



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E DE**  
**COMPUTAÇÃO**  
**MESTRADO ACADÊMICO EM ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO**

**JOAQUIM OSTERWALD FROTA MOURA FILHO**

**CLASSIFICADOR POR VOTAÇÃO BASEADO EM OTIMIZAÇÃO POR ENXAME DE**  
**PARTÍCULAS RELATIVÍSTICAS PARA A DETECÇÃO DE FALHAS SIMPLES E**  
**COMBINADAS EM MÁQUINAS ELÉTRICAS ROTATIVAS**

**SOBRAL**

**2023**

JOAQUIM OSTERWALD FROTA MOURA FILHO

CLASSIFICADOR POR VOTAÇÃO BASEADO EM OTIMIZAÇÃO POR ENXAME DE  
PARTÍCULAS RELATIVÍSTICAS PARA A DETECÇÃO DE FALHAS SIMPLES E  
COMBINADAS EM MÁQUINAS ELÉTRICAS ROTATIVAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica e de Computação do Programa de Pós-Graduação em Engenharia Elétrica e de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica e de Computação. Área de Concentração: Sistemas de Informação.

Orientador: Prof. Dr. Márcio André Baima Amora.

SOBRAL

2023

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

M887c Moura Filho, Joaquim Osterwald Frota.

Classificador por votação baseado em otimização por enxame de partículas relativísticas para a detecção de falhas simples e combinadas em máquinas elétricas rotativas / Joaquim Osterwald Frota Moura Filho. – 2023.

117 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Campus de Sobral, Programa de Pós-Graduação em Engenharia Elétrica e de Computação, Sobral, 2023.

Orientação: Prof. Dr. Márcio André Baima Amora.

1. Identificação de falha. 2. Falhas combinadas. 3. Classificador híbrido. 4. Máquinas elétricas rotativas.  
I. Título.

CDD 621.3

---

JOAQUIM OSTERWALD FROTA MOURA FILHO

CLASSIFICADOR POR VOTAÇÃO BASEADO EM OTIMIZAÇÃO POR ENXAME DE  
PARTÍCULAS RELATIVÍSTICAS PARA A DETECÇÃO DE FALHAS SIMPLES E  
COMBINADAS EM MÁQUINAS ELÉTRICAS ROTATIVAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica e de Computação do Programa de Pós-Graduação em Engenharia Elétrica e de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica e de Computação. Área de Concentração: Sistemas de Informação.

Aprovada em: 20/06/2023

BANCA EXAMINADORA

---

Prof. Dr. Márcio André Baima Amora (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Arthur Plínio de Souza Braga  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Vanessa Siqueira de Castro Teixeira  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Vandilberto Pereira Pinto  
Universidade da Integração Internacional da  
Lusofonia Afro-Brasileira (Unilab)

À minha família, por sua capacidade de acreditar em mim e investir em mim. Mãe, seu cuidado e dedicação foi que deram, em alguns momentos, a esperança para seguir. Pai, sua presença significou segurança e certeza de que não estou sozinho nessa caminhada.

## AGRADECIMENTOS

Primeiramente a Deus, por toda a força e esperança que me deu durante a minha vida.

À minha família: Joaquim Moura, Sueli Moura e Germano Moura. Por me acompanharem por toda a minha jornada e me ajudarem tanto nos momentos bons, quanto nos ruins.

Aos meus tios e padrinhos: William Bandeira e Sylvia Bandeira. Obrigado por toda a ajuda e incentivo durante minha vida acadêmica.

Ao meu orientador Prof. Dr. Márcio André Baima Amora por todo apoio e orientação durante essa fase da minha vida. Um excelente profissional, amigo, competente e disciplinado, que estou sempre admirando e o tenho como referência para a vida. Gostaria de agradecer de coração pelos conselhos e ensinamentos que proporcionam o meu crescimento profissional e pessoal.

Aos meus amigos: Jhenifer de Oliveira, Breno Meneses, Marcelo Estevão, Gean Dias, Flávio Carvalho, Werley Gonçalves, Ariel Pineda, Joan Kennedy, Bryan Nicholas, Kamila Gomes e Ruann Campos. O tempo é o bem mais valioso que existe na vida, obrigado a vocês por disponibilizarem parte dele para mim. Cada um de vocês representou uma boa conversa, amizades sólidas, noites de estudos, diversão e momentos de alegria.

Aos professores: Vanessa Teixeira, Vandilberto Pinto, Marcelo Marques, Iális Cavalcante e Reuber Régis. Obrigado pelos ensinamentos e conselhos. Graças a vocês, gradualmente, venho me tornando um melhor pesquisador e profissional.

Aos meus colegas do mestrado de Engenharia Elétrica e da Computação pelos ensinamentos e auxílios vivenciados durante o curso.

Aos meus colegas do grupo de Inteligência artificial. Por todos os trabalhos e artigos que criamos e criaremos.

À Instituição FUNCAP, pelo apoio financeiro com a manutenção da bolsa de auxílio.

Aos professores e funcionários da Universidade Federal do Ceará, em especial ao Campus de Sobral, por me proporcionarem uma educação de excelência.

“Vós que viveis e sempre atribuíis tudo o que ocorre na Terra aos movimentos celestes, como se tal movimento imprimisse em todas as coisas uma necessidade, se assim fosse, em vós seria destruído o livre arbítrio, e não seria justo que o homem tivesse por bem a alegria e por mal, a dor.” (Dante Alighieri, 1304 - 1321)

## RESUMO

Com o avanço no processo de automatização das indústrias, diversas metodologias para a identificação de falhas em máquinas elétricas rotativas estão sendo propostas. Um defeito nestes equipamentos geram perdas na produtividade, aumento nos custos e risco a acidentes por parte dos operadores. Desta forma, o presente trabalho propõe uma nova metodologia para realizar o diagnóstico tanto de falhas simples, como falhas compostas em máquinas elétricas rotativas. Para a realização dessa nova abordagem, é criado um classificador híbrido por votação ponderada que combina as previsões dos algoritmos: rede neural perceptron de multicamadas, máquina de vetores de suporte, *k*-vizinhos mais próximos, floresta aleatória, *gradient boosting* e o lightGBM. Além disso, para a busca dos pesos de cada classificador é empregado o uso da otimização por enxame de partículas relativísticas. O algoritmo é aplicado em três bancos de dados que possuem alguns dos principais tipos de falhas que ocorrem nestes dispositivos, entre eles, se destacam o desbalanceamento, desalinhamento, falhas nos rolamentos e curto-circuito em enrolamentos. As falhas mecânicas são identificadas neste trabalho de forma isolada e combinada. Os resultados encontrados demonstram a robustez da técnica proposta que conseguiu resultados de acurácia de 88,68% para o primeiro banco de dados, 100% para o segundo e 98,95% para o terceiro. Os resultados, quando comparados com outros trabalhos da literatura, se mostraram equivalentes ou melhores, expondo assim a sua aplicabilidade no diagnóstico de falhas simples e combinadas em máquinas elétricas rotativas.

**Palavras-chave:** Identificação de falha. Falhas combinadas. Classificador híbrido. Máquinas elétricas rotativas.



## ABSTRACT

With the advance in the process of automating industries, various methodologies for identifying faults in rotating electrical machines are being proposed. A fault in this equipment leads to losses in productivity, increased costs and the risk of accidents on the part of operators. This paper proposes a new methodology for diagnosing both simple and compound faults in rotating electrical machines. To implement this new approach, a weighted voting hybrid classifier is created that combines the predictions of the following algorithms: multilayer perceptron neural network, support vector machine,  $k$ -nearest neighbors, random forest, gradient boosting and lightGBM. In addition, relativistic particle swarm optimization is used to find the weights for each classifier. The algorithm is applied to three databases that contain some of the main types of faults that occur in these devices, including unbalance, misalignment, bearing faults and short-circuit windings. The mechanical faults identified in this work are isolated and combined. The results show the robustness of the proposed technique, which achieved accuracy results of 88.68% for the first database, 100% for the second and 98.95% for the third. The results, when compared with other studies in the literature, were fair.

**Keywords:** Fault identification. Combined failures. Hybrid classifier. Rotating electrical machines.

## LISTA DE FIGURAS

Figura 1 – Sinal de vibração no domínio da frequência . . . . .	35
Figura 2 – Exemplo do desbalanceamento . . . . .	40
Figura 3 – Exemplos de desalinhamento . . . . .	40
Figura 4 – Exemplo de rolamento . . . . .	41
Figura 5 – Exemplo de validação cruzada com cinco dobras . . . . .	44
Figura 6 – Exemplo de árvore de decisão . . . . .	45
Figura 7 – Exemplo de classificação de uma amostra usando o $k$ -NN para um $k = 3$ . . .	47
Figura 8 – Exemplo de classificação usando o SVM . . . . .	49
Figura 9 – Exemplo gráfico de um neurônio artificial . . . . .	51
Figura 10 – Exemplo da estrutura de uma MLP . . . . .	52
Figura 11 – Exemplo de estrutura em série . . . . .	55
Figura 12 – Exemplo de estrutura em paralelo . . . . .	55
Figura 13 – Exemplo de estrutura mista . . . . .	56
Figura 14 – Esquema do empilhamento . . . . .	61
Figura 15 – Bancada experimental do primeiro banco de dados . . . . .	73
Figura 16 – Bancada experimental do segundo banco de dados . . . . .	75
Figura 17 – Falhas criadas pelos autores do segundo banco de dados . . . . .	76
Figura 18 – Exemplo de sinal de vibração para cada classe . . . . .	77
Figura 19 – Bancada experimental do terceiro banco de dados . . . . .	78
Figura 20 – Falha experimental colocada na fase R do motor síncrono com diferentes potências: (a) 1,0 kW, (b) 1,5 kW e (c) 3,0 kW . . . . .	79
Figura 21 – Esquema das falhas geradas nos enrolamentos do estator . . . . .	79
Figura 22 – Exemplo do processo para definição dos hiperparâmetros dos classificadores formadores . . . . .	82
Figura 23 – Definição dos pesos para cada classificador formador . . . . .	83
Figura 24 – Teste do modelo . . . . .	84

## LISTA DE TABELAS

Tabela 1 – Predições dadas por cada um dos classificadores . . . . .	57
Tabela 2 – Probabilidades de classificação . . . . .	57
Tabela 3 – Análise por média . . . . .	57
Tabela 4 – Exemplo de contagem de borda . . . . .	58
Tabela 5 – Exemplo de uma matriz de confusão de uma classificação binária . . . . .	63
Tabela 6 – Especificações do motor de corrente contínua . . . . .	73
Tabela 7 – Especificações dos sensores empregados no primeiro banco de dados . . . . .	73
Tabela 8 – Número de amostras para cada condição de operação da máquina no primeiro banco de dados . . . . .	74
Tabela 9 – Número de amostras para cada condição de operação do segundo banco de dados . . . . .	75
Tabela 10 – Principais parâmetros dos motores síncronos de ímãs permanentes utilizados	78
Tabela 11 – Hiperparâmetros e espaço de busca para cada classificador . . . . .	81
Tabela 12 – Hiperparâmetros dos classificadores formadores para o primeiro banco de dados . . . . .	86
Tabela 13 – Valores dos pesos médios definidos por cada otimizador no primeiro banco de dados . . . . .	87
Tabela 14 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o primeiro banco de dados . . . . .	88
Tabela 15 – Resultados de acurácia final para cada classificador para o primeiro banco de dados . . . . .	89
Tabela 16 – Sensibilidade, precisão e <i>F-score</i> do classificador combinado por média ponderada com o uso do RPSO para o primeiro banco de dados . . . . .	89
Tabela 17 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o primeiro banco de dados . . . . .	90
Tabela 18 – Hiperparâmetros dos classificadores formadores para o segundo banco de dados	91
Tabela 19 – Valores dos pesos médios definidos por cada otimizador para o segundo banco de dados . . . . .	91
Tabela 20 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o segundo banco de dados . . . . .	92

Tabela 21 – Resultados de acurácia final para cada classificador para o segundo banco de dados . . . . .	93
Tabela 22 – Sensibilidade, precisão e <i>F-score</i> do classificador combinado por média ponderada com o uso do RPSO para o segundo banco de dados . . . . .	93
Tabela 23 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o segundo banco de dados . . . . .	94
Tabela 24 – Hiperparâmetros dos classificadores formadores para o terceiro banco de dados por análise de corrente . . . . .	95
Tabela 25 – Valores dos pesos médios definidos por cada otimizador para o terceiro banco de dados por análise de corrente . . . . .	95
Tabela 26 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o terceiro banco de dados por análise de corrente . . . . .	96
Tabela 27 – Resultados de acurácia final para cada classificador para o terceiro banco de dados por análise de corrente . . . . .	97
Tabela 28 – Sensibilidade, precisão e <i>F-score</i> do classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de corrente . . . . .	97
Tabela 29 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o terceiro banco de dados por análise de corrente . . . . .	97
Tabela 30 – Hiperparâmetros dos classificadores formadores para o terceiro banco de dados por análise de vibração . . . . .	98
Tabela 31 – Valores dos pesos médios definidos para o classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de vibração . . . . .	99
Tabela 32 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o terceiro banco de dados por análise de vibração . . . . .	99
Tabela 33 – Resultados de acurácia final para cada classificador para o terceiro banco de dados por análise de vibração . . . . .	99

Tabela 34 – Sensibilidade, precisão e <i>F-score</i> do classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de vibração . . . . .	100
Tabela 35 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o terceiro banco de dados por análise de vibração . . . . .	100
Tabela 36 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas combinadas em MER . . . . .	102
Tabela 37 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas focadas em rolamentos . . . . .	104
Tabela 38 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas elétricas em máquinas síncronas de ímã permanente . . . . .	104

## LISTA DE ABREVIATURAS E SIGLAS

<i>k</i> -NN	<i>k</i> -Nearest Neighbors
ABVT	<i>Alignment Balance Vibration Trainer</i>
AD	Árvores de decisão
AG	Algoritmo genético
ALO	<i>Ant Lion Optimization</i>
AUC	<i>Area Under the Curve</i>
BICO-CNN	<i>Bicoherence analysis - Convolutional Neural Network</i>
CART	<i>Classification and Regression Trees</i>
CBAM-DRN	<i>Deep Residual Network with Convolutional Block Attention Module</i>
CNN	<i>convolutional neural network</i>
COBYLA	<i>Constrained Optimization By Linear Approximation</i>
CSA	<i>Crow Search Algorithm</i>
CSC-CNN	<i>Cyclic Spectral Coherence - Convolutional Neural Network</i>
EELM	<i>Ensemble Extreme Learning Machine</i>
FA	Floresta Aleatória
FBG	<i>Fiber Bragg Grating</i>
FFT	<i>Fast Fourier Transform</i>
GAN	<i>Generative Adversarial Network</i>
GB	<i>Gradient Boosting</i>
GNNBFD	<i>Graph Neural Network-based Bearing Fault Detection</i>
GWO	<i>Grey Wolf Optimization</i>
IC	Inteligência Computacional
KNORA	<i>K-Nearest-Oracles</i>
KNORA-E	<i>K-Nearest Oracle-Eliminate</i>
KNORA-U	<i>K-Nearest Oracle-Union</i>
LBFGS	Limited-Broyden–Fletcher–Goldfarb–Shanno
LDV	<i>Laser Doppler Vibrometer</i>
LightGBM	<i>Light Gradient Boosting Machine</i>
MCSA	<i>Motor Current Signature Analysis</i>
MER	Máquinas Elétricas Rotativas

MLP	<i>Multilayer Perceptron</i>
OCDET	<i>Optimized Compensation Distance Evaluation Technique</i>
PSD	<i>Power Spectral Density</i>
PSO	<i>Particle Swarm Optimization</i>
RBF	<i>Radial Basis Function</i>
ReLU	<i>Rectified Linear Unit</i>
RMS	<i>Root Mean Square</i>
RNA	Redes Neurais Artificiais
RPSO	<i>Relativistic Particle Swarm Optimization</i>
SCA	<i>Sine–Cosine Algorithm</i>
SDP	<i>Symmetrized Dot Pattern</i>
SGD	<i>Stochastic Gradient Descent</i>
SK-CNN	<i>Spectral Kurtosis - Convolutional Neural Network</i>
STFT	<i>Short-Time Fourier Transforms</i>
SVM	<i>Support Vector Machines</i>
SWD	<i>Swarm Decomposition</i>
TanH	tangencial hiperbólica
VMD	<i>Variational Mode Decomposition</i>
WOA	<i>Whale Optimization Algorithm</i>
WPT	<i>Wavelet Packet Transform</i>
XGBoost	<i>Extreme Gradient Boosting</i>

## LISTA DE SÍMBOLOS

$N$	Número de amostras
$x_j$	Valor de cada amostra, onde $j = (1,2,\dots,N)$
$RMS$	Valor do RMS
$m$	Momento
$o$	Ordem do momento
$g_1$	Assimetria
$g_2$	Curtose
$\bar{x}$	Média amostral
$f_E$	Frequência da falha no domínio da frequência
$f_1$	Frequência fundamental em Hz
$h$	Número de harmônicos
$f_c$	Frequência característica da falha em Hz
$A_1, A_2, A_3$	Atributos
$V$	Verdadeiro
$F$	Falso
$X_1, X_2, X_3$	Valores
$p_m$	Ponto médio de separação
$S$	Conjunto
$S_1, S_2$	Subconjunto um e dois
$IG(S_1, S_2)$	Ganho de informação
$n_1, n_2$	Número de pontos no subconjunto um e dois respectivamente
$n$	Número total de pontos
$I(S)$	Função de impureza
$I_g(S)$	Índice gini
$p_i$	Probabilidade de uma classe $i$
$I_e(S)$	Entropia



$d_{ef1}$	Distância euclidiana
$d_{ef2}$	Distância <i>Manhattan</i>
$d_{ef3}$	Distância <i>Minkowski</i>
$x_{ek}$	Instâncias no ponto e onde $k = (1,2,\dots,a)$
$x_{fk}$	Instâncias no ponto onde $k = (1,2,\dots,a)$
$r$	Raio
$a$	Número de atributos
$d$	Dimensão
$\bar{W}$	Vetor d dimensional
$\bar{X}$	Vetor de instâncias
$b$	Valor de uma constante
$s$	Valor de uma instância
$\xi_s$	Folga para a instância $s$
$C$	Parâmetro de compensação
$\gamma$	Variável que estabelece a importância de cada exemplo para o treinamento
$RBF$	Valor do RBF
$X_t$	Conjunto de entradas, onde $t = (1,2,\dots,u)$
$P_t$	Pesos para cada conexão, onde $t = (1,2,\dots,g)$
$f_{tanh}(l)$	Função de ativação tangente hiperbólica
$f_{sig}(l)$	Função de ativação sigmoidal
$f_{relu}(l)$	Função de ativação linear retificada
$B$	<i>Bias</i>
$Y$	Saída
$g$	Neurônio genérico
$Z1, Z2$	Entradas da MLP
$Y1, Y2$	Saídas da MLP
$M1$	Modelo um
$M2$	Modelo dois

$M3$	Modelo três
$Mn$	Modelo n
$w_a, w_b, w_c$	Peso para o classificador A, B e C
$VP$	Verdadeiro positivo
$VN$	Verdadeiro negativo
$FP$	Falso positivo
$FN$	Falso negativo
$Acc$	Acurácia
$P$	Precisão
$Sens$	Sensibilidade
$Sol$	Solução
$F(x_t)$	Função, onde $t = (1, 2, \dots, n)$
$L(x)$	Polinômio linear
$D_1, \dots, D_n$	Direções linearmente dependentes
$p_0, p_1, \dots, p_n$	Pontos de busca em um espaço
$\lambda$	Constante empregada no método de Powell
$x_{\sigma d}$	Posição da partícula $\sigma$ .
$v_{\sigma d}$	Velocidade da partícula $\sigma$ .
$r_1, r_2$	Número aleatório entre 0 e 1
$w$	Ponderação de inércia
$c_1$	Parâmetro cognitivo
$c_2$	Parâmetro social
$p_{\sigma d}$	Melhor posição encontrada por uma partícula $\sigma$
$p_{gd}$	Melhor posição encontrada por todas as partículas até o momento
$v$	Velocidade
$M$	Massa
$\psi$	Fator de Lorentz
$c$	Velocidade da luz

$p$

Momento relativístico

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>21</b>
<b>1.1</b>	<b>Motivação</b>	<b>22</b>
<b>1.2</b>	<b>Desenvolvimento da pesquisa</b>	<b>23</b>
<b>1.3</b>	<b>Contribuição da dissertação</b>	<b>24</b>
<b>1.3.1</b>	<i>Trabalhos publicados</i>	<b>25</b>
<b>1.4</b>	<b>Estrutura do documento</b>	<b>26</b>
<b>2</b>	<b>REVISÃO DA LITERATURA</b>	<b>27</b>
<b>2.1</b>	<b>Identificação de falhas em máquinas elétricas rotativas</b>	<b>27</b>
<b>2.1.1</b>	<i>Detecção por modelagem matemática</i>	<b>27</b>
<b>2.1.2</b>	<i>Detecção por medições de grandezas da máquina</i>	<b>28</b>
<b>2.1.3</b>	<i>Detecção por inteligência computacional</i>	<b>29</b>
<b>2.2</b>	<b>Detecção de múltiplas falhas em máquinas elétricas rotativas</b>	<b>30</b>
<b>2.3</b>	<b>Emprego de modelos <i>ensemble</i> na classificação de falhas</b>	<b>31</b>
<b>2.4</b>	<b>Utilização de classificadores por votação com pesos</b>	<b>32</b>
<b>2.5</b>	<b>Considerações parciais do capítulo 2</b>	<b>33</b>
<b>3</b>	<b>FALHAS EM MÁQUINAS ELÉTRICAS ROTATIVAS</b>	<b>34</b>
<b>3.1</b>	<b>Máquinas elétricas rotativas</b>	<b>34</b>
<b>3.2</b>	<b>Análise da vibração</b>	<b>35</b>
<b>3.3</b>	<b>Análise de corrente</b>	<b>38</b>
<b>3.4</b>	<b>Principais tipos de falhas encontradas</b>	<b>39</b>
<b>3.4.1</b>	<i>Desbalanceamento</i>	<b>39</b>
<b>3.4.2</b>	<i>Desalinhamento</i>	<b>40</b>
<b>3.4.3</b>	<i>Falhas nos rolamentos</i>	<b>41</b>
<b>3.4.4</b>	<i>Combinação de falhas mecânicas</i>	<b>42</b>
<b>3.4.5</b>	<i>Curto-circuito nas bobinas e nas espiras do estator de uma máquina elétrica rotativa</i>	<b>42</b>
<b>3.5</b>	<b>Considerações parciais do capítulo 3</b>	<b>42</b>
<b>4</b>	<b>CLASSIFICAÇÃO DE DADOS</b>	<b>43</b>
<b>4.1</b>	<b>Formas de divisão do banco de dados</b>	<b>43</b>
<b>4.1.1</b>	<i>Hold-out</i>	<b>43</b>

4.1.2	<i>Validação cruzada por K-fold</i>	44
4.1.3	<i>Validação cruzada por Leave-one-out</i>	44
4.2	<b>Modelos de classificação</b>	45
4.2.1	<i>Árvore de decisão</i>	45
4.2.2	<i>K - vizinhos mais próximos</i>	46
4.2.3	<i>Máquina de vetores de suporte</i>	48
4.2.4	<i>Rede neural perceptron multicamadas</i>	50
4.3	<b>Considerações parciais do capítulo 4</b>	53
5	<b>COMBINAÇÃO DE MODELOS</b>	54
5.1	<b>Modelos combinados em série</b>	54
5.2	<b>Modelos combinados em paralelo</b>	55
5.3	<b>Modelos combinados de forma mista</b>	56
5.4	<b>Análise das predições</b>	56
5.4.1	<i>Voto Majoritário</i>	56
5.4.2	<i>Média</i>	57
5.4.3	<i>Predição dinâmica</i>	57
5.4.4	<i>Contagem de borda</i>	58
5.5	<b>Algoritmos de combinação de modelos</b>	59
5.5.1	<i>Bagging</i>	59
5.5.2	<i>Boosting</i>	59
5.5.3	<i>Modelos empilhados</i>	60
5.6	<b>Métricas para a avaliação do classificador</b>	62
5.6.1	<i>Matriz de confusão</i>	62
5.6.2	<i>Acurácia</i>	63
5.6.3	<i>Precisão</i>	63
5.6.4	<i>Sensibilidade</i>	63
5.6.5	<i>F-score</i>	64
5.7	<b>Considerações parciais do capítulo 5</b>	64
6	<b>OTIMIZAÇÃO</b>	65
6.1	<b>Otimizações básicas</b>	65
6.1.1	<i>Otimização exaustiva</i>	66
6.1.2	<i>Otimização randômica</i>	66

6.2	COBYLA . . . . .	66
6.3	Powell . . . . .	67
6.4	Otimização por meta-heurísticas . . . . .	68
6.4.1	<i>Otimização por enxame de partículas</i> . . . . .	68
6.4.2	<i>Otimização por enxame de partículas relativísticas</i> . . . . .	69
6.4.3	<i>Otimização por algoritmos genéticos</i> . . . . .	70
6.5	Considerações parciais do capítulo 6 . . . . .	71
7	METODOLOGIA . . . . .	72
7.1	Primeiro banco de dados . . . . .	72
7.2	Segundo banco de dados . . . . .	74
7.3	Terceiro banco de dados . . . . .	77
7.4	Computador e bibliotecas . . . . .	79
7.5	Pré-processamento dos bancos de dados . . . . .	80
7.6	Criação do modelo . . . . .	80
7.7	Considerações parciais do capítulo 7 . . . . .	84
8	RESULTADOS . . . . .	85
8.1	Resultados do primeiro banco de dados . . . . .	85
8.2	Resultados do segundo banco de dados . . . . .	90
8.3	Resultados do terceiro banco de dados . . . . .	94
8.3.1	<i>Diagnóstico por análise de corrente</i> . . . . .	94
8.3.2	<i>Diagnóstico por análise de vibração</i> . . . . .	98
8.4	Comparativo dos resultados com outros autores . . . . .	100
8.4.1	<i>Comparativo para o primeiro banco de dados</i> . . . . .	101
8.4.2	<i>Comparativo para o segundo banco de dados</i> . . . . .	102
8.4.3	<i>Comparativo para o terceiro banco de dados</i> . . . . .	103
8.5	Considerações parciais do capítulo 8 . . . . .	105
9	CONCLUSÕES E TRABALHOS FUTUROS . . . . .	107
9.1	Trabalhos futuros . . . . .	107
	REFERÊNCIAS . . . . .	109

## 1 INTRODUÇÃO

As Máquinas Elétricas Rotativas (MER) são essenciais para a indústria, isso porque vários produtos são criados em larga escala devido ao seu uso. São equipamentos robustos que muitas vezes necessitam de um alto investimento. Por serem tão importantes, é necessário ter um processo de manutenção eficiente (HAND, 2015).

A manutenção pode ser efetuada de quatro maneiras: corretiva, preventiva, preditiva e sistemática. A manutenção corretiva só ocorre quando o defeito já aconteceu. Esse tipo de manutenção possui diversas desvantagens, porque interrompe a produção, diminui a vida útil do equipamento e gera mais despesas (BONALDI, 2005). Já na manutenção preventiva são feitos agendamentos de revisões para evitar a falha ou a quebra da máquina. Nessas revisões são realizadas medições, lubrificações e quando necessário a troca de peças. Ela tem como principal característica ser realizada antes que o defeito atinja o equipamento (BONALDI, 2005).

Na manutenção preditiva é realizado um extenso monitoramento da máquina, no qual, são feitos testes e medições não invasivas com o equipamento geralmente em funcionamento. Nele, podem estar inclusos o uso de técnicas de inteligência computacional para a afeição do diagnóstico. Por fim, na manutenção sistemática os elementos de máquina do equipamento possuem uma data de validade para serem substituídas. Esse tipo de sistema efetua-se muito em aplicações da aeronáutica (BONALDI, 2005).

Muitas empresas pecam na manutenção, desta forma, com o tempo e a má utilização, as condições de funcionamento das máquinas vão diminuindo e graves falhas podem acabar acontecendo, acarretando em perdas na produção, diminuição na eficiência, risco aos operadores ou até mesmo a quebra do equipamento (HAND, 2015).

Devido a esses fatores, a comunidade acadêmica vem desenvolvendo diversos trabalhos cujo enfoque tem sido no diagnóstico das MER. Entre eles, três linhas de pesquisa se destacam, com o diagnóstico baseado: na modelagem matemática, em medições de grandezas da máquina e no uso da inteligência computacional (ALI *et al.*, 2019).

A primeira se utiliza da modelagem matemática para prever futuros problemas que surgirão no equipamento. Já a segunda, com o auxílio do analista, utiliza medições de corrente, tensão, velocidade e vibração para efetuar a análise. Por fim, tem-se o diagnóstico utilizando a inteligência computacional, no qual o modelo de aprendizado de máquina é criado para conduzir a identificação de falhas (ALI *et al.*, 2019).

Este último, com a utilização de algoritmos de classificação, está sendo bastante empregado no processo de automatização da indústria, no qual, sem a necessidade de um analista, um monitoramento preditivo já ocorre, alertando a equipe de manutenção sobre possíveis problemas na máquina e em seus elementos.

Em alguns casos, como em falhas combinadas, apenas um modelo de classificação talvez não seja o suficiente para efetuar um bom diagnóstico. As falhas combinadas apresentam dados que se aproximam das suas condições de operação geradoras, causando dúvidas ao modelo sobre qual classe a amostra pertence (MARTINS *et al.*, 2022). Isso ocorre porque cada algoritmo possui suas vantagens e desvantagens que afetam a divisão das regiões de classificação. A cada caso, em que se testa um modelo, observa-se o seu desempenho e dependendo do banco de dados e de seu pré-processamento podem ser obtidos resultados melhores ou piores na classificação.

Os algoritmos apresentam desvantagens e devem ser empregados em bancos de dados que se encaixam em suas características. Visando melhorar os resultados na classificação, sem perder as particularidades das diversas técnicas, foram propostos algoritmos de combinação de modelos. Estes conseguem melhorar o desempenho do diagnóstico e separar de uma forma mais precisa as regiões de cada classe.

## 1.1 Motivação

No estudo de detecção de falhas nas MER, ainda são poucos os trabalhos atualmente na literatura que abordam além de defeitos simples, defeitos compostos. As falhas compostas são mais difíceis de serem analisadas porque geralmente o classificador se confunde com as falhas mais simples (MARTINS *et al.*, 2021).

Além disso, a maioria das metodologias utilizadas estão recorrendo ao *deep learning* para conseguir bons resultados, o que não é tão acessível a todos, já que ela precisa de muito processamento e tempo para ser treinado e testado adequadamente. Outro método bastante aplicado é a criação de amostras sintéticas, essa que por um lado é capaz de se apresentar como uma solução viável, pode acarretar em erros de generalização nos modelos (SANTOS *et al.*, 2018).

Por este motivo, o presente trabalho visa realizar a detecção de falhas simples e combinadas em máquinas rotativas, buscando conseguir bons resultados, sem a necessidade de aplicar técnicas de *deep learning* e criar dados sintéticos.

Também é desenvolvido uma nova técnica de aprendizado de máquina híbrida que



junta um classificador por votação média ponderada com o algoritmo de otimização por enxame de partículas relativísticas para o ajuste de pesos, capaz de gerar ótimos resultados na classificação de dados e selecionar o peso que cada classificador terá no diagnóstico. A mistura de modelos tradicionais com técnicas de otimização pode diminuir os erros, a falta de generalização e melhorar os resultados na classificação de dados desbalanceados.

Os algoritmos são desenvolvidos usando a linguagem de programação Python em sua versão 3,8. A escolha dessa linguagem se deve ao fato do seu grande acervo de bibliotecas e documentações para o emprego do aprendizado de máquina.

O algoritmo de otimização por enxame de partículas relativísticas foi desenvolvido recentemente por Roder *et al.* (2020) e ainda foi muito pouco explorado em trabalhos acadêmicos. Além disso, por empregar o uso de vários fenômenos relativísticos como a distorção espaço temporal, a inércia e o efeito da massa no enxame, o algoritmo proposto se mostrou alcançando bons resultados nos trabalhos de Roder *et al.* (2020) e (RODER *et al.*, 2021).

Como relatado no artigo de Roder *et al.* (2020), com o aumento na complexidade da otimização, o algoritmo conseguiu não se prender a resultados mínimos locais. Isso se torna relevante na combinação de classificadores, pois pode possibilitar que com a combinação dos diversos modelos e o uso da metodologia descrita em Roder *et al.* (2020) que seja obtido uma solução global melhor que a soluções indicadas pelos classificadores individuais que em muitas situações podem está relacionadas aos mínimos locais.

