estatísticos, conforme (Bagri et al., 2024; Ramezani et al., 2019), além de indicadores artificiais de pouca explicabilidade criados com algoritmos de *Deep Learning* (DL) (Cheng et al., 2022b; Xu et al., 2023).

Já para os cálculos de predição, são utilizados modelos físicos, criados a partir da modelagem matemática baseada no modo de falha, modelos estatísticos, baseados em conhecimentos empíricos e probabilidades, e modelos de IA, capazes de aprender e reconhecer padrões automaticamente, (Lei et al., 2018).

## 2.2 Processamento de Sinais

O PHM de equipamentos é realizado através da observação, análise e manipulação de sequências de dados medidos e registrados consecutivamente e cronologicamente ao longo de um intervalo de tempo, comumente denominadas séries temporais ou sinais. A Figura 2.3 ilustra um sinal genérico onde cada ponto azul corresponde a um valor  $y(t)$  de um parâmetro de funcionamento qualquer medido num instante de tempo  $t$ .

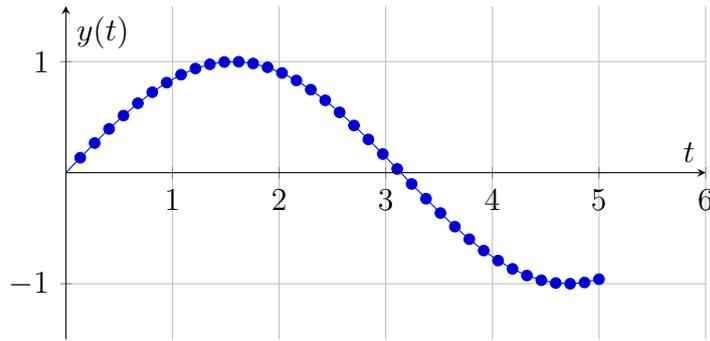


Figura 2.3: Sinal ou Série temporal genérica.

Contudo, valores dos sinais de medições coletados de equipamentos em ambientes reais carregam consigo, além das informações inerentes ao próprio equipamento, variações, ruídos e perturbações do ambiente. Conseqüentemente, muitas vezes as informações de fato relevantes estão diluídas no sinal.

No contexto real, onde máquinas e sistemas são constituídos de diversos dispositivos, entre os quais rolamentos e engrenagens são frequentes, o sinal de um parâmetro medido é resultado da soma das contribuições de todos os dispositivos, além das interferências citadas. Assim, a identificação padrões associados aos respectivos dispositivos é praticamente impossível.

Nesse sentido, métodos de tratamento de dados, como o Processamento Digital de Sinais (PDS), surgem como ferramentas filtragem e remoção das irrelevâncias contidas num sinal, além de dividi-lo em componentes associadas a diferentes dispositivos ou perturbações.

O PDS é o campo da engenharia dedicado ao estudo e desenvolvimento de métodos de análise, transformação e extração de informações de sinais. Através ferramentas matemáticas, estatísticas e computacionais para modelagem, síntese, recuperação e aprendizado, e aplicações em diversas áreas como comunicação, imagem, e sensores etc, essas técnicas permitem a visualização de características intrínsecas do sinal muitas vezes previamente imperceptíveis, (Moura, 2009).

Para fins de prognóstico de máquinas, entre as ferramentas de processamento de sinais frequentemente utilizada nos últimos anos, o *Variational Mode Decomposition* (VMD) destaca-se por sua capacidade de decompor um sinal em sub-sinais associados às suas diferentes componentes.

### 2.2.1 Variational Mode Decomposition

De forma geral, o VMD é uma técnica de decomposição de sinais que divide uma série temporal em vários modos com propriedades de dispersão específicas que, quando somados, compõem a série temporal original.

De forma mais técnica, o VMD é uma técnica de PDS não recursiva e quasi-ortogonal multi-escala baseada no domínio da frequência. O cerne da decomposição é a construção e solução de problemas variacionais, que combina filtros de Wiener para remoção de ruído, transformada de Hilbert para solucionar o espectro marginal e a introdução de um método multiplicador direcional alternativo para alcançar um problema irrestrito (Dragomiretskiy and Zosso, 2014) (Chen et al., 2022b).

Dado um sinal  $x(t) = (x_1, x_2, \dots, x_N)$ , o método VMD possibilita sua decomposição em um número  $K$  de modos  $u_k(t) = (u_{k_1}, u_{k_2}, \dots, u_{k_N})$ , com larguras de banda limitadas e características de esparsidade particulares. (Dragomiretskiy and Zosso, 2014) propôs a decomposição pelo seguinte esquema sequencial:

- Para cada modo  $u_k$ , computar o sinal analítico associado por meio da transformada de Hilbert para obter um espectro de frequência unilateral.
- Para cada modo, deslocar o espectro de frequência do modo para a “banda base”, misturando com um exponencial sintonizado para a respectiva frequência central estimada.
- A largura de banda é estimada através da suavidade gaussiana  $H^1$  do sinal demodulado, ou seja, a norma  $L^2$  ao quadrado do gradiente.

O problema variacional restrito resultante segue:

$$\min_{\{u_k\}, \{\omega_k\}} \left\{ \sum_{k=1}^K \left\| \partial_t \left[ \left( \delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \quad \text{s.t.} \quad \sum_{k=1}^K u_k = x(t) \quad (2.1)$$

Onde  $\{u_k\} = \{u_1, u_2, \dots, u_k\}$  representa  $K$  modos,  $\{\omega_k\} = \{\omega_1, \omega_2, \dots, \omega_k\}$  representa suas frequências centrais correspondentes e  $x(t)$  é o sinal original. Naturalmente,  $\sum_{k=1}^K u_k = x(t)$  é interpretado como o somatório de todos os modos, sendo também a restrição para a resolução do problema .

Para lidar eficientemente com a restrição, os autores combinam o método de penalidade quadrática e os multiplicadores de Lagrange, culminando no uso de um Lagrangiano aumentado, conforme a equação a seguir:

$$\begin{aligned}
L(\{u_k\}, \{\omega_k\}, \lambda) = & \alpha \left\| \partial_t \left[ \left( \delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \\
& + \left\| f(t) - \sum_{k=1}^K u_k(t) \right\|_2^2 \\
& + \left\langle \lambda(t), f(t) - \sum_{k=1}^K u_k(t) \right\rangle
\end{aligned} \tag{2.2}$$

Onde  $\alpha$  é o segundo fator de penalidade,  $\lambda$  é o multiplicador de Lagrange.

A solução do problema de minimização 2.1 é o ponto de sela do Lagrangiano aumentado  $L$  em uma sequência de sub-otimizações iterativas chamada *Alternate Direction Method of Multipliers* (ADMM), conforme o Algoritmo 1:

---

**Algoritmo 1** ADMM: Conceito de otimização para VMD

---

1: Inicializar  $\{u_k^1\}, \{\omega_k^1\}, \{\lambda^1\}, n \leftarrow 0$

2: **repita**

3:  $n \leftarrow n + 1$

4: **para**  $k = 1 : K$  **do faça**

5:     Atualizar  $u_k$  :

6:

$$u_k^{(n+1)} \leftarrow \arg_{u_k} \min L(\{u_{i < k}^{n+1}\}, \{u_{i \geq k}^n\}, \{\omega_i^n\}, \{\lambda^n\}) \tag{2.3}$$

7: **fim para**

8: **para**  $k = 1 : K$  **do faça**

9:     Atualizar  $\omega_k$  :

10:

$$\omega_k^{(n+1)} \leftarrow \arg_{\omega_k} \min L(\{u_i^{n+1}\}, \{\omega_{i < k}^{n+1}\}, \{\omega_{i \geq k}^n\}, \{\lambda^n\}) \tag{2.4}$$

11: **fim para**

12:     Dual ascent:

13:

$$\lambda^{n+1} \leftarrow \lambda^n + \tau(X - \sum_k u_k^{n+1}) \tag{2.5}$$

14: **até** Convergência:  $\sum_k \|u_k^{n+1} - u_k^n\|_2^2 / \|u_k^n\|_2^2 < \epsilon$

---

Para garantir a otimização do problema descrito dor 2.1 faz-se necessário calcular modos  $u_k$  e frequências centrais  $\omega_k$  que minimizem seu custo. Os modos e as frequências centrais são atualizadas pelas equações a seguir:

$$\hat{u}_k^{n+1}(w) = \frac{\hat{X}(w) - \sum_{i \neq k} \hat{u}_i(w) + \frac{\hat{\lambda}(w)}{2}}{1 + 2\alpha(w - \omega_k)^2} \tag{2.6}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty w |\hat{u}_k(w)|^2 dw}{\int_0^\infty |\hat{u}_k(w)|^2 dw} \quad (2.7)$$

onde  $\hat{X}(w)$ ,  $\hat{u}_i(w)$ ,  $\hat{\lambda}(w)$  e  $\hat{u}_k^{n+1}(w)$  representam as FFTs de  $x(t)$ ,  $u_i(t)$ ,  $\lambda(t)$  e  $u_k^{n+1}(w)$ , respectivamente.

Substituindo os resultados das sub-otimizações calculados por 2.6 e 2.7 no Algoritmo 1, e otimizando diretamente no domínio de Fourier, obtêm-se o algoritmo completo para a decomposição dos modos variacionais através do Algoritmo 2:

---

**Algoritmo 2** Otimização completa para VMD

---

1: Inicializar  $\{\hat{u}_k^1\}$ ,  $\{\omega_k^1\}$ ,  $\hat{\lambda}^1$ ,  $n \leftarrow 0$

2: **repita**

3:      $n \leftarrow n + 1$

4:     **para**  $k = 1 : K$  **faça**

5:         Atualizar  $\hat{u}_k(\omega)$  para todo  $\omega \geq 0$ :

6:

$$\hat{u}_k^{n+1}(\omega) \leftarrow \frac{\hat{X}(\omega) - \sum_{i < k} \hat{u}_i^{n+1}(\omega) - \sum_{i > k} \hat{u}_i^n(\omega) + \frac{\hat{\lambda}^n(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k^n)^2} \quad (2.8)$$

7:         Atualizar  $\omega_k$ :

8:

$$\omega_k^{n+1} \leftarrow \frac{\int_0^\infty \omega |\hat{u}_k^{n+1}(\omega)|^2 d\omega}{\int_0^\infty |\hat{u}_k^{n+1}(\omega)|^2 d\omega} \quad (2.9)$$

9:     **fim para**

10:     Dual ascent para todo  $\omega \geq 0$ :

11:

$$\hat{\lambda}^{n+1}(\omega) \leftarrow \hat{\lambda}^n(\omega) + \tau \left( \hat{X}(\omega) - \sum_k \hat{u}_k^{n+1}(\omega) \right) \quad (2.10)$$

12: **até** Convergência:  $\sum_k \frac{\|\hat{u}_k^{n+1} - \hat{u}_k^n\|_2^2}{\|\hat{u}_k^n\|_2^2} < \epsilon$

---

O resultado do algoritmo implica na decomposição de um sinal  $x$  em um número  $k$  de modos  $u_k$ , também chamadas de *Intrinsic Mode Functions* (IMF).

Baseado na restrição da equação (2.1), adotaremos a seguinte notação da equação (2.11) para um sinal  $x$  decomposto em  $K$  modos pelo VMD. O pseudo código para cálculo do VMD pode ser observado no Algoritmo 3.

$$x = \sum_{k=1}^K u_k = (u_1 + u_2 + \dots + u_K) \quad (2.11)$$

A Figura 2.4 mostra um sinal submetido ao VMD para divisão de um sinal em quatro modos ( $k = 4$ ). Do lado esquerdo vemos o sinal original e suas respectivas subcomponentes enquanto do lado direito vemos a Transformada de Fourier dos respectivos modos. Podemos também observar a decomposição do sinal ao redor de

---

**Algoritmo 3** Decomposição de Modo Variacional (VMD)
 

---

- 1: **Inicialização**  $\{u_k^1\}, \{\omega_k^1\}, \hat{\lambda}^1$
  - 2:  $n = 0$
  - 3: **enquanto**  $\left(\sum_{k=1}^K \frac{\|\hat{u}_k^{n+1}(\omega) - \hat{u}_k^n(\omega)\|_2^2}{\|\hat{u}_k^n(\omega)\|_2^2}\right) < \epsilon$  **faça**
  - 4:      $n = n + 1$
  - 5:     **para**  $k = 1 \dots K$  **faça**
  - 6:         Atualizar  $u_k$ :  $u_k^{n+1} = \arg \min_{u_k} L(\{u_i^{n+1}\}_{i < k}, \{u_i^{n+1}\}_{i \geq k}, \{\omega_i^n\}, \hat{\lambda}^n)$
  - 7:     **fim para**
  - 8:     **para**  $k = 1 \dots K$  **faça**
  - 9:         Atualizar  $\omega_k$ :  $\omega_k^{n+1} = \arg \min_{\omega_k} L(\{u_i^{n+1}\}_{i < k}, \{u_i^{n+1}\}_{i \geq k}, \{\omega_i^{n+1}\}, \hat{\lambda}^n)$
  - 10:    **fim para**
  - 11:    Atualizar  $\hat{\lambda}$ :  $\hat{\lambda}^{n+1} = \hat{\lambda}^n + \tau \left(X - \sum_{k=1}^K u_k^{n+1}\right)$
  - 12: **fim enquanto**
  - 13: **Retornar**  $\{u_k\}$
- 

frequências centrais  $\omega_k$ .

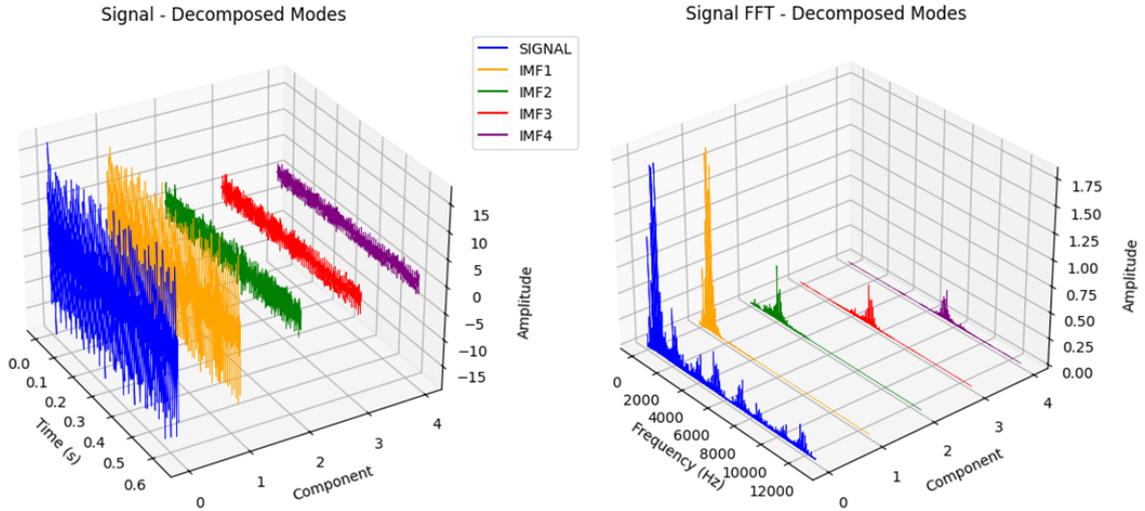


Figura 2.4: Resultado da aplicação do VMD.

O VMD é uma técnica eficaz para decomposição de sinais complexos e componentes individuais intrínsecas. Pode-se dizer que a filtragem de sinais se destaca como sua principal utilização, facilitando trabalhos de predição de vida útil restante em equipamentos industriais e o diagnóstico de doenças em sinais biomédicos como eletrocardiograma (ECG) e eletroencefalograma (EEG), (Wang et al., 2024) e (Pradhan, 2020).

## 2.3 Recurrent Neural Networks

*Recurrent Neural Networks* (RNNs) são uma família de *Neural Networks* (NN) para o processamento de dados sequenciais (ou séries temporais) (Bengio et al., 2017). Ao contrário das NNs tradicionais de *feedforward*, as RNNs possuem um *loop* de *feedback* que permite lembrar as entradas passadas, tornando-as ideais para tarefas onde a ordem dos elementos é importante.

A Figura 2.5 ilustra uma RNN genérica, na qual uma amostra de dados dependente do tempo  $X_h$  fornece a entrada para uma camada oculta conectada recursivamente, composta por  $L$  unidades descritas pelo vetor de atividade  $h_h$ . Essa atividade é lida para formar uma saída dependente do tempo  $Y_h$ . A RNN tem entradas parametrizadas por meio de cada camada, com suas matrizes de pesos  $U, V$  e  $W$  (associados às camadas de entrada, oculta e de saída, respectivamente).

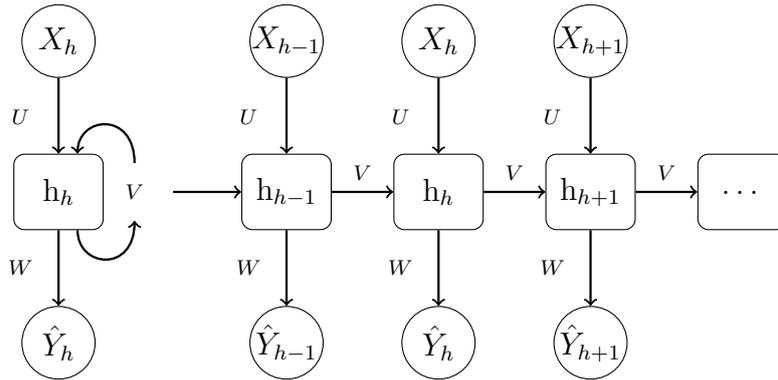


Figura 2.5: Esquema ilustrativo de uma RNN.

### 2.3.1 Rede Neural Recorrente de Aprendizado Recursivo

Normalmente, os pesos  $U, V$  e  $W$  são ajustados durante uma etapa de treinamento offline e estático, com gradientes calculados pela técnica *Back-propagation Through Time* (BPTT), o que pode ser visto como um mecanismo de aprendizado. No entanto, para utilizar uma RNN em cenários dinâmicos (que necessitem de retreinamentos periódicos) seu mecanismo de aprendizado deve ser ajustado para suportar aprendizado recursivo incremental.

Para lidar com isso, o *Random Feedback Local Online* (RFLO) foi desenvol-

vido como um mecanismo de aprendizado recursivo para RNNs (Murray, 2019). A técnica é uma abordagem baseada em gradientes, adaptada para depender apenas de informações locais sobre a entrada e a saída, além de uma projeção aleatória do erro de saída da RNN no *feedback*.

As equações a seguir definem a *Recursive Learning Recurrent Neural Networks* (RLRNN) proposta em termos de suas entradas  $X_h$  e saídas previstas  $\hat{Y}_h$ . Onde,  $a_h$  é a entrada total,  $h_h$  é o vetor de atividade, e  $e_h$  é o erro de previsão.

$$a_h = \tanh(UX_h + Vh_{h-1}) \quad (2.12)$$

$$h_h = h_{h-1} + \frac{1}{t}[-h_{h-1} + \tanh(a_h)] \quad (2.13)$$

$$\hat{Y}_h = Wh_h \quad (2.14)$$

$$e_h = Y_h - \hat{Y}_h \quad (2.15)$$

O mecanismo de aprendizado baseado em gradientes RFLO é definido pelas seguintes equações de atualização dos pesos. Onde  $\eta$  são as taxas de aprendizado,  $B$  é uma matriz aleatória de pesos de *feedback*,  $p_h$  e  $q_h$  são os produtos acumulados da atividade pré e pós-sinápticas nas sinapses recorrentes e de entrada, respectivamente.

$$\Delta W_h = \frac{1}{T} \sum_{t=1}^h \eta_1 e_t h_t \quad (2.16)$$

$$\Delta V_h = \frac{1}{T} \sum_{t=1}^h \eta_2 [B \cdot e_t] p_t \quad (2.17)$$

$$\Delta U_h = \frac{1}{T} \sum_{t=1}^h \eta_3 [B \cdot e_t] q_t \quad (2.18)$$

$$p_h = \frac{1}{t} \tanh(a_{h-1}) h_{h-1} + \left(1 - \frac{1}{t}\right) p_{h-1} \quad (2.19)$$

$$q_h = \frac{1}{t} \tanh(a_{h-1}) X_{h-1} + \left(1 - \frac{1}{t}\right) q_{h-1} \quad (2.20)$$

## 2.4 Sistemas Inteligentes

No ambiente industrial contemporâneo, máquinas operam sob condições variáveis, complexas e incertas devido ao desgaste progressivo de componentes, mudanças ambientais (temperatura, umidade, vibração, pressão etc), além alterações

nos requisitos de produção. Esse cenário dinâmico impõe desafios significativos para a predição de RUL dos equipamentos.

Historicamente, diversas ferramentas de inteligência computacional foram desenvolvidas para a realização de prognóstico de falhas em equipamentos, contudo a maior parte dos modelos de predição de RUL desenvolvidos a partir de abordagens convencionais são treinados com um conjunto fixo de dados pré-existentes e assumem condições operacionais constantes ao longo do tempo, tornando-se inflexíveis e limitados diante de uma realidade com mudanças graduais e abrupta nos dados.

Diante disso, o conceito de Sistemas Inteligentes (SI) surge como uma solução mais robusta, flexível e eficaz aplicável ao prognóstico de falhas. Ao contrário das abordagens estacionárias, SIs incluem mecanismos de aprendizado contínuo e autoajuste capazes de adequá-los à instabilidade das mudanças operacionais, proporcionando modelos de predição mais confiáveis e versáteis.

Embora apresentem características construtivas dinâmicas, SIs são classificados em *Sistemas Adaptativos* (SA), quando capazes de autoajustar seus coeficientes, ou *Sistemas Evolutivos* (SE), quando capazes de autoajustar seu coeficiente e sua estrutura (Angelov, 2002; Lesot and Marsala, 2020).

### 2.4.1 Sistemas Adaptativos

SAs são modelos de aprendizado contínuo capazes de recalcular seus parâmetros, garantindo predições eficazes pela adaptação às mudanças dos dados, porém mantendo uma estrutura fixa e complexidade constante (Angelov, 2002).

Modelos computacionais, como *Adaptive Neuro-Fuzzy Inference System* (ANFIS), combinam (nn) e lógica *fuzzy* para ajuste de seus parâmetros com base em algoritmos de aprendizado supervisionado, como retropropagação e mínimos quadrados. No entanto, sua estrutura (número de regras Fuzzy) é previamente definida e não se altera à chegada de novos dados (Angelov, 2014a).

Vale ressaltar que, embora muitas publicações usem o termo "adaptativo" para descrever seus modelos, à luz da definição proposta acima, o ajuste de parâmetros não é realizado continuamente, mas sim desencadeado por algum evento, não sendo verdadeiramente "adaptativos".

## 2.4.2 Sistemas Evolutivos

SEs representam uma classe mais avançada e dinâmica de sistemas inteligentes, diferenciando-se dos SAs por sua capacidade não apenas de ajustar seus parâmetros, mas também de modificar sua estrutura à medida que novos dados são incorporados, (Leite et al., 2020). Tal comportamento confere aos SEs versatilidade para lidar com mudanças abruptas e gradativas nos padrões dos dados, tornando-os mais robustos e adequados para ambientes industriais dinâmicos e incertos.

Um diferencial dos SEs é a sua capacidade autônoma na adaptação estrutural, diferentemente dos SAs, que operam dentro de uma estrutura fixa. Em outras palavras, enquanto um SAs pode recalibrar seus parâmetros para refletir pequenas mudanças nos dados, um SE pode reconfigurar completamente sua estrutura interna (além de seus parâmetros), garantindo que o modelo permaneça eficiente mesmo sob mudanças operacionais bruscas e inesperadas, além de modificar a complexidade do modelo. Nesse contexto, a lógica *fuzzy* é significativamente explorada. Ao invés de trabalhar com dados de valor exato, regras *fuzzy* são utilizadas para representá-los em graus de pertinência, permitindo que os SEs trabalhem com probabilidades graduais, no lugar de valores rígidos e precisos.

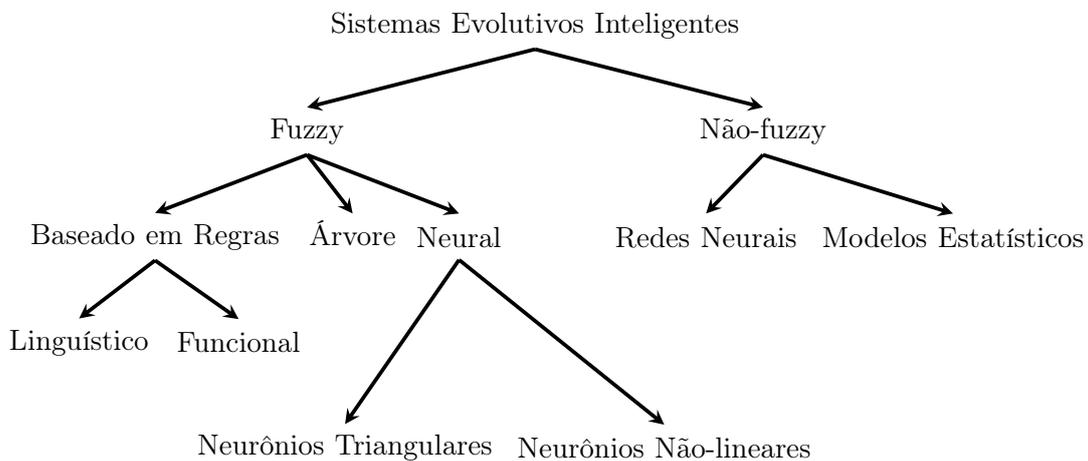


Figura 2.6: Classificação de alguns SEIs. Adaptado de: Lemos et al. (2013)

Sistemas evolutivos *fuzzy* podem, dinamicamente, criar e modificar regras para se adaptar às características dos dados, ou fundir regras redundantes e remover regras irrelevantes para reduzir sua estrutura. Ademias, regras *fuzzy* representam os dados de forma linguística e interpretável, permitindo associá-los às circunstâncias

operacionais do sistema. A Figura 2.6 apresenta alguns tipos de SEs.

Um exemplo clássico de SE *fuzzy* é *Evolving Takagi-Sugeno* (eTS), proposto por (Angelov, 2002). Trata-se de um modelo baseado em regras que evolui sua estrutura e parâmetros em tempo real, adaptando-se a novos dados sem a necessidade de intervenção externa. Outro exemplo é *Dynamic Evolving Neuro-Fuzzy Inference System* (DENFIS) de (Kasabov and Song, 2002), que combina NNs e lógica *fuzzy* para criar modelos dinâmicos e adaptativos.

## 2.5 Sistemas Evolutivos e Computação Granular

Apesar das vantagens apresentadas pelos SEs, existem desafios relativos à gestão de sua complexidade e interpretabilidade, pois à medida que novos dados são observados, pode haver uma expansão ou contração estrutural ocasionando o excesso ou a escassez de regras, neurônios ou funções de decisões, comprometendo sua eficiência computacional. Nesse contexto, a Computação Granular (CG) se apresenta como uma abordagem proeminente de aperfeiçoamento para SEs.

### 2.5.1 Computação Granular

A Computação Granular é um paradigma de processamento de dados que objetiva a criação de Grânulos de Informação (GI), uma espécie de “*cluster*” capaz de representar conjuntos de dados agrupados por similaridade, adjacência funcional, indistinguibilidade, e são capazes de encapsular propriedades e padrões descritivos de um fenômeno (Pedrycz, 2001).

Diferentes métodos de granulação de dados são utilizados, cada um adequado a diferentes tipos de aplicações e níveis de incerteza. A lógica *Fuzzy*, por exemplo, permite a criação de grânulos baseados em graus de pertinência, possibilitando uma transição suave entre diferentes categorias e tornando o modelo mais flexível diante da incerteza (Leite et al., 2010). A granulação por intervalos aritméticos agrupa os dados dentro de intervalos numéricos fixos, sendo útil para redução de complexidade e modelagem estatística (Leite et al., 2009). Já a granulação por densidade, utilizada em métodos como o *Density-based spatial clustering of applications with noise* (DBSCAN), identifica grânulos de informação com base na concentração dos dados,

sendo eficiente para detecção de padrões e anomalias (Jacob and Ramani, 2012).

Independentemente do método, a qualidade da granulação deve ser analisada. Dado um grânulo  $\mathcal{G}_i$ , a granularidade  $\rho_i$  (razão entre a quantidade de seus elementos internos e a quantidade total de dados observados) é uma métrica que permite avaliar sua generalização e especificidade, garantindo que os grânulos sejam internamente homogêneos, porém distintos entre si.

Ainda nesse contexto, (Pedrycz and Homenda, 2013) introduz o *Principle of Justifiable Granularity* (PJG), como uma ferramenta para auxiliar na granulação coerente dos dados. A cardinalidade dos elementos de um grânulo e o tamanho de sua área de cobertura permitem quantificar sua generalização e a especificidade. Assim, um grânulo é justificável quando uma quantidade significativa de dados é abrangido por uma pequena área de cobertura.

O autor também apresenta um método de alocação ótima de granularidade da informação, onde os grânulos são formados de maneira a otimizar essa relação entre cobertura e especificidade através de técnicas de otimização que ajustam dinamicamente os parâmetros dos grânulos para melhor se adequar aos dados disponíveis.

A Figura 2.7 ilustra a ideia por trás da granulação dos dados. A partir de um conjunto finito de dados  $(x_1, \dots, x_{10})$  é possível organizá-los em GIs caracterizados por parâmetros  $\mathcal{P}_i$  calculados segundo abordagens de conjuntos Fuzzy, conjuntos aproximados, intervalos aritméticos, tipicidade e excentricidade etc.

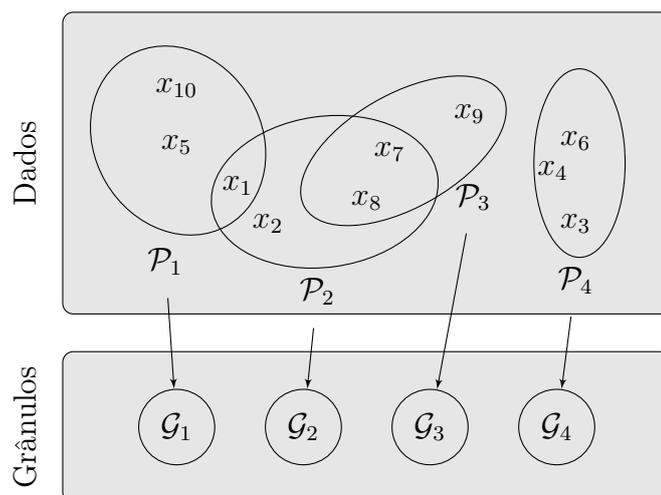


Figura 2.7: Granulação de dados.

## 2.5.2 Sistemas Granulares Evolutivos

A fundamentação de SEs na CG proporciona uma representação mais flexível e adaptativa dos dados, permitindo que a estrutura do modelo evolua de forma mais eficiente e escalável. A granulação permite a organização coesa e o encapsulamento dos dados em características (parâmetros como média, densidade, variância etc) daquele instante, consecutivamente adaptadas nos instantes seguintes.

O Sistema Granular Evolutivo (SGE), ou Modelo Granular Evolutivo (MGE), não opera exclusivamente com os dados propriamente ditos, mas com sua representação granular, tornando-se menos vulneráveis às oscilações e variações abruptas nos dados.

O protótipo ilustrado pela Figura 2.7 expõe a ideia de um modelo de predição evolutivo sustentado pela granulação de dados. A partir de um dado  $x_h$  num instante  $h$ , ocorre a criação ou exclusão de grânulos  $\mathcal{G}_i$  associados a uma série de parâmetros  $\mathcal{P}_{ih}$ . Finalmente, uma função  $g(x_h, \mathcal{P})$ , de qualquer natureza, realiza a predição de um valor  $y_h$  a partir de  $x_h$  e respectivos parâmetros granulares.

Naturalmente, a criação de grânulos é desencadeada pela inexistência de grânulo capaz de abranger um dado. Já a fusão de grânulos pode ser ocasionada pela sobreposição das zonas de abrangência de dois ou mais grânulos, e a exclusão pode ser mediante a inatividade de grânulo ao longo do tempo. Ressalta-se que os dois últimos eventos citados não possuem um critério bem estabelecidos, sendo definidos pelo desenvolvedor.

