

UNIVERSIDADE FEDERAL DO PAMPA

Eduardo Davila Kilian

**DETECÇÃO DE FALTAS DE ALTA IMPEDÂNCIA: UMA ABORDAGEM
UTILIZANDO DIFERENTES TÉCNICAS DE APRENDIZADO DE MÁQUINA**

Alegrete

2024

Eduardo Davila Kilian

**DETECÇÃO DE FALTAS DE ALTA IMPEDÂNCIA: UMA ABORDAGEM
UTILIZANDO DIFERENTES TÉCNICAS DE APRENDIZADO DE MÁQUINA**

Dissertação apresentada ao Programa de Pós-graduação *Stricto Sensu* em Engenharia Elétrica da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Eduardo Machado dos Santos

Alegrete
2024

Ficha catalográfica elaborada automaticamente com os dados fornecidos
pelo(a) autor(a) através do Módulo de Biblioteca do
Sistema GURI (Gestão Unificada de Recursos Institucionais) .

K48d Kilian, Eduardo Davila
DETECÇÃO DE FALTAS DE ALTA IMPEDÂNCIA: UMA ABORDAGEM
UTILIZANDO DIFERENTES TÉCNICAS DE APRENDIZADO DE MÁQUINA /
Eduardo Davila Kilian.
132 p.

Dissertação(Mestrado)-- Universidade Federal do Pampa,
MESTRADO EM ENGENHARIA ELÉTRICA, 2024.

"Orientação: Eduardo Machado dos Santos".

1. Falta de Alta Impedância. 2. Aprendizagem de Máquina. 3.
Proteção de Sistemas Elétricos de Potência. 4. Sistema de
Distribuição. 5. Arco Elétrico. I. Título.

EDUARDO DAVILA KILIAN

**DETECÇÃO DE FALTAS DE ALTA IMPEDÂNCIA: UMA ABORDAGEM UTILIZANDO
DIFERENTES TÉCNICAS DE APRENDIZADO DE MÁQUINA**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Mestre em Engenharia Elétrica.

Dissertação defendida e aprovada em: 04/12/2024.

Banca examinadora:

Prof. Dr. Eduardo Machado dos Santos

Orientador

UNIPAMPA

Prof. Dr. Ghendy Cardos Junior

UFSM

Prof. Dr. Chrystian Dalla Lana da Silva

UNIPAMPA



Assinado eletronicamente por **GHENDY CARDOSO JUNIOR, Usuário Externo**, em 04/12/2024, às 11:12, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **CHRYSYTIAN DALLA LANA DA SILVA, PROFESSOR DO MAGISTERIO SUPERIOR**, em 04/12/2024, às 11:13, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **EDUARDO MACHADO DOS SANTOS, PROFESSOR DO MAGISTERIO SUPERIOR**, em 04/12/2024, às 15:13, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



A autenticidade deste documento pode ser conferida no site https://sei.unipampa.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1606769** e o código CRC **B78FA1F5**.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por me guiar diante dos caminhos da vida e por me dar a oportunidade de estar, continuamente, aprendendo e me desenvolvendo como ser humano.

Sou grato a minha Mãe, Jussara Davila Kilian, e meu Pai, Leomar Vilson Kilian, por sempre me apoiarem e estarem ao meu lado nessa jornada que tenho trilhado. A ajuda deles tem um papel fundamental, não somente na minha formação acadêmica, como também na minha formação como cidadão crítico e pensante na sociedade.

Meus agradecimentos também vão para o meu orientador, Prof. Dr. Eduardo Machado dos Santos, o qual teve um papel vital na forma como aprendi a fazer pesquisa e cuja a ajuda e os aconselhamentos contribuíram imensamente na minha habilidade de abordar e solucionar problemas. Além disso, agradeço-o pela amizade e confiança ao longo desses anos, o que certamente teve um papel fundamental no meu desenvolvimento como engenheiro e pesquisador.

Agradeço aos meus amigos de longa data Fabiano, Abner Arthur e Cláudio, os quais sempre estiveram ao meu lado nas mais diversas fases da minha vida e cuja a companhia sempre me rendeu ótimas conversas e risadas.

Também agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo suporte financeiro, essencial para a realização desta pesquisa ao longo dos dois anos.

Por fim, agradeço ao Prof. Dr. Chrystian Dalla Lana da Silva e a todo o pessoal do GESEP pela companhia ao longo de todos esses anos de pesquisa.

RESUMO

Este Trabalho apresenta uma abordagem para a aplicação de técnicas de *machine learning* (ML) no contexto da detecção de faltas de alta impedância (FAI) em redes elétricas de distribuição. A metodologia proposta visa detectar a ocorrência de FAIs a partir da extração de características do sinal de corrente da subestação, evitando falsos alarmes de outros eventos, como chaveamento de bancos de capacitores, energização de transformadores e entrada de carga. Para isso, as características utilizadas são extraídas da distribuição estatística dos dados dentro de uma janela móvel, que percorre o sinal de corrente. Além disso, é aplicada uma técnica de seleção de características para avaliar o impacto de cada uma delas no desempenho dos algoritmos de ML e verificar se a remoção de alguma característica afeta negativamente a capacidade de classificação do modelo. Foram realizadas 1722 simulações, sendo 864 casos de FAI e 856 de outros eventos que ocorrem no sistema elétrico de distribuição. Adicionalmente, foi criado um conjunto de validação composto por 574 casos, sendo 288 de FAI e 286 de outros eventos. Por fim, um conjunto de teste foi utilizado, com o mesmo número de casos do conjunto de validação. No conjunto de teste, para avaliar a robustez da metodologia frente a diferentes níveis de ruído, foram introduzidos valores de SNR entre 20 dB e 100 dB. Os resultados mostraram que a metodologia combinada com diferentes modelos de ML, conseguiu manter métricas de desempenho acima de 90% para níveis de ruído superiores a 40 dB, mostrando-se uma ferramenta robusta para a correta detecção de FAIs.

Palavras-chave: Falta de Alta Impedância, Aprendizagem de Máquina, Proteção de Sistemas Elétricos de Potência, Sistema de Distribuição, Arco Elétrico.

ABSTRACT

This study presents an approach for applying machine learning (ML) techniques in the context of high-impedance fault (HIF) detection in power distribution networks. The proposed methodology aims to detect the occurrence of HIFs by extracting features from the substation current signal while avoiding false alarms triggered by other events, such as capacitor bank switching, transformer energization, and load connection. To achieve this, the selected features are extracted from the statistical distribution of the data within a moving window that scans the current signal. Additionally, a feature selection technique is applied to assess the impact of each feature on the performance of the ML algorithms and to determine whether the removal of any feature negatively affects the model's classification capability. A total of 1,722 simulations were conducted, comprising 864 cases of HIF and 856 cases of other events occurring in the distribution system. Furthermore, a validation set was created, consisting of 574 cases, with 288 HIF cases and 286 other events. Finally, a test set was used, containing the same number of cases as the validation set. In the test set, to evaluate the robustness of the methodology against different noise levels, SNR values ranging from 20 dB to 100 dB were introduced. The results demonstrated that the methodology, when combined with different ML models, maintained performance metrics above 90% for noise levels above 40 dB, proving to be a robust tool for accurate HIF detection.

Keywords: High Impedance Fault, Machine Learning, Electric Power Systems Protection, Distribution System, Electric Arc.

LISTA DE ILUSTRAÇÕES

Figura 1 – Formato da corrente de arco.	26
Figura 2 – Distribuição dos valores de tensão na formação do arco elétrico da FAI. . .	27
Figura 3 – Corrente de FAI que apresenta comportamento aleatório, não-linearidade e assimetria.	28
Figura 4 – Exemplo de <i>Buildup e Shoulder</i>	29
Figura 5 – Exemplo de intermitência em um sinal de FAI.	30
Figura 6 – Modelo de Arco Emanuel.	31
Figura 7 – Exemplo de comportamento da segunda e terceira harmônica em relação a ΔV e $\tan \theta$	32
Figura 8 – Modelo de arco usando duas resistências variantes no tempo.	32
Figura 9 – Exemplo de curva característica tensão-corrente para uma FAI em regime permanente.	33
Figura 10 – Exemplo de curva característica tensão-corrente linearizada.	33
Figura 11 – Resistências variantes no tempo nos instantes de máxima tensão e corrente. .	34
Figura 12 – Modelo de arco usado por Gautam e Brahma.	35
Figura 13 – Arcos Emanuel em paralelo.	36
Figura 14 – Ocorrência de vários arcos em uma superfície de cimento.	36
Figura 15 – Diferença entre problemas de classificação e regressão.	39
Figura 16 – <i>Overfitting e Underfitting</i>	40
Figura 17 – Procedimento do <i>k-fold Cross-Validation</i>	41
Figura 18 – Exemplo de Árvore de Decisão.	43
Figura 19 – Funcionamento do algoritmo <i>Ball Tree</i>	46
Figura 20 – <i>Bagging e Boosting</i>	47
Figura 21 – Treinamento usando amostras <i>Bootstrap</i>	48
Figura 22 – Funcionamento da <i>Random Forest</i>	49
Figura 23 – Funcionamento do <i>Adaboost</i>	50
Figura 24 – Exemplo de um espaço de características.	54
Figura 25 – Fronteiras de decisão definidas por diferentes algoritmos.	55
Figura 26 – Diferentes formas de curtose.	57
Figura 27 – Diferentes tipos de assimetria.	58
Figura 28 – Sequência de remoção das características e suas respectivas importâncias. .	63
Figura 29 – Desempenho da <i>Random Forest</i> para diferentes números de características. .	64
Figura 30 – Dados de Treinamento: Curtose x Assimetria Amostral.	65
Figura 31 – Fronteiras de decisão delimitadas pelo treinamento de cada modelo de ML. .	68
Figura 32 – Análise dos modelos treinados para DT, KNN e RF.	71
Figura 33 – Análise dos modelos treinados para ET, AdaB e GBM.	72
Figura 34 – Análise dos modelos treinados para XGBM, LGBM e CGBM.	73
Figura 35 – Fluxograma da metodologia proposta.	74

Figura 36 – Exemplo de operação da lógica para um caso de FAI.	76
Figura 37 – Avaliação de diferentes critérios de detecção com a <i>Random Forest</i>	77
Figura 38 – Sistema teste utilizado.	79
Figura 39 – Aspectos do Modelo de Arco 1.	81
Figura 40 – Aspectos do Modelo de Arco 2.	82
Figura 41 – Aspectos do Modelo de Arco 3.	83
Figura 42 – Aspectos do Modelo de Arco 4.	83
Figura 43 – Exemplo de situação envolvendo rompimento do condutor seguido por arco elétrico.	85
Figura 44 – Distribuição dos Valores de Pico de Regime das Correntes de FAI Simuladas.	86
Figura 45 – Operação no tempo para o caso com modelo de arco 1.	91
Figura 46 – Operação no tempo para o modelo de arco 2.	92
Figura 47 – Operação no tempo para o caso com o modelo de arco 3.	94
Figura 48 – Operação no tempo para o caso com modelo de arco 4.	95
Figura 49 – Operação no tempo para o chaveamento de carga resistiva.	97
Figura 50 – Operação no tempo para o chaveamento de carga capacitiva.	98
Figura 51 – Operação no tempo para o chaveamento de carga indutiva.	100
Figura 52 – Operação no tempo para o caso de energização de transformadores.	101
Figura 53 – Operação no tempo para o caso de curto-circuito bifásico-terra.	102
Figura 54 – Matriz de confusão e erros dos classificadores.	104
Figura 55 – Tempos de detecção dos classificadores para os casos de FAI.	107
Figura 56 – Comportamento dos erros de detecção da metodologia para os diferentes classificadores em diversos níveis de ruído.	109
Figura 57 – Erros de detecção por tipo de evento para níveis de SNR de 45, 40 e 35 dB.	110
Figura 58 – Erros de detecção por tipo de evento para níveis de SNR de 30, 25 e 20 dB.	111
Figura 59 – Comportamento dos erros para FAI com e sem rompimento do condutor para diferentes níveis de ruído.	112