## 1.2 Desenvolvimento da pesquisa

Como já mencionado, essa pesquisa tem como objetivo a criação de um classificador combinado (*ensemble*) que seja capaz de auxiliar no diagnóstico de falhas simples e compostas das MER. Para a criação dessa metodologia, diversas etapas foram seguidas durante o desenvolvimento da pesquisa.

Inicialmente foi elaborado uma revisão sistemática sobre metodologias e bancos de dados empregadas na identificação de falhas nas MER. Os trabalhos foram selecionados seguindo alguns critérios como a questão da pesquisa, métodos de busca das fontes, tipos e idiomas dos trabalhos, estratégia de extração de informação e sumarização dos resultados.

Nesta busca optou-se em focar em trabalhos atuais, dos quais, apresentassem novidades relevantes sobre o tema. Realizando essa análise observou-se que as principais formas de identificação de falhas nas MER são por meio da análise de vibração e corrente.

Além disso, observou-se a dificuldade no diagnóstico de falhas combinadas nesses componentes. Devido a isso, foram pesquisadas novas metodologias que empregavam o aprendizado de máquina na identificação destas falhas. Infelizmente muitas das metodologias exploradas atualmente apresentam limitações, nas quais, estão associadas ao elevado tempo de processamento e custo computacional.

Desta forma, o foco da pesquisa foi a de empregar técnicas mais simples, que conseguissem resultados melhores ou equivalentes a outros autores. Com isso, o propósito da pesquisa passou a ser o uso de técnicas *ensemble*.

Depois da aplicação de vários testes, observou-se uma melhora nos resultados de acurácia de classificação empregando o uso de classificadores por votação ponderada. O emprego desse método de classificação *ensemble* acabou gerando uma dificuldade na definição dos pesos. Depois de alguns testes e estudos de outras metodologias, recorreu-se ao emprego de algoritmos de otimização. Entre eles, o uso da otimização por enxame de partículas relativísticas alcançou bons resultados, o que levou ao emprego desta técnica.

Para garantir a consistência da metodologia criada, foram selecionados três bancos de dados de falhas simples e compostas em MER. Além disso, emprega-se o uso da validação cruzada, como forma de evidenciar os resultados encontrados neste trabalho.

### 1.3 Contribuição da dissertação

As principais contribuições desenvolvidas neste trabalho são apresentadas a seguir:

1. Desenvolvimento de uma metodologia capaz de extrair características dos dados de corrente e vibração de forma simples e rápida sem a necessidade de transformar os dados para o domínio da frequência.
2. Criação de um novo classificador combinado através de votação por média ponderada composto por várias técnicas de aprendizado de máquina capaz de melhorar os resultados de acurácia na detecção de falhas simples e combinadas em MER.
3. Emprego de uma nova técnica (otimização por enxame de partículas relativísticas) para a busca pelos melhores pesos para o classificador com votação por média ponderada.

Como contribuições secundárias podem ser citados:

1. Uma revisão da literatura que aborda diversas metodologias empregadas no diagnóstico das MER, presente no capítulo 2.
2. Um comparativo entre a metodologia criada e outros classificadores *ensemble*.

3. Um comparativo entre medições de corrente e vibração para diagnóstico de falhas de curto-circuito em bobinas e espiras em uma máquina síncrona de ímã permanente.
4. Emprego de bancos de dados recentes que refletem falhas simples e compostas em três tipos de MER.

### ***1.3.1 Trabalhos publicados***

Os seguintes trabalhos apresentados em congressos e eventos foram desenvolvidos no período da dissertação, sendo relacionados ao problema abordado ou ao uso de inteligência computacional aplicada.

1. **MOURA FILHO, J. O. F.**; SILVA, M. E. ; PARENTE, J. K. C. ; AMORA, M. A. B. ; PINTO, V. P. Diagnóstico de falhas em máquinas elétricas rotativas utilizando técnicas de ensemble learning. In: XV Simpósio Brasileiro de Automação Inteligente, 2021, Online. São Paulo: Sociedade Brasileira de Automática, 2022.
2. **MOURA FILHO, J. O. F.**; SILVA, M. E. ; PINTO, V. P. ; AMORA, M. A. B. Modelagem da Degradação em Turbinas de Aeronave sob Condições Reais de Voo. In: XXIV Congresso Brasileiro de Automática, 2022, Fortaleza. Anais do XXIV Congresso Brasileiro de Automática. São Paulo: Sociedade Brasileira de Automática, 2022.
3. SILVA, M. E. ; **MOURA FILHO, J. O. F.** ; SUNI, J. C. P. ; AMORA, M. A. B. Classificação Inteligente de Falhas Multiestágio em Alimentadores Primários de Distribuição de Energia Elétrica. In: XXIV Congresso Brasileiro de Automática, 2022, Fortaleza. Anais do XXIV Congresso Brasileiro de Automática. São Paulo: Sociedade Brasileira de Automática, 2022.
4. SILVA, M. E. ; **MOURA FILHO, J. O. F.**; PARENTE, J. K. C. ; AMORA, M. A. B. **Aplicação de Random Forest na classificação de condições de balanceamento de um motor de indução trifásico.** XXIX Encontro de Iniciação Científica, 2021, Sobral. Revista Encontros Universitários da UFC. Fortaleza: Universidade Federal do Ceará, 2021. v. 5. p. 494-494.
5. **MOURA FILHO, J. O. F.** ; SILVA, M. E. ; PARENTE, J. K. C. ; PINTO, V. P. ; AMORA, M. A. B. **Extração de regras de decisão analisáveis para o diagnóstico de falhas mecânicas em máquinas elétricas rotativas.** In: XIV Encontro de Pesquisa de Pós-Graduação, 2021, Sobral. Revista Encontros Universitários da UFC. Fortaleza: Universidade Federal do Ceará, 2021. v. 6. p. 516-516.

6. SILVA, M. E.; MOURA FILHO, J. O. F. ; PARENTE, J. K. C. ; AMORA, M. A. B. **Previsão de Velocidade do Vento em Parque Eólico a partir de uma Rede Neural Recorrente Autoregressiva Não-Linear com Entradas Exógenas (NARX)**. In: XL Encontro de Iniciação Científica, 2021, Sobral. Revista Encontros Universitários da UFC. Fortaleza: Universidade Federal do Ceará, 2021. v. 6. p. 623-623.

Por fim, esta dissertação representa a continuidade do trabalho de conclusão de curso realizado com o tema de diagnóstico de falha em máquinas elétricas rotativas, cujo o foco foi no diagnóstico de falhas simples como o desalinhamento, desbalanceamento e defeitos nos rolamentos, usando o banco de dados público MAFAULDA (MAFAULDA, 2017):

MOURA FILHO, J. O. F. **Diagnóstico de falhas em máquinas elétricas rotativas utilizando técnicas de reconhecimento de padrões** 2021. 82 f. Graduação em Engenharia Elétrica – Universidade Federal do Ceará, Sobral, 2021.

#### 1.4 Estrutura do documento

Este trabalho está organizado ao todo em nove capítulos. O primeiro capítulo, como já apresentado, mostra uma introdução ao tema de diagnóstico de falhas nas MER. Além disso, é comentado sobre a motivação de elaboração desta dissertação, sobre os objetivos alcançados, desenvolvimento da pesquisa, contribuições e os trabalhos já publicados.

No capítulo 2 é relatado o estado da arte, exemplificando como outros trabalhos da literatura abordam o tema e quais técnicas estão sendo mais utilizadas atualmente. Já no capítulo 3 é realizada uma fundamentação sobre as falhas nas MER, sendo mostrado os principais tipos de falhas, como elas ocorrem e suas formas de detecção.

Os conceitos sobre classificação de dados são apresentados no capítulo 4. Nele são explicados algumas das principais técnicas de reconhecimento de padrões que também foram utilizadas neste trabalho. Já no capítulo 5 são mostradas as formas de combinação destas técnicas e maneiras de avaliá-las. Técnicas de otimização são explicitadas no capítulo 6.

No capítulo 7 são ilustrados os bancos de dados utilizados e a metodologia empregada para a criação do novo algoritmo. Os resultados, análises e comparações com outros trabalhos são evidenciados no capítulo 8. Por fim, no capítulo 9 é apresentado as conclusões e os trabalhos futuros.

## 2 REVISÃO DA LITERATURA

Neste capítulo são apresentados os procedimentos empregados por outros autores para a identificação de falhas em MER. Além disso, é feita uma revisão sobre técnicas combinadas de classificadores utilizados atualmente.

Na seção 2.1 é discorrido sobre os métodos aplicados na detecção de falhas, tendo enfoque em trabalhos que abordam a modelagem matemática, medições de grandeza e uso da Inteligência Computacional (IC). Trabalhos que evidenciam a classificação de falhas combinadas são apresentados na seção 2.2. Já na seção 2.3, são relatados artigos, dos quais, se utilizam de técnicas combinadas de modelos para realizar o reconhecimento das falhas. Na seção 2.4 são abordados técnicas de votação ponderada aplicado por outros autores. Por último, na seção 2.5 são expostas às considerações sobre o capítulo.

### 2.1 Identificação de falhas em máquinas elétricas rotativas

Como já mencionado, a literatura aborda esse tema por meio de três linhas de pesquisa: detecção de falha por modelos matemáticos, pela medição de grandezas da máquina e pela IC. Cada um deles possui suas vantagens e desvantagens, o que torna, relativo, qual maneira seria mais adequada de resolver o problema.

#### 2.1.1 Detecção por modelagem matemática

A detecção de falhas utilizando modelagem matemática se torna importante por facilitar as pesquisas, diminuindo o número de experimentos empregados na simulação de diferentes falhas. Isso porque com o modelo matemático da máquina pode-se simular amostras de falhas. Além disso, é possível prever defeitos observando apenas a mudança de parâmetros das equações dos dispositivos.

No trabalho de Espinoza-Sepulveda e Sinha (2021) é elaborado uma rede neural perceptron multicamadas para realizar o diagnóstico de falhas, onde são usados dados de vibração reais para a sua criação. Para validar o modelo, é criado uma modelagem matemática usando o método dos elementos finitos para simular falhas mecânicas relacionadas a máquina como: desalinhamento, frouxidão no pedestal e fricção do rotor. O autor conseguiu 100% de acurácia na identificação das condições da máquina.

Já no trabalho de Djagarov *et al.* (2021) é criado um modelo matemático universal

de uma máquina de indução, no qual, é possível simular todos os tipos de falhas. Com isso os autores conseguiram identificar falhas como o curto-circuito entre as espiras, circuito aberto nos enrolamentos do rotor, problemas de isolamento e danos nos rolamentos.

É abordado por Palavicino e Sarlioglu (2020) um modelo para máquinas síncronas de relutância e de ímã permanente, capaz de obter as formas de onda do torque e da corrente, visando prever futuras falhas no dispositivo. Os autores ainda comparam a metodologia empregada com a análise dos elementos infinitos.

No artigo de Yassa e Rachek (2020) é criado um modelo matemático que descreve as condições de funcionamento de uma máquina síncrona de ímã permanente. Dessa forma, pode-se identificar o curto-circuito no enrolamento do estator.

### **2.1.2 Detecção por medições de grandezas da máquina**

Uma das formas mais simples e mais empregadas na indústria atualmente. A detecção de falhas pela medição de grandezas é feita de forma preventiva e precoce, evitando paradas não programadas e custos extras com manutenção. O analista realiza uma série de mensurações e posteriormente examina os resultados para encontrar ou não a incidência de uma falha.

No trabalho de Jeong *et al.* (2022) é proposto um novo indicador para a detecção de falhas de curto-circuito em máquinas síncronas de ímã permanente. Para realizar este feito, é coletado dados de magnitude e fase da corrente, para que seja posteriormente feita uma análise da corrente de sequência positiva e da corrente de sequência negativa. Os resultados mostraram uma grande eficiência no método proposto.

É desenvolvida por Irhoumah *et al.* (2019) uma nova metodologia estática para a detecção de falha de curto-circuito em máquinas de indução e em máquinas síncronas. O método é não invasivo e utiliza sensores de fluxo localizados no eixo da máquina. Os sensores medem o campo magnético externo e na proximidade do equipamento. Posteriormente é feito um cálculo do coeficiente de Pearson nos dois sinais, o que permite identificar possíveis problemas.

No artigo de Sonnaillon *et al.* (2010) é realizado um método de monitoramento de temperatura sem a utilização de sensores aplicado em motores de indução. Ele é baseado na variação da resistência e da temperatura dos enrolamentos do estator. Para a aplicação deste método, um pequeno sinal de corrente alternada é aplicado no motor com o objetivo de medir a resistência do estator.

É relatado no artigo de Mohammed e Djurović (2021) um método de extração

simultânea de dados mecânicos e térmicos das condições de funcionamento do rolamento. O diferencial deste trabalho está no uso de sensores *Fiber Bragg Grating* (FBG) para conduzir as medições. Os resultados experimentais utilizando o sensor mostraram a captação da falha e a possibilidade de detecção desta.

### **2.1.3 Detecção por inteligência computacional**

Todas as técnicas mostradas anteriormente podem ser utilizadas com a IC. A sua utilização nestes procedimentos vem se desenvolvendo a cada ano com a automatização das indústrias. Sua principal característica e vantagem, vem do monitoramento preditivo, surgindo dessa forma sistemas automatizados e com o funcionamento interligado com a operação da máquina.

A vertente que mais se destaca na IC para o diagnóstico das MER é a aprendizagem de máquina para a classificação de falhas. Entre as técnicas de aprendizagem de máquina mais utilizadas atualmente estão a aprendizagem profunda (*deep learning*) e a combinação dessas técnicas através da aprendizagem em conjunto (*ensemble learning*). No *ensemble learning*, classificadores são combinados visando conseguir melhores resultados na classificação.

O modelo de *deep learning* como relatado por Li *et al.* (2022) é uma técnica robusta mas que necessita, normalmente, de muito tempo para o treinamento e otimização. Com isso diversos autores estão buscando formas de reduzir este problema.

É proposto por Li *et al.* (2022) uma combinação entre uma rede neural profunda binarizada com o emprego da técnica floresta aleatória. Primeiro é aplicado a rede neural profunda visando fazer a extração de características das amostras de falhas. Posteriormente se aplica o modelo floresta aleatória para efetuar a classificação. Empregando esses métodos, os autores conseguiram uma redução de tempo na criação do modelo e um aumento nos resultados de acurácia em comparação com outros classificadores testados.

É desenvolvido por Habbouche *et al.* (2022) uma metodologia na qual se aplica o *Variational Mode Decomposition* (VMD) para realizar a extração de características. O VMD é uma ferramenta de processamento de sinais na qual decompõe o sinal de entrada em diferentes funções. Para a classificação de falhas é empregada uma rede convolucional unidimensional. Empregando o método no diagnóstico de falhas em rolamentos, o autor conseguiu uma acurácia de 99,22%.

No artigo de Tang *et al.* (2022) é apontado uma nova forma de realizar a extração de

características de dados de vibração. Para isso o autor emprega o uso da entropia de dispersão multivariada multiescala melhorada combinada com a entropia hierárquica. Para a classificação dos dados utiliza-se o *Support Vector Machines* (SVM). Os autores empregaram um banco de dados de falhas em rolamentos, conseguindo resultados acima de 99% de acurácia.

Já no artigo de Saberi *et al.* (2022) é aplicado o *Light Gradient Boosting Machine* (LightGBM), mais uma técnica de seleção de atributos, para classificar falhas em rolamentos. Utilizando essa metodologia, o autor conseguiu uma acurácia de 98,55%.

Como os dados de falhas geralmente são escassos e pode ocorrer o desequilíbrio entre o número de amostras de cada classe. É proposto no artigo de Miao *et al.* (2022) a criação de dados sintéticos utilizando a rede neural generativa adversarial, do inglês, *Generative Adversarial Network* (GAN). Como resultados, as amostras geradas pela rede se mostraram de qualidade e melhoraram o desempenho do modelo na classificação de dados.

## **2.2 Detecção de múltiplas falhas em máquinas elétricas rotativas**

Muitos trabalhos na literatura abordam apenas as falhas simples, mas normalmente em ambientes industriais é comum se ter mais de uma falha avariando a máquina. Isso acontece porque geralmente algum problema mecânico acaba afetando outros elementos de máquina do sistema, trazendo assim um ambiente multi-falhas (ZHANG *et al.*, 2021).

No trabalho de Zhang *et al.* (2021) é sugerido a aplicação de análise de sensibilidade, para a criação de um novo método que utiliza máquina de aprendizagem extrema, mais a teoria da evidência para diagnosticar falhas em conjunto com bancos de dados desequilibrados. As falhas foram desenvolvidas, sendo experimentalmente compostas por defeitos em rolamentos e engrenagens.

No artigo de Huo *et al.* (2022) é proposto um sistema de coleta com multissensores, possuindo uma fusão de dados entre imagens térmicas e medições de vibrações. Junto a isso é utilizado o *deep learning*. O objetivo do autor é classificar um sistema de falhas compostas no qual os dados apresentem muitos ruídos. O método apresentou um alto desempenho na classificação dos dados.

É proposto por Li *et al.* (2021) uma rede adversária de transferência de dois estágios para a detecção de múltiplas falhas mecânicas em máquinas. O algoritmo é capaz de rotular novas amostras e separá-las de dados já conhecidos. Experimentos empregando um conjunto de dados de máquinas validam os resultados.



No trabalho de Chen *et al.* (2020), um conjunto de dados é coletado com vários tipos de falhas no rotor, no rolamento e uma combinação destas. Ao final do processo se gerou 48 condições de funcionamento da máquina. Para a classificação dos dados é empregado o uso combinado de duas redes neurais convolucionais. O autor conseguiu uma acurácia de 95,93%.

No artigo de Dineva *et al.* (2019) é utilizado uma abordagem de classificação multi-rotulo para a detecção de falhas múltiplas apresentando sinais ruidosos. O autor utilizou técnicas mais simples como a Árvores de decisão (AD) e o *k-Nearest Neighbors* (*k*-NN). Os resultados encontrados foram em torno de 80% utilizando como métrica de avaliação o *F-score*.

### 2.3 Emprego de modelos *ensemble* na classificação de falhas

Técnicas de *ensemble learning* estão sendo muito empregados na literatura, isso se deve, a sua capacidade de gerar bons resultados em bancos de dados diversos. Modelos *ensemble* como a floresta aleatória, *gradient boosting*, *stacking* e classificadores dinâmicos se destacam pela sua precisão e por sua capacidade diversificada de gerar resultados.

No artigo de Inyang *et al.* (2023) é proposto um *stacking* de classificadores profundos no diagnóstico de falhas simples e compostas em máquinas. O *stacking* é um modelo, no qual, as predições dos classificadores que o compõem são empilhados, formando assim um novo banco de dados, que posteriormente, é utilizado como entrada para um novo classificador. Para o processamento do sinal foram usados a curtose espectral, a coerência espectral cíclica e a análise de biocoerência. Os autores conseguiram ao final do experimento uma acurácia de 98,9%.

É feito por Tian *et al.* (2017) um processo de votação entre classificadores mais simples para diagnosticar falhas em rolamentos. A classe que receber mais votos é escolhida para rotular a amostra. Os classificadores utilizados são o *k*-NN, AD e o SVM. Os resultados encontrados se mostraram superiores aos classificadores individuais.

No trabalho de Wang *et al.* (2020) é apresentado um novo classificador *ensemble* chamado de *Ensemble Extreme Learning Machine* (EELM), empregado para a classificação de falhas compostas em máquinas rotativas. O EELM consistem em duas redes de aprendizado extremo, a primeira é utilizada para realizar o agrupamento e a segunda para a classificação de vários rótulos. Os resultados experimentais ilustram que o modelo criado apresenta os melhores resultados gerais.

No artigo de Pacheco *et al.* (2022) é proposto um método para a criação de classificadores de falhas mecânicas baseados em redes convolucionais que podem ser empregados em

diferentes tipos de máquinas rotativas e condições de trabalho. O diagnóstico proposto consegue identificar falhas em engrenagens e em partes dos rolamentos. É empregado o uso da otimização bayesiana para encontrar os melhores hiperparâmetros dos modelos. Metodologias *ensemble* como o *stacking* e votação de classificadores pela média são empregados. O classificador criado, é testado e comparado com outras técnicas como a floresta aleatória, XGboost e Resnet50.

## 2.4 Utilização de classificadores por votação com pesos

Uma técnica bastante popular de *ensemble learning* é o processo de votação de classificadores. Para isso, emprega-se o uso das predições ou probabilidades que cada classificador obteve para um determinada amostra e por meio de votação é escolhido o rótulo para ela. Essa metodologia será discutida melhor no capítulo 5.

Uma das formas de realizar esse processo é empregando pesos aos classificadores no momento da votação. Atualmente esta forma de votação está sendo amplamente trabalhada, mas um problema constante entre os pesquisadores é definir o peso para cada classificador.

No trabalho de Manconi *et al.* (2022) é desenvolvido um processo de votação por pesos para o diagnóstico de COVID-19 utilizando um exame de tórax gerado por tomografia computadorizada. Os autores empregaram o uso de dois modelos: o Inception-V1 e Inception-V3. A votação por peso foi construída separadamente por meio de um *ensemble* de cada algoritmo. Os resultados mostram que a combinação obteve resultados melhores que os algoritmos individuais.

No artigo de Shen *et al.* (2022) é feito uma votação por classe, na qual os pesos são definidos empregando a taxa de detecção e a taxa de falso positivo, em combinação com o algoritmo de otimização *Chaos Bat*. Os resultados encontrados foram altos e mostraram a eficiência do método na detecção de intrusos em ataques cibernéticos.

O trabalho de El-Kenawy *et al.* (2020) apresenta como desafio a classificação de características do COVID-19. Para isso, é utilizado um classificador por votação ponderada, no qual os pesos são selecionados utilizando o algoritmo de otimização *Guided Whale Optimization Algorithm* que é baseado na otimização por enxame de partículas. As técnicas mais simples empregadas para compor a detecção foram as redes neurais artificiais, SVM, *k*-NN e AD. A metodologia proposta alcançou uma área sob a curva, do inglês, *Area Under the Curve* (AUC) de 0,995, mostrando assim a eficiência do modelo.

São combinados por Koohestani *et al.* (2019) o *k*-NN, SVM e Naive Bayes para criar classificadores *ensemble*. Esses modelos são desenvolvidos para fazer o monitoramento

preditivo do desempenho de motoristas na direção de veículos. No método de votação, os pesos para cada classificador são escolhidos empregando o otimizador da formiga-leão, do inglês, *Ant Lion Optimization* (ALO), o algoritmo de otimização de baleia, do inglês, *Whale Optimization Algorithm* (WOA), a otimização de enxame de partículas e o otimizador do lobo cinzento, do inglês, *Grey Wolf Optimization* (GWO). O melhor resultado encontrado pelo autor foi de 97,50% usando a votação e com os pesos sendo otimizados pelo método do lobo cinzento.

No artigo de Moldovan (2020) é proposto um método de votação ponderada empregando o algoritmo de pesquisa do corvo, do inglês *Crow Search Algorithm* (CSA), para a detecção de epilepsia. São usados três algoritmos mais simples para a criação do *ensemble*: SVM, AD e  $k$ -NN. O autor conseguiu 83% de acurácia.

## 2.5 Considerações parciais do capítulo 2

O capítulo apontou diversos trabalhos que realizam o diagnóstico de falhas simples e combinadas em MER. Entre eles, foram expostas às linhas de pesquisas aplicadas neste tema (modelagem matemática, medições de grandezas e o uso de IC), metodologias e algoritmos para a identificação inteligente de falhas. Além disso, foram apresentadas algumas técnicas de combinação de classificadores utilizadas atualmente.

A literatura vem buscando formas de realizar a coleta de dados, extração de características e o reconhecimento do padrão de falha. Com o uso da inteligência computacional, esse processo ficou cada vez mais automatizado e preditivo. No próximo capítulo serão abordados noções sobre as MER, formas de identificação de falhas e alguns dos principais defeitos que ocorrem nesses dispositivos.

### 3 FALHAS EM MÁQUINAS ELÉTRICAS ROTATIVAS

Este capítulo apresenta conceitos sobre o estudo de falhas em MER. Os métodos geralmente mais utilizados para a verificação das condições de funcionamento destes equipamentos são a análise da vibração e a análise da assinatura da corrente. É necessário salientar que existem outras técnicas como o monitoramento de ruído, temperatura e óleo.

Na seção 3.1 é elaborado uma introdução sobre MER. Já na seção 3.2 é apresentado conceitos da análise de vibração, exemplificando sua metodologia e os equipamentos utilizados para o seu desenvolvimento. É discutido na seção 3.3 a análise da corrente, mostrando também seus métodos e os dispositivos mais empregados na sua concepção. Na seção 3.4 são apresentados os principais tipos de falhas que ocorrem nas MER e como é possível identificá-las. Por fim, na seção 3.5 é feito uma análise dos conteúdos abordados neste capítulo.

#### 3.1 Máquinas elétricas rotativas

A principal função das MER é a de conversão de energia, que pode ser realizada através da transformação de energia elétrica para a mecânica, no caso do motor, ou o contrário, no caso do gerador. Elas podem ser classificadas quanto a sua alimentação, sendo divididas em máquinas de corrente contínua e máquinas de corrente alternada. As MER são comumente formadas por duas partes: rotor (componente girante) e estator (componente estacionário) (KOSOW, 2005; UMANS, 2014).

As máquinas de corrente alternada ainda podem ser divididas em outras duas subclasses, a saber, máquinas síncronas e assíncronas (indução). Em uma máquina de indução a frequência de rotação na parte girante difere da frequência da rede em regime permanente. Já na máquina síncrona essa frequência é igual ao da rede (UMANS, 2014).

Na indústria, as MER são geralmente empregadas na função de motor e acopladas a outros componentes e cargas. Exemplos destes dispositivos são centrifugas, bombas de água, ventiladores e esteiras. Eles são formados por elementos de máquinas como correntes, polias, acopladores, caixa de engrenagens e rolamentos.

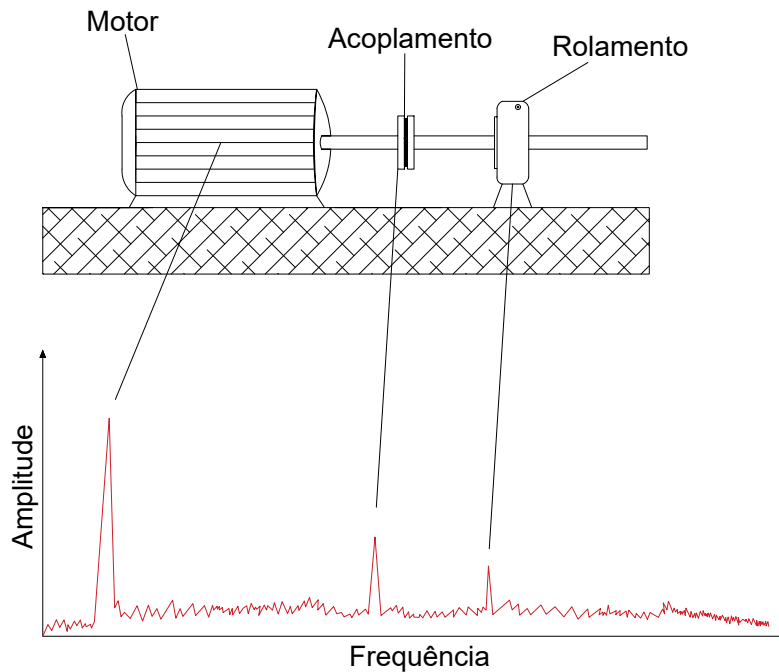
Como já mencionado anteriormente, com o tempo e com a má instalação e utilização, as MER podem apresentar falhas em seus componentes e nas cargas, que são danosas a produção. Por isso se torna necessário a identificação destes problemas (HAND, 2015). Para isso pode-se empregar o uso da análise de vibração e de corrente. Essas duas técnicas quando empregadas de

maneira correta são capazes de representar os padrões de falhas.

### 3.2 Análise da vibração

O funcionamento das MER gera vibrações, deste modo, falhas podem ser identificadas quando há uma mudança no padrão destes sinais. Devido a isso, é possível realizar o diagnóstico deste equipamento. Para a elaboração dessa análise, sensores ou analisadores são colocados em pontos da máquina e de seus elementos. É mostrado na Figura 1 um exemplo de sinal de vibração no domínio da frequência, onde, o motor e seus componentes apresentam harmônicos característicos.

Figura 1 – Sinal de vibração no domínio da frequência



Fonte: Elaborado pelo autor (2023).

Os principais tipos de sensores empregados são transdutores de velocidade, sensores de deslocamento, acelerômetros e o vibrômetro laser doppler, também conhecido pelo seu termo em inglês *Laser Doppler Vibrometer* (LDV) (GHAZALI; RAHIMAN, 2021). Os transdutores de velocidade são dispositivos que funcionam com o princípio de indução eletromagnética, possuem uma alta sensibilidade, sendo sensíveis principalmente a vibrações laterais (GOYAL; PABLA, 2016).

Os transdutores de deslocamento são dispositivos capazes de realizar medições de

vibrações em baixas frequências. São aplicados na detecção de falhas de desbalanceamento e de desalinhamento (GHAZALI; RAHIMAN, 2021). Os acelerômetros são transdutores que utilizam o efeito piezoelétrico, no qual, existe uma pressão gerada pela vibração que atinge um material piezoelétrico capaz de produzir cargas elétricas. Eles são muito utilizados nas medições de vibrações laterais (GOYAL; PABLA, 2016).

Para terminar, existe o LDV, o qual, é um equipamento capaz de medir a velocidade absoluta de um corpo que está vibrando. Este processo ocorre porque o LDV consegue medir o deslocamento do laser até chegar no seu alvo, ou seja, corpo no qual se quer medir a vibração (GOYAL; PABLA, 2016).

Com o sinal captado, é feito um processo de extração de características, no qual, o analista busca encontrar dados que se distanciam do padrão. Para isso, utilizam-se técnicas, nas quais, é possível transformar o valor de um sinal em apenas uma amostra ou em um conjunto delas. Esse processo pode ser realizado no domínio do tempo, no domínio da frequência ou nos dois (GHAZALI; RAHIMAN, 2021).

O domínio do tempo é uma das formas mais simples de se realizar o diagnóstico por análise de vibração, devido ao fato, que não é necessário transformar os dados para outro domínio. Uma das principais formas de fazer a extração de características neste domínio é utilizando o *Root Mean Square* (RMS). Empregando este método são obtidos conteúdos energéticos do sinal de vibração (GHAZALI; RAHIMAN, 2021). O RMS pode ser calculado como mostrado na Equação 1 adaptada do trabalho de Rocha (2018):

$$RMS = \sqrt{\left(\frac{1}{N} \sum_{j=1}^N x_j^2\right)} \quad (1)$$

onde  $j = (1, 2, \dots, N)$ , em que  $N$  é número de amostras e  $x_j$  é o valor de cada amostra.

Além do RMS, outras técnicas são aplicadas no sinal, entre elas, pode-se mencionar a assimetria e a curtose. A assimetria mede a falta de simetria de uma distribuição. Já a curtose tem como propósito descobrir o grau de achatamento de uma distribuição (KOKOSKA; ZWILLINGER, 2000; GHAZALI; RAHIMAN, 2021).

Existem diversas formas de se calcular a assimetria, uma delas emprega o conceito de momento apresentado na Equação 2 adaptado do trabalho de Kokoska e Zwillinger (2000):

$$m_o = \frac{1}{N} \sum_{j=1}^N (x_j - \bar{x})^o \quad (2)$$

onde  $m$  representa o momento,  $o$  é a ordem do momento,  $N$  é o número de amostras e  $\bar{x}$  é a média amostral. Já a assimetria, representada por  $g_1$ , é apresentado na Equação 3 retirado do trabalho de Kokoska e Zwillinger (2000):

$$g_1 = \frac{m_3}{m_2^{3/2}} \quad (3)$$

Por fim, a curtose, apresentada como  $g_2$ , pode ser representada como mostrado na Equação 4 retirado do trabalho de Kokoska e Zwillinger (2000):

$$g_2 = \frac{m_4}{m_2^2} \quad (4)$$

Já no domínio da frequência é realizado uma transformação dos dados do domínio do tempo para o domínio da frequência. Essa conversão é geralmente realizada aplicando a transformada rápida de Fourier, do inglês, *Fast Fourier Transform* (FFT). Após ser utilizada, a FFT, é observado se há ou não a presença de outras frequências, além da frequência natural. Depois da conversão também são empregados métodos que podem ser estatísticos, como média, variância e o desvio padrão, para realizar a extração de características.