Em todo caso, a variação da quantidade  $i$  de grânulos  $\mathcal{G}$  afeta a quantidade de operações realizadas da etapa de parametrização à predição, influenciando diretamente no custo computacional requerido para os cálculos, ao contrario dos algoritmos de abordagem convencional de sua estrutura fixa.

A granulação dos dados, aspecto fundamental dos SGEs, não está associada a uma metodologia em especial, havendo diferentes abordagens com diferentes performances. Lógica Fuzzy, intervalos aritméticos, conjuntos aproximados, tipicidade e excentricidade são apenas alguns exemplos de métodos de granulação utilizados. A Tabela 2.1 sugere alguns SGEs e suas respectivas abordagens de granulação.

Modelo	Método de Granulação	Referência Bibliográfica
Evolving Takagi–Sugeno - $\epsilon$ TS	Lógica Fuzzy	(Angelov and Filev, 2004)
Fuzzy Based Evolving Model - FBeM	Lógica Fuzzy	(Leite et al., 2011)
Evolving Granular Neural Network - eGNN	Lógica Fuzzy	(Leite et al., 2010)
Interval Based Evolving Model - IBeM	Intervalos Aritméticos	(Leite et al., 2009)
TEDAPredict	Tipicidade e Excentricidade	(Kangin and Angelov, 2015)
TEDAClass	Tipicidade e Excentricidade	(Kangin and Angelov, 2015)
AutoCloud	Tipicidade e Excentricidade	(Bezerra et al., 2020)

Tabela 2.1: Sistemas Granulares Evolutivos e Métodos de Granulação.

### 2.5.2.1 Análise de Dados Baseada em Tipicidade e Excentricidade

Entre os SGEs exibidos pela Tabela 2.1, destacam-se os modelos com granulação de dados baseada análise da Tipicidade e Excentricidade, *Typicality and Eccentricity Data Analytics* (TEDA) em inglês, (Angelov, 2014b), devido sua fácil implementação, boa interpretabilidade e baixo custo computacional (Bezerra et al., 2020; Kangin et al., 2016).

Fundamentado pelas propostas de similaridade e dissimilaridade inicialmente exploradas por (Angelov and Yager, 2012), o TEDA propõe a distância (Euclidiana, Mahalanobis, cossenoidal etc) entre os dados para mensurar suas proximidades (numa ideia análoga à pertinência Fuzzy) e então organizá-los em subespaços coesos.

Esses "subespaços coesos de dados" apresentam diferentes taxonomias baseadas nas características de suas zonas de abrangência. Enquanto *clusters* apresentam subespaços com regiões e fronteiras bem estabelecidos, os GIs de (Pedrycz, 2000) e as nuvens (*clouds*, em inglês) de (Angelov and Yager, 2012) apresentam regiões e fronteiras eventualmente difusas/sobrepostas, além de outras similaridades e distinções muito sutis. Dessa forma, neste trabalho, nuvens serão consideradas um subtipo de grânulo de informação, sendo referidos desta forma.

Reiterando a definição proposta na Subseção 2.5.1, GIs são criados pelo agrupamento de dados de acordo com critérios de semelhança. No caso do TEDA, com base nos critérios de Tipicidade  $\tau$  e de Excentricidade  $\xi$ , indicadores de similaridade

e dissimilaridade, respectivamente.

A excentricidade descreve o quão distante/dissimilar uma amostra específica é em relação às demais, (Bezerra et al., 2020). Matematicamente, é definida como a soma das distâncias  $d$  de uma amostra de dados em particular a todas outras amostras, dividido pela soma das distâncias de todas as amostras de dados a todas outras amostras, assim a excentricidade compreende um valor entre 0 e 1. Para um grânulo  $\mathcal{G}_i$  num instante  $h$ , a excentricidade  $\xi_{h,i}$  de uma amostra  $x_h$  é definida por (Angelov, 2014a) e (Costa et al., 2015) conforme (2.21).

$$\xi_h = \frac{\sum_{j=1}^h d(X_h, X_j)}{\sum_{j=1}^h \sum_{k=1+1}^h d(X_j, X_k)} \quad (2.21)$$

$$\tau_h = 1 - \xi_h \quad (2.22)$$

A tipicidade se relaciona com a proximidade/similaridade de uma amostra de dados com os valores de amostras anteriores. Matematicamente, é definida como o complemento da diferença entre 1 e o valor da excentricidade. Para um grânulo  $\mathcal{G}_i$  num instante  $h$ , a tipicidade  $\tau_{h,i}$  de uma amostra  $x_h$  é definida conforme (2.22)

A granulação por Tipicidade e Excentricidade de uma amostra de dados  $x_h$  num instante  $h$  é ilustrada pela Figura 2.8, onde num universo de  $c$  grânulos, calcula-se a Excentricidade  $\xi_{ih}$  de  $x_h$  diante de cada grânulo de informação  $\mathcal{G}_i \mid i \in \mathbb{Z}^+ \leq c$ , permitindo avaliar a similaridade ou dissimilaridade de  $x_h$  diante de  $\mathcal{G}_i$

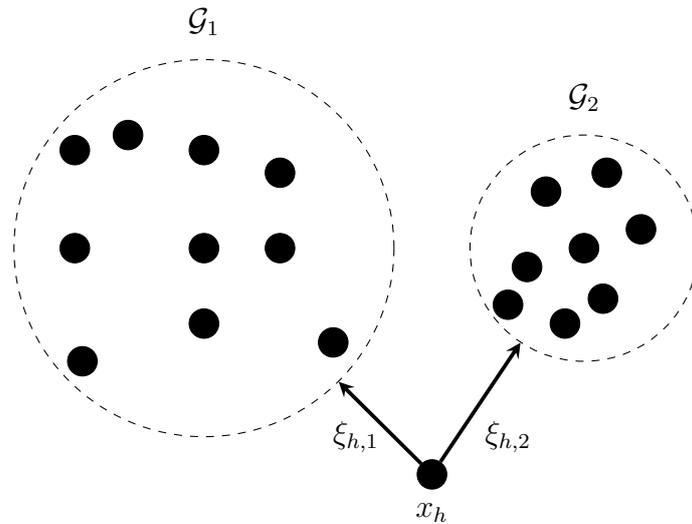


Figura 2.8: Granulação de dados baseado em TEDA.

A vantagem proposta pelo TEDA, detalhado em (Angelov, 2014b), está na característica recursiva de execução dos cálculos de Tipicidade e Excentricidade de uma amostra  $x_h$  para um grânulo  $\mathcal{G}_i$ , baseada exclusivamente nos parâmetros caracterizadores  $\mathcal{P}_{ih} = \{s_{ih}, \mu_{ih}, u_{ih}, \sigma_{ih}^2\}$  (respectivamente, cardinalidade, média, média do produto interno e variância), excluindo a necessidade de armazenamento de dados.

Assim, a equação para cálculo da Excentricidade proposta anteriormente é ajustada para um formato de cálculo recursivo, enquanto a Tipicidade continua calculada por (2.22). Neste trabalho, o cálculo das distâncias  $d$  entre amostras, de (2.21), será realizado distância Euclidiana, conforme (2.23).

$$\xi_h = \frac{1}{s_h} + \frac{(\mu_h - X_h)^\top (\mu_h - X_h)}{s_h \sigma_h^2} \quad (2.23)$$

Onde os parâmetros caracterizadores  $\mathcal{P}_i$  são calculados recursivamente pelas equações (2.24), (2.25), (2.26) e (2.27).

$$s_h = s_{h-1} + 1 \quad (2.24)$$

$$\mu_h = \mu_{h-1} \frac{(s_h - 1)}{s_h} + \frac{X_h}{s_h}, \quad (2.25)$$

$$u_h = u_{h-1} \frac{(s_h - 1)}{s_h} + \frac{X_h^\top X_h}{s_h}, \quad (2.26)$$

$$\sigma_h^2 = u_h - \mu_h^\top \mu_h. \quad (2.27)$$

Após os cálculo dos parâmetros  $\mathcal{P}_i$  de um grânulo  $\mathcal{G}_i$  associado a uma amostra  $x_h$ , verifica-se sua localização diante da zona de abrangência de  $\mathcal{G}_i$ . (Angelov, 2014b; Bezerra et al., 2020; Kangin et al., 2016) estabelecem que isso ocorre segundo a inequação proposta por (2.28).

$$\xi_h \leq \frac{m^2 + 1}{s_h} \quad (2.28)$$

Onde  $m$  é um parâmetro de sensibilidade para criação de grânulos e, caso a condição não seja atendida, a criação de um novo grânulo é desencadeada. Ressalta-se também que (2.28) pode ser reescrita em função da tipicidade  $\tau$  e de outros tipos de distância, sofrendo pequenas alterações.

### 2.5.2.2 Clusterização Baseada em TEDA

As publicações científicas acerca do TEDA apresentam algoritmos para clusterização de dados com alguma recorrência, onde é possível observar predominância da fundamentação utilizada no algoritmo TEDACluster proposto por (Kangin and Angelov, 2015). Contudo, (Bezerra et al., 2020) apresenta o algoritmo AutoCloud, onde sutis diferenças proporcionam vantagens em relação ao TEDACluster.

### 2.5.2.3 Algoritmo TEDACluster

No algoritmo TEDACluster, para cada amostra  $x_h$  e grânulo  $\mathcal{G}_i$  capazes de atender à inequação (2.28), calcula-se o respectivo nível de ativação  $w_i$ . A amostra  $x_h$  é então associada ao grânulo de maior  $w_i$  e os parâmetros  $\mathcal{P}_i$  são atualizados. Contudo o algoritmo não prevê mecanismo de exclusão/fusão de grânulos, tendendo unicamente à expansão de sua estrutura ao longo do tempo.

$$w_{ih} = \frac{\tau_{ih}}{\sum_{j=1}^c \tau_{jh}} \quad (2.29)$$

### 2.5.2.4 Algoritmo AutoCloud

Distintamente, o AutoCloud propõe a associação de toda amostra  $x_h$  a um grânulo  $\mathcal{G}_i$  capaz de atender à (2.28), possibilitando que uma amostra seja contemplada por um ou mais grânulos simultaneamente.

Conseqüentemente, um mecanismo de combinação granular é proposto apoiado pela simultaneidade granular apresentada pelas amostras. Quando a interseção amostral  $s_h^{\mathcal{G}_i \cap \mathcal{G}_j}$  de dois grânulos  $\mathcal{G}_i$  e  $\mathcal{G}_j$  for suficientemente elevada, a fusão granular é desencadeada conforme as Inequações (2.30) e (2.31).

$$s_h^{\mathcal{G}_i \cap \mathcal{G}_j} > s_{ih} - s_h^{\mathcal{G}_i \cap \mathcal{G}_j}, \quad (2.30)$$

$$s_h^{\mathcal{G}_i \cap \mathcal{G}_j} > s_{jh} - s_h^{\mathcal{G}_i \cap \mathcal{G}_j}. \quad (2.31)$$

Evidentemente, a fusão dos grânulos  $\mathcal{G}_i$  e  $\mathcal{G}_j$  cria um grânulo resultante  $\mathcal{G}_r$  que contempla as amostras singulares e simultâneas dos grânulos originais. Assim, a cardinalidade resultante  $s_{rh}$  do novo grânulo é calculada por (2.32). Já os outros parâmetros resultantes  $\mathcal{P}_r$  de  $\mathcal{G}_r$  são calculados pela média ponderada pelas

cardinalidades  $s_{ih}$  e  $s_{jh}$  dos GIs originais, conforme (2.33), (2.34) e (2.35).

$$s_{rh} = s_{ih} + s_{jh} - s_h^{\mathcal{G}_i \cap \mathcal{G}_j} \quad (2.32)$$

$$\mu_{rh} = \frac{s_{ih}\mu_{ih} + s_{jh}\mu_{jh}}{s_{ih} + s_{jh}} \quad (2.33)$$

$$u_{rh} = \frac{s_{ih}u_{ih} + s_{jh}u_{jh}}{s_{ih} + s_{jh}} \quad (2.34)$$

$$\sigma_{rh}^2 = \frac{(s_{ih} - 1)\sigma_{ih}^2 + (s_{jh} - 1)\sigma_{jh}^2}{s_{ih} + s_{jh} - 2} \quad (2.35)$$

Naturalmente, o mecanismo de fusão granular proposto diminui a quantidade de grânulos existente. Assim, diferentemente do TEDACluster, o AutoCloud apresenta uma estrutura expansível ou compressível mais versátil, podendo eventualmente, reduzir a complexidade estrutural do algoritmo. A seguir são expostos os pseudocódigos referentes à execução do algoritmo AutoCloud.

O Algoritmo 4 é responsável pela criação de grânulos. É executado na primeira iteração do Autocloud e sempre que não houver grânulo capaz de comportar um dado.

---

**Algoritmo 4** Módulo de Criação de Grânulo

---

- 1: **Procedimento** CriarGrânulo( $x_h, c$ )
  - 2:  $s_{ch} \leftarrow 1$
  - 3:  $\mu_{ch} \leftarrow x_h$
  - 4:  $u_{ch} \leftarrow x_h$
  - 5:  $\sigma_{ch}^2 \leftarrow 0$
  - 6: **Saída**  $\mathcal{G}_c = (s_{h,c}, \mu_{h,c}, [\mu_{x^T x}], [\sigma_{h,c}]^2, \Theta_{h,c})$
- 

O Algoritmo 5 é responsável pela cálculo dos novos parâmetros caracterizadores dos grânulos diante de cada amostra. Quando a inequação (2.28) é atendida, atualizam-se os parâmetros do grânulo.

---

**Algoritmo 5** Calcular Parâmetros do Grânulo

---

- 1: **Procedimento** Calcular( $x_h, i$ )
  - 2: Calcular  $s_{ih}$  pela Equação (2.24)
  - 3: Calcular  $\mu_{ih}$  pela Equação (2.25)
  - 4: Calcular  $u_{ih}$  pela Equação (2.26)
  - 5: Calcular  $\sigma_{ih}^2$  pela Equação (2.27)
  - 6: **Saída**  $\mathcal{P}_i = (s_{ih}, \mu_{ih}, u_{ih}, \sigma_{ih}^2)$
- 

O Algoritmo 6 verifica a sobreposição entre os grânulos. Quando for significativamente elevada, os grânulos são fundidos.

---

**Algoritmo 6** Módulo de Verificação de Fusão

---

- 1: **Procedimento** Verificar( $\mathcal{G}_i, \mathcal{G}_j$ )
  - 2: **se**  $s_h^{\mathcal{G}_i \cap \mathcal{G}_j} > s_{ih} - s_h^{\mathcal{G}_i \cap \mathcal{G}_j}$  ou  $s_h^{\mathcal{G}_i \cap \mathcal{G}_j} > s_{jh} - s_h^{\mathcal{G}_i \cap \mathcal{G}_j}$  **então**
  - 3:     Calcular  $s_{rh}$  pela Equação (2.32)
  - 4:     Calcular  $\mu_{rh}$  pela Equação (2.33)
  - 5:     Calcular  $u_{rh}$  pela Equação (2.34)
  - 6:     Calcular  $\sigma_{rh}^2$  pela Equação (2.35)
  - 7:     **Saída**  $\mathcal{P}_r = (s_{rh}, \mu_{rh}, u_{rh}, \sigma_{rh}^2)$
  - 8: **fim se**
- 

### 2.5.2.5 Predição Baseada em TEDA

Embora predições não tenham sido inicialmente exploradas no TEDA de (Angelov, 2014b), o autor destaca sua viabilidade. Posteriormente, (Kangin and Angelov, 2015; Soares et al., 2017) exploraram oreditores TEDA, porém mantendo a granulação estabelecida pelo algoritmo TEDACluster.

Os preditores TEDA possuem similaridades com outros modelos sustentados por lógica Fuzzy, apresentando um conjunto de regras  $R_i$  do tipo "se, então" com uma etapa antecedente, seguida de uma consequente, conforme estipulado em (2.36).

$$R_i(x): \text{ Se } x \text{ é } z_i, \text{ então } y_i = g(x, \mathcal{Q}_i) \quad (2.36)$$

Na regra  $R_i$ , a etapa antecedente "se  $x$  é  $z_i$ " é responsável pela granulação dos dados, enquanto a consequente  $y_i = g(x, \mathcal{Q}_i)$  prediz  $y_i$  a partir de uma amostra  $x$  e coeficientes  $\mathcal{Q}_i$ , cujo cálculo é realizado por uma função genérica  $g$  que pode assumir a forma constante, linear, não linear, polinomial, de rede neural etc.

Analogamente, os preditores TEDAPredict e o TEDAClass, citados anteriormente, são estabelecidos por regras com uma etapa antecedente de granulação de dados, "se  $x$  é  $\mathcal{G}_i$ ", e uma etapa consequente de predição, "então  $y_i$  é  $x^T \Theta_i$ ".

$$\text{Regra: Se } x \text{ é } \mathcal{G}_i \text{ então } \hat{y}_i = x^T \Theta_i \quad (2.37)$$

Onde, para cada grânulo  $\mathcal{G}_i$  associado a uma amostra  $x$ , existe uma predição granular  $\hat{y}_i$  baseada nos seus parâmetros  $\Theta_i$ , referentes a uma matriz de coeficientes de uma regressão linear. A amostra  $x_h$  é associada ao grânulo  $\mathcal{G}_i$  de maior nível de ativação  $w_i$ , segundo à condição imposta em (2.28).

$$w_{ih} = \frac{\tau_{ih}}{\sum_{j=1}^c \tau_{jh}} \quad (2.38)$$

Posteriormente, quando a amostra  $x_h$  for associada a um grânulo  $\mathcal{G}_i$ , seus respectivos parâmetros caracterizadores  $\mathcal{P}_i$  são atualizados, e a matriz de coeficientes  $\Theta_i$  recalculada com base no método dos mínimos quadrados recursivo, do inglês *Recursive Least Squares* (RLS), cujos algoritmos detalhados podem ser verificados em (Bargiela and Pedrycz, 2003). Finalmente a predição singular  $\hat{y}_h$  é calculada conforme (2.39). O Algoritmo 7 mostra o pseudocódigo utilizado pelo TEDAPredict.

$$\hat{y}_h = \sum_{i=1}^c w_{h,i}(x_h) \hat{y}_{ih} \quad (2.39)$$

---

#### Algoritmo 7 Preditor TEDA

---

```

1: Procedimento PreditorAutoCloud ( $X_h, Y_h, m, c = 0$ )
2: enquanto o algoritmo receber dados faça
3:   Criar  $\leftarrow 1$ 
4:   se  $h = 1$  então
5:      $c \leftarrow 1$ 
6:      $\mathcal{G}_c \leftarrow$  Criar( $X_h, Y_h, c$ ) Executando o Algoritmo 4
7:   senão
8:     se  $h \geq 2$  então
9:        $w_{\max} \leftarrow 0, \mathcal{G}_{\max} \leftarrow 0$ 
10:      para  $t$  façado grânulo  $\mathcal{G}_i$ 
11:         $\mathcal{P}_i^* \leftarrow$  Calcular ( $X_h, i$ ) Executando o Algoritmo 5
12:        Calcular  $\xi_{h,i}(X_h), \tau_{h,i}(X_h)$  e  $w_{h,i}(X_h)$  pelas equações (2.23), (2.22) e (2.38)
13:        Calcular as predições granulares  $\hat{Z}_i$  por (2.37)
14:        Calcular a predição singular  $\hat{Z}_h$  por (2.39)
15:        se  $w_{\max} < w_i$  então
16:           $w_{\max} \leftarrow w_i, \mathcal{G}_{\max} \leftarrow i, \mathcal{P}_{\max} \leftarrow \mathcal{P}_i^*$ 
17:        fim se
18:        se a então Inequação (2.28) for atendida
19:           $\mathcal{P}_i \leftarrow \mathcal{P}_i^*$ 
20:           $\Theta_i \leftarrow \Theta_i^*$  Calculado usando RLS
21:          Criar  $\leftarrow 0$ 
22:        fim se
23:      fim para
24:      se Criar = 1 então
25:         $c \leftarrow c + 1$ 
26:         $\mathcal{G}_c \leftarrow$  CriarGrânulo( $X_h, Y_h, c$ ) Executando o Algoritmo 4
27:      fim se
28:    fim se
29:  fim se
30: fim enquanto

```

---

## 2.6 Explicabilidade e Interpretabilidade de Modelos de Predição

Apesar dos elevados níveis de precisão alcançados por modelos de predição contemporâneos, a complexidade intrínseca de suas operações compromete a transparência dos seus processos e o entendimento de suas predições, tornando-os comumente chamados de “caixas-pretas” (Burrell, 2016).

Em áreas críticas, como finanças, manutenção preditiva e direção autônoma, onde as decisões/predições tomadas pelos modelos podem ter um impacto significativo, a rastreabilidade de como os dados de entrada são convertidos em predições é imprescindível para garantir sua transparência e confiabilidade. Portanto, a interpretabilidade e a explicabilidade tem se destacado como novos conceitos no área de IA (Ortigossa et al., 2024).

Interpretabilidade pode ser definida como a habilidade de apresentar algo de uma forma entendível (Doshi-Velez and Kim, 2017). Em termos de IA, exceto pelo resultado de sua predição, a interpretação de seu raciocínio não é fácil, o que impede que seus resultados sejam compreendidos de forma adequada. Contudo, se uma predição não for interpretável, é possível associá-la a mecanismos de compreensão capazes de evidenciar/destacar os procedimentos/decisões de cálculo utilizados, tornando-a explicável. Assim, a explicabilidade pode ser definida com uma contraparte da racionalização da tomada de decisão (Chakraborty et al., 2017). Entre as abordagens para prover explicabilidade aos modelos de IA podemos destacar:

- Métodos *proxy* são modelos simplificados (regressão linear ou árvores de decisão, por exemplo) que aproximam o comportamento de um modelo complexo em uma região específica do espaço de entrada, tornando as decisões e previsões do modelo completo mais compreensíveis, (Gilpin et al., 2018).
- O mapeamento de saliência é uma abordagem usada para quantificar a importância das características de uma entrada e verificar as mais influentes para a decisão do modelo (Gomez et al., 2022).
- A extração automática de regras é uma abordagem que busca gerar explicações baseadas em regras explícitas, como regras *fuzzy* “se, então” (Andrews et al.,

1995).

- A computação granular é uma abordagem que objetiva a criação de conjuntos de dados organizados por critérios de similaridade denominados GIs (Andrews et al., 1995).

# Capítulo 3

## Revisão da Literatura

Este capítulo discorre acerca das publicações referentes ao prognóstico de falhas em rolamentos. Para isso foi realizada uma busca na plataforma *Scopus* com os filtros para a delimitação do assunto deste trabalho descritos pela Tabela 3.1

Palavras-chave	Período	Área do Assunto	Tipo de Documento	Tipo de Equipamento
Prognostic, RUL, Prediction, Bearing	2022 a 2025	Engineering, Computer Science	Article	Roller Bearings

Tabela 3.1: Filtros para delimitação da busca na plataforma Scopus.

### 3.1 Bibliografias Analisadas

A busca filtrou apenas publicações recentes (entre 2022 e 2025) acerca de prognóstico de falhas em rolamentos, retornado um total de 54 publicações e pode ser replicada através da chave “TITLE-ABS-KEY(prognostic AND rul AND prediction AND bearing) AND PUBYEAR > 2021 AND PUBYEAR < 2026 AND ( LIMIT-TO ( SUBJAREA,“ENGI” ) OR LIMIT-TO ( SUBJAREA,“COMP” ) ) AND ( LIMIT-TO ( EXACTKEYWORD,“Roller Bearings” ) ) AND ( LIMIT-TO ( DOCTYPE,“ar” ) )”.

As informações pertinentes das 30 primeiras publicações, ordenadas pelo critério de maior quantidade de citações, foram extraídas, organizadas e detalhadas nas Tabelas 3.2 e 3.3.

Da esquerda para a direita nas Tabelas 3.2 e 3.3, a primeira coluna indica a contagem das publicações, a segunda e a terceira indicam as referências bibliográficas da publicação pesquisada e da base de dados de rolamento utilizada. A quarta coluna indica a técnica de pré-processamento de dados usada ou sua ausência e a quinta indica o domínio de processamento dos dados. A sexta coluna a natureza do IS utilizado e a sétima indica o modelo de predição utilizado para o cálculo de prognóstico.

Nº	Referência do Artigo	Base de Dados	Pré processamento	Domínio de Processamento	Características e/ou Indicador de Saúde	Modelo de Prognóstico
01	Ni et al. (2022)	Wang et al. (2018)	Ausente	Tempo	Estatísticos	GRU
02	Deng et al. (2023)	Wang et al. (2018), Nectoux et al. (2012)	Normalização, Alinhamento Tempo-escala	Tempo e Tempo-frequência	Estatísticos e de Imagem	PI-BDDN
03	Zhu et al. (2022)	Wang et al. (2018), Nectoux et al. (2012)	CWT	Tempo e Tempo-frequência	Estatísticos, Trigonométricos e de Imagem	BNN
04	Cheng et al. (2022b)	Wang et al. (2018)	Critério $3\sigma$	Tempo	<i>Auto-encoder</i>	AEQRNN
05	Xu et al. (2023)	Wang et al. (2018)	Ausente	Tempo	<i>Auto-encoder</i>	CNN-BiLSTM
06	Chang et al. (2022)	Nectoux et al. (2012)	<i>Marginal Spectrum</i>	Tempo-frequência	<i>Encoders</i> de Múltiplas Camadas	TFT
07	Mao et al. (2023)	Wang et al. (2018), Nectoux et al. (2012)	Ausente	Domínio Adaptativo	CNN	Deep domain-adversarial regression
08	Guo et al. (2024)	Wang et al. (2018)	CEEMDAN	Tempo e Frequência	Estatísticos e CNN	Processo de Wiener
09	Zhang et al. (2023)	Wang et al. (2018)	Normalização Escalonamento de Tempo	Tempo	Estatísticos e LSTM	LSTM-TDNN
10	Rathore and Harsha (2022a)	Nectoux et al. (2012)	DWT	Tempo, Frequência e Tempo-frequência	Estatísticos	SBiLSTM
11	Rathore and Harsha (2022c)	Nectoux et al. (2012)	Filtro de Savitzky-Golay e DWT	Domínio Adaptativo	Estatísticos	TBiLSTM
12	Deng et al. (2022)	Nectoux et al. (2012)	Filtro CNN	Adaptativo	CNN	MsDCCRB
13	Li et al. (2023)	Wang et al. (2018)	VBAKF	Tempo	TSP	Kalman filtering
14	Alfarizi et al. (2023)	Base de dados própria	EMD	Tempo	Indicadores Estatístico	<i>Random Forest</i>

Tabela 3.2: Referências analisadas e informações características - Parte 1.

N°	Referência do Artigo	Base de Dados	Pré processamento	Domínio de Processamento	Características e/ou Indicador de Saúde	Modelo de Prognóstico
15	Yang et al. (2024)	Nectoux et al. (2012)	Não	Tempo e Frequência	Indicadores Estatísticos e <i>Auto-encoder</i>	RSA-MSGCU
16	Chen et al. (2022a)	Wang et al. (2018)	Normalização e Janela Deslizante	Tempo	CNN	SACovFormer
17	Cheng et al. (2022a)	Wang et al. (2018)	VMD	Tempo Frequência e Tempo-frequência	Indicadores Estatísticos	MDDNN
18	Kong and Li (2022)	Wang et al. (2018)	Não	Tempo	Indicadores Estatísticos	SBiLSTM
19	Wang et al. (2022b)	Wang et al. (2018)	Janela Deslizante	Tempo	<i>Convolution, Pooling</i>	MEHDLN
20	Rathore and Harsha (2022b)	Nectoux et al. (2012)	DWT	Tempo, Frequência e Tempo-frequência	Indicadores estatísticos	BiLSTM
21	Qi et al. (2024)	Wang et al. (2018), Qiu et al. (2006)	<i>1/3-binary tree</i>	Tempo, Frequência e Tempo-frequência	Indicadores Estatísticos	MHE
22	Behera and Misra (2023)	Nectoux et al. (2012)	MTF e CNN	Tempo-frequência	Características da Imagem	MMF-DTL
23	Yao et al. (2024)	Wang et al. (2018)	Normalização	Tempo	<i>Convolutional Encoder</i>	CDAE e Bi-LSTM
24	Guo and He (2022)	Nectoux et al. (2012)	Não	Tempo, Frequência e Tempo-frequência	Indicadores Estatísticos	RVM
25	Wang et al. (2023)	Wang et al. (2018) e Base de Dados Própria	Não	Domínio do Tempo	Indicadores Estatísticos	Staged RUL Prediction Model
26	Gu et al. (2022)	Wang et al. (2018)	EMD	Tempo, Frequência e Tempo-frequência	Indicadores Estatísticos	ELM-WPHM
27	Li and Deng (2023)	Nectoux et al. (2012)	CWT	Tempo-frequência	Características da Imagem	CNN-Bayesian LSTM
28	Li et al. (2024b)	Wang et al. (2018)	Não	Tempo e Frequência	Indicadores Estatísticos	MSCATN
29	Zheng et al. (2024)	Wang et al. (2018)	Não	Tempo	<i>Auto-encoder</i>	DRL
30	Li et al. (2024a)	Nectoux et al. (2012)	Não	Tempo, Frequência e Tempo-frequência	Indicadores Estatísticos	GTFDAU

Tabela 3.3: Referências analisadas e informações características - Parte 2.

## 3.2 Bases de Dados Utilizadas

No âmbito do prognóstico de falhas em rolamentos, a vibração mecânica é predominantemente utilizada como dado para criação de modelos de predição de RUL e está presente em todas as bases de dados utilizadas na pesquisas citadas. O

gráfico da Figura 3.1 apresenta a frequência de citações de cada base utilizada.

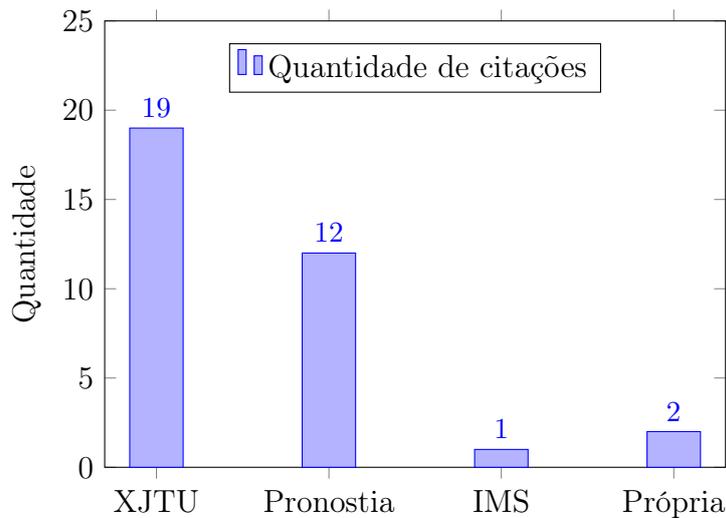


Figura 3.1: Quantidade de citações por bases de dados utilizadas.

Todas as bases de dados de vibração citadas são resultados de testes *run-to-failure* conduzidos em laboratórios, sendo que as bases de dados proprietárias (Alfarizi et al., 2023) e (Wang et al., 2023) não foram disponibilizadas, enquanto as demais são facilmente encontradas em repositórios *online*.

A aquisição dos dados de vibração foi realizada com acelerômetros de diferentes taxas de amostragem fixados sobre os mancais de rolamento nas direções vertical e horizontal. A Tabela 3.4 determina as frequências de amostragem de cada base.

Base de dados	Frequência de Amostragem
Wang et al. (2018)	25.600 Hz
Nectoux et al. (2012)	25.600 Hz
Qiu et al. (2006)	20.000 Hz
Base proprietária de Alfarizi et al. (2023)	Não informado
Base proprietária de Wang et al. (2023)	25.600 Hz

Tabela 3.4: Frequência de amostragem de dados das bases utilizadas.

As bases disponibilizadas registram quantidades diferentes de testes *run-to-failure* sob diferentes condições operacionais, destacando-se as bases (Wang et al., 2018) e (Qiu et al., 2006) por, sobretudo apresentar, a tipo de falha sofrida pelo rolamento ao final do teste, diferentemente de (Nectoux et al., 2012). A Tabela 3.5 demonstra as características de condições operacionais dos testes.

