



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

JOSÉ WESLEY FEITOSA OLIVEIRA

**Utilização de Abordagens de Múltiplos Classificadores para Classificação de Falhas  
em Sistemas Elétricos**

Recife

2025

JOSÉ WESLEY FEITOSA OLIVEIRA

**Utilização de Abordagens de Múltiplos Classificadores para Classificação de Falhas em Sistemas Elétricos**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

**Área de Concentração:** Inteligência Computacional

**Orientador (a):** Paulo Salgado Gomes de Mattos Neto

**Coorientador (a):** Eraylson Galdino da Silva

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Oliveira, José Wesley Feitosa.

Utilização de Abordagens de Múltiplos Classificadores para Classificação de Falhas em Sistemas Elétricos / José Wesley Feitosa Oliveira. - Recife, 2024.

86f.: il.

Dissertação (Mestrado) - Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós- Graduação em Ciência da Computação, 2024.

Orientação: Paulo Salgado Gomes de Mattos Neto.

Coorientação: Eraylson Galdino da Silva.

Inclui referências.

1. Sistemas Elétricos de Potência; 2. Detecção de Falhas; 3. Sistemas de Múltiplos Classificadores; 4. Conjunto; 5. Seleção Dinâmica de Classificadores. I. Neto, Paulo Salgado Gomes de Mattos. II. Silva, Eraylson Galdino da. III. Título.

UFPE-Biblioteca Central

**José Wesley Feitosa Oliveira**

**“Utilização de Abordagens de Múltiplos Classificadores para  
Classificação de Falhas em Sistemas Elétricos”**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 12/12/2024.

**BANCA EXAMINADORA**

---

Prof. Dr. Leandro Maciel Almeida  
Centro de Informática / UFPE

---

Prof. Dr. Manoel Henrique da Nóbrega Marinho  
Escola Politécnica de Pernambuco / UPE

---

Prof. Dr. Paulo Salgado Gomes de Mattos Neto  
Centro de Informática / UFPE  
(**orientador**)

Dedico este trabalho aos meus avós Maria Lúcia (*in memoriam*) e Rozemiro Alves (*in memoriam*), deixo-lhes esta pequena homenagem por tudo que fizeram por mim.

## **AGRADECIMENTOS**

Sou grato a Deus, em primeiro lugar, por ter me dado a força e a capacidade necessárias para superar os desafios e chegar até este momento.

Agradeço à minha esposa, Laís, que sempre me apoiou e me deu forças para continuar e não desistir, aos meus pais, Ricardo e Marygleide, pelo incentivo e comprometimento que sempre tiveram com minha formação escolar e à minha irmã, Karen, pelo companheirismo e parceria.

Deixo meus agradecimentos ao meu orientador, Prof. Paulo Salgado e ao meu coorientador, Prof. Eraylson Galdino, sou grato por toda paciência, clareza e disposição que tiveram para me conduzir ao longo do processo de aprendizagem, sou grato aos amigos que fiz no mestrado, ao meu companheiro de pesquisa Dioeliton Passos e aos colegas do grupo de pesquisa, todos vocês tiveram um importante papel em minha caminhada.

Por fim, sou grato a todos que fazem Programa de Pós Graduação do Centro de Informática da Universidade Federal de Pernambuco, por sua excelência e comprometimento.

A principal função de um sistema de proteção é assegurar a desconexão de todo sistema elétrico submetido a qualquer anormalidade que o faça operar fora dos limites previstos ou de parte dele. Em segundo lugar, o sistema de proteção tem a função de fornecer as informações necessárias aos responsáveis por sua operação, de modo a facilitar a identificação dos defeitos e a sua consequente recuperação (FILHO; MAMEDE, 2013).

## RESUMO

A operação dos sistemas de transmissão de energia elétrica depende da detecção e classificação rápida de falhas para atuar os sistemas de proteção e evitar danos maiores. Os sistemas de classificação de falhas baseados em aprendizado de máquina foram muito utilizados por terem um bom desempenho e facilidade de interpretação. Os sistemas automatizados de proteção e supervisão podem auxiliar os operadores detectando as anomalias e falhas, fornecendo uma estrutura robusta para que os defeitos sejam detectados e classificados da maneira mais efetiva possível. Contudo, as metodologias atuais utilizam predominantemente os modelos de classificação tradicionais, que possuem dificuldades com conjuntos de dados desequilibrados e captura de padrões em ambientes ruidosos. Para superar essas limitações, este estudo explora as abordagens do Sistema de Classificadores Múltiplos (MCS), que possui vantagem sobre os modelos tradicionais para este tipo de situação. Os Sistemas de Classificadores Múltiplos (MCS), de maneira geral, superaram os modelos clássicos para dados ruidosos de corrente e tensão simulados em linha de transmissão e subestação, onde os valores de acurácia, precisão e *recall* sofreram uma degradação mais acentuada em abordagens de múltiplos classificadores quando comparadas com os modelos tradicionais. Esta pesquisa enfatiza o potencial das técnicas de MCS como uma solução robusta para classificação de falhas em linhas de transmissão e subestações de energia elétrica, os resultados dos MCS apresentaram maior estabilidade, sendo menos sensíveis ao aumento do ruído, tornando a classificação mais confiável e facilitando a correção do defeito no sistema, reduzindo o tempo de análise em campo e das medidas das grandezas elétricas para classificar o tipo de falha que ocorreu no sistema.

**Palavras-chaves:** Sistemas Elétricos de Potência. Detecção de Falhas. Sistemas de Múltiplos Classificadores. Conjunto. Seleção Dinâmica de Classificadores

## ABSTRACT

The operation of power transmission systems depends on the rapid detection and classification of faults to activate protection systems and prevent further damage. Machine learning-based fault classification systems have been widely used for their good performance and ease of interpretation. Automated protection and supervision systems can assist operators by detecting anomalies and faults, providing a robust framework for fault detection and classification in the most effective way possible. However, current methodologies predominantly use traditional classification models, which have difficulties with imbalanced data sets and capturing patterns in noisy environments. To overcome these limitations, this study explores the Multiple Classifier System (MCS) approaches, which have advantages over traditional models for this situation. Multiple Classifier Systems (MCS), in general, outperformed classical models for noisy current and voltage data simulated in transmission lines and substations, where accuracy, precision, and recall values suffered a more pronounced degradation in multiple classifier approaches when compared to traditional models. This research emphasizes the potential of MCS techniques as a robust solution for fault classification in transmission lines and electrical substations. The MCS results showed greater stability, being less sensitive to increased noise, making the classification more reliable and facilitating the correction of the defect in the system, reducing the time of field analysis and measurements of electrical quantities to classify the type of fault that occurred in the system.

**Keywords:** Electrical Power Systems. Fault Detection. Multiple Classifier Systems. Ensemble. Dynamic Classifier Selection

## LISTA DE FIGURAS

Figura 1 – Etapas do sistema elétrico de potência . . . . .	22
Figura 2 – Diagrama unifilar representativo de um sistema elétrico de potência . . . . .	23
Figura 3 – Gráficos da capacidade de geração de energia elétrica brasileira para o ano de 2025 . . . . .	24
Figura 4 – Relé eletromecânico de sobrecorrente do tipo indução . . . . .	28
Figura 5 – IED SIPROTEC 7SJ82 da fabricante SIEMENS, utilizado para proteção de linhas de transmissão e banco de capacitores . . . . .	29
Figura 6 – Fluxograma do modelo K-NN . . . . .	31
Figura 7 – Variação do MAPRE (%) com relação ao valor de $k$ . . . . .	32
Figura 8 – Árvore de decisão com dois níveis de dados . . . . .	33
Figura 9 – Modelo <i>Random Forest</i> composto por múltiplas árvores de decisão. A predição final resulta da predição mais frequente em cada árvore . . . . .	34
Figura 10 – Exemplo do modelo <i>XGBoost</i> . A previsão final para um dado exemplo é a soma das previsões de cada árvore. . . . .	35
Figura 11 – Fluxograma do modelo de previsão <i>LightGBM</i> com a inserção do <i>grid search</i> . . . . .	38
Figura 12 – Fluxograma do modelo <i>Catboost</i> . . . . .	39
Figura 13 – Uma estrutura geral do MCS, ou <i>Ensemble Learning</i> , empregando Seleção Estática. Esta abordagem seleciona um ou mais modelos para classificar todos os padrões de teste. . . . .	41
Figura 14 – Uma estrutura geral do MCS, ou <i>Ensemble Learning</i> , empregando Seleção Dinâmica. Esta abordagem seleciona um ou mais modelos de uma Região de Competência para classificar cada padrão do conjunto de teste. . . . .	44
Figura 15 – Distribuição de classes da Base de Dados 01 . . . . .	51
Figura 16 – Distribuição de classes da Base de Dados 02 . . . . .	53
Figura 17 – Acurácia obtida para cada classificador individualmente com o acréscimo do ruído na Base de Dados 1 . . . . .	64
Figura 18 – Acurácia obtida para cada técnica de Seleção Estática com o acréscimo do ruído na Base de Dados 1 . . . . .	66
Figura 19 – Acurácia obtida para cada técnica de Seleção Dinâmica com o acréscimo do ruído na Base de Dados 1 . . . . .	68

Figura 20 – Acurácia, Precisão e <i>Recall</i> obtidas para os melhores modelos das abordagens avaliadas na Base de Dados 1 . . . . .	69
Figura 21 – Acurácia obtida para cada classificador individualmente com o acréscimo do ruído na Base de Dados 2 . . . . .	71
Figura 22 – Acurácia obtida para cada técnica de Seleção Estática com o acréscimo do ruído na Base de Dados 2 . . . . .	73
Figura 23 – Acurácia obtida para cada técnica de Seleção Dinâmica com o acréscimo do ruído na Base de Dados 2 . . . . .	75
Figura 24 – Acurácia, Precisão e <i>Recall</i> obtidas para os melhores modelos das abordagens avaliadas na Base de Dados 2 . . . . .	76

## LISTA DE TABELAS

Tabela 1 – Testes Estatísticos para a base de dados 01 . . . . .	54
Tabela 2 – Testes Estatísticos para a base de dados 02 . . . . .	54
Tabela 3 – Níveis de ruído utilizados em cada uma das bases de dados . . . . .	56
Tabela 4 – Hiperparâmetros usados no <i>grid-search</i> para cada modelo . . . . .	57
Tabela 5 – Hiperparâmetros utilizados para cada modelo de seleção estática . . . . .	58
Tabela 6 – Hiperparâmetros utilizados para cada modelo de seleção dinâmica . . . . .	59
Tabela 7 – Métricas para avaliação de classificação. Para todas as métricas, quanto maior o valor, melhor o desempenho da classificação. . . . .	61
Tabela 8 – Avaliação dos classificadores na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	63
Tabela 9 – Avaliação das técnicas de Seleção Estática na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	65
Tabela 10 – Avaliação das técnicas de Seleção Dinâmica na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	67
Tabela 11 – Avaliação dos classificadores na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	70
Tabela 12 – Avaliação das técnicas de Seleção Estática na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	72
Tabela 13 – Avaliação das técnicas de Seleção Dinâmica na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito. . . . .	74

## LISTA DE ABREVIATURAS

**LightGBM** *Light Gradient Boosting Machine*

**XGBoost** *Extreme Gradient Boosting*

**ATP** *Alternative Transient Program*

**ATPDraw** *Alternative Transient Program Draw*

**BKS** *Behavioral Knowledge Space*

**DCS-LA** *Dynamic Classifier Selection by Local Accuracy*

**DESP** *Dynamic Ensemble Selection Performance*

**EFB** *Exclusive Feature Bundling*

**FN** Falso Negativo

**FP** Falso Positivo

**GOSS** *Gradient-based One-Side Sampling*

**IED** *Intelligent Electronic Device*

**IEEE** *Institute of Electrical and Electronics Engineers*

**K-NN** *K-Nearest Neighbors*

**KNORA** *K-Nearest Oracles*

**KNORA-E** *K-Nearest Oracles Eliminat*

**KNORA-U** *K-Nearest Oracles Union*

**MAPRE** *Mean Absolute Percent Relative Error*

**MCB** *Multiple Classifier Behavior*

**MCS** *Multiple Classifier Systems*

**META-DES** *Meta-learning for Dynamic Ensemble Selection*

**ML** *Machine Learning*

**OLA** *Overall Local Accuracy*

**ONS** Operador Nacional do Sistema Elétrico

**RoC** *Region of Competence*

**SCADA** *Supervisory Control and Data Acquisition*

**SEP** Sistema Elétrico de Potência

**SPCS** Sistema de Proteção Controle e Supervisão

**VN** Verdadeiro Negativo

**VP** Verdadeiro Positivo

## LISTA DE SÍMBOLOS

$\sigma$  Desvio Padrão

$\mu$  Valor Médio

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>17</b>
1.1	MOTIVAÇÃO	17
1.2	TRABALHOS RELACIONADOS	18
1.3	OBJETIVOS	20
<b>1.3.1</b>	<b>Objetivos Gerais</b>	<b>20</b>
<b>1.3.2</b>	<b>Objetivos Específicos</b>	<b>20</b>
1.4	QUESTÕES DA PESQUISA	20
1.5	ESTRUTURA DO TRABALHO	21
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>22</b>
2.1	SISTEMAS ELÉTRICOS DE POTÊNCIA	22
<b>2.1.1</b>	<b>Geração</b>	<b>23</b>
<b>2.1.2</b>	<b>Transmissão</b>	<b>24</b>
<b>2.1.3</b>	<b>Distribuição e Consumo</b>	<b>25</b>
<b>2.1.4</b>	<b>Proteção de Sistemas Elétricos</b>	<b>26</b>
2.2	<i>MACHINE LEARNING</i>	30
<b>2.2.1</b>	<b><i>K-Nearest Neighbors (K-NN)</i></b>	<b>30</b>
<b>2.2.2</b>	<b><i>Árvore de Decisão</i></b>	<b>31</b>
<b>2.2.3</b>	<b><i>Random Forest</i></b>	<b>33</b>
<b>2.2.4</b>	<b>Modelos baseados em <i>Gradient Boosting</i></b>	<b>34</b>
2.2.4.1	<i>Extreme Gradient Boosting (XGBoost)</i>	35
2.2.4.2	<i>Light Gradient Boosting Machine (LightGBM)</i>	36
2.2.4.3	<i>CatBoost</i>	37
<b>3</b>	<b>MÉTODO PROPOSTO</b>	<b>41</b>
3.1	SELEÇÃO ESTÁTICA	41
<b>3.1.1</b>	<b>Voto Majoritário</b>	<b>42</b>
<b>3.1.2</b>	<b>Generalização Empilhada</b>	<b>43</b>
3.2	SELEÇÃO DINÂMICA	43
<b>3.2.1</b>	<b><i>Overall Local Accuracy (OLA)</i></b>	<b>45</b>
<b>3.2.2</b>	<b><i>Dynamic Ensemble Selection Performance (DESP)</i></b>	<b>46</b>
<b>3.2.3</b>	<b><i>K-nearest-oracles (KNORA)</i></b>	<b>47</b>

3.2.3.1	<i>K-Nearest Oracles Eliminate (KNORA-E)</i> . . . . .	47
3.2.3.2	<i>K-Nearest Oracles Union (KNORA-U)</i> . . . . .	47
3.2.4	<b><i>Multiple Classifier Behavior (MCB)</i></b> . . . . .	<b>48</b>
3.2.5	<b><i>Meta-learning for Dynamic Ensemble Selection (META-DES)</i></b> . . . . .	<b>49</b>
4	<b>PROTOCOLO EXPERIMENTAL</b> . . . . .	<b>50</b>
4.1	BASES DE DADOS . . . . .	50
4.2	PROCEDIMENTO EXPERIMENTAL . . . . .	55
5	<b>RESULTADOS</b> . . . . .	<b>62</b>
5.1	BASE DE DADOS 1 . . . . .	62
5.2	BASE DE DADOS 2 . . . . .	67
5.3	DESEMPENHO COMPUTACIONAL . . . . .	75
6	<b>CONCLUSÃO</b> . . . . .	<b>77</b>
6.1	TRABALHOS FUTUROS . . . . .	78
6.2	LIMITAÇÕES DA PESQUISA . . . . .	78
6.3	CONTRIBUIÇÕES DA PESQUISA . . . . .	78
	<b>REFERÊNCIAS</b> . . . . .	<b>80</b>

# 1 INTRODUÇÃO

Este capítulo introduzirá a pesquisa, iniciando pela motivação, que está ligada aos sistemas de energia elétrica, sua importância na sociedade contemporânea e a criticidade da classificação das falhas para a operação dos sistemas de energia elétrica. Este trabalho abordará a classificação de falhas do tipo curtos-circuitos, também conhecidos como "faltas", em linhas de transmissão e subestações, que são estruturas responsáveis por transmitir e controlar o fluxo de energia entre as fontes geradoras e as unidades consumidoras.

Os trabalhos relacionados abordam as pesquisas que utilizaram classificadores tradicionais e *Multiple Classifier Systems* (MCS) para estudos em sistemas elétricos. Os MCS ainda são poucos estudados em aplicações de sistemas elétricos, mas já apresentam desempenho superior aos modelos tradicionais de *Machine Learning* (ML) para outras aplicações com dados ruidosos, o que motivou a aplicação neste trabalho.

Por fim, são mostrados os objetivos gerais e específicos e as questões relacionadas à pesquisa.

## 1.1 MOTIVAÇÃO

Os sistemas de energia elétrica são vitais para o funcionamento e o avanço das sociedades modernas, fornecendo energia para infraestrutura e serviços críticos, incluindo telecomunicações, transporte, abastecimento de água e sistemas de resposta a emergências (O'ROURKE, 2007; DOBSON et al., 2007; YAGAN et al., 2012). Os sistemas de energia modernos consistem em vários componentes elétricos (GONI et al., 2023), que são dinâmicos e suscetíveis a distúrbios ou falhas. Essas falhas podem resultar de problemas internos de rede, como curtos-circuitos, ou fatores externos, como condições ambientais (JANARTHANAM et al., 2022). Grandes usinas de geração de energia devem operar em sincronia com as redes elétricas, tornando essencial que todo o sistema de energia funcione com segurança e eficiência. Nesse contexto, a detecção de falhas é crucial, pois permite a ativação rápida de medidas de proteção, protegendo equipamentos e evitando maiores danos à rede (JAMIL; SHARMA; SINGH, 2015; LUSKOVÁ; LEITNER, 2021).

A detecção de falhas é crucial para recompor o sistema de forma efetiva e rápida. Em sistemas de energia, a recomposição da operação envolve monitorar sensores, dados de desempenho

e parâmetros operacionais para detectar quando algo está errado, como tensão anormal, picos de temperatura ou quedas de energia (HE et al., 2016). Algoritmos automatizados de detecção de falhas auxiliam os operadores destacando anomalias que podem indicar uma falha. Nesse sentido, os operadores de sistemas elétricos podem gerenciar melhor a confiabilidade do sistema e evitar falhas em cascata integrando sistemas avançados de detecção de falhas (como modelos de ML e monitoramento em tempo real).

## 1.2 TRABALHOS RELACIONADOS

Vários estudos na literatura aplicaram uma variedade de modelos de ML para classificação de falhas em linhas de transmissão, incluindo *Árvore de Decisão* (ASMAN et al., 2021), *Random Forest* (VISWAVANDYA; PATEL; SAHOO, 2021), *Extreme Gradient Boosting (XGBoost)* (WANG et al., 2019), *CatBoost* (OGAR; HUSSAIN; GAMAGE, 2022), *Light Gradient Boosting Machine (LightGBM)* (ATRIGNA et al., 2023) e *K-Nearest Neighbors (K-NN)* (ABED et al., 2023). Esses modelos foram aplicados com sucesso para classificar falhas de curtos-circuitos em linhas de transmissão e linhas de distribuição com base em sinais de tensão e corrente. No entanto, o desempenho desses modelos de ML geralmente se deteriora consideravelmente quando expostos a dados ruidosos ou distorções de sinal (HARISH; JAYAN, 2021) devido a problemas relacionados a *overfitting*, *underfitting* e seleção de modelo.

Neste contexto, os MCS são uma alternativa promissora para abordar as limitações de abordagens tradicionais, melhorando a robustez e a precisão dos modelos de classificação de falhas. O MCS consiste em um conjunto de modelos, cada um especializado em reconhecer diferentes padrões, que são selecionados e combinados para atingir uma classificação final. A motivação por trás dessa abordagem é que nenhum modelo individualmente tem desempenho ideal em todos os cenários possíveis (WOLPERT, 1996).

De acordo com (DIETTERICH, 2000), os MCS são técnicas que combinam a previsão de vários modelos individuais para melhorar a precisão, robustez e generalização do aprendizado de máquina. A ideia central dos MCS é que diferentes modelos podem capturar diferentes aspectos dos dados, e sua combinação permite reduzir erros, como o viés e a variância, resultando em um modelo mais preciso e confiável. O funcionamento dessas técnicas baseia-se em gerar múltiplos classificadores a partir de variações nos dados de treinamento, nos algoritmos ou em ambos. Os métodos mais comuns são o *Bagging*, que reduz a variância ao treinar múltiplos modelos em subconjuntos aleatórios dos dados de treinamento e combinar suas previsões, o

---

*Boosting*, que ajusta classificadores sequencialmente, dando mais peso aos erros cometidos anteriormente para melhorar a precisão, e o *Stacking*, que combina diferentes modelos de aprendizado através de um meta-modelo que aprende a partir das previsões dos classificadores base. O sucesso dos MCS depende da diversidade entre os classificadores individuais, pois a combinação de modelos complementares tende a produzir melhores resultados do que qualquer classificador isoladamente.

Poucos estudos aplicaram o MCS à classificação de falhas em sistemas de transmissão elétrica. (TOMA; KIM; KIM, 2021) utilizou conjuntos de classificadores combinados com Redes Neurais Convolucionais (KHANAM et al., 2024), já (TOMA; KIM, 2020) combinou as técnicas de conjunto com a Transformada Discreta de Wavelet (SIPAI et al., 2024) para classificação de faltas em motores elétricos de indução (SOUZA et al., 2022). (GHAEMI et al., 2022) aborda as técnicas baseadas em *Stacked* (HAJIHOSSEINLOU; MAGHSOUDI; GHEZELBASH, 2024) na localização e classificação de faltas em redes de distribuição elétrica e (HARISH; JAYAN, 2021) em linhas de transmissão. Em estudo de 2021, (VAISH et al., 2021) apontou a inexistência de trabalhos utilizando técnicas de conjunto heterogêneo (TEWARI; DWIVEDI, 2020) em aplicações de sistemas elétricos de potência. As abordagens com *Bagging* ou *Boosting* (VAISH et al., 2021; HARISH; JAYAN, 2021) foram mais utilizadas no contexto de sistemas elétricos, essas técnicas combinam a saída de vários modelos para melhorar a precisão geral, mas consideram cada classificador igualmente em todo o conjunto de dados. Por outro lado, resultados relevantes em tarefas de classificação (FRAGOSO et al., 2021) foram obtidos selecionando classificadores específicos para cada padrão de teste, uma técnica conhecida como seleção dinâmica de classificadores, bem como combinando modelos por meio de empilhamento, que gera um metamodelo que aprende a combinar os modelos em um conjunto, resultando em classificações mais precisas (HAJIHOSSEINLOU; MAGHSOUDI; GHEZELBASH, 2024). Essa lacuna destaca a necessidade de investigar se as abordagens de MCS, como aprendizado empilhado e seleção dinâmica de classificadores, podem melhorar significativamente o desempenho da classificação de falhas em sistemas de transmissão elétrica, particularmente em dados ruidosos.

## 1.3 OBJETIVOS

### 1.3.1 Objetivos Gerais

Este trabalho busca investigar a eficácia dos MCS, especificamente os modelos de empilhamento e seleção dinâmica de classificadores em cenários ruidosos, e compará-los com modelos de classificação tradicionais. As técnicas de MCS utilizam conjunto de classificadores para classificar instâncias do conjunto de teste com base no desempenho de cada classificador no conjunto de treinamento e validação.

### 1.3.2 Objetivos Específicos

1. Estudar a eficácia dos MCS para classificar curtos-circuitos em linhas de transmissão e subestações de energia elétrica
2. Verificar o desempenho dos MCS em dados ruidosos de sistemas elétricos
3. Propor técnicas de classificação de curtos-circuitos em subestações e linhas de transmissão que sejam menos susceptíveis a ruídos que modelos de classificação tradicionais

## 1.4 QUESTÕES DA PESQUISA

Com intuito de atingir os objetivos da pesquisa, o presente estudo visa responder as seguintes questões:

1. "Modelos baseados em MCS atingem melhores resultados do que modelos tradicionais para classificação de curto-circuito em linhas de transmissão e subestações de energia elétrica?"
2. "Os MCS conseguem melhorar a sensibilidade ao ruído em relação aos classificadores individualmente?"

## 1.5 ESTRUTURA DO TRABALHO

Esta dissertação foi dividida em seis capítulos. O Capítulo 1 apresenta a pesquisa, falando sobre o seu contexto, os objetivos e as questões relacionadas à pesquisa. O Capítulo 2 aborda os conceitos teóricos dos temas estudados, como funcionam os sistemas elétricos e os modelos clássicos de classificação. O Capítulo 3 é dedicado aos MCS, são apresentadas as técnicas de seleção estática e dinâmica, como elas funcionam e suas vantagens e desvantagens. O Capítulo 4 trata-se de como o processo dos experimentos foi realizado, detalhando as bases de dados e as manipulações realizadas, explica como se decorreu o processo de seleção dos hiperparâmetros dos modelos de classificação com utilização de *grid search*, a aplicação dos MCS e as métricas utilizadas para avaliação. O Capítulo 5 detalha os resultados obtidos nas duas bases de dados por meio de tabelas e gráficos para facilitar a visualização e compreensão, por fim são comparados graficamente os melhores resultados de cada tipo de abordagem: monolítica e MCS (estática e dinâmica). Concluindo, o Capítulo 6 encerra o documento com as considerações finais sobre o desempenho de cada abordagem estudada, as possibilidades de trabalhos futuros e as limitações da pesquisa.