Outra técnica também utilizada no domínio da frequência é a análise de envelope, do inglês, *envelope analysis*. Ela é muito empregada na detecção de falhas nos rolamentos, isso ocorre por sua capacidade de separar o ruído dos sinais de baixa frequência (GHAZALI; RAHIMAN, 2021; LEE *et al.*, 2021).

A última forma na qual a análise de vibração pode ser realizada é utilizando uma mescla entre o domínio do tempo e o da frequência. Nesta forma, tanto as características variantes no tempo, quanto as componentes de frequência conseguem ser detectadas (GHAZALI; RAHIMAN, 2021). Entre as principais técnicas utilizadas nesta forma estão a transformada de Wavelet e a transformada de Hilbert.

Com as amostras coletadas e extraídas é possível detectar falhas mecânicas que estejam comprometendo o funcionamento das MER e de seus componentes. Essa detecção pode ser feita por um especialista ou por algum modelo de aprendizado de máquina.

As principais desvantagens da análise de vibração são a necessidade de acesso ao motor ou aos componentes dele, dificuldade na identificação de falhas de origem elétrica e em alguns casos, a necessidade de se trabalhar com vários sinais para se conseguir um bom

diagnóstico. Também, se houver a presença do inversor, pode ocorrer a diminuição na qualidade no espectro da frequência do sinal (BONALDI, 2005).

### 3.3 Análise de corrente

Falhas elétricas ou até mecânicas podem influenciar nos valores das correntes nas MER. Por este motivo, outra técnica muito empregado na manutenção preditiva é a análise de corrente. Essa análise, como no caso anterior, pode ser realizada no domínio do tempo e da frequência. Quando é feita na frequência pode receber o nome de análise do espectro da corrente.

A análise do espectro da corrente, também conhecida pelo seu termo em inglês, *Motor Current Signature Analysis* (MCSA) é uma técnica muito aplicada para diagnosticar as condições de funcionamento da máquina de indução. A técnica se consiste em captar os dados de corrente de uma fase do equipamento, efetuar uma amostragem e posteriormente um estudo no domínio da frequência (BONALDI, 2005; SILVA, 2015).

Os componentes de falha são representados no espectro da frequência como mostra a Equação 5 adaptado do trabalho de Silva (2015).

$$f_E = f_1 \pm h * f_c \quad (5)$$

onde  $f_e$  é a frequência da falha no domínio da frequência,  $f_1$  é a frequência fundamental,  $h$  é o número de harmônicos e  $f_c$  é a frequência característica da falha (SILVA, 2015).

Um mau funcionamento em componentes da máquina faz com que haja uma modulação no entreferro produzindo-se assim frequências de harmônicos na corrente de alimentação, desde modo, o emprego do MCSA serve para detectar essas distorções (JUNG *et al.*, 2006). O MCSA consegue detectar falhas nos enrolamentos do estator, problemas no rotor e defeitos nos rolamentos (CORNE *et al.*, 2015).

Para realizar as medições de corrente na máquina são utilizados dispositivos como condicionadores de sinais, analisadores de qualidade de energia e sensores de corrente. Depois que os dados são coletados, como na análise de vibração, diversas metodologias podem ser aplicadas para converter os dados para o domínio da frequência. Entre esses processos tem-se a FFT.

Após as amostras serem representadas no domínio da frequência, pode-se realizar o processo de extração de características. Nele, como mencionado anteriormente, é feito uma



redução do sinal, pegando diversas particularidades que podem ser empregadas para se detectar padrões.

As desvantagens dessa técnica são uma maior dificuldade de diagnosticar falhas mecânicas e problemas gerados na qualidade do espectro da corrente quando existe a presença de inversores de frequência (BONALDI, 2005).

### **3.4 Principais tipos de falhas encontradas**

As falhas nas MER podem ser de origem mecânica ou elétrica. Quando são de origem mecânica ocorrem geralmente pelo tempo de uso, defeitos nas peças e má utilização do equipamento. Já as falhas elétricas se originam de complicações na alimentação (desbalanceamento de fases), problemas com a carga (sobrecarga), transientes de tensão (picos de tensão ao ligar o circuito ou uma carga) e curto-circuito nas bobinas (estator ou rotor) (HAND, 2015).

Como mencionado anteriormente, as análises de vibração e de corrente são capazes de identificar falhas, sendo uma técnica complementar a outra. A análise de corrente possui uma disposição melhor na identificação de falhas elétricas, já a análise de vibração detecta melhor falhas de origens mecânicas (BONALDI, 2005). Neste trabalho serão analisadas as falhas mecânicas e a ocorrência de curto-circuito nas bobinas do estator da máquina, que representa umas das faltas elétricas mais importantes, a seguir; essas faltas serão comentadas com mais detalhes.

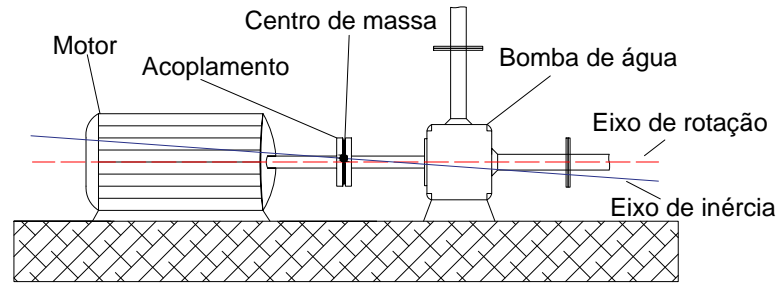
#### **3.4.1 Desbalanceamento**

O desbalanceamento acontece quando existe um desequilíbrio entre o eixo de rotação da máquina e seu centro de massa. Este problema está relacionado a distribuição indevida de carga na máquina, que acaba atrapalhando na rotação do equipamento (LÓPEZ, 2014).

Essa falha provoca vibrações e pode gerar defeitos em outros elementos da máquina. É um problema comum, mas se não for tratada pode prejudicar ou até mesmo parar o funcionamento do equipamento.

Entre as suas principais causas estão a falta de balanceamento na carga, imperfeição na fabricação das peças, problemas na montagem da máquina, sujeiras acumuladas e falta de manutenção (LÓPEZ, 2014). É apresentado na Figura 2 um exemplo de desbalanceamento.

Figura 2 – Exemplo do desbalanceamento

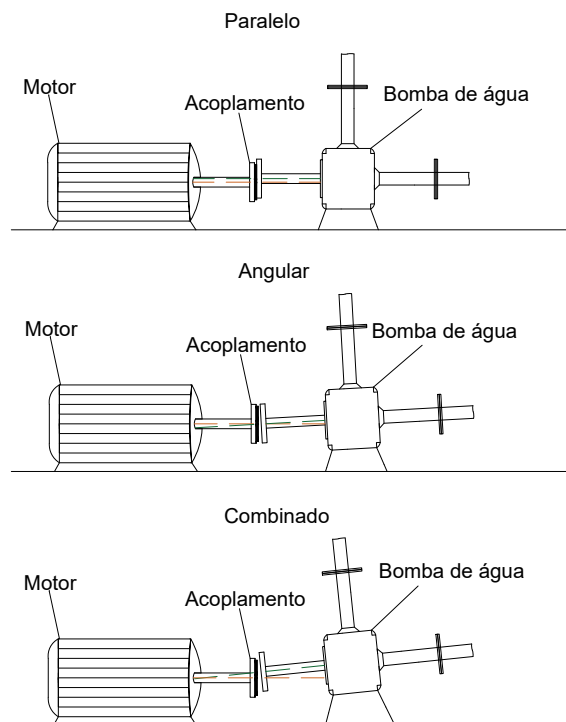


Fonte: Elaborado pelo autor (2023).

### 3.4.2 Desalinhamento

Os acoplamentos são elementos de máquinas capazes de realizar a transmissão mecânica em componentes rotativos. Quando o acoplamento da máquina e da carga não está alinhado, tem-se uma falha de desalinhamento. O desalinhamento, também, é um dos problemas mais corriqueiros na indústria, podendo gerar graves defeitos ou até mesmo a quebra do equipamento em determinadas circunstâncias (ROCHA, 2018; LÓPEZ, 2014). Na Figura 3 são apresentados exemplos de desalinhamento.

Figura 3 – Exemplos de desalinhamento



Fonte: Elaborado pelo autor (2023).

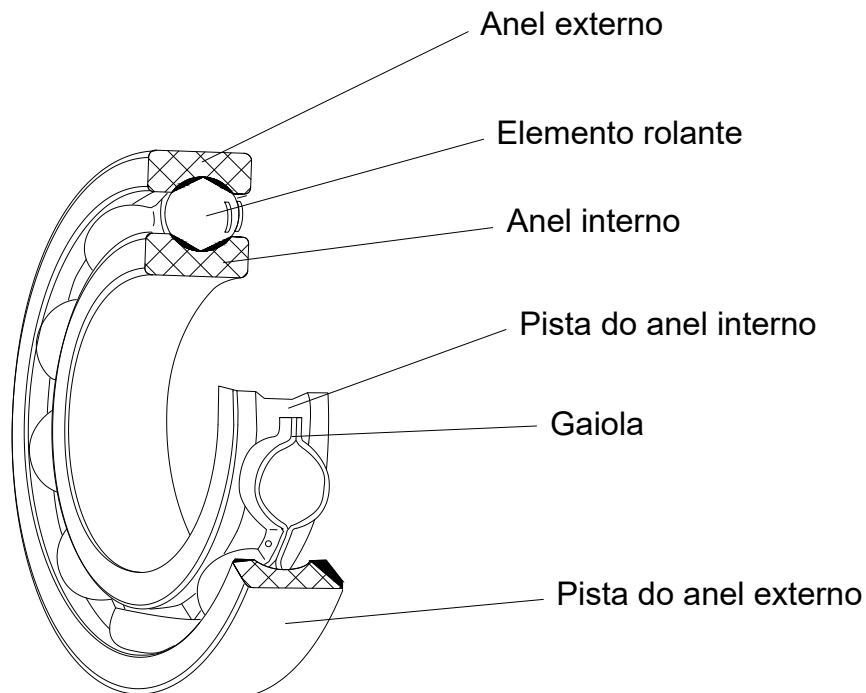
O desalinhamento possui como consequências o aumento das vibrações em todo o sistema, perda de eficiência e a diminuição da vida útil. Ele decorre da falta de manutenção, má instalação do equipamento, mau uso e da falta de nivelamento das superfícies da máquina e em torno dela (LÓPEZ, 2014). O desalinhamento pode ocorrer de três maneiras: paralelo, angular e combinado.

### 3.4.3 Falhas nos rolamentos

Os rolamentos são elementos de máquinas capazes de reduzir o atrito de componentes rolantes e ficam geralmente localizados no eixo rotacional da máquina. Os defeitos nesses componentes acontecem pelo tempo de utilização, falta de manutenção, sujeiras e pela má utilização ou montagem do equipamento (LÓPEZ, 2014).

As falhas, como trincas, machucados ou algum tipo de contaminação, estão localizadas na pista do anel externo e na pista do anel interno, na gaiola ou no elemento rolante. Defeitos nos rolamentos causam vibrações, sendo assim possível a detecção da falha. Na Figura 4 é exibido um exemplo de rolamento.

Figura 4 – Exemplo de rolamento



Fonte: Elaborado pelo autor (2023).

#### **3.4.4 Combinação de falhas mecânicas**

Em alguns casos, a máquina pode apresentar mais de uma falha durante o seu funcionamento, isso acontece porque os defeitos apresentados anteriormente acabam gerando desgastes e problemas em outros elementos da máquina. Uma das combinações de falhas possíveis é a do desalinhamento com o desbalanceamento (MARTINS *et al.*, 2022).

#### **3.4.5 Curto-circuito nas bobinas e nas espiras do estator de uma máquina elétrica rotativa**

O curto-circuito é um problema comum em máquinas elétricas e pode ser ocasionado por picos de alta tensão, enrolamentos sobreaquecidos, perda de isolamento, condutores de bobinas soltos, cortados ou contaminados. Uma avaria deste tipo pode levar ao rebobinamento ou troca da máquina (HAND, 2015).

Quando algumas espiras de uma fase se encontram em curto, um circuito fechado é formado entre elas, fazendo com que se circule uma alta corrente. A energia consumida por essa corrente aumenta a corrente da fase possibilitando a sua identificação (HAND, 2015). A corrente em circulação é capaz de derreter o circuito formado, fazendo uma abertura nele. Este problema também é de fácil identificação visual devido à carbonização que ocorre nos enrolamentos (HAND, 2015).

### **3.5 Considerações parciais do capítulo 3**

Neste capítulo foram apresentados alguns conceitos relevantes sobre falhas nas MER. O objetivo foi a de apresentar conceitos e técnicas para a detecção de defeitos nestes equipamentos. Também foi exposto os principais tipos de falhas estudadas e como estas podem ser detectadas.

A análise de vibração e de corrente são as principais formas de se identificar defeitos nestes equipamentos, isso porque, quando existe algum problema na máquina, um aumento de vibração e mudanças no valor de corrente se mostram como indicativos para alguma falha.

Uma análise destes dois fatores pode ser realizado tanto no domínio do tempo, quanto no domínio da frequência. Cada um destes possui vantagens e desvantagens que são necessárias serem analisadas antes de começar o processo. No próximo capítulo são apresentados conceitos sobre a classificação de dados, como as formas de dividir o banco de dados e alguns dos algoritmos mais empregados.

## 4 CLASSIFICAÇÃO DE DADOS

Na classificação são criados modelos capazes de predizer rótulos de elementos pertencentes a um banco de dados. Para isso o algoritmo "aprende" os padrões e dessa forma pode conseguir classificar corretamente uma amostra.

Os modelos de classificação, no processo de sua criação, podem sofrer de dois erros, a saber, representação e de generalização. No erro de representação, o modelo é simples demais para representar de forma correta os dados. Já no erro de generalização o modelo é sobreajustado e pode absorver ruídos dos dados, apresentando posteriormente problemas na generalização. Isso quer dizer que o algoritmo quando apresentado a dados novos (não empregados em seu treinamento) poderá exprimir muitos erros em suas predições (CASTRO; FERRARI, 2016).

Neste capítulo é apresentado uma introdução a classificação de dados, no qual, na seção 4.1 são mostradas as forma de divisão do banco de dados, na seção 4.2 são conceituados alguns algoritmos de classificação e na seção 4.3 é feito um resumo e consideração final do capítulo.

### 4.1 Formas de divisão do banco de dados

Para a criação de um algoritmo de classificação de dados, deve-se preocupar com o processo de avaliação de desempenho destes elementos. Para isso, pode-se dividir o banco de dados, usando parte dele para criar o modelo e a outra parte para teste. Existem diversas formas de fazer essa divisão, entre elas pode-se destacar o: *Hold-out*, validação cruzada por *K-fold* e a validação cruzada por *Leave-one-out*.

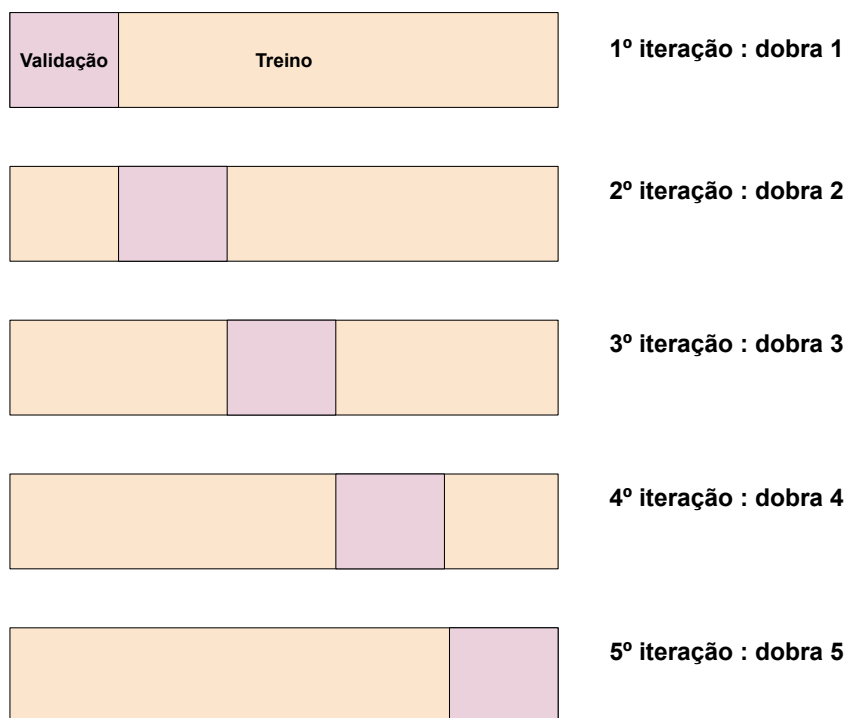
#### 4.1.1 *Hold-out*

O *Hold-out* é a forma mais simples de divisão. Nela dividimos o banco de dados em duas partes: treinamento e teste. Os modelos são criados empregando as amostras de treinamento, e testados usando o teste. O *hold-out* também pode apresentar mais uma divisão, do qual, denomina-se de validação. Ela é empregada para o ajuste dos parâmetros dos modelos (CASTRO; FERRARI, 2016; FACELI *et al.*, 2011).

### 4.1.2 Validação cruzada por K-fold

Nesta forma de divisão, além da divisão de duas partes em treinamento e teste, também é feita uma separação nos dados de treinamento em  $k$  dobras do inglês *folds*. Nela, a cada interação é empregado  $k - 1$  partes para o treinamento e 1 para validação. O número de iterações também é  $k$  e a cada interação uma das partes é usada como validação (CASTRO; FERRARI, 2016; FACELI *et al.*, 2011). Na Figura 5 é mostrado um exemplo de validação cruzada com cinco dobras.

Figura 5 – Exemplo de validação cruzada com cinco dobras



Fonte: Elaborado pelo autor (2023).

### 4.1.3 Validação cruzada por Leave-one-out

Por fim, na validação cruzada *leave-one-out*, no qual possui um funcionamento parecido com a de *K-fold* só que com um valor de  $k$  igual ao número de elementos do banco de dados de treinamento. Desta forma, a cada interação apenas uma amostra é usada para validação. Esse processo é muito empregado em bancos de dados que possuem poucos elementos (FACELI *et al.*, 2011).

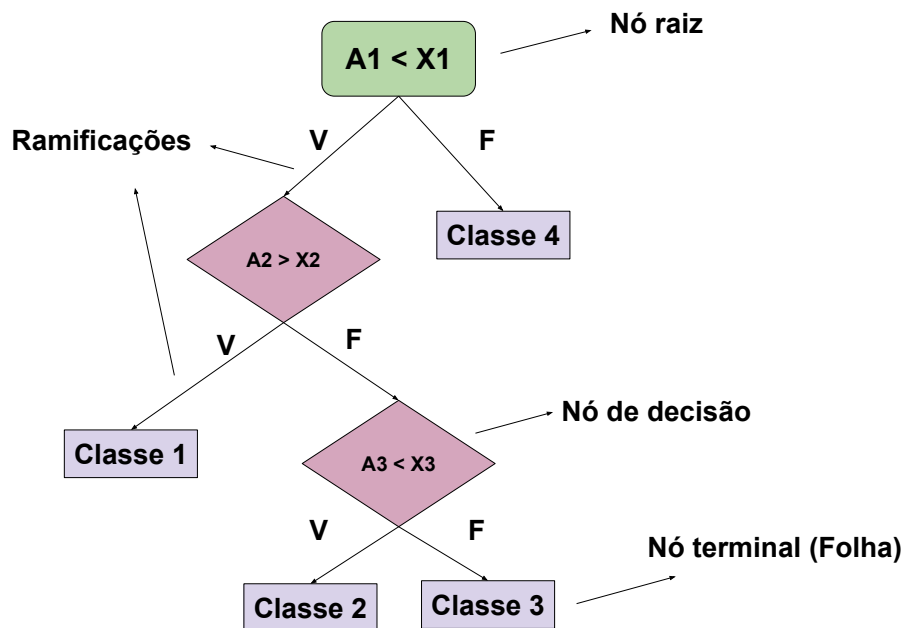
## 4.2 Modelos de classificação

Existem diversos modelos capazes de reconhecer padrões e dessa forma classificá-los. Entre eles, se destacam a AD, o  $k$ -NN, SVM e as Redes Neurais Artificiais (RNA). Além disso, também é bastante comum realizar a combinação desses modelos visando aumentar os resultados gerais.

### 4.2.1 Árvore de decisão

A AD é uma técnica de aprendizado de máquina que particiona os dados em segmentos de escolhas de "se" e "então". O seu modelo cria um grafo com um formato parecido a de uma árvore, no qual, possibilita a interpretação das regras desenvolvidas por ele. No modelo, a primeira regra é denominado de nó raiz, já as outros, são os nós de decisão. Eles são interligados por ramificações, das quais, podem culminar em um nó terminal (folha), onde são atribuídos os rótulos das classes para cada conjunto de escolhas (CASTRO; FERRARI, 2016; FACELI *et al.*, 2011). Na Figura 6 é apresentado um exemplo de uma AD, onde  $A1$ ,  $A2$  e  $A3$  representam atributos, já o  $V$  exprime verdadeiro, o  $F$  é falso e o  $X1$ ,  $X2$  e  $X3$  são valores.

Figura 6 – Exemplo de árvore de decisão



Fonte: Elaborado pelo autor (2023).

Existem várias formas de se criar uma AD, entre elas, pode-se citar os algoritmos ID3, CART, C4.5 e o C5.0. Em especial, será abordado o uso do CART, já que foi empregado

na criação dos modelos deste trabalho. O algoritmo de classificação e regressão de árvores do inglês *Classification and Regression Trees* (CART), foi desenvolvido por Breiman *et al.* (1984) e propõe um método de criação, no qual, divide o banco de dados em partições para produzir o maior ganho de informação nos nós (KRZYWINSKI; ALTMAN, 2017; FACELI *et al.*, 2011).

Para a geração do CART, é primeiramente feito uma divisão homogênea das amostras em dois subconjuntos. O ponto médio dessa separação é denominado de  $p_m$ , sendo selecionado visando maximizar o ganho de informação (KRZYWINSKI; ALTMAN, 2017). Na Equação 6 é apresentado a formulação do ganho de informação retirado do trabalho de Krzywinski e Altman (2017).

$$IG(S_1, S_2) = I(S) - n_1 \frac{I(S_1)}{n} - n_2 \frac{I(S_2)}{n} \quad (6)$$

onde o conjunto  $S$  é dividido em dois subconjuntos  $S_1$  e  $S_2$ , nos quais, possuem  $n_1$  e  $n_2$  pontos respectivamente. Já  $I(S)$  é denominada de função de impureza e tem como propósito medir a mistura de uma classe no subconjunto e  $n$  é o número total de pontos (KRZYWINSKI; ALTMAN, 2017). O índice gini é um exemplo de função de impureza e sua formulação é mostrada na Equação 7.

$$I_g(S) = \sum p_i(1 - p_i) \quad (7)$$

onde  $p_i$  é a probabilidade de uma classe  $i$  no conjunto  $S$ . Outra medida de impureza é a entropia e ela pode ser observada na Equação 8.

$$I_e(S) = - \sum p_i \log_2(p_i) \quad (8)$$

Os processos de separação continuam, sempre buscando maximizar os ganho de informação a cada separação (nó). Ao final do processo, o modelo gerou diversas regras de decisão que culminam nos rótulos das classes.

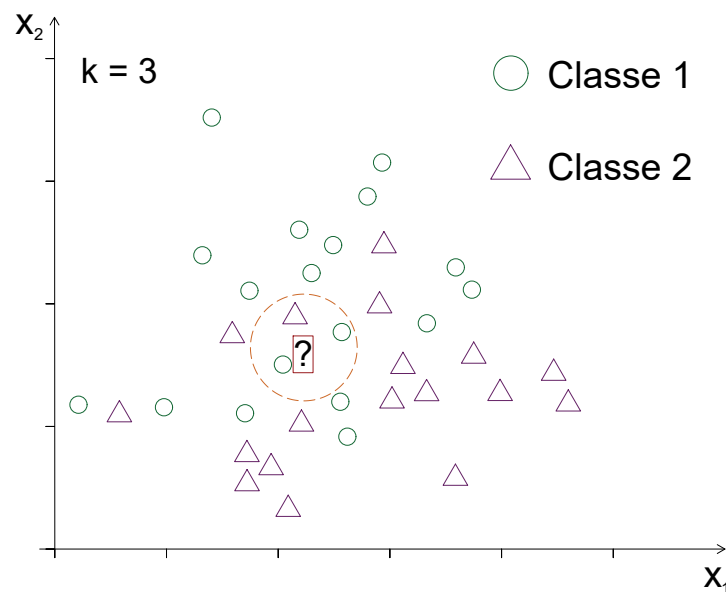
#### 4.2.2 *K - vizinhos mais próximos*

Este classificador é um dos mais simples encontrados na literatura. A predição deste algoritmo é feita analisando os elementos mais próximos da amostra que se quer rotular. Ele



observa quais são as classes dos  $k$  elementos mais próximos e por um processo de votação, ou seja, a classe mais predominante é escolhida como o rótulo da amostra de teste (CASTRO; FERRARI, 2016; GRUS, 2021). Na Figura 7 é mostrado um exemplo de classificação de uma amostra usando o  $k$ -NN para um  $k = 3$ .

Figura 7 – Exemplo de classificação de uma amostra usando o  $k$ -NN para um  $k = 3$



Fonte: Elaborado pelo autor (2023).

Como pode ser observado na Figura 7, para um  $k = 3$ , a amostra é rotulada como pertencente a classe 1, isso ocorre porque entre os três elementos mais próximos, dois pertencem à classe 1.

Em casos de empate pode-se escolher aleatoriamente um vencedor, diminuir o valor de  $k$  até encontrar um vencedor ou ponderar os votos em relação à distância de cada elemento. É um algoritmo com um custo computacional baixo, consegue bons resultados de acurácia, é muito fácil de ser criado e possui poucos parâmetros de ajuste (GRUS, 2021; CASTRO; FERRARI, 2016).

A principal distância empregada em sua construção é a euclidiana, mas existem outras que se destacam e conseguem bons desempenhos. Entre elas pode-se salientar a distância de *Manhattan* e a de *Minkowski*. Nas Equações 9, 10 e 11, adaptadas do livro de Castro e Ferrari (2016), são mostrados as formulações das distâncias euclidiana, *Manhattan* e *Minkowski* respectivamente:

$$d_{ef1} = \sqrt{\sum_{k=1}^a (x_{ek} + x_{fk})^2} \quad (9)$$

$$d_{ef2} = \sum_{k=1}^a |x_{ek} + x_{fk}| \quad (10)$$

$$d_{ef3} = \left( \sum_{k=1}^a (x_{ek} + x_{fk})^r \right)^{\frac{1}{r}}, (r \geq 1) \quad (11)$$

onde  $k = (1, 2, \dots, a)$ ,  $a$  é o número de atributos,  $x_{ek}$  e  $x_{fk}$  são as instâncias e  $r$  é o raio.

### 4.2.3 Máquina de vetores de suporte

O SVM busca pelo melhor hiperplano que separe as regiões de cada classe em um problema de classificação. Achar esse hiperplano é um problema de otimização e nem sempre existe um hiperplano adequado para dividir a região. Por isso, em alguns casos um ajuste de dimensões utilizando a função de kernel pode ser adequado na busca da melhor forma de separar os dados (GRUS, 2021).

Esse algoritmo é geralmente empregado em classificação binária, na qual, existe apenas duas classes. Um exemplo de hiperplano para um problema linear  $d$  dimensional pode ser definido como apresentado na Equação 12 retirada do trabalho de (AGGARWAL, 2015):

$$\bar{W} * \bar{X} + b = 0 \quad (12)$$

onde  $\bar{W}$  é um vetor  $d$  dimensional constituído de coeficientes do hiperplano, já  $\bar{X}$  é o vetor de instâncias e  $b$  é uma constante. Já os dois vetores de suporte simétricos podem ser dados por  $\bar{W} * \bar{X} + b = 1$  e  $\bar{W} * \bar{X} + b = -1$ . Com isso, na criação do modelo, o objetivo é encontrar o  $\bar{W}$  e o  $b$  que maximize a margem de separação entre os hiperplanos em um banco de dados de treinamento (AGGARWAL, 2015). A distância entre os dois hiperplanos é dado por  $2/||\bar{W}||$ . Maximizar essa função é a mesma coisa que minimizar  $||\bar{W}||^2/2$ . Para a resolução do problema, para que cada elemento pertença a uma classe, temos então as seguintes inequações mostradas nas Equações 13 e 14 adaptado do livro de (AGGARWAL, 2015):

$$\bar{W} * \bar{X} + b \geq +1 \quad (13)$$

$$\bar{W} * \bar{X} + b \leq -1 \quad (14)$$

Em casos em que os dados não são linearmente separáveis, pode-se utilizar uma margem flexível, no qual, uma folga  $\xi$  para as instâncias é introduzido, transformando as Equações 13 e 14, nas Equações 15 e 16 adaptado do livro de (AGGARWAL, 2015):

$$\bar{W} * \bar{X} + b \geq +1 - \xi \quad (15)$$

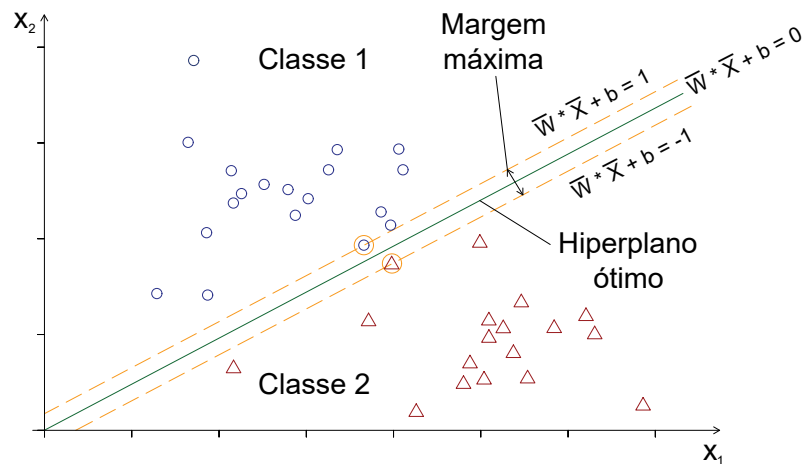
$$\bar{W} * \bar{X} + b \leq -1 + \xi \quad (16)$$

Também é necessário colocar restrições de não negatividade nas funções de folga, com isso podemos expressar a Equação 17 adaptada do trabalho de (AGGARWAL, 2015):

$$\frac{\|\bar{W}\|^2}{2} + C * \sum_{s=1}^K \xi_s \quad (17)$$

onde  $C$  é um parâmetro de compensação para a classificação incorreta de amostras do banco de dados de treino e  $s = (1, 2, \dots, K)$ . Na Figura 8 é representado um exemplo de classificação binária usando o SVM.

Figura 8 – Exemplo de classificação usando o SVM



Os procedimentos explanados até agora tratam do SVM em um problema linear, mas a maioria dos problemas de classificação não seguem esse padrão. Para resolver esse problema pode-se mapear os dados para outra dimensão, uma maior por exemplo, na qual exista um hiperplano capaz de separar os dados (GRUS, 2021). O mapeamento é feito utilizando uma função de "kernel" que geralmente pode ser um polinômio, uma *sigmoid* ou uma função de base radial do inglês *Radial Basis Function* (RBF) (CHANG; LIN, 2007; FAN *et al.*, 2008).

A RBF é um dos *kernels* mais empregados, nela existem um parâmetro que deve ser corretamente selecionado para um bom desempenho do algoritmo, a saber, o  $\gamma$ . O  $\gamma$  estabelece a importância de cada exemplo para o treinamento (CHANG; LIN, 2007; FAN *et al.*, 2008). Na Equação 18 é mostrado a formulação do *kernel* RBF, adaptado do trabalho de Chang e Lin (2007):

$$RBF = \exp(\gamma \|v - v'\|^2) \quad (18)$$

sendo  $v$  e  $v'$  vetores de atributos de algum conjunto de dados.