Base de dados	Quantidade de testes (unidades)	Medições por amostra	Período entre amostras (segundos)	Rotação do motor (RPM)	Carga aplicada (kN)	Falhas disponíveis
Wang et al. (2018)	15	32768	60	2100, 2250 e 2400	12, 11 e 10	Sim
Nectoux et al. (2012)	21	2560	10	1800, 1650 e 1500	4, 4.2 e 5	Não
Qiu et al. (2006)	3	20.480	600	2000 RPM	28	Sim

Tabela 3.5: Condições operacionais das bases de dados utilizadas.

### 3.3 Pré-processamento de Dados

Entre as particularidades dos sinais/dados de vibração, o ruído inerente ao aparato de coleta de dados se destaca por eventualmente prejudicar o desempenho do modelo de predição. Ademais, problemas como ausência ou incoerência de dados por falhas na coleta, valor numérico do dado coletado e tamanho da amostra também podem causar o mesmo problema.

Nesse contexto alguns autores preparam os dados numa etapa antecedente ao processamento dos dados seja para filtrá-los, remover ou substituir dados anômalos, normalizá-los, segmentá-los etc. Por outro lado, alguns autores utilizam dados brutos, deixando a cargo do modelo de predição o trabalho de lidar com suas nuances. As diferentes abordagens de pré-processamento, ou a simples ausência, expostas pelas Tabelas 3.2 e 3.3 refletem a inexistência de um consenso acerca da desta etapa.

Podemos organizar os métodos de pré-processamento de dados apresentados em ausentes, quando os dados são utilizados de forma bruta (Ni et al., 2022), preparação, quando são usadas técnicas de janelamento e/ou normalização de dados (Chen et al., 2022a) e filtragem, quando são usadas técnicas de decomposição como VMD (Cheng et al., 2022a). A Tabela 3.6 determina a classificação de cada método de acordo com o autor.

A organização da classificação permite quantificar a proporção de cada método diante do total de publicações, revelar a inexistência de um procedimento padrão. O gráfico da Figura 3.2 demonstra a frequência de aparição de cada método, respectivamente.

Classificação	Referência
Ausente	Guo and He (2022); Kong and Li (2022); Li et al. (2024a,b); Mao et al. (2023); Ni et al. (2022); Wang et al. (2023); Xu et al. (2023); Yang et al. (2024); Zheng et al. (2024)
Preparação	Chen et al. (2022a); Deng et al. (2023); Wang et al. (2022b); Yao et al. (2024); Zhang et al. (2023)
Filtragem	Alfarizi et al. (2023); Behera and Misra (2023); Chang et al. (2022); Cheng et al. (2022a,b); Deng et al. (2022); Gu et al. (2022); Guo et al. (2024); Li and Deng (2023,?); Qi et al. (2024); Rathore and Harsha (2022a,b,c); Zhu et al. (2022)

Tabela 3.6: Classificação do método de pré-processamento de acordo com o autor

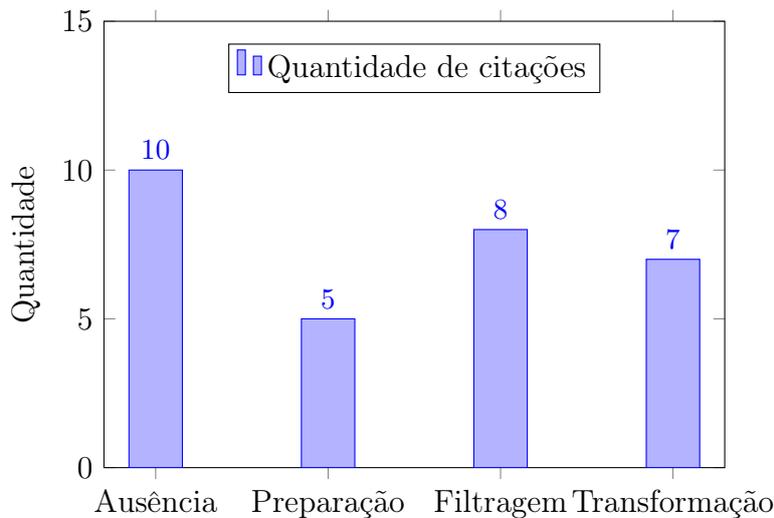


Figura 3.2: Frequência das técnicas de pré-processamento por classificação.

### 3.4 Domínio de processamento

Os sinais de vibração podem ser analisados em diferentes domínios de processamento, cada um oferecendo perspectivas distintas sobre os dados. No domínio do tempo, os dados são analisados em função do tempo e as informações são extraídas diretamente da forma de onda.

No domínio da frequência, os dados são analisados em função da frequência, possibilitando a identificação de padrões periódicos, difíceis de identificar no domínio do tempo, porém perdendo informações temporais e omitindo mudanças instantâneas.

Já no domínio tempo-frequência, os dados são analisados simultaneamente em função do tempo e da frequência, permitindo a visualização de mudanças ins-

tantâneas nos sinais ao longo do tempo.

O domínio de processamento dos sinais de vibração influencia diretamente na extração/seleção de características e na criação de indicadores de saúde. Contudo, para alguns modelos de predição contemporâneos, o domínio de processamento pode se tornar irrelevante, especialmente em abordagens baseadas em aprendizado profundo, que extraem características diretamente dos dados brutos.

Na área de detecção e diagnóstico de falhas em rolamentos, o uso do domínio da frequência já está bem estabelecido, devido à sua capacidade de revelar padrões característicos associados a falhas específicas. No entanto, quando se trata do cálculo de prognóstico, predição da RUL, observa-se uma diversidade de abordagens na revisão realizada. Enquanto alguns autores optam exclusivamente por um único domínio, outros usam os três domínios simultaneamente e outros usam abordagens adaptativas. A Tabela 3.7 mostra o domínio selecionado por cada autor

<b>Classificação</b>	<b>Referência</b>
Domínio do tempo	Alfarizi et al. (2023); Chen et al. (2022a); Cheng et al. (2022b); Kong and Li (2022); Li and Deng (2023); Ni et al. (2022); Wang et al. (2022b, 2023); Xu et al. (2023); Yao et al. (2024); Zhang et al. (2023); Zheng et al. (2024)
Domínio do tempo-frequência	Chang et al. (2022); Li and Deng (2023)
Múltiplos domínios	Cheng et al. (2022a); Deng et al. (2023); Gu et al. (2022); Guo et al. (2024); Guo and He (2022); Li et al. (2024a,b); Qi et al. (2024); Rathore and Harsha (2022a,b); Yang et al. (2024); Zhu et al. (2022)
Domínio da imagem	Behera and Misra (2023)
Domínio adaptativo	Deng et al. (2022); Mao et al. (2023); Rathore and Harsha (2022c)

Tabela 3.7: Classificação do domínio de processamento de acordo com o autor.

A análise revela que, embora não haja consenso quanto ao domínio estabelecido, a predominância está no uso do domínio do tempo e de múltiplos domínios para a criação de um modelo de predição de RUL. O gráfico da Figura 3.3 demonstra a frequência de cada domínio abordado pelos autores.

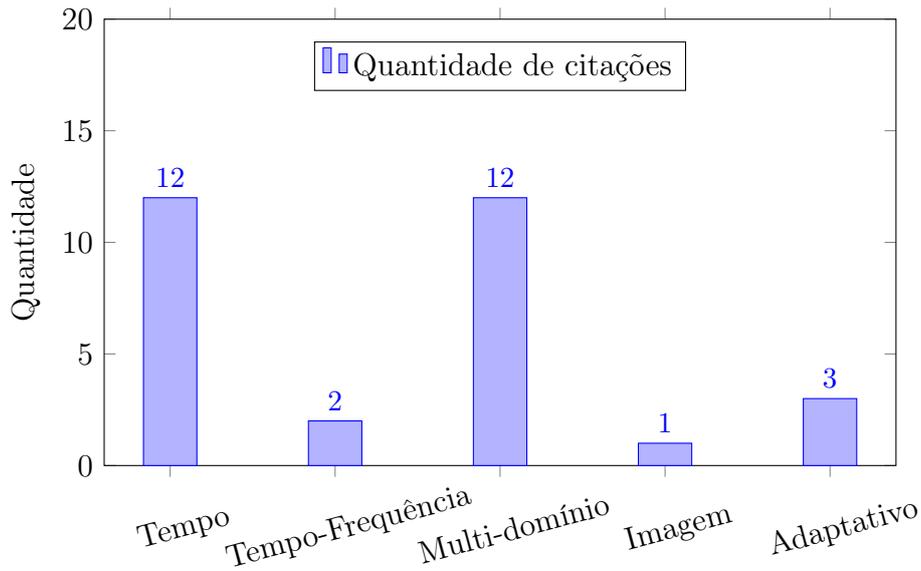


Figura 3.3: Frequência dos domínios de processamento por classificação.

### 3.5 Características e/ou Indicador de Saúde

Um aspecto relevante a ser detalhado pela revisão realizada diz respeito a extração e seleção de características e/ou criação de um IS para realizar as predições de RUL. Nesse sentido, os trabalhos apresentados na revisão realizada, apresentam características extraídas através de equações estatísticas e trigonométricas tanto no domínio do tempo quanto no domínio da frequência.

Outra abordagem observada é utilização de modelos computacionais do tipo DL para extração de características sem o uso de uma formulação matemática explícita bem definida, como *auto-encoders* e CNNs. O resultado são características cujos valores numéricos representam padrões complexos e relações não triviais entre os dados, podendo, inclusive, serem mono ou multidimensionais. Contudo, a ausência de um lastro matemático/físico/estatístico mitiga qualquer explicabilidade acerca dessas características, tornando sua interpretabilidade inexistente.

Também são observados trabalhos que usam as duas abordagens. De forma geral podemos separar as abordagens para extração de características dos dados em estatísticas, aprendidas (através de DL), e combinadas. A Tabela 3.8 mostra a frequência das abordagens para extração de características nos trabalhos pesquisados e gráfico da Figura 3.4 demonstra a frequência das abordagens para extração de

características feita pelos autores.

Classificação	Referência
Estatísticos	Alfarizi et al. (2023); Cheng et al. (2022a); Gu et al. (2022); Guo and He (2022); Kong and Li (2022); Li et al. (2024a,b); Ni et al. (2022); Qi et al. (2024); Rathore and Harsha (2022a,b,c); Zhang et al. (2023)
Aprendidos	Behera and Misra (2023); Chang et al. (2022); Chen et al. (2022a); Cheng et al. (2022b); Deng et al. (2022); Li and Deng (2023,?); Mao et al. (2023); Wang et al. (2022b); Xu et al. (2023); Yao et al. (2024); Zheng et al. (2024)
Combinados	Deng et al. (2023); Guo et al. (2024); Wang et al. (2023); Yang et al. (2024); Zhu et al. (2022)

Tabela 3.8: lassificação do método de extração de características de acordo com o autor.

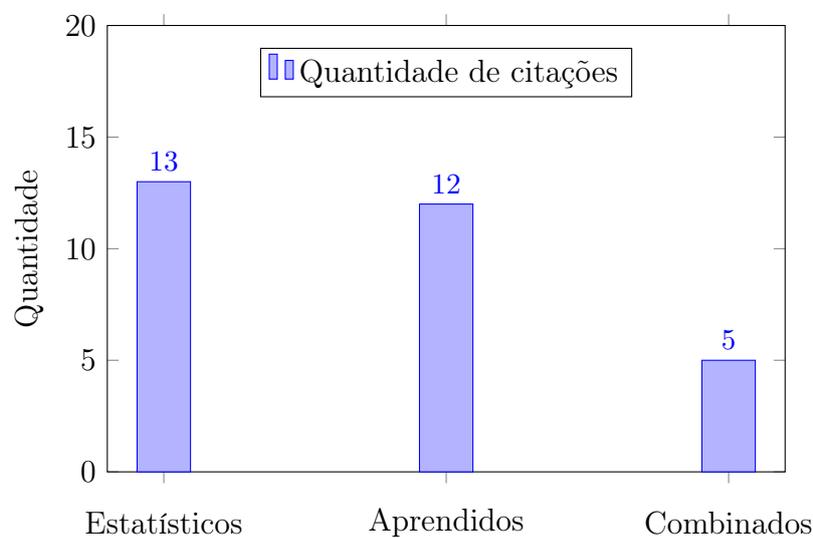


Figura 3.4: Frequência das abordagens para extração de características por classificação.

A revisão da literatura mostra que não há consenso entre as abordagens utilizadas. Porém as características estatísticas e as aprendidas são predominantemente abordadas.

### 3.6 Modelo de Prognóstico

A literatura registra o uso de diversos modelos para predição de RUL ao longo dos anos através de revisões bibliográficas como as propostas por (Lei et al., 2018; Polverino et al., 2023; Ramezani et al., 2019). Os registros apontam abordagens variadas para modelos de predição de RUL como regressões lineares, ML e modelos de DL, entre outras.

Na pesquisa realizada para o período de 2022 a 2025, observam-se predições de RUL realizadas com aprendizado profundo, aprendizado supervisionado e abordagens estatísticas. A Tabela 3.9 mostra a frequência das classificações de modelos de predição de RUL nos trabalhos pesquisados.

Classificação	Referência
<i>Deep Learning</i>	Alfarizi et al. (2023); Behera and Misra (2023); Chang et al. (2022); Chen et al. (2022a); Cheng et al. (2022a,b); Deng et al. (2022); Gu et al. (2022); Guo et al. (2024); Guo and He (2022); Kong and Li (2022); Li et al. (2024a); Li and Deng (2023); Li et al. (2024b); Mao et al. (2023); Ni et al. (2022); Qi et al. (2024); Rathore and Harsha (2022a,b,c); Wang et al. (2022b, 2023); Xu et al. (2023); Yang et al. (2024); Yao et al. (2024); Zhang et al. (2023,?); Zheng et al. (2024); Zhu et al. (2022)
Estatístico	Li and Deng (2023)

Tabela 3.9: Classificação do modelo de prognóstico de acordo com o autor.

A contagem de referências para cada classificação exposta na Tabela 3.9 revela a predominância de modelos de DL para cálculos de prognóstico. O gráfico da Figura 3.5 mostra a frequência das classificações de modelos de predição observados.

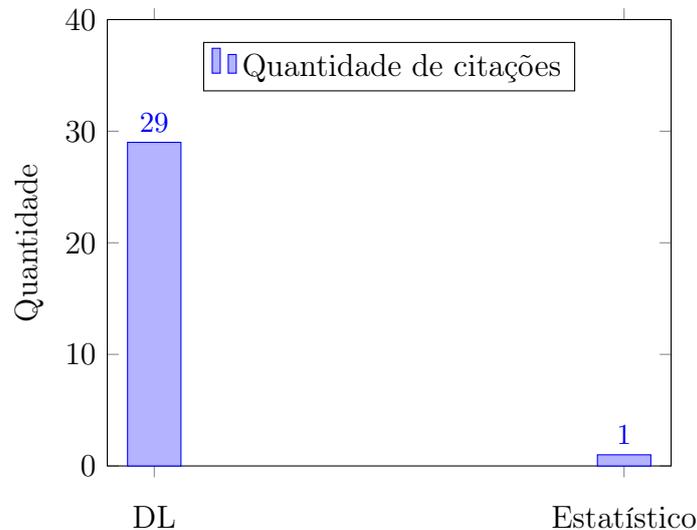


Figura 3.5: Frequência dos modelos de prognóstico característicos por classificação.

Considerando que abordagem de DL é predominante, ressalta-se que não foi observada preocupação a respeito da interpretabilidade ou explicabilidade dos modelos nos trabalhos observados.

## 3.7 Discussão Acerca dos Trabalhos Apresentados

A partir dos trabalhos analisados registrados por esta revisão da literatura, será conduzida uma discussão acerca de algumas de suas características e outros questionamentos, conforme a Tabela 3.10.

<b>Objetivo Principal</b>	<b>Dependência de Histórico de Dados</b>	<b>Atualização ou Adaptabilidade</b>	<b>Evolutividade</b>	<b>Interpretabilidade</b>
Propor modelo otimizado, novo, híbrido, técnica de processamento ?	Depende de dados pré-existentes ou consegue aprender do zero ?	O modelo ajusta seus parâmetros à dinâmica dos dados?	Sua estrutura consegue expandir ou contrair?	O modelo permite interpretabilidade dos dados e previsões

Tabela 3.10: Aspectos de discussão dos trabalhos pesquisados.

### 3.7.1 Objetivo Principal

Os principais focos das publicações revisadas podem estar associados ao modelo de predição ou método de extração de características/criação de um IS. A Tabela 3.11 associa as referências a cada foco citado.

<b>Foco do trabalho</b>	<b>Referência</b>	<b>Quantidade</b>
Modelo de Predição	Alfarizi et al. (2023); Behera and Misra (2023); Chang et al. (2022); Chen et al. (2022a); Cheng et al. (2022a,b); Deng et al. (2022, 2023); Guo et al. (2024); Guo and He (2022); Kong and Li (2022); Li et al. (2023, 2024a); Li and Deng (2023); Li et al. (2024b); Mao et al. (2023); Qi et al. (2024); Rathore and Harsha (2022a,b,c); Wang et al. (2022b, 2023); Yang et al. (2024); Yao et al. (2024); Zhang et al. (2023); Zheng et al. (2024); Zhu et al. (2022)	27
Extração de Características ou Criação de Indicador de Saúde	Gu et al. (2022); Guo et al. (2024); Ni et al. (2022); Xu et al. (2023)	4

Tabela 3.11: Objetivos dos trabalhos de acordo com o autor.

A Tabela 3.11 revela a predominância dos trabalhos relativos aos modelos de predição, 27 no total. Enquanto a extração de características/criação de um IS ocorre 4 vezes. Vale ressaltar que a referência (Guo et al., 2024) está associada aos dois focos.

### 3.7.1.1 Trabalhos Focados em Modelos de Predição

Os trabalhos focados em modelos de predição serão categorizados em trabalhos de otimização, hibridização ou agrupamento de modelos, conforme os critérios:

- Otimização: trabalhos que melhorem a performance de um modelo preditivo pré-existente pela utilização de algoritmos adjacentes para ajuste de hiperparâmetros, para clusterização de dados, para filtragem de sinal ou para extração de características eficientes (Chang et al., 2022; Rathore and Harsha, 2022c),.
- Hibridização: Embora (Kim et al., 2017; Lei et al., 2018) definam modelos híbridos como aqueles criados através da integração de modelos de diferentes naturezas (IA, estatísticos e físicos), alguns trabalhos pesquisados chamam de híbrido o resultado da união de modelos de mesma natureza, (Deng et al., 2023). Dessa forma, modelos incluídos nessas duas circunstâncias serão tratados como híbridos.
- Agrupamento: trabalhos com predições realizadas através da combinação de predições de diferentes modelos (Opitz and Maclin, 1999; Rokach, 2010).

O gráfico da Figura 3.6 quantifica os trabalhos segundo a categorização proposta, e quantifica quantos trabalhos de cada categoria apresentam abordagem de aprendizado por transferência.

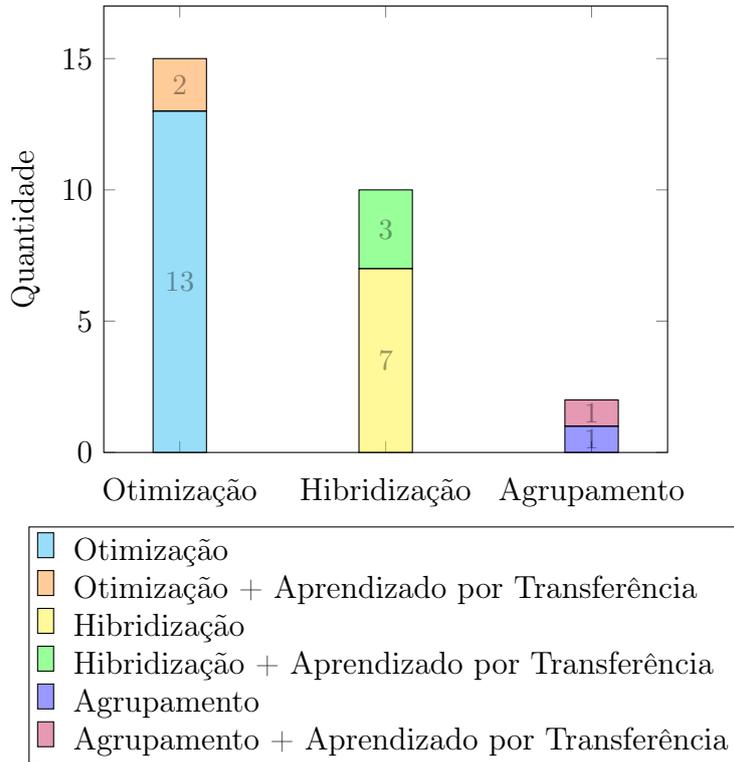


Figura 3.6: Quantidade de abordagens de otimização e hibridização.

Os trabalhos de otimização atingem melhores previsões por abordagens variadas. A primeira é a extração características eficientes através de: *encoders* como (Chang et al., 2022), mecanismos de atenção como (Chen et al., 2022a; Rathore and Harsha, 2022a; Wang et al., 2022b), domínio/janelamento adaptativo com (Deng et al., 2022; Kong and Li, 2022; Qi et al., 2024; Rathore and Harsha, 2022c), algoritmos de *Particle Swarm Optimization* (PSO) como (Rathore and Harsha, 2022b).

A segunda é filtragem de sinais através de: *Empirical Mode Decomposition* (EMD) como (Alfarizi et al., 2023) ou técnica variacional Bayesiana como (Li et al., 2023). A terceira é a clusterização de dados através de clusterização como (Cheng et al., 2022a; Guo and He, 2022). A quarta é o tratamento de incertezas através de um Sistema de inferência com *dropout* de Monte Carlo como (Zhu et al., 2022). E a quinta é a fusão de características através de métodos de ponderamento como (Wang et al., 2023).

Os trabalhos de Hibridização atingem melhores previsões pela fusão de modelos variados. (Deng et al., 2023) une um modelo físico-dinâmico com cinco graus de liberdade a uma rede *Physics-Informed Bayes Deep Dual network PI-BDDN* (PI-

BDDN), enquanto (Zheng et al., 2024) une um modelo físico para informar a taxa de degradação *Physics Degradation Rate-informed* (PDRI) a uma Rede *Multi-feature Spaces Cross Adaption Transformer Network* (MSCATN).

Outros trabalhos usam *autoencoders* ou modelos baseados em *autoencoders* para extração de características e posterior predição como a *Quasi-Recurrent Neural Network* (QRNN) de (Cheng et al., 2022b), a *Transfer Learning-based Time Series Recursive Forecasting* (TL-TSRF) de (Mao et al., 2023), a *Multi-state Temporal Frequency Network* (MSTFN) de (Yang et al., 2024), a *Long Short Term Memory* (LSTM) de (Yao et al., 2024) e o *Deep Reinforcement Learning* (DRL) de (Zheng et al., 2024).

CNNs e LSTMs também podem ser usadas para extração de características, conforme a *Time Delay Neural Network* (TDNN) e LSTM de (Li and Deng, 2023) e (Li and Deng, 2023), respectivamente.

No trabalho de agrupamento de modelos de (Guo et al., 2024), características extraídas no domínio do tempo e no domínio da frequência são submetidas a uma rede Transformer enquanto características extraídas no domínio tempo-frequência a uma CNN, o resultado de cada rede é então concatenado para a realização da predição. Já no trabalho de (Behera and Misra, 2023), características extraídas através da transformação dos sinais em imagens são submetidas a múltiplos modelos de DL, unidas posteriormente em uma camada de fusão ponderada de características e finalmente usadas para predição.

### **3.7.1.2 Trabalhos Focados na Extração de Características/Indicador de Saúde**

Alguns trabalhos focam exclusivamente na extração de características ou criação de um IS que seja eficientemente capaz de prever a RUL, enquanto o algoritmo de predição é preterido. De fato, características ou ISs de características crescente ao longo do tempo influenciam diretamente no desenvolvimento de bons modelos de predição de RUL.

Dadas as características oscilatórias dos sinais de vibração, mesmo extraindo características no domínio da frequência, o tamanho e a posição da janela de dados selecionada sobre uma amostra consecutiva de dados influencia na magnitude das

frequências observadas. Assim, (Ni et al., 2022) propôs um IS derivado da correlação espectral, da distância de Wasserstein e da retificação linear capaz de refletir as mudanças na distribuição probabilística da densidade espectral ao longo do tempo, e contornar as variações cíclicas do sinal.

Por outro lado, outros autores usam das benesses generalizadoras do DL, ignorando totalmente as variações dos sinais de vibração, utilizando dados brutos em seus modelos, e deixando ao modelo a tarefa de lidar com os problemas citados acima. (Xu et al., 2023) usa um modelo de CNN com mecanismo de atenção para adequadamente extrair características ponderadas dos dados. Sequencialmente, as características e seus respectivos pesos são usadas para a construção de um IS monodimensional através da fusão dos resultados.

Diferentemente, (Guo et al., 2024) propõe a extração de características em múltiplos domínios e a criação de um IS criado através da redução de dimensionalidade das energias intrínsecas de cada componente de sinal extraída através de EMD. Finalmente, as características e o IS preliminar são submetidos a um modelo baseado em CNN para a criação de um IS definitivo.

Finalmente, (Gu et al., 2022) extrai características em múltiplos domínios e estabelece um mecanismo de determinação de coeficientes ponderados para métricas de seleção de características (monotonicidade, *trendability*, etc), mantendo apenas as características sensíveis. Essas são fundidas com proposto *Isometric Feature Mapping* (ISOMAP), com o auxílio de um algoritmo baseado *Differential Evolution* (DE), que calcula o peso ideal de cada característica, e cria um IS.

### 3.7.2 Dependência de Dados

Todos os trabalhos apresentados usam modelos treinados e criados a partir de dados históricos pré-existentes, estando restritos a esse pré-requisito. Contudo, tal característica é inviável em condições reais, onde nem sempre há disponibilidade de dados devido a limitações operacionais.

Em cenários industriais reais, a falta ou escassez de dados pode ser dada por equipamentos novos isentos de registros operacionais, mudanças de parâmetros operacionais além do custo de armazenamento e de processamento de grandes volumes de dados.

Dessa forma, os modelos apresentados não conseguem lidar com um dos principais problemas enfrentados pela indústria contemporânea, tendo sua eficácia questionável.

### **3.7.3 Adaptabilidade**

Segundo (Angelov, 2002), a adaptabilidade de um modelo ou sistema é definida como a capacidade de ajustar seus parâmetros em resposta aos novos dados, de forma consecutivamente recursiva. Dessa forma, embora alguns trabalhos usem o termo “adaptativo” ou “adaptação” para se referir à modificação dos parâmetros de seus modelos, a maioria tem o ajuste/alteração de parâmetros desencadeado por algum evento, não sendo verdadeiramente adaptativos.

O que se observa de fato, são métodos de otimização Bayesiana para uma otimização de parâmetros durante a etapa de treino/criação do modelo ou métodos de aprendizado por transferência, nos quais o ajuste de parâmetros é desencadeado pela chegada de novas bases de dados. Todas as referências analisadas têm parâmetros de comportamento estático ou quase estático.

### **3.7.4 Evolutividade**

Segundo (Angelov, 2002), a evolutividade de um modelo ou sistema é definida como a capacidade de ajustar seus parâmetros e sua estrutura em resposta aos novos dados de forma consecutivamente recursiva. Trata-se de uma extensão do conceito de adaptabilidade. Assim, não são observados modelos de característica evolutiva pois nenhum apresenta a capacidade de expandir ou contrair sua estrutura mediante as nuances dos dados.

### **3.7.5 Interpretabilidade**

A maior parte dos trabalhos apresentam modelos de aprendizado profundo para predições de RUL, levando em conta uma etapa de treinamento (dependente de histórico de dados) para posterior criação de um modelo. Contudo, conforme citado anteriormente, contudo esses modelos não apresentam interpretabilidade, necessitando de ferramentas adjacentes capazes de introduzir mecanismos de expli-

cabibilidade às operações dos modelos. O algoritmo *SHapley Additive exPlanations* (SHAP), por exemplo, atribui a cada característica de entrada uma importância (valor *Shapley*) que representa sua contribuição para a previsão de um modelo, permitindo entender o porquê de uma decisão específica (Lundberg and Lee, 2017).

Embora alguns modelos tenham uma abordagem física/matemática capaz de descrever a degradação do rolamento, os valores de degradação calculados são submetidos à algoritmos de previsão baseados em DL, mitigando sua interpretabilidade.

Ainda há algoritmos que usam métricas de “clusterização” para criar indicadores de saúde eficientes, o que poderia proporcionar alguma interpretabilidade à previsão. No entanto, também submetem os valores de entrada a um algoritmo de DL, resultando no mesmo problema citado acima.

### 3.8 Conclusões Acerca da Revisão da Literatura

A revisão permitiu verificar heterogeneidade em diversos processos que levam dos dados à previsão de RUL propriamente dita.

Métodos de pré-processamento de dados diversos são utilizados, podendo ser técnicas de decomposição de sinais sofisticadas (VMD, CWT, EMD) ou simplesmente técnicas de preparação de dados como normalização e escalonamento.

O domínio de processamento de dados também é variado. Embora alguns trabalhos usem simultaneamente todos os domínios, outros usam apenas o domínio do tempo. Isso ocorre devido presença de algoritmos de seleção eficiente de características relevantes e a algoritmos eficientes para extração de características, respectivamente.

Enquanto diversos autores demonstram preocupação quanto ao método de extração de características, recorrendo a múltiplos domínios e técnicas de janelamento dinâmico, alguns o ignoram completamente, deixando a cargo de modelos computacionais a tarefa de extrair características, embora essas não tenham um lastro matemático/estatístico/físico interpretável.

Os métodos de seleção de características apresentados pode ou não ser orientada por indicadores de desempenho. E em alguns trabalhos as características são usadas diretamente para a previsão de RUL, e em outros são fundidas em um IS.

Quanto aos modelos de predição observados, embora novos a maioria são: modelos pré-existentes otimizados/adaptados (seja por otimização Bayesiana, processamento dinâmico de sinais ou clusterização), modelos híbridos (formados pela combinação de dois ou mais modelos) ou modelos formados por aprendizado por agrupamento. De qualquer forma, observa-se que a etapa de predição dos algoritmos proposto é predominantemente realizada por modelo de DL baseados LSTMs e CNNs.

Em nenhum dos trabalhos se verificou modelos dotados de mecanismos para ajuste de parâmetros e/ou estrutura em tempo real, nem aprendizado contínuo, enquadrando os algoritmos verificados nesta revisão como não-evolutivos. Além disso, verificou-se modelos dependentes de treinamento com dados pré-existente em todas as publicações. Ademais, na revisão bibliográfica apresentada, não se verifica registros de modelos evolutivos aplicados à predição de RUL em rolamentos, embora tenham sido utilizados para predições em outras séries temporais genéricas, (Maciel et al., 2016), (Soares et al., 2018), (Leite and Škrjanc, 2019) e (Wang et al., 2022a)

Por fim, verifica-se a pertinência do desenvolvimento de algoritmos evolutivos para prognóstico, não apenas para suprir a escassez de publicações da área, mas também para lidar com boa parte das desvantagens e limitações apresentadas pelos algoritmos ao longo desta revisão.

### 3.9 Prognóstico Evolutivo de Falhas

Dada a ausência de abordagens evolutivas para cálculo de RUL na revisão realizada segundo os filtros da tabela 3.1, realizou-se uma busca específica para uso *Evolving Models* (EMs) em prognóstico de falhas abrangendo outros equipamentos além dos rolamentos. As tabela 3.12 lista as publicações verificadas. Entre as abordagens para execução de prognóstico evolutivo, observou-se a predominância da abordagem *fuzzy* e uma única ocorrência de abordagem por Filtro de Partículas (FP).

A abordagem *Particle Filter and Differential Evolution* (PF-DE) proposta por Cosme et al. (2018) utiliza um FP com operadores genéticos baseados em evolução diferencial para mitigar a perda de diversidade das partículas, introduzindo

Referência	Modelo Evolutivo Proposto
Cosme et al. (2018)	<i>Particle Filter and Differential Evolution</i>
Camargos et al. (2020)	<i>Error Based Evolving Takagi-Sugeno Model</i>
Camargos et al. (2021)	
Boutrous et al. (2022a)	<i>Evolving Ellipsoidal Fuzzy Information Granule</i>
Boutrous et al. (2022b)	

Tabela 3.12: Publicações acerca do prognóstico evolutivo de falhas.

novos operadores de mutação e seleção. O resultado se mostrou eficiente para o prognóstico de falhas em rolamentos.