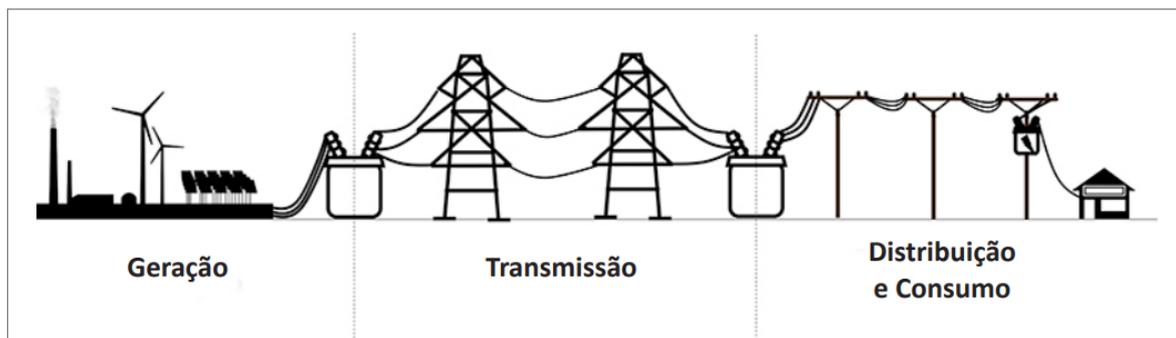
## 2 FUNDAMENTAÇÃO TEÓRICA

Os referenciais teóricos utilizados para sustentar a pesquisa foram divididos em duas partes principais: Sistema Elétrico de Potência (SEP) e modelos de classificação inteligentes. Inicialmente, foram introduzidos os conceitos dos sistemas elétricos, mostrando sua divisão e relevância para as sociedades humanas por meio de dados oficiais dos órgãos reguladores brasileiros. A seguir, aborda-se os modelos de classificação utilizados na pesquisa, tentando fazer um estudo o mais justo possível, foram utilizados desde modelos clássicos e bastante difundidos como K-NN e Árvore de Decisão até o mais recentes e inovadores baseados em *Gradient Boosting*, como *XGBoost*, *LightGBM* e *Catboost*.

### 2.1 SISTEMAS ELÉTRICOS DE POTÊNCIA

Nesta seção serão abordados os conceitos do SEP e sua importância. O SEP é dividido em quatro partes: geração, transmissão, distribuição e consumo. Cada uma dessas etapas possui suas particularidades e suas operações requerem atenção e cuidado por se tratarem de sistemas sensíveis e potencialmente perigosos para os seres humanos. A Figura 1 representa um sistema elétrico de potência com as etapas de geração, transmissão, distribuição e consumo.

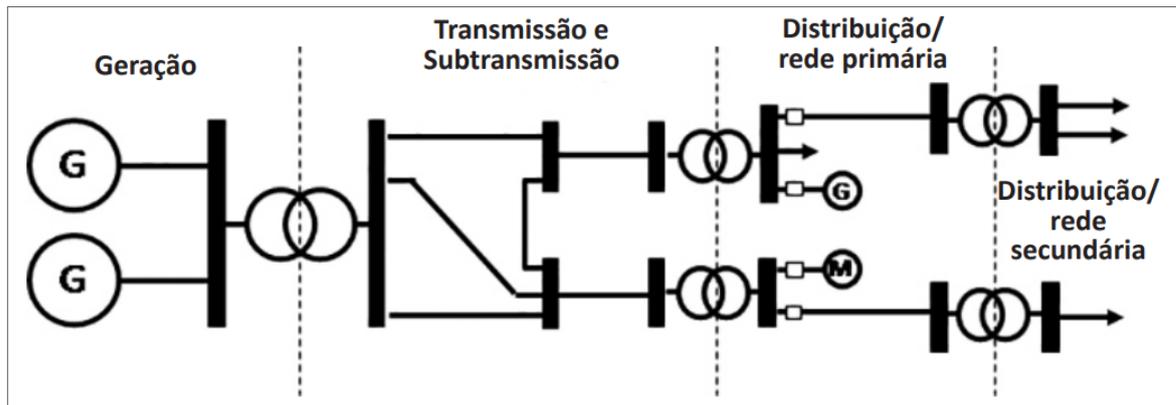
Figura 1 – Etapas do sistema elétrico de potência



Fonte: (SILVA, 2019)

A Figura 2 representa um diagrama unifilar de um SEP, os diagramas unifilares são amplamente utilizados por profissionais que atuam com sistemas elétricos, pois fornece informações precisas de maneira simples e clara, ele facilita a compreensão das interconexões dos equipamentos e dispositivos, como geradores, transformadores, linhas de transmissão, disjuntores, barramentos, cargas e outros elementos.

Figura 2 – Diagrama unifilar representativo de um sistema elétrico de potência



Fonte: (SILVA, 2019)

### 2.1.1 Geração

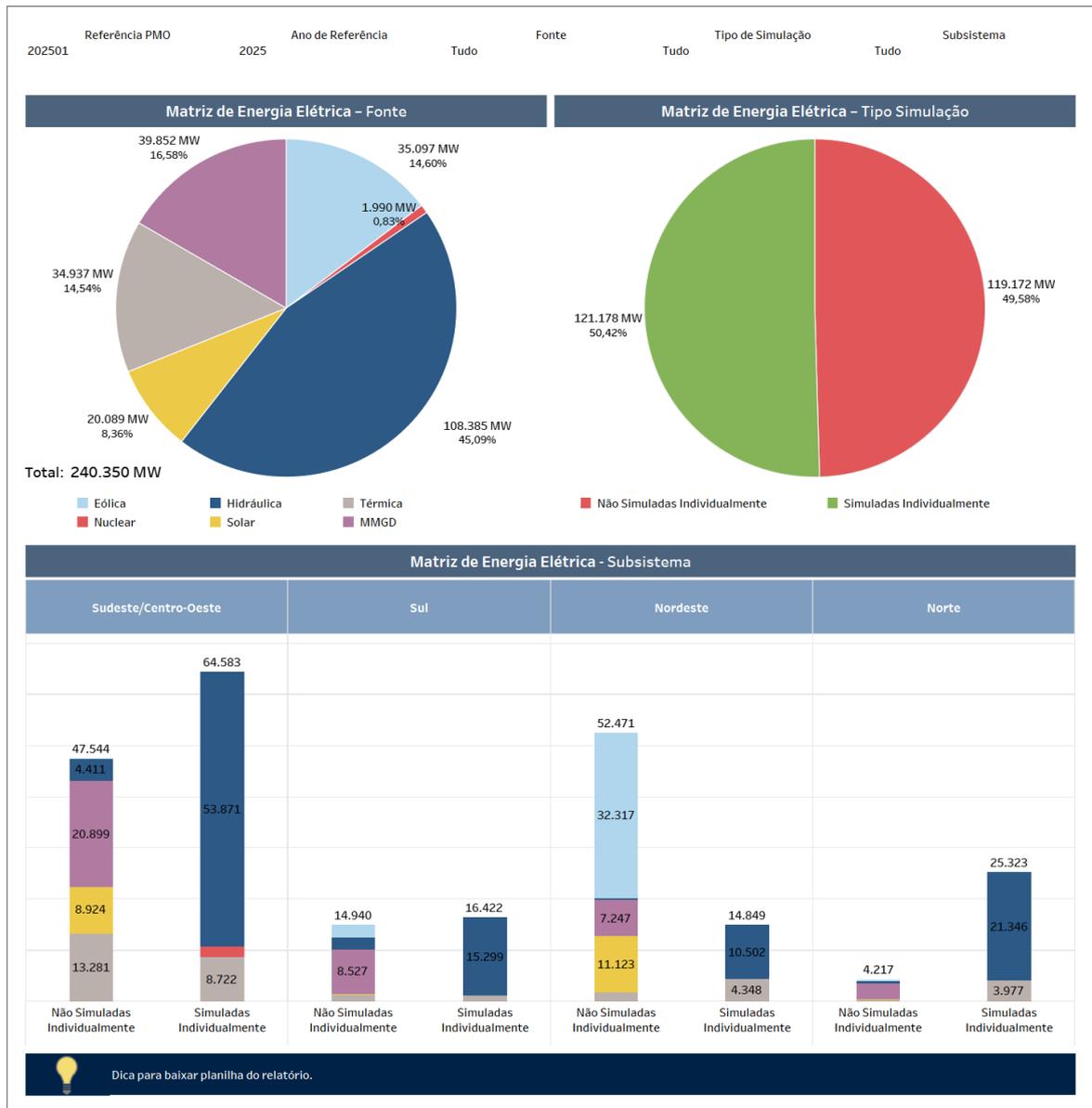
A geração é a primeira etapa do processo e consiste em converter diferentes formas de energia em energia elétrica. Atualmente, existem diversas formas de gerar eletricidade, desde as fontes baseadas em queima de combustíveis fósseis como carvão ou derivados do petróleo, conhecidas como usinas termelétricas até as nucleares, que utilizam fissão nuclear de núcleos de átomos pesados, como urânio e plutônio.

O Brasil, por ser um país rico em recursos naturais, tem as hidrelétricas como principal fonte de geração de energia elétrica, este meio de geração utiliza o movimento da água em quedas ou barragens para movimentar turbinas, gerando eletricidade. Nos últimos anos, porém, a geração fotovoltaica, que utiliza a luz do sol, e a eólica, que utiliza a força dos ventos, aumentaram exponencialmente o potencial energético do Brasil, sobretudo na região Nordeste. Em um momento onde a humanidade busca reduzir as emissões de gases de efeito estufa na atmosfera, as fontes de energia limpa, como são conhecidas a hidráulica, solar e fotovoltaica, desempenham um importante papel na matriz energética nacional <sup>1</sup>.

A Figura 3 foi retirada da página do Operador Nacional do Sistema Elétrico (ONS) e traz os gráficos da geração de energia elétrica brasileira, os dados são mostrados em Mega-Watt e divididos por região e tipo de geração.

<sup>1</sup> <<https://www.ons.org.br/paginas/sobre-o-sin/o-sistema-em-numeros>>

Figura 3 – Gráficos da capacidade de geração de energia elétrica brasileira para o ano de 2025



Fonte: ONS (Operador Nacional do Sistema Elétrico, 2025)

### 2.1.2 Transmissão

Após a etapa da geração, a energia elétrica produzida nas usinas é transmitida por longas distâncias até as unidades de distribuição, esse processo é feito pelas subestações e linhas de transmissão.

Na saída das unidades geradoras, transformadores elevam a tensão da energia elétrica para níveis mais altos, entre 230 kV e 765 kV, com o objetivo de minimizar perdas por efeito Joule. A energia elétrica é então transportada pelas linhas de transmissão, que são longas

---

estruturas compostas por cabos condutores e torres para sustentação dos cabos. Essas linhas são responsáveis por interligar as subestações, as subestações compreendem estruturas intermediárias onde são realizados o controle, proteção e transformação de tensão. Elas contêm transformadores, dispositivos de controle de corrente e tensão, além de sistemas de proteção.

As subestações de transmissão podem ser classificadas de diversas formas, nesse estudo, vamos apresentar as classificações por função, desta maneira, temos os seguintes tipos de subestações:

**Subestações Elevadoras:** São instaladas nas saídas das usinas geradoras, essas subestações elevam a tensão gerada para os níveis de transmissão, geralmente entre 230 kV e 765 kV. Essa elevação de tensão visa diminuir as perdas por calor ao longo das linhas de transmissão.

**Subestações de Transmissão:** Interligam as linhas de transmissão e viabilizam o controle e a qualidade de energia do sistema, além de manter o fluxo de potência equilibrado.

**Subestações Abaixadoras:** Abaixam o nível de tensão da transmissão para os níveis de distribuição local, geralmente entre 13,8 kV e 34,5 kV.

**Subestações de Manobra:** Esse tipo de subestação, em particular, não realiza transformação de tensão, sua função é realizar o controle e a manobra de linhas de transmissão e distribuição. Elas são essenciais para isolar partes do sistema para manutenção ou em caso de falhas.

### 2.1.3 Distribuição e Consumo

A distribuição e o consumo da energia são as etapas finais do SEP. A distribuição leva a energia elétrica da rede de transmissão até os consumidores finais (residências, indústrias e comércios). Ao final do processo de transmissão, são instaladas subestação abaixadoras, geralmente construídas próximas aos centros de consumo, nelas a tensão é reduzida para níveis mais baixos, entre 13,8 kV e 34,5 kV. Neste nível de tensão, a eletricidade é distribuída através de cabos e postes até os transformadores de distribuição, estes reduzem a tensão para o último nível, que é aplicado pelos consumidores em geral: 127 V, 220 V ou 380 V.

Indústrias e pontos comerciais de alto consumo de energia como *shopping centers*, possuem uma subestação dedicada, uma vez que, o processo convencional de distribuição e consumo, citado anteriormente, não é viável para estes casos e causaria desequilíbrios no sistema.

#### 2.1.4 Proteção de Sistemas Elétricos

A proteção dos sistemas elétricos visa garantir que o sistema opere de maneira confiável e segura, para isso são instalados diversos equipamentos e dispositivos nas subestações como relés, *Intelligent Electronic Device* (IED), disjuntores, transformadores de corrente e potencial e *Supervisory Control and Data Acquisition* (SCADA), esse conjunto de ferramentas viabiliza o conjunto conhecido como Sistema de Proteção Controle e Supervisão (SPCS).

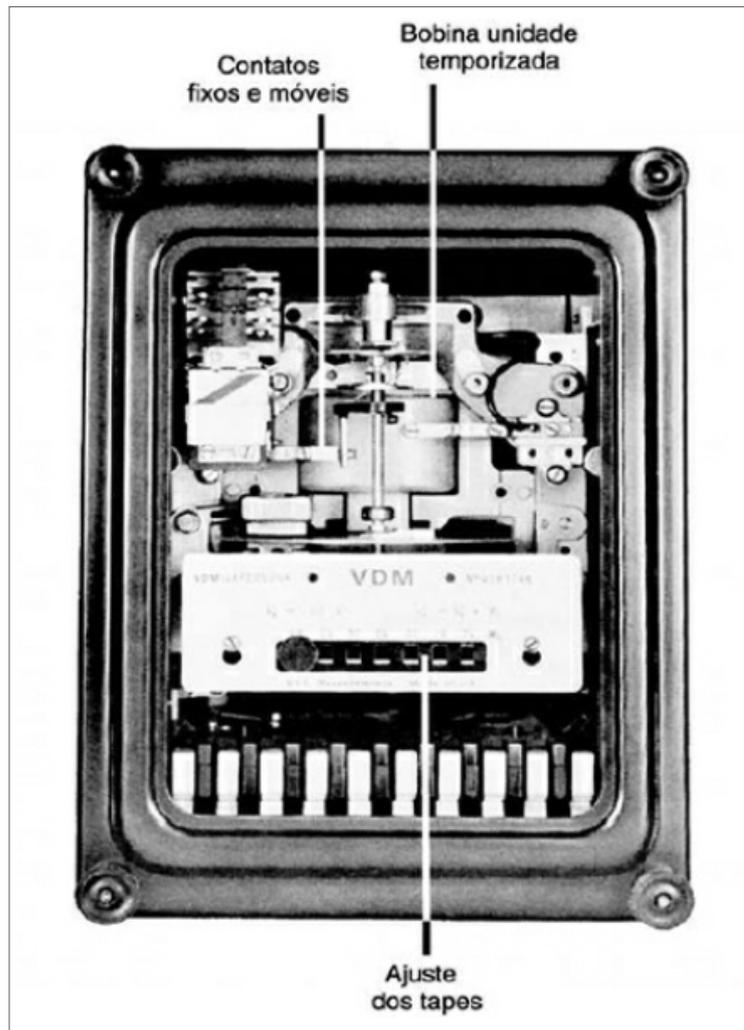
O SPCS deve considerar algumas propriedades fundamentais para garantir um bom desempenho. A seletividade, por exemplo, é uma técnica utilizada no estudo de proteção e coordenação que assegura que apenas o elemento de proteção mais próximo ao defeito desconecte a parte defeituosa do sistema elétrico. Já as zonas de atuação referem-se à capacidade do elemento de proteção de determinar se a ocorrência está dentro ou fora da zona protegida. Quando o defeito está dentro dos limites da zona protegida, o elemento de proteção deve agir e acionar a abertura do disjuntor associado em um tempo previamente definido. Se o defeito ocorre fora desses limites, o relé não deve ser sensibilizado ou, caso seja, deve bloquear seu sistema restritor de atuação. A velocidade de operação também é essencial, sendo definida como o menor tempo possível de atuação, desde que respeite um tempo mínimo pré-determinado. Essa característica permite reduzir ou eliminar avarias no sistema protegido, minimizar o tempo de afundamento de tensão durante as ocorrências e possibilitar a resincronização das fontes de geração. Outra propriedade importante é a sensibilidade, que é a capacidade do elemento de proteção de identificar com precisão a faixa e os valores indicados para sua operação ou não operação. O nível de sensibilidade pode ser avaliado numericamente, utilizando a relação entre a corrente de curto-circuito em seu valor máximo no ponto mais extremo da zona de proteção e a corrente mínima capaz de acionar o elemento de proteção. A confiabilidade é outra propriedade indispensável, garantindo que o elemento de proteção cumpra suas funções com segurança e exatidão. Além disso, a automação é uma característica que possibilita ao elemento de proteção operar automaticamente quando solicitado pelas grandezas elétricas que o sensibilizam, retornando à sua posição de operação sem intervenção humana, caso isso seja conveniente, após o término da ocorrência. Outras propriedades fundamentais do SPCS

---

incluem a imunidade dos relés a sobrecargas e sobretensões momentâneas, bem como às oscilações naturais de corrente, tensão e frequência consideradas normais pelo projeto. Os relés devem ser projetados com bobinas e circuitos de baixo consumo de energia e devem manter suas características inalteradas diante de diferentes configurações do sistema elétrico (FILHO; MAMEDE, 2013).

Os relés têm evoluído continuamente desde o surgimento do primeiro dispositivo de proteção eletromecânico, em 1901. Esse dispositivo inicial era um relé de sobrecorrente baseado no princípio de indução. Em 1908, foi desenvolvido o princípio da proteção diferencial de corrente, seguido, em 1910, pelas proteções direcionais, e somente por volta de 1930 surgiu a proteção de distância. Ao longo do tempo, a qualidade e a complexidade dos dispositivos eletromecânicos avançaram, permitindo que os esquemas de proteção alcançassem níveis cada vez maiores de sofisticação e confiabilidade. Na década de 1930, surgiram os primeiros relés de proteção com tecnologia baseada em componentes eletrônicos, utilizando semicondutores. No entanto, os relés eletrônicos ou estáticos não foram prontamente aceitos no mercado devido à predominância dos relés eletromecânicos, que já eram fabricados com alta qualidade, robustez e competitividade. Nos anos 1980, com o rápido avanço da microeletrônica, surgiram os primeiros relés de proteção baseados em tecnologia digital. Algumas concessionárias, temendo confiar exclusivamente nos novos relés digitais, os utilizaram como proteção de retaguarda em conjunto com os relés eletromecânicos. Com a chegada da tecnologia digital, houve uma mudança significativa no conceito de vida útil dos sistemas de proteção. Os relés eletromecânicos, devido à sua construção robusta, apresentavam uma vida útil de 20 a 30 anos. Em contrapartida, a vida útil dos relés digitais não é limitada pelo desgaste dos componentes, mas pela obsolescência da tecnologia da informação que os sustenta. À medida que os *softwares* aplicados aos sistemas de proteção digitais evoluem, com maior capacidade de programação e lógica, torna-se necessário desenvolver novos relés com as mesmas funções para aproveitar os avanços tecnológicos (FILHO; MAMEDE, 2013). A figura 4 exibe um relé eletromecânico do tipo indução para proteção contra sobrecorrente.

Figura 4 – Relé eletromecânico de sobrecorrente do tipo indução



Fonte: (FILHO; MAMEDE, 2013)

Os relés digitais, atualmente também são conhecidos como IED, são indispensáveis nos esquemas de proteção, especialmente com a crescente automação dos sistemas elétricos industriais e de potência. Constituem-se de circuitos eletrônicos com chips de alta velocidade de processamento, operando por meio de programas dedicados que analisam as informações recebidas dos transformadores de medida. Comandos são executados externamente com base na avaliação realizada pelo relé. Seus ajustes podem ser feitos diretamente no painel frontal, por meio de teclas de membrana com instruções específicas, ou conectando um microcomputador via comunicação serial RS-232. Os relés digitais revolucionaram a proteção elétrica, oferecendo vantagens antes inexistentes. Além das funções básicas de proteção, incluem recursos como comunicação, medição elétrica, controle, sinalização remota e acesso remoto. Eles mantêm os mesmos princípios das tecnologias predecessoras, como os relés eletromecânicos e estáticos, mas adicionam maior velocidade, sensibilidade, interface amigável, armazenamento

de informações e novas funcionalidades. Embora continuem a utilizar grandezas de entrada analógicas, como tensão e corrente, essas grandezas são convertidas internamente em sinais digitais por conversores Analógico/Digital. Introduzidos no Brasil na década de 1980, os relés digitais ganharam impulso nos anos 1990, com o avanço e a universalização da digitalização nos sistemas elétricos. Diferentemente de seus antecessores, eles processam valores medidos como tensão, corrente e frequência, além de realizar operações lógicas e aritméticas. Esses relés são equipados com tecnologia avançada para condicionamento e conversão de sinais analógicos, utilizando filtros para reduzir ruídos e transformadores para isolamento galvânico. O conversor Analógico/Digital, auxiliado por um multiplexador, converte as grandezas analógicas em dados numéricos enviados aos microprocessadores. Estes últimos realizam funções de medição, proteção e controle, apresentando os resultados no *display* e/ou enviando comandos aos canais de saída. Os microprocessadores também desempenham funções de supervisão e comunicação, operando com algoritmos dedicados. Essas características tornam os relés digitais ferramentas essenciais para a proteção e supervisão de sistemas elétricos modernos, unindo eficiência, confiabilidade e adaptabilidade às demandas tecnológicas atuais (FILHO; MAMEDE, 2013). A Figura 5 mostra um relé digital, ou IED, de fabricação SIEMENS <sup>2</sup> utilizado para proteção de linhas de transmissão e bancos de capacitores.

Figura 5 – IED SIPROTEC 7SJ82 da fabricante SIEMENS, utilizado para proteção de linhas de transmissão e banco de capacitores



Fonte: SIEMENS, 2024

<sup>2</sup> <<https://www.siemens.com/global/en.html>>

## 2.2 MACHINE LEARNING

Segundo (MICHALSKI JAIME G. CARBONELL, 1983), ML é descrito como um campo da inteligência artificial que se concentra no desenvolvimento de algoritmos e técnicas que permitem aos computadores aprenderem a partir de dados e experiências anteriores, sem serem explicitamente programados para cada tarefa específica. Destaca-se que ML envolve a construção de modelos computacionais que podem generalizar padrões a partir de exemplos observados, permitindo que o sistema melhore seu desempenho ao longo do tempo.

(MICHALSKI JAIME G. CARBONELL, 1983) cita em sua obra três tipos de aprendizado: Aprendizado supervisionado, Aprendizado não supervisionado e Aprendizado por reforço. O primeiro consiste em sistemas que recebem exemplos rotulados e aprendem uma função de mapeamento entre entrada e saída. O segundo quando o sistema analisa dados sem rótulos para identificar padrões ou estruturas subjacentes. E o último quando o agente aprende interagindo com o ambiente e recebendo recompensas ou penalidades com base em suas ações.

Buscando otimizar o processo de classificação de falhas no sistema elétrico, foram escolhidos classificadores supervisionados consolidados na literatura para esta tarefa, nas próximas seções, estes modelos de classificação serão explorados.