Como também já foi mencionado, o SVM é geralmente utilizado em classificação binária, mas ele também é viável em problemas que possuem vários rótulos. Para isso, pode-se empregar o uso de uma metodologia chamada de um contra um. É através da Equação 19 adaptada do trabalho de Chang e Lin (2007) que se sabe o número de classificadores que serão utilizados para o desenvolvimento dessa metodologia:

$$n_{SVM} = \frac{n^\circ \text{ de classes} * (n^\circ \text{ de classes} - 1)}{2} \quad (19)$$

onde  $n_{SVM}$  representa o número de SVMs que são criados e  $n^\circ \text{ de classes}$  representa o número de classes. Nesta metodologia cada SVM fica responsável por uma classificação binária. O final do processo resulta em várias divisões que separam os elementos de cada classe (CHANG; LIN, 2007; FAN *et al.*, 2008).

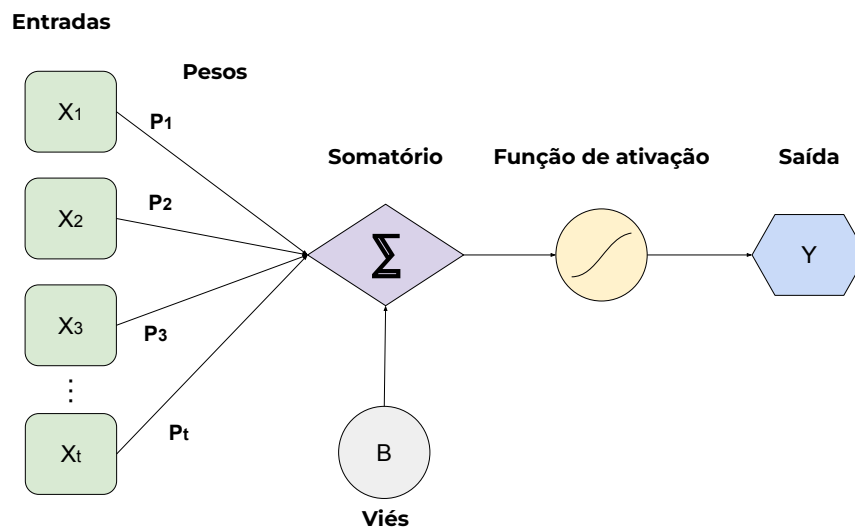
#### **4.2.4 Rede neural perceptron multicamadas**

As RNA são inspiradas pelo funcionamento do nosso sistema nervoso, no qual, é capaz de processar e armazenar informações. Este modelo preditivo é formado por uma cadeia de neurônios artificiais interligados. Cada um deles é uma pequena unidade de processamento que

verifica as entradas, executa cálculos e gera posteriormente resultados. Diferentemente da árvore de decisão, na qual, é explícita as regras de decisão para a sua criação, as RNA funcionam em um sistema de "caixa – preta", onde não é possível identificar as formas nas quais o problema foi resolvido (CASTRO; FERRARI, 2016; GRUS, 2021).

O neurônio básico é uma pequena unidade de processamento composta por um conjunto de entradas  $X_t$ , onde  $t = (1, 2, \dots, u)$ , com pesos para cada conexão  $P_t$  e um viés  $B$  (*bias*). Esse sistema, que possui  $Y$  como saída, é tratado com uma combinação linear de entradas e pesos mais o valor do viés. O viés possui como função diminuir ou aumentar o domínio de uma entrada na ativação do neurônio (CASTRO; FERRARI, 2016). Na Figura 9 é apresentado um modelo gráfico de um neurônio artificial.

Figura 9 – Exemplo gráfico de um neurônio artificial



Fonte: Elaborado pelo autor (2023).

Para evitar a linearidade do sistema é adicionado uma função de ativação que engloba todos os componentes do neurônio básico. Entre outras, as funções podem ser do tipo tangencial hiperbólica (TanH), sigmoide ou linear retificada, do inglês *Rectified Linear Unit* (ReLU). Nas Equações 21, 20 e 22, adaptadas do trabalho de Castro e Ferrari (2016), são apresentadas as representações matemáticas dessas funções na ordem mencionada anteriormente:

$$f_{\tanh}(l) = \frac{e^l - e^{-l}}{e^l + e^{-l}} \quad (20)$$

$$f_{\text{sig}}(l) = \frac{1}{1 + e^{-l}} \quad (21)$$

$$f_{relu}(l) = \begin{cases} 1, & \text{se } l > 0 \\ 0, & \text{se } l \leq 0 \end{cases} \quad (22)$$

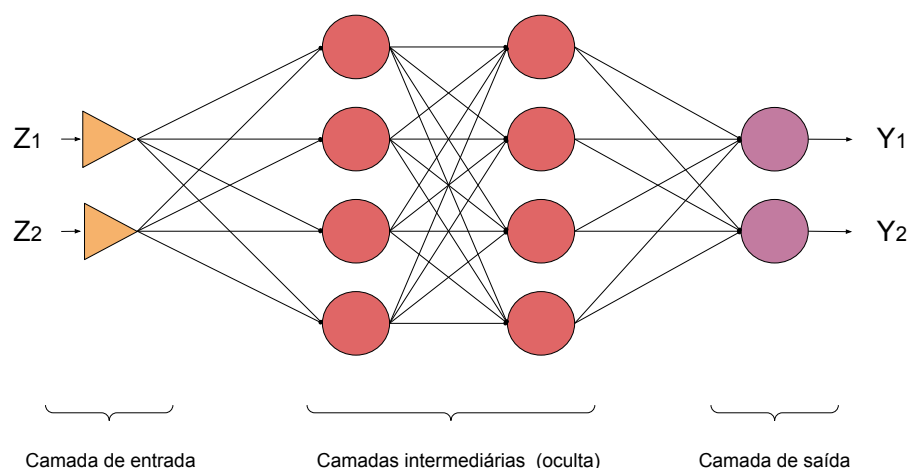
onde  $l$  é o argumento da função. Desta forma, a formulação para um neurônio genérico  $g$  é dado pela Equação 23:

$$Y_g = f(l_g) = f\left(B_g + \sum_{i=1}^u X_i P_{gt}\right) \quad (23)$$

As RNA possuem como topologia básica grupos de neurônios interligados e divididos em três camadas: entrada, oculta e saída. Na camada de entrada, os neurônios recebem as informações dos dados e as transmitem para a camada oculta. Já na camada oculta, outro grupo de neurônios recebem as amostras das unidades de entrada, processam e as transmitem para a camada de saída, na qual, gerará os resultados para o problema. A camada oculta pode possuir várias camadas intermediárias (CASTRO; FERRARI, 2016; GRUS, 2021).

Uma das redes mais utilizadas é a rede neural perceptron multicamadas, do inglês, *Multilayer Perceptron* (MLP). Ela apresenta a topologia básica, podendo ter em sua camada oculta uma ou mais camadas intermediárias. Na Figura 10 é mostrado um exemplo da estrutura de uma MLP, onde  $Z_1$  e  $Z_2$  são entradas e  $Y_1$  e  $Y_2$  as saídas do problema.

Figura 10 – Exemplo da estrutura de uma MLP



Fonte: Elaborado pelo autor (2023).

Para o treinamento dessa rede é empregado um algoritmo de retropropagação do erro, do qual, um conjunto de dados de entrada é inserido na rede e em cada camada as amostras são

processadas até gerar um resultado na camada de saída. As soluções obtidas são comparadas com as saídas desejadas e um erro é calculado. Se o valor do erro não for mínimo ou bem próximo de 0, os pesos e o viés são atualizados até que essa condição seja realizada (FACELI *et al.*, 2011).

Para as atualizações dos pesos, de forma que este minimize o erro, é empregado o uso de otimizadores, entre eles pode-se citar LBFGS, Adam e o gradiente descendente estocástico. O Limited-Broyden–Fletcher–Goldfarb–Shanno (LBFGS) é um algoritmo de otimização pertencente ao grupo de métodos Quasi-Newton para problemas irrestritos e não lineares (SANSEVERINO, 2014).

O Adam é um algoritmo definido por Kingma e Ba (2014), cujo objetivo é ser um otimizador estocástico que também usa pouca memória, usando apenas gradientes de primeira ordem. O gradiente descendente estocástico do inglês *Stochastic Gradient Descent* (SGD) é um algoritmo de otimização que também é baseado em gradiente.

O gradiente pode ser definido como um vetor de derivadas parciais que aponta na direção na qual a função cresce mais rapidamente. Para realizar essa otimização, a cada iteração, o algoritmo define aleatoriamente uma amostra do banco de dados de treinamento para realizar o cálculo do gradiente. Esse processo ocorre até que acabe o número de iterações ou se encontre uma solução viável (GRUS, 2021).

### 4.3 Considerações parciais do capítulo 4

Como foi visto no capítulo, a classificação de dados é uma parte do aprendizado de máquina capaz de rotular um conjunto de instâncias, necessitando para isso, de um treinamento para o classificador.

Para realizar esse treinamento o banco de dados deve ser dividido em alguns segmentos para que se possa treinar e testar. A forma em que se realiza essa divisão depende do problema a ser resolvido, por exemplo, em bancos de dados menores o uso da validação cruzada é mais empregada, já em um conjunto de dados muito grande a *hold-out* é utilizado.

Também foram apresentadas alguns algoritmos de classificação, cada um deles possuindo suas particularidades em rotular instâncias. Esses classificadores serão utilizados na construção dos modelos combinados. No próximo capítulo são mostradas as maneiras de combinação destas técnicas e algumas métricas empregadas para avaliar o desempenho dos algoritmos de classificação.

## 5 COMBINAÇÃO DE MODELOS

Os algoritmos de reconhecimento de padrões podem ser combinados, com o objetivo de melhorar o desempenho na classificação, trabalhar com dados desbalanceados e diminuir problemas de generalização. Existem casos em que apenas um modelo de aprendizado de máquina não consegue fazer a separação dos dados em grupos de decisão adequadamente; por isso, é empregado o uso do *ensemble* para realizar esse processo (AGGARWAL, 2015).

Técnicas baseados em árvores de decisão como a Floresta Aleatória (FA), o *Gradient Boosting* (GB), *Extreme Gradient Boosting* (XGBoost), *CatBoost* e o LightGBM ganham popularidade na classificação de dados tabulares. Já na classificação de imagens, o conjunto de técnicas de *deep learning* estão conseguindo alcançar altos desempenhos.

A combinação de modelos pode ser feita de forma homogênea, quando se emprega o mesmo modelo na sua concepção ou de forma heterogênea, na qual, utiliza-se de uma diversidade de classificadores. Já a estrutura de construção dos modelos combinados, pode ser efetuada de três maneiras, a saber, em série, em paralelo e de forma mista (PONTI Jr., 2011).

Neste capítulo, nas seções 5.1, 5.2 e 5.3 são discorridos sobre as estruturas de combinação dos classificadores. Já na seção 5.4 é esclarecido as formas nas quais as previsões são agregadas e analisadas em conjunto. Na seção 5.5 são mostrados alguns dos algoritmos de combinação de modelos mais utilizados. São mostradas algumas métricas de avaliação de classificadores na seção 5.6. Por fim, na seção 5.7 são apresentadas as considerações sobre o capítulo.

### 5.1 Modelos combinados em série

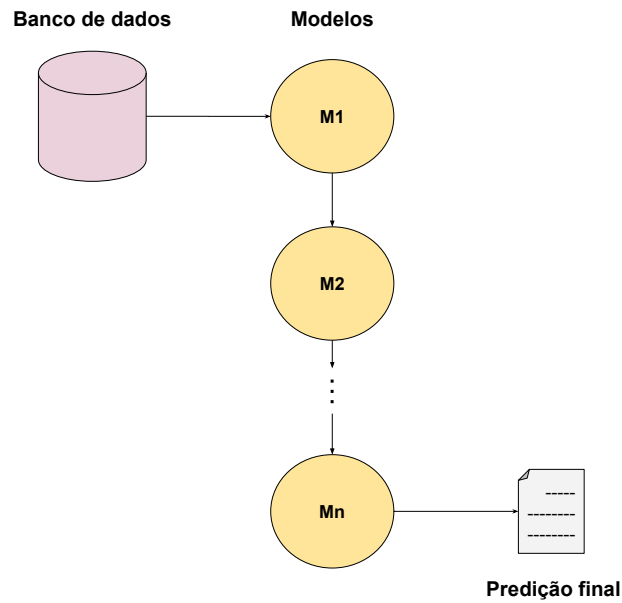
Nessa estrutura de construção, os modelos são dispostos em sequência por ordem de complexidade. Para cada nova amostra, o primeiro modelo  $M_1$  gera uma previsão. Quando o nível de confiança na classificação (grau de certeza na classe correta) de alguma amostra do conjunto de dados for baixa, essa será passada para o segundo modelo  $M_2$ , que é especializado em casos mais difíceis. Esse sistema ocorre até chegar ao último modelo  $M_n$  ou quando se tiver um nível de confiança alto na classe da amostra (ASMITA; SHUKLA, 2014; CANABARRO, 2018).

O objetivo dessa metodologia é que cada modelo aumente os resultados do modelo anterior (ASMITA; SHUKLA, 2014). Na Figura 11 é apresentado um exemplo deste processo



de agregação.

Figura 11 – Exemplo de estrutura em série

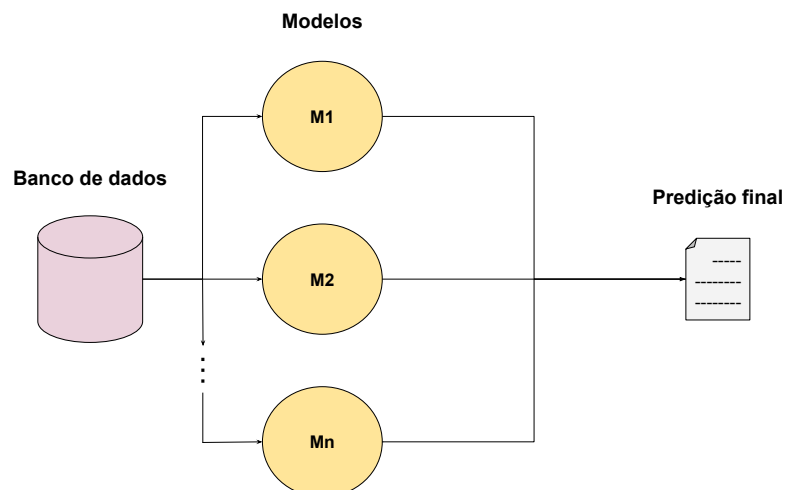


Fonte: Elaborado pelo autor (2023).

## 5.2 Modelos combinados em paralelo

Na estrutura de construção em paralelo, os modelos são criados independentemente, para que posteriormente suas predições sejam analisadas em conjunto e assim gerem uma predição final (ASMITA; SHUKLA, 2014). Na Figura 12 é apresentado um exemplo dessa estrutura.

Figura 12 – Exemplo de estrutura em paralelo

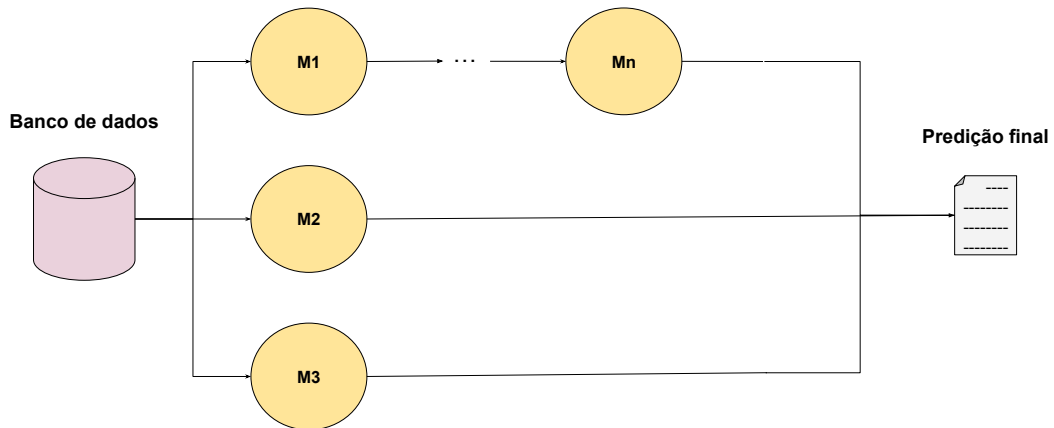


Fonte: Elaborado pelo autor (2023).

### 5.3 Modelos combinados de forma mista

A combinação de forma mista emprega os conceitos da combinação em série e em paralelo. O objetivo disso é diversificar ainda mais o processo na busca de melhores desempenhos. Na Figura 13 é mostrado como esse processo ocorre.

Figura 13 – Exemplo de estrutura mista



Fonte: Elaborado pelo autor (2023).

### 5.4 Análise das predições

Com a coleta das predições de cada classificador, se torna necessário realizar a análise e agregação dos resultados obtidos. A análise das predições pode ocorrer de diversas formas, entre elas se destacam: o voto majoritário, a média, predição dinâmica e contagem de borda.

#### 5.4.1 Voto Majoritário

As amostras são rotuladas pelo voto da maioria dos classificadores (KITTLER *et al.*, 1998). Como exemplo, suponha uma classificação binária, na qual existem três classificadores diferentes treinados individualmente no mesmo banco de dados. Uma amostra é testada nesses três classificadores e as predições dadas por estes são mostradas na Tabela 1.

Pelo voto majoritário a amostra é rotulada como pertencendo à classe 2. Em caso de empate pode-se escolher uma classe aleatoriamente, ponderar o valor dos classificadores ou até mesmo por em ordem de classificação crescente.

Tabela 1 – Predições dadas por cada um dos classificadores

Modelo	Predição
Classificador A	classe 2
Classificador B	classe 1
Classificador C	classe 2

Fonte: Elaborado pelo autor (2023).

#### 5.4.2 Média

Neste caso, a classificação de uma amostra é dado pela média das probabilidades de cada modelo. Cada classificador quando é testado em um banco de dados apresenta um conjunto de probabilidades da amostra pertencer a alguma classe. Na Tabela 2 é mostrado um exemplo de probabilidades de uma amostra pertencer à classe 1 ou a classe 2.

Tabela 2 – Probabilidades de classificação

Modelo	Classe 1	Classe 2
Classificador A	0,4	0,6

Fonte: Elaborado pelo autor (2023).

Neste sentido, segundo o classificador A, a amostra pertence à classe 2. A análise por média pode ser elaborada por média simples ou pela ponderada. Na Tabela 3 é apresentado um exemplo de média simples, no qual,  $W_a$ ,  $W_b$  e  $W_c$  são iguais a 1.

Tabela 3 – Análise por média

Classes	Classificador A	Classificador B	Classificador C	Média simples
Classe 1	$W_a * 0,4$	$W_b * 0,8$	$W_c * 0,1$	0,43
Classe 2	$W_a * 0,6$	$W_b * 0,2$	$W_c * 0,9$	0,57

Fonte: Elaborado pelo autor (2023).

No caso da Tabela 3, o maior valor de média das probabilidades foi 0,57, fazendo assim com que a classe rotulada para a amostra seja a classe 2.

#### 5.4.3 Predição dinâmica

Neste tipo de análise, o conjunto de classificadores é chamado de *pool* de classificadores. O classificador ou classificadores da *pool* são selecionados dinamicamente para cada

amostra do banco de dados de teste. Essa seleção é feita através de características da amostra na qual se quer rotular (KO *et al.*, 2008).

Exemplos de algoritmos dinâmicos são o *K-Nearest-Oracles* (KNORA), *K-Nearest Oracle-Union* (KNORA-U) e o *K-Nearest Oracle-Eliminate* (KNORA-E). O KNORA é um algoritmo que busca no banco de dados de validação os pontos mais próximos (*k*-vizinhos) da amostra de teste que se pretende classificar. Posteriormente procura na *pool* de classificadores os modelos que melhor classificaram esses elementos para serem utilizados na previsão da amostra de teste (KO *et al.*, 2008).

O KNORA-U e o KNORA-E são algoritmos derivados do KNORA. O KNORA-U usa modelos que classificam corretamente qualquer um das amostras *K* vizinhos mais próximos. Já o KNORA-E usa apenas modelos que classificam corretamente todas amostras *K* vizinhos mais próximos (KO *et al.*, 2008).

#### 5.4.4 Contagem de borda

A contagem de borda atribui uma pontuação a cada probabilidade de classe indicada pelos classificadores. Essa pontuação é definida por critérios próprios do autor que a aplica. Ao final do processo, os pontos são somados e vence aquela classe que obtiver o maior número de pontos (ASMITA; SHUKLA, 2014). Na Tabela 4 é mostrado um exemplo de contagem de borda.

Tabela 4 – Exemplo de contagem de borda

Classes	Classificador A		Classificador B		Classificador C		Resultado
	Probabilidade	Pontuação	Probabilidade	Pontuação	Probabilidade	Pontuação	Somatório
1	0,4	1	0,8	2	0,1	1	4
2	0,6	2	0,2	1	0,9	2	5

Fonte: Elaborado pelo autor (2023).

Como pode ser observado na Tabela 4, cada probabilidade do classificador recebeu uma pontuação. Os maiores valores de probabilidade receberam uma pontuação de valor 2 e as menores 1. Ao final do processo somou-se a pontuação para cada classe. A classe 2 conseguiu a maior pontuação, valor de 5, desta forma a amostra será rotulada como pertencente a classe 2. Se caso ocorra empate pode ser realizado os mesmos procedimentos do voto majoritário.

## 5.5 Algoritmos de combinação de modelos

Com o avanço do estudo na combinação de modelos e suas estruturas, foram propostas técnicas que realizam essa agregação. Entre as mais populares estão o *bagging*, *boosting* e o empilhamento (*stacking*).

### 5.5.1 *Bagging*

No *bagging* os modelos são criados individualmente e em paralelo, empregando subconjuntos aleatórios do banco de dados de treinamento. Neste processo são criados diversos modelos, cada um, possuindo vantagens e desvantagens advindas do seu subconjunto de treinamento (WITTEN; FRANK, 2011). Devido a isso, dependendo do subconjunto utilizado, o resultado de acurácia pode ser alto ou baixo para determinado modelo (ASMITA; SHUKLA, 2014).

Para realizar a previsão final no banco de dados de teste, é realizado um processo de votação por maioria, no qual, como já mencionado, as amostras são rotuladas pela decisão da maioria dos modelos. Essa técnica é normalmente utilizada em uma combinação de modelos homogênea e com algoritmos de aprendizagem de máquina instáveis como as árvores de decisão e as redes neurais (WITTEN; FRANK, 2011; ASMITA; SHUKLA, 2014).

Algoritmos denominados de instáveis são aqueles que com pequenas mudanças no conjunto de treinamento podem apresentar uma alta alteração no seu valor de acurácia. O *bagging* quando empregado com árvores de decisão, pode receber o nome de floresta aleatória (ASMITA; SHUKLA, 2014).

A FA é um algoritmo variante da abordagem de *bagging* que usa as árvores de decisão para a criação do seu modelo. Ele segue o passo a passo do *bagging*, com a criação de árvores, empregando os subconjuntos de treinamento, e a predição final sendo realizada pelo voto majoritário. A diferença é a utilização da seleção aleatória de atributos para a construção dos modelos. Com isso se consegue uma maior diversidade de modelos na construção do *bagging* (AGGARWAL, 2015).

### 5.5.2 *Boosting*

Já no *boosting* os modelos são concebidos sequencialmente, onde, no início, são criados classificadores mais simples e menos complexos, também chamados de fracos, para

que em seguida modelos mais complexos sejam desenvolvidos com base nos erros de predições anteriores (FRIEDMAN *et al.*, 2009; ASMITA; SHUKLA, 2014). Essa técnica também é geralmente empregada em uma combinação de modelos homogênea.

O objetivo desta metodologia é fazer com que os novos modelos criados se tornem especialistas nas instâncias errôneas preditas pelos modelos antigos (ASMITA; SHUKLA, 2014). O *boosting* também é muito empregado com as árvores de decisão, dessa junção, surgiram algoritmos como o GB e o LightGBM.

O GB emprega a metodologia do *boosting* em árvores de decisão. Ele também pode ser empregado em outros modelos. Esse algoritmo cria sequencialmente diversas árvores de decisão, nas quais, cada uma tenta diminuir o erro de previsão gerada pela árvore anterior empregando uma função de perda (FRIEDMAN *et al.*, 2009; BOEHMKE; GREENWELL, 2019). Exemplos de funções de perda são o exponencial e a logarítmica.

Ajustar essa função de perda para diminuir o erro em cada interação do algoritmo, gera um problema que necessita de otimização. Para isso é empregado o algoritmo gradiente descendente. Essa técnica tem como princípio, a análise do gradiente (um vetor de derivadas parciais que aponta na direção onde a função cresce). Desta forma, primeiramente, pega-se um ponto aleatório, calcula-se o gradiente, posteriormente move o ponto na direção na qual o vetor indica e contabiliza o gradiente novamente. Faz esse processo até encontrar uma solução ótima (FRIEDMAN *et al.*, 2009; BOEHMKE; GREENWELL, 2019; GRUS, 2021).

Já o LightGBM é um derivado do GB, no qual, tem como objetivo, conseguir melhores resultados quando o banco de dados é grande e possui uma alta dimensionalidade. Como diferencial, o ganho de informação de um nó da árvore não é calculado para todas as instâncias, mas apenas naquelas que possuem um maior gradiente. Como vantagens para essa técnica tem-se um menor uso da memória, resultados melhores de acurácia e um processamento distribuído (KE *et al.*, 2017a).

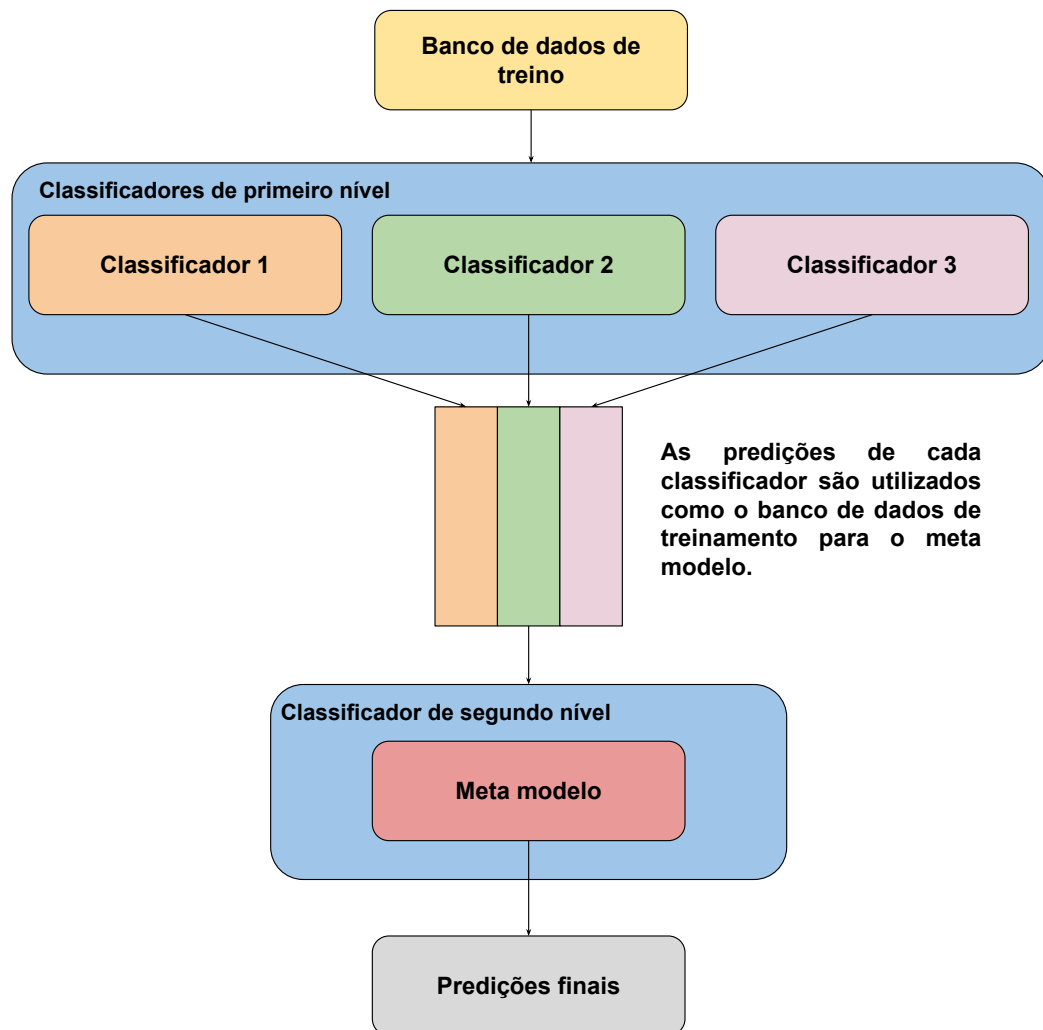
Também é realizado um agrupamento de atributos mutualmente exclusivos, para torná-los apenas um atributo. O objetivo disso é reduzir o número de atributos de uma forma que não diminua o desempenho do classificador criado (KE *et al.*, 2017a).

### 5.5.3 Modelos empilhados

O empilhamento ou *stacking* no inglês é um modelo *ensemble* desenvolvido por Wolpert (1992). A ideia principal do modelo é a de empilhar predições individuais de cada

algoritmo, para posteriormente utilizá-las como entrada para um novo classificador (também chamado de meta-modelo). Esse processo é denominado de meta aprendizado, no qual, os modelos de base são conhecidos como classificadores de primeiro nível e o meta modelo que aprendeu com as predições, de classificador de segundo nível (AGGARWAL, 2015; WOLPERT, 1992). Na Figura 14 é apresentado o esquemático do empilhamento.

Figura 14 – Esquema do empilhamento



Fonte: Elaborado pelo autor (2023).

A técnica foi criada visando efetuar uma combinação não linear de classificadores, desta forma, tenta corrigir um problema de generalização que pode ocorrer em um determinado conjunto de treino. O meta modelo aprenderá a relação entre as predições e os rótulos das classes, podendo assim classificar corretamente o rótulo de uma amostra. Além do uso das predições, também pode-se empregar as probabilidades de cada classe no treinamento do classificador de

segundo nível (AGGARWAL, 2015; WOLPERT, 1992).

O empilhamento pode acontecer com mais níveis de classificadores dependendo do problema a ser realizado. Ele também é muito empregado com o uso da validação cruzada, isso porque, em sua forma original, é empregado o mesmo banco de dados de treinamento para treinar os classificadores de primeiro nível e preparar o treinamento de segundo nível, podendo acarretar um ajuste excessivo (AGGARWAL, 2015).

Esse algoritmo apresenta, geralmente, resultados melhores que o *bagging* e o *boosting*, sendo muito empregado em conjuntos de modelos heterogêneos. Como desvantagem, diferente de outros algoritmos, tem-se a necessidade de criação do meta modelo, isso torna o processo mais demorado e mais custoso computacionalmente.

## 5.6 Métricas para a avaliação do classificador

Após treinar os classificadores é importante validar seu desempenho em um banco de dados de teste, para isso são empregados métricas de avaliação como, entre outros, a matriz de confusão, acurácia, precisão, sensibilidade e *F-score*.

### 5.6.1 Matriz de confusão

Em uma classificação binária, na qual queremos classificar se um indivíduo pertence a uma classe positiva ou a uma classe negativa, temos verdadeiro positivo (*VP*), verdadeiro negativo (*VN*), falso positivo (*FP*) e o falso negativo (*FN*).

*VP* representa elementos da classe positiva que foram classificados como positivo, já *VN* são elementos da classe negativa que foram preditos como negativo. O *FP* são amostras da classe negativo classificados como positivo. Por fim, *FN* são elementos da classe positiva preditos como negativo.

A matriz de confusão tem como objetivo mostrar as relações entre os elementos preditos pelo modelo e o real rótulo de classificação das amostras. Ela também pode ser empregada em classificações multi classes. Na Tabela 5 é mostrado um exemplo de matriz de confusão para uma classificação binária (positivo e negativo).



Tabela 5 – Exemplo de uma matriz de confusão de uma classificação binária

		Classe prevista	
		Classe positiva	Classe negativa
Classe original	Classe positiva	VP	FN
	Classe negativa	FP	VN

Fonte: Adaptado de (CASTRO; FERRARI, 2016).