Os demais autores optam por modelos baseados em *fuzzy* para a inserir interpretabilidade aos modelos. O modelo evolutivo *Error Based Evolving Takagi-Sugeno Model* (EBETS) de Camargos et al. (2020) tira vantagem do aprendizado *offline* para agregação de conhecimento e posteriormente é capaz de realizar boas previsões de RUL de rolamentos e respectiva quantificação de incerteza associadas. Já em (Camargos et al., 2021) uma proposta similar é utilizada para realizar o prognóstico de falhas de baterias.

Posteriormente, (Boutrous et al., 2022a,b) propuseram o modelo *Evolving Ellipsoidal Fuzzy Information Granules* (EEFIG), onde a abordagem *fuzzy* foi utilizada para a criação de GIs aliados a um preditor do tipo RLS para estimar degradação e respectiva incerteza de múltiplos passos a frente, fornecendo um modelo de prognóstico interpretável de falhas para previsão de RUL em transistores e outro semi-condutores.

Salvo Cosme et al. (2018), os demais autores propuseram consequentes/preditores baseados em uma soma ponderada de previsões calculadas pelo algoritmo RLS. Contudo, tal método trata-se de uma versão adaptativa de regressão linear, o que eventualmente torna os modelos propostos frágeis diante da não linearidade característica dos dados relativos à degradação de equipamentos.

### 3.10 Modelo proposto

A capacidade de lidar com não linearidades nos dados sob uma estrutura mutável, com parâmetros atualizáveis iterativamente, ao mesmo tempo em que fornece previsões interpretáveis, torna o EM uma abordagem apropriada para lidar com

prognósticos de falhas em máquinas, cujo objetivo é estimar sua RUL com base em dados caracterizados por incerteza, variação dinâmica e condições operacionais não estacionárias.

Apesar das características versáteis e do potencial promissor demonstrado pelos modelos evolutivos, sua aplicação em prognósticos de falhas e estimativa de RUL permanece relativamente pouco explorada na literatura. Embora alguns estudos tenham destacado a evolução de sistemas para modelagem de prognósticos de falhas em contextos como mancais, baterias e semicondutores, o número de pesquisas e comparações com registro do desempenho de modelos evolutivos ainda é limitado, evidenciando a escassez de publicações na área.

Este trabalho explora novos modelos granulares evolutivos baseado no TEDA. Os modelos propostos aproveitam a abordagem fuzzy simplificada fornecida pelo TEDA para garantir predições interpretáveis. Ao incorporar consequentes associados aos GIs criados pelo AutoCloud, a explicabilidade das predições realizadas poderá ser analisada do ponto das variações estruturais observadas.

Considerando a ausência do *framework* TEDA aplicados à predição de RUL, também serão avaliados diferentes tipos de consequente associados ao AutoCloud. Abordagens baseadas em RLSVDF, RBFs e RNN serão avaliadas como possíveis candidatos a preditores do modelo, ao invés do então unicamente usado RLS.

O RLSVDF estende o RLS melhorando seu desempenho em ambientes não estacionários através de uma matriz de “direção de esquecimento” que guia o quanto o algoritmo deve confiar nos dados passados versus os mais recentes. Uma forma de potencializar ainda mais sua capacidade preditiva é aplicar RBFs às entradas do modelo. Essa transformação não linear permite mapear os dados originais para um espaço de características de maior dimensão, onde relações complexas podem ser aproximadas linearmente. Com isso, o RLSVDF passa a operar de forma mais eficiente mesmo diante de comportamentos altamente não lineares nos dados, ampliando sua capacidade de generalização e adaptação a regimes variáveis do sistema monitorado.

Ao incorporar as capacidades de memória temporal das RNNs, um modelo evolutivo torna-se capaz de reter e explorar informações históricas embutidas em dados de séries temporais, permitindo-lhe acompanhar padrões graduais de variação

e capturar dinâmicas de sistemas em evolução. Essa integração aprimora tanto a adaptabilidade quanto a explicabilidade do modelo, tornando-o especialmente adequado para fluxos de dados em tempo real e prognósticos de falhas em ambientes industriais não estacionários e incertos.

# Capítulo 4

## Metodologia Proposta

Este capítulo discorre acerca das etapas consecutivas a serem realizadas alcançar os objetivos deste trabalho: O desenvolvimento de um modelo de predição evolutivo para cálculos de prognóstico.

Por meio da revisão bibliográfica realizada neste trabalho e em outras, (Lei et al., 2018), (Polverino et al., 2023) e (Ramezani et al., 2019), pôde-se identificar procedimentos comuns aos trabalhos e estabelecer uma sequência de etapas para a realização do cálculo de prognóstico, servindo também de guia para as discussões apresentadas ao longo deste capítulo. A Figura 4.1 expõe a sequência de etapas propostas.

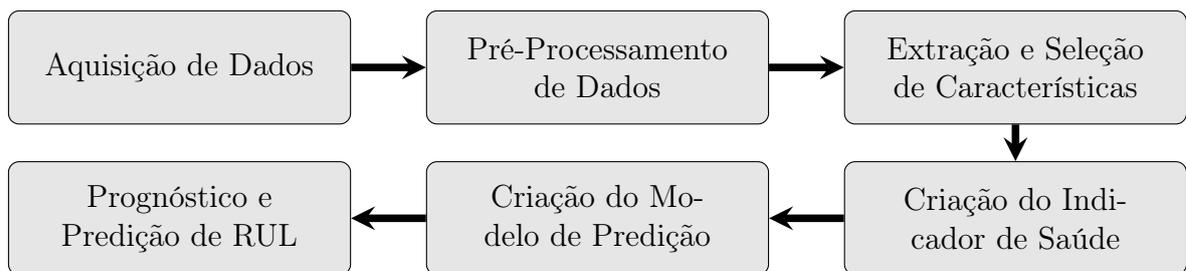


Figura 4.1: Etapas de realização do prognóstico.

### 4.1 Aquisição de Dados

A aquisição de dados é a primeira etapa para realização de prognóstico, sendo definida como o processo de registro e armazenamento de diferentes tipos de parâmetros de funcionamento medidos através de vários sensores instalados no equipamento

monitorado, (Lei et al., 2018).

Neste trabalho, serão utilizados dados de vibração mecânica de rolamentos em motores elétricos disponibilizados publicamente em repositórios *online*. Serão utilizadas as bases de dados XJTU (Wang et al., 2018) e Pronostia (Nectoux et al., 2012), cujas citações e características foram apresentadas na Seção 3.2 do Capítulo 3.

#### 4.1.1 Rolamentos

Em diversas máquinas rotativas modernas, rolamentos se destacam entre os componentes mecânicos mais críticos, influenciando diretamente no desempenho do equipamento e seus componentes adjacentes, tornando-se objeto de estudo de diversos trabalhos relativos ao prognóstico, (Nectoux et al., 2012; Wang et al., 2017).

Seu propósito é a redução da fricção rotacional e suportação de cargas axiais e radiais. Resumidamente, rolamentos são formados por elementos rolantes fixados entre dois anéis e mantidos equidistantes por uma peça chamada gaiola. A figura 4.2 a seguir mostra um rolamento e seus componentes.

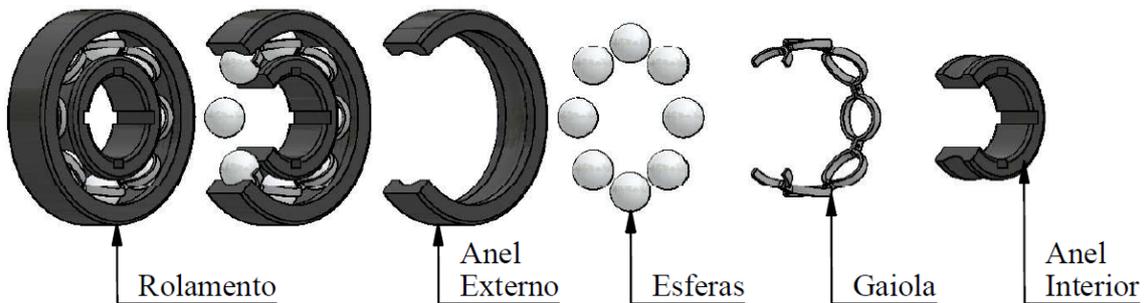


Figura 4.2: Rolamento e seus componentes em vista explodida.

Normalmente, o anel interno do rolamento é fixado sobre um eixo rotativo enquanto seu anel externo é apoiado sobre um compartimento específico da máquina. O funcionamento se dá pela rotação do conjunto formado pelo eixo mais anel interno e pela translação dos elementos rolantes. Inevitavelmente, o atrito do deslizamento das esferas sobre as pistas dos anéis interno e externo provoca fricção entre as superfícies de contato dos componentes, desgastando-os. A consequência é o surgimento de falhas como fissuras, erosão e deformação dos componentes, provocando aumento de temperatura e vibração, etc. A Figura 4.3 exemplifica a montagem de rolamento num motor elétrico.

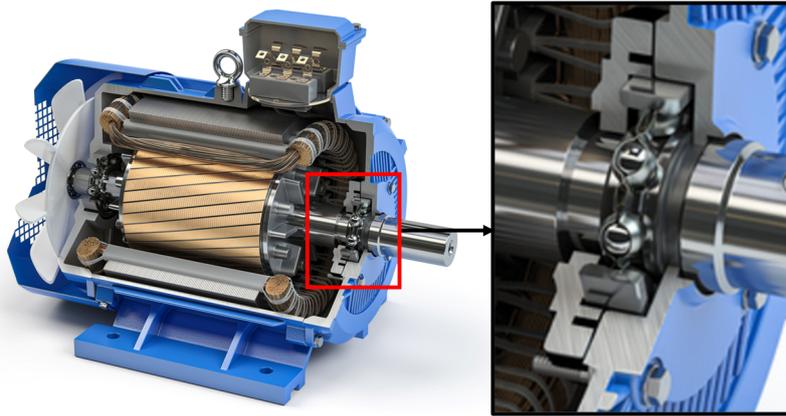


Figura 4.3: Motor elétrico em vista cortada. Adaptado de (Thomas, 2024)

No geral, o mapeamento de falhas de rolamentos é simples, podendo ou não ocorrer isoladamente em qualquer componente, e causadas por falhas de montagem, atrito, corrosão etc. As figuras a seguir adaptadas de mostram os possíveis defeitos citados.

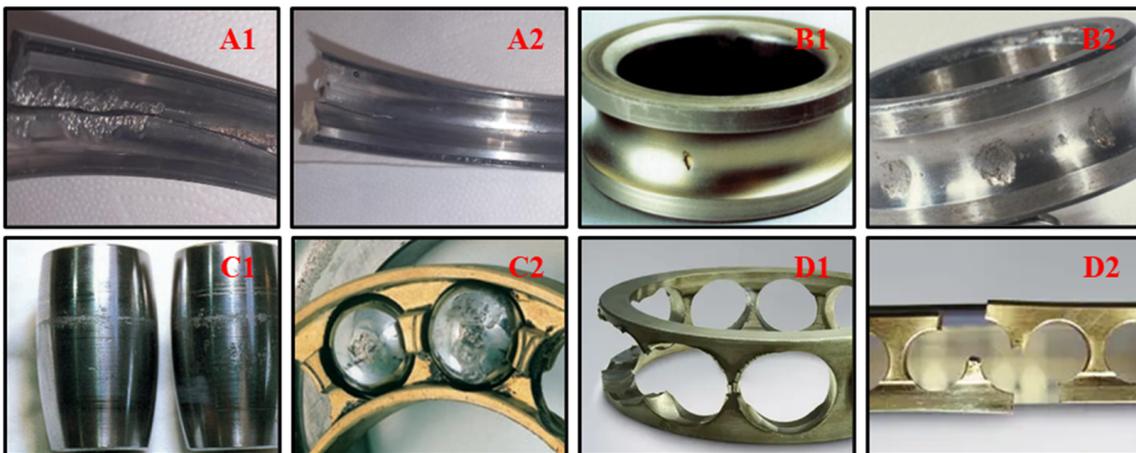


Figura 4.4: Falhas em: A - anel externo, B - anel interno, C - elemento rolante, D - gaiola. Adaptado de (Dario, 2024; James, 2024; NSK, 2024; NTN, 2024).

Em todos os cenários citados, a principal consequência das falhas de rolamentos em máquinas rotativas é o aumento da vibração mecânica. Naturalmente, o impacto da colisão dos elementos rolamentos com as irregularidades superficiais mostradas nas Figuras acima provoca o aumento da vibração, podendo aumentar ainda mais de acordo com a dimensão da falha.

## 4.2 Pré-processamento de Dados

Será feita uma filtragem de sinal para a remoção das componentes de vibração irrelevantes, restando apenas as informações pertinentes ao rolamento. Para isso, será utilizado o método proposto por (Li et al., 2020), onde a entropia das bandas de frequência guiam a filtragem de sinal para evidenciar apenas as vibrações emitidas pelo rolamento.

O método propõe um algoritmo capaz de determinar a componente mais relevante de um sinal de vibração a partir de uma decomposição de sinais utilizando a ferramenta VMD (apresentada na seção 2.2.1 do capítulo 2) aliada a um mecanismo de otimização para a definição da quantidade ideal de IMFs de decomposição e a uma ferramenta baseada na entropia da banda de frequência, do inglês FBE, para seleção da IMFs mais relevante.

### 4.2.1 Entropia da Banda de Frequência - FBE

Proposta por (Liu et al., 2013) a FBE mede a distribuição de energia nas diferentes bandas de frequência de um sinal de vibração e, ao calcular a entropia de cada uma dessas bandas, pode-se identificar padrões que indicam o estado de saúde do rolamento. O método segue as seguintes etapas:

#### 1. Análise no Domínio Tempo-Frequência:

O cálculo da FBE é permeado pela Densidade Espectral, onde é possível observar as variações das amplitudes de cada banda de frequência ao longo do tempo. Para isso, um sinal  $x(t)$  é submetido à STFT, calculada conforme (4.1).

$$\text{STFT}\{x(t)\} = X(\tau, \omega) = \int_{-\infty}^{+\infty} x(t) \omega(t - \tau) e^{-j\omega t} dt \quad (4.1)$$

Onde  $N$  é a quantidade de amostras,  $\omega(t)$  a janela da função, comumente a janela de Hanning, e  $\tau$  o deslocamento temporal. Porém, para fins de compreensão, o resultado da STFT de um sinal  $x(t) = (x_1, x_2, \dots, x_N)$  será dado por (4.2).

$$\text{STFT}\{x(t)\} = \begin{bmatrix} r_{1,1} & \cdots & r_{1,C} \\ \vdots & \cdots & \vdots \\ r_{M,1} & \cdots & r_{M,C} \end{bmatrix} \quad (4.2)$$

Onde  $r$  é a amplitude da STFT na frequência,  $M$  o ponto de frequência,  $C = N/L$  a quantidade de Transformadas de Fourier realizadas no intervalo de tempo da amostragem, e  $L$  a quantidade de amostras que a janela de Fourier se move a cada passo para capturar a evolução do sinal ao longo do tempo.

Por conseguinte, é possível descrever a amplitude de cada componente de frequência  $f_k$  ao longo do tempo, onde  $k = (1, 2, \dots, M)$ , conforme (4.3).

$$X(f_k) = (r_{k,1}, r_{k,2}, \dots, r_{k,C}) \quad (4.3)$$

## 2. Densidade Espectral Média:

Em sequência, calcula-se a densidade espectral média por (4.4).

$$P_{X(f_k)} = \frac{1}{N} \sum_{F=1}^N X(f_k; F)^2 \quad (4.4)$$

Onde  $F$  é o espectro da componente de frequência  $f_k$  no domínio tempo-frequência.

## 3. Entropia da Amplitude Espectral:

A terceira etapa consiste em estimar entropia da amplitude espectral de cada componente de frequência  $f_k$  de acordo com as equações (4.5), (4.6) e (4.7).

$$H_{sk} = \frac{-\left(\sum_{m=1}^C p_{m,k} \ln(p_{m,k})\right)}{\ln(C)} \quad (4.5)$$

$$p_{m,k} = \frac{P_{X(f_k)}(F_m)}{\sum_{n=1}^C P_{X(f_k)}(F_n)} \quad (4.6)$$

$$\sum_{m=1}^C p_{m,k} = 1 \quad (4.7)$$

## 4. Cálculo da Entropia da Banda de Frequência:

Após o cálculo da entropia da amplitude espectral de cada componente no domínio tempo-frequência, a FBE é definida conforme a por (4.8).

$$H_{sf} = (H_{s1}, H_{s2}, \dots, H_{sM}) \quad (4.8)$$

O termo  $p_{m,k}$  descreve o percentual da  $m$ -ésima componente de frequência  $f_k$  na densidade espectral. Particularmente, a mudança de energia supostamente varia periodicamente; então o valor da FBE será menor, ou maior se a energia da banda de frequência oscilar randomicamente no domínio tempo-frequência.

### 5. Definição da Frequência Central:

A FBE pode ser utilizada para estimar o sinal ciclostacionário e identificar a banda de ressonância do sinal. A ferramenta é empregada no processamento de sinais para identificar um filtro de detecção, cuja frequência central é obtida na frequência onde o valor do FBE é mínimo, conforme as equações (4.9) e (4.10).

$$H_{sf}^{\min} = \min(H_{sf}) = \min(H_{s1}, H_{s2}, \dots, H_{sM}) \quad (4.9)$$

$$f_{H_{sf}^{\min}}^j = \arg \min(H_{sf}^{\min}) \quad (4.10)$$

Como a FBE é calculada com base na STFT, a frequência central  $f_{H_{sf}^{\min}}^j$  é influenciada pelo comprimento da janela  $N_w$ . Assim para confirmar a frequência central, diferentes entropias são estimadas com base em diferentes comprimentos de janela, sempre associando a frequência central à menor entropia.

### 6. Definição do Filtro Passa Banda:

A banda de filtragem é calculada por (4.11).

$$\Delta f_{H_{sf}^{\min}} = \left[ \frac{2f_{H_{sf}^{\min}}^j - \Delta f_0}{2}, \frac{2f_{H_{sf}^{\min}}^j + \Delta f_0}{2} \right] \quad (4.11)$$

Onde  $\Delta f_0$  representa a banda de alcance do filtro, dependente do valor ideal de  $N_w$  e da frequência de amostragem  $f_s$  do sinal, conforme (4.12).

$$\Delta f_0 \approx \frac{1.5f_s}{N_w} \quad (4.12)$$

O pseudocódigo do método proposto acima por (Liu et al., 2013) está registrado no Algoritmo 8.

---

**Algoritmo 8** Entropia da Banda de Frequência - FBE

---

- 1: Entrada:  $x_h = (x_1, x_2, \dots, x_n)$
  - 2: Calcular:  $\text{STFT}\{x(t)\} = \begin{bmatrix} r_{1,1} & \dots & r_{1,C} \\ \vdots & \dots & \vdots \\ r_{M,1} & \dots & r_{M,C} \end{bmatrix}$  pelas equação 4.1
  - 3: **para** cada  $f_k$  **faça**
  - 4:   Calcular:  $P\{X(f_k)\}$  pela equação 4.4
  - 5: **fim para**
  - 6: **para** cada  $P\{X(f_k)\}$  **faça**
  - 7:   Calcular:  $H_{sk} = (H_{s1}, H_{s2}, \dots, H_{sM},)$  pelas equações (4.5), (4.6) e (4.7)
  - 8: **fim para**
  - 9: Definir:  $f_{H_{\min}^j}$  pelas equações (4.9) e (4.10)
  - 10: Calcular:  $\Delta f_{sf}^{\min}$  pelas equações (4.11) e (4.12)
  - 11: Saída:  $\Delta f_{sf}^{\min}$
- 

## 4.2.2 VMD Otimizado pela FBE

Com base na FBE, (Li et al., 2020) apresenta um método para a realização de um VMD otimizado, com a quantidade correta do número de componentes  $k$  e a seleção da componente mais relevante para a detecção e diagnóstico de falhas em rolamentos.

O método segue as seguintes etapas:

### 1. Determinação número de modos $k$ ideal do VMD:

Dado um sinal, o fator de penalidade  $\alpha$  e a largura de banda  $\tau$ , parâmetros do VMD, são inicializadas em  $\alpha = 2000$  e  $\tau = 0$ , respectivamente. Sequencialmente, o VMD é aplicado para valores  $k$  consecutivamente crescentes num intervalo finito onde  $k \in [2, 15]$  com incrementos unitários, (Tang and Wang, 2016).

Posteriormente, calcula-se a curtose do envelope para cada modo de cada VMD aplicado. A maior curtose do envelope de cada VMD é registrada para comparação.

Assumindo que o número de modos  $u_k$  do VMD  $k \in [2, 15]$ , cada envelope pode ser calculado por:

$$\tilde{u}_k(t) = \left| \frac{1}{\pi t} u_k(t) \right|, k = 1, 2, \dots, K \quad (4.13)$$

Onde,  $k$  representa o  $k$ -ésimo modo de  $K$ ,  $\tilde{u}_k(t)$  é o valor absoluto resultante da Transformada de Hilbert de cada modo  $k$  e  $u_k(t)$  representa o modo  $i$  gerado quando o número de modos é  $K$ .

Em seguida, a curtose  $\mathcal{K}$  do envelope de cada  $k$ -ésimo modo de  $K$  é calculada por:

$$\mathcal{K}(i) = \frac{E.(\tilde{u}_k - \mu(\tilde{u}_k))^4}{\sigma(\tilde{u}_k)^4}, k = 1, 2, \dots, K \quad (4.14)$$

Onde,  $\mu(\tilde{u}_k)$  é a média de  $\tilde{u}_k$ ,  $\sigma(\tilde{u}_k)$  é o desvio padrão de  $\tilde{u}_k$  e o numerador  $E.(\tilde{u}_k - \mu(\tilde{u}_k))^4$  da equação (2.11) é o momento central de quarta ordem de  $\tilde{u}_k$ . Para cada  $k$ , o conjunto de curtoses para cada modo é:

$$\bar{\mathcal{K}}_k = (\mathcal{K}_1, \mathcal{K}_2, \dots, \mathcal{K}_k) \quad (4.15)$$

Por 4.14, podemos definir um máximo valor local de curtose de envelope para cada VMD tal que:

$$\mathcal{K}_{\text{máx}_k} = \max(\mathcal{K}_1, \mathcal{K}_2, \dots, \mathcal{K}_k) \quad (4.16)$$

Finalmente, tem-se um conjunto de máximos locais de curtose de envelope para cada VMD conforme:

$$\bar{\mathcal{K}}_{\text{máx}_k} = (\mathcal{K}_{\text{máx}_1}, \mathcal{K}_{\text{máx}_2}, \dots, \mathcal{K}_{\text{máx}_k}), k \in [2, 15] \quad (4.17)$$

O máximo global é então obtido por:

$$\mathcal{K}_{\text{máx global}} = \max(\bar{\mathcal{K}}_{\text{máx}_k}) = \max(\mathcal{K}_{\text{máx}_1}, \mathcal{K}_{\text{máx}_2}, \dots, \mathcal{K}_{\text{máx}_k}), k \in [2, 15] \quad (4.18)$$

O valor de  $k$  associado aos máximo global das curtoses de envelope é tido como o valor ideal  $k_{\text{ideal}}$  para o número de decomposições do VMD.  $k_{\text{ideal}}$  é obtido por:

$$k_{\text{ideal}} = \arg \max(\mathcal{K}_{\text{máx global}}) \quad (4.19)$$

## 2. Cálculo das FBEs:

Seguindo o algoritmo proposto por (Liu et al., 2013), calcula-se a FBE do sinal original  $x(t)$  e das respectivas IMFs  $u_k$  associadas à decomposição promovida pelo VMD.

Conforme (2.11), para cada componente IMF $_j$  proveniente da decomposição do sinal  $x(t)$  em  $k_{\text{ideal}}$  modos temos que:

$$x(t) = \sum_{j=1}^{k_{\text{ideal}}} u_j = u_1 + u_2 + \dots + u_{k_{\text{ideal}}} \quad (4.20)$$

A FBE é então aplicada a cada uma das IMFs resultando, para cada uma delas, na entropia de cada banda de frequência do sinal conforme a equação a seguir:

$$\bar{\mathcal{E}}_{j\mathcal{LF}} = (\mathcal{E}_{j\mathcal{L}1}, \mathcal{E}_{\mathcal{L}j2}, \dots, \mathcal{E}_{j\mathcal{L}N}), \quad j \in [1, k_{\text{ideal}}] \quad (4.21)$$

Onde,  $j$  representa a  $j$ -ésima IMF do VMD selecionado. O termo  $\mathcal{LF}$  representa a  $\mathcal{F}$ -ésima linha de frequência consecutivas da  $j$ -ésima IMF, onde  $\mathcal{L} = 1, 2, \dots, N$  representa as janelas de frequência consecutivas resultado do janelamento da STFT e  $\mathcal{F}$  é a frequência natural da divisão.

## 3. Seleção do modo (IMF) mais relevante do VMD:

Dadas as FBEs de cada IMF e do sinal original, verificam-se as menores entropias calculadas e suas respectivas frequências conforme a equação a seguir:

$$\mathcal{E}_{j_{\text{min}}} = \min(\mathcal{E}_{1\mathcal{LF}}, \mathcal{E}_{2\mathcal{LF}}, \dots, \mathcal{E}_{j\mathcal{LF}}), \quad j \in [1, k_{\text{ideal}}] \quad (4.22)$$

As frequências associadas às menores entropias são tratadas como as frequên-

cias centrais para a aplicação de um filtro passa banda. E então a largura de banda  $\Delta f_0$  e  $\Delta f^j$ , respectivamente associadas à  $x(t)$  e  $IMF_j$ , são obtidas por:

$$\Delta f_0 \approx 1.5f_s/N_w^0 \quad (4.23)$$

$$\Delta f^j \approx 1.5f_s/N_w^{j_{ideal}} \quad (4.24)$$

Onde  $f_s$  é a frequência de amostragem do sinal, e  $N_w^0$  e  $N_w^{j_{ideal}}$  são os tamanhos ótimos das janelas usadas nas STFTs dos sinais durante a aplicação da FBE. Tem-se então que o intervalo banda de frequência de filtragem de  $x(t)$  e de cada IMF são, respectivamente:

$$\Delta f_{x(t)} = \left[ \frac{2f_{\varepsilon_{0_{\min}}} - \Delta f_0}{2}, \frac{f_{\varepsilon_{0_{\min}}} + \Delta f_0}{2} \right] \quad (4.25)$$

$$\Delta f_{IMF_j} = \left[ \frac{2f_{\varepsilon_{j_{\min}}} - \Delta f^j}{2}, \frac{2f_{\varepsilon_{j_{\min}}} + \Delta f^j}{2} \right] \quad (4.26)$$

Onde  $f_{\varepsilon_{0_{\min}}}$  e  $f_{\varepsilon_{j_{\min}}}$  são as frequências das menores entropias calculadas para  $x(t)$  e  $IMF_j$ . Finalmente, deve-se constatar o intervalo (4.25) como um subconjunto de (4.26), conforme (4.27).

$$\left[ \frac{2f_{\varepsilon_{j_{\min}}} - \Delta f^j}{2}, \frac{2f_{\varepsilon_{j_{\min}}} + \Delta f^j}{2} \right] \subset \left[ \frac{2f_{\varepsilon_{0_{\min}}} - \Delta f_0}{2}, \frac{2f_{\varepsilon_{0_{\min}}} + \Delta f_0}{2} \right] \quad (4.27)$$

Caso a constatação de (4.27) seja verdadeira, a componente  $IMF_j$  associada ao intervalo é tida como a mais relevante, denotando-a por  $u_{ideal}$ .

A IMF mais relevante  $u_{ideal}$  é capaz de evidenciar as falhas em rolamento com mais eficiência quando comparada ao sinal original, dessa forma o envelope de Hilbert dessa IMF será tratada então como dados de entrada para as etapas subsequentes. O pseudo código do método proposto acima por (Li et al., 2020) é baseada nos trabalho de (Liu et al., 2013) e (Li et al., 2019), e está registrado no Algoritmo 9.

---

**Algoritmo 9** Módulo de Pré-Processamento de Sinal

---

```
1: Procedimento: PréProcessamento( $x(t), \alpha, \tau, k_{\text{máx}}$ )
2: para  $k$  de 2 até  $k_{\text{máx}}$  faça
3:   Calcular:  $\text{VMD}(x_h, \alpha, \tau, k) = (u_1, \dots, u_k)$  pelo Algoritmo 3
4:   para todo  $u_i$ , onde  $i \in [1, k]$  faça
5:     Calcular:  $\mathcal{K}(i)$  por (4.13) e (4.14)
6:     Definir:  $\mathcal{K}_{\text{máx}_k}$  por (4.16)
7:   fim para
8: fim para
9: Definir:  $k_{\text{ideal}}$  por (4.17), (4.18) e (4.19)
10: para todo  $u_i$ , onde  $i \in [1, k_{\text{ideal}}]$  faça
11:   Calcular: FBE  $\mathcal{E}_{j_{\text{min}}}$  pelo Algoritmo 8
12: fim para
13: Definir:  $E_{\text{min}}^j$  por (4.22)
14: Calcular:  $\Delta f_{x_h}$  e  $\Delta f_{\text{IMF}_j}$  por (4.25) e (4.26)
15: se  $\Delta f_{\text{IMF}_{h,j}} \subset \Delta f_{x_h}$  então
16:    $j_{\text{ideal}} \leftarrow \arg \min(\mathcal{E}_{j_{\text{min}}})$ 
17:    $\text{Modo}_{\text{ideal}}$  ou  $u_{\text{ideal}} \leftarrow \text{Modo}_{h,j_{\text{ideal}}}$ 
18: senão
19:   Desconsiderar a primeira  $\mathcal{E}_{j_{\text{min}}}$  repetir as linhas 13 à 17
20: fim se
21: Saída:  $u_{\text{ideal}}$ 
```

---

### 4.3 Da Extração à Seleção de Características

Seguindo as metodologias observadas na revisão da literatura exposta no capítulo 3, este trabalho propõe a extração características no domínio do tempo e no domínio da frequência, as quais serão submetidas aos critérios de seleção propostos mais adiante.

#### 4.3.1 Características Extraídas

O estado de degradação será avaliado por 16 características extraídas por métodos estatísticos nos domínios do tempo e da frequência, conforme as proposições de (Yang et al., 2024). O cálculo de cada característica pode ser observado nas Tabelas 4.1 e 4.2, respectivamente.

#### 4.3.2 Critério de Seleção de Características

O desempenho do modelo de predição de RUL está diretamente associada às características, fazendo necessário o uso de métricas de avaliação das características para a seleção das mais relevantes, conforme apontado por (Qi et al., 2024).

Nome	Fórmula	Nome	Fórmula
Média	$\mu = \frac{1}{N} \sum_{i=1}^N x_i$ (4.28)	Média Absoluta	$\mu_{\text{abs}} = \frac{1}{N} \sum_{i=1}^N  x_i $ (4.29)
Variância	$\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu_{\text{abs}})^2}{N}$ (4.30)	Desvio Padrão	$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \mu_{\text{abs}})^2}{N}}$ (4.31)
Assimetria ( <i>Skewness</i> )	$\mathcal{S} = \frac{1}{N} \sum_{i=1}^N \frac{(x_i - \mu_{\text{abs}})^3}{\sigma^3}$ (4.32)	Kurtose	$\mathcal{K} = \frac{1}{N} \sum_{i=1}^N \frac{(x_i - \mu_{\text{abs}})^4}{\sigma^4}$ (4.33)
Entropia	$\mathcal{E} = - \sum_{i=1}^N [P(x_i) \log(x_i)]$ (4.34)	RMS	$\text{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$ (4.35)
Máximo	$\text{Máx} = \max\{ x_i \}$ (4.36)	Pico-a-pico	$\text{Pico} = \max\{ x_i \} - \min\{ x_i \}$ (4.37)

Tabela 4.1: Características extraídas no domínio do tempo.

#### 4.3.2.1 Monotonicidade

Em operações reais, a degradação de uma máquina é irreversível e gradativa, sendo um componente em falha incapaz de recuperar suas plenas condições sem intervenção mantenedora, apresentando tendência de crescimento monotônica.