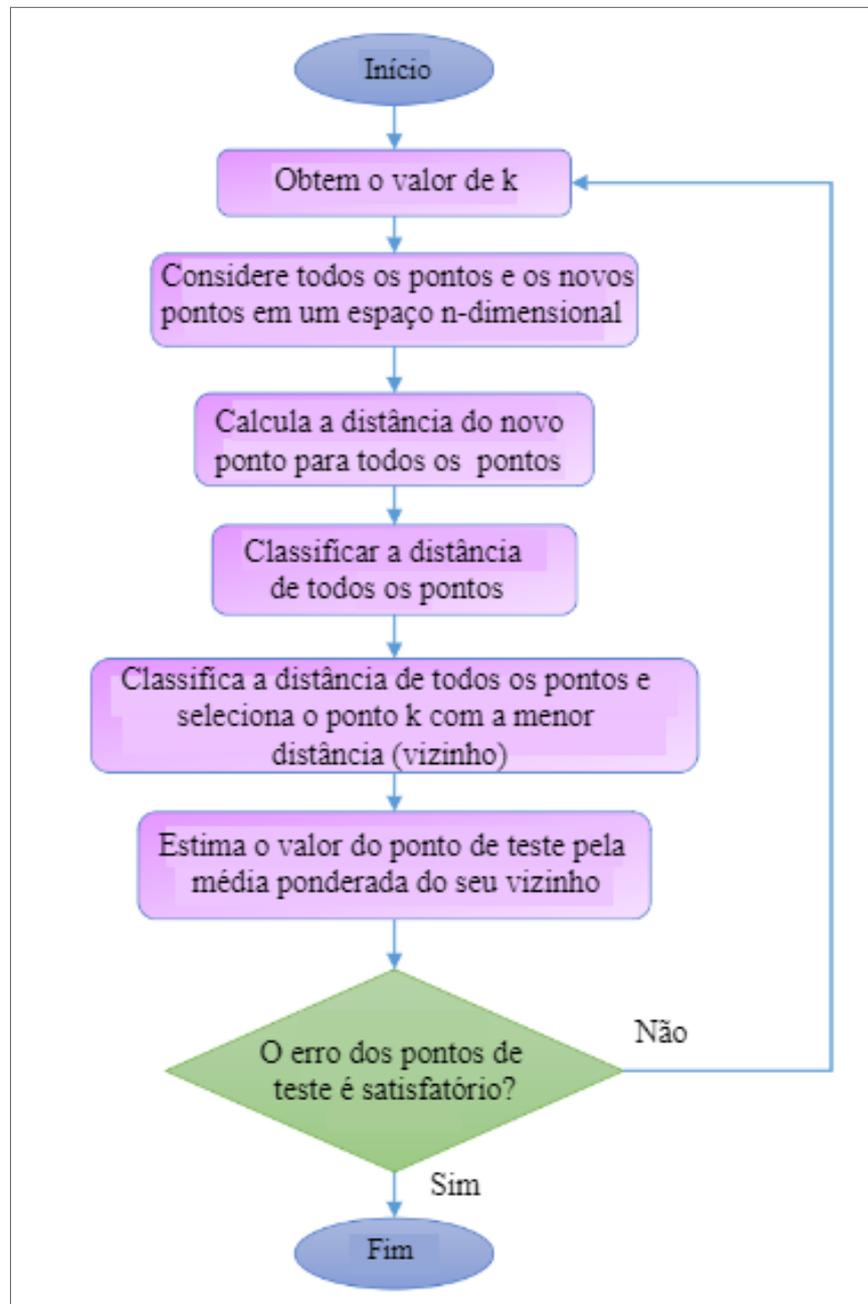
### 2.2.1 *K-Nearest Neighbors* (K-NN)

O K-NN é um modelo de classificação que funciona localizando os  $K$  pontos de dados na amostra de treinamento mais semelhantes a um novo ponto de dados, formando uma "região de vizinhos mais próximos" e fazendo uma previsão com base nessa região. No caso de uma tarefa de classificação, para cada novo ponto de dados, o K-NN atribui uma classe determinando a classe majoritária entre os vizinhos mais próximos (ABED et al., 2023; FANG et al., 2021). A Figura 6 mostra o fluxo do processo de aprendizagem do K-NN.

O K-NN é um classificador de fácil entendimento e implementação, pois não necessita de treinamento, podendo ser adaptado para vários tipos de cenários e dados por não ser paramétrico. Por outro lado, o processo de aprendizagem pode ser lento e demorado por calcular a distância entre o novo ponto e todos os outros pontos do conjunto. Conforme mostrado na Figura 7, o valor de  $k$  pode alterar o desempenho do modelo, é possível notar que o valor do *Mean Absolute Percent Relative Error* (MAPRE) (%) sofre alterações para cada valor de  $k$ , tendo os melhores resultados entre 0 e 100, mas ainda possui oscilações dentro

dessa faixa de valores.

Figura 6 – Fluxograma do modelo K-NN



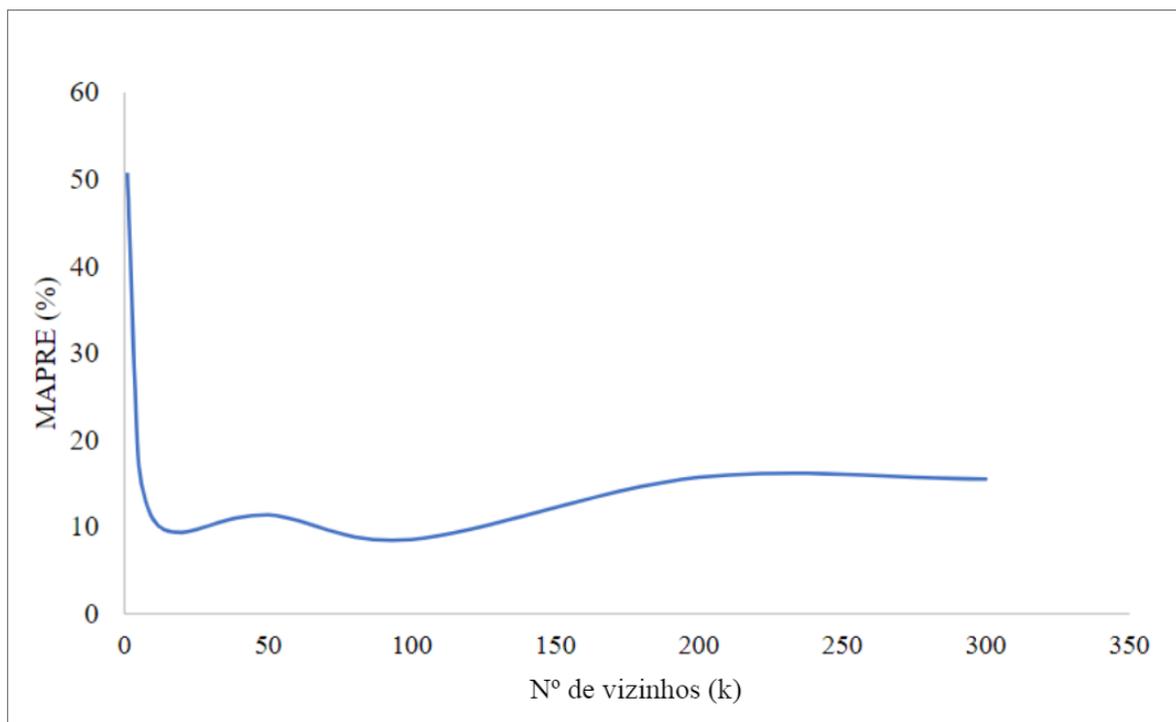
Fonte: (MAHDIANI M.R., 2020)

### 2.2.2 Árvore de Decisão

A Árvore de Decisão é um modelo de aprendizado que usa uma abordagem baseada em regras para construir uma estrutura de árvore binária para tomada de decisão. Uma Árvore de Decisão mapeia um domínio de dados para um conjunto de respostas dividindo recursivamente

o domínio em subdomínios, garantindo que cada divisão obtenha informações mais significativas do que o nó original (SUTHAHARAN; SUTHAHARAN, 2016). A estrutura final de uma Árvore de Decisão consiste em nós de decisão e nós folha. A Árvore de Decisão é um modelo de caixa branca para classificação, oferecendo a vantagem da transparência em seu processo de tomada de decisão. Essa transparência é obtida pela interpretação das regras da árvore, que revelam o caminho lógico por trás de cada previsão (JAMEHBOZORG; SHAHRTASH, 2010; NAVADA et al., 2011). A Figura 8 apresenta uma árvore de decisão utilizada por (SUTHAHARAN; SUTHAHARAN, 2016), neste exemplo é possível realizar duas análises: na ramificação mais a esquerda, deverá ser gerada mais uma regra, visto que ainda instâncias pertencentes às duas classes em estudo; e a ramificação mais a direita não deveria ter sido realizada, já que todos os casos são contemplados pela regra.

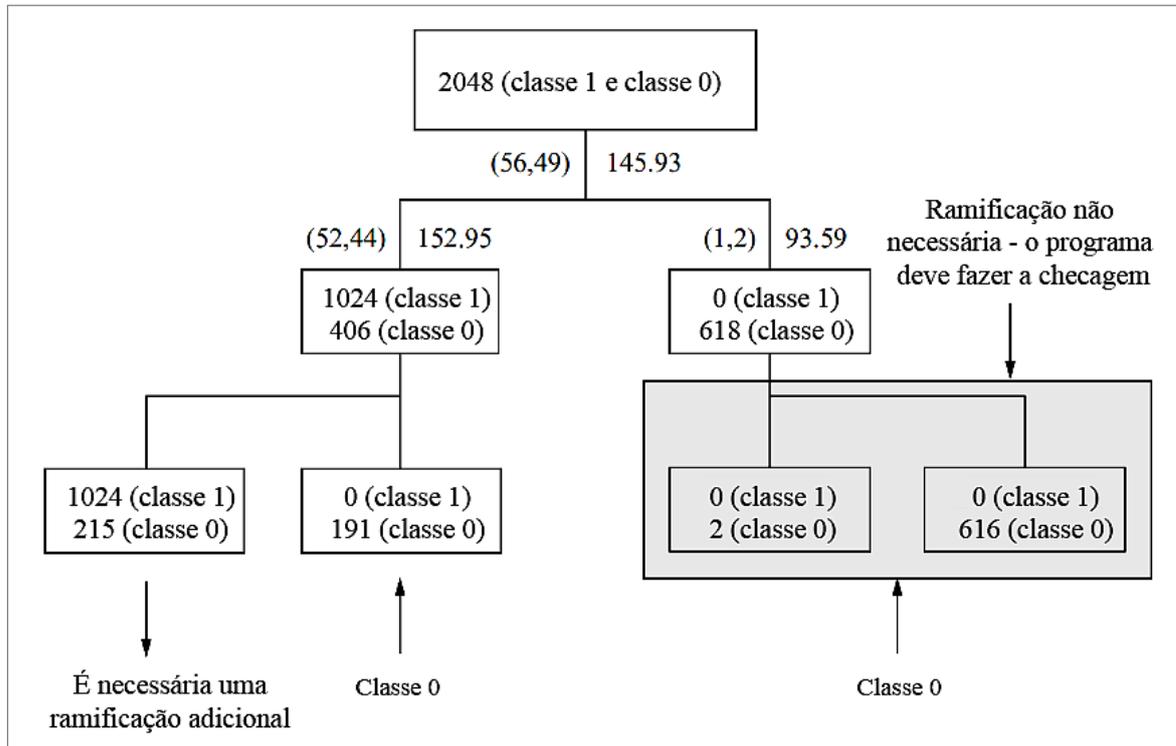
Figura 7 – Variação do MAPRE (%) com relação ao valor de  $k$



Fonte: (MAHDIANI M.R., 2020)

A Árvore de Decisão é um modelo caixa branca de fácil interpretação e simples visualização, não exige normalização dos dados ou codificação de variáveis categóricas, se tornando uma excelente alternativa para classificação de bancos de dados simples e menores, onde poderá fornecer regras interpretáveis. Contudo este classificador possui tendência a sofrer de *overfitting* quando os dados são complexos, a árvore se torna muito profunda e acaba se ajustando aos ruídos, outro problema de árvores muito profundas é que sua interpretabilidade será perdida.

Figura 8 – Árvore de decisão com dois níveis de dados



Fonte: (SUTHAHARAN; SUTHAHARAN, 2016)

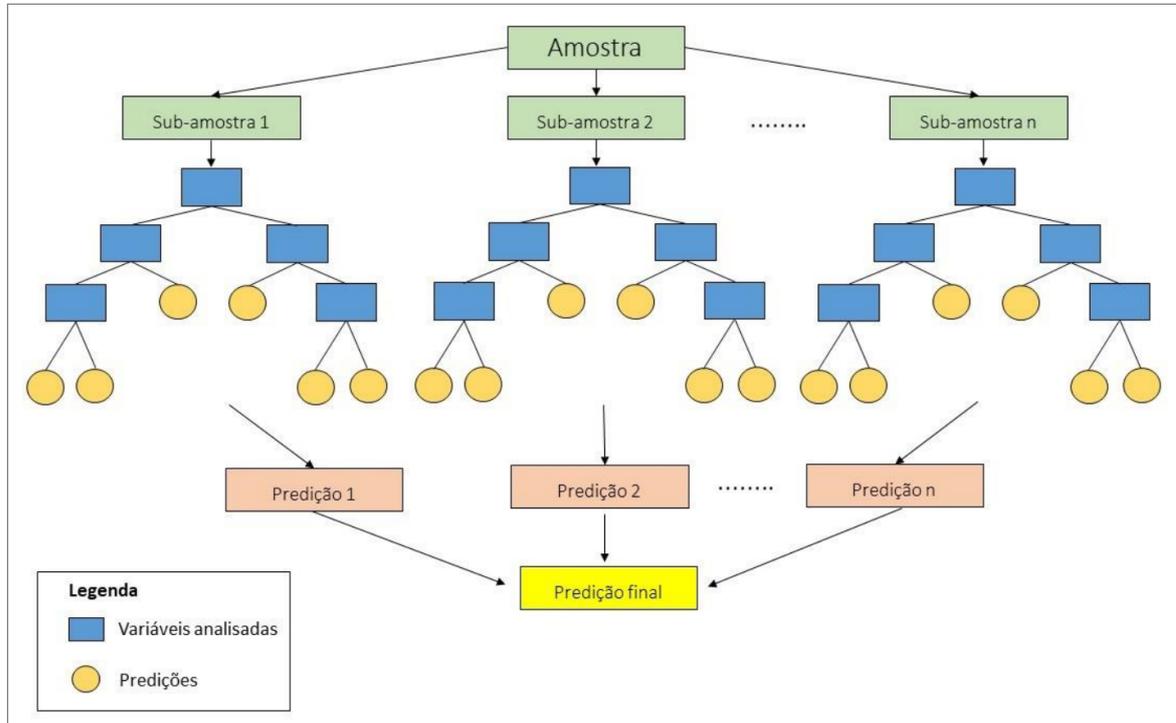
### 2.2.3 Random Forest

*Random Forest* é um modelo de aprendizado composto por várias árvores de decisão. Durante o treinamento, cada árvore é construída em um subconjunto diferente dos dados de treinamento, selecionados usando a técnica de *bagging* (amostragem aleatória com substituição). Este processo introduz diversidade entre as árvores, produzindo modelos independentes e não correlacionados, o que ajuda a reduzir o *overfitting* e melhora a generalização (MRABET et al., 2022). Na fase de teste, a classificação final é determinada por uma votação majoritária de todas as árvores, com a classe que recebe mais votos selecionada como a previsão final (VISWAVANDYA; PATEL; SAHOO, 2021). A Figura 9 apresenta o processo seguido pelo modelo *Random Forest*, pode-se notar as várias que são formadas durante o fluxo de aprendizagem.

A utilização de múltiplas árvores pode favorecer a redução do *overfitting* em dados complexos, outra característica positiva desse tipo de abordagem é a redução de *outliers*, quando comparada com modelos que utilizam apenas uma árvore de decisão. Entretanto, o *Random Forest* é naturalmente mais complexo computacionalmente do que modelos individuais de árvore de decisão, necessitando de maior recurso computacional, sobretudo para bases de dados

maiores, grandes conjuntos de dados também podem causar perda de explicabilidade, já que, quanto maior o número de árvores, mais difícil se torna a interpretação de suas regras.

Figura 9 – Modelo *Random Forest* composto por múltiplas árvores de decisão. A predição final resulta da predição mais frequente em cada árvore



Fonte: (LANA, 2022)

#### 2.2.4 Modelos baseados em *Gradient Boosting*

Os modelos baseados em *Gradient Boosting* criam um conjunto usando a técnica de *boosting*, onde novos classificadores são treinados com base nos resíduos do modelo atual. Esses modelos empregam descida de gradiente para minimizar a função de perda  $L(y, \hat{y})$  adicionando iterativamente modelos que corrigem os resíduos do conjunto combinado (FRIEDMAN, 2001). O gradiente da função de perda em relação às previsões atuais  $\hat{y}$  é calculado como:

$$g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i} \quad (2.1)$$

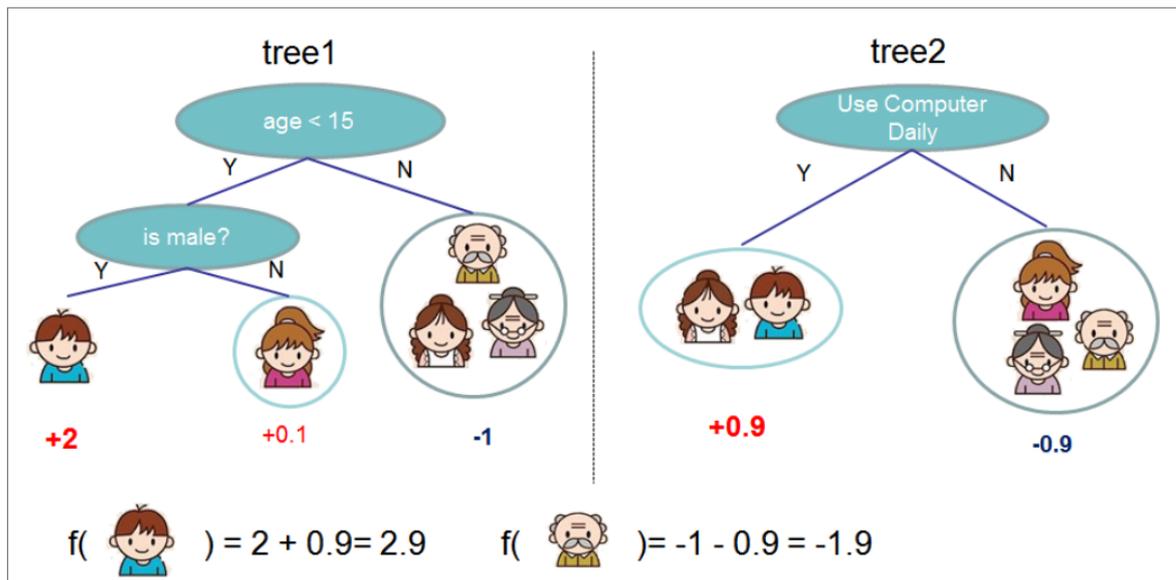
Além do gradiente, o *XGBoost* e o *LightGBM* utilizam a segunda derivada (Hessiana) para ajuste mais preciso:

$$h_i = \frac{\partial^2 L(y_i, \hat{y}_i)}{\partial \hat{y}_i^2} \quad (2.2)$$

### 2.2.4.1 Extreme Gradient Boosting (XGBoost)

O *XGBoost* foi projetado para otimizar a velocidade computacional e o desempenho do modelo. Este modelo adiciona termos de regularização para controlar a complexidade do modelo, o que é útil para evitar o ajuste excessivo e melhorar a generalização do modelo. O *XGBoost* é um sistema de aprendizado de máquina escalável baseado em *boosting* de árvores, sendo uma técnica eficaz que gera resultados satisfatórios em diversos desafios de classificação. (CHEN; GUESTRIN, 2016).

Figura 10 – Exemplo do modelo *XGBoost*. A previsão final para um dado exemplo é a soma das previsões de cada árvore.



Fonte: (CHEN; GUESTRIN, 2016)

O *XGBoost* lida com todos os padrões de esparsidade de forma unificada, indo além, este modelo explora a esparsidade para converter a complexidade da computação linear no número de entradas não ausentes na entrada, desse modo quando a esparsidade dos dados é introduzida no algoritmo, ele é executado até 50 vezes mais rápido. O *XGBoost* incorpora um modelo regularizado para evitar *overfitting*. Isso se assemelha ao modelo *Random Forest*, mas simplifica o objetivo e o algoritmo para paralelização. A amostragem de coluna é uma técnica simples, mas eficaz, emprestada do *Random Forest*. Embora o aprendizado com reconhecimento de esparsidade seja essencial em outros tipos de modelos, como modelos lineares, poucos trabalhos sobre aprendizado em árvore consideraram esse tópico de forma fundamentada. O *XGBoost* foi a primeira abordagem unificada para lidar com todos os tipos de padrões de esparsidade (OGAR; HUSSAIN; GAMAGE, 2022).

A etapa de treinamento do *XGBoost* começa com a definição de uma função de perda que mede a diferença entre as previsões do modelo e os valores reais dos dados. Em cada iteração, uma nova árvore é adicionada ao modelo para minimizar essa perda. A função objetivo do *XGBoost* inclui um termo de regularização, que penaliza a complexidade do modelo e evita o *overfitting*. A regularização é aplicada por meio de penalidades nos pesos das folhas da árvore, ajudando a suavizar as previsões e aumentar a capacidade de generalização do modelo. Para otimizar o processo de aprendizado, o *XGBoost* utiliza uma abordagem de segunda ordem na minimização da função de perda. Isso significa que, além de considerar o gradiente da função de perda (primeira derivada), ele também incorpora a segunda derivada (Hessiana), permitindo uma melhor convergência e tomada de decisão mais precisa nas divisões da árvore. Com isso, cada árvore é construída tentando minimizar uma função de perda aproximada que considera os gradientes e hessianos das previsões atuais (CHEN; GUESTRIN, 2016).

O *XGBoost* é altamente escalável, funcionando bem em grandes conjuntos de dados, seu algoritmo é otimizado para lidar com dados esparsos, facilitando sua aplicação em dados reais, outra vantagem é o suporte a computação paralela e distribuída, que aumenta a velocidade de treinamento em ambientes com múltiplos núcleos. Contudo, este modelo também apresenta algumas desvantagens, apesar de possuir um bom desempenho para grandes bancos de dados, ele ainda poderá necessitar de alto custo computacional para buscar exatidão em dados complexos. Como possui muitos hiperparâmetros, é necessário ajustar os valores de profundidade de árvore e taxa de aprendizado, por exemplo, para que o modelo retorne os melhores resultados. O *XGBoost* também tende a perder interpretabilidade ao combinar muitas árvores de decisão.

#### 2.2.4.2 *Light Gradient Boosting Machine (LightGBM)*

O *LightGBM* é semelhante ao *XGBoost*, mas emprega uma estratégia distinta de crescimento de árvore por folha. Este método permite que o *LightGBM* desenvolva árvores de uma maneira que reduza a perda de forma mais eficaz, geralmente resultando em tempos de treinamento mais rápidos e precisão aprimorada. Em (PAN, 2024) foi proposta uma melhoria no algoritmo do *LightGBM* com a utilização de *grid search*, conforme mostra a Figura 11. A utilização do *grid search* também foi utilizada neste estudo e os valores estarão descritos no Capítulo 4.

Em (KE et al., 2017), o autor propõe duas técnicas para aprimorar o *LightGBM*: O *Gradient-*

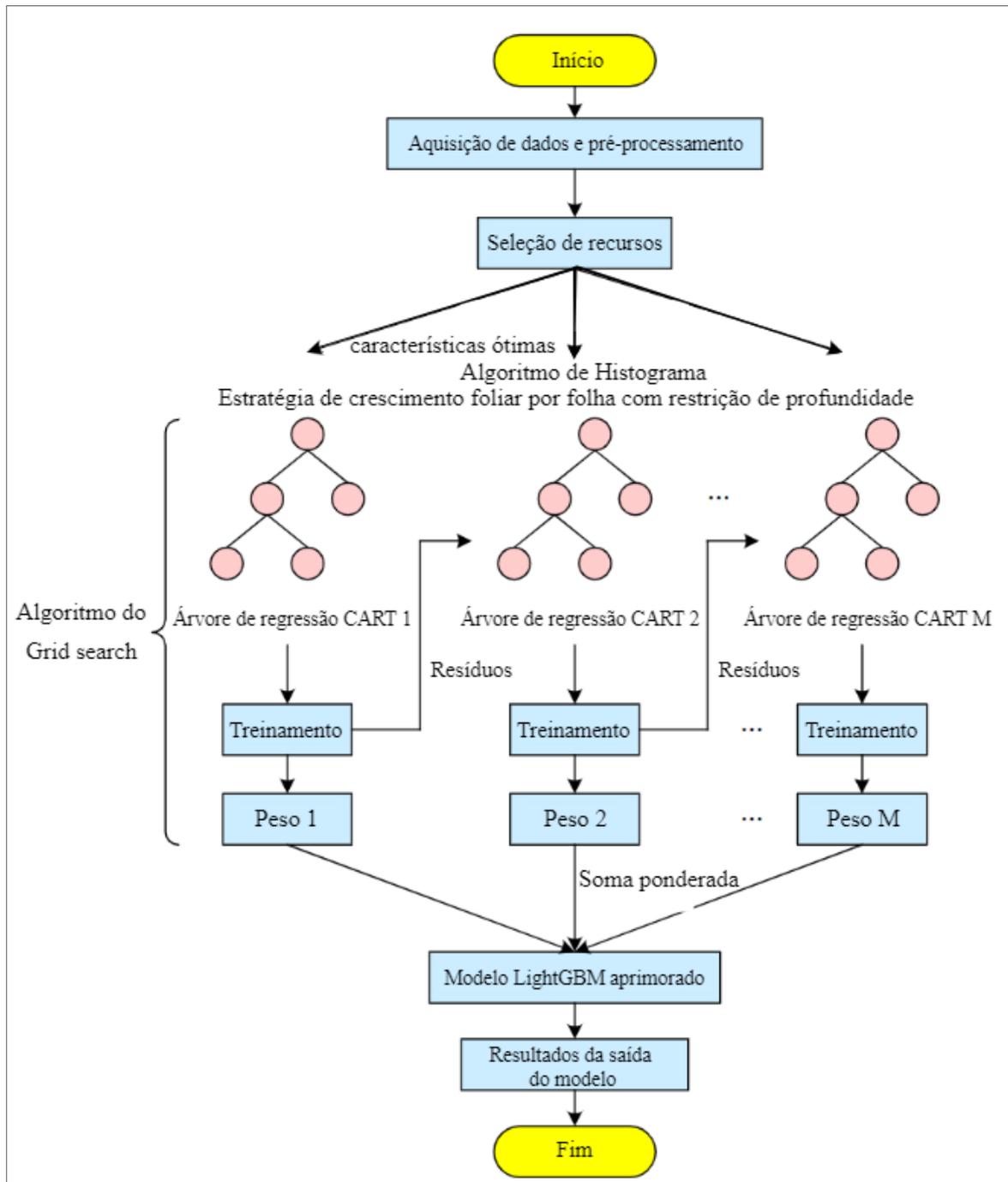
*based One-Side Sampling* (GOSS), utilizado para reduzir o número de instâncias usadas no treinamento, ele mantém as instâncias mais difíceis de treinar e amostra aleatoriamente as mais fáceis, focando assim, nas mais relevantes. O *Exclusive Feature Bundling* (EFB), que agrega características mutuamente exclusivas em uma única, reduzindo o número de atributos que serão processados sem perda de informação relevante. O EFB evita que o modelo realize cálculos desnecessários em dados onde muitos atributos têm valor zero.