LISTA DE TABELAS

Tabela 1 – Quantidade de dados usados para o treinamento dos modelos de ML.	66
Tabela 2 – Quantidade de dados usados para a validação modelos de ML.	70
Tabela 3 – Resultado do cálculo da área usando a regra do trapézio.	77
Tabela 4 – Valores de impedância acumulados por trecho.	80
Tabela 5 – Distâncias de cada ponto do sistema até a subestação.	80
Tabela 6 – Características de cada uma das cargas.	81
Tabela 7 – Intervalos nos quais os parâmetros dos arcos elétricos foram variados à cada simulação.	84
Tabela 8 – Porcentagem de queda de corrente resultante da desconexão de um determinado trecho.	85
Tabela 9 – Intervalos utilizados para as simulações de cada tipo de transitório.	87
Tabela 10 – Métricas de desempenho dos classificadores.	105
Tabela 11 – Métricas de desempenho médias englobando todos os classificadores.	105
Tabela 12 – Número de classificações erradas e <i>ranking</i> dos melhores tempos médios de detecção das FAIs.	108
Tabela 13 – Métricas Gerais dos Classificadores.	113
Tabela 14 – Melhores e piores resultados de cada uma das métricas para diferentes níveis de ruído, desconsiderando o AdaB.	114
Tabela 15 – Melhores e piores resultados dos tempos médios de detecção, desconsiderando o AdaB.	116

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Considerações Gerais	17
1.2	Motivação	17
1.3	Objetivos	18
1.4	Revisão Bibliográfica	18
1.5	Estrutura do Trabalho	22
2	FUNDAMENTAÇÃO TEÓRICA	25
2.1	Considerações Gerais	25
2.2	Falta de Alta Impedância	25
2.2.1	Características da Corrente de FAI	25
2.2.2	Deteção da FAI como uma Tarefa de Classificação de Padrões	28
2.2.3	Modelos de Arco Elétrico	30
2.2.3.1	Modelo Emanuel	30
2.2.3.2	Resistências em série e variantes no tempo	31
2.2.3.3	Modelo apresentado por Gautam	34
2.2.3.4	Modelo apresentado por Sedighi	35
2.2.3.5	Modelo de FAI apresentado por Torres	37
2.3	<i>Machine Learning</i>	38
2.3.1	Problemas de Classificação e Regressão no Contexto de <i>Supervised Learning</i>	38
2.3.2	<i>Overfitting, Underfitting e Cross-Validation</i>	39
2.4	Algoritmos de Classificação	41
2.5	Árvores de Decisão (DT)	42
2.5.1	O Algoritmo ID3	42
2.5.2	O Algoritmo C4.5	44
2.5.3	O Algoritmo CART	45
2.6	<i>K-Nearest Neighbors</i> (KNN)	45
2.7	Combinação de Classificadores (Métodos <i>Ensemble</i>)	46
2.7.1	<i>Random Forest</i> (RF)	47
2.7.2	<i>Extremely Randomized Trees</i> (ET)	48
2.7.3	<i>Adaboost</i> (AdaB)	49
2.7.4	<i>Gradient Boosting Machine</i> (GBM)	49
2.8	Métodos <i>Wrappers</i> para Seleção de Características	50
2.8.1	Seleção Baseada em Árvores	51
2.9	Considerações Finais	52
3	METODOLOGIA PROPOSTA	53

3.1	Considerações Gerais	53
3.2	Extração de Características	53
3.2.1	Envoltória Superior e Inferior	54
3.2.2	Área da Envoltória Superior	56
3.2.3	Média	57
3.2.4	Curtose	57
3.2.5	Assimetria Amostral	58
3.3	Banco de Dados	58
3.3.1	Etapas de Criação do Banco de Dados	59
3.3.2	Separação dos Casos	59
3.3.3	Balanceamento dos Dados	60
3.4	Métricas de Desempenho do Classificador em Relação ao Banco de Dados	61
3.5	Seleção de Características	62
3.6	Treinamento e Validação dos Modelos de ML	64
3.7	Lógica de Operação	72
3.8	Escolha do Critério de Detecção	75
3.9	Considerações Finais	77
4	SIMULAÇÕES E RESULTADOS	79
4.1	Considerações Gerais	79
4.2	Sistema Teste	79
4.2.1	Simulações de FAI	81
4.2.2	Tipos de Eventos Simulados	86
4.3	Métricas de Desempenho da Metodologia em Relação aos Casos de Teste	88
4.4	Resultados	89
4.4.1	Operação da Metodologia no Tempo	90
4.4.1.1	Caso de FAI com o modelo de arco 1	90
4.4.1.2	Caso de FAI com o modelo de arco 2	92
4.4.1.3	Caso de FAI com o modelo de arco 3	93
4.4.1.4	Caso de FAI com o modelo de arco 4	95
4.4.1.5	Caso de entrada de carga resistiva	96
4.4.1.6	Caso de entrada de carga capacitiva	97
4.4.1.7	Caso de entrada de carga indutiva	99
4.4.1.8	Caso de energização de transformador	100
4.4.1.9	Caso de curto-circuito	102
4.4.2	Métricas Gerais	103
4.5	Métricas Gerais do Banco de Teste com Adição de Ruído	108
4.6	Considerações Finais	116
5	CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	119

5.1	Sugestões para Trabalhos Futuros	120
	REFERÊNCIAS	123

1 INTRODUÇÃO

1.1 Considerações Gerais

A Falta de Alta impedância (FAI) ocorre devido ao contato de um condutor energizado do sistema elétrico de potência (SEP) com um objeto ou superfície de alta impedância. Infelizmente, devido à natureza altamente resistiva desse objeto ou superfície de contato, a corrente resultante não é detectada pelos dispositivos convencionais de proteção. Além disso, juntamente com essa falta, é comum o aparecimento de um arco elétrico, o qual é resultado do rompimento do dielétrico causado pelos níveis de tensão do SEP. Logo, embora esse problema não possa ser detectado pelos dispositivos de sobrecorrente, a presença do arco elétrico deixa uma assinatura na forma de onda de corrente, a qual permite identificar a ocorrência dessa falta. Entretanto, por causa da natureza complexa e aleatória desse tipo de problema, mesmo que existam várias propostas na literatura, nenhuma delas apresentou uma solução definitiva, tanto em termos de implementação quanto em termos de confiabilidade. Dessa forma, o estudo de técnicas de detecção de faltas de alta impedância ainda é um tópico que permite muitas contribuições e melhoramentos, permitindo assim que várias soluções ainda possam ser apresentadas.

1.2 Motivação

Uma vez que a FAI ameaça a integridade física da população através de um contato não intencional com o cabo energizado ou através de incêndios resultantes do contato do condutor com algum material inflamável, o que também pode causar danos materiais e ambientais, é importante que sua ocorrência seja detectada no menor tempo possível. Entretanto, já que essa falta apresenta uma impedância elevada e de comportamento não-linear, a corrente resultante desse problema não é o suficiente para sensibilizar os dispositivos convencionais de proteção. Assim, essa falta acaba muitas vezes não sendo detectada, oferecendo riscos às pessoas e animais que venham a transitar próximos ao local (Ghaderi; III; Mohammadpour, 2017; Wang; Dehghanian, 2020; Bhandia *et al.*, 2020; Santos *et al.*, 2016).

Esse é um problema tão sério que vários estudos foram conduzidos para entender a frequência de sua ocorrência e a capacidade dos dispositivos convencionais de proteção de lidar com essa falta. Por exemplo, segundo o *Power System Relaying Committee (PSRC)*, os dispositivos convencionais de proteção detectaram e resolveram apenas 35 situações de FAI de um total de 200 ensaios, ou seja, 17,5 % (Bhandia *et al.*, 2020; Wang; Dehghanian, 2020). Além disso, como discutido em Wei *et al.* (2020), de acordo com as estatísticas registradas, 10 - 20 % das faltas no sistema de distribuição são FAI, podendo esse número ser maior se forem considerados os casos não registrados. Também existem dados a respeito de sua letalidade. Nesse sentido, Lopes *et al.* (2020) discutem que no Brasil a FAI é uma das principais causas de morte em sistemas de distribuição e que, na Austrália em 2009, faltas no sistema elétrico causaram cinco das queimadas do *Black Saturday Bushfire*, o qual causou mais 170 mortes, além de vários feridos.

Mesmo com todas essas estatísticas, ainda não existe um método de detecção que seja capaz de detectar de forma geral as FAIs e, portanto, ainda são necessários vários estudos no sentido de melhorar os métodos de detecção ou trazer novas propostas para o tema. Apesar disso, os trabalhos presentes na literatura dos últimos anos têm mostrado que o uso da inteligência artificial (IA) tem um grande potencial para melhorar o desempenho dos esquemas de proteção, mostrando que sua aplicação é uma tendência promissora (Wang; Dehghanian, 2020; Moloi; Davidson, 2022).

1.3 Objetivos

Esse trabalho tem como objetivo geral apresentar uma abordagem baseada em ML para detectar a ocorrência de FAIs a partir de características extraídas do sinal de corrente no domínio do tempo. Dessa forma, possibilitando avaliar como o uso de diferentes algoritmos de ML, em diferentes cenários, afetam a correta classificação dos eventos. Para atingir esse propósito, alguns objetivos específicos foram delimitados, sendo estes listados a seguir:

- Implementar diferentes modelos de arco elétrico para que seja possível analisar o desempenho da metodologia para diferentes modelagens de FAI;
- Criar um sistema teste e simular casos de chaveamento de banco de capacitores, chaveamento de cargas indutivas, curtos-circuitos monofásicos, bifásicos, bifásicos-terra, trifásicos, trifásicos-terra, energização de transformadores, entrada de carga e FAI, de modo a gerar uma diversa gama de casos;
- Implementar ferramentas de extração de característica e, usando metodologias de *machine learning*, avaliar o impacto de cada uma dessas características no desempenho da metodologia;
- Treinar diferentes modelos de *machine learning* e depois usá-los no banco de casos de teste para avaliar a sua performance utilizando diferentes métricas de desempenho; e
- Testar o desempenho da metodologia frente a diferentes níveis de ruído introduzidos nos sinais de corrente dos casos de teste.

1.4 Revisão Bibliográfica

Ao longo das décadas, vários trabalhos foram sendo desenvolvidos no sentido de criar novas metodologias para a detecção de FAIs. Por exemplo, em Chinchali (1988), foi proposta uma técnica de detecção de faltas de alta impedância baseada no aumento da energia no sinal das frequências 180 Hz e 210 Hz da corrente em 60 Hz. Essa metodologia apresenta três níveis para a realização da detecção, sendo estes os níveis de distúrbio, evento e falta, os quais vão sendo avançados de acordo com o comportamento da energia do sinais em relação a média calculada até então.

No artigo de Lien *et al.* (1999), é proposta uma técnica para detectar a ocorrência de FAI, a qual se utiliza da corrente de neutro para realizar a detecção. Essa metodologia funciona monitorando a variância da energia das harmônicas de segunda, quarta e sexta ordem. Assim, a partir desses valores de variância obtidos, é realizada a comparação com um valor limítrofe adaptativo, o qual é construído a partir da utilização de ferramentas estatísticas. Além disso, são utilizados contadores que vão sendo acumulados de acordo com as amostras de variância obtidas, sendo estas usadas posteriormente para completar o critério de detecção da FAI.

Sultan, Swift e Fedirchuk (1994) apresentam uma metodologia que utiliza a assimetria do sinal de corrente para realizar a detecção da FAI. Para isso, o valor de assimetria entre o pico positivo e negativo obtido para um determinado ciclo é comparado com a assimetria calculada para o ciclo seguinte. Além disso, é realizada uma comparação entre as amplitudes dos picos positivos do sinal, assim como dos picos negativos, buscando detectar possíveis aumentos no valor da corrente. Dessa forma, caso os valores de assimetria e crescimento de corrente se tornem maiores do que 5% dos valores nominais, é atualizada a pontuação da saída do detector. Esses valores da saída são então comparados com um valor de referência, de modo a detectar se está ou não ocorrendo um arco elétrico.

Sheng e Rovnyak (2004) apresentam uma metodologia baseada em *Decision Trees* para realizar a detecção da FAI. Nesse trabalho, foram utilizadas como entrada do algoritmo características como o valor rms da corrente, a magnitude da segunda, terceira e quinta harmônicas e o ângulo da terceira harmônica. Essas características são então enviadas para a *Decision Tree* já treinada, a qual é responsável por indicar, em sua saída, se os dados são relativos a uma FAI ou não.

Em Lai *et al.* (2005), são obtidos os coeficientes *wavelet* a partir da aplicação da *Discrete Wavelet Transform* (DWT) nos sinais de corrente e tensão, usando a db4 como *wavelet* mãe. Desse modo, é utilizado um algoritmo de *Machine Learning* conhecido como *Nearest Neighbourhood Rule* (NNR) para realizar a classificação de padrões com base no valor rms calculado a partir dos coeficientes da *wavelet*.