Segundo (Bejaoui et al., 2020), a monotonicidade pode ser medida pela diferença absoluta entre o número de derivadas positivas e negativas para cada característica conforme:

$$M_i(f_i) = \frac{1}{K-1} \left| \text{No.of} \frac{d}{df_i} > 0 - \text{No.of} \frac{d}{df_i} < 0 \right| \quad (4.46)$$

Onde  $M_i$  é a monotonicidade para a  $i$ -ésima característica  $f_i$  com comprimento  $K$ . O valor da diferença entre o número de positivos  $\text{No.of} \frac{d}{df_i} > 0$  e negativos  $\text{No.of} \frac{d}{df_i} < 0$  retorna o valor de monotonicidade, de 0 a 1.

Nome	Fórmula	Nome	Fórmula
Energia Total	$E_X = \sum_{k=1}^K X_k^2$ (4.38)	Energia Média	$E_m = \frac{1}{N} \sum_{k=1}^K X_k^2$ (4.39)
Variância da Energia	$\sigma_E^2 = \frac{\sum_{k=1}^K (X_k - E_m)^2}{K}$ (4.40)	Desvio Padrão da Energia	$\sigma_E = \sqrt{\frac{\sum_{k=1}^K (X_k - E_m)^2}{K}}$ (4.41)
Assimetria ( <i>Skewness</i> )	$S_E = \frac{1}{K} \sum_{k=1}^K \frac{(X_k - E_m)^3}{\sigma_E^3}$ (4.42)	Kurtose	$\mathcal{K}_E = \frac{1}{K} \sum_{k=1}^K \frac{(X_k - E_m)^4}{\sigma_E^4}$ (4.43)
Concentração de Energia na Banda	$Con = \frac{\sum_{k=j}^{K'=J} X_j^2}{\sum_{k=1}^K X_k^2}$ (4.44)	RMS da Energia	$RMS_E = \frac{1}{K} \sum_{k=1}^K \frac{(X_k - E_m)^4}{\sigma_E^4}$ (4.45)

Tabela 4.2: Características extraídas no domínio da frequência.

#### 4.3.2.2 Trendabilidade

A degradação de um equipamento tende a aumentar à medida que o tempo passa. Assim, a propriedade chamada *trendability* mede a correlação entre o IS e o tempo, (Thoppil et al., 2021). A *trendability* pode ser calculada por:

$$Tre(x, T) = \frac{K \left( \sum_{k=1}^K x_k t_k \right) - \left( \sum_{k=1}^K x_k \right) \left( \sum_{k=1}^K t_k \right)}{\sqrt{\left[ K \sum_{k=1}^K x_k^2 - \left( \sum_{k=1}^K x_k \right)^2 \right] \left[ K \sum_{k=1}^K t_k^2 - \left( \sum_{k=1}^K t_k \right)^2 \right]}} \quad (4.47)$$

Onde  $t_k$  é o instante  $k$  éximo valor de tempo e  $x_k$  é o valor do IS no instante  $t_k$ .  $Tre_1(X, T)$  varia de -1 a 1. Aproxima-se de 1 quando há correlação linear positiva e -1 quando negativa.

## 4.4 Criação do Indicador de Saúde

(Lei et al., 2018) define IS como valores numéricos capazes de quantificar o nível de degradação de um equipamento, dividindo-os em ISs físicos, quando possuem lastro matemático/físico/estatístico, e ISs virtuais quando são formados pela

fusão de múltiplos ISs físicos. Lei et al. (2018) ainda demonstram que alguns trabalhos submetem ao modelo de predição as características extraídas diretamente como ISs individuais, e outros as combinam por meio de algoritmos como Análise da Componente Principal, do inglês *Principal Component Analysis* (PCA), conforme observado na revisão do Capítulo 3.

A revisão da literatura também revelou trabalhos que abordam o uso de ferramentas de DL como *Autoencoders* e *encoders* para criação de ISs, porém sem qualquer significado prático. Contudo, a proposta deste trabalho aborda a alimentação do modelo de predição de RUL com ISs singulares, múltiplos ISs simultaneamente e ISs formados pela fusão de múltiplos ISs através do PCA citado acima.

#### 4.4.1 Análise da Componente Principal

A análise da componente principal, é um método computacional amplamente para compressão de dados e extração de características. Do ponto de vista estatístico, o objetivo do PCA é reduzir a redundância entre variáveis randômicas que compõem um vetor randômico  $x \in R^m$ , o que é medido pelas correlações entre elas.

Nesse sentido, o PCA desvincula as características e reduz a dimensionalidade, encontrando novas características que são combinações lineares da originais. Suas formulações podem ser encontradas em (Hotelling, 1933) e a implementação proposta por (Levada, 2021) no Algoritmo 10.

---

#### Algoritmo 10 Análise da Componente Principal

---

- 1: **Entrada:** Dados  $X \in \mathbb{R}^{n \times m}$  com  $n$  amostras e  $m$  variáveis.
- 2: **Saída:** Componentes principais PCA  $\in \mathbb{R}^{n \times d}$ .
- 3: Calcular a média amostral:

$$\mu_x = \frac{1}{n} \sum_{i=1}^n x_i$$

- 4: Calcular a matriz de covariância:

$$\Sigma_x = \frac{1}{n-1} \sum_{i=1}^n (x_i - \mu_x)(x_i - \mu_x)^T$$

- 5: Calcular os autovalores e autovetores de  $\Sigma_x$ .
- 6: Definir a matriz de transformação  $T = [w_1, w_2, \dots, w_d]$ , com os  $d$  autovetores associados aos maiores autovalores.
- 7: Projetar os dados  $X$  no subespaço PCA:

$$y_i = Tx_i \quad \text{para } i = 1, 2, \dots, n$$

- 8: Retornar PCA, onde  $\text{PCA} = [y_1, y_2, \dots, y_n]$ .
-

---

**Algoritmo 11** Módulo de Extração de Características e Criação do Indicador de Saúde

---

- 1: **Procedimento:**  $\text{CalcularIS}(u_{\text{ideal},h}, fs)$
  - 2:  $\text{Características}_h \leftarrow []$
  - 3: Calcular características pelas equações (4.28) à (4.45)
  - 4: **para** Cada Equação em [Equação (4.28), Equação(4.45)] **faça**
  - 5:      $\text{Características}_h \leftarrow \text{Equação}(u_{\text{ideal},h})$
  - 6: **fim para**
  - 7: **Fundir**  $\text{Características}_h$  pelo Algoritmo 10
  - 8:  $\text{IS}_h \leftarrow \text{PCA}(\text{Características}_h)$
  - 9: **Saída:**  $\text{IS}_h$
- 

O Algoritmo 11 descreve o módulo responsável por receber um sinal de vibração, extrair suas características e transformá-lo em um ISs utilizando o PCA proposto acima.

#### 4.4.2 Estágio de Saúde

O Estágio de Saúde (ES) ou Estágio de Degradação (ED) vinculado ao ISs também influencia no desempenho do modelo de predição de RUL. Assim, o processo de degradação pode ser dividido em diferentes ESs de acordo com a tendência apresentada pelo ISs antes do cálculo da RUL, garantindo uma predição eficiente pela associação do ISs ao seu ES. (Li et al., 2023) propõe uma técnica variacional Bayesiana para reconhecer o ES adaptativamente enquanto (Wang et al., 2023) propõe um processo de clusterização fuzzy definição do ES.

Neste trabalho o estágio de saúde será quantificado por um Indicador de Estágio de Degradação (IED), calculado utilizando a correlação de Pearson (4.48), conforme proposto por (Wang et al., 2015) e (Lei et al., 2018).

$$\text{Corr}(x_k) = \frac{\sum_{k=1}^K (x_k - \bar{x})(x_1 - \bar{x}_1)}{\sqrt{\sum_{k=1}^K (x_k - \bar{x})^2 (x_1 - \bar{x}_1)^2}} \quad (4.48)$$

Onde  $x_k$  e  $x_1$  são séries temporais dos valores de vibração no instante  $k$ , e  $\bar{x}_k$  e  $\bar{x}_1$  são as médias das séries temporais dos valores de vibração.

### 4.5 Criação do Preditor AutoCloud

Os preditores TEDA fundamentados no TEDACluster, apresentados na seção 2.5.2.3, associam dados a um único grânulo  $\mathcal{G}_i$  referente ao maior nível de ativação granular  $w_{ih}$  calculado por (2.38). Contudo, nas respectivas publicações, os al-

goritmos expostos não apresentam mecanismos de exclusão de grânulos, tendendo unicamente à expansão de sua estrutura ao longo do tempo.

Dessa forma, este trabalho propõe um algoritmo evolutivo para predição de RUL apoiado no TEDA e fundamentado nas métricas granulação estabelecidas pelo AutoCloud de (Bezerra et al., 2020) expostas na Seção 2.5.2.4, proporcionando um algoritmo de predição com estrutura expansível e comprimível diante das variações dos dados, o Preditor AutoCloud (P.AutoCloud).

Adicionalmente, diante da possibilidade de associação entre um estágio de saúde a um respectivo IS, propõe-se também o uso do IED para granulação de dados e o uso dos ISs para predições futuras dos próprios ISs, implicando na alteração da regra (2.37) dos preditores TEDA expostos anteriormente, estabelecendo uma nova Regra (4.49).

$$R_i(X_h): \text{ Se } Z_h \text{ é } \mathcal{G}_i, \text{ então } \hat{Y}_{ih} = g_i(X_h, \mathcal{Q}_i) \quad (4.49)$$

Onde  $Z_h$  e  $X_h$ , definidos pelas equações (4.50) e (4.51), representam respectivamente os valores de ES e ISs do instante  $h$  a  $l$  valores passados. Enquanto  $\hat{Y}_i$ , definido por (4.52), indica a predição granular valores futuros, calculada em função de  $Y_h$  através de uma função/modelo  $g_i$  (de qualquer natureza) com respectivos seus pesos e coeficientes  $\mathcal{Q}_i$ .

$$X_h = [x_{h-l}, \dots, x_h] \quad (4.50)$$

$$Y_h = [\hat{y}_{h-l+1}, \dots, \hat{y}_{h+1}] \quad (4.51)$$

$$Z_{ih} = [Z_{h-l}, \dots, Z_h] \quad (4.52)$$

Caso não seja desejado utilizar uma variável exclusiva para o processo de granulação, ao considerar  $X_h = Y_h$ , a regra exposta em (4.49) adquire o formato da regra exposta em (2.37).

Nos preditores TEDA citados, a função (ou modelo)  $g$  assume a forma linear tal que  $g_i = X_h^\top \cdot \Theta$ , onde  $\Theta$  é calculado por um algoritmo de RLS, contudo este trabalho adota modelos alternativos ao para  $g$ .

Do ponto de vista do aprendizado contínuo, o preditor TEDA, conforme o Algoritmo 7, contempla apenas a atualização dos parâmetros referentes ao grânulo

de maior nível de ativação, enquanto este trabalho propõe a atualização ponderada pelo nível de ativação dos parâmetros de todos os grânulos.

A predição singular  $\hat{Y}_h$  é calculada pela soma de cada predição granular  $\hat{Y}_{ih}$  ponderada pelo seu peso  $w_{ih} = \frac{\tau_{ih}}{\sum_{j=1}^n \tau_{jh}}$ .

$$\hat{Y}_h = \sum_{i=1}^n w_{ih} \hat{Y}_{ih} \quad (4.53)$$

O pseudocódigo exposto pelo Algoritmo 12 demonstra o procedimentos para implementação do Preditor AutoCloud descrito acima com um consequente genérico definido nas próximas seções.

---

**Algoritmo 12** Preditor AutoCloud

---

```

1: Procedimento PreditorAutoCloud ( $X_h, Y_h, Z_h, m, c = 0$ )
2: enquanto o algoritmo receber dados faça
3:   Criar  $\leftarrow 1$ 
4:   se  $h = 1$  então
5:      $c \leftarrow 1$ 
6:      $\mathcal{G}_c \leftarrow$  Criar( $X_h, c$ ) Executando o Algoritmo 4
7:   senão
8:     se  $h \geq 2$  então
9:       para todo grânulo  $\mathcal{G}_i$  faça
10:         $\mathcal{P}_i^* \leftarrow$  Calcular( $X_h, i$ ) Executando o Algoritmo 5
11:        Calcular  $\xi_{h,i}(X_h), \tau_{h,i}(X_h)$  e  $w_{h,i}(X_h)$  pelas equações (2.23), (2.22) e (2.37)
12:        Calcular as predições granules  $\hat{Z}_{ih}$  por (4.49)
13:        Calcular a predição singular  $\hat{Z}_h$  por (4.53)
14:        se a Inequação (2.28) for atendida então
15:           $\mathcal{P}_i \leftarrow \mathcal{P}_i^*$ 
16:          Criar  $\leftarrow 0$ 
17:        fim se
18:      fim para
19:      se Criar = 1 então
20:         $c \leftarrow c + 1$ 
21:         $\mathcal{G}_c \leftarrow$  CriarGrânulo( $X_h, Y_h, c$ ) Executando o Algoritmo 4
22:      fim se
23:      para todo para de grânulos  $\mathcal{G}_i$  e  $\mathcal{G}_j$  faça
24:        VerificarFusão( $\mathcal{G}_i, \mathcal{G}_j$ ) Executando o Algoritmo 6
25:      fim para
26:    fim se
27:  fim se
28:   $Z_h$  se torna disponível
29:  para todo grânulo  $\mathcal{G}_i$  faça
30:    Atualizar os pesos e coeficientes  $\mathcal{C}_i$  do modelo  $g_i$  por  $w_i$ 
31:  fim para
32: fim enquanto

```

---

### 4.5.1 Preditor AutoCloud de Consequente Linear

Similar aos preditores TEDA apresentados anteriormente, o P.AutoCloud de consequente linear também propõe uma regra cujo consequente granular  $Z_i$  é calculado em função da entrada  $Y_h^\top$  e dos coeficientes  $\Theta_i$ . Contudo, propõe que  $\Theta_i$  seja calculado pelo RLSVDF. Portanto, (2.37) assume a forma de (4.54).

$$R_i(X_h): \text{ Se } Z_h \text{ é } \mathcal{G}_i, \text{ então } Y_i = X_h^\top \Theta_i \quad (4.54)$$

Por ter sido inicialmente concebido para granulação de dados, o AutoCloud não contemplou a matriz  $\Theta_i$  para regressão linear. Portanto, o mecanismo de fusão de grânulos, apresentado na Seção 2.5.2.4, não apresenta o cálculo de uma matriz resultante  $\Theta_r$  de dois grânulos  $\mathcal{G}_i$  e  $\mathcal{G}_j$ .

Assim, é proposto que o cálculo da matriz resultante  $\Theta_r$  seja realizado por uma média ponderada pelas tipicidades  $\tau_{ih}$  e  $\tau_{jh}$ , conforme (4.55).

$$\Theta_{rh} = \frac{\tau_{ih}\Theta_{ih} + \tau_{jh}\Theta_{jh}}{\tau_{ih} + \tau_{jh}} \quad (4.55)$$

Por fim, durante a atualização dos coeficiente do modelo pelo RLSVDF, os níveis de ativação são utilizados para ponderar o ajuste de  $\Theta$  conforme proposto em (Cordovil Jr et al., 2022).

### 4.5.2 Preditor AutoCloud de Consequente Não-Linear

Embora os modelos evolutivos tradicionais apresentem vantagens significativas na adaptação através dados com incertezas e variações dinâmicas, a suposição de linearidade no consequente das regras pode limitar sua capacidade de capturar comportamentos complexos em sistemas reais. Em cenários de prognóstico de falhas, onde os padrões de degradação são fortemente não lineares e dependem de múltiplas variáveis sob condições operacionais não estacionárias, torna-se necessário o uso de consequentes mais expressivos.

Assim, esta seção introduz versões do P.AutoCloud com Consequente Não-Linear, uma abordagem que incorpora RBF e outra que incorpora RNNs no consequente do modelo, ampliando sua capacidade de modelar relações intrinsecamente não lineares de forma interpretável e adaptativa.

#### 4.5.2.1 Preditor AutoCloud RBF

A introdução de não linearidade ao modelo pode ser realizadas através da adição de uma RBF para parametrizar as entradas assimiladas pelo modelo. Dessa forma, é proposta uma versão adaptada do P.AutoCloud, o *Evolving Granular Neural Network with Radial Basis Function* (EGNN-RBF).

O proposto EGNN-RBF se diferencia do P.AutoCloud ao adicionar uma RBF para introduzir componentes de não linearidade ao modelo. Desse ponto de vista, tal mudança implica nas seguintes adaptações à regra (4.54):

$$R_i(X_h): \text{ Se } Z_h \text{ é } \mathcal{G}_i, \text{ então } Y_i = \psi(X_h)^\top \Theta_i \quad (4.56)$$

$i \in \mathbb{N}_{\leq c}$  onde  $X_h = [x_{h-l} \dots, x_h]^\top$  é uma janela de dados contendo a amostra atual do IS e as  $l$  amostras passadas mais recentes,  $\Theta \in \mathbb{R}^{l+1}$  é o vetor de parâmetros a ser estimado recursivamente pelo algoritmo RLSVDF ponderado (Cordovil Jr et al., 2022) (igualmente ao P.AutoCloud), e  $\psi(X_h) = [1, \phi_1 \dots, \phi_c]$  é um vetor de parâmetros onde cada  $\phi$  é calculado por meio de uma RBF cossenoidal da diferença entre  $X_h$  e centroide  $\mu_j$  previamente estabelecido.

$$\phi(\|X_h - \mu_j\|) = \cos(\|X_h - \mu_j\|) \quad (4.57)$$

O fluxograma na Figura 4.5 representa o caminho dos dados através do modelo, desde a entrada, a parametrização por RBF, até a predição da saída. Observe que o modelo neuro-fuzzy evolutivo proposto permite a criação de novos grânulos  $\mathcal{G}_i$  quando dados atípicos são apresentados.

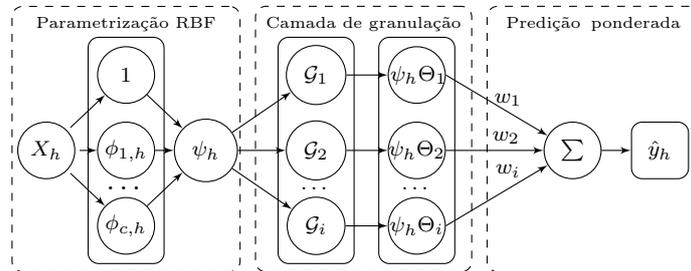


Figura 4.5: Fluxo de dados através do preditor AutoCloud de consequente RBF.

#### 4.5.2.2 Preditor AutoCloud RNN

Alternativamente às RBFs, a não linearidade pode ser introduzida ao modelo através de uma arquitetura neural. Nesse sentido, é proposta uma versão do P.AutoCloud, com consequente calculado através de uma RNN, o *Evolving Granular Recurrent Neural Network* (EGRNN).

O proposto EGRNN se diferencia do P.AutoCloud ao introduzir uma RLRNN no consequente da regra (4.49), alterando-a para a seguinte forma.

$$\text{Regra } i : \text{ Se } Z_h \text{ é } \mathcal{G}_i, \text{ então } \hat{Y}_{ih} = \text{RNN}_{ih}(X_h) \quad (4.58)$$

Similar ao método para cálculo de coeficiente resultantes em casos de fusão entre grânulos proposto em (4.55), no EGRNN os valores das matrizes de pesos resultantes de  $\text{RNN}_{rh} — U_{rh}, V_{rh}$  e  $W_{rh}$  — são calculados pela média ponderada pela tipicidade de cada grânulo.

$$W_{rh} = \frac{\tau_{ih}W_{ih} + \tau_{jh}W_{jh}}{\tau_{ih} + \tau_{jh}}, V_{rh} = \frac{\tau_{ih}V_{ih} + \tau_{jh}V_{jh}}{\tau_{ih} + \tau_{jh}}, U_{rh} = \frac{\tau_{ih}U_{ih} + \tau_{jh}U_{jh}}{\tau_{ih} + \tau_{jh}} \quad (4.59)$$

A Figura 4.6 ilustra a arquitetura do EGRNN, onde uma determinada amostra de dados  $X_h$  passa pela camada de granulação, acionando as políticas de manipulação de grânulos, atravessa a camada de predição granular, na qual o EGRNN de cada GIs estima uma saída granular  $\hat{Y}_{ih}$  e as agrega por meio de uma soma ponderada, denotada pelo operador  $\Sigma$ , fornecendo, por fim, a saída singular  $\hat{Y}_h$ .

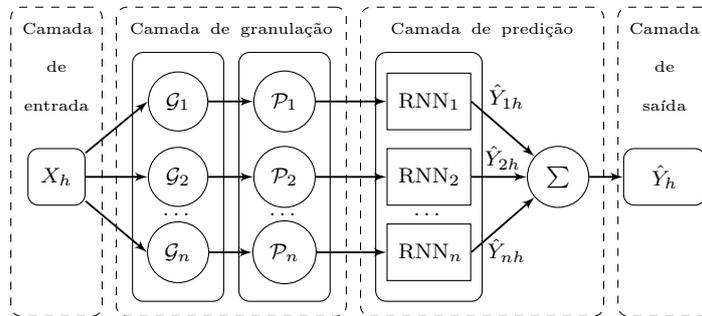


Figura 4.6: Evolving recurrent neural network framework.

## 4.6 Quantificação de Incerteza

Computação Granular é uma estrutura computacional na qual os dados são organizados em GIs, que representam agrupamentos de elementos com características ou padrões semelhantes. O conceito de granularidade, nesse contexto, refere-se ao nível de detalhe ou abstração desses GIs, podendo variar de altamente específicos (baixa granularidade) até mais generalizados (alta granularidade).

Portanto, a granularidade impacta diretamente na interpretabilidade dos GIs, conduzindo à fusão de diferentes informações em um único GI ou à sua divisão em múltiplos GIs. Para lidar com essa questão, (Pedrycz and Homenda, 2013) propõe o PJG como um método para projetar GIs adequados.

O PJG avalia a justificabilidade de um GI com base em dois parâmetros antagônicos: cobertura (a cardinalidade dos elementos  $x_h$  que se enquadram em seus limites) e especificidade (seu tamanho). A especificidade é considerada uma função decrescente do tamanho do GI,  $\overline{sp}_i = \text{tamanho}(\mathcal{G}_i)$ . A cobertura é considerada uma função crescente da cardinalidade dos elementos do GI,  $\overline{cov}_i = \text{cardinalidade}(\mathcal{G}_i)$ . Assim, a granularidade  $\rho$  deve proporcionar GIs simultaneamente tão compactos quanto numerosos.

A otimização de  $\overline{cov}$  e  $\overline{sp}$  pode ser tratada como um problema de otimização escalar, maximizando-se o produto entre cobertura e especificidade, considerando-o como uma função de  $\rho$  (Al-Hmouz et al., 2024), levando ao seguinte problema:

$$V(\rho) = \overline{cov} \cdot \overline{sp} \text{ e } \rho_{\text{opt}} = \operatorname{argmax}_{\rho} V(\rho) \quad (4.60)$$

O conceito de granularidade pode ser utilizado em modelos evolutivos para obter predições intervalares. Elevando seus parâmetros ou suas saídas a uma contraparte granular que admite um  $\rho$ , é possível predizer limites máximos e mínimos ao redor de uma predição singela (Al-Hmouz et al., 2024).

### 4.6.1 Incerteza por Parâmetros Granulares

Aplicando a ideia de parâmetros granulares ao modelo granular proposto, os parâmetros  $\Theta$  são transformados em um intervalo granular  $[\Theta_{\min}, \Theta_{\max}]$ , conduzindo a predição do modelo a um intervalo granular  $[\hat{Y}_{\min}, \hat{Y}_{\max}]$  também. Assumindo a

predição como uma função da amostra de dado atual  $X$  e de pesos ou coeficientes  $\mathcal{Q}$ , temos  $\hat{Y} = g(X, \mathcal{Q})$ , então temos:

$$\mathcal{Q}_{\min} = \min(\mathcal{Q} \cdot (1 - \rho), \mathcal{Q} \cdot (1 + \rho)) \quad (4.61)$$

$$\mathcal{Q}_{\max} = \max(\mathcal{Q} \cdot (1 - \rho), \mathcal{Q} \cdot (1 + \rho)) \quad (4.62)$$

$$\hat{Y}_{\min} = \min(g(X, \mathcal{Q}_{\min}), g(X, \mathcal{Q}_{\max})) \quad (4.63)$$

$$\hat{Y}_{\max} = \max(g(X, \mathcal{Q}_{\min}), g(X, \mathcal{Q}_{\max})) \quad (4.64)$$

## 4.6.2 Incerteza por Saídas Granulares

A abordagem proposta por (Al-Hmouz et al., 2024) torna a saída granular ao construir um intervalo em torno do resultado produzido pelo modelo numérico. O procedimento aproxima o erro de predição  $e_h$  como uma função  $f_e$  — polinomial, segundo (Al-Hmouz et al., 2024) — da saída do modelo  $\hat{Y}_h$  e estima os erros de predição como  $\hat{e}_h = f_e(\hat{Y}_h)$ . Em seguida, o produto  $\rho \cdot \hat{e}_h$  é utilizado como margem de erro da predição, o que conduz  $\hat{Y}_h$  ao intervalo  $[\hat{Y}_{\min}, \hat{Y}_{\max}]_h$  conforme segue.

$$[\hat{Y}_{\min}, \hat{Y}_{\max}]_h = [\hat{Y}_h - \rho \cdot \hat{e}_h, \hat{Y}_h + \rho \cdot \hat{e}_h] \quad (4.65)$$

Em vez de uma função polinomial, este trabalho propõe uma função logarítmica para aproximar o erro de predição, adaptando a metodologia descrita acima. Essa função pode ser modelada por meio do RLS.

## 4.7 Prognóstico e Predição de RUL

O principal objetivo do prognóstico é estimar a RUL de equipamentos baseado na série temporal IS. Assumindo que o IS é suficientemente monotônico e decrescente ( $x_h$  reduz à medida que o Fim de Vida (FV) se aproxima), existe um limiar relacionado ao FV do equipamento tal que  $x_h \leq \text{FV}$ . Considerando um equipamento que atinge seu fim de vida num instante  $H$ , a RUL real é definida por (4.66).

$$\text{RUL}_h = H - h \quad (4.66)$$

Por outro lado, num instante de referência  $h$ , a estimativa de RUL é feita através da predição consecutiva de ISs até que se atinja o valor do FV num instante  $H$ , conforme (4.67). Dessa forma, o P.AutoCloud é utilizado para calcular RUL.

$$\hat{RUL}_h = \inf \{H \in \mathbb{N} : \hat{x}_{h+H} \leq \text{FV}\}. \quad (4.67)$$

Como as bases de dados contém amostras de dados do início ao fim da vida útil de cada rolamento, para cada instante  $h$  existe uma vida útil restante real  $RUL_h$  associada e comparável à vida útil estimada  $\hat{RUL}_h$ . O pseudocódigo exposto no Algoritmo 13 demonstra o procedimento para cálculo de RUL em função de qualquer versão do P.AutoCloud.

---

**Algoritmo 13** Módulo de Predição de RUL

---

```

1: Inicializar:  $R\hat{U}L = 0, FV$ 
2: Entrada:  $Y_h, \hat{Z}_h$ 
3:  $\hat{Y}_h = Y_h$ 
4:  $z = \hat{Z}_h$ 
5: enquanto  $z \leq FV$  faça
6:    $\hat{Y}_h.inserir(z)$ 
7:    $\hat{Y}_h.deletar(\hat{Y}_h[0])$ 
8:    $z = \sum_{i=1}^c w_{ih} \cdot P.AutoCloud(\hat{Y}_h)$ 
9:    $R\hat{U}L = R\hat{U}L + 1$ 
10: fim enquanto
11: Saída: RUL

```

---

### 4.7.1 Quantificação de Incerteza em Predição de RUL

Também será calculada a incerteza relativa às predições de RUL para estimar o Intervalo de Confiança (IC) capaz de indicar onde a verdadeira RUL do equipamento se encontra através de consecutivas predições intervalares, conforme (4.63), (4.64) e (4.65).

Considerando um modelo de predição denotado por  $\mathcal{M}$ , o procedimento de quantificação de incerteza denotado por  $\mathcal{U}$  e um procedimento de seleção de intervalo  $[\underline{Y}_h, \bar{Y}_h]$  denotado por  $\mathcal{S}$ , a Figura 4.7 representa a quantificação da incerteza da RUL por meio da propagação de erro do IS.

O pseudocódigo exposto pelo Algoritmo 14 demonstra consecutivamente todas as etapas executadas para a realização da predição de RUL, em função de qualquer versão do P.AutoCloud, com quantificação de incerteza calculadas pelas abordagens expostas em (4.63) e (4.64) ou (4.7).

---

**Algoritmo 14** P.AutoCloud para Predição de RUL

---

```
1: Inicializar:  $c = h = 1$ ,  $RUL = RUL_{\min} = RUL_{\max} = 0$ ,  $m, \rho$ , EoL
2: Estrada:  $X_h, Y_h$ 
3: Criar grânulo  $\mathcal{G}_c, c = c + 1$ 
4: se  $h \geq 2$  então
5:    $Create = 1$ 
6:   para todos os grânulos  $\mathcal{G}_i$  faça
7:     Calcular  $\mathcal{P}_i$  e  $\hat{Z}_{ih}$ 
8:     se (2.28) é atendida então
9:        $Create = 0$ 
10:    fim se
11:  fim para
12:  se  $Create = 1$  então
13:    Criar grânulo  $\mathcal{G}_{c+1}, c = c + 1$ 
14:  fim se
15:  para todo par de grânulos  $\mathcal{G}_i$  e  $\mathcal{G}_j$  faça
16:    se (2.30) ou (2.31) forem atendidas então
17:      Fundir  $\mathcal{G}_i$  e  $\mathcal{G}_j$ 
18:    fim se
19:  fim para
20:  Calcular  $\hat{Z}_h$  usando (4.53)
21:  Calcular  $[\underline{Z}_h, \bar{Z}_h]$  usando (4.63) e (4.64) ou (4.7)
22:   $Y_h$  se torna disponível
23:  Atualizar  $\Theta_i$  ou  $W_i, V_i, U_i$  do P.AutoCloud
24:   $z = \underline{Z}_h, z = \hat{Z}_h, \bar{z} = \bar{Z}_h = \hat{Z}_h$ 
25:   $\bar{Y}_h, \hat{Y}_h, \underline{Y}_h = Y_h$ 
26:  enquanto  $z \geq \text{EoL}$  faça
27:     $\hat{Y}_h.\text{insert}(z)$ 
28:     $\hat{Y}_h.\text{delete}(\hat{Y}_h[0])$ 
29:     $z = \sum_{i=1}^c w_{ih} \cdot \text{P.AutoCloud}_i(\hat{Y}_h)$ 
30:     $RUL = RUL + 1$ 
31:  fim enquanto
32:  enquanto  $\bar{z} \geq \text{EoL}$  e  $\bar{z} \geq \text{EoL}$  faça
33:     $\bar{Y}_h.\text{insert}(z_{\max})$ 
34:     $\bar{Y}_h.\text{delete}(\bar{Y}_h[0])$ 
35:     $\underline{Y}_h.\text{insert}(z_{\min})$ 
36:     $\underline{Y}_h.\text{delete}(\underline{Y}_h[0])$ 
37:     $y_{\max} = \sum_{i=1}^c w_{ih} \cdot \text{P.AutoCloud}_i(\bar{Y}_h)$ 
38:     $y_{\min} = \sum_{i=1}^c w_{ih} \cdot \text{P.AutoCloud}_i(\underline{Y}_h)$ 
39:    se  $y_{\min} \geq \text{FV}$  então
40:       $RUL_{\min} + = 1$ 
41:    fim se
42:    se  $y_{\max} \geq \text{FV}$  então
43:       $RUL_{\max} + = 1$ 
44:    fim se
45:  fim enquanto
46: fim se
47: Saída:  $RUL_{\min}, RUL, RUL_{\max}$ 
```

---

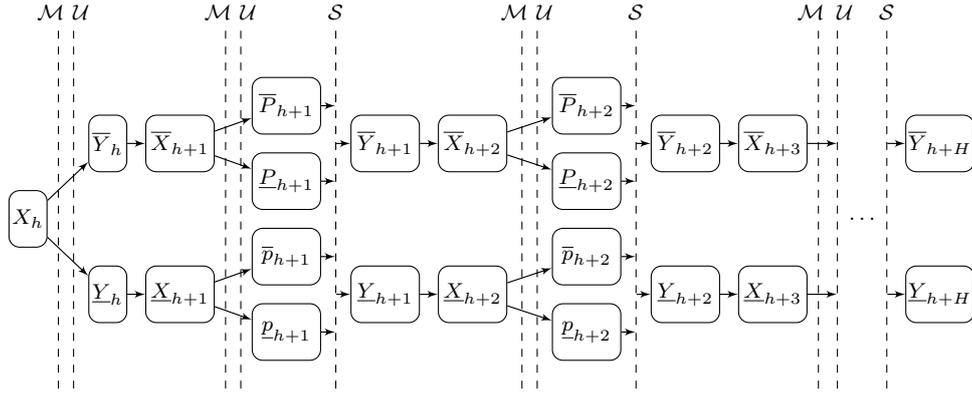


Figura 4.7: Incerteza da previsão de RUL pela propagação da estimativa de erro.