O algoritmo completo do *LightGBM*, exemplificado na figura 11, é iniciado pela etapa de pré-processamento, onde o EFB é aplicado para redução de características e os histogramas para quantização das características são construídos. Em seguida as árvores são treinadas, em cada iteração o gradiente de dados é calculado, o GOSS seleciona as instâncias relevantes e os histogramas são usados para encontrar as melhores divisões. Logo após a criação das árvores, o modelo é atualizado com a adição sequencial das árvores e a previsão é atualizada para reduzir o erro residual. Por fim, o desempenho do modelo é avaliado e ajustado com a busca de hiperparâmetros (KE et al., 2017).

Mesmo com as vantagens citadas anteriormente, o *LightGBM* pode ser menos eficiente em bases de dados pequenas, se comparado com o *CatBoost*, que utiliza ordenação e técnicas de redução de vazamento de alvo. O GOSS e o EFB aumentam a eficiência, mas necessitam de ajuste preciso dos seus hiperparâmetros, onde pequenos desvios podem impactar negativamente o modelo. Embora seja eficiente e amplamente usado, o *XGBoost* pode ser mais lento e consumir mais memória do que o *LightGBM*, especialmente em bases de dados maiores. O *LightGBM* pode acelerar o processo de treinamento, utilizando histogramas e o GOSS.

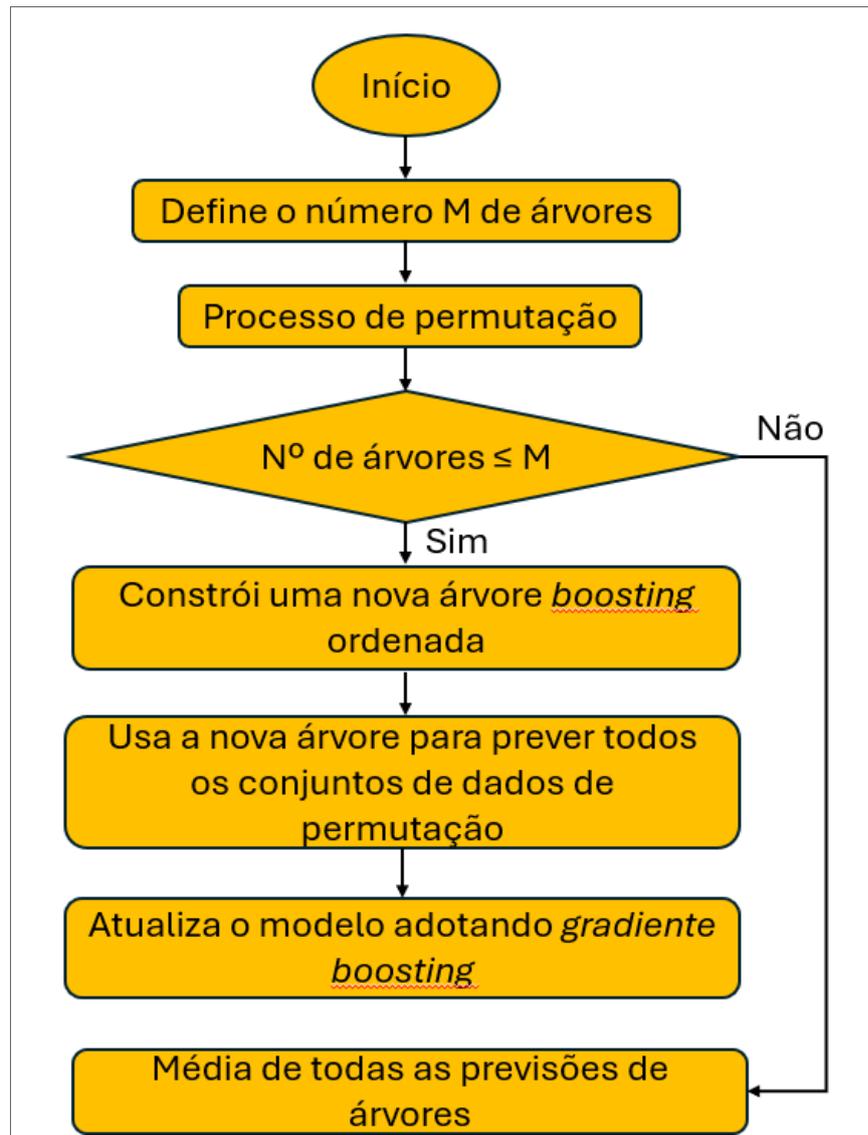
#### 2.2.4.3 *CatBoost*

O *CatBoost* é um classificador que simplifica a preparação de dados ao manipular efetivamente valores ausentes para variáveis numéricas e variáveis categóricas não codificadas, reduzindo a necessidade de pré-processamento extensivo. Ao contrário do *XGBoost* e do *LightGBM*, que exigem codificação manual de recursos categóricos, o *CatBoost* processa esses recursos nativamente, levando a um desempenho potencialmente melhor e implementação mais direta. A Figura 12 mostra o fluxograma do processo de modelagem do modelo *CatBoost*.

Figura 11 – Fluxograma do modelo de previsão *LightGBM* com a inserção do *grid search*

Fonte: (PAN, 2024)

O modelo de aprendizado do *CatBoost* é baseado em árvores de decisão obliviosas, que são árvores balanceadas onde todas as divisões em um mesmo nível usam a mesma regra de divisão. Durante o treinamento, o algoritmo constrói essas árvores de forma iterativa, minimizando a função de perda com base em gradientes de segunda ordem, ou seja, utilizando tanto o gradiente da função de erro quanto sua derivada de segunda ordem para encontrar os melhores ajustes.

Figura 12 – Fluxograma do modelo *Catboost*

Fonte: Adaptado de (YALÇIN et al., 2023)

O funcionamento do *CatBoost* inicia-se com a preparação dos dados, onde as variáveis categóricas são processadas. Cada categoria recebe uma estimativa baseada apenas nos exemplos que a precederam em uma permutação aleatória dos dados de treinamento. Dessa forma, garante-se que nenhuma instância é influenciada por sua própria informação no cálculo da estatística, evitando vazamento de alvo. Após o pré-processamento das variáveis categóricas, o modelo é treinado. Em implementações tradicionais de *boosting*, o modelo é treinado usando os mesmos dados para calcular os gradientes e para ajustar as árvores, resultando em um viés nas previsões. O *CatBoost* evita esse problema ordenando os dados de forma aleatória e criando várias permutações. Em cada iteração do treinamento, o *CatBoost* executa os seguintes passos: primeiro, ele calcula os resíduos das previsões atuais, determinando o erro entre

---

os valores reais e as previsões anteriores. Em seguida, as novas árvores são construídas para minimizar esse erro, depois que as novas árvores são adicionadas ao modelo, uma taxa de aprendizado é aplicada para evitar que o modelo se ajuste excessivamente aos dados de treinamento, permitindo uma convergência mais estável. Durante a fase de inferência, o modelo treinado é aplicado a novos dados para gerar previsões. As variáveis categóricas nos novos dados são processadas de maneira consistente com o treinamento, garantindo que as mesmas transformações sejam aplicadas para preservar a qualidade das previsões (PROKHORENKOVA et al., 2019).

O *CatBoost* consegue lidar com modelos categóricos diretamente, sem a necessidade de conversões complexas, e assim como o *XGBoost*, também retorna bons resultados para dados esparsos, uma vantagem do *CatBoost* em relação aos modelos *XGBoost* e *LightGBM* é a menor suscetibilidade ao *overfitting*, pois possui a característica de cada nível utilizar o mesmo nível de divisão, o que não ocorre para os outros dois classificadores, que utilizam árvores mais complexas. Entretanto o *CatBoost* possui limitação de hiperparâmetros ajustáveis, isto pode se tornar uma limitação por não possuir flexibilização de ajustes, o *CatBoost* também pode ser mais lento que o *LightGBM* em determinadas aplicações, visto que suas múltiplas permutações de dados durante a fase de treinamento aumentam sua eficácia, mas elevam o período de processamento (PROKHORENKOVA et al., 2019).

O *CatBoost* é mais eficiente em manipular dados com atributos categóricos e em contornar erros de vazamento de alvo. Embora o *LightGBM* seja mais rápido em grandes bases de dados, o *CatBoost* pode ser mais eficaz quando os dados contém muitas variáveis categóricas complexas, onde a ordenação das características é crucial para evitar enviesamento. Por fim, o *LightGBM* é mais indicado para bancos de dados grandes, pois é mais eficiente no uso da memória, já o *CatBoost* leva vantagem na precisão de bases de dados menores e com variáveis categóricas e o *XGBoost* produz melhores resultados em aplicações que necessitam de divisões precisas.

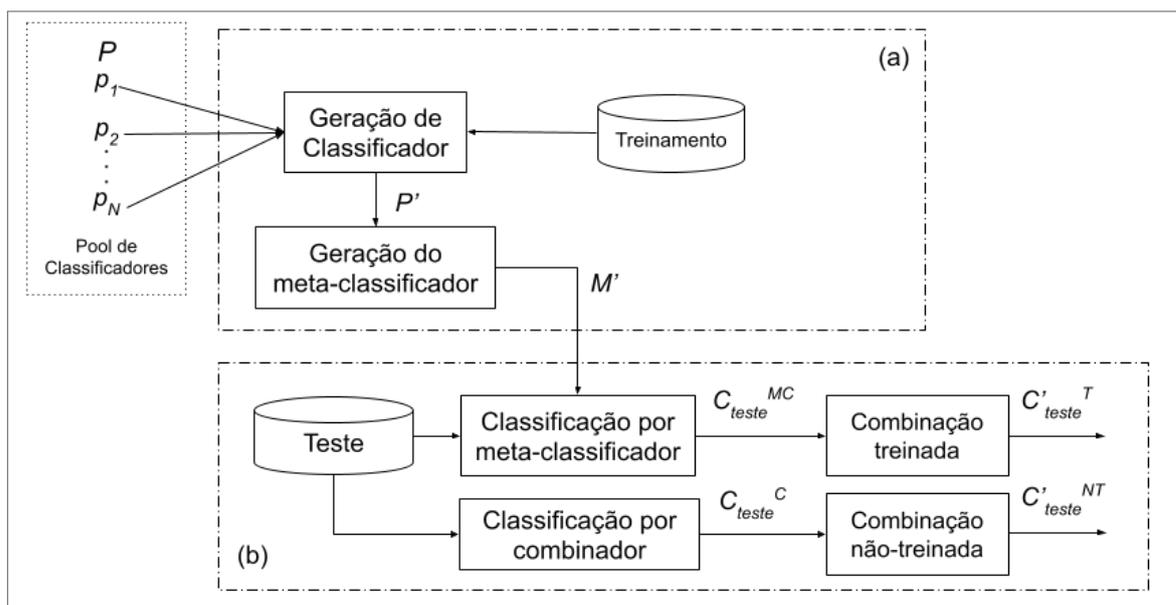
### 3 MÉTODO PROPOSTO

Neste trabalho, foram avaliados diferentes abordagens de MCS, utilizando seleção estática e dinâmica, para tarefas de detecção de falhas elétricas. Este trabalho não abordou modelos mais recentes de Redes Neurais ou *Deep Learning*, devido à baixa explicabilidade e interpretabilidade desses modelos, as técnicas baseadas em ML utilizadas na pesquisa produzem regras que podem ser interpretadas pela operação dos sistemas elétricos, onde os operadores não são especialistas em sistemas inteligentes de computação.

#### 3.1 SELEÇÃO ESTÁTICA

Na Seleção Estática, Figura 13 e Algoritmo 1, todo o conjunto de classificadores do *pool* é considerado para a classificação final. O MCS desenvolvido neste trabalho consiste em duas fases: (a) treinamento e (b) teste. Na fase de treinamento (a), um conjunto de classificadores  $P = \{p_1, p_2, \dots, p_n\}$ , onde  $n$  é o tamanho do *pool*, é treinado usando o conjunto de dados de treinamento, resultando em um *pool*  $P'$ . Em modelos que utilizam a generalização empilhada, o *pool*  $P'$  é a entrada a geração do meta-classificador, este meta-classificador é gerado a partir de um algoritmo de classificação, como Árvore de Decisão ou Regressão Logística.

Figura 13 – Uma estrutura geral do MCS, ou *Ensemble Learning*, empregando Seleção Estática. Esta abordagem seleciona um ou mais modelos para classificar todos os padrões de teste.



Fonte: Elaborada pelo autor (2024)

---

**Algorithm 1** Seleção Estática de Classificadores
 

---

**Require:**  $D_{treino}$  (dados de treinamento),  $D_{teste}$  (dados de teste)

**Ensure:**  $C'_{teste}T$  (previsão com combinação treinada),  $C'_{teste}NT$  (previsão com combinação não treinada)

1: **Fase (a): Treinamento**

2: **function** TREINAMENTO( $D_{treino}$ )

3:    $P \leftarrow$  Gerar\_Classificadores( $D_{treino}$ )

4:    $P' \leftarrow P$

5:    $M' \leftarrow$  Gerar\_Meta\_Classificador( $P', D_{treino}$ )

6:   **return** ( $P', M'$ )

7: **end function**

8: **Fase (b): Teste**

9: **function** TESTE( $D_{teste}, P', M'$ )

10:    $C_{teste}MC \leftarrow$  Classificar\_Usando\_Meta\_Classificador( $M', D_{teste}$ )

11:    $C_{teste}C \leftarrow$  Classificar\_Usando\_Combinador( $P', D_{teste}$ )

12:    $C'_{teste}T \leftarrow$  Combinação\_Treinada( $C_{teste}MC$ )

13:    $C'_{teste}NT \leftarrow$  Combinação\_Não\_Treinada( $C_{teste}C$ )

14:   **return** ( $C'_{teste}T, C'_{teste}NT$ )

15: **end function**

16: **Execução do Processo**

17: ( $P', M'$ )  $\leftarrow$  Treinamento( $D_{treino}$ )

18: Resultado  $\leftarrow$  Teste( $D_{teste}, P', M'$ )

19: **return** Resultado

---

Na fase de teste (b), o pool  $P'$ , cada nova instância de teste é passada para o classificador, que pode ser o meta-classificador ou um combinador estático, como o voto majoritário, que retornará  $C_{teste}MC$  ou  $C_{teste}C$ , respectivamente. A classificação por meta-classificador é considerada uma combinação treinada e tem na sua saída a classificação  $C'_{teste}T$ , já a classificação por combinador estático, como voto majoritário, não possui etapa de treinamento e é considerada uma combinação não treinada  $C'_{teste}NT$ .

### 3.1.1 Voto Majoritário

O método de Voto Majoritário, introduzido por (BREIMAN, 1996), consiste em cada classificador base prevendo uma classe para uma nova instância de dados, com a predição final determinada pela classe que recebe o maior número de votos. (AEENEH; ZLATANOV; YU, 2024) propõe um modelo, em que o problema de classificação multiclasse, para modelos instáveis, onde pequenas alterações nos dados de treinamento podem levar a variações significativas nas predições, a taxa de erro do Voto Majoritário decai exponencialmente em direção a zero à medida que o número de classificadores independentes aumenta. Por outro lado, a taxa de

erro do Voto Majoritário cresce exponencialmente com o número de eleitores independentes se essas condições não forem atendidas.

### 3.1.2 Generalização Empilhada

A generalização empilhada, introduzida por (WOLPERT, 1992), envolve o treinamento de um meta-classificador para combinar as previsões dos classificadores base. Este meta-classificador é treinado usando as saídas dos modelos base na amostra de treinamento. Durante a inferência, para um novo padrão, as previsões dos modelos base são usadas como entradas para o meta-classificador, que então as combina para retornar a previsão final. Este método permite que o meta-classificador aproveite efetivamente os pontos fortes de diversos modelos base, levando a um melhor desempenho preditivo e robustez quando esses modelos exibem alta variabilidade. As técnicas de empilhamento conseguem reduzir a variância e o viés de forma simultânea, se destacando de outras de *ensemble*, que funcionam de forma individualizada (HAJIHOSEINLOU; MAGHSOUDI; GHEZELBASH, 2024). Neste trabalho, foram avaliados dois algoritmos como meta-classificadores: Árvore de Decisão e Regressão Logística, ambos usados em generalização empilhada (HAJIHOSEINLOU; MAGHSOUDI; GHEZELBASH, 2024; CUI et al., 2021).

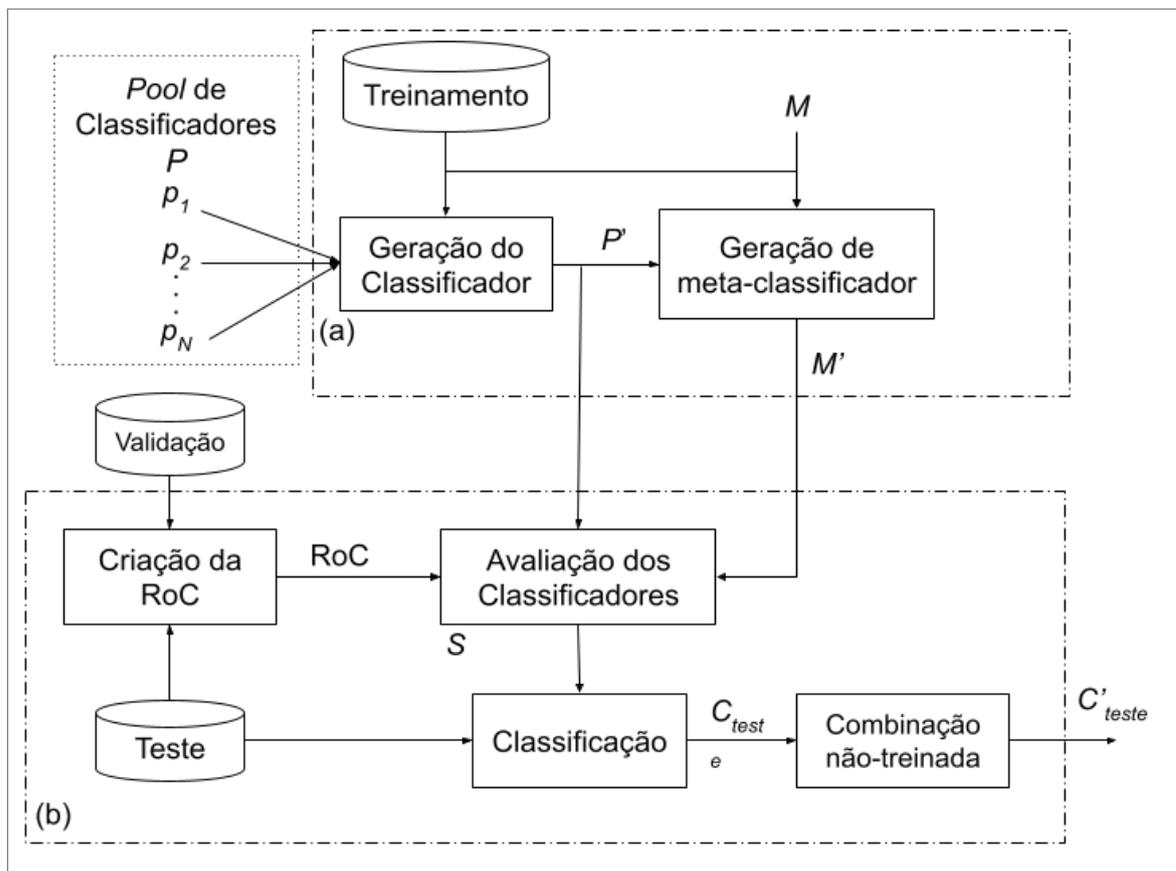
## 3.2 SELEÇÃO DINÂMICA

Na Seleção Dinâmica, Figura 14 e Algoritmo 2, um subconjunto de classificadores do *pool* é selecionado para cada nova instância de teste. Este subconjunto pode ser composto de 1 a  $n$  classificadores, onde  $n$  é o tamanho do *pool*. O MCS desenvolvido nesta parte do trabalho consiste em duas fases: (a) treinamento e (b) teste. A fase de treinamento é composta de duas etapas: Geração do Classificador e Geração do Meta-Classificador. Na etapa Geração do Classificador, semelhante à Seleção Estática, um *pool*  $P'$  de  $n$  classificadores é gerado. A etapa Geração do Meta-Classificador é executada apenas quando a abordagem de Seleção Dinâmica requer um meta-classificador para selecionar o melhor modelo. Nesta etapa, as amostras de treinamento são usadas para extrair meta-características, e o algoritmo  $M$  é aplicado para gerar o Meta-Classificador.

A fase de teste consiste em três etapas, e o caminho para a classificação final depende da abordagem de Seleção Dinâmica escolhida. A primeira etapa é a criação da *Region of*

*Competence* (RoC), que é executada por abordagens dinâmicas que exigem uma Região de Competência. A RoC é criada usando as instâncias de validação mais semelhantes à nova instância de teste, para selecionar o conjunto de modelos com o melhor desempenho para classificar instâncias dentro deste RoC. A segunda etapa é a "Avaliação dos Classificadores", que envolve selecionar o conjunto de modelos com base em critérios específicos. A maioria das abordagens seleciona classificadores com base em seu desempenho dentro da Região de Competência. No entanto, também avaliamos o uso de um Meta-Classificador, que extrai metarecursos da nova instância e seleciona o conjunto de modelos mais adequado para classificá-la. A saída desta etapa é um conjunto  $S$  de modelos selecionados, que pode variar de 1 a  $N$  modelos. Finalmente, na etapa de classificação, a saída de cada modelo no conjunto selecionado é obtida. Se apenas um classificador for selecionado, sua classificação é retornada como a classificação final ( $C'_{teste}$ ). No entanto, se mais de um modelo for selecionado, o  $C'_{teste}$  é obtido por votação majoritária na etapa final.

Figura 14 – Uma estrutura geral do MCS, ou *Ensemble Learning*, empregando Seleção Dinâmica. Esta abordagem seleciona um ou mais modelos de uma Região de Competência para classificar cada padrão do conjunto de teste.



Fonte: Elaborada pelo autor (2024)

Neste trabalho, avaliamos seis algoritmos de seleção dinâmica de última geração: *Overall Local Accuracy* (OLA), *Dynamic Ensemble Selection Performance* (DESP), *K-Nearest Oracles Eliminat* (KNORA-E), *K-Nearest Oracles Union* (KNORA-U), *Multiple Classifier Behavior* (MCB) e *Meta-learning for Dynamic Ensemble Selection* (META-DES).

---

**Algorithm 2** Seleção Dinâmica de Classificadores
 

---

**Require:**  $D_{treino}$  (dados de treinamento),  $D_{val}$  (dados de validação),  $D_{teste}$  (dados de teste)

**Ensure:**  $C'_{teste}$  (predições finais)

```

1: Fase (a): Treinamento
2: function TREINAMENTO( $D_{treino}$ )
3:    $P \leftarrow$  Gerar_Classificadores( $D_{treino}$ )
4:    $P' \leftarrow P$ 
5:    $M' \leftarrow$  Gerar_Meta_Classificadores( $P', D_{treino}$ )
6:   return ( $P', M'$ )
7: end function
8: Fase (b): Teste
9: function TESTE( $D_{val}, D_{teste}, P', M'$ )
10:   $RoC \leftarrow$  Criar_RoC( $P', D_{val}$ )
11:   $S \leftarrow$  Avaliar_Classificadores( $P', D_{teste}, RoC$ )
12:   $C_{teste} \leftarrow$  Realizar_Classificacao( $S$ )
13:   $C'_{teste} \leftarrow$  Combinacao_Nao_Treinada( $C_{teste}, M'$ )
14:  return  $C'_{teste}$ 
15: end function
16: Execução do Processo
17: ( $P', M'$ )  $\leftarrow$  Treinamento( $D_{treino}$ )
18: Resultado  $\leftarrow$  Teste( $D_{val}, D_{teste}, P', M'$ )
19: return Resultado

```

---

### 3.2.1 Overall Local Accuracy (OLA)

A *Dynamic Classifier Selection by Local Accuracy* (DCS-LA), também conhecida como OLA, introduzida por (WOODS; KEGELMEYER; BOWYER, 1997), é um método para selecionar o classificador mais competente para cada amostra de teste com base em estimativas de precisão local dentro de uma RoC. Para cada instância de teste, a RoC, composta pelos  $k$ -vizinhos mais próximos do conjunto de treinamento que são mais semelhantes à nova instância, é criada. Em seguida, o classificador com a maior precisão dentro da RoC é selecionado para retornar a classificação final.

O OLA considera apenas a taxa de acerto local, o que o torna de simples aplicação, sem levar em conta outras métricas, como variância ou confiança. Assim, o modelo se torna eficaz em cenários em que um único classificador apresenta um desempenho melhor na região

local. No entanto, essa facilidade de aplicação também o deixa limitado, pois o método não explora a possibilidade de combinar múltiplos classificadores dentro da RoC ou de usar ponderações mais sofisticadas, e em situações onde os desempenhos dos classificadores variam mais sutilmente, outras técnicas, que serão abordadas ao longo desta seção, podem oferecer resultados mais robustos. Além disso, o desempenho do modelo está relacionado à escolha adequada do parâmetro  $k$  e da métrica de distância utilizada para definir a RoC, podendo variar significativamente conforme as características da base de dados.

### 3.2.2 *Dynamic Ensemble Selection Performance (DESP)*

O método DESP, introduzido por (WOLOSZYNSKI et al., 2012), consiste em selecionar dinamicamente (ou seja, para cada exemplo de teste) um subconjunto de classificadores do conjunto que apresentam melhor desempenho do que um classificador aleatório. Portanto, classificadores fracos, que afetariam negativamente o desempenho de um sistema de classificação são eliminados. O desempenho do RC é definido como  $RC = \frac{1}{L}$ , onde  $L$  é o número de classes no problema. Caso todos os classificadores no conjunto sejam avaliados com baixo desempenho, a precisão da classificação do sistema pode ser aumentada usando um classificador aleatório, porém (WOLOSZYNSKI et al., 2012) justifica teoricamente a utilização do Voto Majoritário.