### 5.6.2 Acurácia

A acurácia é uma métrica de avaliação que mede o desempenho geral de um modelo de classificação. Ela é a relação entre as predições acertadas pelo classificador e o número total de elementos testado pelo modelo. Na Equação 24 é apresentado a fórmula da acurácia:

$$Acc = \frac{n^{\circ} \text{ acertos}}{\text{total de elementos}} \quad (24)$$

Como a acurácia é uma métrica global, ela não é capaz de nos informar dados locais de cada classe, e em bancos de dados desbalanceados, apenas ela, não consegue definir o quão bom o classificador se comportou. Isso porque uma classe que possui muitos elementos em comparação a outra pode acabar classificando melhor e puxando para cima o resultado geral. Dessa forma se torna importante observar como está a classificação local em cada classe.

### 5.6.3 Precisão

A precisão pode ser definida como uma medida de qualidade (CASTRO; FERRARI, 2016). Em termos de uma classificação binária pode-se calculá-la através da Equação 25:

$$P = \frac{VP}{VP + FP} \quad (25)$$

### 5.6.4 Sensibilidade

A sensibilidade representa a completude do modelo (CASTRO; FERRARI, 2016). Para uma classificação binária, na Equação 26, temos:

$$Sens = \frac{VP}{VP + FN} \quad (26)$$

### 5.6.5 *F-score*

O *F-score* é uma média harmônica entre a sensibilidade e a precisão, seus valores se encontram entre 0 e 1 (CASTRO; FERRARI, 2016). Na Equação 27 é apresentada a fórmula do *F-score*:

$$F - score = \frac{2 \cdot VP}{2 \cdot VP + FP + FN} \quad (27)$$

## 5.7 Considerações parciais do capítulo 5

Neste capítulo foram mostradas algumas formas de combinar classificadores, como já mencionado, isso é feito como uma forma de melhorar os resultados da classificação. Também foram apresentadas formas de avaliar o quão bem o classificador está realizando sua função, com isso, pode-se analisar os resultados e fazer comparativos entre eles.

Essas informações são importantes para o desenvolvimento desse trabalho. Isso porque para a criação do classificador se empregou o uso da média ponderada apresentada na seção 5.4. Além disso, algumas técnicas de combinação como os classificadores que utilizam previsões dinâmicas e os modelos empilhados serão utilizados na comparação dos resultados obtidos pelo classificador criado.

No próximo capítulo são comentados alguns algoritmos de otimização. Eles podem ser empregados como forma de encontrar melhores parâmetros para os classificadores e valores de pesos em uma etapa de combinação destes.

## 6 OTIMIZAÇÃO

Este capítulo apresenta uma introdução a otimização, mostrando algumas definições e técnicas. Na seção 6.1 é conceituado as formas mais básicas de otimização. A otimização COBYLA é estudada na seção 6.2, já a Powell é definida na seção 6.3. As meta-heurísticas são apresentadas na seção 6.4. Por fim, as considerações deste capítulo são mostradas na seção 6.5.

A otimização visa encontrar os melhores parâmetros que solucione um determinado problema, para isso são empregados algoritmos que possuem por função maximizar ou minimizar uma função de custo (ÁVILA, 2020). Para o entendimento melhor sobre o assunto é necessário explicar algumas definições apresentadas por Ávila (2020):

- **Função objetiva:** É a equação matemática a ser otimizada.
- **Parâmetros:** São as variáveis que devem ser ajustadas na obtenção do melhor resultado para o problema.
- **Espaço de busca:** Espaço que pode ser delimitado ou não de possíveis valores para os parâmetros.
- **Minimização:** Em determinados casos se procura a minimização de uma função. Por exemplo, pode-se mencionar um problema no qual se busca um produto com um menor custo financeiro.
- **Maximização:** O caso contrário também é válido, quando a busca é por um valor que maximize a função. Um exemplo seria um problema no qual se busca maximizar o rendimento de uma máquina.
- **Minimização unidimensional:** No problema unidimensional a função apresenta apenas uma variável independente.
- **Minimização multidimensional:** Neste tipo de problema a função apresenta mais de um variável independente.

Entre as diversas formas de otimização de um problema evidenciam-se as otimizações mais básicas, COBYLA, Powell e as meta-heurísticas.

### 6.1 Otimizações básicas

Esses tipos de otimizações são as formas mais simples de se encontrar uma solução para a função objetiva. Mesmo que sejam algoritmos de simples criação, eles podem sofrer com o tempo de execução e problemas em encontrar uma solução ótima. Em alguns casos não são

considerados como algoritmos de otimização. A otimização exaustiva e randômica são exemplos de otimizações básicas.

### 6.1.1 Otimização exaustiva

A otimização exaustiva é a forma mais básica de se otimizar qualquer função. Nela, são testados todas as variáveis que se encontram dentro de um espaço de busca. Desta forma, o algoritmo testa um por um até encontrar um resultado ótimo.

### 6.1.2 Otimização randômica

Na otimização de forma randômica é implementado uma procura aleatória por variáveis em um espaço de busca pré-definido pelo programador, desse modo, apenas parte do espaço de busca é testado. Ela é uma das formas mais simples de realizar o processo de otimização e ainda traz vantagens em comparação com uma busca exaustiva, como uma maior velocidade e eficiência. Um exemplo de passos para a criação desse algoritmo, adaptado do trabalho de Luke (2013), é apresentado a seguir:

1. (*Melhor*)  $\leftarrow$  Aplique alguma solução candidata aleatória inicial dentro de um espaço de busca.
2. (*Sol*)  $\leftarrow$  Repita o processo e encontre uma nova solução candidata aleatória.
3. Se a solução (*Sol*) for melhor que a solução (*Melhor*), então
4. (*Melhor*)  $\leftarrow$  (*Sol*)
5. Repita o processo até (*Melhor*) ser uma solução ótima ou acabar o número de iterações, ou tempo.
6. Retorne (*Melhor*).

## 6.2 COBYLA

A otimização restrita por aproximação linear, do inglês *Constrained Optimization By Linear Approximation* (COBYLA), é um método de otimização desenvolvido por Powell (1994) para a resolução de problemas nos quais não se sabe a derivada da função objetivo.

O algoritmo do COBYLA começa criando um modelo aproximado da solução atual e posteriormente define uma nova direção de busca. A nova direção escolhida fica limitada a uma vizinhança em torno da posição inicial. Essa vizinhança é chamada de região de confiança

(ALMEDALLAH *et al.*, 2021).

O algoritmo emprega os valores de uma função  $F(x_t)$ , onde  $t = (1, 2, \dots, n)$ , nos vértices de um simplex. Eles são interpolados por um polinômio linear  $L(x)$ , onde cada iteração necessita de uma região de confiança com um raio  $\Delta > 0$  que é ajustado automaticamente (POWELL, 2007).

Ele estima os gradientes usando interpolação linear. Dessa forma não é necessário o cálculo das derivadas da função objetivo, reduzindo seu custo computacional em comparação com outras técnicas de otimização (ALMEDALLAH *et al.*, 2021).

O algoritmo simplex é formado por um conjunto de operações empregadas na busca de uma solução ótima. Primeiramente, o algoritmo é iniciado com uma solução simples (as soluções no método simplex são os vértices do espaço de busca). Posteriormente o método busca por vértices vizinhos e faz uma comparação entre eles e o vértice inicial. Se o vértice inicial não for o ponto ótimo, pega a melhor solução encontrada atualmente e reinicia o processo de comparação. Esse ciclo ocorre até que o algoritmo encontre a solução ótima (HILLIER; LIEBERMAN, 2010).

### 6.3 Powell

Este algoritmo foi proposto por Powell (1964) e realiza um processo de minimizações unidimensionais, nas quais, são feitas percorrendo todo o conjunto de direções (vetores), sendo atualizado a cada iteração. O processo não emprega o uso de derivadas, por isso, possui um menor custo computacional e consegue resultados mais rápidos (POWELL, 1964; PRESS *et al.*, 2007).

As iterações começam buscando em  $n$  direções linearmente independentes  $(D_1, \dots, D_n)$  o ponto  $p_0$  que seja minimizante da função  $f$ . Posteriormente o método gera novas direções conjugadas e novos pontos são encontrados  $(p_1, \dots, p_n)$ . O processo termina até que seja encontrado o melhor resultado (POWELL, 1964). Para a sua criação deve-se seguir os seguintes passos adaptados do trabalho de Powell (1964):

1. Para  $t = (1, 2, \dots, n)$  determine o valor de  $\lambda_t$  que minimize  $f(p_{t-1} + \lambda_t D_t)$  e defina  $p_t = p_{t-1} + \lambda_t D_t$ .
2. Para  $t = (1, 2, \dots, n - 1)$  substitua  $D_t$  por  $D_{t+1}$ .
3. Substitua  $D_n$  por  $(p_n - p_0)$ .
4. Encontre um  $\lambda$  que minimize  $f(p_n + \lambda \{p_n - p_0\})$  e defina  $p_0$  por  $p_0 + \lambda \{p_n - p_0\}$ .

É mencionado por Press *et al.* (2007) que devido à substituição de  $D_0$  por  $(p_n - p_0)$ , pode produzir um conjunto de direções que acabam se tornando linearmente dependentes, fazendo com que o algoritmo encontre o mínimo da função em apenas um sub-espaço. Devido a esse fator o método de Powell sofreu uma modificação. Essa modificação propõe deixar  $(p_n - p_0)$  como uma nova direção, mas descartar a direção passada na qual a função  $f$  encontrou sua maior diminuição. O algoritmo é implementado por Press *et al.* (2007).

## 6.4 Otimização por meta-heurísticas

As meta-heurísticas fazem parte de um subconjunto da otimização estocástica. Nesse tipo de classe, estão algoritmos e técnicas que aplicam aleatoriedade na obtenção de resultados para otimizar uma função. Em casos que não se conhece muito sobre os aspectos do problema, o uso das meta-heurísticas se torna essencial (LUKE, 2013). A seguir são apresentados algumas meta-heurísticas: otimização por enxame de partículas, otimização por enxame de partículas relativísticas e algoritmos genéticos.

### 6.4.1 Otimização por enxame de partículas

A otimização por enxame de partículas do inglês *Particle Swarm Optimization* (PSO) é uma técnica de otimização estocástica, na qual, diferente de outros algoritmos, possui uma única população que se adapta as novas descobertas do espaço de busca. Ela foi elaborada por Kennedy e Eberhart (1995) e tem como inspiração para a sua criação bandos e enxames de indivíduos. Em sua proposta, as soluções candidatas para a resolução do problema são denominadas de enxame de partículas, as quais, são movidas pelo espaço em busca da solução ótima (PAIVA, 2018; LUKE, 2013).

Existem dois pontos que devem ser analisados em uma partícula, sua localização e velocidade. A localização é o ponto no qual a partícula se encontra no espaço de busca. Já a velocidade é a relação entre o deslocamento da partícula em uma faixa de tempo (PAIVA, 2018; LUKE, 2013).

Para a criação do algoritmo, as partículas começam com locais e velocidades aleatórias no espaço de busca. Posteriormente é observado o local conhecido mais adequado por todas as partículas e o local conhecido mais adequado que foi descoberto por cada uma até agora (PAIVA, 2018; LUKE, 2013).

A cada iteração será feito uma atualização das melhores posições e das velocidades. A atualização da posição e da velocidade da partícula são apresentadas nas Equações 28 e 29 adaptado do trabalho de Paiva (2018):

$$x_{\sigma d} = x_{\sigma d} + v_{\sigma d} \quad (28)$$

$$v_{\sigma d} = wv_{\sigma d} + c_1r_1(p_{\sigma d} - x_{\sigma d}) + c_2r_2(p_{gd} - x_{\sigma d}) \quad (29)$$

onde  $x_{\sigma d}$  e  $v_{\sigma d}$  são a posição e a velocidade da partícula  $\sigma$ , respectivamente,  $w$  é a ponderação de inércia, na qual representa a velocidade inicial da partícula,  $r_1$  e  $r_2$  representam números aleatórios entre 0 e 1,  $c_1$  e  $c_2$  são os parâmetros cognitivos e sociais (são taxas de aprendizado, que auxiliam na auto exploração da partícula e no deslocamento global do enxame, respectivamente),  $p_{\sigma d}$  é a melhor posição encontrada por uma partícula  $\sigma$ , já  $p_{gd}$  é a melhor posição encontrada por todas as partículas até o momento e  $d$  é a dimensão (PAIVA, 2018).

#### 6.4.2 Otimização por enxame de partículas relativísticas

A otimização por enxame de partículas relativísticas, do inglês, *Relativistic Particle Swarm Optimization* (RPSO) é um algoritmo criado por Roder *et al.* (2020), que tenta melhorar o clássico PSO empregando a distorção espaço temporal. O algoritmo usa como base a teoria da relatividade de Albert Einstein proposta por Einstein (1916). Muitas das ideias elaboradas por Einstein foram adaptadas para a mecânica clássica e entre elas existe o conceito de momento (RODER *et al.*, 2020). Na Equação 30 é mostrado como calcular o momento em um espaço tridimensional adaptado do artigo de Roder *et al.* (2020):

$$p(v) = \psi(v)Mv \quad (30)$$

sendo  $v = [v_x, v_y, v_z]$  é a velocidade de um corpo de massa  $M$  localizado em um espaço tridimensional e  $\psi$  é o fator de Lorentz. O fator de Lorentz é descrito na Equação 31 adaptada do trabalho de Roder *et al.* (2020):

$$\psi(v) = \sqrt{\frac{1}{1 - (\frac{|v|}{c})^2}} \quad (31)$$

onde  $v$  é a magnitude do vetor de velocidade e a velocidade da luz é representado por  $c$ .

A principal mudança realizada neste algoritmo é efetuada na Equação 29 mostrada anteriormente. O novo valor da velocidade é exposto na Equação 32:

$$v_{\sigma d} = p(v_{\sigma d}) + c_1 r_1 (p_{\sigma d} - x_{\sigma d}) + c_2 r_2 (p_{gd} - x_{\sigma d}) \quad (32)$$

onde  $p$  representa o momento relativístico em um espaço n-dimensional.

Como mostrado na Equação 30, para o cálculo do momento de cada partícula é necessário o valor de uma massa  $M$ , devido a isso, o algoritmo, utilizando uma distribuição uniforme com valores entre 0 e 1, cria esse valor (RODER *et al.*, 2020).

A mudança feita na Equação 32 apresenta como diferencial o efeito da velocidade da luz e da massa em sua formulação. Devido a isso, as partículas conseguem reconhecer informações espaço temporais do enxame. A principal vantagem deste algoritmo em comparação com o PSO vem da sua alta robustez, tentando aumentar a convergência e não se prendendo a soluções ótimas locais (RODER *et al.*, 2020).

### 6.4.3 Otimização por algoritmos genéticos

O Algoritmo genético (AG) foi explanado por Holland (1975) e utiliza como base de criação conceitos de estratégias de evolução como aptidão, reprodução, remontagem da população e seleção (LUKE, 2013). No início, o algoritmo cria uma população espalhada de forma aleatória dentro de um espaço de busca (COLEY, 1999).

Posteriormente são conduzidas operações de seleção, cruzamento e mutação, com o objetivo de deslocar a população na direção de uma solução ótima para o problema. A seleção tenta encontrar os indivíduos mais aptos (com o melhor desempenho), dos quais, suas informações serão empregadas para as futuras gerações. Os indivíduos menos aptos são descartados (LUKE, 2013; COLEY, 1999).

Na próxima operação é feito a etapa de cruzamento, na qual, de maneira similar a reprodução, os indivíduos trocam informações entre si. Para realizar esse processo pega-se pares de indivíduos selecionados anteriormente e faz a troca de dados. O cruzamento pode ser uniforme, de um ponto ou de dois pontos (LUKE, 2013; COLEY, 1999).

No cruzamento uniforme dados são copiados aleatoriamente do par de indivíduos para a geração do novo indivíduo. Já no cruzamento de um ponto, os dados não são retirados de



forma aleatória e sim usando como base uma marcação que divide as partes que serão retiradas do par de indivíduos. Já no caso do cruzamento de dois pontos são feitas duas marcações para dividir os dados (LUKE, 2013).

Já na mutação é efetuado uma mudança aleatória em um ou mais dados de cada indivíduo recém-criado. Isso é feito visando evitar soluções ótimas locais. Acabando essas três operações, uma nova população é criada e o ciclo se reinicia. O processo termina até que o número de iterações tenha acabado ou quando algum critério tenha sido atingido (COLEY, 1999; LUKE, 2013).

## **6.5 Considerações parciais do capítulo 6**

Neste capítulo foram apresentados conceitos e definições sobre otimização. A otimização é uma parte importante no aprendizado de máquina, podendo ser utilizado na definição de parâmetros dos algoritmos, pesos para os classificadores e na minimização ou maximização de alguma função objetivo.

Neste trabalho, as principais aplicações da otimização são na escolha dos parâmetros do classificador, onde é empregado a busca randômica e na definição de pesos para os classificadores individuais para o processo de criação do classificador por votação por média ponderada.

No próximo capítulo são apresentados os bancos de dados empregados e toda a metodologia proposta para criação e validação do modelo.

## 7 METODOLOGIA

Neste capítulo é apresentado os bancos de dados e metodologias empregadas na criação dos classificadores para o diagnóstico das MER. Na seção 7.1, 7.2 e 7.3 são mostradas as especificações dos bancos de dados empregados. Já na seção 7.4 são expressos as configurações do computador utilizado e as bibliotecas empregadas para a construção dos classificadores. Na seção 7.5 são exibidos as técnicas aplicadas para realizar o pré-processamento dos sinais. A metodologia de criação do modelo é demonstrada na seção 7.6. Por fim, as considerações do capítulo são realizadas na seção 7.7.

Para validar o modelo de aprendizado de máquina utilizou-se três bancos de dados de falhas em MER. O primeiro apresenta defeitos combinados empregando uma bancada experimental que contém um motor de corrente contínua, sendo desenvolvido por Martins *et al.* (2022) e disponível publicamente no site <https://ieee-dataport.org/documents/composed-fault-dataset-comfaulda>.

Já o segundo, criado por Chegini *et al.* (2022), possui suas incorreções localizadas nos rolamentos, nos quais, estão acoplados a um motor de indução e está acessível ao público em <https://www.kaggle.com/datasets/pouriyaamini/intelligent-bearing-fault-diagnosis-dataset?resource=download>. Por fim, o terceiro, desenvolvido por Jung *et al.* (2023), é um banco de dados de curto-circuito nas bobinas e nas espiras dos estatores de motores síncronas de ímãs permanentes. Ele está disponível através do site <https://www.sciencedirect.com/science/article/pii/S2352340923000707>.

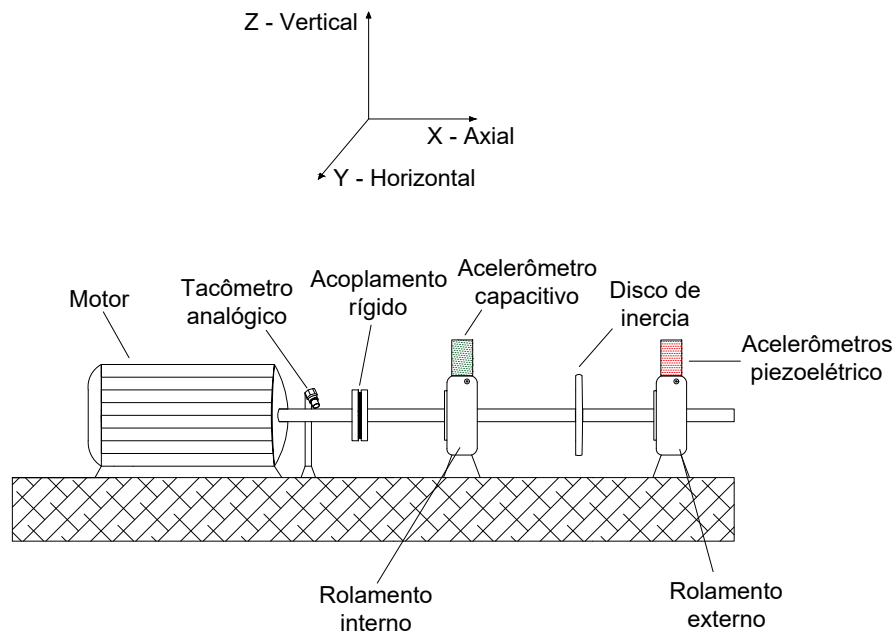
### 7.1 Primeiro banco de dados

O banco de dados é composto por 2162 conjuntos de sinais de vibração, cada um, sendo adquirido em um intervalo de tempo de cinco segundos, a uma frequência de 50 kHz. Cada sinal possui 250000 amostras e para cada medição era ajustado a velocidade do motor. A coleta dos dados foi feita usando a bancada experimental *Alignment Balance Vibration Trainer (ABVT)*, fabricada pela *SpectraQuest*. A bancada do experimento é ilustrada de forma esquemática na Figura 15.

Para realizar as medições foram empregados: um tacômetro analógico, um acelerômetro capacitivo localizado no rolamento interno, três acelerômetros piezoelétricos localizados no rolamento externo. Os acelerômetros realizam medições nas direções axial, vertical e hori-

zontal. Nas Tabelas 6 e 7 são mostrados as especificações do motor e dos sensores utilizados respectivamente.

Figura 15 – Bancada experimental do primeiro banco de dados



Fonte: Elaborado pelo autor (2023).

Tabela 6 – Especificações do motor de corrente contínua

Especificação	Dados
Potência	0,25 hp
Frequência	[12 60] Hz
Massa	22 kg
Comprimento do eixo	520 mm
Diâmetro do eixo	16 mm
Diâmetro do rotor	152,4 mm
Distância entre rolamentos	390 mm

Fonte: Retirado de (MARTINS *et al.*, 2022).

Tabela 7 – Especificações dos sensores empregados no primeiro banco de dados

Sensor	Modelo	Fabricante
Tacômetro	MT-190	Monarch instrument
Acelerometro capacitivo	4332M1-002-060	TEconnetivity
Acelerômetro piezoelétrico	601A01	IMI Sensors

Fonte: Retirado de (MARTINS *et al.*, 2022).

Os atributos do banco de dados são o tempo, as medições do tacômetro, do acelerômetro capacitivo nas direções axial, vertical e horizontal e as dos acelerômetros piezoelétricos também nas direções axial, vertical e horizontal.

Os experimentos possuem 7 condições de operação: normal, desalinhamento horizontal, desalinhamento vertical, desbalanceamento, desalinhamento horizontal combinado com desbalanceamento, desalinhamento vertical combinado com desbalanceamento e uma mistura entre desalinhamento horizontal e desalinhamento vertical. A Tabela 8 mostra o número de sinais coletados para cada condição de operação.

Tabela 8 – Número de amostras para cada condição de operação da máquina no primeiro banco de dados

Classe	Condição de operação	Nº de amostras
1	Normal	245
2	Desalinhamento horizontal	195
3	Desalinhamento vertical	243
4	Desbalanceamento	336
5	Desalinhamento vertical + desbalanceamento	351
6	Desalinhamento horizontal + desbalanceamento	351
7	Desalinhamento vertical + desalinhamento horizontal	441

Fonte: Retirado de (MARTINS *et al.*, 2022).

Para fazer as falhas de desbalanceamento, foram colocados parafusos no disco de inércia com valores de massas conhecidos. Para criar as falhas de desalinhamento foram feitas movimentações na base do motor, sendo estes no sentido horizontal ou vertical.

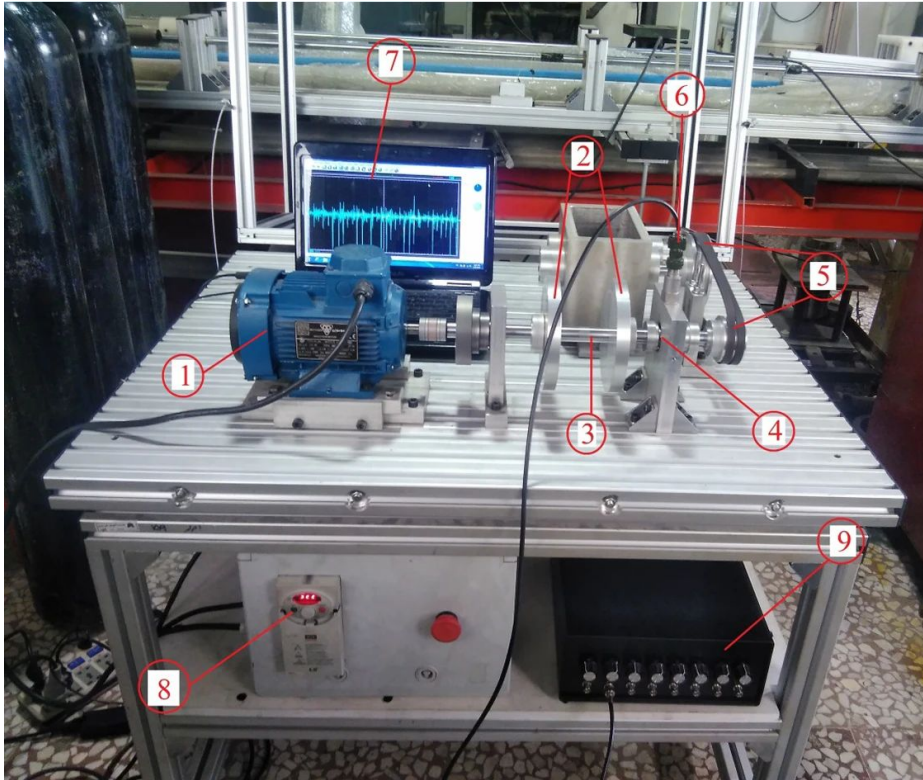
## 7.2 Segundo banco de dados

O segundo banco de dados retirado de Chegini *et al.* (2021) é composto por 178 sinais de vibração, cada um desses possuindo 3570 amostras. A bancada experimental feita pelos autores é constituída por um motor de indução e um inversor com potências de 0,5 cv, dois rolamentos do tipo Nu 204 ECP, um acelerômetro do modelo AC102-1A, um sistema polia e correia com uma relação de 2:1, um eixo com um diâmetro de 20 mm e comprimento de 50 mm, dois discos de inércia para simular desbalanceamento e um sistema de aquisição de dados com uma frequência máxima de amostragem de 250 kHz e 8 canais (CHEGINI *et al.*, 2022). Na Figura 16 é apresentado a bancada experimental para a coleta dos dados.

Os dados foram adquiridos a uma amostragem de 17,25 kHz e a uma velocidade de 1900 rpm. Eles são divididos em nove classes que vão desde do sistema saudável até o sistema

com falhas em conjunto (CHEGINI *et al.*, 2022). Na Tabela 9 são mostradas as classes que compõe o banco de dados.

Figura 16 – Bancada experimental do segundo banco de dados



- 1 - Motor    2 - Discos de inércia    3 - Eixo    4 - Rolamento**  
**5 - Mecanismo de correia    6 - Acelerômetro    7 - Computador**  
**8 - Inversor    9 - Sistema de aquisição de dados**

Fonte: Adaptado de (CHEGINI *et al.*, 2022).

Tabela 9 – Número de amostras para cada condição de operação do segundo banco de dados

Classe	Condição de operação	Nº de amostras
1	Rolamento saudável	20
2	Um sulco de largura pequena no anel interno do rolamento	20
3	Um sulco de largura grande no anel interno do rolamento	19
4	Um defeito de superfície no anel interno do rolamento	20
5	Dois ranhuras de largura pequena no anel interno do rolamento	20
6	Um sulco de largura grande no anel externo do rolamento	19
7	Um defeito de superfície no anel externo do rolamento	20
8	Um sulco no anel externo e interno do rolamento	20
9	Desbalanceamento	20

Fonte: Adaptado de (CHEGINI *et al.*, 2021; CHEGINI *et al.*, 2022).

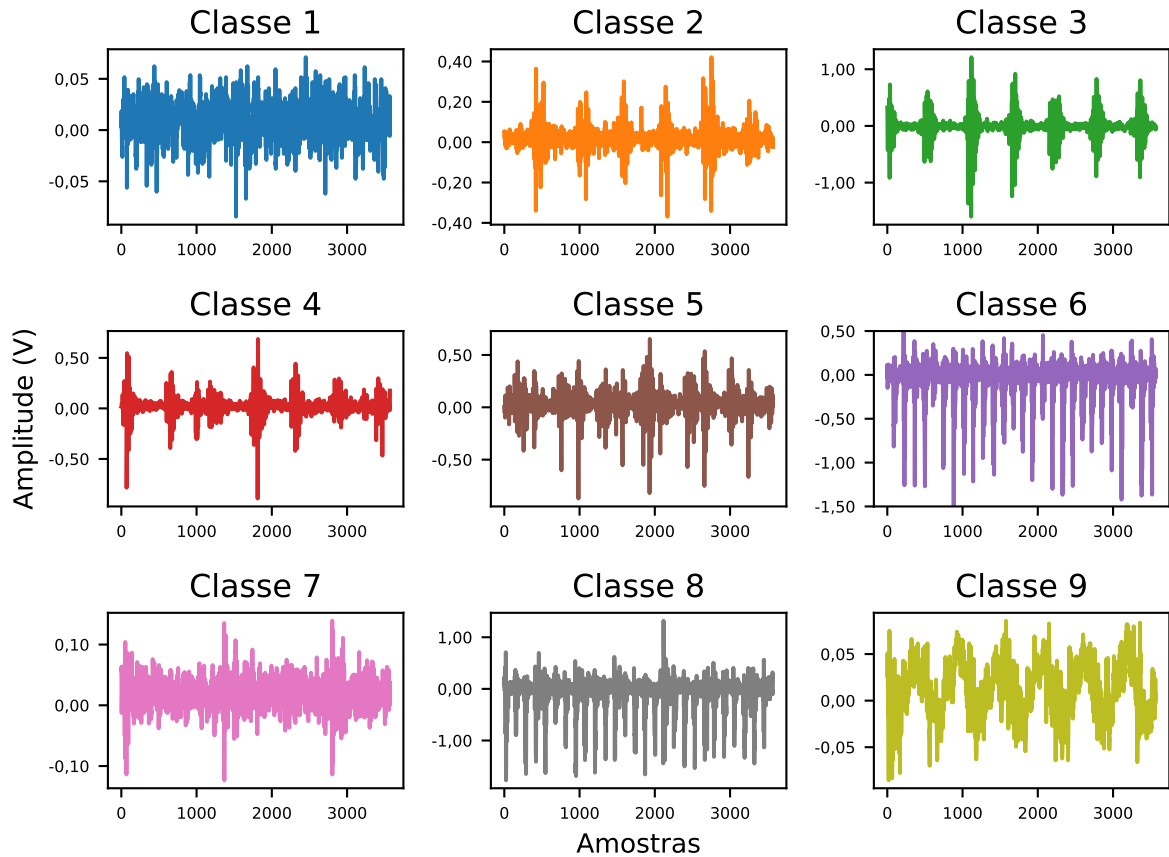
Para fazer o desbalanceamento foram colocadas massas nos discos de inércia. Nas Figuras 17 e 18 são apresentadas respectivamente, as falhas nos rolamentos criadas pelos autores, com cada número representando um tipo de falha apresentada na Tabela 9 e os sinais exemplos para cada classe no domínio do tempo, com a amplitude estando em volts.

Figura 17 – Falhas criadas pelos autores do segundo banco de dados



Fonte: Retirado de (CHEGINI *et al.*, 2022).

Figura 18 – Exemplo de sinal de vibração para cada classe



Fonte: Elaborado pelo autor (2023).

Observando a Figura 18 é possível perceber as diferenças entre os sinais de cada tipo de classe. Para conseguir uma boa classificação destes dados, se torna necessário extrair características deles, nas quais, consigam refletir essas diferenças. É por este motivo que se emprega o uso do RMS, da curtose e da assimetria.

### 7.3 Terceiro banco de dados

O terceiro banco contém dois conjuntos de dados, um com sinais de corrente e outro com sinais de vibração, de três motores síncronos de ímãs permanentes que operam em condições saudáveis, com curto-circuito nas bobinas ou com curto-circuito nas espiras. As falhas são adicionadas experimentalmente empregando o uso de resistências *bypass* colocadas entre as bobinas e as espiras (JUNG *et al.*, 2023).

Os motores possuem diferentes potências (1 kW, 1,5 kW e 3 kW), sendo testados em uma mesma configuração experimental com uma velocidade de rotação de 3000 rpm, uma carga com 15% do limite de torque permitido de 1,5 Nm e com frequências de amostragem de 100

kHz para as medições de corrente e 25,6 kHz para a vibração (JUNG *et al.*, 2023). Na Tabela 10 são apresentados as especificações dos motores empregados.