## 4.8 Quantificação dos Erros do Modelo

A avaliação quantitativa da performance da previsão de RUL será feita através dos indicadores RMSE e *Mean Absolute Percentage Error* (MAPE), definido por (Ni et al., 2022) pelas equações (4.68) e (4.69).

$$\text{RMSE} = \frac{\sqrt{\sum_{h=1}^H (\hat{\text{RUL}}_h - \text{RUL}_h)^2}}{H} \quad (4.68)$$

$$\text{MAPE} = \frac{\sum_{h=1}^H |\hat{\text{RUL}}_h - \text{RUL}_h|}{H} \quad (4.69)$$

# Capítulo 5

## Resultados

Esta seção discorre acerca dos resultados de predição de RUL em rolamentos utilizando o P.AutoCloud proposto na seção 4.5 e apresentado pelo Algoritmo 14.

O Preditor AutoCloud é um modelo evolutivo construído sobre a estrutura do TEDA, empregando consequentes de arquitetura linear com parâmetros estimados pelo RLSVDF, podendo introduzir não linearidade através de RBFs, e consequente de arquitetura não linear através de RNNs para gerar predições. O modelo é projetado especificamente para granular os dados de forma significativa através do PJG, garantindo interpretabilidade e explicabilidade, ao mesmo tempo que aproveita a capacidade do RLSVDF e das RNNs de reter conhecimento de entradas passadas e lidar de forma eficaz com a não linearidade inerente às NNs.

Esta seção do trabalho tem como objetivo avaliar o desempenho do Preditor AutoCloud em prognóstico de falhas em rolamentos por meio de simulações utilizando dados de vida útil acelerada provenientes de conjuntos de dados de vibração. Os objetivos principais são: avaliar a interpretabilidade do modelo por meio da análise dos GIs gerados, verificar e comparar o desempenho das abordagens RLSVDF, RBF e RNN para predições dentro da estrutura do EM, e examinar as abordagens de quantificação de incerteza propostas. Objetivos adicionais incluem a avaliação do processo de aprendizado contínuo (especificamente, a modificação de parâmetros e estrutura) por meio do monitoramento da redução do erro e quantidade de grânulos a cada iteração.

Serão apresentadas, consecutivamente, as bases de dados utilizadas, o resultado do pré-processamento proposto, a seleção e extração de características, a criação

do IS bem como IED, os resultados da predição de RUL e respectivos desempenhos, e finalmente as comparações entre os modelos propostos.

## 5.1 Base de Dados

Conforme apresentado na Seção 3.2, a base de dados XJTU proposta por (Wang et al., 2018) é predominantemente utilizada em testes durante o desenvolvimento de algoritmos para cálculos de prognóstico de falhas em rolamentos, sendo a base de dados escolhida para os testes do modelo proposto neste trabalho.

A base XJTU-SY *Bearing Dataset* contém registros completos do ciclo de vida de 15 rolamentos de elementos de mesmo modelo submetidos a testes de degradação acelerada. Os experimentos foram realizados pelo Instituto de *Design Science and Basic Component da Xi'an Jiaotong University (XJTU)* em parceria com a *Changxing Sumyoung Technology Co., Ltd. (SY)*, conforme (Wang et al., 2018).

Para cada condição operacional, a base de dados expõe a identificação de cada rolamento, e respectivas quantidades de arquivos armazenados durante o teste, o tempo de vida útil e o componente em falha (Pista interna, Pista externa, Gaiola ou Esfera).

Condições operacionais	Identificação do rolamento	Número de arquivos	Vida útil do rolamento	Elemento em falha
Condição 1 (35Hz/12kN)	<b>Rolamento1_1</b>	<b>123 amostras</b>	<b>02 h 03 min</b>	<b>P.externa</b>
	Rolamento1_2	161 amostras	02 h 41 min	P.externa
	Rolamento1_3	158 amostras	02 h 38 min	P.externa
	Rolamento1_4	122 amostras	02 h 02 min	Gaiola
	<b>Rolamento1_5</b>	<b>52 amostras</b>	<b>00h 52 min</b>	<b>P.Externa</b>
Condição 2 (37.5Hz/11kN)	Rolamento2_1	491 amostras	08 h 11 min	P.interna
	Rolamento2_2	161 amostras	02 h 41 min	P.externa
	Rolamento2_3	533 amostras	08 h 53 min	Gaiola
	<b>Rolamento2_4</b>	<b>42 amostras</b>	<b>00 h 42 min</b>	<b>P.externa</b>
	Rolamento2_5	339 amostras	05 h 39 min	P.externa
Condição 3 (40Hz/10kN)	Rolamento3_1	2538 amostras	42 h 18 min	P.externa
	Rolamento3_2	2496 amostras	41 h 36 min	Todos
	Rolamento3_3	371 amostras	06 h 11 min	P.interna
	Rolamento3_4	1515 amostras	25 h 15 min	P.interna
	Rolamento3_5	114 amostras	01 h 54 min	P.externa

Tabela 5.1: Base de dados de rolamentos XJTU.

Neste trabalho, optou-se por focar rolamentos com o mesmo tipo de falha (no caso, falhas de pista externa). A Tabela 5.1 lista os rolamentos registrados na base de dados, destacando em negrito os rolamentos utilizado neste trabalho. Para fins de abreviação, os termos R1\_1, R1\_5 e R2\_4 referem-se respectivamente aos Rolamento1\_1, Rolamento1\_5, e Rolamento2\_4 destacados na Tabela 5.1.

Os rolamentos tiveram seus sinais de vibração consecutivamente coletados a cada minuto, utilizando acelerômetros montados ortogonalmente, registrando amostras de dados com 1,28 segundos de duração à frequência de amostragem de 25,6 kHz em arquivos CSV, permitindo a análise de sua degradação progressiva, conforme a Figura 5.1.

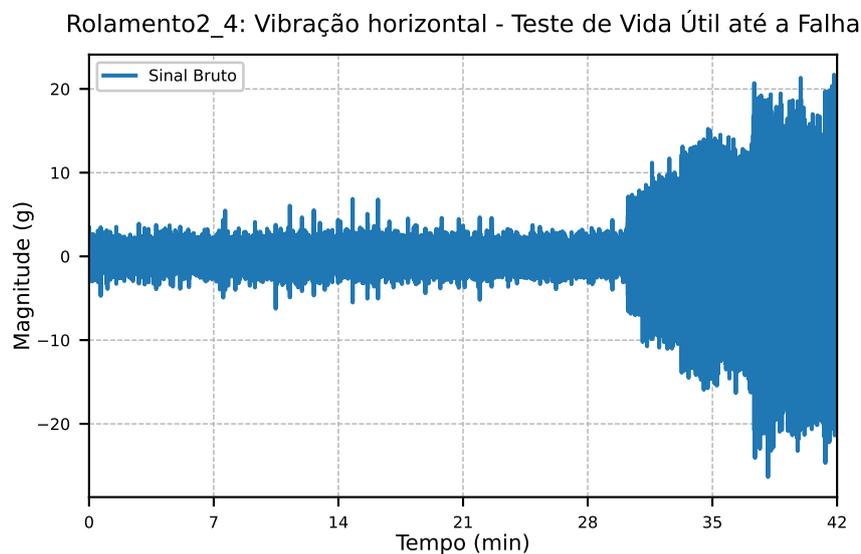


Figura 5.1: Rolamento 2\_4 - Teste do início ao fim da vida útil.

Os testes foram conduzidos até que a amplitude do sinal de vibração excedesse 10 vezes o valor máximo registrado na fase normal de operação, indicando falha iminente. No entanto, os dados coletados representam a vibração resultante de todos os componentes do sistema em funcionamento, ruídos e variações de medição, e não apenas do rolamento em análise.

Dessa forma, a utilização dos sinais de vibração em sua forma bruta pode comprometer a precisão das análises, dificultando a avaliação correta da degradação do rolamento e a eficácia dos métodos de prognóstico de falhas. A seguir será demonstrado o resultado do pré-processamento utilizando o VMD e a FBE propostos anteriormente nas Seções 2.2.1 e 4.2.1, respectivamente.

## 5.2 Pré-processamento de Dados

A fim de mitigar as influências de vibração provenientes de componentes irrelevantes e ruídos de medição, os dados brutos de vibração foram processados com o Algoritmo 9, conforme detalhado na Seção 4.2.2. O resultado é um sinal com características predominantemente associadas ao rolamento, mitigando interferências externas.

As Figuras 5.2, 5.3 e 5.4 ilustram a comparação entre os sinais brutos (azul) e filtrados (laranja) para os rolamentos analisados em suas últimas amostras antes da falha. Notam-se padrões de vibração distintos entre os Rolamentos 1\_1, 1\_5 e 2\_4, cujas falhas ocorrem respectivamente na gaiola, na pista interna e externa, e na pista externa.

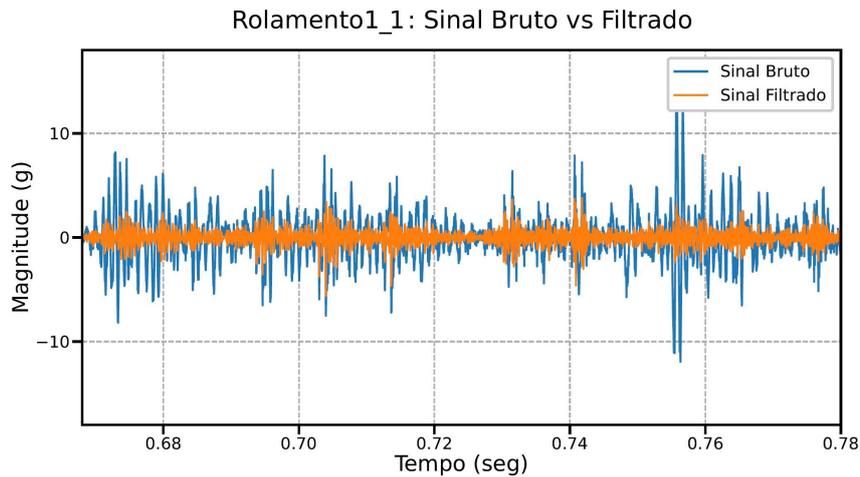


Figura 5.2: Rolamento 1\_1 - Sinal bruto e sinal filtrado.

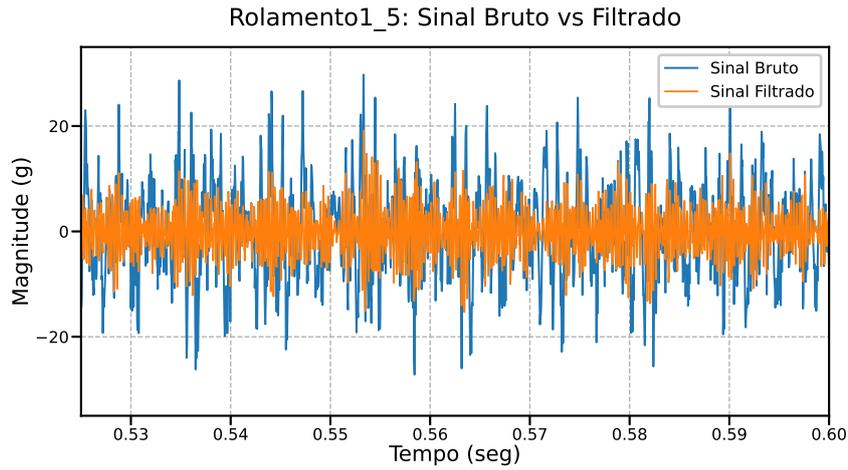


Figura 5.3: Rolamento 1\_5 - Sinal bruto e sinal filtrado.

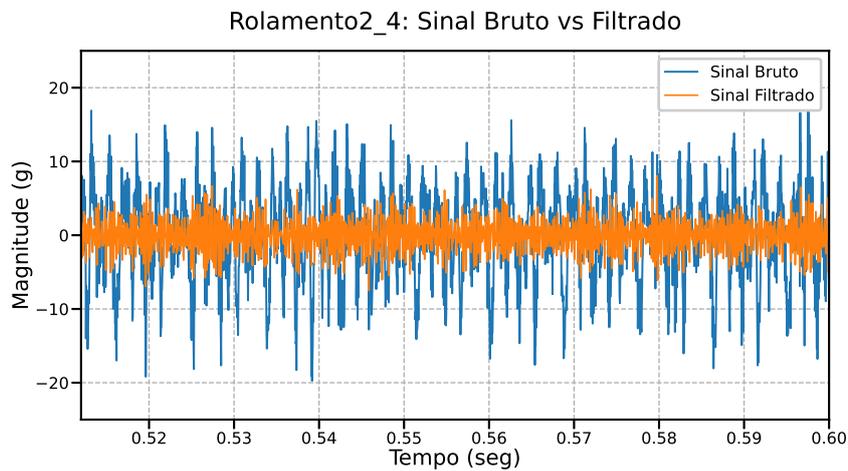


Figura 5.4: Rolamento 2\_4 - Sinal bruto e sinal filtrado.

Considerando que cada amostra contém 32.768 medições, realiza-se, em seguida, um processo de extração de características, reduzindo a dimensionalidade dos dados para uma única representação da amostra, facilitando a análise e interpretação das informações.

### 5.3 Extração e Seleção de Características

Baseado nos indicadores propostos pelas equações das Tabelas 4.1 e 4.2 da seção 4.3.1, fez-se a extração de características de cada amostra de dados dos Rolamentos 1\_1, 1\_5 e 2\_4.

O resultado do processo resumiu cada amostra de dados a um único valor representativo do nível de degradação do rolamento no respectivo instante de coleta. Contudo, algumas características não apresentaram comportamento crescente progressivo, esperado, fazendo necessário selecionar as melhores características extraídas através das métricas propostas 4.3.2. O conjunto de gráficos da Figura 5.5 mostra as características selecionadas já normalizadas.

Comparação de Características - Rolamentos 1\_1, 1\_5 e 2\_4

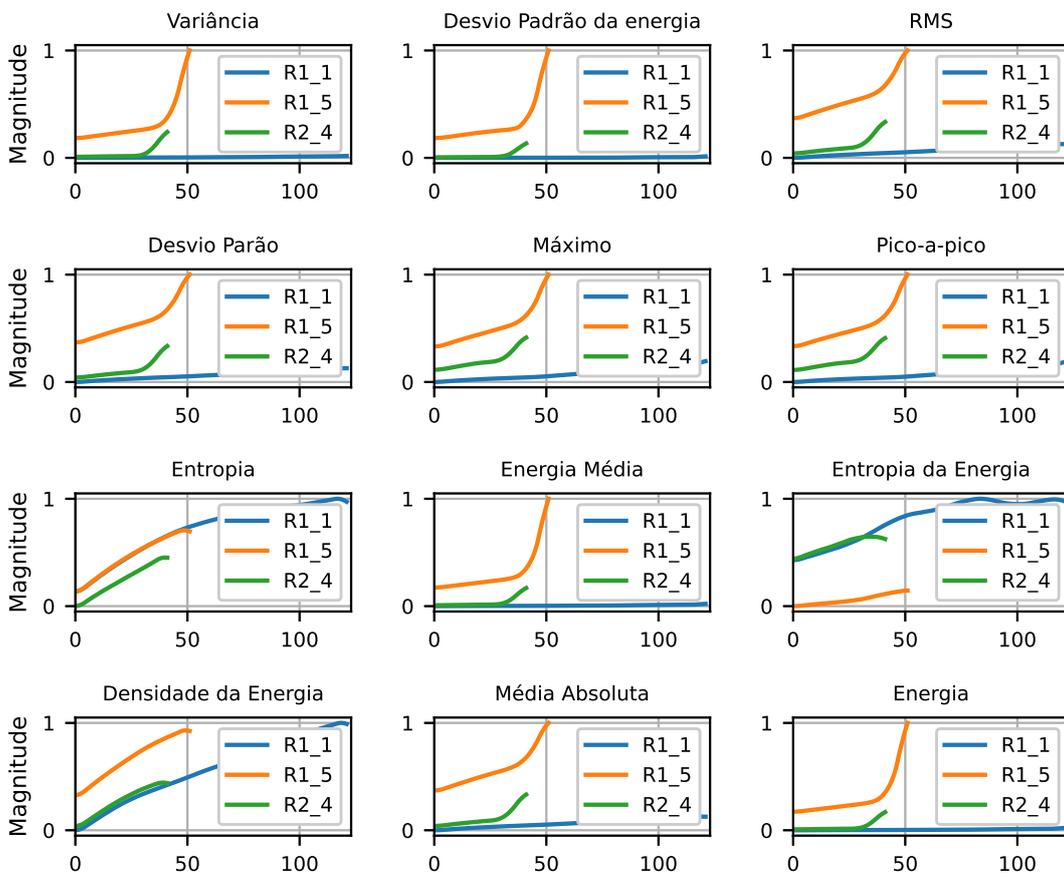


Figura 5.5: Características selecionadas dos Rolamentos 1\_1, 1\_5 e 2\_4.

Optou-se por descartar as características que não atingissem um valor de monotonicidade (M), trendabilidade (T) e correlação (C) simultaneamente superiores a 0,5. A Figura 5.6 mostra os valores obtido para o Rolamento 1\_5.

## Rolamento1\_5: Métricas de Avaliação das Características

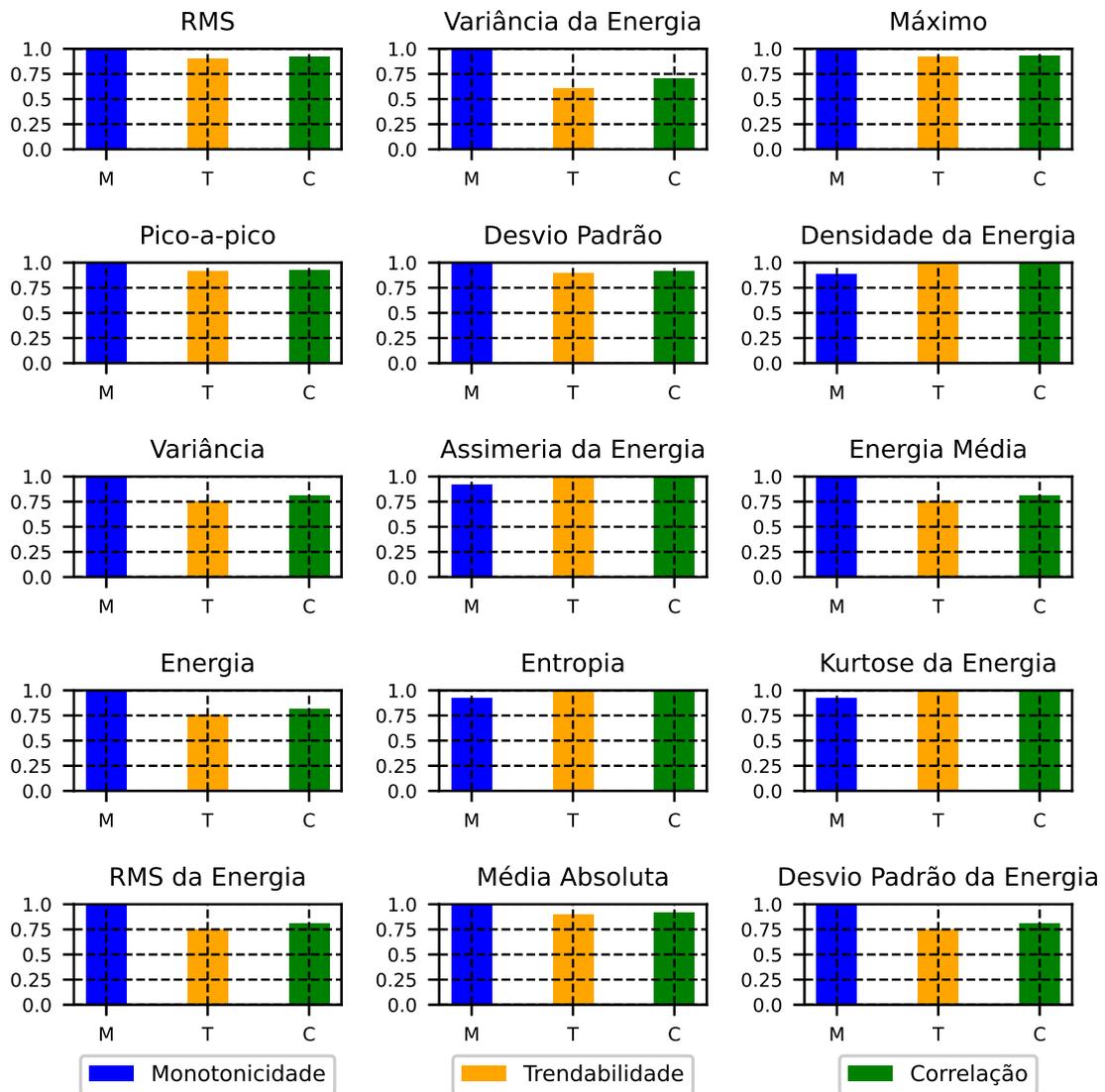


Figura 5.6: Métricas de avaliação de Características: Rolamento 1\_5.

## 5.4 Criação dos Indicadores de Saúde e Estágio de Degradação

Para medir o estágio e o nível de desgaste dos rolamentos e posteriormente melhorar o desempenho do preditor AutoCloud, calculou-se o IED, com base na

correlação dos sinais de vibração segundo (4.48), e o IS através das características listadas nas Tabelas 4.1 e 4.2, selecionadas pelos critérios de monotonicidade de *trendability*, e fundidas pelo PCA proposto no algoritmo 10. O resultado obtido são as séries temporais ilustradas pelos gráficos das Figuras 5.7 a) e b).

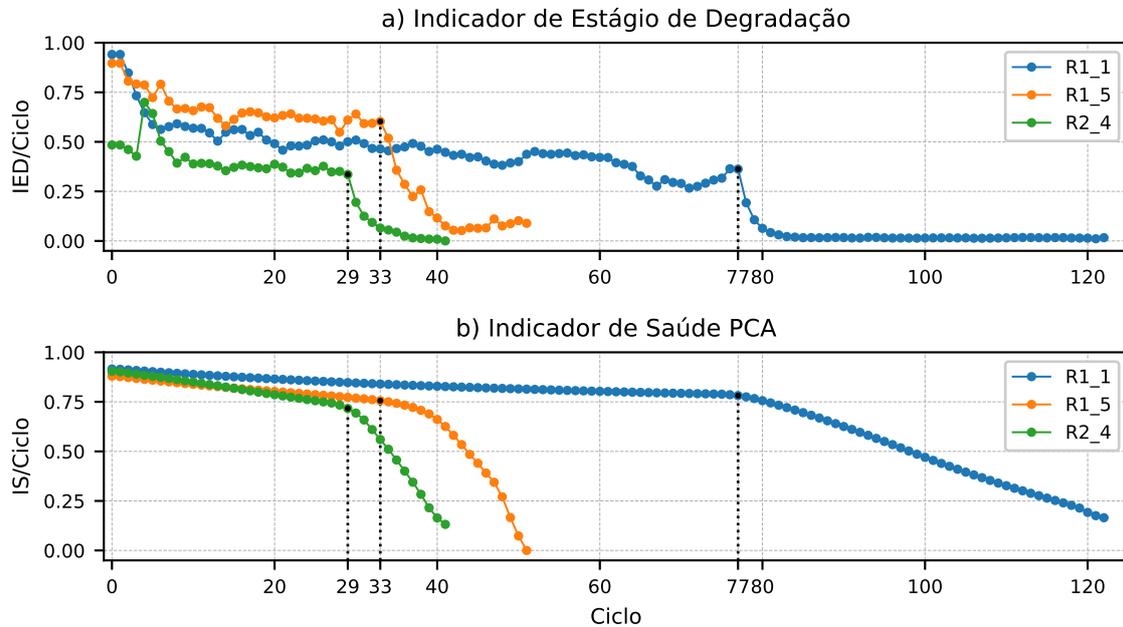


Figura 5.7: Indicadores de saúde e estágio de degradação dos Rolamentos 1\_1, 1\_5 e 2\_4.

A Figura 5.7 a) exibe as curvas de IED dos rolamentos ao longo do tempo, onde se observam platôs de ED diferentes entre os primeiros e últimos ciclos de vida útil. O formato das séries sugerem um comportamento similar na evolução do ED, onde os pontos pretos destacados pelas retas pontilhadas verticais destacam o instante de tempo a partir do qual ocorre a transição entre o primeiro e o segundo ED de cada rolamento. Notam-se também valores próximos de 0 para o IED nos últimos EDs.

A Figura 5.7 b) exibe as curvas de IS dos rolamentos ao longo do tempo. O formato das séries de IS também são similares, onde se observam um desgaste lento seguido por um veloz à medida que os ciclos de vida se aproximam do fim. Os pontos pretos destacados pelas retas pontilhadas verticais destacam o instante de tempo a partir do qual ocorre a transição entre as velocidades de degradação do IS de cada rolamento.

Considerando a presença do mesmo tipo de falha em todos os rolamentos, as

curvas de IED e IS de todos os rolamentos revelam comportamentos similares ao longo dos ciclos de vida. A análise simultânea dos gráficos revela que as mudanças de ED e velocidade de desgaste ocorrem no mesmo instante de tempo, evidenciando sua correlação.

## 5.5 Predição de RUL com o Preditor Autocloud

Após a criação do IED e IS, realizaram-se simulações dos cálculos de prognóstico dos Rolamentos 1\_1, 1\_5 e 2\_4 com o Algoritmo 14 proposto. Para uma comparação mais homogênea, foi estabelecido o mesmo FVs para todos os rolamentos em todas as simulações executadas, objetivando a comparação de desempenho diante de um cenário similar.

Conforme discorrido na Seção 4.7, o P.AutoCloud é usado, a partir de um instante de tempo  $h$ , para cálculo de predições sequenciais de IS até atingir um limiar pré-estabelecido num instante de tempo  $H$ . A vida útil restante predita  $R\hat{U}L_h$  é o intervalo de tempo entre o instante de referência  $h$  e o predito  $H$ . O processo é então repetido nos instantes de tempo consecutivos até que o nível de degradação real do equipamento atinja o limiar.

Durante as simulações observaram-se variações nos resultados obtidos em função dos valores previamente definidos para os hiperparâmetros do modelo, mesmo diante de suas características adaptativas e evolutivas. Destaca-se o parâmetro de sensibilidade para criação de grânulos  $m$  devido sua influência direta na quantidade de GIs, afetando diretamente a interpretabilidade e explicabilidade do modelo, e sua influência indireta nas predições, devido ao cálculo de  $\hat{Y}_h$  ser ponderado pelas tipicidades  $\tau_h$  de cada GI.

Os demais hiperparâmetros do RLSVDF, do RBF ou da RLRNN também afetam diretamente o resultado das predições  $\hat{Y}_h$ . Outro parâmetro, a granularidade  $\rho$ , também afeta o desempenho do modelo, uma vez que impacta diretamente na quantificação do intervalo de incerteza e na justificabilidade dos GIs. Assim, deve-se realizar um ajuste adequado de cada hiperparâmetro para garantir o bom desempenho do modelo.

### 5.5.1 Otimização de Hiperparâmetros

Antes da execução definitiva de cada versão do algoritmo para cada curva de degradação dos rolamentos, foi realizado um processo de otimização dos hiperparâmetros com o objetivo de minimizar os erros de predição do modelo e manter o número de GIs criados entre 2 e 8 (a fim de evitar a total generalização e a total especificação). Para isso, usou-se o *framework* de otimização de hiperparâmetros Optuna, devido sua abordagem *Bayesiana* prover boas combinações de valores como poucas iterações. A Tabela 5.2 estabelece parâmetros otimizados, os respectivos intervalos de busca de cada parâmetro e o passo utilizado.

Parâmetro	Símbolo	Máximo	Mínimo	Passo
Sensibilidade de granulação	$m$	1.2	1.9	0.01
Granularidade	$\rho$	0.01	1.00	0.001
Janela de dados	$l$	3	10	1
Nº de amostras passadas	$s$	2	10	1
Fator de esquecimento	$\alpha$	0.99	0.9999	Logarítmico
Fator de direção	$\beta$	$10^{-3}$	1	Logarítmico
Taxa de aprendizado	$\eta$	$10^{-5}$	$10^{-1}$	Logarítmico
Comprimento do vetor de atividade da RNN	$h_n$	3	20	1
Dependência temporal da RNN	t	1	25	1

Tabela 5.2: Otimização de hiperparâmetros: Espaços de busca.

O processo de otimização foi realizado em duas etapas. A primeira para definir os valores de hiperparâmetros capazes de entregar o menor erro de predição de RUL (pela minimização do MAPE). A segunda foi para definir o valor da granularidade  $\rho$  usada na quantificação de incerteza (pela maximização do produto  $\overline{cov} \cdot \overline{sp}$ ) obedecendo o PJG.

A seguir, são apresentadas as melhores combinações de hiperparâmetros para cada tipo de conseqüente proposto para o Preditor AutoCloud e respectivo desem-

penho para predição de RUL com quantificação de incerteza.

### 5.5.2 P.AutoCloud: RLSVDF

A Tabela 5.3 apresenta a melhor combinação de hiperparâmetros para o Preditor AutoCloud de consequente RLSVDF, onde  $l$  é o tamanho da janela de dados,  $s$  é o número de amostras passadas armazenadas,  $\alpha$  e  $\beta$  são respectivamente os fatores de esquecimento e direção do RLSVDF,  $m$  é o fator de sensibilidade para criação de grânulos e  $\rho$  é a granularidade. Vale destacar que os parâmetros pré-otimizados não se adaptam de forma recursiva, o que afeta o desempenho do modelo.

<b>Rolamento</b>	$l$	$s$	$\alpha$	$\beta$	$m$	$\rho$	<b>MAPE</b>
<b>R1_1</b>	10	2	0,991	0,001	1,86	0,009	2487,60
<b>R1_5</b>	10	2	0,996	0,057	1,76	0,009	301,11
<b>R2_4</b>	8	8	0,999	1,000	1,60	0,009	165,60

Tabela 5.3: Preditor AutoCloud: RLSVDF - Otimização de hiperparâmetros.

Embora apresentem o mesmo tipo de falha, salvo  $l$  e  $s$  para R1\_1 e R1\_5, não houve valores similares para os mesmos hiperparâmetros sintonizados em função da minimização do MAPE.

A Figura 5.8 ilustra o resultado da granulação dos ISs durante a execução do algoritmo com destaque para o instante em que o valor de IS atinge o FV. Pode-se observar a criação de até quatro grânulos distintos (G1, G2, G3 e G4), cobrindo amostras com taxas de variação semelhantes — mais altas próximas ao EOL — e sugerindo estágios distintos de degradação. Além disso, a semelhança entre as curvas de IS de R1\_1, R1\_5 e R2\_4 pode estar associada à presença do mesmo tipo de falha (pista externa) em todos os casos.

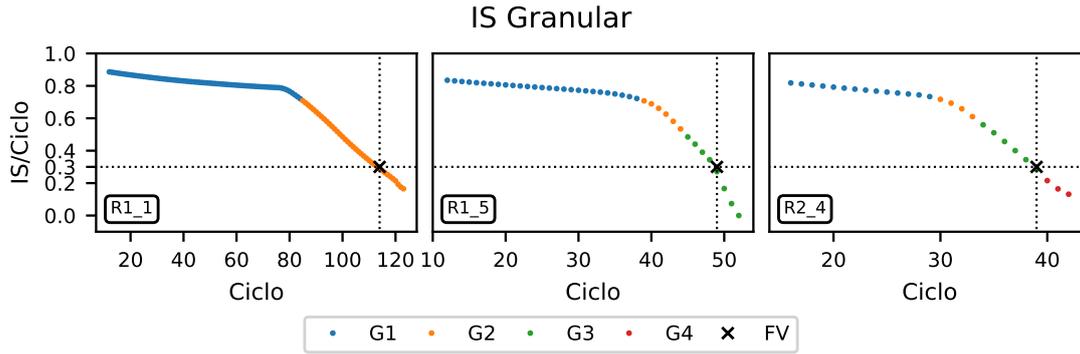


Figura 5.8: Granulação dos indicadores de saúde dos Rolamentos 1\_4, 1\_5 e 2\_4.