O DESP permite uma seleção eficiente dos classificadores, eliminando aqueles que têm maior probabilidade de gerar ruído e reduzir o desempenho do sistema, essa característica garante que a classificação final seja baseada em modelos que apresentem, ao menos, um desempenho superior ao acaso, o que pode ser útil em problemas onde há um número elevado de classificadores fracos. Por outro lado, uma limitação dessa técnica é que ele depende da escolha correta do limiar de desempenho para eliminar os classificadores. Da mesma forma, a técnica pode não apresentar bons resultados em cenários onde o número de classes é elevado, pois o valor de RC se torna muito pequeno, o que pode tornar difícil distinguir classificadores competentes dos mais fracos. Outrossim, ao descartar classificadores, o sistema pode acabar perdendo a diversidade que é essencial em MCS.

### 3.2.3 *K-nearest-oracles (KNORA)*

Os modelos *K-Nearest Oracles* (KNORA), conforme proposto por (KO; SABOURIN; BRITTO, 2008), buscam escolher o melhor classificador, selecionando o conjunto mais adequado para cada amostra.

O conceito dos KNORA é similar ao proposto pelo OLA ao buscar a melhor vizinhança de padrões de teste, porém distingui-se pelo uso direto de sua propriedade de ter amostras de treinamento na região com as quais encontra o melhor conjunto para uma determinada amostra. Para qualquer ponto de dados de teste, o KNORA simplesmente encontra seus *K*-vizinhos mais próximos no conjunto de validação, descobre quais classificadores classificam corretamente esses vizinhos no conjunto de validação e os usa como o conjunto para classificar o padrão fornecido naquele conjunto de teste (KO; SABOURIN; BRITTO, 2008). Baseado nos modelos KNORA, (KO; SABOURIN; BRITTO, 2008) propôs os métodos KNORA-E e KNORA-U.

#### 3.2.3.1 *K-Nearest Oracles Eliminate (KNORA-E)*

KNORA-E consiste em selecionar classificadores que classificam corretamente todas as instâncias dentro do RoC. Se nenhum classificador classificar corretamente todas as amostras nessa região, o tamanho da região será reduzido até que pelo menos um classificador atenda ao critério. Se ainda assim nenhum classificador atender ao critério, o conjunto completo será utilizado. A decisão final é tomada usando o método do Voto Majoritário. Essa abordagem é bastante rigorosa, garantindo que apenas os classificadores que apresentam o melhor desempenho na RoC sejam considerados, essa característica faz com que o KNORA-E seja eficaz em garantir que os piores classificadores sejam excluídos, melhorando a precisão final. No entanto, sua exigência de acurácia total pode limitar sua aplicabilidade em cenários onde as instâncias são ruidosas ou muito complexas, uma vez que nem sempre é possível encontrar classificadores que atinjam tal nível de acerto.

#### 3.2.3.2 *K-Nearest Oracles Union (KNORA-U)*

KNORA-U envolve selecionar classificadores que classificam corretamente pelo menos uma instância dentro da RoC. Cada classificador selecionado recebe um voto ponderado com base no número de instâncias que ele classifica corretamente dentro da RoC. A decisão final é tomada

pela classe que acumula mais votos dos classificadores selecionados. Conforme explicado, o KNORA-U é mais flexível, comparado com o KNORA-E, em relação ao desempenho dos classificadores o que pode ser vantajoso em cenários onde a heterogeneidade das instâncias ou a presença de ruído dificultam que algum classificador tenha um desempenho perfeito. Ao permitir que classificadores parcialmente corretos contribuam para a decisão final, o KNORA-U aumenta a diversidade do conjunto de classificadores selecionados, o que pode ajudar a melhorar a robustez do sistema em situações onde nenhum classificador domina completamente a RoC.

### 3.2.4 *Multiple Classifier Behavior (MCB)*

O método MCB, introduzido por (GIACINTO; ROLI, 2001), consiste em selecionar dinamicamente classificadores com base em sua precisão local dentro de uma RoC, que é determinada usando métricas de similaridade e o conceito de *Behavioral Knowledge Space* (BKS). O comportamento dos classificadores é representado por um vetor MCB, que inclui as previsões de classe feitas por cada classificador para a instância de teste. A similaridade entre o vetor MCB da instância de teste e aqueles de seus vizinhos é então computada. Vizinhos com similaridade acima de um certo limite são usados para refinar o RoC. Finalmente, o classificador com a maior precisão dentro deste RoC refinado é escolhido para fazer a classificação final.

A principal vantagem desse método é sua capacidade de identificar o comportamento conjunto dos classificadores, em vez de avaliar apenas o desempenho individual em relação às amostras de teste, isso possibilita a seleção dos modelos de classificação com base em sua capacidade de trabalhar em conjunto, o que pode solucionar os casos em que a combinação de modelos é importante, trazendo melhores resultados quando há um padrão discernível no comportamento dos classificadores dentro de um espaço definido. Por outro lado, essa abordagem também traz uma complexidade adicional, já que o refinamento da RoC e o cálculo das similaridades requerem um processamento mais intenso, o que pode impactar a escalabilidade do método para grandes conjuntos de dados, o modelo também pode ser menos eficiente quando o comportamento varia de maneira imprevisível ou quando a RoC refinada não é representativa o suficiente.

### 3.2.5 *Meta-learning for Dynamic Ensemble Selection (META-DES)*

O método META-DES, introduzido por (CRUZ; SABOURIN; CAVALCANTI, 2014), consiste em selecionar dinamicamente classificadores com base em múltiplos critérios que avaliam a competência de cada classificador. Na fase de treinamento, um meta-classificador  $M'$  é gerado por meio de um processo de extração de meta-recursos. Esse processo envolve a geração de múltiplos conjuntos de meta-recursos, cada um representando um critério diferente relacionado ao desempenho dos classificadores base, como sua precisão local, consenso em previsões e nível de confiança para a amostra de entrada. Esses meta-recursos são então usados para treinar o meta-classificador, que aprende a prever se um classificador base é competente o suficiente para classificar corretamente novas instâncias de teste. Na fase de testes, para cada instância, o meta-classificador prevê se um modelo base é competente para classificá-la; se mais de um classificador for selecionado, a classificação final é obtida por meio do método do Voto Majoritário.

O META-DES é vantajoso por levar em consideração uma variedade de informações para tomar decisões sobre a seleção de classificadores. Enquanto alguns métodos de seleção dinâmica se baseiam em um único critério (como a precisão local), o META-DES integra múltiplos fatores, como nível de confiança e o consenso das previsões, o que permite uma visão mais abrangente da competência dos classificadores. Esse processo de meta-aprendizado garante que o sistema aprenda a explorar os pontos fortes de diferentes classificadores em situações variadas. No entanto, a principal desvantagem desta técnica é sua complexidade computacional, uma vez que o processo de extração de meta-recursos e o treinamento do meta-classificador envolvem cálculos adicionais. Dependendo da quantidade de classificadores base e da dimensão dos meta-recursos, o tempo de treinamento e teste pode ser muito elevado e tornar inviável sua aplicação, sendo necessário maiores recursos computacionais, tornando-o mais adequado para problemas onde a precisão é mais importante que a eficiência computacional.

## 4 PROTOCOLO EXPERIMENTAL

Este capítulo abordará como as bases de dados foram geradas, analisadas e os procedimentos para inserção e tratamento do ruído, na segunda sessão será abordado o procedimento experimental com os modelos de classificação utilizados no estudo, detalhando os hiperparâmetros escolhidos e a busca pelas melhores combinações de cada modelo.

### 4.1 BASES DE DADOS

O primeiro conjunto de dados foi criado usando o *software Alternative Transient Program Draw* (ATPDraw)<sup>1</sup>, versão 7.0. O ATPDraw é uma ferramenta gráfica de modelagem e simulação de sistemas elétricos, atuando como uma interface para o *software Alternative Transient Program* (ATP), um dos programas mais amplamente utilizados para análises de transitórios eletromagnéticos e eletromecânicos em sistemas elétricos de potência. O ATPDraw permite a construção de circuitos de forma intuitiva e facilita a análise de fenômenos elétricos complexos, como transitórios de manobra, faltas e ressonâncias em redes elétricas. A simulação de faltas em linhas de transmissão e subestações é uma das principais aplicações do ATPDraw, permitindo a análise do comportamento do sistema elétrico sob condições de curto-circuito e outros distúrbios. Essas simulações são essenciais para avaliar a resposta do sistema, dimensionar equipamentos de proteção e garantir a confiabilidade da operação da rede elétrica

A modelagem foi realizada a partir do setor de 138 kV de uma subestação com duas linhas de transmissão conectadas aos seus barramentos, cada linha foi considerada como uma carga de 36 MW e 18 MVar. Vários tipos de curtos-circuitos foram simulados entre disjuntores e saídas de linha, cobrindo faltas fase-terra, fase-fase, fase-fase-terra e trifásicas. As medições de corrente foram feitas no disjuntor de conexão do transformador no barramento principal e nos disjuntores de conexão de linha que compartilham o mesmo barramento. Além disso, a tensão nas saídas de linha foi medida. A simulação foi executada durante 200 segundos, com dados coletados em intervalos de 0,01 segundo, resultando em um banco de dados de 20.000 amostras. Este banco de dados é estruturado da seguinte forma: as colunas características representam medições de tensão nas saídas das duas linhas para as fases A, B e C. Seguindo essas colunas estão as medições de corrente no circuito acima dos disjuntores.

As classes são categorizadas da seguinte forma:

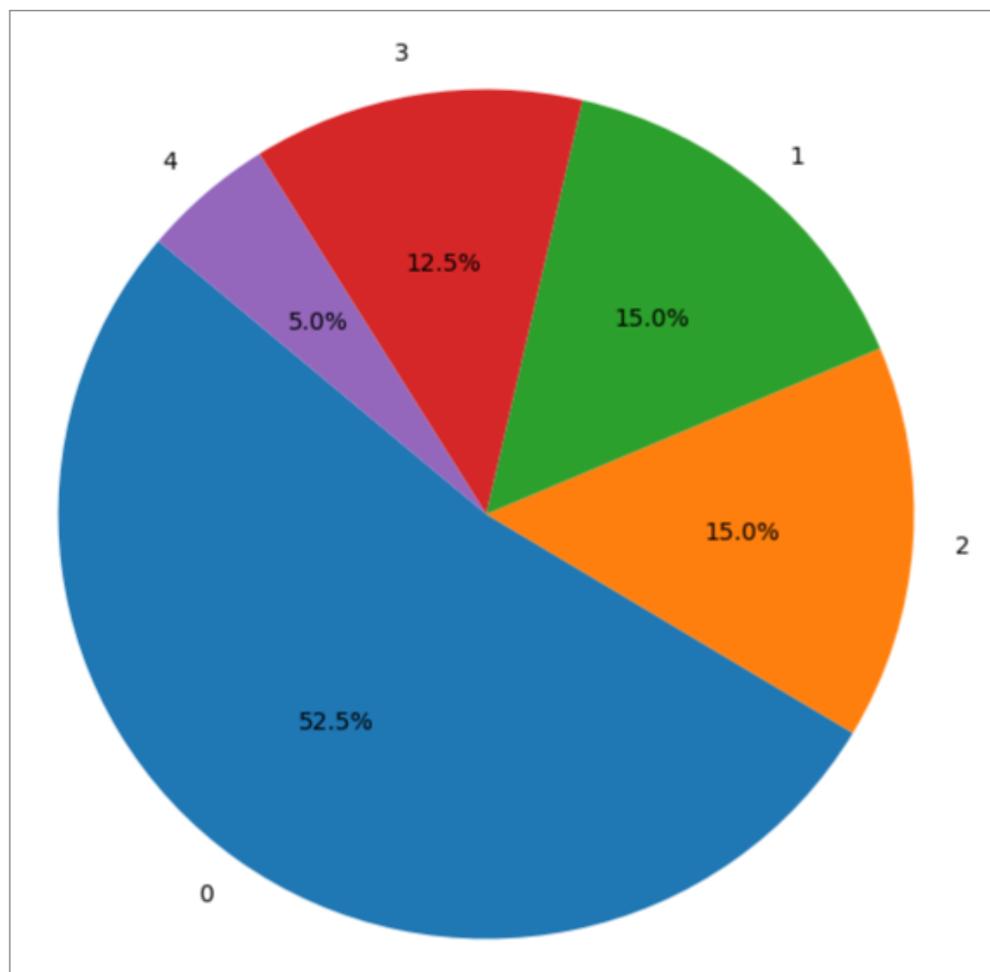
---

<sup>1</sup> <<https://www.atpdraw.net/>>

- Classe 0: Sem falta
- Classe 1: Falta fase-terra
- Classe 2: Falta fase-fase
- Classe 3: Falta fase-fase-terra
- Classe 4: Falta trifásica

A figura 15 apresenta a distribuição das classes do conjunto de dados 01, percebe-se um desbalanceamento comum para dados de sistemas elétricos, em que a quantidade de dados em que o sistema funciona sem falhas é sempre superior aos dados de falhas (MEDEIROS, 2017). A Classe 4 (faltas trifásicas) tem a menor ocorrência, devido à sua natureza, onde as três fases do circuito precisam estar em curto, situação mais atípica em uma rede elétrica.

Figura 15 – Distribuição de classes da Base de Dados 01



Fonte: Elaborada pelo autor (2024)

O segundo banco de dados, extraído de (ENSINA L.A., 2023), foi adaptado considerando apenas as variáveis de energia dos sinais de corrente e tensão, e valores RMS de corrente e tensão também. Como o sistema é trifásico, o banco de dados final contém doze atributos e duas variáveis que podem ser usadas como alvos: "*fault\_location*" e "*fault\_type*". O primeiro é usado para localizar onde a falha ocorreu e tem valores discretos entre 4,14 e 414, cada um representando uma distância em quilômetros. A segunda variável tem valores categóricos indicando as fases em que a falha ocorreu, como AB, BC e BG, as letras A, B e C representam as fases do sistema e G o terra. No presente trabalho, "*fault\_type*" foi usado como alvo. O sistema estudado foi baseado em uma das linhas de transmissão do Sistema de Barramento *Institute of Electrical and Electronics Engineers* (IEEE) 09<sup>2</sup> e simulado no *software* ATPDraw, no entanto, o autor não apresentou o diagrama de simulação. Para este estudo, algumas manipulações foram feitas no banco de dados, como exclusão de atributos que não se aplicavam à análise e normalização dos valores originais, os dados estavam na escala de 10<sup>6</sup>, e foram divididos por este valor, a fim de facilitar sua manipulação. Essas manipulações não causam prejuízo à análise, já que os atributos de corrente e tensão foram mantidos e a variação de valores permaneceu dentro da amplitude que geralmente são medidos em subestações.

As classes da base 2 são categorizadas da seguinte forma:

- Classe 0: Falta AB
- Classe 1: Falta ABC
- Classe 2: Falta ABG
- Classe 3: Falta AC
- Classe 4: Falta ACG

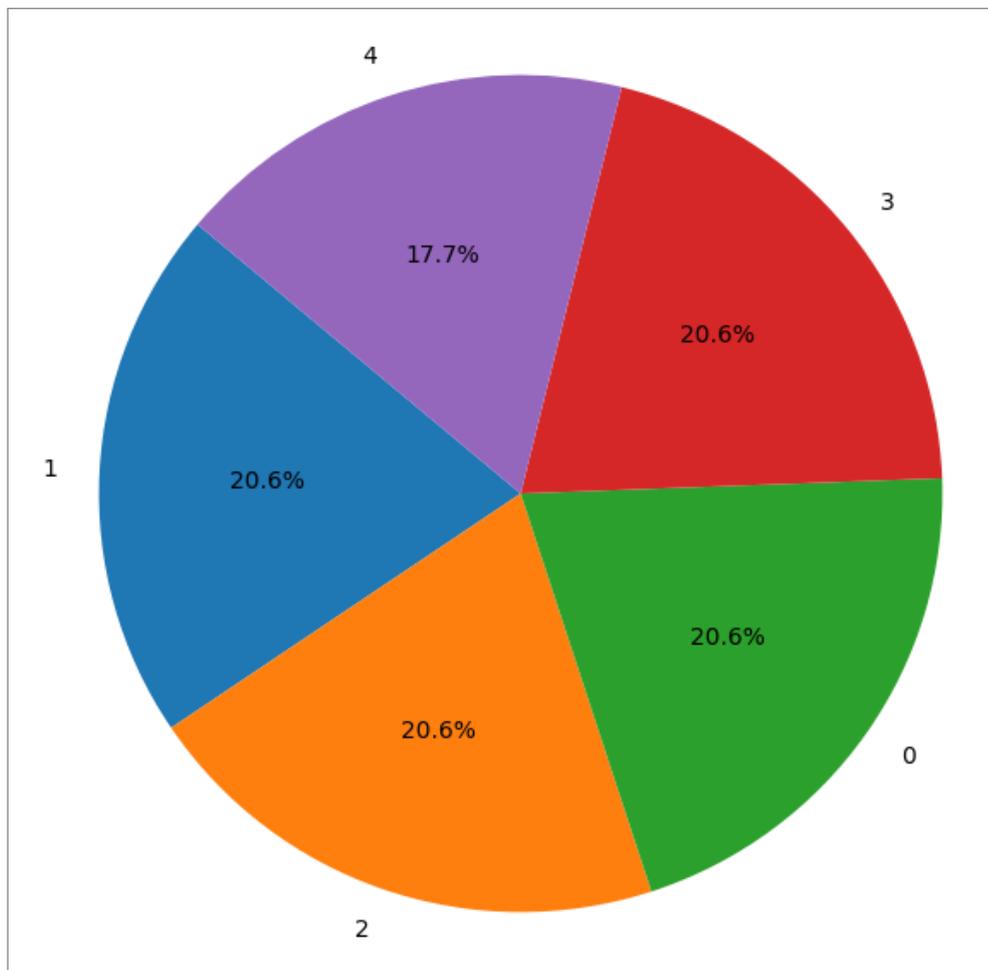
A figura 16 apresenta a distribuição das classes do conjunto de dados 02, percebe-se que esta base é balanceada, uma característica diferente do primeiro conjunto, esta diferença pode avaliar a capacidade de adaptação dos modelos de MCS à bases de dados com particularidades diferentes.

A fim de detalhar e entender melhor os dados, foram realizadas análises estatísticas para detecção de normalidade, grau de Curtose e Simetria. Para a análise de normalidade aplicou-se os testes de Shapiro-Wilk (SHAPIRO; WILK, 1965), Kolmogorov-Smirnov (KOLMOGOROV,

<sup>2</sup> <<https://www.pscad.com/knowledge-base/article/25>>

1933; SMIRNOV, 1948), Anderson-Darling (ANDERSON; DARLING, 1952) e Jarque-Bera (JARQUE; BERA, 1980). A tabela 1 traz os resultados da base 01 e a Tabela 2 traz os resultados da base 02. Em termos estatísticos, os dados semelhantes, os testes de normalidade indicam que eles não seguem uma distribuição normal, o  $p - valor = 0$  nos testes de Sphiro-Wilk, Kolmogorov-Smirnov e Jarque-Bera apontam para uma divergência da distribuição normal. O teste de Anderson-Darling também atestou a não normalidade das duas bases de dados, onde todos os níveis de significância testados (15%, 10%, 5%, 2,5% e 1%) ficaram muito abaixo da Estatística dos dados. O teste de Assimetria foi o único que apontou diferença entre as bases, a base 01 tende a ser assimétrica, com valor de  $Sk > 0,5$  o que indica uma cauda mais longa à direita da média, já a base 02 apontou um valor de  $Sk$  próximo de 0, o que indica um alto grau de simetria dos dados. Por último, foi calculado o Grau de Curtose dos dados, sendo, mais uma vez, semelhantes, os dois valores de  $K$  foram menores que 0, indicando que as bases possuem caudas mais leves e são mais achatadas que a normal.

Figura 16 – Distribuição de classes da Base de Dados 02



Fonte: Elaborada pelo autor (2024)

Tabela 1 – Testes Estatísticos para a base de dados 01

Teste Estatístico	Parâmetro	Valor
Shapiro-Wilk	Estatística	0,7709
	p-valor	0
Kolmogorov-Smirnov	Estatística	0,5
	p-valor	0
Anderson-Darling	Estatística	1919,3112
	Nível 15%	0,5760
	Nível 10%	0,6560
	Nível 5%	0,7870
	Nível 2,5%	0,9180
	Nível 1%	1,0920
Jarque-Bera	Estatística	2897,9840
	p-valor	0
Simetria	$Sk$	0,8948
Curtose	$K$	-0.5246

Fonte: Elaborada pelo autor (2025)

Tabela 2 – Testes Estatísticos para a base de dados 02

Teste Estatístico	Parâmetro	Valor
Shapiro-Wilk	Estatística	0,8917
	p-valor	0
Kolmogorov-Smirnov	Estatística	0,6357
	p-valor	0
Anderson-Darling	Estatística	2845,2001
	Nível 15%	0,5760
	Nível 10%	0,6560
	Nível 5%	0,7870
	Nível 2,5%	0,9180
	Nível 1%	1,0920
Jarque-Bera	Estatística	5497,9061
	p-valor	0
Simetria	$Sk$	0,0381
Curtose	$K$	-1,2665

Fonte: Elaborada pelo autor (2025)

Sistemas elétricos reais possuem vários tipos de situações que podem gerar ruídos nos valores medidos: interferência eletromagnética, efeito corona, transitórios eletromagnéticos

e distorções harmônicas são alguns exemplos de perturbações que podem ocorrer durante um curto-circuito, chaveamentos ou mesmo com o sistema funcionando em equilíbrio e que introduzem ruídos nas medições. Dito isto e com a intenção de aproximar os dados da realidade, foram introduzidos ruídos gaussianos durante as etapas de pré-processamento. O Ruído Gaussiano consegue produzir a variabilidade necessária aos dados para atestar a robustez dos modelos. Apesar de não representar idealmente os ruídos elétricos ocasionados pelos sistemas, trabalhos anteriores utilizaram esse tipo de ruído em estudos de classificação para sistemas elétricos com resultados satisfatórios: (BHUIYAN et al., 2022) e (K et al., 2022) aplicaram o Ruído Gaussiano para classificação de faltas em sistemas de transmissão, o primeiro utiliza modelos de *Deep Learning* e o segundo de ML para classificação, já (KARAN; YEH, 2020) e (FAHIM et al., 2020) focaram na classificação de faltas no âmbito das micro redes, que são sistemas elétricos pequenos compostos por geradores, linhas, transformadores e barramentos. Por último, (KURUKURU et al., 2020) utilizou Ruído Gaussiano em sua abordagem para classificação de faltas em sistemas fotovoltaicos. Desse modo, pode-se perceber a aplicabilidade e versatilidade que o Ruído Gaussiano possui, podendo ser aplicado em sistemas elétricos diversos sem grandes prejuízos de distanciamento dos ruídos reais gerados durante a operação dos sistemas.

Neste trabalho, o Ruído Gaussiano foi adicionado às duas bases de dados da seguinte maneira: para o primeiro banco de dados, os valores de Ruído Gaussiano estão na faixa de 10.000 a 60.000, valores estes que estão na mesma magnitude da base de dados para que não haja enviesamento ou sobreposição dos ruídos em relação aos dados. Para o segundo banco de dados, devido à sua normalização, os valores de Ruído Gaussiano variaram entre 0,1 e 1,5. A Tabela 3 apresenta as variações de amplitude (desvio padrão) e valor médio dos ruídos aplicados às bases de dados.

## 4.2 PROCEDIMENTO EXPERIMENTAL

Cada conjunto de dados foi dividido em duas amostras distintas: 80% para treinamento e 20% para teste. Essa divisão garante que os modelos tenham dados suficientes para aprender, preservando uma parte para avaliar o poder de generalização das abordagens em dados não vistos.

Tabela 3 – Níveis de ruído utilizados em cada uma das bases de dados

Base de dados	$\mu$ (Valor médio)	$\sigma$ (Desvio padrão)
01	0	10.000
		20.000
		30.000
		40.000
		50.000
		60.000
02	0	0,1
		0,3
		0,5
		0,7
		1,0
		1,3
		1,5

Fonte: Elaborada pelo autor (2025)

Os hiperparâmetros para cada modelo foram selecionados por meio de validação cruzada de pesquisa em grade com 5 *folds*. Essa técnica ajuda a mitigar o *overfitting*, garantindo que os modelos sejam validados em vários subconjuntos dos dados. Para cada conjunto de valores da pesquisa em grade, os dados de treinamento são divididos em cinco partes iguais e, durante cada iteração, quatro partes são usadas para treinamento, enquanto a parte restante é usada para validação. A combinação de valores de hiperparâmetros com a maior precisão média é selecionada. Os valores de hiperparâmetros para cada modelo são mostrados na Tabela 4 e foram selecionados com base em estudos anteriores que utilizaram estes modelos de classificação, com a intenção de inserir o máximo de diversidade possível no *grid-search*.

As abordagens de seleção estática e dinâmica foram avaliadas usando os dois conjuntos de dados descritos anteriormente. Todas as abordagens selecionadas são bem estabelecidas na literatura de aprendizagem de conjunto e alcançaram resultados notáveis em vários domínios de aplicação (CRUZ; SABOURIN; CAVALCANTI, 2018; LAZZARINI; TIANFIELD; CHARISSIS, 2023; ZHU et al., 2023; CORDEIRO; CAVALCANTI; CRUZ, 2023). Este estudo avaliou os seguintes métodos de seleção estática: Voto Majoritário, Generalização Empilhada com Árvore de Decisão (*Stacked DT*) e Generalização Empilhada com Regressão Logística (*Stacked LR*). Para seleção dinâmica, avaliamos as seguintes abordagens: DESP, KNORA-E, KNORA-U, MCB, META-DES e OLA. As Tabelas 5 e 6 mostram os hiperparâmetros usados para as técnicas de seleção estática e

dinâmica, respectivamente. Os hiperparâmetros empregados são os valores padrão do pacote DESlib<sup>3</sup> da linguagem Python.