Em Gautam e Brahma (2012), foi utilizada a morfologia matemática para realizar a extração de características do sinal de tensão. Nesse sentido, é utilizada a *Closing-Opening-Difference-Operation* (CODO) para detectar a ocorrência de perturbações no sinal de tensão, as quais aparecem na forma de *spikes* na saída dessa operação, necessitando que sejam 15% maiores do que o máximo valor registrado da saída CODO durante a operação normal. Após detectada a perturbação, são definidos dois tempos, o tempo de espera e o tempo de *reset*, os quais são utilizados na etapa de detecção da FAI. O primeiro é responsável por evitar os múltiplos *spikes* consecutivos que aparecem na saída do CODO durante a ocorrência de diferentes transitórios como, por exemplo, chaveamento de banco de capacitores. Dessa forma, os *spikes* só passam a ser contabilizados após transcorrido esse tempo. Já o segundo, é relativo ao tempo de espera entre um *spike* e outro, sendo descartada a possibilidade de FAI caso um segundo *spike* não seja identificado nesse intervalo.

O método proposto por Thomas, Bhaskar e Prakash (2016) utiliza as medições de tensão do alimentador para realizar a detecção da FAI em situações onde há rompimento do condutor. Para isso, são utilizadas as componentes de sequência dessas medições para monitorar os níveis de desbalanço de tensão. A detecção ocorre se a componente de sequência positiva cair abaixo de um valor limítrofe e a porcentagem das tensões de sequência zero e negativa estiverem dentro de um intervalo específico estabelecido para cada uma.

Já Lima *et al.* (2021) utilizaram *Short-Time Fourier Transform* (STFT) para extrair componentes harmônicas de baixa frequência do sinal de corrente do sistema, como a magnitude e fase da terceira harmônica e magnitude da segunda e quinta harmônicas. Essas características são então monitoradas e, caso a magnitude da terceira harmônica permaneça acima de um valor limítrofe por um intervalo de tempo pré-estabelecido, as magnitudes da segunda e quinta harmônicas também são testadas. A falta de alta impedância é detectada se a magnitude de cada uma dessas harmônicas estiver acima do seu respectivo valor limítrofe estabelecido. Em paralelo, caso o valor do ângulo da fase da terceira harmônica ficar abaixo de um valor pré-estabelecido pelo mesmo intervalo de tempo, a FAI também é detectada.

Também, uma metodologia que usa transformadas *wavelet* e redes neurais evolutivas para detectar FAI foi apresentada em Silva *et al.* (2018). Nesse trabalho, diferentes famílias de *wavelet* foram estudadas de modo a obter as características as quais melhor diferenciavam uma FAI dos demais eventos. Além disso, abordando a falta de alta impedância como um problema variante no tempo, o uso de uma técnica capaz de se adaptar a mudanças no sistema de forma *online* como as redes neurais evolutivas foi apresentada como a solução para lidar com esse problema.

No trabalho de Kavi, Mishra e Vilathgamuwa (2018) foram utilizadas técnicas de morfologia matemática (MM), onde as características não-lineares da MM são exploradas por uma aplicação em cascata de várias funções de filtragem de modo a formar um detector de faltas morfológico de múltiplos estágios para a extração de características da FAI. Com isso, o objetivo é utilizar o comportamento aleatório e não-linear da corrente resultante do arco elétrico para realizar a detecção da falta de alta impedância a partir desse detector de faltas morfológico.

Ainda, Sarwagya, De e Nayak (2018) apresentaram uma abordagem baseada na soma ao longo de um ciclo de parâmetros calculados a partir da tensão residual, chamados de componentes de sobreposição, para realizar a detecção da FAI. Uma vez que essa soma for maior que um dado valor limítrofe, é esperado o tempo de 0,1 s e então é verificado novamente se essa soma ainda está acima do referido valor. Caso essa condição ainda seja satisfeita, então a lógica detecta aquele evento como sendo uma falta de alta impedância.

Além destes trabalhos, em Gomes, Ozansoy e Ulhaq (2018), foram usadas as componentes de alta frequência do sinal de tensão para realizar a detecção de FAIs em contato com a vegetação. Para isso, foram extraídas características como os coeficientes de detalhe da transformada *wavelet*, os seus picos, a energia e a densidade espectral de potencia (PSD, do inglês *Power Spectral Density*) do sinal, sendo essas características usadas juntamente com o

algoritmo *AdaBoost* para realizar a detecção da FAI.

No artigo de Lima, Brito e Souza (2019), é aplicada a transformada de *Stockwell* no sinal de corrente da subestação para extrair o valor do ângulo de fase da sua terceira harmônica, o qual tem o seu desvio padrão monitorado. A falta de alta impedância é detectada quando esse desvio padrão fica abaixo de um valor limítrofe, auto-adaptativo, durante um determinado intervalo de tempo.

Destaca-se ainda que vários métodos de *machine learning* como, por exemplo, *Adaptative Neuro Fuzzy Inference systems* (ANFIS), *Support Vector Machine* (SVM) e *Multilayer Perceptron* (MLP) para detectar a FAI foram utilizados em Veerasamy *et al.* (2019). Tal detecção é feita aplicando a transformada *wavelet* de modo a decompor os sinais de corrente das três fases em coeficientes de detalhe e aproximação, é calculado o valor do desvio padrão para o sinal aproximação com os coeficientes de detalhe do nível 1 e nível 5, sendo esse desvio padrão usado por esses algoritmos para fazer a classificação da falta.

Gadanayak e Mallick (2019) propuseram a análise do conteúdo inter-harmônico do sinal de corrente. Nesse método, o espectro harmônico do sinal é obtido a partir da *Maximum Overlap Discrete Wavelet Packet Transform* (MODWPT) e, então, as componentes inter-harmônicas são separadas das componentes inteiras usando o *Modified Empirical Mode Decomposition Based on Special Knots* (KEMD). Após essa separação, é então calculada a energia das inter-harmônicas e, com base nisso, a metodologia decide se o evento é uma FAI ou não.

Já Wang e Dehghanian (2020) fazem uso de uma *Convolutional Neural Network* (CNN) para a detecção da FAI. Nesse método, os escalogramas dos sinais de corrente das três fases são obtidos usando *Pseudo-Continuous Quadrature Wavelet Transform* (PCQWT) e então quantizados em imagens digitais. A CNN é alimentada com essas imagens de forma a identificar características da FAI a partir dos escalogramas e assim realizar a sua classificação.

Por sua vez, Veerasamy *et al.* (2021) aplicam a DWT para extrair os coeficientes de detalhe e aproximação dos sinais de corrente das três fases do sistema. Após isso, é calculado um valor de energia a partir desses coeficientes para, então, realizar a classificação usando o *Long-Short Term Memory Recurrent Neural Networks* (LSTM RNN).

Além disso, Roy e Debnath (2021) calculam a PSD da matriz de covariância da transformada *Wavelet* e, após isso, obtêm o seu valor médio. Na sequência, esse valor é normalizado e, então, comparado com um valor limítrofe de forma que, com base nessa comparação, é classificada ou não a ocorrência da FAI.

Ainda, Gao *et al.* (2021) aplicam a *Empirical Wavelet Transform* (EWT) para extrair as componentes de tempo-frequência da *Normalized Faulty Energy*, sendo esta calculada a partir da corrente de sequência negativa. Depois disso, utilizando essas componentes, é obtida a *Entropy Permutation* e, na sequência, a *Permutation Variance* para, assim, detectar a ocorrência da FAI.

Já em Moloi e Davidson (2022), é utilizado o SVM juntamente com a DWT para realizar a detecção da FAI. Além disso, é implementado o método de otimização *Genetic Algorithm* (GA) para selecionar as características e melhorar o desempenho do método de ML.

Em Wontroba *et al.* (2022), é proposto um método para a detecção de FAI com rompimento do cabo. Esse método pode ser dividido em dois blocos, onde o primeiro é responsável pela detecção do rompimento do cabo e o segundo, pela detecção da FAI. Para a realização dessas duas tarefas, o primeiro bloco recebe os sinais de corrente das três fases e a partir deles calcula as componentes de fase, cuja variação é usada como critério para detectar o rompimento do cabo. Já o segundo bloco calcula os fasores dos harmônicos de primeira até a sétima ordem da corrente de neutro e, com base neles, define os demais critérios para realizar a detecção da falta de alta impedância.

Por fim, Guo *et al.* (2003) usam a *Generative Adversal Network* (GAN) para lidar com o desbalanço entre dados que são pertencentes à FAI e os que são relativos à condição normal de operação do sistema. Para isso, a partir do aprendizado da distribuição intrínseca dos dados da falta de alta impedância, são gerados novos casos de modo a balancear as classes presentes para o treinamento e assim obter uma melhor performance do método de *deep learning*. Após isso, usando o conjunto de dados agora balanceado, é treinado uma CNN para detectar os casos de FAI.

1.5 Estrutura do Trabalho

No Capítulo 2, é apresentada a fundamentação teórica a respeito das faltas de alta impedância, o que inclui as suas principais características e como a sua modelagem computacional é feita. Ainda, são explicados conceitos relativos à área de *machine learning* que permitem entender como a detecção de FAIs pode ser abordada como uma tarefa de classificação de padrões. Por fim, é apresentada a teoria de cada um dos algoritmos de ML utilizados nesse Trabalho, incluindo também a técnica de seleção de características empregada.

O Capítulo 3 inicia com o aprofundamento sobre o que é a extração de características e a sua importância para um bom desempenho das metodologias de ML. Além disso, são apresentadas quais foram as características extraídas para esse Trabalho. Na sequência, é detalhado como foi feita a criação do banco de dados e quais os procedimentos realizados de modo que esses dados estivessem prontos para serem usados no treinamento e validação dos modelos de ML. Também, são apresentadas as métricas utilizadas para avaliar o treinamento dos modelos e realizar a sua validação. Outra parte importante desse Capítulo é a aplicação da técnica de seleção de características no conjunto de treinamento, bem como o uso das características selecionadas por esta no aprendizado dos algoritmos de ML. Ainda, é explicado como funciona a metodologia proposta e como foi feita a seleção do critério de detecção utilizado.

O Capítulo 4 apresenta o sistema teste utilizado para simular cada um dos eventos avaliados nesse Trabalho, além de como foi feita a modelagem dos arcos elétricos de FAI. Em relação a isso, são apresentadas as especificidades desse sistema teste e os valores dos parâmetros de simulação de cada um dos eventos. Também, são discutidas as métricas de validação utilizadas para avaliar a operação da metodologia no banco de casos de teste. Após isso, são apresentados os resultados da aplicação da metodologia nos casos de teste sem ruído, iniciando por uma

análise individual do comportamento desta diante de 9 casos analisados em detalhe e depois os resultados gerais para a aplicação em todo o conjunto de teste. Por fim, são apresentados os resultados das métricas de desempenho da metodologia para o banco de teste com diferentes valores de SNR e como o aumento no nível de ruído afeta a performance da metodologia.

Finalmente, no Capítulo 5, são apresentadas as conclusões relativas a esse Trabalho e as sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Considerações Gerais

Para que seja possível entender todos os aspectos relacionados ao uso de técnicas de *machine learning* (ML) para a detecção de faltas de alta impedância, é necessário conhecer quais são as características desse tipo de falta e, caso o objetivo for obter o banco de dados através de simulações, como modelar esse fenômeno computacionalmente. Além disso, após compreender as questões relacionadas à natureza desse evento, é preciso saber como abordá-lo de forma que ele possa ser resolvido pelas ferramentas existentes, nesse caso, como abordar a falta de alta impedância como um problema de classificação de padrões, permitindo assim que técnicas de ML possam ser aplicadas para a sua solução. Outra questão importante, além de conhecer as características da FAI e como abordá-la, é entender o funcionamento dos algoritmos de ML e a teoria por trás de cada um deles, uma vez que cada algoritmo apresenta suas próprias particularidades com vantagens e desvantagens, as quais irão depender das características dos dados analisados. Assim, como cada uma dessas questões estão relacionadas dentro da proposta desse Trabalho, esse Capítulo se concentra na discussão de cada uma delas, apontando aspectos que vão desde a definição e características das faltas de alta impedância até a teoria por trás de cada uma das ferramentas usadas para detectá-la.

2.2 Falta de Alta Impedância

Uma FAI ocorre quando um condutor energizado entra em contato com uma superfície altamente resistiva, resultando em uma corrente de falta com valores insuficientes para sensibilizar os dispositivos convencionais de proteção. Esse contato pode ocorrer com objetos como, por exemplo, uma árvore ou com o próprio solo através do rompimento do cabo ou queda do poste (Lopes *et al.*, 2020; Theron; Pal; Varghese, 2018). Além disso, uma vez que o condutor continua energizado, este ameaça a segurança de pessoas e animais, sendo também capaz de causar incêndios, os quais podem levar a danos econômicos e ambientais. Por esse motivo, é importante desenvolver técnicas que sejam capazes de identificar a ocorrência desse tipo de problema no menor tempo possível.