Os resultados de predição da RUL para cada rolamento pode ser analisada na Figura 5.9. É perceptível que, durante as predições iniciais, o algoritmo não é capaz de fornecer estimativas precisas de RUL. No entanto, por meio do aprendizado contínuo, o modelo ajusta os valores de  $\Theta$ , cria novos grânulos e, por fim, atinge predições mais próximas dos valores reais.

Para R1\_1, a zona de incerteza ficou exatamente sobreposta à predição de RUL no trecho de ciclos mostrados, pois os valores de granularidade  $\rho$  não foram suficientes para reduzir ou aumentar a quantidade de passos até que IS predito atingisse FV, durante a execução do algoritmo de predição de IS múltiplos passos a frente.

De outra forma, para R1\_5 e R2\_4, a zona de incerteza é evidente e apresenta comportamento de afunilamento à medida que os ciclos de vida dos rolamentos se aproximam de FV. Em R1\_5 a borda inferior da zona de incerteza excede ligeiramente a zona de tolerância de  $-30\%$  em relação a RUL real entre os ciclos 17 e 36, enquanto em R2\_4 o mesmo ocorre entre os ciclos 32 e 36. Contudo a borda superior da zona de incerteza para ambos rolamentos excedeu significativamente a zona de tolerância de  $+30\%$  até o ciclo 36 para R1\_5 e 32 para R2\_4.

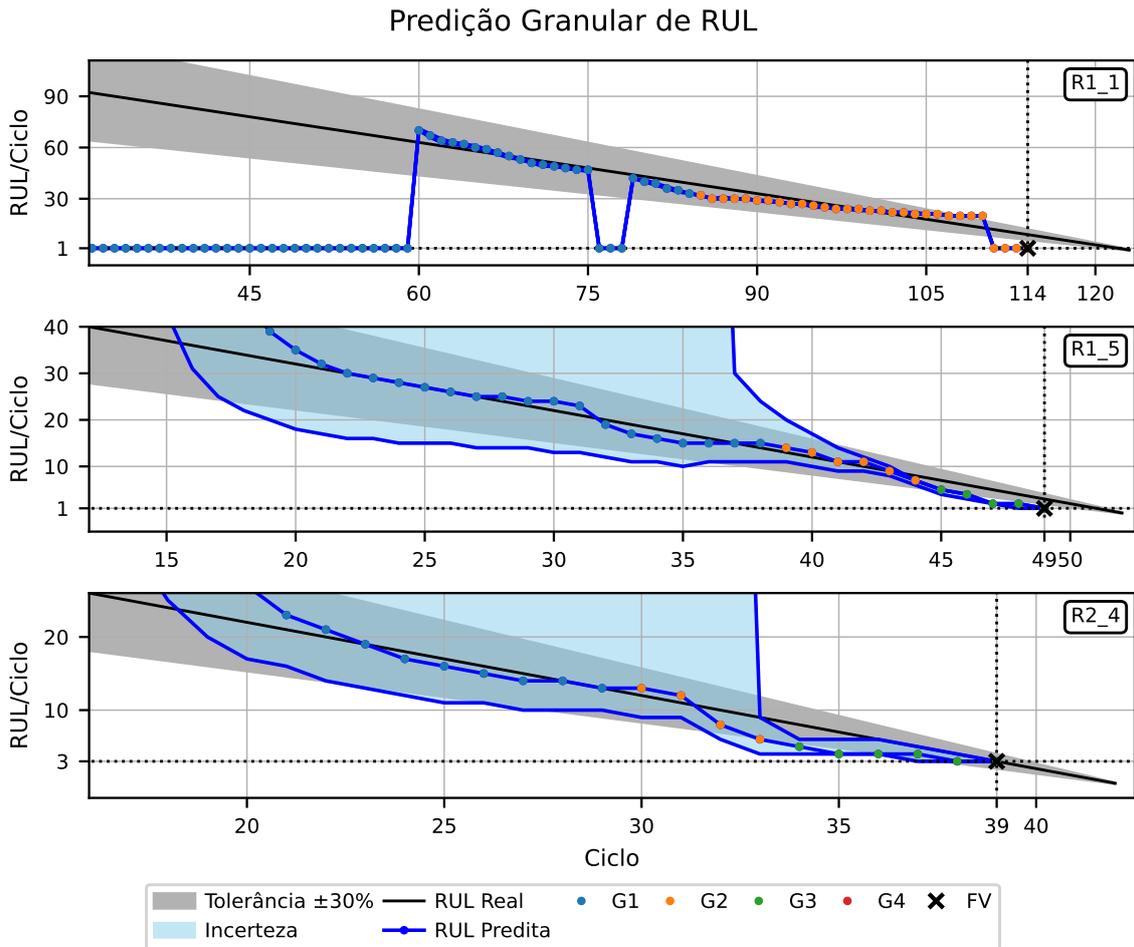


Figura 5.9: Predição de RUL granular dos Rolamentos 1\_4, 1\_5 e 2\_4.

Uma análise simultânea da granulação exposta nos gráficos das Figuras 5.8 e 5.9 revela a criação de novos GIs durante as transições do IS e a melhoria no desempenho do modelo com o passar dos ciclos. Também pode-se destacar o afunilamento da zona de incerteza no ciclos próximos à criação de GIs.

O aumento da quantidade de grânulos ao longo do tempo revela a variação estrutural do modelo enquanto a melhoria na acurácia das predições de RUL ocorre devido ao ajuste recursivo dos parâmetros  $\Theta$  dos RLSVDFs de cada grânulo. O aprendizado contínuo pode ser melhor visualizado nos gráficos da Figura 5.10, onde é exposto o RMSE de cada rolamento ao longo do tempo.

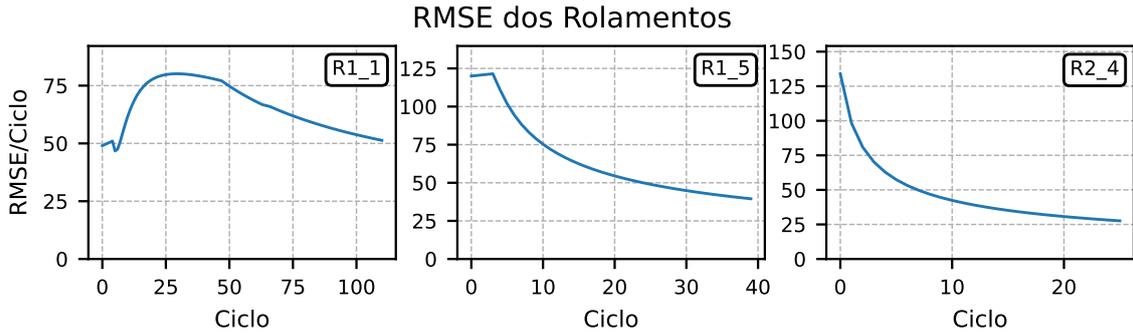


Figura 5.10: RMSE da previsão de RUL dos Rolamentos 1\_4, 1\_5 e 2\_4.

### 5.5.3 EGNN-RBF

A Tabela 5.4 apresenta a melhor combinação de hiperparâmetros para o EGNN-RBF de conseqüente RLSVDF com parametrização de entradas por RBF, onde  $l$  é o tamanho da janela de dados,  $s$  é o número de amostras passadas armazenadas,  $\alpha$  e  $\beta$  são respectivamente os fatores de esquecimento e direção do RLSVDF,  $m$  é o fator de sensibilidade para criação de grânulos e  $\rho$  é a granularidade.

Rolamento	$l$	$s$	$\alpha$	$\beta$	$m$	$\rho$	MAPE
R1_1	5	3	0,9999	1	1,62	0,01	3091,55
R1_5	5	7	0,9999	1	1,67	0,55	575,33
R2_4	8	8	0,9990	1	1,60	0,55	391,61

Tabela 5.4: EGNN-RBF - Otimização de hiperparâmetros.

Diferentemente do modelo apresentado anteriormente, este adiciona uma etapa de parametrização das entradas  $X_h$  por meio de uma RBF, conseqüentemente alterando os hiperparâmetros listados na tabela 5.3. Vale destacar que os parâmetros pré-otimizados não se adaptam de forma recursiva, o que afeta o desempenho do modelo. Especialmente o  $\mu$  do TEDA, utilizado para parametrizar as entradas não lineares do RBF cossenoidal, e do  $\rho$ , usado para quantificar o intervalo de incerteza.

Embora apresentem o mesmo tipo de falha, salvo  $l$  e  $\alpha$  para R1\_1 e R1\_5 e  $\beta$  para todos os rolamentos, não houve valores similares para os mesmos hiperparâmetros sintonizados em função da minimização do MAPE. Vale destacar que os parâmetros pré-otimizados não se adaptam de forma recursiva, o que afeta o de-

sempenho do modelo. Especialmente o  $\mu$  do TEDA, que é utilizado para calcular as entradas não lineares do tipo RBF cossenoidal  $\rho$ , usado para quantificar o intervalo de incerteza.

A Figura 5.11 ilustra o resultado da granulação dos ISs durante a execução do algoritmo com destaque para o instante em que o valor de IS atinge o FV. Pode-se observar semelhança entre as curvas dos ISs de R1\_1, R1\_5 e R2\_4, possivelmente pelo mesmo tipo de falha. Além disso, observa-se a criação de até quatro grânulos distintos (G1, G2, G3, G4 e G5).

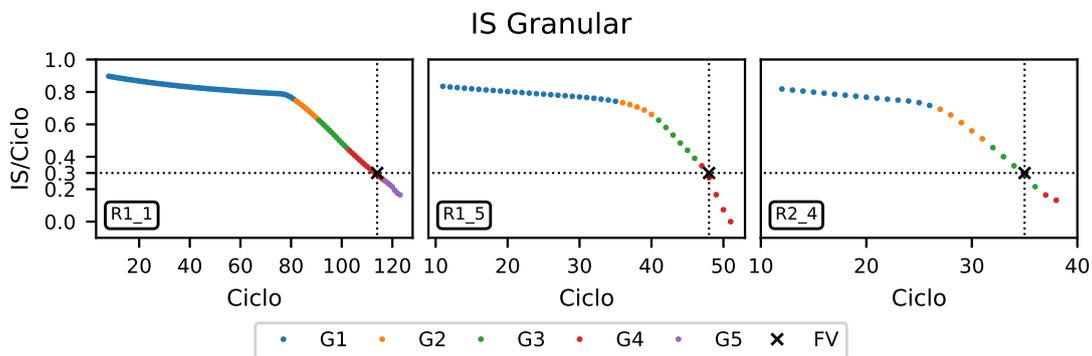


Figura 5.11: Granulação dos indicadores de saúde dos Rolamentos 1\_4, 1\_5 e 2\_4.

Diferentemente do modelo anterior, a granulação é realizada sobre  $\psi_h = \text{RBF}(X_h, \mu)$ , e não diretamente sobre  $X_h$ . Conseqüentemente, a granulação exibida nos gráficos da Figura 5.11 é um reflexo da granulação de  $\psi_h$ . Ademais, a não linearidade imposta pela RBF mitiga as evidências de relação entre os GIs e o comportamento dos ISs observadas anteriormente.

Os resultados de predição de RUL para cada rolamento pode ser analisada na Figura 5.12. É perceptível que, durante as predições iniciais, o algoritmo não é capaz de fornecer estimativas precisas de RUL. No entanto, por meio do aprendizado contínuo, o modelo ajusta os valores de  $\Theta$ , cria novos grânulos e, por fim, atinge predições mais próximas dos valores reais. E em todos os rolamentos, a zona de incerteza apresentou comportamento de afunilamento ao longo do tempo.

Para R1\_1, desde o ciclo 37 a borda inferior da zona de incerteza excedeu a zona de tolerância de  $-30\%$  em relação a RUL real, e a partir do ciclo 82 a borda superior da zona de incerteza também. Nos últimos 35 ciclos de vida (os mais críticos), a zona de incerteza ficou muito aquém da margem de tolerância de

$\pm 30\%$  em relação a RUL real e praticamente sobreposta à RUL predita, removendo a confiabilidade das predições.

Nos rolamentos R1\_5 e R2\_4, a zona de incerteza permanece dentro da tolerância de  $\pm 30\%$  em relação a RUL real até os últimos 17 ciclos de vida, contudo a zona de incerteza se mostrou muito estreita durante a maior parte dos ciclos e sobreposta à RUL predita nos últimos 8 ciclos de vida para R1\_5 e 6 ciclos de vida para R2\_4, também prejudicando a confiabilidade das predições.

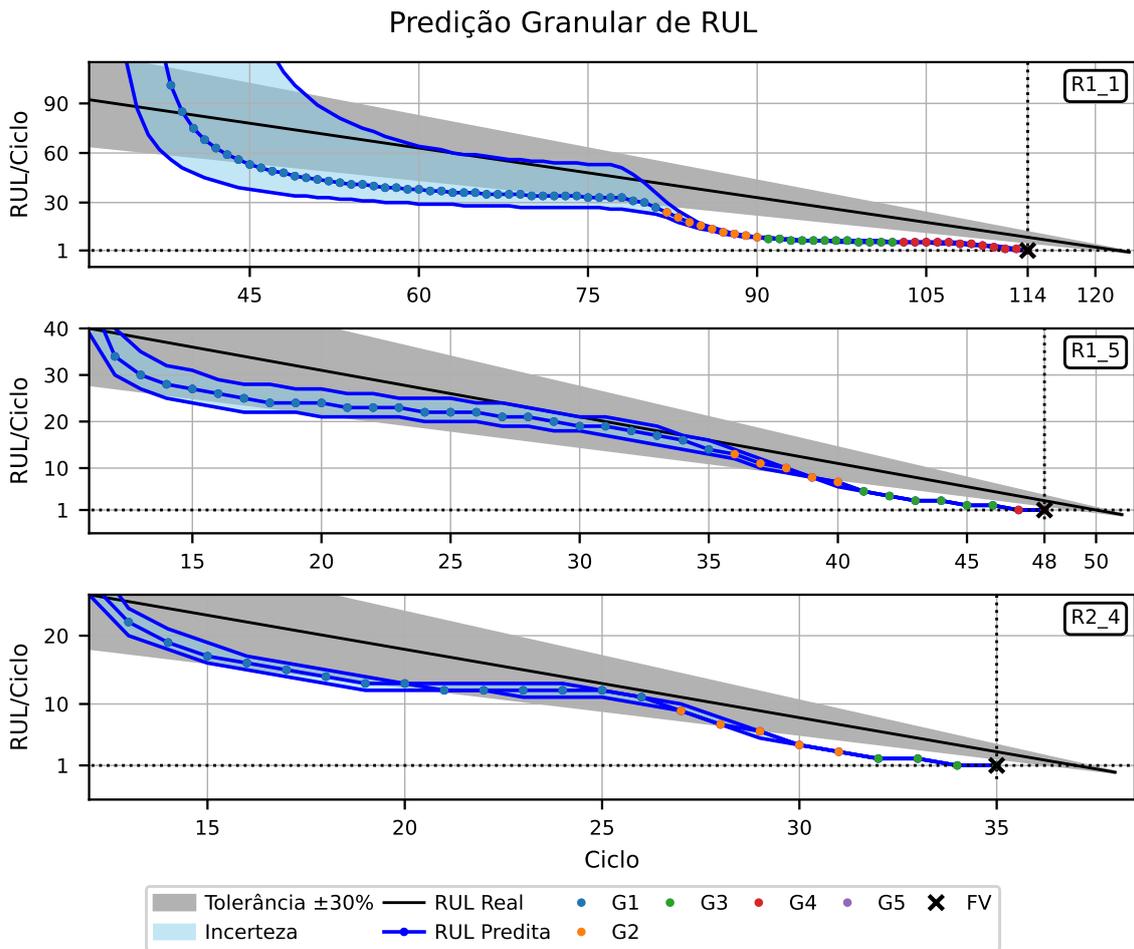


Figura 5.12: Predição de RUL granular dos Rolamentos 1\_4, 1\_5 e 2\_4.

A análise simultânea da granulação exposta nos gráficos das Figuras 5.11 e 5.12 não revela qualquer relação entre as mudanças no padrão de desgaste dos rolamentos e a criação de GIs, comprometendo a explicabilidade do modelo.

Embora o aumento da quantidade de grânulos ao longo do tempo revele a variação estrutural do modelo, os GIs não agregam interpretabilidade ao modelo. Conseqüentemente, embora haja melhoria na acurácia das predições de RUL devido

ao ajuste recursivo dos parâmetros  $\Theta$  dos RLSVDFs de cada grânulo, o modelo ainda é estruturalmente opaco. O aprendizado contínuo pode ser melhor visualizado nos gráficos da Figura 5.13, onde é exposto o RMSE de cada rolamento ao longo do tempo.

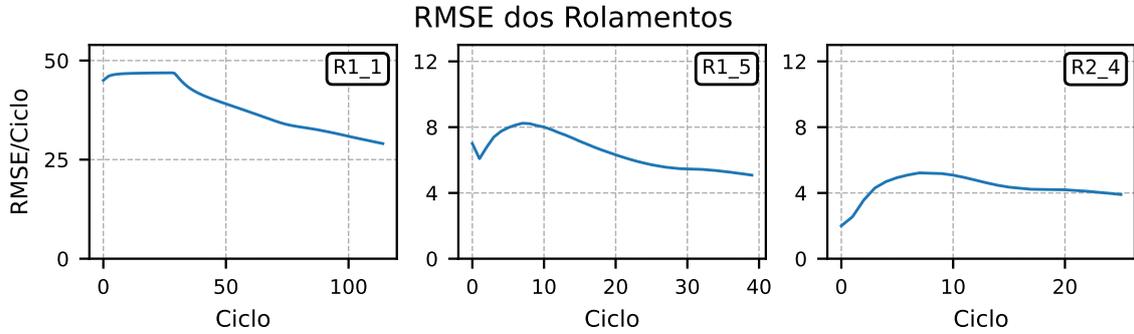


Figura 5.13: RMSE da previsão de RUL dos Rolamentos 1\_4, 1\_5 e 2\_4.

#### 5.5.4 EGRNN

A Tabela 5.5 apresenta a melhor combinação de hiperparâmetros para o EGNN-RBF de consequente RLRNN, onde  $l$  é o tamanho da janela de dados,  $h_n$  é o tamanho do vetor de unidades recorrentes,  $\eta_1$ ,  $\eta_2$  e  $\eta_3$  são as taxas de aprendizado das matrizes de pesos das camadas de entrada, recorrente e saída da RLRNN,  $t$  é o fator de ponderação da RLRNN,  $m$  é o fator de sensibilidade para criação de grânulos e  $\rho$  é a granularidade.

Rolamento	$l$	$h_n$	$\eta_1$	$\eta_2$	$\eta_3$	$m$	$t$	$\rho$	MAPE
R1_1	5	18	$1,45 \cdot 10^{-5}$	$2,50 \cdot 10^{-5}$	$1,13 \cdot 10^{-3}$	1,26	10	0,12	398,78
R1_5	5	18	$3,03 \cdot 10^{-5}$	$5,25 \cdot 10^{-3}$	$1,68 \cdot 10^{-5}$	1,37	5	0,17	29,40
R2_4	5	15	$4,27 \cdot 10^{-3}$	$2,72 \cdot 10^{-3}$	$1,43 \cdot 10^{-3}$	1,30	23	0,08	22,42

Tabela 5.5: EGRNN - Otimização de hiperparâmetros.

Os gráficos superiores da Figura 5.14 ilustram a granulação sobre o IED de cada rolamento, enquanto os gráficos inferiores ilustram o IS granulado pelo IED. Em ambos os casos, destacam-se os instantes e valores de IED e IS no FV estabelecido, além dos grânulos (G\_1 a G\_8) criados ao longo do tempo. Ademais,

considerando a ocorrência do mesmo tipo de falha em todos os rolamentos, notam-se similaridades entre as curvas de IED e IS.

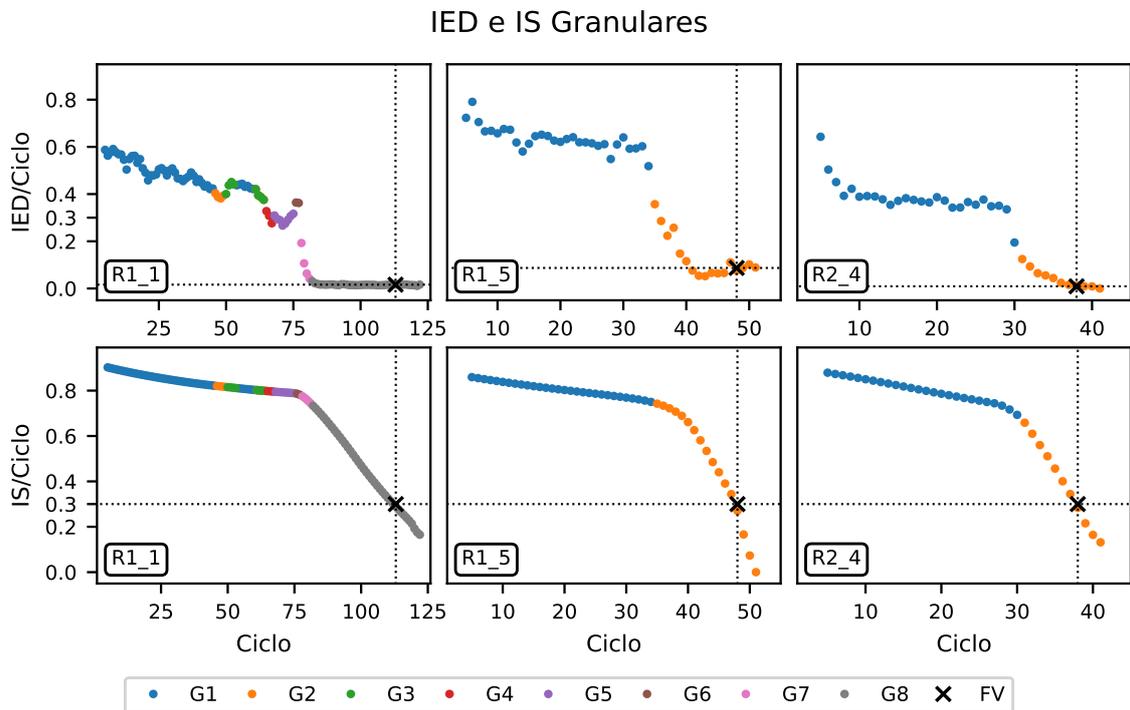


Figura 5.14: Granulação dos indicadores de estágio e saúde dos Rolamentos 1\_4, 1\_5 e 2\_4.

Embora semelhantes, notam-se valores e padrões de dispersão ligeiramente distintos nos gráfico de IED. Observam-se diferentes intervalos de valores de IED durante a vida útil, mas que se aproximam de 0 no instante de tempo destacado por FV, 0.017 para R1\_1, 0.075 para R1\_5 e 0.012 para R2\_4. Em contrapartida, considerando o intervalo de tempo entre o início e o FV, os valores e formatos dos gráficos de IS são bem semelhantes.

Nas curvas plotadas nos gráficos de IED, os dados são distribuídos em platôs verticalmente separados no tempo, reforçando a ideia de estágios de degradação. Por sua vez, os dados consecutivamente plotados nos gráficos de IS apresentam uma leve taxa de variação seguida por uma taxa mais elevada, demonstrando a aceleração no desgaste dos rolamentos.

Do ponto de vista granular, os GIs criados pelos IEDs ocupam separadamente os platôs formados, sendo evidente a predominância de um único grânulo no platô inferior. Por conta das diferentes quantidades de amostras, dos formatos da disper-

são dos dados e diferentes valores de  $m$  calculados de cada rolamento, formam-se oito GIs para R1\_1 e apenas dois para R1\_5 e R2\_4. Vale ressaltar a ausência fusão de grânulos nas simulações registradas.

A granulação dos dados de IED confere interpretabilidade e explicabilidade ao modelo ao observar seu reflexo sobre o IS. Sua análise simultânea com os gráficos de IS mostram que o último estágio de degradação está associado ao período de rápida degradação do IS, G8 para R1\_1 e G2 para R1\_5 e R2\_4. Além disso, o último estágio de degradação está associado a um único GI em todos os rolamentos.

Os resultados de predição de RUL para cada rolamento pode ser analisada na Figura 5.15. É perceptível que, durante as predições iniciais, o algoritmo não é capaz de fornecer estimativas precisas de RUL. No entanto, por meio do aprendizado contínuo, o modelo ajusta os valores das matrizes de pesos  $U$ ,  $V$  e  $W$  das RNNs, cria novos grânulos e, por fim, atinge predições mais próximas dos valores reais.

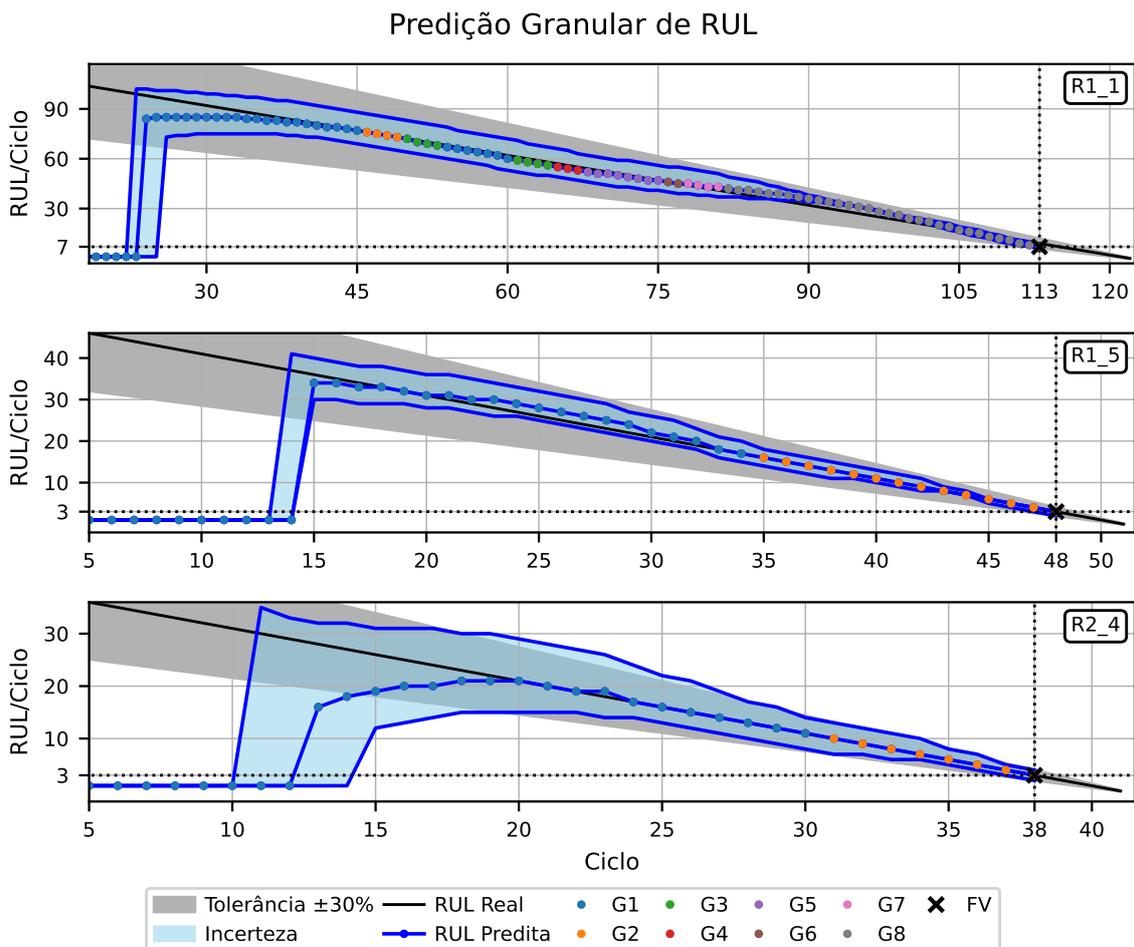


Figura 5.15: Predição de RUL granular dos Rolamentos 1\_4, 1\_5 e 2\_4.

Dada uma determinada quantidade de ciclos, o modelo foi capaz de entregar previsões de RUL próximas dos valores reais em todos os modelos, iniciando no ciclo 24 para R1\_1, ciclo 15 para R1\_5 e ciclo 19 para R2\_4. Nos mesmos ciclos, as zonas de incerteza cobertas por cada simulação também se adequaram e acompanharam o afunilamento da zona de tolerância de  $\pm 30\%$ . Finalmente, do ponto de vista granular, a predição de RUL é predominantemente precisa nos últimos EDs.

O aumento da quantidade de grânulos ao longo do tempo revela a variação estrutural do modelo enquanto a melhoria na acurácia das previsões de RUL ocorre devido ao ajuste recursivo das matrizes de peso  $U$ ,  $V$  e  $W$  das RNNs de cada grânulo. O aprendizado contínuo pode ser melhor visualizado nos gráficos da Figura 5.16, onde é exposto o RMSE de cada rolamento ao longo do tempo.

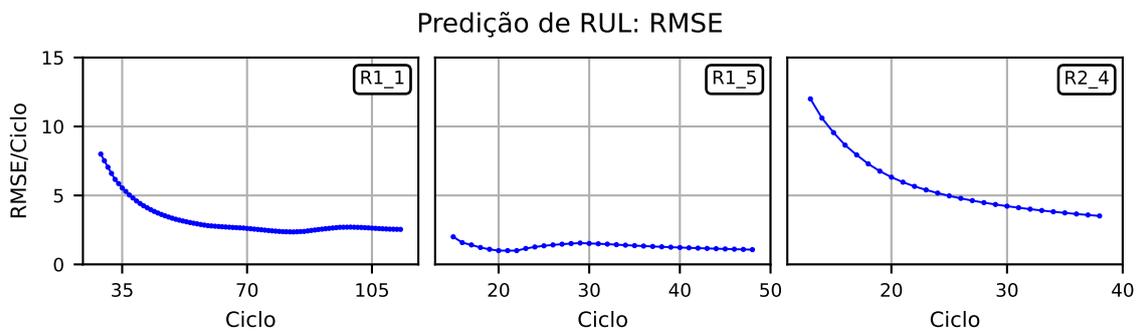


Figura 5.16: RMSE da predição de RUL dos Rolamentos 1\_4, 1\_5 e 2\_4.

## 5.6 Avaliação entre Modelos

Há uma escassez de publicações na literatura acerca de abordagens evolutivas fundamentadas na computação granular para cálculo do prognóstico de falhas em rolamentos, conforme apresentado na Seção 3.9 deste trabalho. Ademais, até o dia 10 de julho de 2025, não há registros de cálculos de prognóstico com abordagens evolutivas que utilizem a mesma base de dados usada neste trabalho, o que dificulta a realização de comparações entre modelos.

Devido os parâmetros terem sido otimizados em função da redução do MAPE, o tamanho das janelas de amostras utilizadas em cada modelo para cada rolamento não é o mesmo, o que também prejudica a comparações entre eles.

Do ponto de vista do aprendizado contínuo, todos os modelos apresentam

ajuste recursivo de parâmetros ( $\Theta$  para o RLSVDF e  $U, V$  e  $W$  para a RLRNN) e entregam predições de RUL mais próximas do real ao longo do tempo.

O efeito do ajuste de parâmetros no aprendizado pode ser observado nas predições de RUL das iterações referentes ao primeiro grânulo criado (G1) em todos rolamentos para todos modelos (Figuras 5.9, 5.12 e 5.15). Nesse período, mantendo uma estrutura mono-granular os modelos conseguem atingir predições com mais acuracidade apenas ajustando parâmetros.