Tabela 4 – Hiperparâmetros usados no *grid-search* para cada modelo

<b>Modelo</b>	<b>Parâmetro</b>	<b>Valor</b>
K-NN	K-vizinhos	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
	Medida	{'uniform', 'distance'}
	Métrica	{'euclidean', 'manhattan', 'minkowski'}
Árvore de Decisão	Controle de aleatoriedade	{0, 1, 2, 42}
	Critério	{'gini', 'entropy'}
	Profundidade máxima	{2, 3, 4, 5, 6, 7, 8, 9, 10, 11}
	Mínimo de amostras por folha	{1, 2, 4, 6}
<i>Random Forest</i>	Número de estimadores	{1, 10, 30, 100, 200}
	Controle de aleatoriedade	{0, 42}
	Profundidade máxima	{2, 10, 30, None}
	Número de características	{'auto', 'sqrt', 'log2'}
	Mínimo de amostras por folha	{1, 2}
<i>XGBoost</i>	Taxa de aprendizado	{0.1, 0.547, 0.6427}
	Profundidade máxima	{2, 4, 6, 8, 10}
	Número de estimadores	{2, 4, 8, 10, 200}
	Peso mínimo das instâncias	{1, 3, 5}
	Proporção de amostras	{0.7, 0.8, 0.9}
<i>LightGBM</i>	Taxa de aprendizado	{0.001, 0.01, 0.1}
	Profundidade máxima	{2, 4, 6, 8, 10}
	Peso mínimo das instâncias	{20}
	Número de estimadores	{2, 4, 8, 10, 200}
	Número de folhas	{7, 31}
	Tipo de <i>Booster</i>	{'gbdt', 'goss'}
<i>CatBoost</i>	Taxa de aprendizado	{0.1, 0.01, 0.001}
	Profundidade Máxima	{2, 4, 6, 8, 10}
	Número de estimadores	{2, 4, 8, 10, 200}

**Fonte:** Elaborada pelo autor (2024)

<sup>3</sup> <https://deslib.readthedocs.io/en/latest/>

As abordagens de seleção estática e dinâmica são comparadas com os seis modelos de classificação apresentados neste trabalho. Para garantir uma comparação justa, esses modelos de classificação constituem o conjunto classificadores que será usado pelas técnicas de MCS para classificar as falhas estudadas, posteriormente, os resultados serão aferidos para buscar o melhor modelo.

Tabela 5 – Hiperparâmetros utilizados para cada modelo de seleção estática

<b>Modelo de Seleção Estática</b>	<b>Hiperparâmetro</b>	<b>Valor</b>
Voto Majoritário	votação	{'hard'}
<i>Stacked DT</i>	Meta-classificador	{DecisionTree}
	Critério do meta-classificador	{'gini'}
	Mínimo de amostras por folha do meta-classificador	{1}
	Divisão mínima de amostras por meta-classificador	{2}
	Divisor do meta-classificador	{'best'}
<i>Stacked LR</i>	Meta-classificador	{LogisticRegression}

**Fonte:** Elaborada pelo autor (2024)

A avaliação dos modelos foi conduzida usando as principais métricas de análise: acurácia, precisão, e *recall*. A acurácia avalia a correção geral do modelo, fornecendo uma medida de quão bem o modelo se desempenha em todas as classes. A precisão e o *recall*, por outro lado, avaliam o desempenho do modelo especificamente na classe positiva. A precisão se concentra na correção de previsões positivas, indicando a proporção de resultados positivos verdadeiros entre todas as previsões positivas. O *recall* avalia a integridade das previsões positivas, representando a proporção de resultados positivos verdadeiros dos casos positivos reais. A Tabela 7 apresenta as equações, intervalos e nomes de alias para cada métrica. Verdadeiro Positivo (VP) refere-se a instâncias em que o modelo prevê corretamente a classe positiva. Verdadeiro Negativo (VN) refere-se a instâncias em que o modelo prevê corretamente a classe negativa. Falso Positivo (FP) ocorre quando o modelo prevê incorretamente a classe positiva para uma instância negativa. Falso Negativo (FN) ocorre quando o modelo prevê incorretamente a classe negativa para uma instância positiva.

Tabela 6 – Hiperparâmetros utilizados para cada modelo de seleção dinâmica

<b>Modelo de Seleção Dinâmica</b>	<b>Hiperparâmetro</b>	<b>Valor</b>
DESP	DFP	{False}
	Proporção de amostras da base de seleção dinâmica	{0.5}
	Proporção de amostras consideradas incompetentes	{0.3}
	K-vizinhos	{7}
	Classificador K-NN	{'knn'}
	Métrica do K-NN	{'minkowski'}
	Modo	{'selection'}
KNORA-E KNORA-U	DFP	{False}
	Proporção de amostras da base de seleção dinâmica	{0.5}
	Proporção de amostras consideradas incompetentes	{0.3}
	K-vizinhos	{7}
	Classificador K-NN	{'knn'}
	Métrica do K-NN	{'minkowski'}
MCB	DFP	{False}
	Proporção de amostras da base de seleção dinâmica	{0.5}
	Proporção de amostras consideradas incompetentes	{0.3}
	Limiar de diferença	{0.1}
	K-vizinhos	{7}
	Classificador K-NN	{'knn'}

<b>Modelo de Seleção Dinâmica</b>	<b>Hiperparâmetro</b>	<b>Valor</b>
	Métrica do K-NN	{'minkowski'}
META-DES	DFP	{False}
	Proporção de amostras da base de seleção dinâmica	{0.5}
	Heurística de complexidade	{1.0}
	Proporção de amostras consideradas incompetentes	{0.3}
	Parâmetro Kappa	{5}
	K-vizinhos	{7}
	Classificador K-NN	{'knn'}
	Métrica do K-NN	{'minkowski'}
	Meta-classificador	{None}
	Modo	{'selection'}
OLA	DFP	{False}
	Proporção de amostras da base de seleção dinâmica	{0.5}
	K-vizinhos	{7}
	Classificador K-NN	{'knn'}
	Métrica do K-NN	{'minkowski'}
	Estimação de erro por K-NN	{False}

**Fonte:** Elaborada pelo autor (2024)

Tabela 7 – Métricas para avaliação de classificação. Para todas as métricas, quanto maior o valor, melhor o desempenho da classificação.

<b>Métrica</b>	<b>Equação</b>	<b>Limites</b>
Acurácia	$\frac{VP+VN}{VP+VN+FP+FN}$	[0, 1]
Precisão	$\frac{VP}{VP+FP}$	[0, 1]
<i>Recall</i>	$\frac{VP}{VP+FN}$	[0, 1]

**Fonte:** Elaborada pelo autor (2024)

## 5 RESULTADOS

Neste capítulo serão analisados os resultados experimentais, abrangendo os classificadores e as técnicas de seleção estática e dinâmica, que foram apresentados anteriormente, para os dois conjuntos de dados estudados. As abordagens são avaliadas usando as métricas de Acurácia, Precisão e *Recall* em sete e oito cenários com variação de ruído para as Bases de Dados 1 e 2, respectivamente.

### 5.1 BASE DE DADOS 1

A tabela 8 apresenta as métricas de Acurácia, Precisão e *Recall* usadas para analisar o desempenho dos modelos individuais. Os resultados, em geral, diminuíram com o aumento do nível de ruído. *Random Forest* e K-NN obtiveram os melhores desempenhos para o cenário sem ruído, enquanto a Árvore de Decisão obteve os menores valores. No entanto, o K-NN foi o modelo mais afetado pelo aumento do nível de ruído. De fato, o K-NN obteve o segundo pior resultado do nível de ruído entre 10.000 e 60.000. Este resultado mostra a sensibilidade do K-NN a dados ruidosos. A Árvore de Decisão obteve os piores valores em todos os cenários. Este resultado ruim pode ser causado por *overfitting* na amostra de treinamento ou desequilíbrio de classe.

*LightGBM*, *XGBoost* e *CatBoost*, com a primeira, segunda e terceira posições, respectivamente, apresentaram resultados mais estáveis com adição de ruído. Três modelos atingiram métricas maiores que 99% para todos os cenários. Este resultado mostra que esses classificadores conseguiram aprender a distribuição das diferentes classes para este conjunto de dados. Este comportamento pode ter ocorrido porque, apesar da adição de ruído, os dados não apresentam alta complexidade.

A Figura 17 mostra a evolução da acurácia para os diferentes níveis de ruído. É possível notar a degradação do desempenho de todos os modelos, principalmente K-NN, Árvore de Decisão e *Random Forest*, respectivamente.

A tabela 9, apresentada abaixo, dispõe as métricas de Acurácia, Precisão e *Recall* para as técnicas de Seleção Estática (Voto majoritário, *Stacked DT* e *Stacked LR*). Para todos os modelos, os valores das métricas tendem a diminuir com o aumento do nível de ruído. Ambas as técnicas de empilhamento de modelos obtiveram um resultado mais estável, variando em

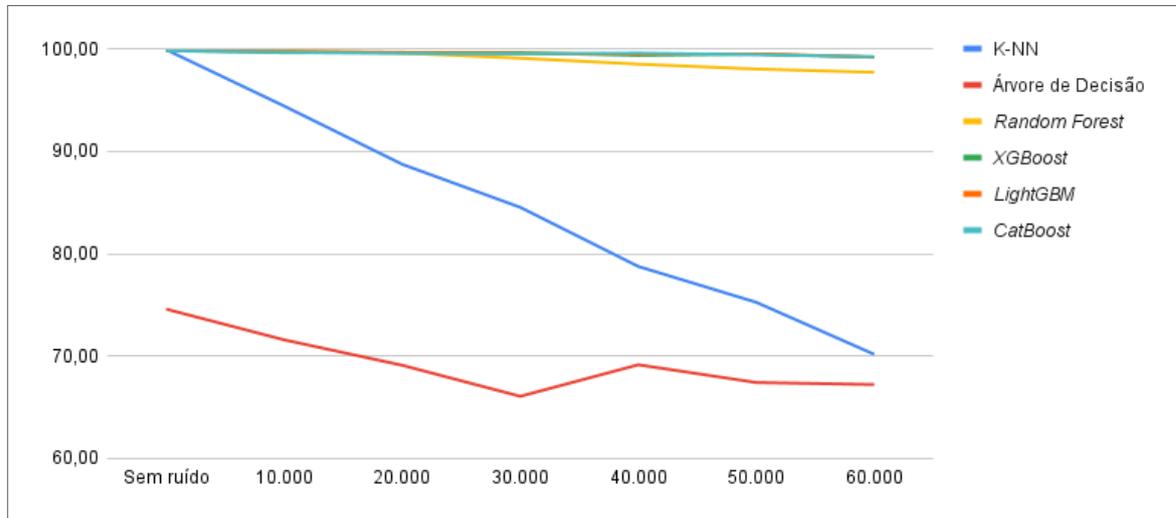
torno de 99% para todas as métricas. Esse resultado mostra que a estratégia de atribuição de pesos foi eficaz. De fato, os modelos individuais *LightGBM*, *XGBoost* e *CatBoost* alcançaram os melhores resultados e, portanto, receberam os maiores pesos nos conjuntos empilhados. Por outro lado, o modelo de voto majoritário sofreu com a adição de ruído, pois as métricas diminuíram de 99,90%, 99,90% e 99,90% para 97,92%, 98,00% e 97,92%, respectivamente. Este resultado mostra que os modelos *Árvore de Decisão*, *K-NN* e *Random Forest* devem ter influenciado a decisão do Voto Majoritário. Este resultado mostra a importância da qualidade do *pool* de modelos (criação e treinamento).

Tabela 8 – Avaliação dos classificadores na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

Nível de Ruído	Métrica	Classificadores					
		K-NN	Árvore de Decisão	<i>Random Forest</i>	<i>XGBoost</i>	<i>LightGBM</i>	<i>CatBoost</i>
Sem ruído	Acurácia	<b>99,90</b>	74,58	<b>99,90</b>	99,83	99,85	99,80
	Precisão	<b>99,90</b>	77,24	<b>99,90</b>	99,83	99,85	99,80
	<i>Recall</i>	<b>99,90</b>	74,58	<b>99,90</b>	99,83	99,85	99,80
10.000	Acurácia	94,40	71,58	<b>99,80</b>	99,62	99,75	99,67
	Precisão	94,36	74,52	<b>99,80</b>	99,63	99,75	99,67
	<i>Recall</i>	94,40	71,58	<b>99,80</b>	99,62	99,75	99,67
20.000	Acurácia	88,72	69,10	99,58	99,62	<b>99,65</b>	99,55
	Precisão	88,45	72,02	99,58	99,62	<b>99,65</b>	99,55
	<i>Recall</i>	88,72	69,10	99,58	99,62	<b>99,65</b>	99,55
30.000	Acurácia	84,52	66,07	99,08	<b>99,62</b>	99,55	99,50
	Precisão	83,94	68,64	99,08	<b>99,63</b>	99,55	99,50
	<i>Recall</i>	84,52	66,07	99,08	<b>99,62</b>	99,55	99,50
40.000	Acurácia	78,75	69,15	98,50	99,35	99,48	<b>99,58</b>
	Precisão	77,58	69,63	98,50	99,35	99,48	<b>99,58</b>
	<i>Recall</i>	78,75	69,15	98,50	99,35	99,48	<b>99,58</b>
50.000	Acurácia	75,25	67,42	98,02	99,48	<b>99,50</b>	99,42
	Precisão	73,40	67,60	98,05	99,48	<b>99,50</b>	99,42
	<i>Recall</i>	75,25	67,42	98,02	99,48	<b>99,50</b>	99,42
60.000	Acurácia	70,17	67,22	97,70	99,17	99,20	<b>99,22</b>
	Precisão	67,34	60,77	97,74	99,18	99,20	<b>99,22</b>
	<i>Recall</i>	70,17	67,22	97,70	99,17	99,20	<b>99,22</b>

Fonte: Elaborada pelo autor (2024)

Figura 17 – Acurácia obtida para cada classificador individualmente com o acréscimo do ruído na Base de Dados 1



Fonte: Elaborada pelo autor (2024)

A Figura 18 mostra a evolução da acurácia para os diferentes níveis de ruído. A degradação do desempenho de todos os modelos, principalmente do Voto Majoritário, pode ser notada.

A tabela 10, apresentada abaixo, traz o desempenho das técnicas de Seleção Dinâmica (DESP, KNORA-E, KNORA-U, MCB, META-DES e OLA) para a Base de Dados 1. Para todas as técnicas, os valores das métricas tendem a diminuir com o aumento do nível de ruído. KNORA-E, KNORA-U e META-DES atingiram um desempenho mais estável, variando todas as métricas em torno de 99%. Este resultado mostra que a criação da região de competência e a estratégia de seleção dinâmica desses modelos são mais eficazes do que DESP, MCB e OLA. De fato, o modelo OLA sofreu com o aumento do ruído, pois as métricas acurácia, precisão e *recall* diminuíram de 99,90%, 99,90% e 99,90% para 94,97%, 95,11% e 94,97%, respectivamente. Esses resultados mostram que métodos baseados na RoC, que dependem de instâncias de treinamento semelhantes à instância de teste, como OLA, apresentam maior sensibilidade ao ruído. Em contraste, abordagens baseadas em desempenho, como as empregadas por KNORA-E e KNORA-U, ou o uso de meta-recursos como a abordagem META-DES, demonstram maior robustez.

A Figura 19 mostra a evolução da acurácia para os diferentes níveis de ruído. A degradação do desempenho de todos os modelos, principalmente do DESP, MCB e OLA, pode ser notada.

A Figura 20 compara os valores das métricas alcançados pelos melhores modelos, medidos pela média aritmética dos resultados, das abordagens avaliadas: Monolítica e MCS (Seleção Estática e Seleção Dinâmica). Ela mostra que modelos baseados em *ensembles* atingiram

os melhores resultados em todos os cenários. Do cenário sem ruído até com amplitude de 50.000, *Stacked LR* atingiu os melhores valores de acurácia, precisão e *recall*; enquanto isso, para 60.000, META-DES foi superior para todas as métricas. É essencial destacar que as abordagens de *ensemble* atingiram valores de desempenho mais altos do que os modelos unitários na maioria dos cenários. Por exemplo, *Stacked LR* e META-DES atingiram um desempenho superior de cerca de 48% e 41% em termos de Acurácia com relação à Árvore de Decisão e K-NN, respectivamente.

Tabela 9 – Avaliação das técnicas de Seleção Estática na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

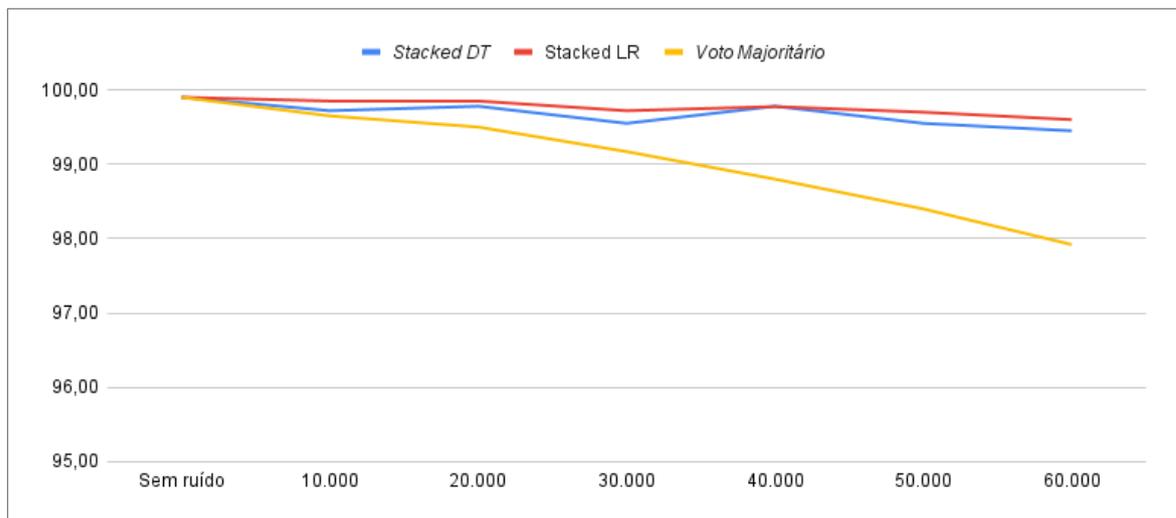
Nível de Ruído	Métrica	Modelos de Seleção Estática		
		Voto majoritário	<i>Stacked DT</i>	<i>Stacked LR</i>
Sem ruído	Acurácia	<b>99,90</b>	<b>99,90</b>	<b>99,90</b>
	Precisão	<b>99,90</b>	<b>99,90</b>	<b>99,90</b>
	<i>Recall</i>	<b>99,90</b>	<b>99,90</b>	<b>99,90</b>
10.000	Acurácia	99,65	99,72	<b>99,85</b>
	Precisão	99,65	99,73	<b>99,85</b>
	<i>Recall</i>	99,65	99,72	<b>99,85</b>
20.000	Acurácia	99,50	99,78	<b>99,85</b>
	Precisão	99,50	99,78	<b>99,85</b>
	<i>Recall</i>	99,50	99,78	<b>99,85</b>
30.000	Acurácia	99,17	99,55	<b>99,72</b>
	Precisão	99,18	99,55	<b>99,73</b>
	<i>Recall</i>	99,17	99,55	<b>99,78</b>
40.000	Acurácia	98,80	<b>99,78</b>	<b>99,78</b>
	Precisão	98,82	<b>99,78</b>	<b>99,78</b>
	<i>Recall</i>	98,80	<b>99,78</b>	<b>99,78</b>
50.000	Acurácia	98,40	99,55	<b>99,70</b>
	Precisão	98,44	99,55	<b>99,70</b>
	<i>Recall</i>	98,40	99,55	<b>99,70</b>
60.000	Acurácia	97,92	99,45	<b>99,60</b>
	Precisão	98,00	99,45	<b>99,60</b>
	<i>Recall</i>	97,92	99,45	<b>99,60</b>

Fonte: Elaborada pelo autor (2024)

Os resultados para a Base de Dados 1 demonstram que as abordagens de MCS superaram os classificadores individuais na maioria dos cenários, particularmente na presença de ruído. Entre os classificadores, métodos de aumento de gradiente como *LightGBM*, *XGBo-*

*ost* e *CatBoost* mostraram um desempenho mais estável, mantendo altos valores métricos mesmo quando o nível de ruído aumentou, enquanto K-NN e Árvore de Decisão foram os mais afetados pelo ruído. Modelos de conjunto estático, especialmente *Stacked LR*, provaram ser eficazes na manutenção de desempenho robusto em todos os níveis de ruído, beneficiando-se da combinação ponderada de modelos de alto desempenho. Nas abordagens de seleção dinâmica, KNORA-E, KNORA-U e META-DES demonstraram resiliência ao ruído, mantendo desempenho quase consistente. A degradação do desempenho de métodos como OLA, MCB e DESP destaca sua sensibilidade ao ruído. Assim, as abordagens de MCS avaliadas, particularmente *Stacked LR* e META-DES, forneceram os melhores resultados, validando a eficácia da combinação e seleção de modelos para aumentar a robustez e a precisão em ambientes ruidosos.

Figura 18 – Acurácia obtida para cada técnica de Seleção Estática com o acréscimo do ruído na Base de Dados 1



Fonte: Elaborada pelo autor (2024)

O acréscimo de ruído ocasionou uma diminuição de 48,35% nas métricas do classificador K-NN, já os melhores classificadores, tipo *Gradient Boosting*, sofreram uma degradação de 0,66%. Ao analisar os MCS, percebe-se uma variação de apenas 0,3% no modelo *Stacked LR* e de 0,51 % para o KNORA-U, apesar do ganho parecer baixo, a degradação dos melhores classificadores duas vezes maior do que a do melhor MCS, em um ambiente de milhares de dados, o aumento na quantidade de classificações corretas pode ser significativo.

Tabela 10 – Avaliação das técnicas de Seleção Dinâmica na Base de Dados 1 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

Nível de Ruído	Métrica	Modelos de Seleção Dinâmica					
		KNORA-U	KNORA-E	DESP	META-DES	OLA	MCB
Sem ruído	Acurácia	<b>99,93</b>	99,85	99,85	99,85	99,90	99,88
	Precisão	<b>99,93</b>	99,85	99,85	99,85	99,90	99,88
	<i>Recall</i>	<b>99,93</b>	99,85	99,85	99,85	99,90	99,88
10.000	Acurácia	<b>99,80</b>	99,70	99,65	99,75	98,72	99,02
	Precisão	<b>99,80</b>	99,70	99,65	99,75	98,72	99,03
	<i>Recall</i>	<b>99,80</b>	99,70	99,65	99,75	98,72	99,02
20.000	Acurácia	99,70	99,67	99,52	<b>99,75</b>	97,20	98,53
	Precisão	99,70	99,68	99,53	<b>99,75</b>	97,26	98,52
	<i>Recall</i>	99,70	99,67	99,52	<b>99,75</b>	97,26	98,53
30.000	Acurácia	99,50	99,48	99,12	<b>99,70</b>	96,83	98,32
	Precisão	99,50	99,48	99,14	<b>99,70</b>	96,90	98,32
	<i>Recall</i>	99,50	99,48	99,12	<b>99,70</b>	96,83	98,32
40.000	Acurácia	99,58	99,48	98,67	<b>99,70</b>	95,55	97,15
	Precisão	99,58	99,48	98,70	<b>99,70</b>	95,61	97,15
	<i>Recall</i>	99,58	99,48	98,67	<b>99,70</b>	95,55	97,15
50.000	Acurácia	99,48	99,40	98,78	<b>99,65</b>	95,60	97,82
	Precisão	99,48	99,40	98,79	<b>99,65</b>	95,64	97,82
	<i>Recall</i>	99,48	99,40	98,78	<b>99,65</b>	95,60	97,82
60.000	Acurácia	99,42	99,33	98,60	<b>99,62</b>	94,97	96,95
	Precisão	99,43	99,33	98,63	<b>99,63</b>	95,11	96,96
	<i>Recall</i>	99,42	99,33	98,60	<b>99,62</b>	94,97	96,95

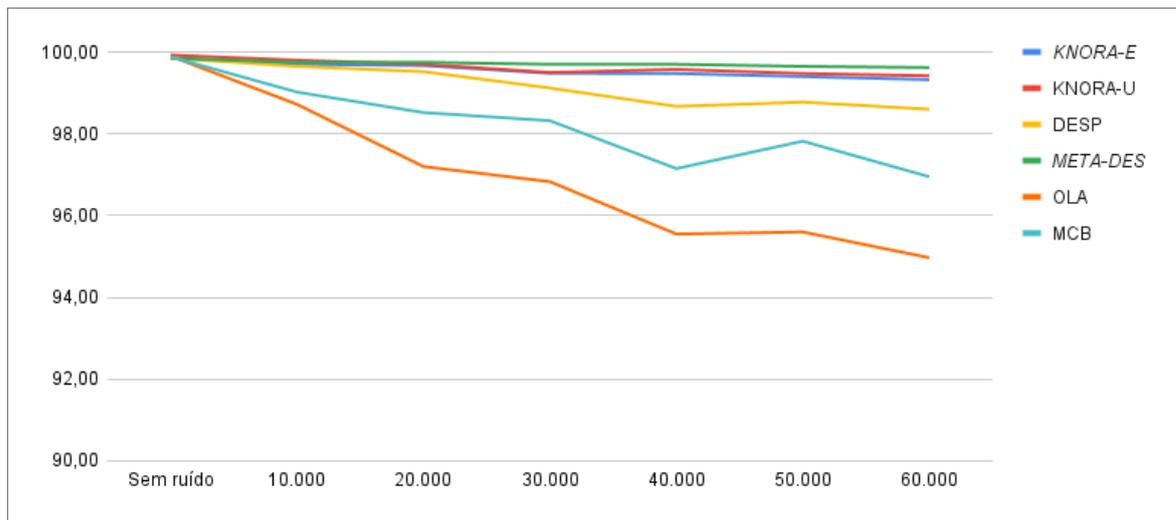
Fonte: Elaborada pelo autor (2024)

## 5.2 BASE DE DADOS 2

A tabela 11, representada abaixo, dispõe as métricas de desempenho Acurácia, Precisão e *Recall* alcançadas pelos classificadores individualmente para a Base de Dados 2. Os valores das métricas para todos os modelos avaliados têm uma tendência de queda com o aumento do nível de ruído. Os modelos *XGBoost* e *LightGBM* atingiram os melhores valores métricos em todos os cenários. O *LightGBM* alcançou a primeira posição nos dois primeiros cenários, enquanto o *XGBoost* foi o melhor nos outros. Ambos atingiram o mesmo desempenho para um nível de ruído de 0,3 (terceiro cenário). K-NN e Árvore de Decisão obtiveram os dois piores valores métricos para todos os cenários, respectivamente. No entanto, entre eles, o desempenho do

K-NN foi mais afetado pelo aumento do nível de ruído. Por exemplo, a Acurácia dos modelos K-NN e Árvore de Decisão caiu 11,90% e 16,78%, respectivamente. Este resultado mostra a sensibilidade do K-NN a dados ruidosos e um provável *overfitting* da Árvore de Decisão na amostra de treinamento.