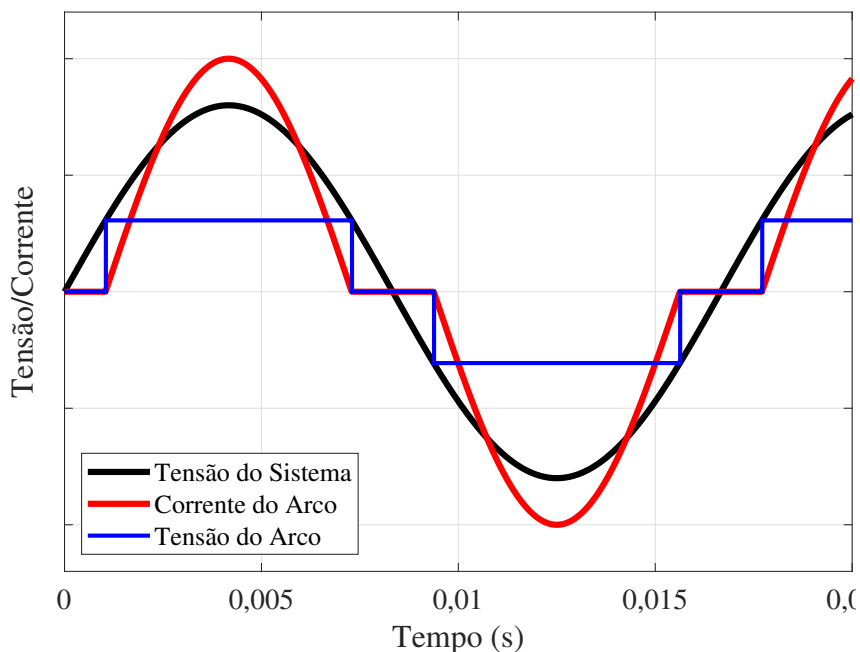
2.2.1 Características da Corrente de FAI

Já que os dispositivos convencionais não identificam essa falta devido ao seus baixos níveis de corrente, é necessário utilizar outras características para conseguir diferenciar estes eventos dos demais que ocorrem no sistema de distribuição. Para isso, recorre-se à assinatura deixada pelo arco elétrico, o qual geralmente está presente durante esse tipo de falta. Dessa forma, pode referir-se a esse tipo de problema como falta de alta impedância com arco elétrico (Mishra; Panigrahi, 2019). Com base nisso, têm-se as seguintes características:

- **Não-linearidade:** a condução do arco começa no momento em que a magnitude da

tensão do sistema se torna maior do que um determinado valor conhecido como tensão de *breakdown* ou tensão de *restrike* e termina quando a tensão do sistema volta a ser menor. Assim, a extinção e reignição do arco elétrico acontecem pelo menos duas vezes a cada ciclo. Esse fenômeno ocorre porque, para que haja a condução de corrente do condutor em relação à superfície que ele está em contato, deve haver primeiro a ruptura do dielétrico. Sendo assim, até que o campo elétrico do sistema seja capaz de vencer essa barreira, não há o surgimento do arco (Ghaderi; III; Mohammadpour, 2017; Zamanan; Sykulski, 2006). Esse comportamento é apresentado na Figura 1, onde a corrente do arco é mostrada juntamente com a tensão do sistema, ambos os valores estão em Por Unidade (p.u.).

Figura 1 – Formato da corrente de arco.



Fonte: adaptado de Zamanan e Sykulski (2006).

- **Assimetria:** outra característica observada na corrente resultante da FAI é que existe uma assimetria entre os seus semiciclos, podendo tanto o semiciclo positivo ser maior que o negativo quanto o contrário (Zamanan; Sykulski, 2006). Nesse sentido, como é comentado em Sultan, Swift e Fedirchuk (1994), em alguns casos, esse fenômeno pode ser resultado de um comportamento retificante do solo, onde as estruturas vítreas que são formadas da sílica aquecida pelo arco atuam como um cátodo emissor de elétrons. Dessa forma, a queda de tensão ao longo do cátodo é menor quando o condutor é positivo. Logo, a tensão do arco, a qual tem que ser superada pela do sistema, torna-se menor para o semiciclo positivo (V_{arc}) do que para o negativo (V_{arc}^-). Assim, uma vez que a tensão de arco é composta pela queda de tensão do ânodo (v_a), pela queda de tensão do arco (v_l) e pela queda de tensão do cátodo, que aqui será denotado como v_k para o semiciclo positivo e v_k^- para o negativo, pode-se usar as Equações 2.1, 2.2 e 2.3.

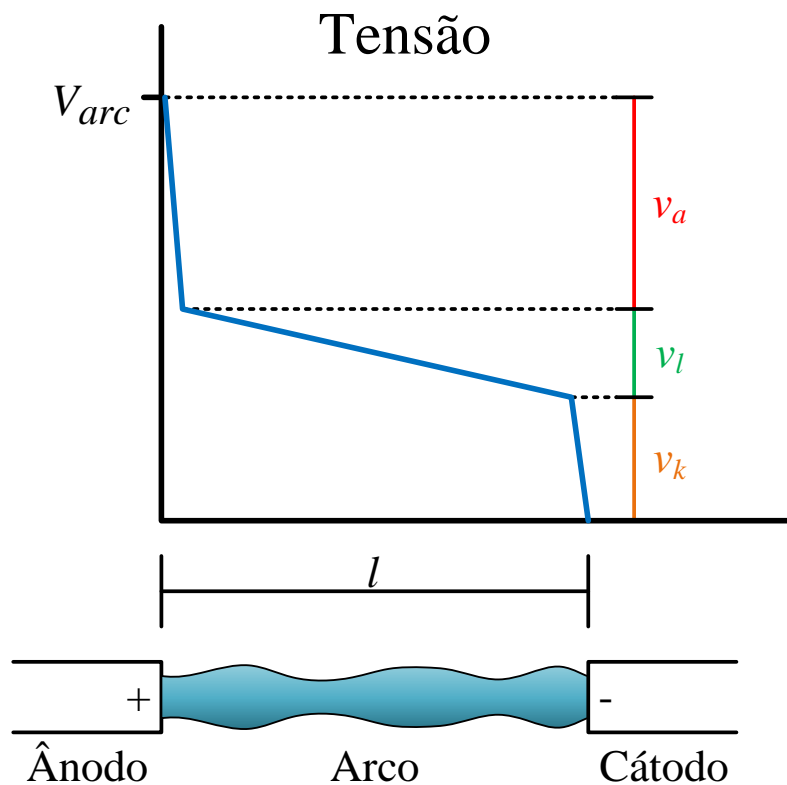
$$V_{arc} = v_a + v_l + v_k \quad (2.1)$$

$$V_{arc}^- = v_a + v_l + v_k^- \quad (2.2)$$

$$v_k < v_k^- \Rightarrow V_{arc} < V_{arc}^- \quad (2.3)$$

A Figura 2 mostra a distribuição dos valores de tensão para o condutor (Ânodo), arco elétrico e solo (Cátodo).

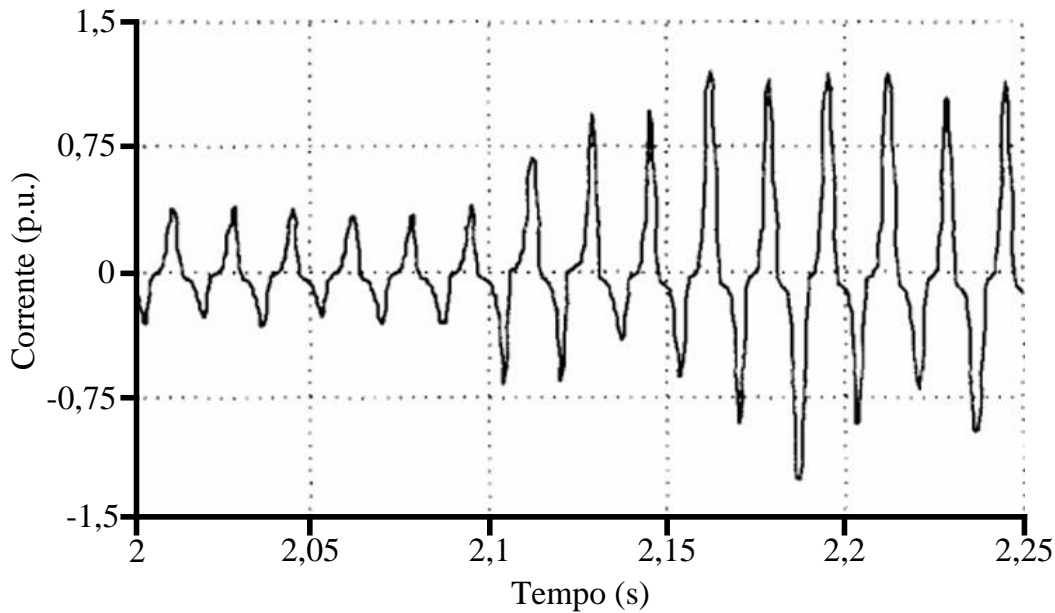
Figura 2 – Distribuição dos valores de tensão na formação do arco elétrico da FAI.



Fonte: adaptado de Sultan, Swift e Fedirchuk (1994).

- **Comportamento Aleatório:** a magnitude da corrente varia de forma aleatória uma vez que o arco rearranja as características do espaço de ar em volta do condutor e também influencia nas transformações que vão ocorrendo na resistência do solo, conforme a sílica vai sendo aquecida e as estruturas vítreas conhecidas como fulguritos vão sendo formadas (Sultan; Swift; Fedirchuk, 1994). A Figura 3 apresenta um sinal de corrente de FAI, onde pode-se notar todas as três características citadas até agora, principalmente o comportamento aleatório.

Figura 3 – Corrente de FAI que apresenta comportamento aleatório, não-linearidade e assimetria.



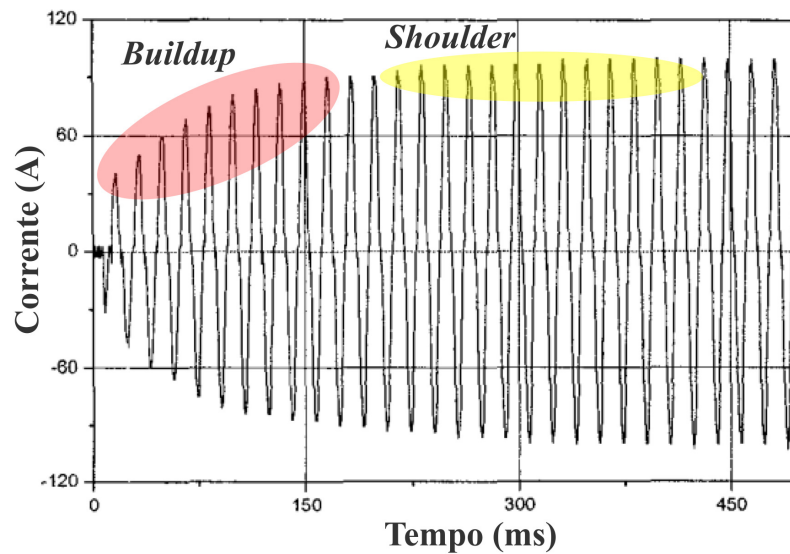
Fonte: adaptado de Sultan, Swift e Fedirchuk (1994).

- **Buildup e Shoulder:** a corrente da FAI apresenta um crescimento gradual de seus valores até um valor máximo, sendo esse processo conhecido como *buildup*. Isto se deve ao contato inicial do condutor com o solo ser pouco efetivo, o que vai mudando conforme o arco elétrico vai penetrando o solo e, assim, aumentando o contato efetivo (Jeerings; Linders, 1989). Além disso, após o término do crescimento dos valores de corrente, o *buildup* é interrompido por alguns ciclos, sendo esse comportamento conhecido como *shoulder* (Nam *et al.*, 2001). Essas duas características são apresentadas na Figura 4.
- **Intermitência:** uma vez que o contato do cabo com o solo é interrompido durante alguns ciclos, o arco elétrico acaba sendo reduzido e restaurado novamente várias vezes (Costa *et al.*, 2015). A Figura 5 apresenta esse comportamento.

2.2.2 Detecção da FAI como uma Tarefa de Classificação de Padrões

Uma vez que a FAI ameaça a integridade física de pessoas e animais, os métodos aplicados para essa falta devem ser capazes de detectar sua ocorrência no maior número de cenários possível, buscando sempre o menor tempo de detecção. Dessa forma, ao longo das décadas foram sendo realizados vários estudos que buscavam entender quais ferramentas e abordagens poderiam alcançar esses objetivos e, assim, evitar que qualquer dano aconteça.