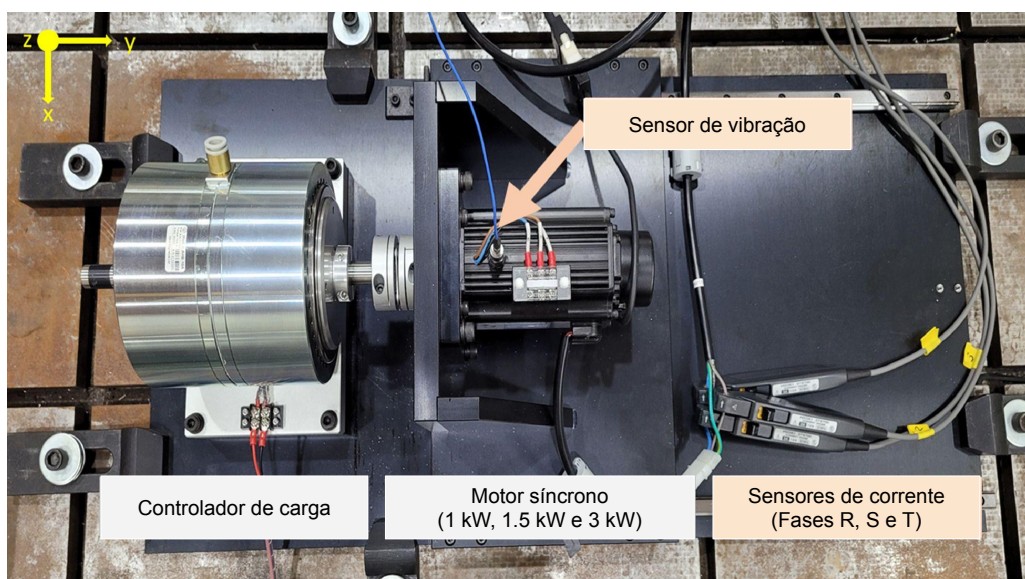
Tabela 10 – Principais parâmetros dos motores síncronos de ímãs permanentes utilizados

Parâmetros	Motor (1 kW)	Motor (1,5 kW)	Motor (3 kW)
Fabricante	Higen motors	Higen motors	Higen motors
Potência (W)	1000	1500	3000
Tensão de entrada (V)	380	380	380
Frequência (Hz)	60	60	60
Número de fases	3	3	3
Número de polos	4	4	4
Torque nominal (Nm)	3,18	4,77	9,55
Velocidade nominal (RPM)	3000	3000	3000
Fluxo magnético (mT)	400	350	300
Inércia do rotor ( $kgm^2$ )	2,07	7,48	14,34
Valor da resistência entre espiras (Ohm)	0,1385	0,0958	0,1087
Valor da resistência entre bobinas (Ohm)	0,0409	0,3021	0,1534

Fonte: Adaptado de (JUNG *et al.*, 2023).

Para a coleta de dados de vibração foram utilizados um acelerômetro (PCB352C34) e um módulo NI9234 por 120 segundos. As medidas de vibração se encontram apenas na direção z. Já para os dados de corrente foram usados três sensores CT6700 (um para cada fase do motor) da fabricante Hioki e também um módulo NI9234 por 120 segundos (JUNG *et al.*, 2023). Nas Figuras 19, 20 e 21 são mostradas as configurações dos experimento e as formas de como as falhas foram introduzidas nos motores.

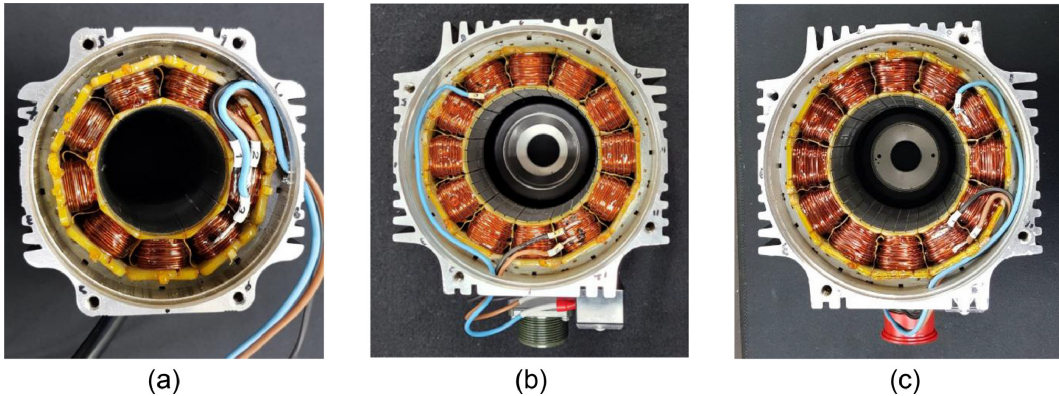
Figura 19 – Bancada experimental do terceiro banco de dados



Fonte: Adaptado de (JUNG *et al.*, 2023).

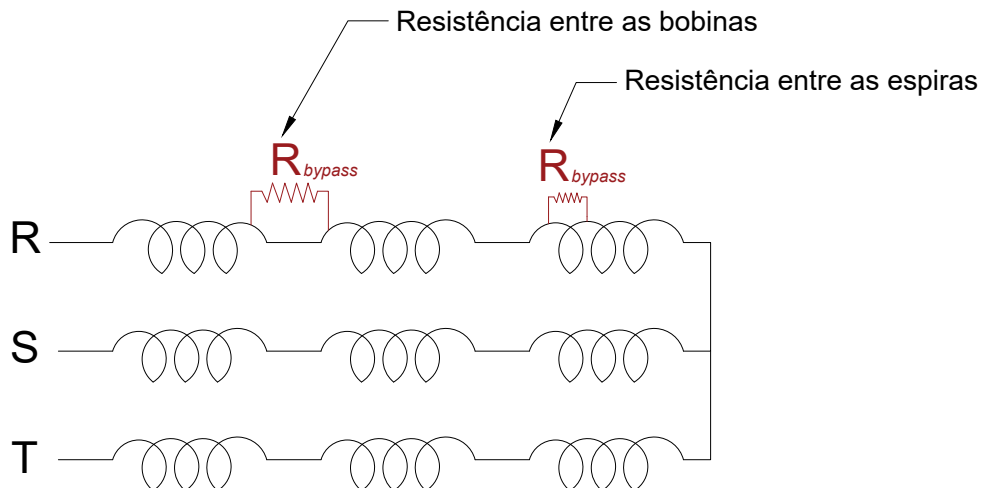


Figura 20 – Falha experimental colocada na fase R do motor síncrono com diferentes potências: (a) 1,0 kW, (b) 1,5 kW e (c) 3,0 kW



Fonte: Retirado de (JUNG *et al.*, 2023).

Figura 21 – Esquema das falhas geradas nos enrolamentos do estator



Fonte: Elaborado pelo autor (2023).

Os dados de vibração são compostos por 48 sinais retirados do acelerômetro na direção  $z$ , sendo estes, 6 com as máquinas em condição de operação saudável, 21 contendo curto-circuito nas espiras e 21 com curto-circuito nas bobinas. Já os dados de corrente são formados por um conjunto de sinais das fases R, S e T. Cada um deles, possui 48 sinais e a mesma distribuição dos dados de vibração.

#### 7.4 Computador e bibliotecas

O computador usado para a criação do modelo foi um Dell com processador intel(R) Core(TM) i5-7200U, 2,5 GHz/4 Gb RAM, Sistema Operacional Windows 10 x64. Os algoritmos

foram desenvolvidos usando a linguagem de programação python (versão 3.8) e as bibliotecas scikit-learn (PEDREGOSA *et al.*, 2011), scipy (VIRTANEN *et al.*, 2020), pandas (TEAM, 2020), numpy (HARRIS *et al.*, 2020), lightGBM (KE *et al.*, 2017b), deslib (CRUZ *et al.*, 2020) e ooptimizer (ROSA *et al.*, 2019).

## 7.5 Pré-processamento dos bancos de dados

No primeiro banco de dados, cada conjunto de sinais, por possuírem muitos elementos (250000), tiveram que passar por um processo de extração de características, no qual, empregando o uso do RMS, foi decomposto em apenas um vetor de amostras. Além disso, foi retirado o atributo do tempo, por não representar um recurso importante nesta aplicação. Em casos, que o estudo fosse focado na análise de séries temporais, este recurso se tornaria relevante. Ao final do processo, o banco de dados ficou com 2162 elementos e 7 atributos.

Já no segundo banco de dados, cada um dos 178 sinais passaram pela etapa de extração de características, foi empregado o uso do RMS, da curtose e da assimetria. Desta forma, o banco de dados gerado ficou com 178 amostras e com três atributos.

O terceiro banco de dados, como já mencionado, é formado por sinais de vibração e corrente a uma frequência amostragem de 25,6 kHz e 100 kHz respectivamente. Os dados foram trabalhados de forma independente para que se pudesse comparar posteriormente o diagnóstico adquirido pelos dois. Para conseguir mais amostras do banco de dados, cada sinal foi dividido em 100 partes, totalizando 4800 sinais para cada forma de coleta.

Dos 4800, 2100 são de falhas de curto-circuito nas espiras, 2100 são de falhas de curto-circuito nas bobinas e 600 são em condições de operação normal. Cada sinal também passou por um processo de extração de características, para os dados de vibração, empregou-se o RMS, assimetria e curtose. Já nas fases da corrente utilizou-se apenas o RMS. Em todos os bancos de dados os sinais foram trabalhados no domínio do tempo.

## 7.6 Criação do modelo

O modelo de aprendizado de máquina criado é um classificador combinado por votação de média ponderada, composto por MLP, SVM,  $k$ -NN, FA, GB, LightGBM. No processo de votação, como apresentado na seção 5.4, é feita uma média ponderada das probabilidades de cada modelo para as classes de diagnóstico. Essas técnicas foram escolhidas pela heterogeneidade,

diversidade e independência em que elas realizam o processo de classificação de dados, tornando assim, o modelo de combinação mais propensos a resultados maiores de acurácia. No trabalho de Lorbieski (2014) é abordado sobre a diversidade dos modelos e o impacto deste no desempenho do classificador combinado.

Para a criação do algoritmo foi realizado uma divisão no banco de dados de 80% para treinamento e validação, e 20% para teste. Nos dados de treinamento e validação são realizados três processos: treinamento dos classificadores formadores, definição dos hiperparâmetros e dos pesos para cada classificador.

Para a definição dos hiperparâmetros foi aplicado uma busca randômica. Os hiperparâmetros são, por exemplo: na MLP (número de neurônios, número de camadas, função de otimização), e no  $k$ -NN (número de vizinhos, cálculo utilizado para a distância).

Para a realização desse processo, um número de 100 iterações foi escolhido. A cada iteração, dentro de um espaço de busca, são definidos aleatoriamente os hiperparâmetros e um processo de validação cruzada de 5 dobras (*folds*) é realizado para cada classificador formador.

Ao final do processo, cada classificador formador possuirá uma iteração, na qual, o conjunto de hiperparâmetros escolhidos foram os que obtiveram o melhor resultado de acurácia média no processo de validação cruzada. A Figura 22 mostra como esse processo ocorre.

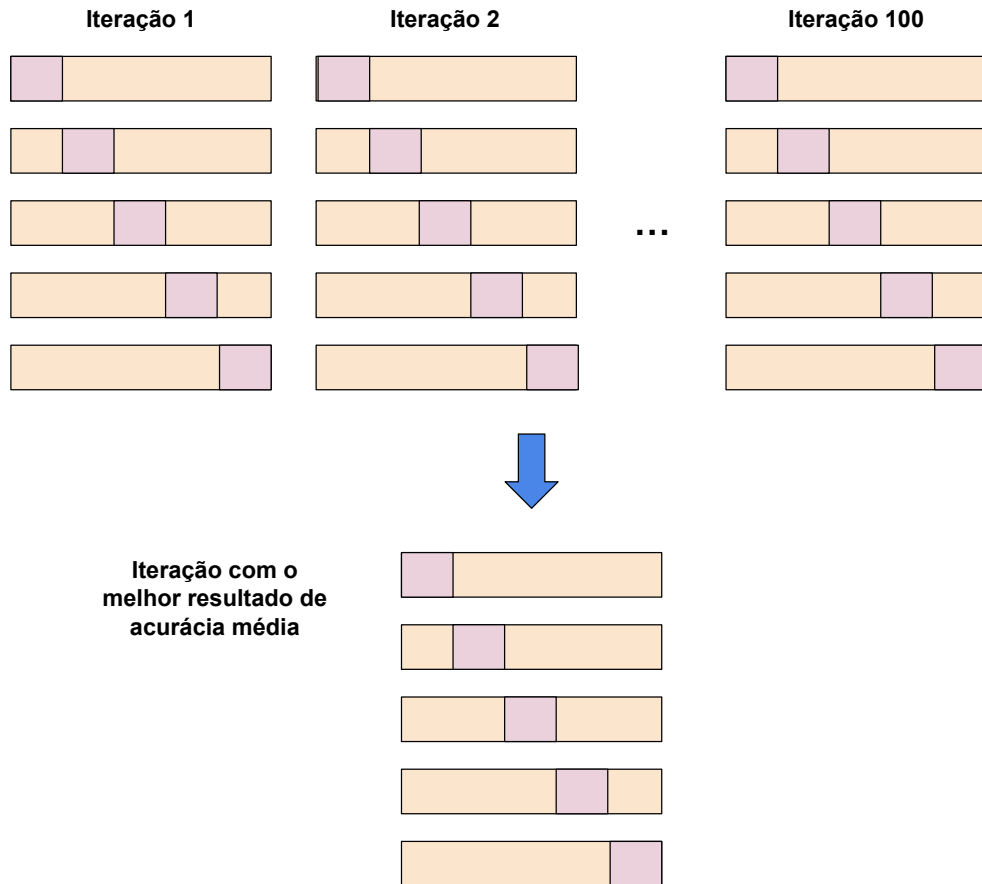
Na Tabela 11 é apresentado os principais hiperparâmetros e o espaço de busca utilizado nos modelos. Outros parâmetros receberam os valores padrões da biblioteca sklearn.

Tabela 11 – Hiperparâmetros e espaço de busca para cada classificador

Modelos	Hiperparâmetros	Espaço de busca
LightGBM	Número de estimadores	[500 : 1500]
	Profundidade	[1 : 500]
	Taxa de aprendizagem	[1e-3,1e-2,1e-1]
	Número de folhas	[2 : 400]
SVM	C	[100,1000,5000,10000,12500,15000,20000,25000]
	Kernel	[RBF, polinomial, linear]
	gamma	[scale, auto]
GB	Número de estimadores	[100 : 1000]
	Valor min de amostras na divisão do nó	[2,3,4,5]
	Profundidade	[10 : 200]
	Valor min. de amostras na divisão da folha	[1,2,3,4]
MLP	Número de interações	[100 : 2000]
	Função de ativação	[TanH, ReLU, sigmoidal]
	Otimização do peso	[LBFGS, SGD, Adam]
k-NN	Número de vizinhos	[1 : 500]
	Distância	[euclidiana, minkowski, manhantan]
FA	Número de estimadores	[100 : 1000]
	Profundidade	[10 : 200]
	Critério de impureza	[gini, entropia]

Fonte: Elaborado pelo autor (2023).

Figura 22 – Exemplo do processo para definição dos hiperparâmetros dos classificadores formadores



Fonte: Elaborado pelo autor (2023).

A busca randômica foi empregada nesse caso por ser relativamente mais rápida que outros otimizadores e por encontrar resultados razoáveis nesse tipo de aplicação. No caso da MLP o número de neurônios e de camadas foi definido experimentalmente. O espaço de busca foi definido também de forma experimental.

No SVM o  $\gamma$  pode ser calculado como mostrado nas Equações 33 e 34 (BUITINCK *et al.*, 2013):

$$auto = \frac{1}{n^{\circ} \text{ de atributos}} \quad (33)$$

$$scale = \frac{1}{n^{\circ} \text{ de atributos} * X.var()} \quad (34)$$

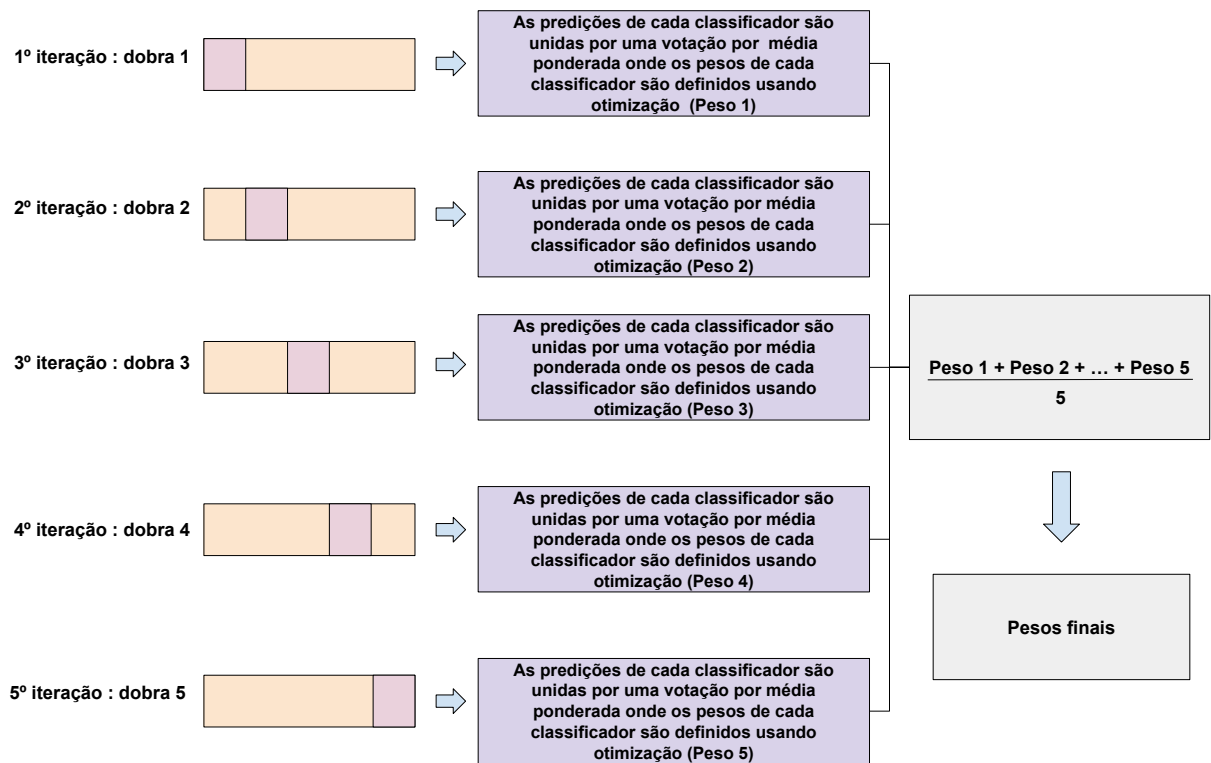
onde  $X.var()$  representa a variância das amostras de entrada.

Com os hiperparâmetros selecionados e empregados em cada classificador formador, uma etapa de validação cruzada com 5 dobras ocorre. A cada iteração da validação, as predições

dos classificadores são juntadas por meio de média ponderada, na qual, os valores dos pesos para cada classificador formador são definidos usando técnicas de otimização.

O espaço de busca definido para os otimizadores são valores entre 0 e 1. Esse intervalo foi escolhido como forma de dar uma nota ao classificador, quanto mais próximo de 1, maior a sua importância. Cada dobra da validação gerará um conjunto de pesos (peso 1,...,peso 5), que posteriormente, por meio de uma média aritmética, darão os pesos finais para cada classificador formador. A Figura 23 mostra como os pesos são definidos no processo de validação cruzada.

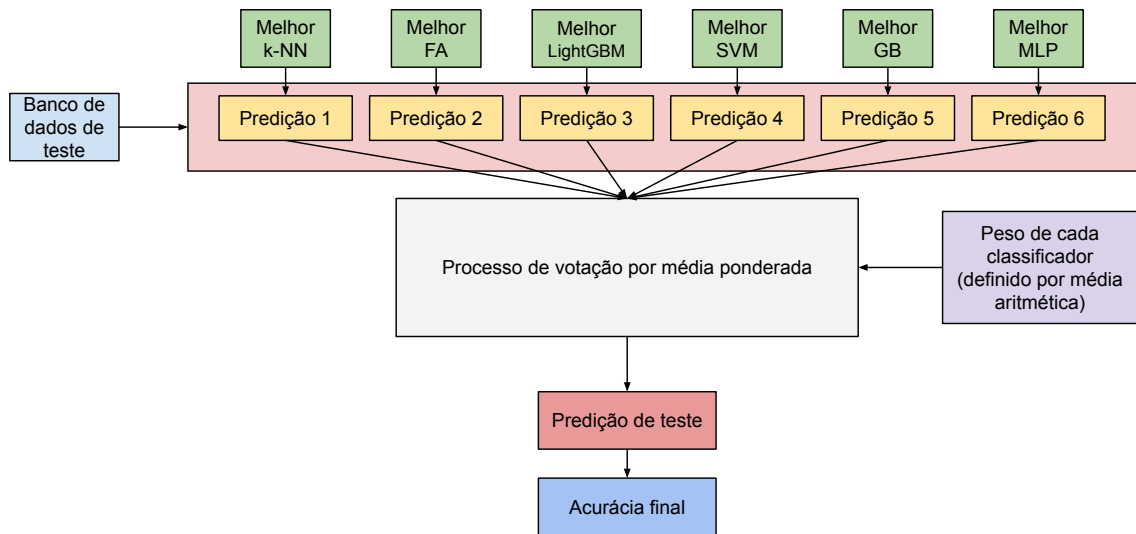
Figura 23 – Definição dos pesos para cada classificador formador



Fonte: Elaborado pelo autor (2023).

Para testar o modelo é utilizado os 20% do banco de dados retirado no início do processo. Para fazer isso são empregados os melhores classificadores (maior resultado de acurácia) encontrados nas dobras da validação cruzada feita anteriormente. Posteriormente, os modelos criados fazem as previsões para as amostras de teste. As previsões dos modelos são juntadas utilizando os pesos selecionados no processo de validação. Por fim, a predição final para cada amostra de teste é comparada com os rótulos reais da classe de teste e desta forma consegue-se os resultados do processo. A Figura 24 mostra como é feito o teste final do modelo.

Figura 24 – Teste do modelo



Fonte: Elaborado pelo autor (2023).

Essa metodologia é utilizada apenas nos classificadores que empregam a votação por média ponderada e a criação do algoritmo é implementado em cada banco de dados separadamente.

## 7.7 Considerações parciais do capítulo 7

O capítulo apresentou os bancos de dados utilizados, mostrando como foram feitas as coletas, quais sensores foram empregados e as condições de operação aplicadas as máquinas. Além disso, foi apresentado uma descrição do computador manuseado para realizar as simulações, bem como a linguagem de programação e as bibliotecas.

Também, foi mostrado a metodologia de criação do algoritmo, apresentando, desde da etapa de pré-processamento até o processo de validação e busca dos melhores hiperparâmetros e pesos.

Essa metodologia será empregada no capítulo de resultados. Nele, são feitas comparações entre o classificador que emprega o uso do RPSO e outros tipos de classificadores. Além disso, é observado a aplicabilidade das análises de corrente e de vibração no diagnóstico de falhas de curto-circuito em bobinas e espiras de máquinas síncronas de ímã permanente. Também é feito uma comparação com outros trabalhos que propuseram soluções para o diagnóstico de falhas em MER.

## 8 RESULTADOS

Neste capítulo são apresentados os resultados e discussões sobre as metodologias desenvolvidas para o diagnóstico de falhas simples e combinadas nas MER. Como já mencionado no capítulo 7 foram desenvolvidos classificadores para cada um dos três bancos de dados.

Primeiramente são apresentados os hiperparâmetros e pesos definidos. Posteriormente é feita uma análise dos resultados obtidos observando os valores de acurácia média e desvio padrão da etapa de validação. Por fim, são mostrados os resultados encontrados na etapa de teste.

Neste trabalho, MLP, SVM,  $k$ -NN, FA, GB e LightGBM são empregados para formar os classificadores *ensemble*. Para efeito de comparação com o classificador combinado por votação ponderada usando o RPSO, as técnicas KNORA-U, KNORA-E, empilhamento e votação por média simples são utilizadas. Além disso, é feito um comparativo entre técnicas de otimização empregadas para a definição dos pesos, nas quais são: COBYLA, Powell, AG e PSO.

Nos classificadores *ensemble* (KNORA-U, KNORA-E e empilhamento) foram feitos uma validação cruzada com 5 dobras no banco de dados de treinamento. Posteriormente o melhor modelo encontrado era utilizado no banco de dados de teste. Ao final do capítulo é realizado um comparativo com outras soluções propostas na literatura para o diagnóstico de falhas nas MER. Por fim, na seção 8.5, são feitas considerações sobre os resultados.

Neste trabalho não foi empregado o uso de dados sintéticos, como mencionado anteriormente, o uso destes pode ocasionar em erros de má generalização do modelo.

### 8.1 Resultados do primeiro banco de dados

Como mencionado do capítulo 7, o banco de dados é dividido em 80% para treinamento e validação, e 20% para teste. Com o banco de dados de treinamento e validação são escolhidos os valores dos hiperparâmetros e os pesos para cada classificador. Na Tabela 12 é apresentado os hiperparâmetros dos classificadores formadores.

Com os valores dos hiperparâmetros definidos é feito o processo de definição dos pesos dos classificadores para a formação do *ensemble*. Como mencionado na seção 7.6, é feita uma média aritmética entre os pesos de cada dobra da validação, como foi mostrado na Figura 23.

Para a criação do empilhamento foi empregado experimentalmente como meta modelo uma MLP. Os hiperparâmetros dela foram de uma camada oculta com 100 neurônios, um número de iterações de 200, a função de ativação foi a tangente hiperbólica e para a otimização do peso foi usado o Adam. O KNORA-U e KNORA-E conservaram os valores padrões da biblioteca deslib.

Tabela 12 – Hiperparâmetros dos classificadores formadores para o primeiro banco de dados

Modelos	Hiperparâmetros	Valor do hiperparâmetro
LightGBM	Número de estimadores	1243
	Profundidade	271
	Taxa de aprendizagem	0,1
	Número de folhas	207
SVM	C	20000
	Kernel	RBF
	gamma	scale
GB	Número de estimadores	300
	Valor min de amostras na divisão do nó	3
	Profundidade	48
	Valor min. de amostras na divisão da folha	4
MLP	Número de camadas ocultas	2
	Neurônios presentes em cada camada oculta	100
	Número de interações	1651
	Função de ativação	TanH
	Otimização do peso	Adam
<i>k</i> -NN	Número de vizinhos	1
	Distância	manhattan
FA	Número de estimadores	300
	Profundidade	105
	Critério de impureza	entropia

Fonte: Elaborado pelo autor (2023).

Os otimizadores foram definidos para realizar o processo em 100 iterações. A AG foi criada com um tamanho de população de 50 indivíduos, com uma probabilidade de seleção de 0,75, de mutação de 0,25 e de cruzamento de 0,5. Já o PSO, foi inicializado com 50 partículas, ficou com uma ponderação de inércia  $w$  de 0,7, com parâmetro cognitivo  $c_1$  de 1,7 e com parâmetro social  $c_2$  de 1,7. Por fim, o RPSO, também foi inicializado com 50 partículas, teve também seus parâmetros cognitivo e sociais com valores de 1,7. As escolhas dos parâmetros foram feitas experimentalmente. Na Tabela 13 são mostrados os valores dos pesos médios definidos por cada otimizador.

Observando os pesos definidos pelos otimizadores na Tabela 13 pode-se observar que na maioria deles, os classificadores SVM e MLP conseguiram valores de pesos maiores que os demais. Isso provavelmente demonstra que em casos mais difíceis de se rotular, essas duas técnicas estão sendo mais relevantes. A explicação para isso é devido à capacidade dessas



técnicas de adquirir conhecimento sobre as relações não lineares entre a entrada e saída do banco de dados.

Tabela 13 – Valores dos pesos médios definidos por cada otimizador no primeiro banco de dados

Otimizadores	Peso LightGBM	Peso SVM	Peso GB	Peso MLP	Peso <i>k</i> -NN	Peso FA
COBYLA	0,1843	<b>0,6225</b>	0,0504	0,4967	0,1294	0,3088
Powell	0,5213	<b>0,6477</b>	0,5041	0,6414	0,6023	0,6229
AG	0,2422	0,3885	0,0696	<b>0,8644</b>	0,4819	0,1748
PSO	0,2555	<b>0,6973</b>	0,0759	0,6441	0,4764	0,1717
RPSO	0,1867	0,4402	0,0940	<b>0,7793</b>	0,3456	0,1969

Fonte: Elaborado pelo autor (2023).

Já o GB e a FA ficaram com valores de pesos pequenos em relação aos demais. Este fato pode mostrar uma dificuldade das técnicas baseadas em árvores na classificação desse conjunto de dados. No caso da GB é ainda pior porque a maioria dos pesos ficaram com valores bem próximos de 0. De todos os otimizadores, o Powell foi o que mais se distanciou do padrão de valores encontrados pelos outros, possuindo resultados que apresentaram pouca variação.

A metodologia de pesos é interessante porque faz uma seleção automática dos classificadores para um determinado banco de dados. Com isso, os modelos que se adaptam melhores ao banco de dados recebem pesos maiores que os demais. Além disso, podemos supor um caso em que determinado classificador está sofrendo de má generalização e por consequência errando em determinadas amostras do banco de dados. A classificação por pesos pode retirar a relevância desse modelo de classificação e dessa forma reduzir a má generalização geral do *ensemble*.

Na Tabela 14 são apresentados os resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de validação. O desvio padrão é apresentado como forma de demonstrar a distância dos valores de acurácia em relação à média da validação cruzada de cada classificador.

Como pode ser analisado na Tabela 14, os classificadores que formam o *ensemble* ficaram com resultando de acurácia média em torno de 81%, podendo chegar até 83,40% com a MLP. Já o SVM ficou com o pior resultado, conseguindo apenas 79,17%. Já os classificadores combinados usando as técnicas dinâmicas KNORA-U e KNORA-E, tiveram seus resultados em torno de 84%. O empilhamento e a votação por média simples conseguiram também uma acurácia em torno de 84%.

O aumento nos resultados evidência a capacidade dos algoritmos de combinação de

modelos, que com suas "indicações" diversificadas, conseguem classificar melhor as amostras. Os resultados aumentam, quando é empregado o uso da votação por média ponderada. Nele, cada otimizador conseguiu encontrar valores de pesos médios que possibilitam aumentar ainda mais os resultados de acurácia. Isso ocorre porque, como mencionado anteriormente, um processo de seleção dos classificadores é realizado, dando mais peso para aquele que acerta e menos, para o que erra.

Tabela 14 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o primeiro banco de dados

Modelo	Acurácia média (%)	Desvio padrão
LightGBM	80,79	0,0119
SVM	79,17	0,0146
GB	80,85	0,0128
MLP	83,40	0,0106
<i>k</i> -NN	82,47	0,0181
FA	80,39	0,0105
KNORA-U	84,49	0,0160
KNORA-E	84,09	0,0135
Empilhamento	84,38	0,0137
Votação por média simples	84,61	0,0124
Votação por média ponderada + COBYLA	86,29	0,0128
Votação por média ponderada + Powell	85,13	0,0086
Votação por média ponderada + AG	<b>86,75</b>	0,0088
Votação por média ponderada + PSO	86,40	0,0075
Votação por média ponderada + RPSO	86,69	0,0097

Fonte: Elaborado pelo autor (2023).

Como pode ser observado, essas metodologias conseguiram resultados em torno de 86%. Além disso, o desvio padrão também diminuiu com o uso dos classificadores em combinação com as meta-heurísticas.

Com os classificadores treinados e validados, pode-se realizar o teste final. Para isso, como apresentado na seção 7.6, os melhores classificadores formadores da validação são empregados para formar os *ensembles*. Os 20% do banco de dados retirado inicialmente, é empregado no teste final. Na Tabela 15 é explanado os resultados de acurácia final de todos os classificadores e combinações.

Como apresentado na Tabela 15, a técnica proposta conseguiu o melhor resultado de acurácia, 88,68%, mostrando a eficiência do modelo na classificação de falhas nas MER. Outros modelos como a votação por média ponderada com AG e PSO conseguiram resultado bem próximos, nos quais, também possibilitam sua utilização no problema.

Tabela 15 – Resultados de acurácia final para cada classificador para o primeiro banco de dados

Modelo	Acurácia (%)
LightGBM	79,90
SVM	79,90
GB	79,21
MLP	84,98
<i>k</i> -NN	82,44
FA	79,67
KNORA-U	84,75
KNORA-E	85,45
Empilhamento	86,37
Votação por média simples	84,29
Votação por média ponderada + COBYLA	86,83
Votação por média ponderada + Powell	85,45
Votação por média ponderada + AG	88,22
Votação por média ponderada + PSO	87,75
Votação por média ponderada + RPSO	<b>88,68</b>

Fonte: Elaborado pelo autor (2023).