Figura 19 – Acurácia obtida para cada técnica de Seleção Dinâmica com o acréscimo do ruído na Base de Dados 1



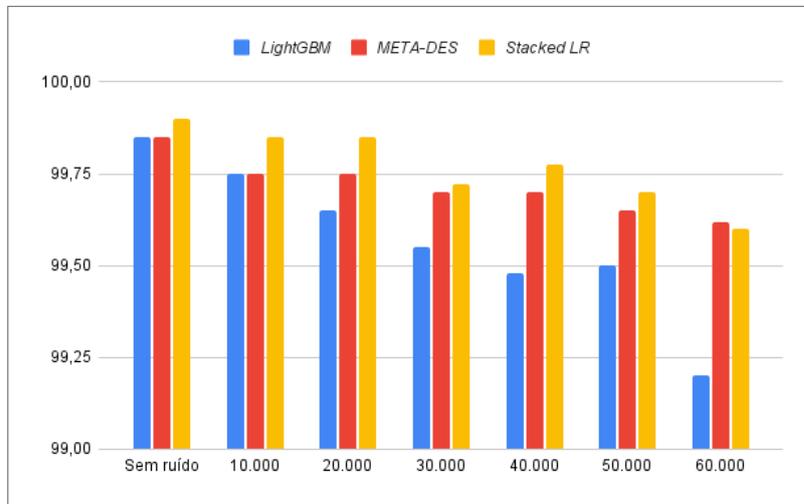
Fonte: Elaborada pelo autor (2024)

*LightGBM*, *XGBoost* e *CatBoost*, com a primeira, segunda e terceira posições, respectivamente, apresentaram resultados mais estáveis com adição de ruído. Três modelos atingiram métricas maiores que 99% para todos os cenários. Este resultado mostra que esses modelos individuais conseguiram aprender a distribuição das diferentes classes para este conjunto de dados. Este comportamento pode ter ocorrido porque, apesar da adição de ruído, os dados não apresentam alta complexidade.

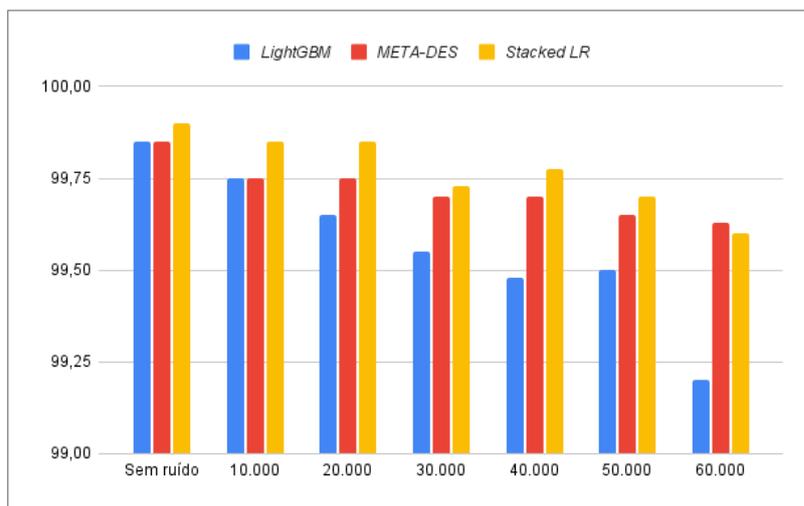
A Figura 21 mostra a degradação de todos os modelos individuais em termos da acurácia com o aumento do nível de ruído. É possível ver que K-NN e Árvore de Decisão obtiveram os piores valores. No entanto, *CatBoost*, *LightGBM*, *Random Forest* e *XGBoost* foram os modelos que foram mais impactados pelo aumento do nível de ruído. De fato, a acurácia desses modelos individuais piorou em cerca de 20% do primeiro cenário (sem ruído) para o último cenário (ruído = 1,5).

Figura 20 – Acurácia, Precisão e *Recall* obtidas para os melhores modelos das abordagens avaliadas na Base de Dados 1

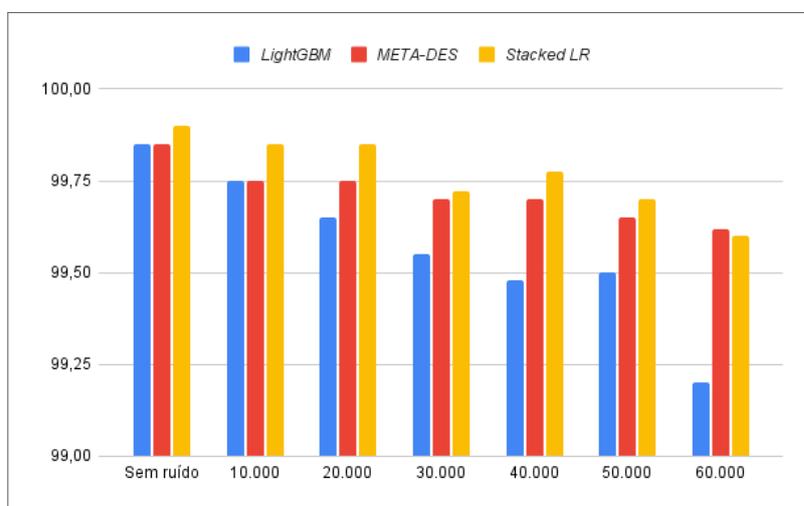
(a) Acurácia



(b) Precisão



(c) *Recall*



Fonte: Elaborada pelo autor (2024)

Tabela 11 – Avaliação dos classificadores na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

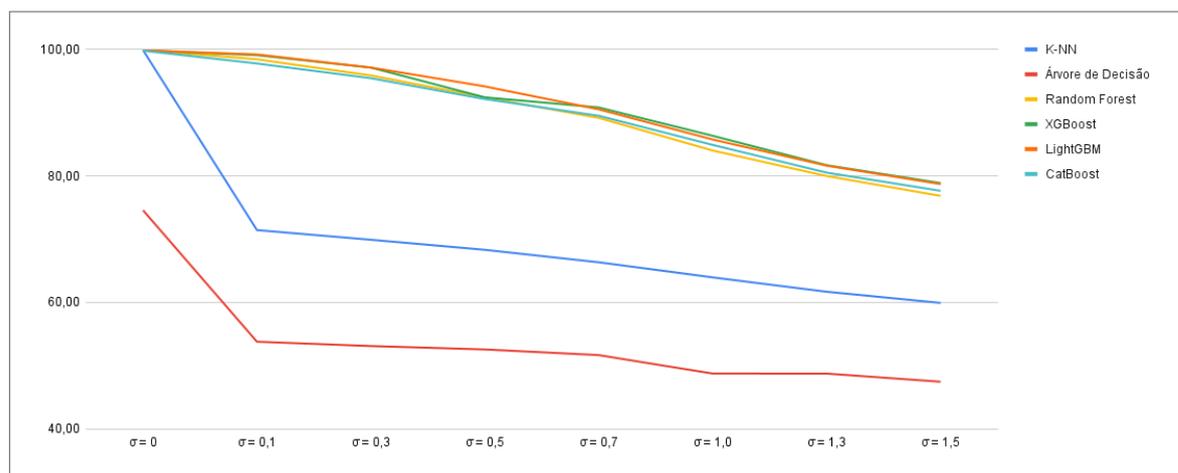
Nível de Ruído	Métrica	Classificadores					
		K-NN	Árvore de Decisão	<i>Random Forest</i>	<i>XGBoost</i>	<i>LightGBM</i>	<i>CatBoost</i>
Sem ruído	Acurácia	72,00	53,87	99,77	99,79	<b>99,83</b>	99,40
	Precisão	71,95	43,66	99,77	99,79	<b>99,83</b>	99,40
	<i>Recall</i>	72,00	53,87	99,77	99,79	<b>99,83</b>	99,40
0,1	Acurácia	71,44	53,79	98,45	99,10	<b>99,21</b>	97,79
	Precisão	71,44	43,58	98,45	99,10	<b>99,21</b>	97,79
	<i>Recall</i>	71,44	53,79	98,45	99,10	<b>99,21</b>	97,79
0,3	Acurácia	69,91	53,10	95,89	<b>97,14</b>	<b>97,14</b>	95,45
	Precisão	69,84	42,84	95,89	<b>97,14</b>	<b>97,14</b>	95,45
	<i>Recall</i>	69,91	53,10	95,89	<b>97,14</b>	<b>97,14</b>	95,45
0,5	Acurácia	68,32	52,56	92,42	92,43	<b>94,16</b>	92,16
	Precisão	68,22	48,32	92,43	92,43	<b>94,16</b>	92,16
	<i>Recall</i>	68,32	52,56	92,42	92,43	<b>94,16</b>	92,16
0,7	Acurácia	66,35	51,67	89,19	<b>90,83</b>	90,53	89,52
	Precisão	66,40	48,01	89,22	<b>90,84</b>	90,54	89,52
	<i>Recall</i>	66,35	51,67	89,19	<b>90,83</b>	90,53	89,52
1,0	Acurácia	63,96	48,75	84,02	<b>86,35</b>	85,76	84,91
	Precisão	63,83	49,19	84,12	<b>86,39</b>	85,81	84,91
	<i>Recall</i>	63,96	48,75	84,02	<b>86,35</b>	85,76	84,91
1,3	Acurácia	61,69	48,74	79,99	<b>81,70</b>	81,63	80,53
	Precisão	61,53	48,98	80,06	<b>81,73</b>	81,68	80,53
	<i>Recall</i>	61,69	48,74	79,99	<b>81,70</b>	81,63	80,53
1,5	Acurácia	59,92	47,46	76,85	<b>78,87</b>	78,71	77,65
	Precisão	59,75	46,72	76,92	<b>78,91</b>	78,82	77,65
	<i>Recall</i>	59,92	47,46	76,85	<b>78,87</b>	78,71	77,65

Fonte: Elaborada pelo autor (2024)

A tabela 12, apresentada abaixo, dispõe as métricas de acurácia, precisão e *recall* para as técnicas de Seleção Estática (Voto majoritário, *Stacked DT* e *Stacked LR*). Para todos os modelos, os valores das métricas tendem a diminuir com o aumento do nível de ruído. *Stacked LR* e Voto Majoritário obtiveram um resultado mais estável, variando entre 99% e 78% para todas as métricas. Esse resultado mostra que a estratégia de atribuição de pesos utilizando um modelo de árvore, como o *Stacked DT*, não foi eficaz, já a atribuição de pesos baseada em regressão logística foi menos sensível ao ruído. Esse comportamento provavelmente

ocorreu devido à natureza da Árvore de Decisão, que é propensa a *overfitting*, especialmente quando confrontado com dados ruidosos. À medida que o ruído aumenta, as árvores dentro do conjunto podem capturar flutuações aleatórias, levando a uma generalização mais pobre e degradação significativa do desempenho. Por outro lado, o modelo de voto majoritário foi a melhor técnica de seleção estática, mostrando que Árvore de Decisão e K-NN, que erraram a maioria das previsões, não prejudicaram a escolha da classe mais frequente entre os classificadores o Voto Majoritário, mesmo não atribuindo pesos individuais para cada classificador, selecionou a escolha dos quatro melhores modelos individuais (*CatBoost*, *LightGBM*, *Random Forest* e *XGBoost*). Este resultado mostra a importância da qualidade do *pool* de modelos (criação e treinamento).

Figura 21 – Acurácia obtida para cada classificador individualmente com o acréscimo do ruído na Base de Dados 2



Fonte: Elaborada pelo autor (2024)

A Figura 22 mostra a evolução da acurácia para os diferentes níveis de ruído. A degradação do desempenho de todos os modelos pode ser notada, sobretudo do *Stacked DT*.

A tabela 13, apresentada abaixo, traz o desempenho das técnicas de Seleção Dinâmica (DESP, KNORA-E, KNORA-U, MCB, e OLA) para a Base de Dados 2. Para todas as técnicas, os valores das métricas tendem a diminuir com o aumento do nível de ruído. KNORA-U, DESP e META-DES atingiram um desempenho mais estável, apesar de variando todas as métricas entre 99% e 77%. O modelo DESP mostrou desempenho competitivo a partir do nível de ruído 0,5, alcançando um *recall* de 84,77%, em comparação com 84,19% para KNORA-U e 83,99% para META-DES. Em contraste, OLA e MCB foram mais sensíveis ao ruído, com OLA mostrando a queda de desempenho mais significativa devido à sua dependência de instâncias de treinamento locais, que se tornam menos confiáveis à medida que o ruído aumenta.

Tabela 12 – Avaliação das técnicas de Seleção Estática na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

Nível de Ruído	Métrica	Modelos de Seleção Estática		
		Voto majoritário	<i>Stacked DT</i>	<i>Stacked LR</i>
Sem ruído	Acurácia	99,66	99,77	<b>99,80</b>
	Precisão	99,66	99,77	<b>99,80</b>
	<i>Recall</i>	99,66	99,77	<b>99,80</b>
0,1	Acurácia	98,92	98,75	<b>99,23</b>
	Precisão	98,92	98,74	<b>99,23</b>
	<i>Recall</i>	98,92	98,75	<b>99,23</b>
0,3	Acurácia	96,65	95,62	<b>97,05</b>
	Precisão	96,66	95,62	<b>97,05</b>
	<i>Recall</i>	96,65	95,62	<b>97,05</b>
0,5	Acurácia	93,64	91,21	<b>93,84</b>
	Precisão	93,67	91,21	<b>93,84</b>
	<i>Recall</i>	93,64	91,21	<b>93,84</b>
0,7	Acurácia	90,45	86,52	<b>90,76</b>
	Precisão	90,45	86,52	<b>90,77</b>
	<i>Recall</i>	90,45	86,52	<b>90,76</b>
1,0	Acurácia	<b>85,80</b>	80,25	85,61
	Precisão	<b>85,95</b>	80,10	85,45
	<i>Recall</i>	<b>85,80</b>	80,25	85,61
1,3	Acurácia	<b>81,27</b>	74,15	<b>81,27</b>
	Precisão	81,41	74,35	<b>81,47</b>
	<i>Recall</i>	<b>81,27</b>	74,15	<b>81,27</b>
1,5	Acurácia	<b>78,58</b>	70,55	78,34
	Precisão	<b>78,74</b>	71,05	78,56
	<i>Recall</i>	<b>78,58</b>	70,55	78,34

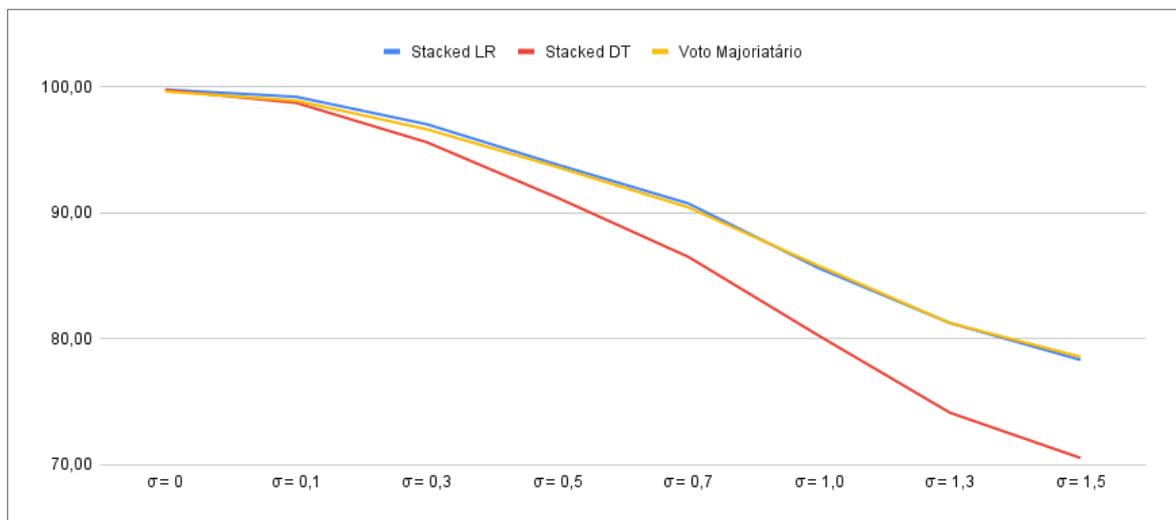
Fonte: Elaborada pelo autor (2024)

A Figura 23 mostra a evolução da acurácia para os diferentes níveis de ruído. A degradação do desempenho de todos os modelos, principalmente do KNORA-E, MCB e OLA, pode ser notada.

A Figura 24 compara os valores das métricas alcançados pelos melhores modelos, medidos pela média aritmética dos resultados, das abordagens avaliadas: Monolíticas MCS (estática e dinâmica). Ela mostra que, inicialmente, sem ruído ( $\sigma = 0$ ), todos os modelos tiveram desempenho semelhante, com métricas acima de 95%. À medida que os níveis de ruído aumentam, o *Stacked LR* exibe um declínio mais acentuado em comparação com o *XGBoost* e

o META-DES, que mantém melhor resiliência. De  $\sigma = 1,0$  em diante, a degradação se torna mais pronunciada para todos os modelos, este comportamento, no entanto, não possui relação direta com os sistemas de potência, sendo uma limitação dos classificadores para níveis de ruído mais alto. META-DES e *XGBoost*, ainda assim, conseguiram superar consistentemente o *Stacked LR*, particularmente em termos de precisão e *recall*, demonstrando maior robustez ao ruído.

Figura 22 – Acurácia obtida para cada técnica de Seleção Estática com o acréscimo do ruído na Base de Dados 2



Fonte: Elaborada pelo autor (2024)

Os resultados para o conjunto de dados 2 indicam que as abordagens de conjunto e seleção dinâmica superaram consistentemente os classificadores individuais à medida que os níveis de ruído aumentaram. Entre os modelos individuais, *XGBoost* e *LightGBM* demonstraram maior resiliência, mantendo forte desempenho em vários níveis de ruído, enquanto K-NN e Árvore de Decisão foram os mais afetados pelo ruído, com K-NN mostrando maior sensibilidade. Modelos de conjunto estático, particularmente *Stacked LR* e Voto Majoritário, exibiram desempenho robusto, com *Stacked LR* se destacando em condições sem ruído e Voto Majoritário mostrando desempenho competitivo conforme o ruído aumentou. Nas abordagens de seleção dinâmica, KNORA-E, KNORA-U e META-DES demonstraram estabilidade e resiliência, enquanto OLA e MCB experimentaram degradação significativa de desempenho em condições ruidosas. No geral, META-DES e *XGBoost* foram os modelos mais robustos, fornecendo consistentemente resultados superiores, validando a eficácia dos métodos de conjunto e seleção dinâmica em melhorar o desempenho e a robustez do modelo na presença de ruído.

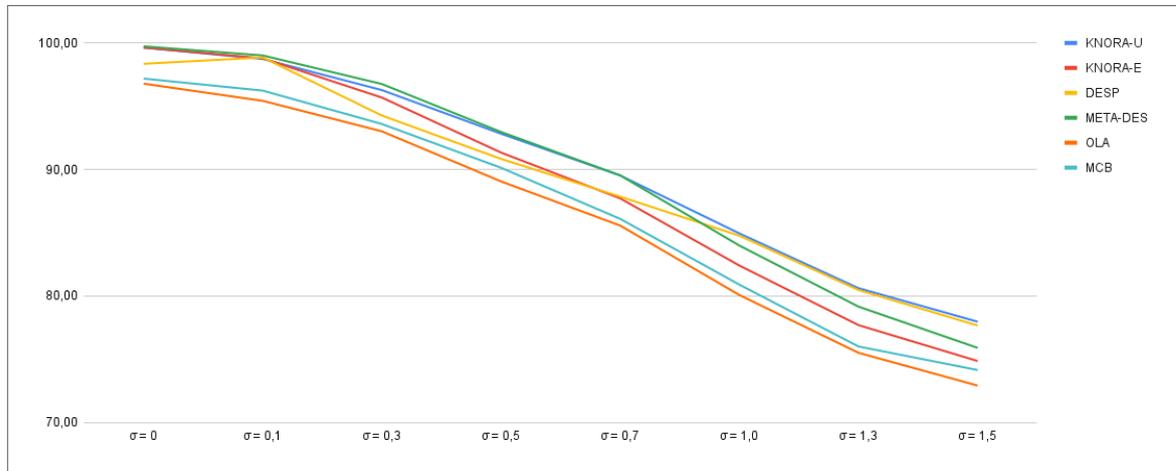
Tabela 13 – Avaliação das técnicas de Seleção Dinâmica na Base de Dados 2 para vários níveis de ruído. O melhor resultado para cada métrica é destacado em negrito.

Nível de Ruído	Métrica	Modelos de Seleção Dinâmica					
		KNORA-U	KNORA-E	DESP	META-DES	OLA	MCB
Sem ruído	Acurácia	99,64	99,64	98,36	<b>99,74</b>	96,79	97,19
	Precisão	99,64	99,64	98,36	<b>99,74</b>	96,79	97,19
	<i>Recall</i>	99,64	99,64	98,36	<b>99,74</b>	96,79	97,19
0,1	Acurácia	98,74	98,79	98,86	<b>99,02</b>	95,43	96,24
	Precisão	98,74	98,79	98,86	<b>99,02</b>	95,43	96,25
	<i>Recall</i>	98,74	98,79	98,86	<b>99,02</b>	95,43	96,24
0,3	Acurácia	96,28	95,69	94,27	<b>96,76</b>	93,02	93,60
	Precisão	96,30	95,71	94,41	<b>96,77</b>	93,02	93,62
	<i>Recall</i>	96,28	95,69	94,27	<b>96,76</b>	93,02	93,60
0,5	Acurácia	92,83	91,34	90,84	<b>92,97</b>	89,06	90,14
	Precisão	92,89	91,40	91,87	<b>93,02</b>	89,09	90,94
	<i>Recall</i>	92,83	91,34	90,84	<b>92,97</b>	89,06	90,14
0,7	Acurácia	89,53	87,72	87,86	<b>89,54</b>	85,57	86,10
	Precisão	89,61	87,80	88,10	<b>89,62</b>	85,61	86,15
	<i>Recall</i>	89,53	87,72	87,86	<b>89,54</b>	85,57	86,10
1,0	Acurácia	<b>84,95</b>	82,41	84,77	83,99	80,08	80,90
	Precisão	<b>84,19</b>	82,25	84,30	83,61	80,00	79,91
	<i>Recall</i>	<b>84,95</b>	82,41	84,77	83,99	80,08	80,90
1,3	Acurácia	<b>80,62</b>	77,70	80,47	79,16	75,51	76,01
	Precisão	<b>80,92</b>	77,90	80,67	79,35	75,71	76,20
	<i>Recall</i>	<b>80,62</b>	77,70	80,47	79,16	75,51	76,01
1,5	Acurácia	<b>77,97</b>	74,86	77,67	75,90	72,92	74,15
	Precisão	<b>78,05</b>	74,98	77,87	75,90	72,98	74,35
	<i>Recall</i>	<b>77,97</b>	74,86	77,67	75,90	72,92	74,15

Fonte: Elaborada pelo autor (2024)

O acréscimo de ruído ocasionou uma diminuição de 20% nas métricas do classificador K-NN, já os melhores classificadores, *XGBoost* e *LightGBM*, sofreram uma degradação de aproximadamente 27%. Ao analisar os MCS, percebe-se também uma variação próxima de 27% nos modelos de seleção estática e de seleção dinâmica, apesar das variações ficarem bem próximas entre os classificadores e os MCS, o *Stacked LR* decaiu mais lentamente, nesse caso, pode-se inferir que níveis maiores ruídos causariam menos impacto a este modelo.