Uma forma de encarar a detecção de faltas de alta impedância é entender a mesma como sendo uma Tarefa de Classificação de Padrões (TCP). Onde uma vez que todo o sistema elétrico apresenta geralmente dois estados, que são o em falta e o normal, pode-se definir esse como

Figura 4 – Exemplo de *Buildup* e *Shoulder*.

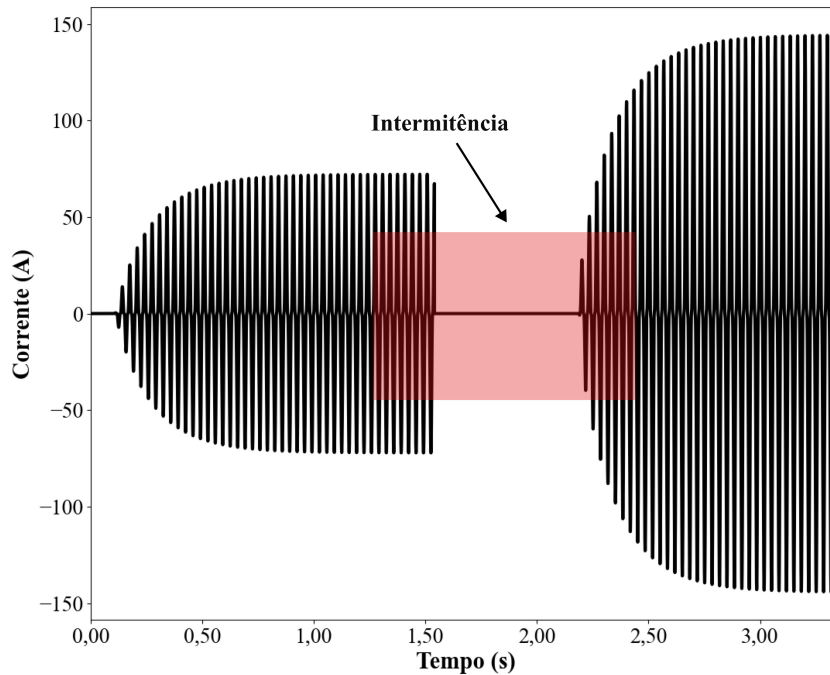
Fonte: adaptado de Nam *et al.* (2001).

um problema de classificação dos estados do sistema. Segundo Ghaderi, III e Mohammadpour (2017), definir o problema dessa forma traz vantagens como:

1. A possibilidade de dividir a tarefa em diferentes partes e lidar com cada uma separadamente, o quê, no contexto de proteção de sistemas elétricos de potência, pode ser definido em três etapas:
 - **Medição:** obtenção dos sinais de corrente e/ou tensão que estão sendo medidos pelos relés através de dados coletados de casos reais ou obtidos a partir de simulações computacionais;
 - **Extração de características:** aplicação de ferramentas de processamento de sinais na medição realizada de modo a obter informações que possibilitem identificar a ocorrência da FAI. Como exemplo, podem-se citar a magnitude da corrente, o ângulo da fase, a energia do sinal, o desvio padrão, a morfologia matemática, o conteúdo harmônico, etc; e
 - **Classificador:** método usado para separar a FAI dos demais eventos do sistema. Por exemplo, a detecção pode ocorrer no momento em que um determinado valor limítrofe é atingido, o que pode ser feito juntamente com um contador, ou fazendo uso de técnicas de *machine learning*.
2. TCP, que fornece um conjunto de métricas que permitem comparar e categorizar as técnicas de detecção.

Dessa forma, o desempenho geral de uma metodologia para a detecção de FAI vai depender: (1) do quão representativos forem os dados dos sinais usados, (2) da capacidade que a

Figura 5 – Exemplo de intermitência em um sinal de FAI.



Fonte: Autor.

extração de características usada tem de evidenciar aquilo que é particular à FAI e (3) do quanto que a lógica de classificação vai conseguir se aproveitar dessa separação resultante da extração.

2.2.3 Modelos de Arco Elétrico

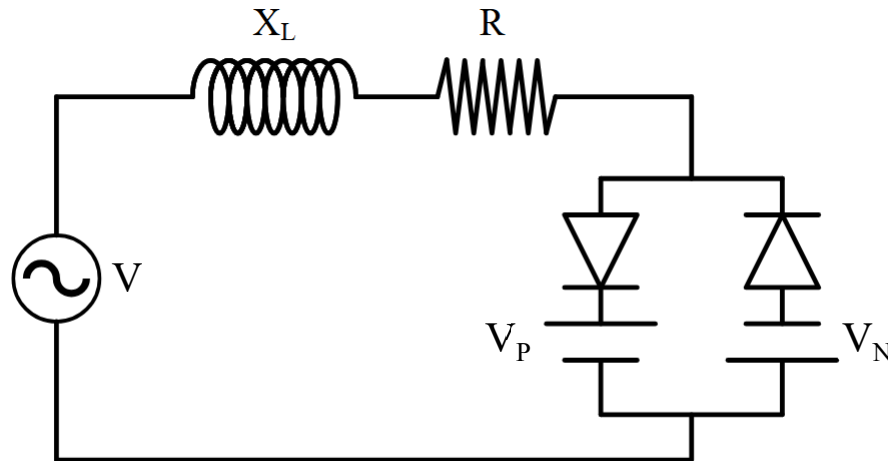
Como foi visto anteriormente, a representatividade dos dados pertencentes aos sinais de corrente e tensão desempenha um importante papel na construção de métodos de detecção que sejam eficazes. Entretanto, devido às limitações de espaço e de recursos, utilizar dados obtidos através de ensaios no mundo real se torna inviável em muitos casos. Nesse sentido, o uso de modelos computacionais para representar o efeito da presença de uma FAI no sistema elétrico se torna extremamente importante. Por esse motivo, ao longo das décadas, vários esforços foram direcionados com o objetivo de criar modelos que fossem capazes de representar da forma mais realista possível as características desse fenômeno.

2.2.3.1 Modelo Emanuel

Emanuel *et al.* (1990) apresenta os resultados obtidos em campo, com o objetivo de estudar o conteúdo harmônico da corrente durante uma FAI. O intuito era demonstrar a confiabilidade de utilizar componentes de baixa frequência para a detecção desse tipo de falta. Além disso, os resultados foram comparados ao modelo de arco proposto, que mostrou respostas compatíveis com as observadas no estudo.

Esse modelo utiliza duas fontes de Corrente Contínua (CC) em antiparalelo, uma para o semiciclo positivo (V_P) e outra para o semiciclo negativo (V_N), cada uma conectada a um diodo. Esses diodos estão em série com uma resistência R e uma reatância X_L , conforme ilustrado na Figura 6.

Figura 6 – Modelo de Arco Emanuel.



Fonte: Adaptado de Emanuel *et al.* (1990).

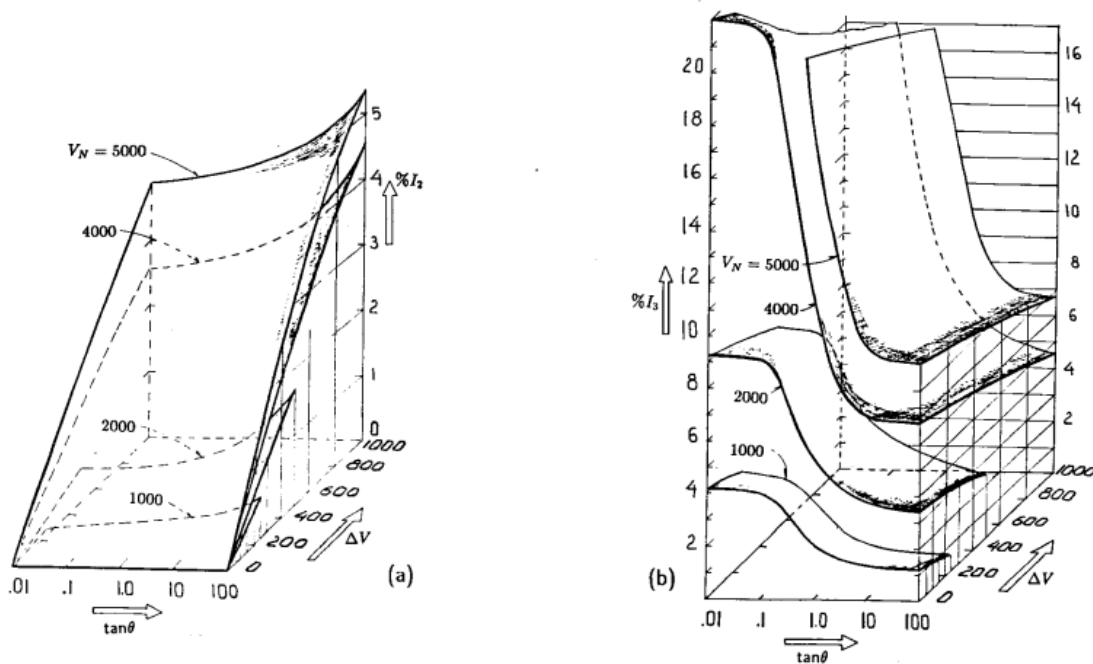
Como os estudos focaram na análise da segunda e terceira harmônicas, os elementos apresentados no circuito do modelo estão diretamente relacionados a essas componentes. A tensão de assimetria (ΔV) que corresponde à diferença entre V_N e V_P , influencia a segunda harmônica. Ainda, definido como a razão entre X_L e R , o parâmetro $\tan \theta$ exerce uma forte influência sobre a terceira harmônica. A Figura 7 ilustra o comportamento da segunda e terceira harmônicas em função da variação de ΔV e $\tan \theta$.

Embora esse modelo represente adequadamente o comportamento das harmônicas de baixa frequência, ele apresenta limitações ao descrever características no domínio do tempo, como o *buildup* e o *shoulder*. Portanto, dependendo do método utilizado para a detecção da falta, o modelo pode não ser suficientemente preciso.

2.2.3.2 Resistências em série e variantes no tempo

Com o objetivo de representar características como *buildup*, *shoulder*, não linearidade e assimetria, Nam *et al.* (2001) utilizaram dados coletados pela *Korea Electric Power Corporation (KEPCO)* para desenvolver um novo modelo de arco. A solução encontrada para descrever a assinatura de uma FAI foi o uso de duas resistências em série, com valores variáveis no tempo, controladas por TACS (*Transient Analysis of Control Systems*). A primeira resistência (R_1) foi atribuída à representação da assimetria e não linearidade, enquanto a segunda (R_2) foi responsável por modelar o *buildup* e o *shoulder*. Essa separação se baseou na compreensão de que o desenvolvimento da falta ocorre em duas etapas: na primeira, a corrente passa por um

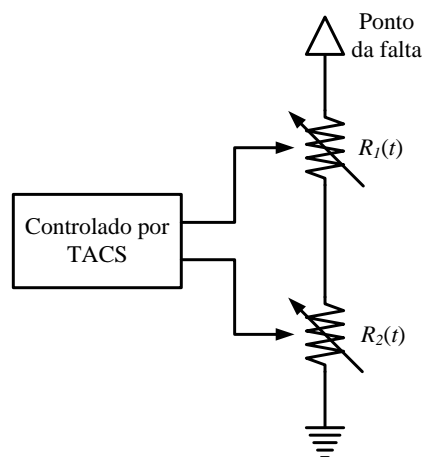
Figura 7 – Exemplo de comportamento da segunda e terceira harmônica em relação a ΔV e $\tan \theta$.



Fonte: Emanuel *et al.* (1990).

regime transitório, crescendo até atingir um valor específico; na segunda, a corrente entra em regime permanente, apresentando comportamento cíclico. A Figura 8 ilustra esse modelo.

Figura 8 – Modelo de arco usando duas resistências variantes no tempo.