O RPSO emprega o uso de fenômenos relativísticos para conseguir bons resultados na otimização. Além disso, o uso de seis modelos para combinação, aumenta a complexidade da otimização, o que também pode explicar a vantagem deste método, que acaba não se prendendo a resultados mínimos locais. Na Tabela 16 são mostradas outras métricas de avaliação de teste por classe para o classificador combinado por média ponderada com o uso do RPSO.

Tabela 16 – Sensibilidade, precisão e *F-score* do classificador combinado por média ponderada com o uso do RPSO para o primeiro banco de dados

Classes	Precisão	Sensibilidade	<i>F-score</i>
Desalinhamento vertical + desalinhamento horizontal	92%	93%	92%
Desbalanceamento + desalinhamento vertical	87 %	89%	88%
Desbalanceamento + desalinhamento horizontal	76 %	86%	81%
Desalinhamento horizontal	91 %	89 %	90%
Desalinhamento vertical	97 %	97 %	97%
Desbalanceamento	87%	76 %	81%
Normal	100%	98%	99%

Fonte: Elaborado pelo autor (2023).

Na Tabela 16 é possível observar os melhores resultados encontrados pelo classificador ao rotular a classe normal e desalinhamento vertical. O resultado na classe normal ainda evidência a capacidade do classificador de conseguir definir se os equipamentos estão em falha ou não.

A maior taxa de erros se encontra na classe de desbalanceamento que conseguiu um resultado de sensibilidade de 76%, o que mostra que o classificador está confundindo as amostras encontradas nessa classe, com outra. A matriz de confusão de teste do classificador

combinado por média ponderada usando RPSO é apresentado na Tabela 17.

Tabela 17 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o primeiro banco de dados

Classes	a	b	c	d	e	f	g
a = Desalinhamento vertical + desalinhamento horizontal	<b>80</b>	3	2	1	0	0	0
b = Desbalanceamento + desalinhamento vertical	0	<b>62</b>	5	0	0	3	0
c = Desbalanceamento + desalinhamento horizontal	1	3	<b>65</b>	2	0	5	0
d = Desalinhamento horizontal	3	1	1	<b>41</b>	0	0	0
e = Desalinhamento vertical	1	0	0	0	<b>36</b>	5	0
f = Desbalanceamento	2	2	12	0	1	<b>53</b>	0
g = Normal	0	0	0	1	0	0	<b>47</b>

Fonte: Elaborado pelo autor (2023).

Já observando a matriz de confusão (Tabela 17) é possível observar diretamente onde estão acontecendo os erros. O classificador está se confundindo entre a classe de desbalanceamento e o conjunto de desbalanceamento com desalinhamento horizontal. Já na classe normal só houve um erro. Também é possível perceber que os erros de predições se concentram nas falhas combinadas, mostrando a dificuldade do classificador de diferenciar falhas simples de falhas compostas.

## 8.2 Resultados do segundo banco de dados

O banco de dados também é dividido em treinamento, validação e teste. Com o banco de dados de treinamento e validação foi definido os parâmetros para os classificadores. Como no caso anterior, o KNORA-U e KNORA-E também conservaram os valores padrões de hiperparâmetros da biblioteca deslib. Na tabela 18 são mostrados os hiperparâmetros escolhidos.

Neste banco de dados o meta modelo do empilhamento também foi uma MLP que possui apenas uma camada oculta com 100 neurônios. A função de ativação foi a tangente hiperbólica, sendo feito em 150 iterações. Além disso, para a otimização das conexões dos pesos foi empregado o SGD. Os parâmetros foram definidos experimentalmente.

Já os otimizadores fizeram a busca em 100 iterações. A AG foi desenvolvida com uma população de 30 indivíduos, possuindo uma probabilidade de seleção, mutação e de cruzamento de 0,70, 0,30 e 0,50, respectivamente. O PSO, foi criado com 30 partículas, um  $w$  de 0,8 e com um  $c_1$  e  $c_2$  de 1,5. Para terminar, o RPSO foi concebido também com 30 partículas, teve seu  $c_1$  e  $c_2$  definido com um valor de 1,6. Os valores dos pesos médios definidos por cada otimizador são apresentados na Tabela 19.

Tabela 18 – Hiperparâmetros dos classificadores formadores para o segundo banco de dados

Modelos	Hiperparâmetros	Valor do hiperparâmetro
LightGBM	Número de estimadores	807
	Profundidade	65
	Taxa de aprendizagem	0,1
	Número de folhas	359
SVM	C	60
	Kernel	RBF
	gamma	auto
GB	Número de estimadores	200
	Valor min de amostras na divisão do nó	2
	Profundidade	48
	Valor min. de amostras na divisão da folha	2
MLP	Número de camadas ocultas	1
	Neurônios presentes em cada camada oculta	100
	Número de interações	1807
	Função de ativação	TanH
	Otimização do peso	Adam
<i>k</i> -NN	Número de vizinhos	1
	Distância	manhattan
FA	Número de estimadores	658
	Profundidade	23
	Critério de impureza	gini

Fonte: Elaborado pelo autor (2023).

Tabela 19 – Valores dos pesos médios definidos por cada otimizador para o segundo banco de dados

Otimizadores	Peso LightGBM	Peso SVM	Peso GB	Peso MLP	Peso <i>k</i> -NN	Peso FA
COBYLA	0,3000	0,1498	0,1	<b>0,3012</b>	0,0960	0,3000
Powell	<b>1</b>	<b>1</b>	0,8624	<b>1</b>	0,7350	<b>1</b>
AG	0,5495	0,5639	0,2035	0,5172	0,2452	<b>0,7204</b>
PSO	<b>0,6595</b>	0,2498	0,2029	0,4260	0,3350	0,4513
RPSO	0,4814	0,4547	0,4886	0,4828	0,3075	<b>0,7517</b>

Fonte: Elaborado pelo autor (2023).

Neste banco de dados os pesos foram maiores nas técnicas que empregam árvores de decisão em sua composição (FA e LightGBM), mostrando que esses algoritmos são importantes na criação do *ensemble*. Observado a Tabela 19 os pesos ficaram mais equilibrados, tendo o GB e o *k*-NN como os piores resultados gerais. Os resultados de acurácia média e o desvio padrão dos classificadores na etapa de treino e validação são apresentados na Tabela 20.

Os resultados de validação dos classificadores formadores apresentados na Tabela 20 mostram uma grande diferença de acurácia, representando que alguns modelos com o *k*-NN e o SVM não conseguiram classificar bem os dados. Além disso, observa-se bom resultados nos classificadores baseados em árvore, evidenciando assim o porquê de possuírem pesos maiores que os demais, como apresentado na Tabela 19.

Também é possível observar, com o uso do desvio padrão, uma grande variação nos

valores de acurácia no processo de validação. Isso aconteceu devido ao fato que o banco de dados é muito pequeno e apenas um erro pode diminuir muito o resultado de acurácia de uma dobra da validação. O KNORA-U ficou com um resultado abaixo do esperado, tendo resultados menores que a FA. Já os outros classificadores ficaram com resultados entre 94% e 95%. O classificador combinado utilizando o RPSO conseguiu o melhor resultado, empatado com o COBYLA e o PSO.

Tabela 20 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o segundo banco de dados

Modelo	Acurácia média (%)	Desvio padrão
LightGBM	92,26	0,0130
SVM	76,08	0,0447
GB	93,67	0,0410
MLP	88,02	0,0754
<i>k</i> -NN	80,32	0,0515
FA	94,38	0,0355
KNORA-U	93,67	0,0343
KNORA-E	94,38	0,0421
Empilhamento	94,40	0,0468
Votação por média simples	94,38	0,0421
Votação por média ponderada + COBYLA	<b>95,09</b>	0,0353
Votação por média ponderada + Powell	94,38	0,0421
Votação por média ponderada + AG	94,38	0,0421
Votação por média ponderada + PSO	<b>95,09</b>	0,0353
Votação por média ponderada + RPSO	<b>95,09</b>	0,0353

Fonte: Elaborado pelo autor (2023).

Com o processo de validação sendo finalizado, são empregados os 20% restante do banco de dados para o teste final dos classificadores. Na Tabela 21 é explanado os resultados de acurácia final de todos os classificadores.

Como pode ser observado na Tabela 21, os resultados de acurácia final foram ótimos, com alguns classificadores formadores conseguindo um resultado de acurácia de 97,22%, errando apenas um elemento dos dados de teste. Já os outros classificadores conseguiram 100% na classificação.

Esses resultados mostram que nesse banco de dados em específico não haveria a necessidade de empregar um método sofisticado, bastando, por exemplo, apenas o uso da FA. Mesmo assim é interessante a aplicação destas técnicas nesse banco de dados, isso porque ele apresenta poucos elementos (mostrando a aplicabilidade das técnicas neste tipo de banco de dados), também os sinais de vibração são retirados usando apenas um acelerômetro (o que mostra que com poucas informações é possível conseguir êxito na classificação) e ainda possui várias

condições de defeitos em rolamentos com o emprego de uma máquina de indução (classificação de várias condições de operação).

Tabela 21 – Resultados de acurácia final para cada classificador para o segundo banco de dados

Modelo	Acurácia (%)
LightGBM	91,66
SVM	77,77
GB	97,22
MLP	94,44
<i>k</i> -NN	97,22
FA	97,22
KNORA-U	<b>100</b>
KNORA-E	<b>100</b>
Empilhamento	<b>100</b>
Votação por média simples	<b>100</b>
Votação por média ponderada + COBYLA	<b>100</b>
Votação por média ponderada + Powell	<b>100</b>
Votação por média ponderada + AG	<b>100</b>
Votação por média ponderada + PSO	<b>100</b>
Votação por média ponderada + RPSO	<b>100</b>

Fonte: Elaborado pelo autor (2023).

As outras métricas de avaliação de teste por classe do classificador combinado por média ponderada com o uso do RPSO são apresentadas na Tabela 22. Devido ao espaço, foram utilizadas as designação apresentadas na Tabela 9 para representar os tipos de falhas desse banco de dados. Como não houve erros, todas as outras métricas apresentaram um resultado de 100%.

Tabela 22 – Sensibilidade, precisão e *F-score* do classificador combinado por média ponderada com o uso do RPSO para o segundo banco de dados

Classes	Precisão	Sensibilidade	<i>F-score</i>
1	100%	100%	100%
2	100%	100%	100%
3	100%	100%	100%
4	100%	100%	100%
5	100%	100%	100%
6	100%	100%	100%
7	100%	100%	100%
8	100%	100%	100%
9	100%	100%	100%

Fonte: Elaborado pelo autor (2023).

Na Tabela 23 é mostrado a matriz de confusão de teste do classificador combinado por média ponderada usando RPSO.

Tabela 23 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o segundo banco de dados

Classes	a	b	c	d	e	f	g	h	i
a = 1	<b>5</b>	0	0	0	0	0	0	0	0
b = 2	0	<b>3</b>	0	0	0	0	0	0	0
c = 3	0	0	<b>6</b>	0	0	0	0	0	0
d = 4	0	0	0	<b>4</b>	0	0	0	0	0
e = 5	0	0	0	0	<b>4</b>	0	0	0	0
f = 6	0	0	0	0	0	<b>4</b>	0	0	0
g = 7	0	0	0	0	0	0	<b>4</b>	0	0
h = 8	0	0	0	0	0	0	0	<b>3</b>	0
i = 9	0	0	0	0	0	0	0	0	<b>3</b>

Fonte: Elaborado pelo autor (2023).

Observando a matriz de confusão (Tabela 23) percebe-se o pouco número de amostras testadas, como já mencionado, isso ocorre porque o banco de dados é bem pequeno. Como evidenciado pela acurácia e pelas outras métricas, não houve nenhum erro na classificação.

### 8.3 Resultados do terceiro banco de dados

O terceiro banco de dados é constituído por dois conjuntos de dados (vibração e corrente) para o diagnóstico de falhas de curto-circuito em bobinas e em espiras de máquinas síncronas de ímã permanente. Por este motivo, pode-se comparar o diagnóstico criado pelos dois. Com isso, primeiramente serão mostrados os resultados encontrados pela análise de corrente e depois os desenvolvidos pela análise de vibração.

#### 8.3.1 Diagnóstico por análise de corrente

Depois que banco de dados é dividido em 80% para treinamento e validação, e 20% para teste, é feito o processo de busca pelos melhores hiperparâmetros. Os classificadores dinâmicos KNORA-U e KNORA-E foram deixados com os valores padrões da biblioteca deslib. Na Tabela 24 são apresentados os hiperparâmetros escolhidos para cada classificador formador.

O empilhamento, escolhido empiricamente, foi uma MLP com uma camada oculta contendo 100 neurônios, sendo criada, com 200 iterações, função de ativação sendo a ReLU e para a otimização dos pesos empregou-se o Adam.

Para os otimizadores, a AG foi composta por uma população de 50 indivíduos e com probabilidades de seleção, mutação e cruzamento de 0,7, 0,35 e 0,4, respectivamente. O



PSO ficou com 50 partículas, um  $w$  de 0,8 e um  $c_1$  e  $c_2$  de 1,2. Por fim, o RPSO ficou com os parâmetros de  $c_1$  e  $c_2$  de 1,9, sendo formado também por 50 partículas. Todos os otimizadores foram desenvolvidos em 100 iterações.

Definido os valores dos hiperparâmetros, na próxima etapa é feita a escolha dos pesos para cada classificador formador. Os pesos são mostrados na tabela 25.

Tabela 24 – Hiperparâmetros dos classificadores formadores para o terceiro banco de dados por análise de corrente

Modelos	Hiperparâmetros	Valor do hiperparâmetro
LightGBM	Número de estimadores	525
	Profundidade	362
	Taxa de aprendizagem	0,01
	Número de folhas	331
SVM	C	25000
	Kernel	RBF
	gamma	scale
GB	Número de estimadores	400
	Valor mín de amostras na divisão do nó	4
	Profundidade	124
	Valor mín. de amostras na divisão da folha	3
MLP	Número de camadas ocultas	2
	Neurônios presentes em cada camada oculta	100
	Número de interações	720
	Função de ativação	TanH
	Otimização do peso	LBFGS
$k$ -NN	Número de vizinhos	1
	Distância	manhantan
FA	Número de estimadores	900
	Profundidade	200
	Critério de impureza	entropia

Fonte: Elaborado pelo autor (2023).

Tabela 25 – Valores dos pesos médios definidos por cada otimizador para o terceiro banco de dados por análise de corrente

Otimizadores	Peso LightGBM	Peso SVM	Peso GB	Peso MLP	Peso $k$ -NN	Peso FA
COBYLA	0,2931	0,3409	0,0330	<b>0,5098</b>	0,3194	0,1013
Powell	0,7937	0,6037	0,7249	0,7087	0,6825	<b>0,9837</b>
AG	0,3109	0,7183	0,4282	0,7330	<b>0,7494</b>	0,2952
PSO	0,4203	0,5153	0,3654	0,5355	0,6485	<b>0,6641</b>
RPSO	0,4278	0,4766	0,2979	<b>0,7845</b>	0,6798	0,3538

Fonte: Elaborado pelo autor (2023).

Entre os otimizadores, os pesos foram maiores na MLP e na FA. São apresentados na Tabela 26 os resultados de acurácia e desvio padrão de todos os classificadores na etapa de validação.

Tabela 26 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o terceiro banco de dados por análise de corrente

Modelo	Acurácia média (%)	Desvio padrão
LightGBM	98,77	0,0052
SVM	96,09	0,0063
GB	98,93	0,0037
MLP	99,03	0,0053
<i>k</i> -NN	99,06	0,0026
FA	98,90	0,0026
KNORA-U	99,24	0,0038
KNORA-E	99,21	0,0029
Empilhamento	99,19	0,0044
Votação por média simples	99,08	0,0034
Votação por média ponderada + COBYLA	99,29	0,0029
Votação por média ponderada + Powell	99,14	0,0044
Votação por média ponderada + AG	99,29	0,0031
Votação por média ponderada + PSO	99,29	0,0031
Votação por média ponderada + RPSO	<b>99,34</b>	0,0033

Fonte: Elaborado pelo autor (2023).

Como já indicado na Tabela 25, o MLP e a FA obtiveram bons resultados na classificação, conseguindo respectivamente valores de acurácia de 99,03% e 98,90%. Com o emprego das técnicas de *ensemble*, houve pequenos aumentos nos resultados, culminando em uma acurácia de 99,34%, obtido pelo classificador combinado por média ponderada e com o uso do RPSO.

O desvio padrão também foi bem pequeno, variando apenas na terceira casa decimal, o que mostra uma baixa variabilidade entre os valores de acurácia conseguidos a cada dobra da validação cruzada. Terminado esse processo, é feito o teste final dos classificadores. Na Tabela 27 são mostrados o resultados finais de acurácia.

Os resultados de acurácia final (Tabela 27) foram próximos e os classificadores que empregam a AG e o RPSO conseguiram o melhor resultado de 98,95%. As outras métricas de avaliação de teste para o classificador com o uso do RPSO são apresentados na Tabela 28.

Observando a Tabela 28 vemos bons resultados na classificação das falhas de curto na espira e curto na bobina. Os piores resultados estão na classe normal. Isso provavelmente ocorre pelo desbalanceamento que essa classe sofre em comparação com as outras. Além disso, como possui poucos elementos, cada erro seu diminui ainda mais a porcentagem de classificação. Por fim, a matriz de confusão é apresentada na Tabela 29.

Como já apresentado na Tabela 29, os erros foram maiores na classe normal. O classificador errou 7 vezes, inferindo que a amostra rotulada como normal pertencia à classe de

curto na espira. Já os erros em amostras de outras classes foram pequenas em relação ao número de amostras que foram testadas, o que mostra uma grande aplicabilidade da técnica proposta.

Tabela 27 – Resultados de acurácia final para cada classificador para o terceiro banco de dados por análise de corrente

Modelo	Acurácia (%)
LightGBM	97,60
SVM	95,31
GB	98,33
MLP	98,75
<i>k</i> -NN	98,43
FA	98,64
KNORA-U	98,85
KNORA-E	98,64
Empilhamento	98,75
Votação por média simples	98,75
Votação por média ponderada + COBYLA	98,75
Votação por média ponderada + Powell	98,75
Votação por média ponderada + AG	<b>98,95</b>
Votação por média ponderada + PSO	98,85
Votação por média ponderada + RPSO	<b>98,95</b>

Fonte: Elaborado pelo autor (2023).

Tabela 28 – Sensibilidade, precisão e *F-score* do classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de corrente

Classes	Precisão	Sensibilidade	<i>F-score</i>
Normal	100%	94%	96%
Curto na espira	98 %	99%	98%
Curto na bobina	99 %	99%	99%

Fonte: Elaborado pelo autor (2023).

Tabela 29 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o terceiro banco de dados por análise de corrente

Classes	a	b	c
a = Normal	<b>111</b>	7	0
b = Curto na espira	0	<b>403</b>	2
c = Curto na bobina	0	1	<b>436</b>

Fonte: Elaborado pelo autor (2023).

### 8.3.2 Diagnóstico por análise de vibração

Para efeito de comparação da forma em que é feita a coleta de sinais, também foi elaborado o diagnóstico de falhas de curto-circuito em bobinas e em espiras de uma máquina síncrona de ímã permanente usando a análise de vibração.

As informações sobre estes dados foram comentadas na seção 7.3. Neste caso, serão apresentados apenas os resultados para o classificador combinado por votação ponderada com o uso do RPSO, considerado o melhor classificador nas análises anteriores.

O mesmo processo empregado nos outros bancos de dados é desenvolvido também aqui. Separação das amostras em 80% para treino e validação e 20% para o teste final. Na Tabela 30 são mostrados os hiperparâmetros encontrados para os classificadores formadores.

Tabela 30 – Hiperparâmetros dos classificadores formadores para o terceiro banco de dados por análise de vibração

Modelos	Hiperparâmetros	Valor do hiperparâmetro
LightGBM	Número de estimadores	555
	Profundidade	72
	Taxa de aprendizagem	0,01
	Número de folhas	358
SVM	C	20000
	Kernel	RBF
	gamma	scale
GB	Número de estimadores	600
	Valor min de amostras na divisão do nó	5
	Profundidade	10
	Valor min. de amostras na divisão da folha	1
MLP	Número de camadas ocultas	2
	Neurônios presentes em cada camada oculta	100
	Número de interações	681
	Função de ativação	TanH
	Otimização do peso	LBFSGS
$k$ -NN	Número de vizinhos	28
	Distância	manhattan
FA	Número de estimadores	100
	Profundidade	143
	Critério de impureza	gini

Fonte: Elaborado pelo autor (2023).

Os parâmetros escolhidos para o otimizador por RPSO foram um  $c_1$  e  $c_2$  de 1,5 e possui um número de partículas igual a 50. O algoritmo foi rodado durante 100 iterações. Os pesos encontrados para cada otimizador são apresentados na Tabela 31.

Os pesos foram maiores nos classificadores LightGBM, MLP e FA, já os menores resultados dos pesos foram no GB e no  $k$ -NN. Na Tabela 32 são mostrados os resultados de acurácia média e desvio padrão de validação dos classificadores formadores e do classificador

combinado por média ponderada com o uso do RPSO.

Tabela 31 – Valores dos pesos médios definidos para o classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de vibração

Otimizadores	Peso LightGBM	Peso SVM	Peso GB	Peso MLP	Peso <i>k</i> -NN	Peso FA
RPSO	<b>0,7248</b>	0,2664	0,1753	0,6742	0,2325	0,4852

Fonte: Elaborado pelo autor (2023).

Tabela 32 – Resultados de acurácia média e o desvio padrão de todos os classificadores na etapa de treino e validação para o terceiro banco de dados por análise de vibração

Modelo	Acurácia média (%)	Desvio padrão
LightGBM	90,31	0,0046
SVM	88,85	0,0089
GB	89,32	0,0099
MLP	88,43	0,0113
<i>k</i> -NN	88,17	0,0150
FA	90,20	0,0082
Votação por média ponderada + RPSO	<b>90,54</b>	0,0080

Fonte: Elaborado pelo autor (2023).

Observando a Tabela 32, podemos notar a diminuição dos resultados em comparação com a análise de corrente, evidenciando como apresentado na seção 3.4, a dificuldade que a análise de vibração têm de diagnosticar falhas elétricas. Nas Tabelas 33 e 34 são mostrados as acurácias de teste dos classificadores e as outras métricas de avaliação do classificador combinado por média ponderada com o uso do RPSO respectivamente.

Tabela 33 – Resultados de acurácia final para cada classificador para o terceiro banco de dados por análise de vibração

Modelo	Acurácia (%)
LightGBM	89,27
SVM	88,43
GB	89,16
MLP	88,12
<i>k</i> -NN	87,39
FA	89,89
Votação por média ponderada + RPSO	<b>90,41</b>

Fonte: Elaborado pelo autor (2023).

Os resultados da acurácia de teste (Tabela 33) se mostraram semelhantes quando comparado com os resultados de validação. Além disso, pode-se observar o aumento no valor

de acurácia (90,41%) que o classificador criado possui em comparação com os classificadores formadores.

Tabela 34 – Sensibilidade, precisão e *F-score* do classificador combinado por média ponderada com o uso do RPSO para o terceiro banco de dados por análise de vibração

Classes	Precisão	Sensibilidade	<i>F-score</i>
Normal	100%	99%	99%
Curto na espira	88 %	89%	89%
Curto na bobina	89 %	89%	89%

Fonte: Elaborado pelo autor (2023).

Como pode ser analisado na Tabela 34, a principal confusão do classificador está nas classes com falha, mostrando a dificuldade no diagnóstico. Já na classe Normal, observando a sensibilidade, vemos o maior índice de acertos (99%). Por fim, na Tabela 35 é apresentado a matriz de confusão de teste do classificador criado.

Tabela 35 – Matriz de confusão de teste do classificador combinado por média ponderada usando RPSO para o terceiro banco de dados por análise de vibração

Classes	a	b	c
a = Normal	<b>127</b>	1	0
b = Curto na espira	0	<b>367</b>	45
c = Curto na bobina	0	46	<b>374</b>

Fonte: Elaborado pelo autor (2023).

Ao analisar a Tabela 35, pode-se ver que os erros estão ocorrendo entre as classes de curto na espira e curto na bobina. Como já evidenciado anteriormente, ocorreram menos erros na classe normal, sendo este apenas um erro. Observando esses resultados percebe-se a eficiência da análise de corrente para o diagnóstico de falhas elétricas como o de curto-circuito em espiras e bobinas. Um fato interessante é que no caso anterior o classificador teve dificuldades na classificação da classe normal, já neste caso, não houve esse problema.

#### 8.4 Comparativo dos resultados com outros autores

Como mostrado no capítulo 2, diversos trabalhos já realizaram o diagnóstico de falhas em MER, por este motivo é importante realizar um comparativo entre algumas metodologias

empregadas atualmente para o diagnóstico dos tipos de falhas apresentadas nesta pesquisa.

#### 8.4.1 Comparativo para o primeiro banco de dados

Outros trabalhos da literatura realizaram o diagnóstico de falhas simples e compostas nas MER empregando diferentes técnicas e metodologias. Os trabalhos que empregam falhas combinadas se diferenciam muito nas suas condições de operação da máquina, neste sentido, uma comparação direta acaba sendo inviável.

Os trabalhos de Martins *et al.* (2021) e Martins *et al.* (2022) apresentam o diagnóstico de falhas simples e combinadas em MER. O primeiro, empregando o uso da FA e de algumas formas de extração apresentadas na Tabela 36, conseguiu uma acurácia de 81,41%. Já o segundo trabalho, conseguiu uma acurácia de 100%, para isso, os autores usam uma técnica chamada de *stacked sparse autoencoder*, além disso, houve um aumento de amostras empregando o uso do *synthetic minority oversampling technique* e do *additive white gaussian noise technique*. O autor também testou o método empregando o banco de dados original, sem o uso de dados sintéticos, conseguindo um resultado de 93,50% usando o SVM para a classificação.

No terceiro trabalho, criado por Inyang *et al.* (2023) e apresentado na Tabela 36, é feita uma combinação de três repositórios, incluindo o COMFAULDA, que incluem falhas simples e combinadas em rolamentos, engrenagens e no eixo. Para realizar esse processo é feito um empilhamento com três tipos de redes convolucionais diferentes: rede neural convolucional com coerência espectral cíclica, sendo em inglês, *Cyclic Spectral Coherence - Convolutional Neural Network* (CSC-CNN), rede neural convolucional com curtose espectral, em inglês, *Spectral Kurtosis - Convolutional Neural Network* (SK-CNN), e rede neural convolucional com análise com bicoerência, do inglês, *Bicoherence analysis - Convolutional Neural Network* (BICO-CNN). Como meta modelo, os autores utilizaram o *Error Correcting Output Code* do MATLAB®. Os autores conseguiram uma acurácia de 98,9%.

O trabalho possui um alto custo computacional por usar um empilhamento com três redes convolucionais utilizando imagens como entrada. Além disso, mesmo com o uso das redes foi necessário utilizar outras técnicas para evidenciar características nas imagens. Ele também não explorou as falhas mais simples do banco de dados COMFAULDA.

O classificador proposto neste trabalho conseguiu uma acurácia de 88,68%, utilizando um banco de dados que possuem sete condições de operações, sendo algumas delas combinações das falhas mais simples, o que pode gerar mais erros. Além disso, foi usado apenas

uma forma de extração de características e não houve adição de dados sintéticos. Os resultados se apresentaram consistentes e condizentes com a literatura do caso exposto. A Tabela 36 mostra um comparativo entre esses trabalhos e a técnica proposta.

Tabela 36 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas combinadas em MER

Autores	Extração	Condições de operação	Método	Acurácia
(MARTINS <i>et al.</i> , 2021)	Fator de crista, assimetria, velocidade do motor e três picos no espectro da frequência do sinal	Normal, desbalanceamento, desbalanceamento + desalinhamento horizontal, desbalanceamento + desalinhamento vertical	FA	81,41%
(MARTINS <i>et al.</i> , 2022)	106 formas de extração de características diferentes	Normal, desbalanceamento, desbalanceamento + desalinhamento horizontal, desbalanceamento + desalinhamento vertical	Stacked Sparse Autoencoder	100%
(INYANG <i>et al.</i> , 2023)	Coerência espectral cíclica, curtose espectral e análise com bicoerência	Normal, desbalanceamento, desbalanceamento + desalinhamento horizontal, desbalanceamento + desalinhamento vertical, desalinhamento horizontal + desalinhamento vertical, 5 condições de falha em engrenagens e 6 condições de falha em rolamentos	CNN	98,9%
Autor	RMS	Normal, desbalanceamento, desbalanceamento + desalinhamento horizontal, desbalanceamento + desalinhamento vertical, desalinhamento horizontal + desalinhamento vertical, desalinhamento horizontal e desalinhamento vertical	Ensemble e RPSO	88,68%

Fonte: Elaborado pelo autor (2023).

#### 8.4.2 Comparativo para o segundo banco de dados

Existem diversos trabalhos que já fizeram o diagnóstico de falhas com foco em rolamentos em MER, possuindo como diferencial, os classificadores e metodologias.

No trabalho de Chegini *et al.* (2022) é empregado um método de decomposição de enxames, do inglês, *Swarm Decomposition* (SWD), junto com uma técnica de avaliação de



distância de compensação otimizada, sendo escrita em inglês, *Optimized Compensation Distance Evaluation Technique* (OCDET), para realizar o processo de seleção de atributos estatísticos para a extração de características. Para encontrar os atributos mais sensíveis para o SVM é feito uma combinação de PSO, *Sine–Cosine Algorithm* (SCA) e *Levy flight distribution*. Com esse processo foram empregadas 27 formas de extração de características. O banco de dados utilizado é o mesmo empregado neste trabalho e o autor conseguiu uma acurácia de teste de 83,01%.

O segundo trabalho desenvolvido por Cui *et al.* (2022) utiliza imagens de padrões de pontos simétricos, do inglês, *Symmetrized Dot Pattern* (SDP) para o diagnóstico de falhas nos rolamentos e uma rede profunda com módulo de atenção de bloco convolucional, escrito em inglês, *Deep Residual Network with Convolutional Block Attention Module* (CBAM-DRN). Os autores conseguiram uma acurácia de 99,65% no diagnóstico de falhas em partes dos rolamentos. Como mencionado anteriormente, o uso da *convolutional neural network* (CNN) pode causar um alto custo computacional, o que não seria viável em algumas aplicações. Emprega o uso de um motor de corrente alternada nos experimentos.

O terceiro trabalho desenvolvido por (XIAO *et al.*, 2023), divide as falhas em três grupos de conjuntos de dados: grupo 1 (normal, falha na pista interna do rolamento), grupo 2 (normal, falha no elemento rolante do rolamento) e grupo 3 (normal, falha na pista externa). Os autores conseguiram um resultado de acurácia média para os grupos 1, 2 e 3 de 98,4%, 97,5% e 99,1%, respectivamente. Para efetuar o diagnóstico ele propõem um novo método chamado de *Graph Neural Network-based Bearing Fault Detection* (GNNBFD). O motor empregado tem uma potência de 2 hp, mas o seu tipo não foi especificado.

Já o classificador criado neste trabalho conseguiu uma acurácia de teste 100%, usando apenas o RMS, a curtose e a entropia para realizar a extração de características. Além disso, são classificadas 9 condições de operações, possuindo falhas nos rolamentos e de desbalanceamento. Os resultados se mostraram superiores ou parecidos com outros trabalhos da literatura, evidenciando concordância com as soluções propostas. Na Tabela 37 são apresentados os trabalhos e suas respectivas técnicas para este tipo de diagnóstico.

#### **8.4.3 Comparativo para o terceiro banco de dados**

Por ser um banco de dados bem recente, não foi encontrado nenhum trabalho que empregasse ele, mesmo assim, existem trabalhos que empregam o uso de aprendizado de máquina na detecção de falhas elétricas em máquinas síncronas de ímã permanente. Na Tabela 38 são

apresentados estes trabalhos.