Figura 23 – Acurácia obtida para cada técnica de Seleção Dinâmica com o acréscimo do ruído na Base de Dados 2



Fonte: Elaborada pelo autor (2024)

### 5.3 DESEMPENHO COMPUTACIONAL

Os experimentos foram executados na versão gratuita da plataforma Google Colab <sup>1</sup> e em linguagem Python 3 <sup>2</sup>, nesta versão, o Colab oferece os seguintes recursos computacionais:

**Memória RAM:** 12,7 GB

**Disco:** 107,7 GB

**Núcleos físicos:** 1

**Total de núcleos:** 2

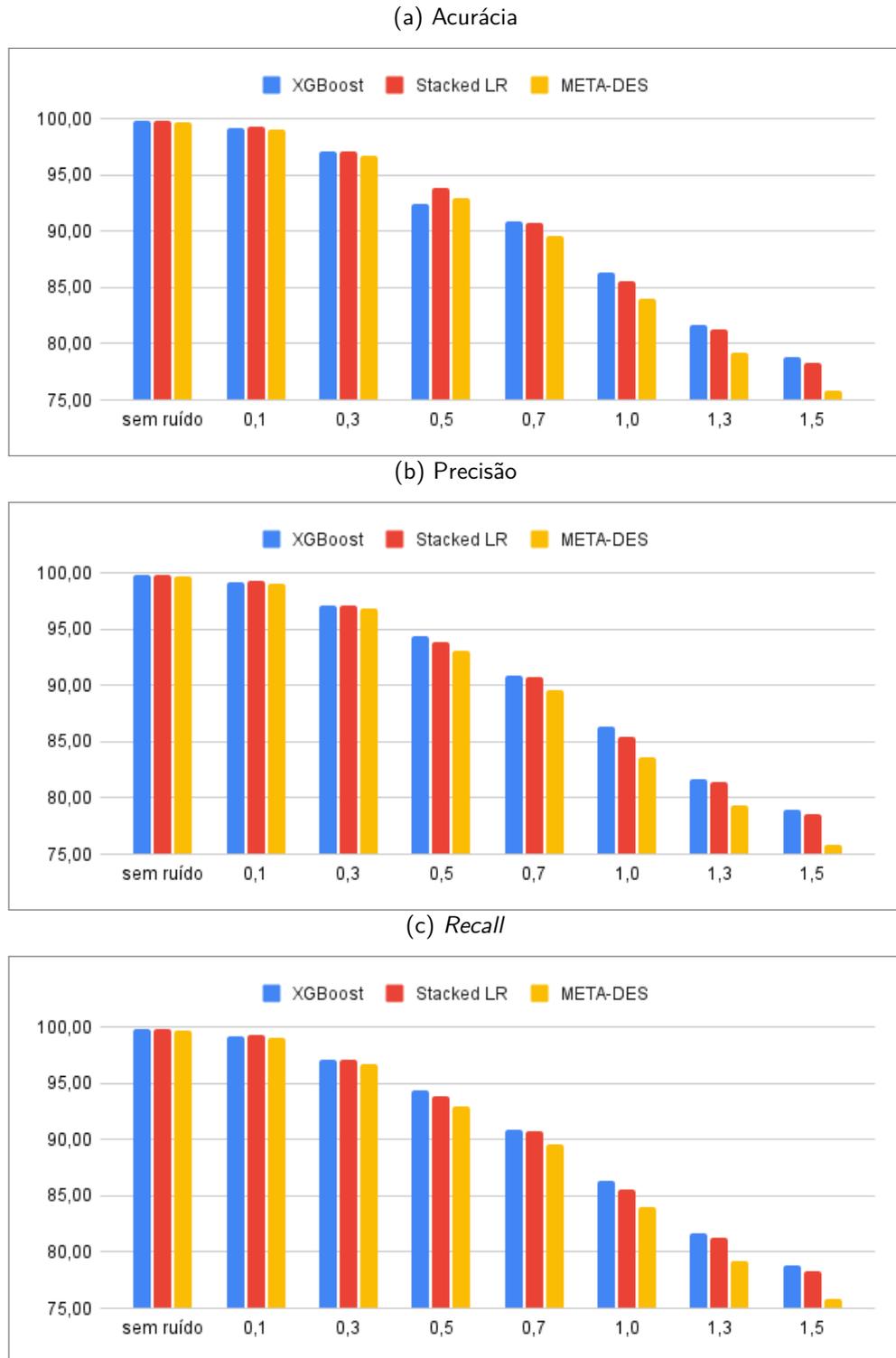
**CPU:** Intel(R) Xeon(R) CPU @ 2.20GHz

Com estes recursos computacionais, a execução do algoritmo levou 42 segundos para a base 1 e 4 minutos e 47 segundos para a base 2. Como subestações de transmissão possuem computadores com configurações bem mais avançadas de processamento e memória, a execução de um algoritmo similar em tempo real levará bem menos tempo, podendo classificar a falha em menos de 1 minuto.

<sup>1</sup> <<https://colab.google/>>

<sup>2</sup> <<https://www.python.org/>>

Figura 24 – Acurácia, Precisão e *Recall* obtidas para os melhores modelos das abordagens avaliadas na Base de Dados 2



Fonte: Elaborada pelo autor (2024)

## 6 CONCLUSÃO

Este trabalho explorou a aplicação de MCS para classificação de falhas em sistemas de transmissão elétrica. Abordagens de conjunto, particularmente *Stacked* e META-DES, superaram métodos tradicionais de modelo único, como Árvore de Decisão e K-NN, demonstrando notável resiliência em ambientes ruidosos. A robustez dos modelos *gradient boosting*, como *LightGBM*, *XGBoost* e *CatBoost*, foi evidente, mantendo altos níveis de precisão mesmo com a introdução de ruído significativo. Além disso, métodos de seleção dinâmica, especialmente KNORA-E, KNORA-U e META-DES, provaram ser mais eficazes no tratamento de ruído em comparação aos métodos estáticos. Essas descobertas destacam o potencial das técnicas de conjunto na melhoria da precisão da detecção de faltas em condições desafiadoras, bem como a importância da diversidade de modelos e estratégias de combinação no aprimoramento do desempenho da classificação ao lidar com dados ruidosos. Além disso, os altos valores métricos medidos para os níveis de ruído estudados atestam a confiabilidade do modelo, a melhora dos resultados e da sensibilidade das MCS em relação aos classificadores individuais atestam os ganhos que esses modelos podem trazer para a classificação de faltas.

O trabalho seguiu os critérios necessários para validar os resultados, estudando sistemas elétricos mais próximos da realidade, as simulações seguiram instalações existentes e esquemas consolidados pelo IEEE, além de serem simulados em um dos *softwares* mais utilizados no mundo para estudos de sistemas elétricos, o ATPDraw. Para introdução do ruído, foi utilizado o Ruído Gaussiano, também amplamente utilizado em estudos similares.

A abordagem proposta pode ser aplicada em sistemas industriais e de transmissão de energia para classificar vários tipos de faltas. Quando integrada com recursos computacionais adequados para processamento rápido de dados, ela aprimora as operações do sistema como uma ferramenta complementar para proteção, controle e supervisão. No entanto, é importante enfatizar que a seleção cuidadosa do conjunto de classificadores é essencial para garantir uma abordagem confiável, permitindo que o MCS generalize efetivamente em diversos cenários de sistemas de energia.

Apesar dos resultados serem próximos e dificultarem a análise do melhor modelo, testes estatísticos como Intervalo de Confiança ou Teste de Hipótese não puderam ser considerados, pois após dez execuções os resultados foram os mesmo, indicando não haver variação dos resultados. Os modelos que apresentaram as melhores médias dos resultados obtidos em cada

cenário foram KNORA-U e META-DES, seguidos pelo *LightGBM* para a base 01. Para a base 02, as melhores médias foram de *XGBoost* e *LightGBM*, seguidos pelo KNORA-U.

As duas bases de dados, no geral e para todos os cenários, apresentaram resultados acima dos 90%, esse fato é uma particularidade de dados de faltas de sistemas elétricos, onde há uma grande variação dos valores medidos em corrente e tensão nas fases que entraram em curto-circuito e, conseqüentemente, sofreram a falta. Contudo, mesmo com a facilidade de classificação, o aprimoramento das classificações e a redução da sensibilidade aumentarão a produtividade da operação na correção das faltas. Este fato também pôde ser verificado em estudos similares, como: (BHUIYAN et al., 2022; K et al., 2022; KARAN; YEH, 2020; FAHIM et al., 2020; KURUKURU et al., 2020; ENSINA L.A., 2023)

## 6.1 TRABALHOS FUTUROS

Para trabalhos futuros, métodos de seleção e combinação de classificadores dinâmicos podem ser propostos para lidar com níveis ainda maiores de ruído ou para aplicar essas técnicas a diferentes tipos de falhas e outros componentes de sistemas de energia elétrica. Aplicações voltadas para qualidade de energia também podem ser abordadas, como: oscilações de frequência, distorções harmônicas e controle de potência são distúrbios que podem ter graves conseqüências no SEP e, por dependerem das unidades geradoras para sua correção, possuem tempo de atuação mais longo podendo causar grandes desligamentos no sistema.

## 6.2 LIMITAÇÕES DA PESQUISA

As bases de dados utilizadas no trabalho foram simuladas em *softwares* de análises matemáticas e elétricas, apesar do incremento de ruído para aproximar as medições dos dados reais, as grandezas medidas pelos sistemas de controle forneceriam mais respaldo à pesquisa, contudo esses dados costumam ser sigilosos e os agentes proprietários não os fornecem.

## 6.3 CONTRIBUIÇÕES DA PESQUISA

Abaixo foram listadas as contribuições realizadas por esta pesquisa:

- Avaliação do impacto no desempenho de abordagens de conjunto estático e seleção

dinâmica quando diferentes níveis de ruído são introduzidos;

- Comparação dos MCS com vários classificadores consolidados da literatura em 14 cenários diferentes no total;
- A aplicação prática dos métodos de Sistemas de Classificadores Múltiplos, especificamente modelos de empilhamento e seleção de classificadores dinâmicos, para classificação de falhas em sistemas de transmissão elétrica;
- Aumento da confiabilidade na classificação de faltas de subestações e linhas de transmissão com modelos que são menos susceptíveis ao ruído.

## REFERÊNCIAS

- ABED, N. K.; ABED, F. T.; AL-YASRIY, H. F.; ALRIKABI, H. T. S. Detection of power transmission lines faults based on voltages and currents values using k-nearest neighbors. *International Journal of Power Electronics and Drive Systems (IJPEDS)*, v. 14, n. 2, p. 1033–1043, 2023.
- AEENEH, S.; ZLATANOV, N.; YU, J. New bounds on the accuracy of majority voting for multiclass classification. *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, 2024.
- ANDERSON, T. W.; DARLING, D. A. Asymptotic Theory of Certain "Goodness of Fit" Criteria Based on Stochastic Processes. *The Annals of Mathematical Statistics*, Institute of Mathematical Statistics, v. 23, n. 2, p. 193 – 212, 1952. Disponível em: <<https://doi.org/10.1214/aoms/1177729437>>.
- ASMAN, S. H.; AZIZ, N. F. A.; AMIRULDDIN, U. A. U.; KADIR, M. Z. A. A. Decision tree method for fault causes classification based on rms-dwt analysis in 275 kv transmission lines network. *Applied Sciences*, MDPI, v. 11, n. 9, p. 4031, 2021.
- ATRIGNA, M.; BUONANNO, A.; CARLI, R.; CAVONE, G.; SCARABAGGIO, P.; VALENTI, M.; GRADITI, G.; DOTOLI, M. A machine learning approach to fault prediction of power distribution grids under heatwaves. *IEEE Transactions on Industry Applications*, IEEE, v. 59, n. 4, p. 4835–4845, 2023.
- BHUIYAN, E. A.; AKHAND, M. A.; FAHIM, S. R.; SARKER, S. K.; DAS, S. K. A deep learning through dbn enabled transmission line fault transient classification framework for multimachine microgrid systems. *International Transactions on Electrical Energy Systems*, v. 2022, n. 1, p. 6820319, 2022. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/6820319>>.
- BREIMAN, L. Bagging predictors. *Machine learning*, Springer, v. 24, p. 123–140, 1996.
- CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 785–794. ISBN 9781450342322. Disponível em: <<https://doi.org/10.1145/2939672.2939785>>.
- CORDEIRO, P. R.; CAVALCANTI, G. D.; CRUZ, R. M. Dynamic ensemble algorithm post-selection using hardness-aware oracle. *IEEE Access*, IEEE, 2023.
- CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D. Meta-des: A dynamic ensemble selection framework using meta-learning. *Pattern Recognition*, Elsevier, v. 48, n. 5, p. 1925–1935, 2014.
- CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D. Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, Elsevier, v. 41, p. 195–216, 2018.
- CUI, S.; YIN, Y.; WANG, D.; LI, Z.; WANG, Y. A stacking-based ensemble learning method for earthquake casualty prediction. *Applied Soft Computing*, Elsevier, v. 101, p. 107038, 2021.
- DIETTERICH, T. G. Ensemble methods in machine learning. In: *Multiple Classifier Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. p. 1–15. ISBN 978-3-540-45014-6.

DOBSON, I.; CARRERAS, B. A.; LYNCH, V. E.; NEWMAN, D. E. Complex systems analysis of series of blackouts: Cascading failure, critical points, and self-organization. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, AIP Publishing, v. 17, n. 2, 2007.

ENSINA L.A., O. L. C. R. Fault distance estimation for transmission lines with dynamic regressor selection. *Neural Computing and Applications*, Neural Comput Applic, v. 36, p. 1741–1759, fev. 2023. ISSN 1433-3058. Disponível em: <<https://doi.org/10.1007/s00521-023-09155-y>>.

FAHIM, S. R.; SARKER, S. K.; MUYEEN, S. M.; SHEIKH, M. R. I.; DAS, S. K. Microgrid fault detection and classification: Machine learning based approach, comparison, and reviews. *Energies*, v. 13, n. 13, 2020. ISSN 1996-1073. Disponível em: <<https://www.mdpi.com/1996-1073/13/13/3460>>.

FANG, J.; YANG, F.; CHEN, C.; YANG, Y.; PANG, B.; HE, J.; LIN, H. Power distribution transformer fault diagnosis with unbalanced samples based on neighborhood component analysis and k-nearest neighbors. In: *2021 Power System and Green Energy Conference (PSGEC)*. [S.l.: s.n.], 2021. p. 670–675.

FILHO, J. M.; MAMEDE, D. R. *Proteção de sistemas elétricos de potência*. [S.l.]: Livros Técnicos e Científicos Editora LTDA, 2013.

FRAGOSO, R. C.; CAVALCANTI, G. D.; PINHEIRO, R. H.; OLIVEIRA, L. S. Dynamic selection and combination of one-class classifiers for multi-class classification. *Knowledge-Based Systems*, Elsevier, v. 228, p. 107290, 2021.

FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, JSTOR, p. 1189–1232, 2001.

GHAEMI, A.; SAFARI, A.; AFSHARIRAD, H.; SHAYEGHI, H. Accuracy enhance of fault classification and location in a smart distribution network based on stacked ensemble learning. *Electric Power Systems Research*, Elsevier, v. 205, p. 107766, 2022.

GIACINTO, G.; ROLI, F. Dynamic classifier selection based on multiple classifier behavior. *Pattern Recognition*, Elsevier, v. 34, n. 9, p. 1879–1881, 2001.

GONI, M. F.; NAHIDUZZAMAN, M.; ANOWER, M.; RAHMAN, M.; ISLAM, M.; AHSAN, M.; HAIDER, J.; SHAHJALAL, M. Fast and Accurate Fault Detection and Classification in Transmission Lines using Extreme Learning Machine. *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, v. 3, p. 100107, 2023. ISSN 2772-6711.

HAJIHOSSEINLOU, M.; MAGHSOUDI, A.; GHEZELBASH, R. Stacking: A novel data-driven ensemble machine learning strategy for prediction and mapping of pb-zn prospectivity in varchah district, west iran. *Expert Systems with Applications*, Elsevier, v. 237, p. 121668, 2024.

HARISH, A.; JAYAN, M. Classification of power transmission line faults using an ensemble feature extraction and classifier method. In: SPRINGER. *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020*. [S.l.], 2021. p. 417–427.

HE, X.; QIU, R. C.; AI, Q.; CHU, L.; XU, X.; LING, Z. Designing for situation awareness of future power grids: An indicator system based on linear eigenvalue statistics of large random matrices. *IEEE Access*, IEEE, v. 4, p. 3557–3568, 2016.

- JAMEHBOZORG, A.; SHAHRTASH, S. M. A decision-tree-based method for fault classification in single-circuit transmission lines. *IEEE Transactions on Power Delivery*, IEEE, v. 25, n. 4, p. 2190–2196, 2010.
- JAMIL, M.; SHARMA, S. K.; SINGH, R. Fault detection and classification in electrical power transmission system using artificial neural network. *SpringerPlus*, SpringerOpen, v. 4, n. 1, p. 1–13, 2015.
- JANARTHANAM, K.; KAMALESH, P.; BASIL, T. V.; KOVILPILLAI, A. K. J. Electrical Faults-Detection and Classification using Machine Learning. In: *2022 International Conference on Electronics and Renewable Systems (ICEARS)*. [S.l.: s.n.], 2022. p. 1289–1295.
- JARQUE, C. M.; BERA, A. K. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, v. 6, n. 3, p. 255–259, 1980. ISSN 0165-1765. Disponível em: <<https://www.sciencedirect.com/science/article/pii/0165176580900245>>.
- K, J.; P, K.; V, B. T.; J, J. A. K. Electrical faults-detection and classification using machine learning. In: *2022 International Conference on Electronics and Renewable Systems (ICEARS)*. [S.l.: s.n.], 2022. p. 1289–1295.
- KARAN, S.; YEH, H.-G. Fault classification in microgrids using deep learning. In: *2020 IEEE Green Energy and Smart Systems Conference (IGESSC)*. [S.l.: s.n.], 2020. p. 1–7.
- 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\\_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf)>.
- KHANAM, R.; HUSSAIN, M.; HILL, R.; ALLEN, P. A comprehensive review of convolutional neural networks for defect detection in industrial applications. *IEEE Access*, v. 12, p. 94250–94295, 2024.
- KO, A. H.; SABOURIN, R.; BRITTO, A. S. From dynamic classifier selection to dynamic ensemble selection. *Pattern Recognition*, Elsevier, v. 41, n. 5, p. 1718–1731, 2008.
- KOLMOGOROV, A. N. Sulla determinazione empirica di una legge di distribuzione. *Giornale dell'Istituto Italiano degli Attuari*, v. 4, p. 83–91, 1933.
- KURUKURU, V. S. B.; BLAABJERG, F.; KHAN, M. A.; HAQUE, A. A novel fault classification approach for photovoltaic systems. *Energies*, v. 13, n. 2, 2020. ISSN 1996-1073. Disponível em: <<https://www.mdpi.com/1996-1073/13/2/308>>.
- LANA, J. *ANÁLISE MULTIESCALAR DOS CONDICIONANTES DA GÊNESE E EVOLUÇÃO DE VOÇOROCAS NAS BACIAS DO RIO DAS VELHAS E DO RIO PARAÓPEBA, MG*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DE OURO PRETO, 06 2022.
- LAZZARINI, R.; TIANFIELD, H.; CHARISSIS, V. A stacking ensemble of deep learning models for iot intrusion detection. *Knowledge-Based Systems*, Elsevier, v. 279, p. 110941, 2023.

- LUSKOVÁ, M.; LEITNER, B. Societal vulnerability to electricity supply failure. *Interdisciplinary Description of Complex Systems: INDECS*, Croatian Interdisciplinary Society, v. 19, n. 3, p. 391–401, 2021.
- MAHDIANI M.R., K. E. H. S. H.-S. A. Modeling viscosity of crude oil using k-nearest neighbor algorithm. . *Advances in Geo-Energy Research*, v. 4, n. 4, p. 435–447, 2020.
- MEDEIROS, C. *Curto-Circuitos Trifásicos no Sistema Elétrico*. 2017. Disponível em: <[https://professor.pucgoias.edu.br/sitedocente/admin/arquivosUpload/18795/material/05\)SistEletricosCap05-CurtoTrif.pdf](https://professor.pucgoias.edu.br/sitedocente/admin/arquivosUpload/18795/material/05)SistEletricosCap05-CurtoTrif.pdf)>. Acesso em: 02 fev. 2025.
- MICHALSKI JAIME G. CARBONELL, T. M. M. R. S. *Machine Learning: An artificial intelligence approach*. [S.l.]: Springer Berlin, Heidelberg, 1983. v. 1.
- MRABET, Z. E.; SUGUNARAJ, N.; RANGANATHAN, P.; ABHYANKAR, S. Random forest regressor-based approach for detecting fault location and duration in power systems. *Sensors*, MDPI, v. 22, n. 2, p. 458, 2022.
- NAVADA, A.; ANSARI, A. N.; PATIL, S.; SONKAMBLE, B. A. Overview of use of decision tree algorithms in machine learning. In: IEEE. *2011 IEEE control and system graduate research colloquium*. [S.l.], 2011. p. 37–42.
- OGAR, V. N.; HUSSAIN, S.; GAMAGE, K. A. Transmission line fault classification of multi-dataset using catboost classifier. *Signals*, MDPI, v. 3, n. 3, p. 468–482, 2022.
- O'ROURKE, T. D. Critical infrastructure, interdependencies, and resilience. *BRIDGE-Washington-National Academy of Engineering-*, National Academy of Engineering, v. 37, n. 1, p. 22, 2007.
- PAN, Y. Innovative strategies for cultivating journalism and communication talents in the all-media era based on the light gbm model. *Applied Mathematics and Nonlinear Sciences*, v. 9, n. 1, p. –, 2024. Disponível em: <<https://doi.org/10.2478/amns.2023.1.00366>>.
- PROKHORENKOVA, L.; GUSEV, G.; VOROBEV, A.; DOROGUSH, A. V.; GULIN, A. *CatBoost: unbiased boosting with categorical features*. 2019. Disponível em: <<https://arxiv.org/abs/1706.09516>>.
- SHAPIRO, S. S.; WILK, M. B. An analysis of variance test for normality (complete samples). *Biometrika*, [Oxford University Press, Biometrika Trust], v. 52, n. 3/4, p. 591–611, 1965. ISSN 00063444, 14643510. Disponível em: <<http://www.jstor.org/stable/2333709>>.
- SILVA, R. S. d. *Sistemas Elétricos de Potência I*. [S.l.]: Editora e Distribuidora Educacional S.A, 2019. único.
- SIPAI, U.; JADEJA, R.; KOTHARI, N.; TRIVEDI, T.; MAHADEVA, R.; PATOLE, S. P. Performance evaluation of discrete wavelet transform and machine learning based techniques for classifying power quality disturbances. *IEEE Access*, v. 12, p. 95472–95486, 2024.
- SMIRNOV, N. Table for Estimating the Goodness of Fit of Empirical Distributions. *The Annals of Mathematical Statistics*, Institute of Mathematical Statistics, v. 19, n. 2, p. 279 – 281, 1948. Disponível em: <<https://doi.org/10.1214/aoms/1177730256>>.

- SOUZA, D. F. de; SALOTTI, F. A. M.; SAUER, I. L.; TATIZAWA, H.; ALMEIDA, A. T. de; KANASHIRO, A. G. A performance evaluation of three-phase induction electric motors between 1945 and 2020. *Energies*, v. 15, n. 6, 2022. ISSN 1996-1073. Disponível em: <<https://www.mdpi.com/1996-1073/15/6/2002>>.
- SUTHAHARAN, S.; SUTHAHARAN, S. Decision tree learning. *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*, Springer, p. 237–269, 2016.
- TEWARI, S.; DWIVEDI, U. D. A comparative study of heterogeneous ensemble methods for the identification of geological lithofacies. *Journal of Petroleum Exploration and Production Technology*, v. 10, p. 1849–1868, 2020. ISSN 2190-0566. Disponível em: <<https://doi.org/10.1007/s13202-020-00839-y>>.
- TOMA, R. N.; KIM, C.-H.; KIM, J.-M. Bearing fault classification using ensemble empirical mode decomposition and convolutional neural network. *Electronics*, MDPI, v. 10, n. 11, p. 1248, 2021.
- TOMA, R. N.; KIM, J.-M. Bearing fault classification of induction motors using discrete wavelet transform and ensemble machine learning algorithms. *Applied Sciences*, MDPI, v. 10, n. 15, p. 5251, 2020.
- VAISH, R.; DWIVEDI, U.; TEWARI, S.; TRIPATHI, S. M. Machine learning applications in power system fault diagnosis: Research advancements and perspectives. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 106, p. 104504, 2021.
- VISWAVANDYA, M.; PATEL, S.; SAHOO, K. Analysis and comparison of machine learning approaches for transmission line fault prediction in power systems. *Journal of Research in Engineering and Applied Sciences*, v. 6, n. 1, p. 24–31, 2021.
- WANG, B.; YANG, K.; WANG, D.; CHEN, S.-z.; SHEN, H.-j. The applications of xgboost in fault diagnosis of power networks. In: IEEE. *2019 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*. [S.l.], 2019. p. 3496–3500.
- WOLOSZYNSKI, T.; KURZYNSKI, M.; PODSIADLO, P.; STACHOWIAK, G. W. A measure of competence based on random classification for dynamic ensemble selection. *Information Fusion*, Elsevier, v. 13, n. 3, p. 207–213, 2012.
- WOLPERT, D. H. Stacked generalization. *Neural networks*, Elsevier, v. 5, n. 2, p. 241–259, 1992.
- WOLPERT, D. H. The lack of a priori distinctions between learning algorithms. *Neural Computation*, MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info . . . , v. 8, n. 7, p. 1341–1390, 1996.
- WOODS, K.; KEGELMEYER, W. P.; BOWYER, K. Combination of multiple classifiers using local accuracy estimates. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 19, n. 4, p. 405–410, 1997.
- YAGAN, O.; QIAN, D.; ZHANG, J.; COCHRAN, D. Optimal allocation of interconnecting links in cyber-physical systems: Interdependence, cascading failures, and robustness. *IEEE Transactions on Parallel and Distributed Systems*, IEEE, v. 23, n. 9, p. 1708–1720, 2012.

YALÇIN, T.; SOLÀ, P.; STEFANIDOU-VOZIKI, P.; DOMINGUEZ-GARCIA, J. L.; DEMIRDELEN, T. Exploiting digitalization of solar pv plants using machine learning: Digital twin concept for operation. *Energies*, v. 16, p. 5044, 06 2023.

ZHU, X.; LI, J.; REN, J.; WANG, J.; WANG, G. Dynamic ensemble learning for multi-label classification. *Information Sciences*, Elsevier, v. 623, p. 94–111, 2023.