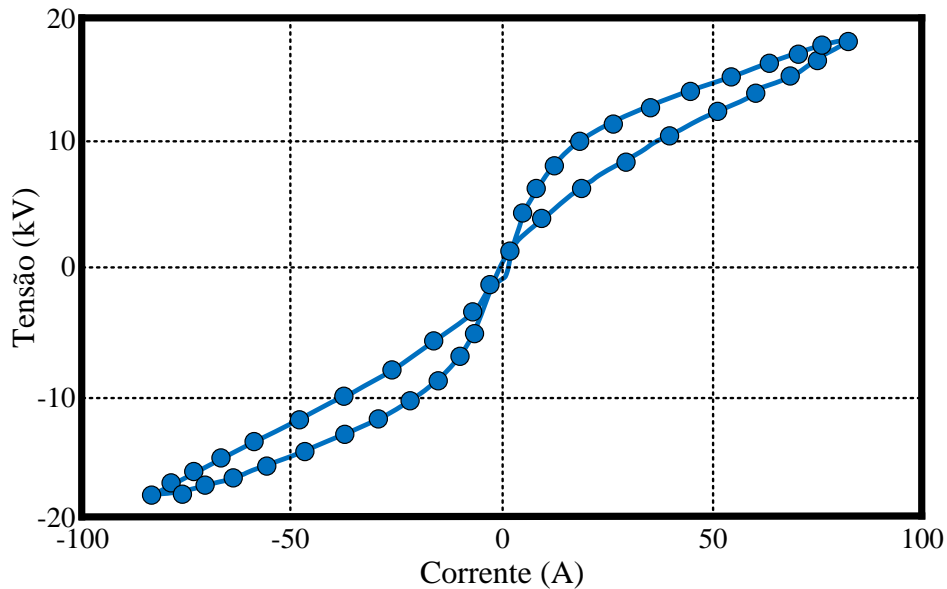


Fonte: Adaptado de Nam *et al.* (2001).

Para modelar a resistência $R_1(t)$, foi realizada a linearização da curva característica tensão-corrente de uma falta de alta impedância, obtida pela KEPCO. A curva tensão-corrente correspondente a um ciclo em regime permanente está ilustrada na Figura 9, enquanto a versão

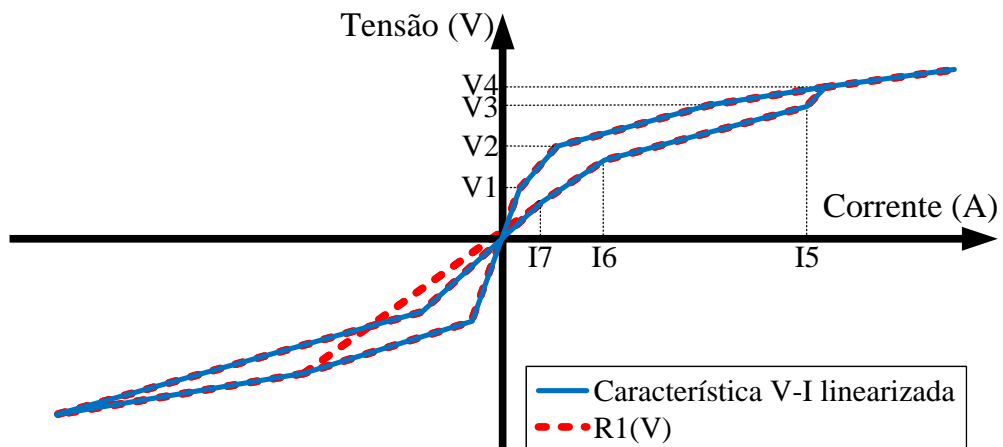
linearizada dessa curva é apresentada na Figura 10.

Figura 9 – Exemplo de curva característica tensão-corrente para uma FAI em regime permanente.



Fonte: Adaptado de Nam *et al.* (2001).

Figura 10 – Exemplo de curva característica tensão-corrente linearizada.



Fonte: Adaptado de Nam *et al.* (2001).

Considerando que a tensão do ramo em falta esteja no intervalo $v_n \leq v(t) \leq v_{n+1}$, onde n representa o n -ésimo valor, o resultado da linearização é apresentado na Equação 2.4 e a resistência resultante é dada pela Equação 2.5.

$$i(t) = i_n + \frac{i_{n+1} - i_n}{v_{n+1} - v_n}, \quad (2.4)$$

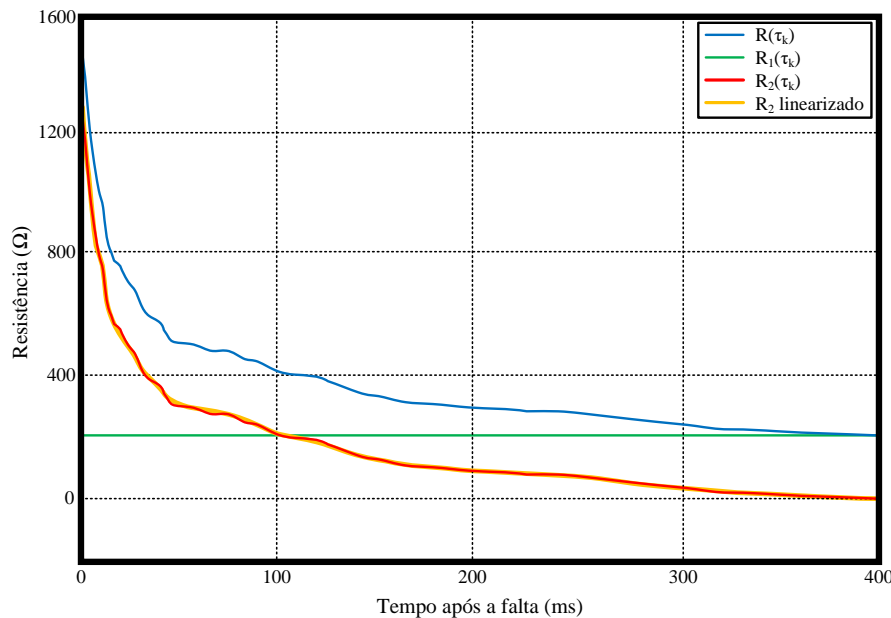
$$R_1(t) = \frac{v(t)}{i(t)} = \frac{v(t)}{i_n + \frac{i_{n+1} - i_n}{v_{n+1} - v_n}}, \quad (2.5)$$

Se a resistência total $R(t)$, que é a soma das duas resistências, já for conhecida, basta subtrair $R_1(t)$ para determinar $R_2(t)$. No entanto, em vez de usar todos os pontos de $R(t)$, realiza-se uma interpolação com os valores máximos de corrente e tensão a cada meio ciclo. Seja τ_k o instante de tempo em que a corrente e a tensão atingem seus valores máximos; então, a equação que descreve a segunda resistência é dada por (2.6).

$$R_2(\tau_k) = R(\tau_k) - R_1(\tau_k) \quad (2.6)$$

A Figura 11 apresenta as curvas interpoladas correspondentes a cada uma das resistências.

Figura 11 – Resistências variantes no tempo nos instantes de máxima tensão e corrente.



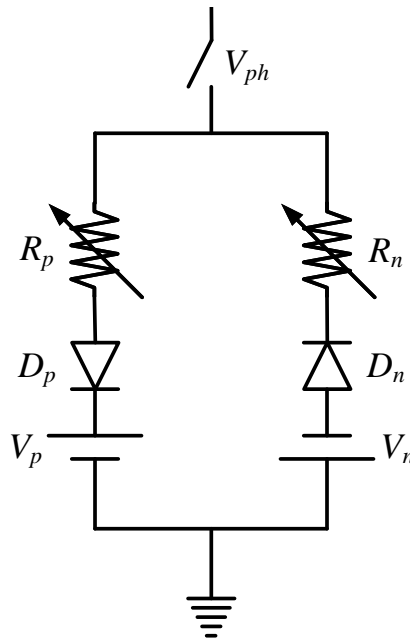
Fonte: Adaptado Nam *et al.* (2001).

2.2.3.3 Modelo apresentado por Gautam

Gautam e Brahma (2012) usaram um modelo de arco com uma estrutura bastante similar com a do arco Emanuel, porém, com diferença no comportamento de suas resistências e de suas fontes de tensão que, agora, é aleatório. Dessa forma, definindo R_p , V_p e D_p como sendo, respectivamente, a resistência, a tensão da fonte CC e o diodo do semiciclo positivo e R_n , V_n e D_n como sendo a resistência, a tensão da fonte CC e o diodo do semiciclo negativo, tem-se o modelo de arco elétrico apresentado na Figura 12. Nessa Figura, V_{ph} é o fasor da tensão de fase do sistema.

As fontes de tensão V_p e V_n apresentam magnitudes diferentes de modo a gerar a característica de assimetria e, assim como as resistências R_p e R_n , seus valores mudam

Figura 12 – Modelo de arco usado por Gautam e Brahma.



Fonte: Adaptado de Gautam e Brahma (2012).

aleatoriamente dentro de uma determinada margem. As variações acontecem a cada 1 ms e são definidas para o sistema IEEE 13 barras com 4,16 kV, segundo Gautam e Brahma (2012), como sendo:

- $V_p = 1$ kV com variação aleatória de $\pm 10\%$;
- $V_n = 0,5$ kV com variação aleatória de $\pm 10\%$;
- $R_p =$ Variação aleatória entre 100Ω - 150Ω ; e
- $R_n =$ Variação aleatória entre 100Ω - 150Ω .

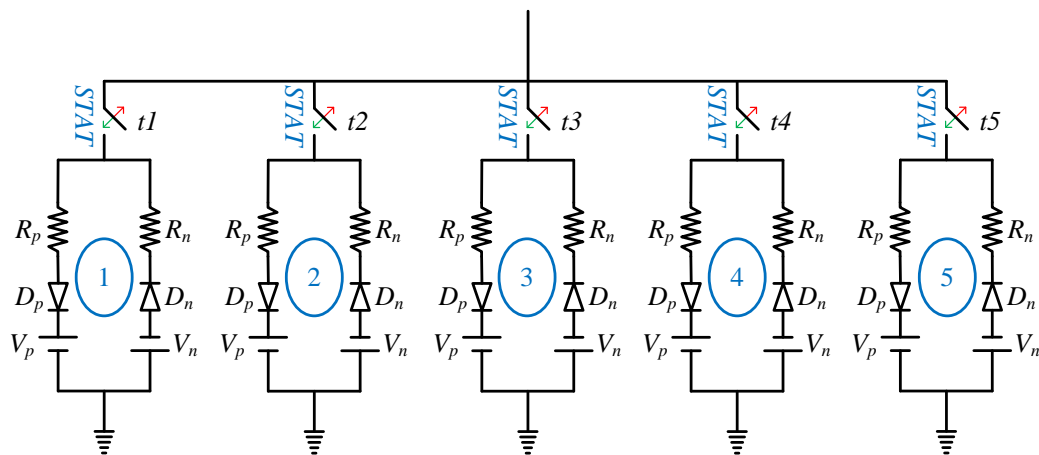
2.2.3.4 Modelo apresentado por Sedighi

Sedighi (2014) utilizou dados de um sistema radial com um alimentador de 20 kV, localizado em Qeshm Island, no Irã. O alimentador tinha uma extensão de 19 km e o ponto onde as faltas foram realizadas estava a aproximadamente 8,5 km de distância do início do alimentador. O sistema era energizado por outro alimentador de 20 kV, conectado por meio de dois transformadores de distribuição (20/0.4 kV, 100 kVA) ligados em *back-to-back*. Os lados de alta tensão estavam conectados em delta, enquanto os lados de baixa tensão estavam em Y. Além disso, os testes foram conduzidos em sete tipos diferentes de solo.

Após a coleta dos dados, foi utilizado o método de Bonferroni, junto com vários modelos de arco de Emanuel em paralelo, como mostrado na Figura 13, para realizar as comparações. O uso de modelos em paralelo se justifica pelo fato de, frequentemente, haver mais de um ponto de contato do cabo rompido com o solo, o que gera múltiplos arcos elétricos que se iniciam e se extinguem aleatoriamente, conforme ilustrado na Figura 14.

Uma das principais constatações dos dados coletados foi que a corrente inicial era em torno de 60% do valor final, levando de 50 a 60 ms para atingir sua magnitude máxima. Com base nisso, o controle de comutação de chaves foi ajustado para que os valores de corrente no modelo simulado se adequassem a essas observações. Além disso, os valores de tensão V_n e V_p foram extraídos da curva v-i, e os demais parâmetros utilizados foram obtidos nos ensaios de falta. Finalmente, a transformada rápida de Fourier foi empregada para extrair todas as componentes de frequência do modelo e o método de Bonferroni foi utilizado para validar os resultados.

Figura 13 – Arcos Emanuel em paralelo.



Fonte: Adaptado de Sedighi (2014).

Figura 14 – Ocorrência de vários arcos em uma superfície de cimento.



Fonte: Sedighi (2014).