A análise do RMSE (gráficos das Figuras 5.10, 5.13 e 5.16) também ilustra o aprendizado contínuo. Observam-se casos em que o valor máximo de RMSE é verificado no primeiro ciclo, reduzindo subsequentemente, ou casos em que o RMSE atinge seu valor máximo nos primeiros ciclos e posteriormente se reduz. Vale ressaltar que, fora o Preditor AutoCloud para R1\_1 e o EGNN-RBF para R2\_4, todos os rolamentos para todos os modelos apresentam valores finais de RMSE inferiores aos máximos registrados, reforçando o aprendizado recursivo.

A comparação entre os desempenhos dos modelos diante dos mesmos rolamentos se dá pelo RMSE medido consecutivamente nos instantes referentes a 25%, 50%, 75% e 100% das quantidades de RUL preditas até o FV. A Tabela 5.6 apresenta o registro dos valores observados durante os testes.

<b>Rolamento</b>	<b>RMSE</b>	<b>P.AutoCloud</b>	<b>EGRBFN</b>	<b>EGRNN</b>
R1_1	RMSE <sub>25%</sub>	79,755	46,890	3,275
	RMSE <sub>50%</sub>	74,059	38,437	2,583
	RMSE <sub>75%</sub>	61,536	33,366	2,586
	RMSE <sub>100%</sub>	53,515	30,211	2,539
R1_5	RMSE <sub>25%</sub>	78,977	8,099	1,155
	RMSE <sub>50%</sub>	57,298	6,621	1,472
	RMSE <sub>75%</sub>	47,207	5,564	1,225
	RMSE <sub>100%</sub>	41,074	5,260	1,071
R2_4	RMSE <sub>25%</sub>	57,413	4,932	6,761
	RMSE <sub>50%</sub>	40,602	4,941	4,788
	RMSE <sub>75%</sub>	33,164	4,222	4,006
	RMSE <sub>100%</sub>	29,356	4,101	3,514

Tabela 5.6: Comparação entre os valores de RMSE para as predições dos modelos.

Podemos observar que o EGRNN apresentou os menores valores de RMSE no

FV verificado. Além disso, fora as simulações do EGNN-RBF em R2\_4 e do EGRNN em R1\_5, todas as simulações apresentaram reduções consecutivas no valor de erro calculado.

Do ponto de vista granular, embora o AutoCloud tenha sido usado em todos os modelos, diferentes tipos de dados foram assimilados no processo de granulação. As simulações do Preditor AutoCloud foram realizadas utilizando as séries de ISs para granulação e predição, para o EGNN-RBF as séries de ISs foram previamente parametrizadas por uma RBF e para o EGRNN utilizou-se uma série de IED para granulação e outra de IS para predição. A Tabela 5.7 mostra a quantidade de grânulos criados ao fim da simulação de cada modelo.

Rolamento	Quantidade de Grânulos		
	P.AutoCloud	EGRBFN	EGRNN
R1_1	2	5	8
R1_5	3	4	2
R2_4	4	4	2

Tabela 5.7: Comparação entre as quantidades de grânulos formadas para cada modelo.

As abordagens provocaram diferenças significativas na variação estrutural e na interpretabilidade do modelo. De fato, todos os modelos expandiram estruturalmente para se adequar às variações assimiladas dos dados, contudo as relações entre os grânulos criados e os dados observados não é a mesma.

Embora o Preditor AutoCloud tenha criado grânulos coincidentemente sobre regiões das curvas de ISs onde se observam taxas de variação distintas (Figura 5.8), a quantidade de simulações realizadas é baixa para confirmar a existência de um vínculo entre os grânulos e a velocidade de degradação, mitigando a explicabilidade das expansões estruturais observadas.

Para o EGNN-RBF, devido a introdução de não linearidade aos dados pelo uso da RBF, os grânulos observados nos gráficos da Figura 5.11 são apenas reflexo dos dados granulados e não sugerem nenhum tipo de associação com os dados de IS. Assim, a granulação não é capaz de explicitar os fatores que desencadearam a variação estrutural.

O uso de uma variável de premissa exclusiva para granulação IED associada

a uma variável exclusiva para predição IS, proposto pelo EGRNN, se mostrou a abordagem mais eficiente em termos de interpretabilidade/explicabilidade. Os gráficos da Figura 5.14 puderam demonstrar evidentemente relação entre os os grânulos criados pelo IED e os trechos onde o desgaste ocorre de forma acelerada, justificando a expansão estrutural.

Por fim, a granulação de dados também permitiu a quantificação de incerteza dos modelos, partindo da granularidade  $\rho$  (resultante do PJG). A abordagem de parâmetros granulares adotada no P.AutoCloud e no EGNN-RBF não demonstrou bom desempenho, criando zonas de incerteza muito estreitas, fora da margem de tolerância, ou totalmente sobrepostas à predição singular de RUL, mesmo os modelos apresentando alguma acurácia. Por outro lado, a abordagem de predições granulares através da propagação de uma estimativa de erro demonstrou melhor desempenho ao entregar uma zona de incerteza que permanece dentro da zona de tolerância e afunila progressivamente.

# Capítulo 6

## Conclusão

Este trabalho explorou a abordagem granular e evolutiva no desenvolvimento de algoritmos de predição aplicados à predição de RUL. Fundamentados no TEDA e no AutoCloud, foram concebidos os modelos P.AutoCloud, EGNN-RBF e EGRNN.

Através da formulação de preditores autorregressivos com ajuste recursivo de parâmetros, os modelos propostos realizaram predições sequenciais com erros consecutivamente menores (a partir de um determinado instante), demonstrando sua capacidade de aprender continuamente por meio dos dados, mesmo partindo da completa ausência de conhecimento e sem necessidade de armazenar dados passados.

Graças à fundamentação no AutoCloud, os modelos foram capazes de criar representações compactas dos padrões observados nos dados, os grânulos de informação, e a eles associar preditores (RLSVDF ou RLRNN) personalizados capazes de estimar boas predições a partir de dados contemplados por um respectivo GI.

Por outro lado, os GIs não necessariamente atribuíram interpretabilidade e explicabilidade aos modelos. Enquanto no EGRNN a projeção dos GIs (criados a partir do IED) sobre o IS pôde evidenciar a associação de diferentes EDs a trechos do IS caracterizados por diferentes velocidades de degradação, nos outros modelos a ausência de uma variável específica para granulação mitigou a explicação da criação de grânulos. Embora no P.AutoCloud os grânulos pareçam estar associados às velocidades de degradação do IS, a evidência não é suficiente para confirmar tal vínculo. Enfim, no EGNN-RBF a granulação do IS parametrizado por uma RBF não foi atribuir explicabilidade à criação de grânulos.

Finalmente, o modelo EGRNN se destacou entre os outros por seu bom de-

sempenho em todos os âmbitos dos objetivos listados neste trabalho. O aprendizado recursivo da RLRNN permitiu a criação de um modelo inteligente capaz de autoajustar seus parâmetros e estrutura em função dos dados consecutivamente assimilados, independentemente de treinamento prévio ou de armazenamento de dados. Ademais, o AutoCloud foi capaz de atribuir interpretabilidade e explicabilidade às variações estruturais observadas ao longo do tempo.

Os resultados obtidos demonstram o EGRNN como uma abordagem promissora para prognóstico de falhas em rolamentos, contudo mais simulações devem ser realizadas para avaliar seu desempenho diante de outros tipos de falha não exploradas neste trabalho. Vale ressaltar a existência de diversas oportunidades a serem exploradas que eventualmente podem melhorar o desempenho do modelo diante dos aspectos analisados neste trabalho, como metodologias alternativas para aprendizado contínuo, variação estrutural, granulação de dados, quantificação de incerteza, além da e modelos de predição incorporação de outros preditores (LSTM, *Gated Recurrent Unit* (GRU), CNN etc).

## 6.1 Publicações Derivadas

Durante o desenvolvimento desta dissertação, foram conduzidos diversos estudos auxiliares que resultaram em quatro artigos científicos tratando dos diferentes modelos baseados em TEDA propostos. Cada publicação explora o uso de modelos evolutivos para prognóstico de falhas em rolamentos, abordando variantes do preditor AutoCloud sob diferentes tipos de dados usados para clusterização e diferentes consequentes utilizados para predição.

### 6.1.1 Prognóstico de falhas em rolamentos baseado em modelos granulares evolutivos.

Publicado no Simpósio Brasileiro de Automação Industrial (SBAI), em 2025, este artigo explora o uso do algoritmo AutoCloud associado a um consequente linear  $\hat{Y}_h = X_h^\top \Theta_i$  para realizar predições. O proposto modelo P.AutoCloud utiliza o algoritmo RLS estimar valores de  $\Theta$  e consegue realizar predições satisfatórias, mas não é capaz de confirmar as hipóteses de vínculos entre os grânulos criados e as

variações estruturais observadas, além de não incluir cálculos de incerteza.

### ***6.1.2 Incremental learning-based rolling bearing prognostics with granular uncertainty quantification.***

Publicado na *6th International Conference on Control and Fault-Tolerant Systems* (SysTol), em 2025, este artigo explora o uso do algoritmo AutoCloud associado a um consequente linear  $\hat{Y}_h = X_h^\top \Theta_i$  para estimar e granular dados de IS. O proposto modelo P.AutoCloud utiliza o algoritmo RLSVDF para estimar valores de  $\Theta$  e realiza boas previsões, pontualmente. Contudo não é capaz de confirmar as hipóteses de vínculos entre os grânulos criados e as variações estruturais observadas. Ademais, o modelo introduziu a quantificação da incerteza das previsões através da ponderação de  $\Theta$  pela granularidade  $\rho$  calculada através do PJG, porém o intervalo de confiança encontrado se mostrou muito largo.

### ***6.1.3 Prognostics of Rolling Bearings based on Neuro-Fuzzy Evolving Models.***

Publicado na *IFAC Joint Conference on Computers, Cognition and Communication* (J3C), em 2025, este artigo explora o uso do algoritmo AutoCloud associado a um consequente linear  $\hat{Y}_h = X_h^\top \Theta_i$  para realizar previsões, onde  $\Theta$  é estimado pelo algoritmo RLSVDF. Contudo, ao invés de granular diretamente dados de IS, o modelo explora o uso de RBFs para agregar uma componente de não linearidade na tentativa de contornar a complexidade dos dados, o que inviabilizou a observação de vínculos entre os grânulos criados e respectivos valores de IS contemplados. Apesar de ter apresentado previsões ocasionalmente ruins o modelo contemplou a quantificação de incerteza das previsões através da ponderação de  $\Theta$  pela granularidade  $\rho$  calculada através do PJG, resultando em intervalos de confiança eventualmente satisfatórios.

#### 6.1.4 *Prognostics of Rolling Bearings based on Neuro-Fuzzy Evolving Models.*

Submetido à *ISA Transactions: The Journal of Automation*, em 2025, este artigo explora o uso do algoritmo AutoCloud associado a um consequente não linear  $\hat{Y}_h = \text{RNN}(X_h)$ . Apresentando o uso de uma variável exclusiva para granulação de dados, o IED, e outra para realização de previsões, o IS, o modelo consegue introduzir explicabilidade às variações estruturais através de grânulos evidentemente criados sobre diferentes EDs e associá-los a ISs com velocidades de degradação distintas. O uso de RNNs como preditores proporcionou bom desempenho ao modelo, garantindo a acurácia das previsões com poucas iterações. Ademais, o proposto método de quantificação de incerteza através da estimativa do erro de previsão proporcionou bons intervalos de confiança, tangentes à margem de tolerância das previsões de RUL.

# Referências Bibliográficas

- Al-Hmouz, R., Pedrycz, W., Awadallah, M., and Ammari, A. (2024). Granular transfer learning. *Neurocomputing*, 600:128126.
- Alfarizi, M. G., Tajiani, B., Vatn, J., and Yin, S. (2023). Optimized random forest model for remaining useful life prediction of experimental bearings. *IEEE Transactions on Industrial Informatics*, 19(6):7771 – 7779. Cited by: 29.
- Andrews, R., Diederich, J., and Tickle, A. B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based systems*, 8(6):373–389.
- Angelov, P. (2014a). Anomaly detection based on eccentricity analysis. In *2014 IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*, pages 1–8.
- Angelov, P. (2014b). Outside the box: an alternative data analytics framework. *Journal of Automation Mobile Robotics and Intelligent Systems*, 8(2):29–35.
- Angelov, P. and Yager, R. (2012). A new type of simplified fuzzy rule-based system. *International Journal of General Systems*, 41(2):163–185.
- Angelov, P. P. (2002). *Evolving rule-based models: a tool for design of flexible adaptive systems*, volume 92. Springer Science & Business Media.
- Angelov, P. P. and Filev, D. P. (2004). An approach to online identification of takagi-sugeno fuzzy models. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):484–498.
- Bagri, I., Tahiry, K., Hraiba, A., Touil, A., and Mousrij, A. (2024). Vibration signal

- analysis for intelligent rotating machinery diagnosis and prognosis: A comprehensive systematic literature review. *Vibration*, 7(4):1013–1062.
- Bargiela, A. and Pedrycz, W. (2003). Recursive information granulation: aggregation and interpretation issues. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 33(1):96–112.
- Behera, S. and Misra, R. (2023). A multi-model data-fusion based deep transfer learning for improved remaining useful life estimation for iiot based systems. *Engineering Applications of Artificial Intelligence*, 119. Cited by: 17.
- Bejaoui, I., Bruneo, D., and Xibilia, M. G. (2020). A data-driven prognostics technique and rul prediction of rotating machines using an exponential degradation model. In *2020 7th International conference on control, decision and information technologies (CoDIT)*, volume 1, pages 703–708. IEEE.
- Bengio, Y., Goodfellow, I., Courville, A., et al. (2017). *Deep learning*, volume 1. MIT press Cambridge, MA, USA.
- Bezerra, C. G., Costa, B. S. J., Guedes, L. A., and Angelov, P. P. (2020). An evolving approach to data streams clustering based on typicality and eccentricity data analytics. *Information Sciences*, 518:13–28.
- Boutrous, K., Bessa, I., Nejari, F., and Puig, V. (2022a). *A Set-Based Uncertainty Quantification of Evolving Fuzzy Models for Data-Driven Prognostics*, page 293–304. Springer International Publishing.
- Boutrous, K., Bessa, I., Puig, V., Nejari, F., and Palhares, R. M. (2022b). Data-driven prognostics based on evolving fuzzy degradation models for power semiconductor devices. *PHM Society European Conference*, 7(1):68–77.
- Burrell, J. (2016). How the machine ‘thinks’: Understanding opacity in machine learning algorithms. *Big data & society*, 3(1):2053951715622512.
- Camargos, M., Bessa, I., Junior, L. A. Q. C., Coutinho, P., Leite, D. F., and Palhares, R. M. (2021). Evolving fuzzy system applied to battery charge capacity prediction for fault prognostics. In *Joint Proceedings of the 19th World Congress of the International Fuzzy Systems Association (IFSA)*. Atlantis Press.

- Camargos, M. O., Bessa, I., D'Angelo, M. F. S. V., Cosme, L. B., and Palhares, R. M. (2020). Data-driven prognostics of rolling element bearings using a novel error based evolving takagi–sugeno fuzzy model. *Appl. Soft Comput.*, 96:106628.
- Chakraborty, S., Tomsett, R., Raghavendra, R., Harborne, D., Alzantot, M., Cerutti, F., Srivastava, M., Preece, A., Julier, S., Rao, R. M., et al. (2017). Interpretability of deep learning models: A survey of results. In *2017 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big data computing, Internet of people and smart city innovation (smartworld/SCALCOM/UIC/ATC/CBDcom/IOP/SCI)*, pages 1–6. IEEE.
- Chang, Y., Li, F., Chen, J., Liu, Y., and Li, Z. (2022). Efficient temporal flow transformer accompanied with multi-head probsparse self-attention mechanism for remaining useful life prognostics. *Reliability Engineering and System Safety*, 226. Cited by: 52.
- Chen, C., Wang, T., Liu, Y., Cheng, L., and Qin, J. (2022a). Spatial attention-based convolutional transformer for bearing remaining useful life prediction. *Measurement Science and Technology*, 33(11). Cited by: 23; All Open Access, Green Open Access.
- Chen, Y., Hu, X., and Zhang, L. (2022b). A review of ultra-short-term forecasting of wind power based on data decomposition-forecasting technology combination model. *Energy Reports*, 8:14200–14219.
- Cheng, Y., Hu, K., Wu, J., Zhu, H., and Lee, C. K. M. (2022a). A deep learning-based two-stage prognostic approach for remaining useful life of rolling bearing. *Applied Intelligence*, 52(5):5880 – 5895. Cited by: 22.
- Cheng, Y., Hu, K., Wu, J., Zhu, H., and Shao, X. (2022b). Autoencoder quasi-recurrent neural networks for remaining useful life prediction of engineering systems. *IEEE/ASME Transactions on Mechatronics*, 27(2):1081 – 1092. Cited by: 63.
- Cordovil Jr, L. A., Coutinho, P. H., Bessa, I., Peixoto, M. L., and Palhares, R. M. (2022). Learning event-triggered control based on evolving data-driven fuzzy gra-

- nular models. *International Journal of Robust and Nonlinear Control*, 32(5):2805–2827.
- Cosme, L. B., D’Angelo, M. F., Caminhas, W. M., Yin, S., and Palhares, R. M. (2018). A novel fault prognostic approach based on particle filters and differential evolution. *Applied Intelligence*, 48:834–853.
- Costa, B. S. J., Bezerra, C. G., Guedes, L. A., and Angelov, P. P. (2015). Online fault detection based on typicality and eccentricity data analytics. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE.
- Dario, R. (2024). Bearing damage analysis: Iso 15243 is here to help you. <https://evolution.skf.com/bearing-damage-analysis-iso-15243-is-here-to-help-you/>. Acesso em: 31 mar. 2025.
- Deng, F., Bi, Y., Liu, Y., and Yang, S. (2022). Remaining useful life prediction of machinery: A new multiscale temporal convolutional network framework. *IEEE Transactions on Instrumentation and Measurement*, 71. Cited by: 32.
- Deng, Y., Du, S., Wang, D., Shao, Y., and Huang, D. (2023). A calibration-based hybrid transfer learning framework for rul prediction of rolling bearing across different machines. *IEEE Transactions on Instrumentation and Measurement*, 72. Cited by: 85.
- Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.
- Dragomiretskiy, K. and Zosso, D. (2014). Variational mode decomposition. *IEEE Transactions on Signal Processing*, 62(3):531–544.
- Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., and Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE.

- Gomez, T., Fréour, T., and Mouchère, H. (2022). Metrics for saliency map evaluation of deep learning explanation methods. In *International Conference on Pattern Recognition and Artificial Intelligence*, pages 84–95. Springer.
- Gu, Y., Bi, Q., and Qiu, G. (2022). Practical health indicator construction methodology for bearing ensemble remaining useful life prediction with isomap-de and elm-wphm. *Measurement Science and Technology*, 33(2). Cited by: 11.
- Guo, J., Wang, Z., Li, H., Yang, Y., Huang, C.-G., Yazdi, M., and Kang, H. S. (2024). A hybrid prognosis scheme for rolling bearings based on a novel health indicator and nonlinear wiener process. *Reliability Engineering and System Safety*, 245. Cited by: 40; All Open Access, Hybrid Gold Open Access.
- Guo, W. and He, M. (2022). An integrated method for bearing state change identification and prognostics based on improved relevance vector machine and degradation model. *IEEE Transactions on Instrumentation and Measurement*, 71. Cited by: 16.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417.
- Jacob, S. G. and Ramani, R. G. (2012). Evolving efficient clustering and classification patterns in lymphography data through data mining techniques. *International Journal on Soft Computing*, 3(3):119.
- James, P. (2024). Bearing failure outer race. <https://assetmanagementprofessionals.org/discussion/bearing-failure-outer-race>. Acesso em: 31 mar. 2025.
- Kalgren, P. W., Byington, C. S., Roemer, M. J., and Watson, M. J. (2006). Defining phm, a lexical evolution of maintenance and logistics. In *2006 IEEE autotestcon*, pages 353–358. IEEE.
- Kan, M. S., Tan, A. C., and Mathew, J. (2015). A review on prognostic techniques for non-stationary and non-linear rotating systems. *Mechanical Systems and Signal Processing*, 62:1–20.

- Kangin, D. and Angelov, P. (2015). Evolving clustering, classification and regression with tedar. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.
- Kangin, D., Angelov, P., and Iglesias, J. A. (2016). Autonomously evolving classifier tedarclass. *Information Sciences*, 366:1–11.
- Kasabov, N. and Song, Q. (2002). Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, 10(2):144–154.
- Kim, N.-H., An, D., and Choi, J.-H. (2017). Prognostics and health management of engineering systems. *Switzerland: Springer International Publishing*.
- Kong, W. and Li, H. (2022). Remaining useful life prediction of rolling bearing under limited data based on adaptive time-series feature window and multi-step ahead strategy. *Applied Soft Computing*, 129. Cited by: 21.
- Lei, Y., Li, N., Guo, L., Li, N., Yan, T., and Lin, J. (2018). Machinery health prognostics: A systematic review from data acquisition to rul prediction. *Mechanical systems and signal processing*, 104:799–834.
- Leite, D., Costa, P., and Gomide, F. (2010). Evolving granular neural network for semi-supervised data stream classification. In *The 2010 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE.
- Leite, D., Gomide, F., Ballini, R., and Costa, P. (2011). Fuzzy granular evolving modeling for time series prediction. In *2011 IEEE international conference on fuzzy systems (FUZZ-IEEE 2011)*, pages 2794–2801. IEEE.
- Leite, D. and Škrjanc, I. (2019). Ensemble of evolving optimal granular experts, owa aggregation, and time series prediction. *Information sciences*, 504:95–112.
- Leite, D., Škrjanc, I., and Gomide, F. (2020). An overview on evolving systems and learning from stream data. *Evolving systems*, 11(2):181–198.
- Leite, D. F., Costa, P., and Gomide, F. (2009). Interval-based evolving modeling. In *2009 IEEE Workshop on Evolving and Self-Developing Intelligent Systems*, pages 1–8. IEEE.

- Lemos, A., Caminhas, W., and Gomide, F. (2013). Evolving intelligent systems: Methods, algorithms and applications. *Emerging paradigms in machine learning*, pages 117–159.
- Lesot, M.-J. and Marsala, C. (2020). Fuzzy approaches for soft computing and approximate reasoning: Theories and applications. *Switzerland: Springer*.
- Levada, A. L. (2021). Pca-kl: a parametric dimensionality reduction approach for unsupervised metric learning. *Advances in Data Analysis and Classification*, 15(4):829–868.
- Li, G., Wei, J., He, J., Yang, H., and Meng, F. (2023). Implicit kalman filtering method for remaining useful life prediction of rolling bearing with adaptive detection of degradation stage transition point. *Reliability Engineering and System Safety*, 235. Cited by: 30.
- Li, H., Liu, T., Wu, X., and Chen, Q. (2019). Application of eemd and improved frequency band entropy in bearing fault feature extraction. *ISA transactions*, 88:170–185.
- Li, H., Liu, T., Wu, X., and Chen, Q. (2020). An optimized vmd method and its applications in bearing fault diagnosis. *Measurement*, 166:108185.
- Li, S., Zhang, C., Liu, L., and Zhang, X. (2024a). Gated transient fluctuation dual attention unit network for long-term remaining useful life prediction of rotating machinery using iiot. *IEEE Internet of Things Journal*, 11(10):18593 – 18604. Cited by: 7.
- Li, W. and Deng, L. (2023). A hybrid model-based prognostics approach for estimating remaining useful life of rolling bearings. *Measurement Science and Technology*, 34(10). Cited by: 10; All Open Access, Bronze Open Access.
- Li, Z.-J., Cheng, D.-J., Zhang, H.-B., Zhou, K.-L., and Wang, Y.-F. (2024b). Multi-feature spaces cross adaption transfer learning-based bearings piece-wise remaining useful life prediction under unseen degradation data. *Advanced Engineering Informatics*, 60. Cited by: 9.

- Liu, T., Chen, J., Dong, G., Xiao, W., and Zhou, X. (2013). The fault detection and diagnosis in rolling element bearings using frequency band entropy. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 227(1):87–99.
- Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.
- Maciel, L., Ballini, R., and Gomide, F. (2016). Evolving granular analytics for interval time series forecasting. *Granular Computing*, 1:213–224.
- Mao, W., Chen, J., Liu, J., and Liang, X. (2023). Self-supervised deep domain-adversarial regression adaptation for online remaining useful life prediction of rolling bearing under unknown working condition. *IEEE Transactions on Industrial Informatics*, 19(2):1227 – 1237. Cited by: 44.
- Moura, J. (2009). What is signal processing[president’s message]. *IEEE Signal Processing Magazine*, 26(6):6–6.
- Murray, J. M. (2019). Local online learning in recurrent networks with random feedback. *Elife*, 8:e43299.
- Nectoux, P., Gouriveau, R., Medjaher, K., Ramasso, E., Chebel-Morello, B., Zerhouni, N., and Varnier, C. (2012). Pronostia: An experimental platform for bearings accelerated degradation tests. In *IEEE International Conference on Prognostics and Health Management, PHM’12.*, pages 1–8. IEEE Catalog Number: CPF12PHM-CDR.
- Ni, Q., Ji, J., and Feng, K. (2022). Data-driven prognostic scheme for bearings based on a novel health indicator and gated recurrent unit network. *IEEE Transactions on Industrial Informatics*, 19(2):1301–1311.
- NSK (2024). Bearing failure rca: Cage damage. <https://www.bearing-news.com/bearing-failure-rca-cage-damage/>. Acesso em: 31 mar. 2025.
- NTN (2024). Care and maintenance of bearings. <https://www.ntnglobal.com/en/products/catalog/pdf/3017E.pdf>. Acesso em: 31 mar. 2025.

- Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198.
- Ortigossa, E. S., Gonçalves, T., and Nonato, L. G. (2024). Explainable artificial intelligence (xai)—from theory to methods and applications. *IEEE Access*.
- Pedrycz, W. (2000). *Granular computing: an introduction*. Springer.
- Pedrycz, W. (2001). Granular computing: an introduction. In *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*, volume 3, pages 1349–1354 vol.3.
- Pedrycz, W. and Homenda, W. (2013). Building the fundamentals of granular computing: A principle of justifiable granularity. *Applied Soft Computing*, 13(10):4209–4218.
- Polverino, L., Abbate, R., Manco, P., Perfetto, D., Caputo, F., Macchiaroli, R., and Caterino, M. (2023). Machine learning for prognostics and health management of industrial mechanical systems and equipment: A systematic literature review. *International Journal of Engineering Business Management*, 15:18479790231186848.
- Pradhan, A. (2020). A survey of classification of eeg signals using emd and vmd for epileptic seizure detection. *International Journal of Engineering Research & Technology*, 9(5):495–498.
- Qi, J., Zhu, R., Liu, C., Mauricio, A., and Gryllias, K. (2024). Anomaly detection and multi-step estimation based remaining useful life prediction for rolling element bearings. *Mechanical Systems and Signal Processing*, 206. Cited by: 18.
- Qiu, H., Lee, J., Lin, J., and Yu, G. (2006). Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics. *Journal of sound and vibration*, 289(4-5):1066–1090.
- Ramezani, S., Moini, A., and Riahi, M. (2019). Prognostics and health management in machinery: A review of methodologies for rul prediction and roadmap. *International Journal of Industrial Engineering and Management Science*, 6(1):38–61.

- Rathore, M. S. and Harsha, S. (2022a). An attention-based stacked bilstm framework for predicting remaining useful life of rolling bearings. *Applied Soft Computing*, 131. Cited by: 32.
- Rathore, M. S. and Harsha, S. (2022b). Prognostics analysis of rolling bearing based on bi-directional lstm and attention mechanism. *Journal of Failure Analysis and Prevention*, 22(2):704 – 723. Cited by: 19.
- Rathore, M. S. and Harsha, S. (2022c). Rolling bearing prognostic analysis for domain adaptation under different operating conditions. *Engineering Failure Analysis*, 139:106414.
- Rokach, L. (2010). Ensemble-based classifiers. *Artificial intelligence review*, 33:1–39.
- Soares, E., Costa Jr, P., Costa, B., and Leite, D. (2018). Ensemble of evolving data clouds and fuzzy models for weather time series prediction. *Applied Soft Computing*, 64:445–453.
- Soares, E., Mota, V., Poucas, R., and Leite, D. (2017). Cloud-based evolving intelligent method for weather time series prediction. In *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6.
- Strömbergsson, D., Marklund, P., Berglund, K., and Larsson, P.-E. (2020). Bearing monitoring in the wind turbine drivetrain: A comparative study of the fft and wavelet transforms. *Wind Energy*, 23(6):1381–1393.
- Tang, G.-J. and Wang, X.-L. (2016). Variational mode decomposition method and its application on incipient fault diagnosis of rolling bearing. *Journal of Vibration Engineering*, 29(4):638–648.
- Thomas, M. (2024). Can a repaired motor be more efficient? <https://www.controlsdrivesautomation.com/AEMT-repaired-motor-more-efficient>. Acesso em: 31 mar. 2025.
- Thoppil, N. M., Vasu, V., and Rao, C. (2021). Health indicator construction and remaining useful life estimation for mechanical systems using vibration signal prognostics. *International Journal of System Assurance Engineering and Management*, 12:1001–1010.

- Wang, B., Lei, Y., Li, N., et al. (2018). Xjtu-sy bearing datasets. *does not apply*.
- Wang, D., Tsui, K.-L., and Miao, Q. (2017). Prognostics and health management: A review of vibration based bearing and gear health indicators. *Ieee Access*, 6:665–676.
- Wang, G., Sun, L., Wang, A., Jiao, J., and Xie, J. (2024). Lithium battery remaining useful life prediction using vmd fusion with attention mechanism and tcn. *Journal of Energy Storage*, 93:112330.
- Wang, L., Zhang, L., and Wang, X.-z. (2015). Reliability estimation and remaining useful lifetime prediction for bearing based on proportional hazard model. *Journal of Central South University*, 22(12):4625–4633.
- Wang, W., Liu, W., and Chen, H. (2022a). Time-series forecasting via fuzzy-probabilistic approach with evolving clustering-based granulation. *IEEE Transactions on Fuzzy Systems*, 30(12):5324–5336.
- Wang, Y., Wu, J., Cheng, Y., Wang, J., and Hu, K. (2022b). Memory-enhanced hybrid deep learning networks for remaining useful life prognostics of mechanical equipment. *Measurement: Journal of the International Measurement Confederation*, 187. Cited by: 20.
- Wang, Z., Zhao, W., Li, Y., Dong, L., Wang, J., Du, W., and Jiang, X. (2023). Adaptive staged rul prediction of rolling bearing. *Measurement: Journal of the International Measurement Confederation*, 222. Cited by: 13.
- Xu, Z., Bashir, M., Liu, Q., Miao, Z., Wang, X., Wang, J., and Ekere, N. (2023). A novel health indicator for intelligent prediction of rolling bearing remaining useful life based on unsupervised learning model. *Computers and Industrial Engineering*, 176. Cited by: 53; All Open Access, Green Open Access.
- Yang, S., Tang, B., Wang, W., Yang, Q., and Hu, C. (2024). Physics-informed multi-state temporal frequency network for rul prediction of rolling bearings. *Reliability Engineering and System Safety*, 242. Cited by: 24.

- Yao, X., Zhu, J., Jiang, Q., Yao, Q., Shen, Y., and Zhu, Q. (2024). Rul prediction method for rolling bearing using convolutional denoising autoencoder and bidirectional lstm. *Measurement Science and Technology*, 35(3). Cited by: 16.
- Zhang, H., Xi, X., and Pan, R. (2023). A two-stage data-driven approach to remaining useful life prediction via long short-term memory networks. *Reliability Engineering and System Safety*, 237. Cited by: 33.
- Zheng, G., Li, Y., Zhou, Z., and Yan, R. (2024). A remaining useful life prediction method of rolling bearings based on deep reinforcement learning. *IEEE Internet of Things Journal*, 11(13):22938 – 22949. Cited by: 8.
- Zhu, R., Peng, W., Wang, D., and Huang, C.-G. (2022). Bayesian transfer learning with active querying for intelligent cross-machine fault prognosis under limited data. *Mechanical Systems and Signal Processing*, 183. Cited by: 74.
- Zio, E. (2016). Some challenges and opportunities in reliability engineering. *IEEE Transactions on Reliability*, 65(4):1769–1782.
- Zio, E. (2022). Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice. *Reliability Engineering System Safety*, 218:108119.