Tabela 37 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas focadas em rolamentos

Autores	Extração	Condições de operação	Método	Acurácia
(CHEGINI <i>et al.</i> , 2022)	27 formas de extração de características diferentes	Rolamento saudável, 7 tipos de defeitos em rolamentos e desbalanceamento	SVM	83.01%
(CUI <i>et al.</i> , 2022)	CBAM-DRN	Rolamento saudável, falha na pista externa, falha na pista interna, falha no elemento rolante e as três falhas ao mesmo tempo	CBAM-DRN	100%
(XIAO <i>et al.</i> , 2023)	23 formas de extração de características diferentes	Grupo 1, grupo 2 e grupo 3	GNNBFD	98,4%, 97,5%, 99,1%
Autor	RMS Curtose e assimetria	Rolamento saudável, 7 tipos de defeitos em rolamentos e desbalanceamento	Ensemble e RPSO	100%

Fonte: Elaborado pelo autor (2023).

Tabela 38 – Técnicas e métodos utilizados pela literatura para o diagnóstico de falhas elétricas em máquinas síncronas de ímã permanente

Autores	Extração	Condições de operação	Método	Acurácia
(KAO <i>et al.</i> , 2019)	WPD, CNN	Normal, falha de desmagnetização, falha de semi-desmagnetização, anel interno danificado e adição de pó de alumínio	CNN	98,8%
(ZERDANI <i>et al.</i> , 2022)	PSD	Normal e 3 zonas de severidade de curto entre espiras	SVM	98,5%
(PIETRZAK; WOLKIEWICZ, 2022)	STFT	Normal Falha no enrolamento do estator	MLP	99,0%
Autor	RMS	Normal, curto nas espiras, curto nas bobinas	Ensemble e RPSO	98,95%

Fonte: Elaborado pelo autor (2023).

No trabalho de Kao *et al.* (2019) é feito a classificação de 5 condições de operação de uma máquina síncrona de ímã permanente propondo duas técnicas para a extração de características: transformação em pacotes de Wavelet, *Wavelet Packet Transform* (WPT) e uma rede neural convolucional. Foi feito um comparativo entre essas duas formas de extração e a CNN

obteve os melhores resultados. Com o emprego dela na extração e na classificação de dados, conseguiu-se uma acurácia de 98,8%.

O segundo trabalho, proposto por Zerdani *et al.* (2022), realiza uma modelagem de um sistema no MATLAB® para a detecção de níveis de severidade de curto em espiras em máquinas síncronas de ímã permanente. A extração de características é feita empregando o uso da densidade espectral de potência, do inglês, *Power Spectral Density* (PSD). A classificação é desenvolvida usando o SVM, conseguindo assim uma acurácia de 98,5%.

Já no trabalho de Pietrzak e Wolkiewicz (2022), empregando o uso da transformada de Fourier de tempo curto, em inglês, *Short-Time Fourier Transforms* (STFT) para a extração de características e a MLP para classificação, conseguiu uma acurácia de 99% para classificar duas condições de operação de uma máquina síncrona de ímã permanente.

Por fim, o classificador criado neste trabalho, usando apenas o RMS para realizar a extração de características e empregando o uso do classificador combinado por média ponderada com o uso do RPSO, conseguiu uma acurácia de teste de 98,95% usando a análise de corrente e classificando três condições de operação de um banco de dados real. Os resultados do classificador se mostram próximos ou melhores do que os encontrados pela literatura para este tipo de problema, mostrando mais uma vez a aplicabilidade do método proposto.

## 8.5 Considerações parciais do capítulo 8

Neste capítulo foram apresentados os resultados de três bancos de dados, cada um possuindo suas particularidades e condições de operação diferentes.

O primeiro tinha como foco um sistema com falhas mecânicas simples e combinadas, do qual, o problema de classificação se tornava difícil porque os classificadores se confundiam entre as amostras dessas duas condições. O classificador combinado por média ponderada e com o uso do RPSO se tornou viável neste diagnóstico, aumentando os resultados dos classificadores formadores e conseguindo soluções equivalentes a outras técnicas. No teste final, o modelo conseguiu uma acurácia de 88,68%.

O segundo banco de dados tem como base as falhas em rolamentos, possuindo como dificuldade poucos amostras para a classificação, fazendo com que existisse uma grande variabilidade nos resultados das dobras da validação cruzada. A técnica proposta neste trabalho conseguiu resultados equivalentes a outros modelos apresentados. A acurácia final de teste foi de 100%.

Por fim, o terceiro banco de dados apresenta amostras de falhas elétricas em máquinas síncronas de ímã permanente e os resultados mais uma vez mostram a aplicabilidade do classificador combinado por média ponderada e com o uso do RPSO, no qual, consegue valores iguais ou maiores que outros modelos.

Também foi observado a diferença nos resultados entre os diagnósticos realizados com análise de corrente e de vibração. No qual, a análise de corrente se mostrou superior neste tipo de problema. Mesmo assim, os resultados da análise de vibração não foram ruins, em especial, pode-se destacar os acertos na classe normal, que diferentemente da análise de corrente, conseguiu resultados individuais ótimos.

É interessante também comentar um pouco sobre os classificadores *ensemble* utilizados. Cada um deles possui vantagens e desvantagens. Por exemplo, os classificadores que empregam o uso das meta-heurísticas possuem como desvantagem a necessidade do ajuste de parâmetros dos otimizadores, o que pode tornar a sua criação mais complicada.

No caso do COBYLA e do Powell, não é necessário realizar esses ajustes, o que ajuda na sua criação. Como desvantagem, eles acabam conseguindo resultados um pouco menores, se comparado com as meta-heurísticas. Mesmo assim em algumas situações, como em bancos de dados grandes é interessante a sua aplicação.

O classificador com empilhamento, além de não conseguir resultados tão bons quanto os outros classificadores, ainda possui como desvantagem a criação do meta-classificador, que ainda precisa de ajustes para realizar sua função com eficiência. No caso do KNORA-U e KNORA-E também existe o ajuste de parâmetros do próprio classificador e um desempenho um pouco abaixo das meta-heurísticas.

Neste capítulo também foi apresentado um comparativo entre metodologias de outros autores que trabalharam com o mesmo banco de dados ou conjuntos parecidos. Mesmo com métodos diferentes, os resultados foram melhores ou equivalentes e com outras soluções. No próximo capítulo é apresentado uma conclusão deste trabalho e as perspectivas de novas pesquisas provenientes desta.

## 9 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou uma nova metodologia para realizar o diagnóstico de falhas simples e combinadas em MER. Para isso, foi proposto um novo classificador de votação ponderada que usa o RPSO para encontrar os pesos para cada modelo. Para testar o método proposto foram utilizados três bancos de dados recentes, dos quais, apresentam diferentes tipos de falhas que ocorrem em MER.

O classificador desenvolvido neste trabalho, teve sua eficiência melhor apresentada no primeiro banco de dados, conseguindo aumentar muito os resultados na classificação de falhas combinadas.

Ao decorrer do trabalho foram apresentadas uma revisão da literatura, uma fundamentação teórica sobre as MER e alguns tipos de falhas que ocorrem nelas. Ademais, foram mostrados conceitos sobre classificação, combinação e otimização.

Os resultados obtidos mostraram ser melhores ou equivalentes às soluções propostas por outras técnicas *ensemble* presentes na literatura, conseguindo para o primeiro, segundo e terceiro banco de dados resultados de acurácia de teste de 88,68%, 100% e 98,95%, respectivamente. Além disso, foram feitos comparativos com outros trabalhos que também realizam o diagnóstico em MER, deste processo, notou-se resultados melhores ou similares aos encontrados por estes.

Desta comparação também evidenciou-se a simplicidade das técnicas de extração de características empregadas neste trabalho, que apenas com o uso do RMS, da curtose e da assimetria no domínio do tempo conseguiu resultados proporcionais a trabalhos que utilizavam diversas técnicas mais complexas e com um custo computacional maior.

No terceiro banco de dados também foi observado a diferença nos resultados entre a análise de vibração e a análise de corrente. Para o caso em questão, falha de curto-circuito em espiras e bobinas, a análise de corrente conseguiu resultados melhores, mostrando a sua melhor aplicabilidade neste tipo de diagnóstico.

### 9.1 Trabalhos futuros

Espera-se a continuidade deste trabalho em alguns tópicos:

- Criação de apenas um único algoritmo de classificação capaz de diagnosticar de forma eficiente diversas falhas elétricas e mecânicas em diferentes máquinas elétricas rotativas.

- Criação do próprio banco de dados em laboratório para validação dos resultados encontrados neste trabalho e em outros presentes na literatura.
- Emprego de novas técnicas de otimização para a busca dos pesos para o classificador de votação por média ponderada.
- Comparativo direto entre o uso de técnicas de *deep learning* com modelos *ensemble* de classificação convencional.
- Observar a eficiência do diagnóstico com a presença das duas formas de análise: vibração e corrente.
- Realizar o processo de prognóstico da vida útil, desta forma, além de dizer o estado atual da máquina, poderá ser realizado previsões a curto e médio prazo das curvas de degradação destes equipamentos.

## REFERÊNCIAS

- AGGARWAL, C. C. **Data Classification: Algorithms and Applications**. New York, USA: CRC Press, 2015. v. 1. 498-501 p.
- ALI, M. Z.; SHABBIR, M. N. S. K.; LIANG, X.; ZHANG, Y.; HU, T. Machine learning-based fault diagnosis for single- and multi-faults in induction motors using measured stator currents and vibration signals. **IEEE Transactions on Industry Applications**, v. 55, n. 3, p. 2378–2391, 2019.
- ALMEDALLAH, M. K.; MUDHAFAR, A. A. A.; CLARK, S.; WALSH, S. D. Vector-based three-dimensional (3d) well-path optimization assisted by geological modelling and borehole-log extraction. **Upstream Oil and Gas Technology**, v. 7, p. 100053, 2021.
- ASMITA, S.; SHUKLA, K. Review on the architecture, algorithm and fusion strategies in ensemble learning. **International Journal of Computer Applications**, v. 108, p. 21–28, 12 2014.
- BOEHMKE, B. C.; GREENWELL, B. M. **Hands-On Machine Learning with R**. [S. l.]: Chapman Hall, 2019. v. 1.
- BONALDI, E. L. **Diagnóstico Preditivo de Avarias em Motores de Indução Trifásicos com MCSA e Teoria de Conjuntos Aproximados** 2005. 185 f. Tese (Doutorado em Engenharia Elétrica) – Programa de Pós-Graduação em Engenharia Elétrica: Universidade Federal de Itajubá, Itajubá, 2005.
- BREIMAN, L.; FRIEDMAN, J.; STONE, C. J.; OLSHEN, R. **Classification and Regression Trees**. first. [S. l.]: Taylor Francis, 1984. v. 1. 368 p.
- BUITINCK, L.; LOUPPE, G.; BLONDEL, M.; PEDREGOSA, F.; MUELLER, A.; GRISEL, O.; NICULAE, V.; PRETTENHOFER, P.; GRAMFORT, A.; GROBLER, J.; LAYTON, R.; VANDERPLAS, J.; JOLY, A.; HOLT, B.; VAROQUAUX, G. API design for machine learning software: experiences from the scikit-learn project. In: **ECML PKDD Workshop: Languages for Data Mining and Machine Learning**. [S. l.: s. n.], 2013. p. 108–122.
- CANABARRO, J. V. **Estratégias de combinação em um sistema de múltiplos classificadores heterogêneos** 2018. 67 f. Graduação em ciências da computação – Unioeste - Universidade Estadual do Oeste do Paraná, Cascavel, 2018.
- CASTRO, L. N. D.; FERRARI, D. G. **Introdução à mineração de dados: conceitos básicos, algoritmos e aplicações**. São Paulo: Saraiva, 2016. v. 1.
- CHANG, C.-C.; LIN, C.-J. Libsvm: A library for support vector machines. **ACM Transactions on Intelligent Systems and Technology**, v. 2, 07 2007.
- CHEGINI, S. N.; AMINI, P.; AHMADI, B.; BAGHERI, A.; AMIRMOSTOFIAN, I. **Intelligent bearing fault diagnosis dataset**. Kaggle, 2021. Disponível em: <https://www.kaggle.com/dsv/2840060>.
- CHEGINI, S. N.; AMINI, P.; AHMADI, B.; BAGHERI, A.; AMIRMOSTOFIAN, I. Intelligent bearing fault diagnosis using swarm decomposition method and new hybrid particle swarm optimization algorithm. **Soft Comput**, v. 26, n. 3, p. 1475–1497, 2022.

CHEN, S.; MENG, Y.; TANG, H.; TIAN, Y.; HE, N.; SHAO, C. Robust deep learning-based diagnosis of mixed faults in rotating machinery. **IEEE/ASME Transactions on Mechatronics**, v. 25, n. 5, p. 2167–2176, 2020.

COLEY, D. A. **An Introduction to Genetic Algorithms for Scientists and Engineers**. [S. l.]: World Scientific Publishing Company, 1999. v. 1. 244 p.

CORNE, B.; VERVISCH, B.; DEBRUYNE, C.; KNOCKAERT, J.; DESMET, J. Comparing mcsa with vibration analysis in order to detect bearing faults — a case study. In: **2015 IEEE International Electric Machines Drives Conference (IEMDC)**. [S. l.: s. n.], 2015. p. 1366–1372.

CRUZ, R. M. O.; HAFEMANN, L. G.; SABOURIN, R.; CAVALCANTI, G. D. C. Deslib: A dynamic ensemble selection library in python. **Journal of Machine Learning Research**, v. 21, n. 8, p. 1–5, 2020. Disponível em: <http://jmlr.org/papers/v21/18-144.html>.

CUI, W.; MENG, G.; GOU, T.; WANG, A.; XIAO, R.; ZHANG, X. Intelligent rolling bearing fault diagnosis method using symmetrized dot pattern images and cbam-drn. **Sensors**, v. 22, n. 24, 2022. Disponível em: <https://www.mdpi.com/1424-8220/22/24/9954>.

DINEVA, A.; MOSAVI, A.; GYIMESI, M.; VAJDA, I.; NABIPOUR, N.; RABCZUK, T. Fault diagnosis of rotating electrical machines using multi-label classification. **Applied Sciences**, v. 23, n. 9, 2019.

DJAGAROV, N.; ENCHEV, G.; KOKIN, S.; DJAGAROVA, J. Fault diagnosis modeling of induction machine. In: **2021 17th Conference on Electrical Machines, Drives and Power Systems (ELMA)**. [S. l.: s. n.], 2021. p. 1–7.

EINSTEIN, A. **Relativity: The special and general theory**. [S. l.]: Holt and Company, 1916. v. 1.

EL-KENAWY, E.-S. M.; IBRAHIM, A.; MIRJALILI, S.; EID, M. M.; HUSSEIN, S. E. Novel feature selection and voting classifier algorithms for covid-19 classification in ct images. **IEEE Access**, v. 8, p. 179317–179335, 2020.

ESPINOZA-SEPULVEDA, N.; SINHA, J. Mathematical validation of experimentally optimised parameters used in a vibration-based machine-learning model for fault diagnosis in rotating machines. **machines**, v. 9, n. 8, p. 155, 2021.

FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. de Leon Ferreira de. **Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina**. Rio de Janeiro: LTC, 2011. v. 1.

FAN, R.-E.; CHANG, K.-W.; HSIEH, C.-J.; WANG, X.-R.; LIN, C.-J. Liblinear: a library for large linear classification. **Journal of Machine Learning Research**, v. 9, p. 1871–1874, 08 2008.

FRIEDMAN, J. H.; TIBSHIRANI, R.; HASTIE, T. **The Elements of Statistical Learning**. [S. l.]: Springer, 2009. v. 1.

GHAZALI, M. H. M.; RAHIMAN, W. Vibration analysis for machine monitoring and diagnosis: A systematic review. **Shock and Vibration**, v. 1, p. 25, 2021.



GOYAL, D.; PABLA, B. S. The vibration monitoring methods and signal processing techniques for structural health monitoring: A review. **Arch Computat Methods Eng**, v. 23, p. 585–594, 2016.

GRUS, J. **Data Science Do Zero: Noções Fundamentais com Python**. second. São Paulo: Alta Books, 2021. v. 1.

HABBOUCHE, H.; AMIRAT, Y.; BENKEDJOUH, T.; BENBOUZID, M. Bearing fault event-triggered diagnosis using a variational mode decomposition-based machine learning approach. **IEEE Transactions on Energy Conversion**, v. 37, n. 1, p. 466–474, 2022.

HAND, A. **Motores elétricos : manutenção e solução de problemas**. Porto Alegre: Bookman, 2015. v. 1.

HARRIS, C. R.; MILLMAN, K. J.; WALT, S. J. van der; GOMMERS, R.; VIRTANEN, P.; COURNAPEAU, D.; WIESER, E.; TAYLOR, J.; BERG, S.; SMITH, N. J.; KERN, R.; PICUS, M.; HOYER, S.; KERKWIJK, M. H. van; BRETT, M.; HALDANE, A.; RÍO, J. F. del; WIEBE, M.; PETERSON, P.; GÉRARD-MARCHANT, P.; SHEPPARD, K.; REDDY, T.; WECKESSER, W.; ABBASI, H.; GOHLKE, C.; OLIPHANT, T. E. Array programming with NumPy. **Nature**, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, set. 2020. Disponível em: <https://doi.org/10.1038/s41586-020-2649-2>.

HILLIER, F. S.; LIEBERMAN, G. J. **Introdução a pesquisa operacional**. São Paulo: Mc Graw Hill, 2010. v. 8.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. first. [S. l.]: University of Michigan Press, 1975.

HUO, Z.; MARTÍNEZ-GARCÍA, M.; ZHANG, Y.; SHU, L. A multisensor information fusion method for high-reliability fault diagnosis of rotating machinery. **IEEE Transactions on Instrumentation and Measurement**, v. 71, p. 1–12, 2022.

INYANG, U. I.; PETRUNIN, I.; JENNIONS, I. Diagnosis of multiple faults in rotating machinery using ensemble learning. **Sensors**, v. 23, n. 2, 2023.

IRHOUMAH, M.; PUSCA, R.; LEFEVRE, E.; MERCIER, D.; ROMARY, R. Detection of the stator winding inter-turn faults in asynchronous and synchronous machines through the correlation between harmonics of the voltage of two magnetic flux sensors. **IEEE Transactions on Industry Applications**, v. 55, n. 3, p. 2682–2689, 2019.

JEONG, H.; LEE, H.; KIM, S.; KIM, S. W. Interturn short fault diagnosis using magnitude and phase of currents in permanent magnet synchronous machines. **Sensors**, v. 22, n. 12, p. 4597, 2022.

JUNG, J.-H.; LEE, J.-J.; KWON, B.-H. Online diagnosis of induction motors using mcsa. **IEEE Transactions on Industrial Electronics**, v. 53, n. 6, p. 1842–1852, 2006.

JUNG, W.; YUN, S.-H.; LIM, Y.-S.; CHEONG, S.; PARK, Y.-H. Vibration and current dataset of three-phase permanent magnet synchronous motors with stator faults. **Data in Brief**, v. 47, p. 108952, 2023. ISSN 2352-3409. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2352340923000707>.

KAO, I.-H.; WANG, W.-J.; LAI, Y.-H.; PERNG, J.-W. Analysis of permanent magnet synchronous motor fault diagnosis based on learning. **IEEE Transactions on Instrumentation and Measurement**, v. 68, n. 2, p. 310–324, 2019.

KE, G.; MENG, Q.; FINLEY, T.; WANG, T.; CHEN, W.; MA, W.; YE, Q.; LIU, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf>.

KE, G.; MENG, Q.; FINLEY, T.; WANG, T.; CHEN, W.; MA, W.; YE, Q.; LIU, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. **Advances in neural information processing systems**, v. 30, p. 3146–3154, 2017.

KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: **Proceedings of ICNN'95 - International Conference on Neural Networks**. [S. l.: s. n.], 1995. v. 4, p. 1942–1948 vol.4.

KINGMA, D.; BA, J. Adam: A method for stochastic optimization. **International Conference on Learning Representations**, 12 2014.

KITTLER, J.; HATEF, M.; DUIN, R.; MATAS, J. On combining classifiers. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 20, n. 3, p. 226–239, 1998.

KO, A. H.; SABOURIN, R.; BRITTO, J. A. S. From dynamic classifier selection to dynamic ensemble selection. **Pattern Recognition**, v. 41, n. 5, p. 1718–1731, 2008.

KOKOSKA, S.; ZWILLINGER, D. **CRC Standard Probability and Statistics Tables and Formulae**. second. Nova York: CHAPMAN HALL, 2000. v. 1.

KOOHESTANI, A.; ABDAR, M.; KHOSRAVI, A.; NAHAVANDI, S.; KOOHESTANI, M. Integration of ensemble and evolutionary machine learning algorithms for monitoring diver behavior using physiological signals. **IEEE Access**, v. 7, p. 98971–98992, 2019.

KOSOW, I. L. **Máquinas elétricas e transformadores**. São Paulo: Globo, 2005. v. 1.

KRZYWINSKI, M.; ALTMAN, N. Classification and regression trees. **Nature Methods**, v. 14, n. 8, p. 757–758, 2017.

LEE, D.-H.; HONG, C.; JEONG, W.-B.; AHN, S. Time–frequency envelope analysis for fault detection of rotating machinery signals with impulsive noise. **Applied Sciences**, v. 11, n. 12, 2021.

LI, H.; HU, G.; LI, J.; ZHOU, M. Intelligent fault diagnosis for large-scale rotating machines using binarized deep neural networks and random forests. **IEEE Transactions on Automation Science and Engineering**, v. 19, n. 2, p. 1109–1119, 2022.

LI, J.; HUANG, R.; HE, G.; LIAO, Y.; WANG, Z.; LI, W. A two-stage transfer adversarial network for intelligent fault diagnosis of rotating machinery with multiple new faults. **IEEE/ASME Transactions on Mechatronics**, v. 26, n. 3, p. 1591–1601, 2021.

LORBIESKI, R. **Modelo de ensembles multiníveis para classificadores** 2018. 89 f. Dissertação (Mestrado em Ciência da Computação.) – Programa de Pós-Graduação em Ciência da Computação, da Universidade Federal de Santa Catarina, Santa Catarina, 2014.

- LUKE, S. **Essentials of Metaheuristics**. second. [S. l.]: Lulu, 2013. Available for free at <http://cs.gmu.edu/~sean/book/metaheuristics/>.
- LÓPEZ, R. Z. **Classificação automática de defeitos em máquinas rotativas**. 2014. 120 f. Graduação em Engenharia Eletrônica e de Computação – Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2014.
- MAFAULDA. **Machinery Fault Database [Online]**. Rio de Janeiro, RJ: [S. n.], 2017. Disponível em: [https://www02.smt.ufrj.br/~offshore/mfs/page\\_01.html](https://www02.smt.ufrj.br/~offshore/mfs/page_01.html). Acesso em: 15 jan. 2023.
- MANCONI, A.; ARMANO, G.; GNOCCHI, M.; MILANESI, L. A soft-voting ensemble classifier for detecting patients affected by covid-19. **Applied Sciences**, v. 12, n. 15, 2022.
- MARTINS, D. H. C. D. S.; LIMA, A.; PINTO, M. F.; HEMERLY, D.; PREGO, T.; SILVA, F.; TARRATAÇA, L.; MONTEIRO, U.; GUTIERREZ, R.; HADDAD, D. Hybrid data augmentation method for combined failure recognition in rotating machines. **Journal of Intelligent Manufacturing**, v. 34, p. 1–19, 01 2022.
- MARTINS, D. H. C. D. S.; VIANA, D.; LIMA, A.; PINTO, M. F.; TARRATAÇA, L.; SILVA, F.; GUTIERREZ, R.; PREGO, T.; MONTEIRO, U.; HADDAD, D. Diagnostic and severity analysis of combined failures composed by imbalance and misalignment in rotating machines. **The International Journal of Advanced Manufacturing Technology**, v. 114, 04 2021.
- MARTINS, D. H. C. de S. S.; PESTANA-VIANA, D.; LIMA, A. A.; GUTIÉRREZ, R. H. R.; VAZ, L.; HADDAD, D. B. **COMFAULDA (Composed Fault Dataset)**. Rio de Janeiro, RJ: [S. n.], 2022. Disponível em: <https://ieee-dataport.org/documents/composed-fault-dataset-comfaulda>. Acesso em: 20 jan. 2023.
- MIAO, J.; WANG, J.; ZHANG, D.; MIAO, Q. Improved generative adversarial network for rotating component fault diagnosis in scenarios with extremely limited data. **IEEE Transactions on Instrumentation and Measurement**, v. 71, p. 1–13, 2022.
- MOHAMMED, A.; DJUROVIĆ, S. Electric machine bearing health monitoring and ball fault detection by simultaneous thermo-mechanical fibre optic sensing. **IEEE Transactions on Energy Conversion**, v. 36, n. 1, p. 71–80, 2021.
- MOLDOVAN, D. Crow search algorithm based ensemble of machine learning classifiers for epileptic seizures detection. In: **2020 International Conference on e-Health and Bioengineering (EHB)**. [S. l.: s. n.], 2020. p. 1–4.
- PACHECO, F.; DRIMUS, A.; DUGGEN, L.; CERRADA, M.; CABRERA, D.; SANCHEZ, R.-V. Deep ensemble-based classifier for transfer learning in rotating machinery fault diagnosis. **IEEE Access**, v. 10, p. 29778–29787, 2022.
- PAIVA, F. A. P. de. **Otimização por enxame de partículas: usando uma adaptação de serendipidade**. first. [S. l.]: Editora IFRN, 2018. 98 p.
- PALAVICINO, P. C.; SARLIOGLU, B. Prediction of failure path current for synchronous reluctance and interior permanent magnet synchronous machines accounting for saturation. In: **2020 IEEE Transportation Electrification Conference Expo (ITEC)**. [S. l.: s. n.], 2020. p. 1165–1170.

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

PIETRZAK, P.; WOLKIEWICZ, M. Machine learning-based stator current data-driven pmsm stator winding fault diagnosis. **Sensors**, v. 22, n. 24, 2022. ISSN 1424-8220.

PONTI Jr., M. P. Combining classifiers: From the creation of ensembles to the decision fusion. In: **2011 24th SIBGRAPI Conference on Graphics, Patterns, and Images Tutorials**. [S. l.: s. n.], 2011. p. 1–10.

POWELL, M. A view of algorithms for optimization without derivatives. **Mathematics today**, v. 43, 01 2007.

POWELL, M. J. D. An efficient method of finding the minimum of a function of several variables without calculating derivatives. **The Computer Journal**, v. 7, n. 2, p. 155–162, 1964.

POWELL, M. J. D. **A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation**. [S. l.]: Springer Netherlands, 1994. 51–67 p.

PRESS, W. H.; TEUKOLSKY, S. A.; VETTERLING, W. T.; FLANNERY, B. P. **Numerical Recipes: The art of scientific computing**. [S. l.]: Cambridge University Press, 2007. v. 3.

ROCHA, D. S. C. **Aprendizado de máquina aplicado ao reconhecimento automático de falhas em máquinas rotativas** 2018. 78 f. Dissertação (Mestrado em Engenharia Elétrica) – Escola de engenharia, Programa de Pós-Graduação em Engenharia Elétrica: Universidade Federal de Minas Gerais, Belo Horizonte, 2018.

RODER, M.; PASSOS, L. A.; ROSA, G. H. de; ALBUQUERQUE, V. H. C. de; PAPA, J. P. Reinforcing learning in deep belief networks through nature-inspired optimization. **Applied Soft Computing**, v. 108, p. 107466, 2021. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494621003896>.

RODER, M.; ROSA, G. H. de; PASSOS, L. A.; PAPA, J. P.; ROSSI, A. L. D. Harnessing particle swarm optimization through relativistic velocity. In: **2020 IEEE Congress on Evolutionary Computation (CEC)**. [S. l.: s. n.], 2020. p. 1–8.

ROSA, G. H. de; RODRIGUES, D.; PAPA, J. P. **Opytimizer: A Nature-Inspired Python Optimizer**. 2019.

SABERI, A. N.; BELAHCEN, A.; SOBRA, J.; VAIMANN, T. Lightgbm-based fault diagnosis of rotating machinery under changing working conditions using modified recursive feature elimination. **IEEE Access**, v. 10, p. 81910–81925, 2022.

SANSEVERINO, E. B. **Aceleração do método de otimização L-BFGS usando tecnologia CUDA**. 2018. 144 f. Dissertação (Mestrado em Engenharia de Sistemas e Computação.) – Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2014.

SANTOS, M. S.; SOARES, J. P.; ABREU, P. H.; ARAUJO, H.; SANTOS, J. Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [research frontier]. **IEEE Computational Intelligence Magazine**, v. 13, n. 4, p. 59–76, 2018.

SHEN, Y.; ZHENG, K.; YANG, Y.; LIU, S.; HUANG, M. Cba-clsv: A class-level soft-voting ensemble based on the chaos bat algorithm for intrusion detection. **Applied Sciences**, v. 12, n. 21, 2022.

SILVA, J. G. B. da. **Modelagem e tratamento de sinais da assinatura elétrica de máquinas para a melhoria do diagnóstico de falhas** 2015. 184 f. Tese (Doutorado em Engenharia Elétrica) – Programa de Pós-Graduação em Engenharia Elétrica: Universidade Federal de Itajubá, Itajubá, 2015.

SONNAILLON, M. O.; BISHEIMER, G.; ANGELO, C. D.; GARCÍA, G. O. Online sensorless induction motor temperature monitoring. **IEEE Transactions on Energy Conversion**, v. 25, n. 2, p. 273–280, 2010.

TANG, Z.; LIU, J.; LI, C. Improved multivariate hierarchical multiscale dispersion entropy: A new method for industrial rotating machinery fault diagnosis. **IEEE Access**, v. 10, p. 102842–102859, 2022.

TEAM, T. pandas development. **pandas-dev/pandas: Pandas**. Zenodo, 2020. Disponível em: <https://doi.org/10.5281/zenodo.3509134>.

TIAN, J.; AZARIAN, M. H.; PECHT, M.; NIU, G.; LI, C. An ensemble learning-based fault diagnosis method for rotating machinery. In: **2017 Prognostics and System Health Management Conference (PHM-Harbin)**. [S. l.: s. n.], 2017. p. 1–6.

UMANS, S. D. **Máquinas elétricas de Fitzgerald e Kingsley**. [S. l.]: Bookman, 2014. v. 1.

VIRTANEN, P.; GOMMERS, R.; OLIPHANT, T. E.; HABERLAND, M.; REDDY, T.; COURNAPEAU, D.; BUROVSKI, E.; PETERSON, P.; WECKESSER, W.; BRIGHT, J.; VAN DER WALT, S. J.; BRETT, M.; WILSON, J.; MILLMAN, K. J.; MAYOROV, N.; NELSON, A. R. J.; JONES, E.; KERN, R.; LARSON, E.; CAREY, C. J.; POLAT, Í.; FENG, Y.; MOORE, E. W.; VANDERPLAS, J.; LAXALDE, D.; PERKTOLD, J.; CIMRMAN, R.; HENRIKSEN, I.; QUINTERO, E. A.; HARRIS, C. R.; ARCHIBALD, A. M.; RIBEIRO, A. H.; PEDREGOSA, F.; VAN MULBREGT, P.; SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. **Nature Methods**, v. 17, p. 261–272, 2020.

WANG, X.-B.; ZHANG, X.; LI, Z.; WU, J. Ensemble extreme learning machines for compound-fault diagnosis of rotating machinery. **Knowledge-Based Systems**, v. 188, 2020.

WITTEN, I. H.; FRANK, E. **Data mining : practical machine learning tools and techniques**. [S. l.]: Elsevier, 2011. v. 3.

WOLPERT, D. H. Stacked generalization. **Neural Networks**, v. 5, n. 2, p. 241–259, 1992.

XIAO, L.; YANG, X.; YANG, X. A graph neural network-based bearing fault detection method. **Scientific Reports**, v. 13, n. 5286, 2023.

YASSA, N.; RACHEK, M. Modeling and detecting the stator winding inter turn fault of permanent magnet synchronous motors using stator current signature analysis. **Mathematics and Computers in Simulation**, v. 167, p. 325–339, 2020.

ZERDANI, S.; ELHAFYANI, M. L.; ZOUGGAR, S. Application of power spectral density and the support vector machine to fault diagnosis for permanent magnet synchronous motor. **SN Applied Sciences**, v. 4, n. 9, 2022.

ZHANG, J.; ZHANG, Q.; HE, X.; SUN, G.; ZHOU, D. Compound-fault diagnosis of rotating machinery: A fused imbalance learning method. **IEEE Transactions on Control Systems Technology**, v. 29, n. 4, p. 1462–1474, 2021.

ÁVILA, S. L. **Otimização paramétrica com computação evolutiva**. Florianópolis: Publicações do IFSC, 2020. v. 1.