2.2.3.5 Modelo de FAI apresentado por Torres

Torres *et al.* (2014) desenvolveram um modelo de arco que simulava as características apresentadas por um arco elétrico ao entrar em contato com o solo a partir da Equação 2.7.

$$\frac{d \ln g}{dt} = \frac{1}{\tau(u, i)} \cdot \left(\frac{ui}{P(u, i)} - 1 \right) \quad (2.7)$$

Onde, u , i e τ são, respectivamente, a tensão, a corrente e a constante de tempo do arco, $g = i/u$ é a condutância instantânea e a energia dissipada é dada por (2.8).

$$P = P_0 + u_0 \cdot |i| \quad (2.8)$$

Sendo G_{ss} a condutância em regime permanente e seu comportamento dado pela Equação 2.9.

$$G_{ss}(i) = \frac{i^2}{P_0 + u_0 \cdot |i|} \quad (2.9)$$

Tem-se que (2.7) pode ser reescrita como (2.10).

$$\frac{dg}{dt} = \frac{G_{ss}(i) - g}{\tau} \quad (2.10)$$

Para que seja possível calcular $G_{ss}(i)$, é considerado que a corrente do arco elétrico tem um comportamento senoidal, de modo que a corrente de regime permanente tenha frequência $2w$. Assim, (2.10) pode ser expandida usando séries de Fourier com harmônicas pares, conforme (2.11).

$$G_{ss}(t) = \sum_{-\infty}^{\infty} G_n e^{j2n\omega t} \quad (2.11)$$

Sendo o período da corrente bem menor do que a constante de tempo do arco, então, pode-se considerar que a condutância de regime permanente é a superposição entre uma constante G_0 e uma componente de rápida oscilação $\delta G_{ss}(t)$, conforme (2.12).

$$G_{ss}(t) = G_0 + \delta G_{ss}(t) \quad (2.12)$$

A componente rapidamente oscilatória pode ser filtrada a partir da operação média realizada em (2.13).

$$G_m(t) = \frac{1}{T} \int_0^T G(t) dt = G_0 \quad (2.13)$$

Considerando que a condutância $g(t)$ pode ser representada a partir da soma de uma componente que não varia significativamente com o período, $g_{lenta}(t)$, e também uma componente de rápida oscilação, $\delta G(t)$, tem-se (2.14).

$$g(t) = g_{lenta}(t) + \delta G(t) \quad (2.14)$$

Aplicando a operação da Equação 2.13 em ambos os lados da Equação 2.10, tem-se (2.15).

$$\frac{dg_{lenta}}{dt} = \frac{G_0 - g_{lenta}}{\tau} \quad (2.15)$$

Resolvendo a equação diferencial em (2.15), tem-se (2.16).

$$g_{lenta} = G_0(1 - e^{-\frac{t}{\tau}}) \quad (2.16)$$

Para obter a resistência média do arco elétrico, basta fazer o inverso de (2.16).

Por fim, após a apresentação desses modelos de arco e a teoria que os fundamenta, é importante ressaltar que alguns destes foram escolhidos para serem empregados nesse Trabalho. Sendo estes os modelos de arco apresentados por Emanuel *et al.* (1990), Gautam e Brahma (2012), Sedighi (2014) e Torres *et al.* (2014).

2.3 Machine Learning

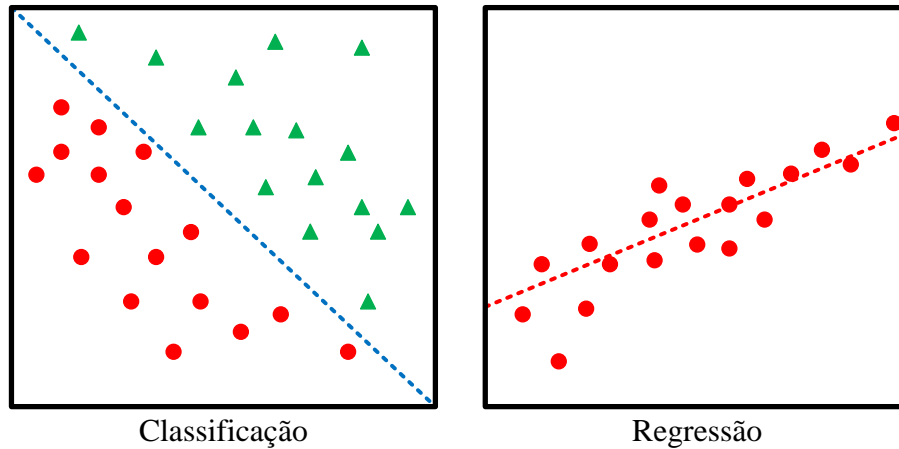
Segundo Sanchez-Martinez *et al.* (2022), *Machine Learning* (ML) é uma sub-área da inteligência artificial (IA) que foca no desenvolvimento de algoritmos os quais possibilitam que computadores possam automaticamente encontrar padrões dentro de um conjunto de dados e se aprimorar com a experiência, sem que seja necessária qualquer instrução explícita. Essa sub-área é composta por várias categorias de aprendizagem, sendo a mais importante para esse Trabalho aquela conhecida como *Supervised Learning* (SL), onde, para cada um dos dados usados, é atribuído um rótulo (*label*) indicando a classe que o mesmo pertence, usando assim essa informação para realizar a fase de treinamento e avaliação do desempenho (Woodman; Mangoni, 2023).

2.3.1 Problemas de Classificação e Regressão no Contexto de *Supervised Learning*

A categoria de SL apresenta duas importantes divisões, as quais podem ser definidas pelo tipo de problema que se busca abordar. A primeira é conhecida como Classificação e lida com problemas que envolve categorizar um determinado dado de forma a analisar se ele pertence ou não a uma determinada classe (classificação binária), ou, dentro de um conjunto de várias classes, a qual delas aquela dado pertence (classificação multiclasse). Já o outro tipo é a Regressão, a qual busca encontrar valores contínuos com base no valor de cada dado (Sarker, 2021). Por exemplo, um problema de classificação seria detectar se um paciente está ou não com um determinado tipo de doença. Já um problema de regressão seria analisar, dado um conjunto de fatores, como que o número de pessoas doentes em uma comunidade variaria em um determinado espaço de tempo. A Figura 15 apresenta esses dois tipos de problema.

Destaca-se ainda que a Figura 15 apresenta como o algoritmo de ML desempenha essas duas tarefas de SL. Para o caso de classificação, a identificação dos dados como sendo ou não pertencentes a uma determinada classe dependerá de como o algoritmo estabelecerá os

Figura 15 – Diferença entre problemas de classificação e regressão.



Fonte: Adaptado de Sarker (2021)

limites que separam essas duas possibilidades. Dessa forma, os dados que ficam abaixo desse limites são definidos como sendo da classe representada pelos pontos e os acima, como sendo da classe representada pelos triângulos. Nesse sentido, para o problema classificação, é tarefa do algoritmo encontrar o melhor posicionamento desses limites, o que ocorre através da fase de treinamento. Entretanto, no caso de regressão, a linha representa a busca do algoritmo por tentar seguir o comportamento estimado da variável independente, dado o valor de alguma variável dependente. Assim, os pontos representam o comportamento real da variável independente e a linha representa a tentativa do algoritmo de tentar prever esse comportamento.

Uma vez que a detecção da falta de alta impedância é um problema de classificação, esse Trabalho foca apenas nesse tipo de SL. Assim, todos os conceitos apresentados nas Seções a seguir são relacionados a esse contexto.

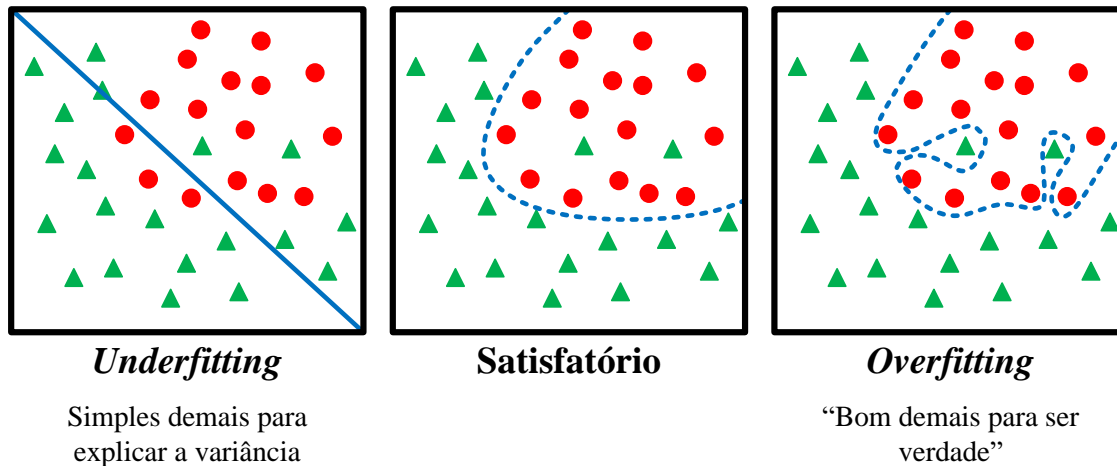
2.3.2 *Overfitting, Underfitting e Cross-Validation*

O processo de aplicação de um algoritmo de ML para uma determinada tarefa de classificação consiste na divisão do conjunto de dados da mesma em duas parcelas, uma de treinamento e outra de teste. Dessa forma, a partir dos dados separados para o treinamento, esse algoritmo encontrará o valor de seus parâmetros que melhor consegue realizar a separação entre as diferentes classes. Com isso, esse algoritmo já na forma de um classificador treinado será aplicado no conjunto de testes e as métricas de seu desempenho serão obtidas. Entretanto, dependendo das características dos dados e dos ajustes usados para o treinamento do algoritmo, podem surgir problemas como *overfitting* e o *underfitting* (Ranganathan; Nakai; Schonbach, 2018).

O *overfitting* ocorre quando o algoritmo aprende tão bem a respeito dos dados de treinamento que o mesmo acaba apresentando um desempenho fraco em qualquer novo dado que for apresentado. Sendo assim, é como se a capacidade de generalização do classificador fosse

afetada por ele ter aprendido até detalhes como as flutuações (ruído) desse conjunto de dados. Já o problema de *underfitting* ocorre quando o conjunto de treinamento é pequeno ou pouco representativo em relação a tarefa que se deseja resolver. Dessa forma, o classificador acaba tendo um desempenho ruim diante de novos dados, por não ter tido uma aprendizagem satisfatória (López; López; Crossa, 2022). Um exemplo desses dois problemas é apresentado na Figura 16.

Figura 16 – *Overfitting* e *Underfitting*.

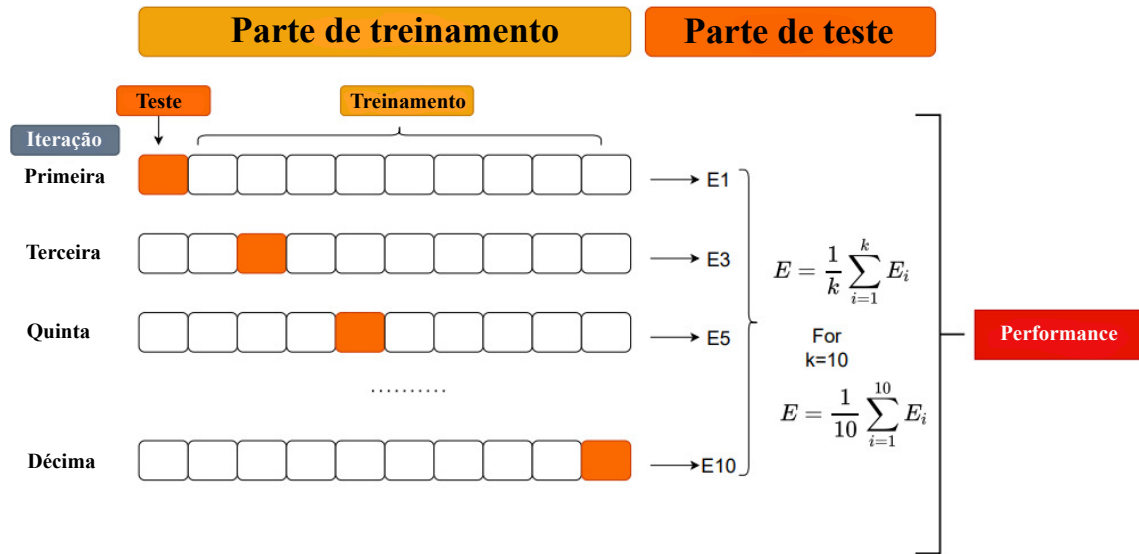


Fonte: Adaptado de Kolluri *et al.* (2020).

Uma boa técnica para detectar problemas de *overfitting* e *underfitting* antes mesmo de colocar o algoritmo em funcionamento é usar uma parte dos dados de treinamento para a validação do classificador. Dessa forma, o algoritmo é treinado e testado no conjunto de validação antes mesmo de ser aplicado nos dados de teste, o que permite identificar qualquer problema antes deste entrar em operação. Nesse sentido, uma técnica que pode ser aplicada para realizar essa tarefa é a *Cross-Validation*, a qual tem como objetivo analisar a capacidade de generalização de um determinado modelo preditivo, nesse caso, o algoritmo de ML (Ranganathan; Nakai; Schonbach, 2018).

Existem vários tipos de *Cross-Validation*, dentre eles, o *k-fold Cross-Validation* consiste em particionar o conjunto de treinamento aleatoriamente em k partes de tamanhos aproximadamente iguais, sendo uma dessas partições usada para realizar a avaliação do desempenho do modelo preditivo e as demais usadas para o treinamento do mesmo. Esse processo é repetido até que cada uma dessas parcelas tenha sido usada para a validação. Por fim, os desempenhos do algoritmo em relação a cada uma das repartições de validação são somados e divididos pelo número de divisões feitas, ou seja, é realizada uma média entre os desempenhos. A Figura 17 apresenta como esse procedimento é realizado.

Além disso, uma vez que o método tradicional realiza as divisões de forma aleatória, diferentes repartições poderiam apresentar proporções desiguais de cada classe, o que poderia levar a repartições pouco representativas em relação à distribuição das classes no conjunto total de treinamento, principalmente, quando há desbalanço entre as classes. Dessa forma, é necessário

Figura 17 – Procedimento do *k-fold Cross-Validation*.

Fonte: Adaptado de Talukder *et al.* (2022).

usar uma técnica conhecida como *Stratified k-fold Cross-Validation*, a qual mantém a proporção entre as classes em cada uma de suas repartições (Prusty; Patnaik; Dash, 2022)).

2.4 Algoritmos de Classificação

Como discutido anteriormente, algoritmos de classificação são responsáveis por identificar a qual grupo um determinado dado é pertencente com base nas delimitações que vão sendo desenvolvidas ao longo de sua fase de treinamento. Assim, após essa fase ter terminado, o que se tem é um modelo preditivo que irá tomar decisões usando parâmetros obtidos, os quais serão responsáveis por construir essas delimitações entre as diferentes classes. Entretanto, como cada conjunto de dados apresenta relações distintas entre os seus atributos (características), diferentes algoritmos irão desenvolver modelos com diferentes níveis de complexidade, onde alguns se sairão melhores ou piores dependendo da capacidade do algoritmo de extrair informações importantes a respeito do conjunto de dados analisado, compondo assim esse modelo preditivo (Sarker, 2021).

Além dessa diferença de desempenho, dependendo do seu nível de complexidade, os modelos apresentarão um grau de interpretabilidade maior ou menor. Dessa forma, modelos mais simples, embora não apresentem um desempenho tão bom para relações mais complexas entre os atributos dos dados, irão ter os seus resultados sendo mais fáceis de interpretar, uma vez que neles é mais claro quais dessas características apresentaram um peso maior para a sua decisão final. Nesse contexto, interpretabilidade é usado no sentido de explicar as decisões do algoritmo de ML em termos que seja possível um fácil entendimento para as pessoas em geral (Czerwinska, 2022).

Na sequência, são apresentados diferentes algoritmos de ML, tendo cada um deles suas vantagens e limitações em relação à cada tipo de situação. Por exemplo, uma árvore de decisão é um algoritmo que apresenta um bom nível de interpretabilidade, uma vez que os critérios utilizados por ela para encontrar os padrões relativos a cada uma das classes podem ser facilmente acessados e organizados, de uma forma a qual possibilita entender a lógica responsável pela árvore tomar as decisões que tomou. Entretanto, para algoritmos de árvores mais complexos, é mais difícil entender quais critérios os levaram às suas conclusões (Ranganathan; Nakai; Schonbach, 2018; Czerwinska, 2022)

2.5 Árvores de Decisão (DT)

Uma árvore de decisão ou, do inglês, *decision tree* (DT) busca separar um conjunto de dados em partes cada vez menores para que, dessa forma, possa solucionar cada uma dessas subdivisões individualmente, seguindo uma abordagem de dividir e conquistar (Ranganathan; Nakai; Schonbach, 2018). Ela realiza essa divisão a partir de uma série de perguntas, agrupando aqueles valores os quais apresentam características semelhantes até o ponto em que seja possível categorizá-los como pertencentes a determinada classe (Timofeev, 2004).

Cada uma dessas perguntas constituem o que é conhecido como “nó”. Sendo assim, o nó responsável pela primeira divisão é chamado de “raiz”, enquanto os “nós internos” são aqueles que realizam as demais divisões ao longo da árvore. Por fim, o nó que indica a escolha final, ou seja, aquele que aponta a qual classe determinado valor pertence é definido como “folha”. Caso os dados cheguem nas folhas da árvore sem estarem completamente separados, então a categoria que apresentar uma maior probabilidade de ocorrência naquele nó folha é a que será usada (Rokach; Maimon, 2005).

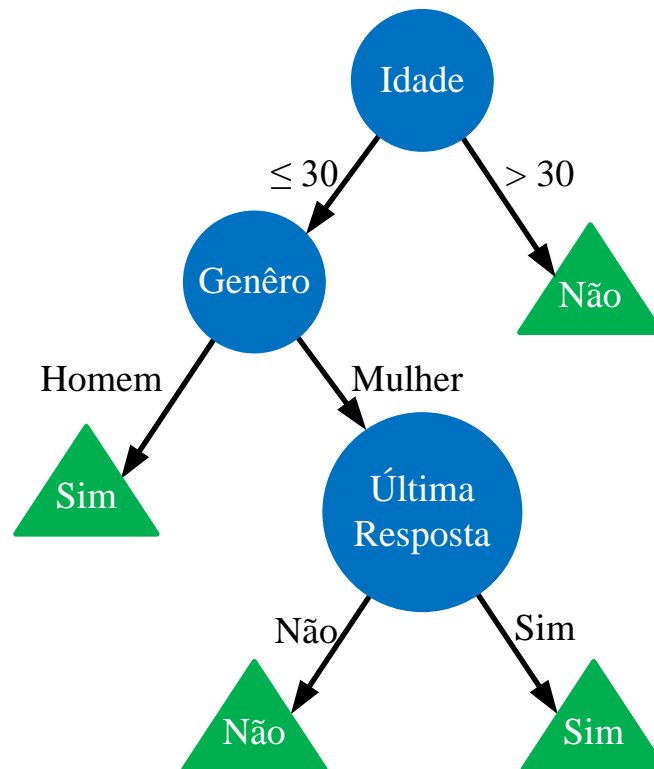
Nesse contexto, a Figura 18 exemplifica o funcionamento de uma árvore de decisão, mostrando o processo de classificação utilizado para identificar se um consumidor em potencial responderia a uma correspondência enviada até ele com base na sua idade, gênero e se respondeu à correspondência anterior. Dessa forma, pode-se notar, por exemplo, que homens com idade menor ou igual a 30 anos sempre respondem.

Existem diferentes algoritmos de árvore de decisão que foram sendo desenvolvidos ao longo dos anos. Dentre eles, destacam-se os algoritmos conhecidos como ID3, C4.5 e CART, os quais se diferem, principalmente, pelo tipo de dados que conseguem comportar, as métricas usadas para realizar a divisão dos dados e pela forma como a árvore é estruturada.

2.5.1 O Algoritmo ID3

Esse algoritmo foi desenvolvido por Quinlan (1986) e usa o ganho de informação (GI) como critério de divisão. Além disso, ele só é capaz de trabalhar com dados categóricos (Hssina *et al.*, 2014), uma vez que não realiza a divisão dos dados com base em intervalos, mas sim em relação a qual elemento aquele determinado dado pertence. Por exemplo, ao construir um nó

Figura 18 – Exemplo de Árvore de Decisão.



Fonte: Adaptado de Rokach e Maimon (2005).

com o atributo Clima, que pode ser classificado como ensolarado, ou nublado, ou chuvoso, o algoritmo poderá apenas dividir os dados conforme eles se enquadrarem em cada uma dessas três possibilidades.

Um conceito que desempenha uma importante função nesse algoritmo é o de entropia, que foi apresentado por Shannon (1948) e possibilita quantificar o quanto as classes estão misturadas em um determinado nó. O valor de entropia de um conjunto de dados S_q é dada pela Equação 2.17.

$$Entropia(S_q) = - \sum_{i=1}^n P_i \log_2(P_i) \quad (2.17)$$

Onde P_i é a fração de objetos pertencentes a classe i presentes no conjunto de dados S_q . Para o caso de classificação binária, uma vez que as classes y são tais que $y = \{0, 1\}$, a entropia pode ser representada por (2.18).

$$Entropia(S_q) = - [P_{pos} \log_2(P_{pos}) + P_{neg} \log_2(p_{neg})] \quad (2.18)$$

Onde P_{pos} indica a fração dos dados de S_q pertencentes a classe analisada e P_{neg} se refere a fração de dados de S_q não pertencentes.

Porém, embora essa métrica demonstre o quanto as classes estão misturadas no conjunto de dados que chega a um determinado nó, ela não indica nada sobre a capacidade que o algoritmo

tem de separar esses dados, de modo a reduzir esse problema da mistura entre classes. Dessa forma, é necessário utilizar uma outra métrica, já mencionada, conhecida como ganho de informação (GI), a qual quantifica o quão bem separadas ficam as classes após passarem por um determinado nó de decisão.

Para isso, é calculada a entropia do conjunto de dados S_q e a entropia dos conjuntos de dados $[S_1, S_2, \dots, S_n]$ os quais são resultantes da divisão de S_q pelo atributo X_q , conforme (2.19).

$$GI(S_q, X_q) = Entropia(S_q) - \sum_{j=1}^n \frac{S_j}{S_q} entropia(S_j) \quad (2.19)$$

Onde S_j representa a parcela do conjunto de dados S_q que foi para o j-ésimo nó filho de L_q , o qual é o nó que reparte o conjunto de dados S_q com base no critério obtido a partir do atributo X_q .

Para a criação de cada nó, analisa-se qual característica (atributo) apresenta um maior ganho de informação. Dessa forma, a que melhor separa as classes contidas naquele conjunto de dados vai sendo escolhida como nó de decisão. Assim, as características que melhor separam os dados tendem a vir primeiro na árvore, enquanto aquelas menos informativas são alocadas no final. Logo, pode-se definir a escolha do nó de decisão a partir de (2.20).

$$X_{escolhido\ para\ S_l} = arg\ max_{X_k} (GI(S_l, X_k)) \quad (2.20)$$

Onde $GI(S_l, X_k)$ é o ganho de informação resultante do uso da k-ésima característica como nó de decisão para o conjunto de dados S_l que chega até ele.

2.5.2 O Algoritmo C4.5

O algoritmo ID3 apresenta certas limitações, dentre as quais destaca-se o fato de que esse método de divisão favorece atributos com muitos valores (Aning; Przybyła-Kasperek, 2022). Dessa forma, Quinlan (2014) propôs um novo critério para a divisão, o qual foi chamado de razão de ganho. Assim, a razão de ganho de um conjunto de dados S_q que é repartido pelo atributo X_q é dada por (2.21).

$$Ganho(S_q, X_q) = \frac{GI(S_q, X_q)}{-\sum_{i=j}^n P_j \log_2(P_j)} \quad (2.21)$$

Assim, a diferença entre o critério de divisão usado no C4.5 em relação àquele utilizado no ID3 reside no fato de que o primeiro normaliza o resultado em termos da quantidade de ramificações que aquele nó terá, reduzindo assim a vantagem que os nós com mais ramos tinham sobre os com menos.

O algoritmo de C4.5 também consegue lidar com valores contínuos, dividindo-os em intervalos e escolhendo aquele que melhor divide as classes. Para isso, quando ele trabalha com um atributo cujo os valores são todos numéricos, para cada valor analisado, ele separa entre os que são menores ou iguais e os que são maiores do que o número em questão. Aquele

que obtiver o maior ganho será o escolhido para representar a separação para aquele atributo (Aning; Przybyła-Kasperek, 2022). Além disso, o algoritmo C4.5 apresenta a capacidade de receber valores faltantes e possui a possibilidade de realizar a podagem da árvore, evitando assim problemas como *overfitting*.

2.5.3 O Algoritmo CART

O algoritmo CART constrói árvores binárias usando um critério conhecido como Índice Gini, o qual se baseia no nível de impureza de um determinado conjunto de dados (Rutkowski *et al.*, 2014) e é representado pela Equação 2.22.

$$Gini(S_q) = 1 - \sum_{i=1}^n P_i^2 \quad (2.22)$$

Uma vez que o algoritmo CART constrói apenas árvores binárias, cada nó terá apenas dois nós filhos conforme (2.23).

$$\Delta Gini(S_q, X_q) = Gini(S_q) - \left[\frac{S_1}{S_q} Gini(S_1) + \frac{S_2}{S_q} Gini(S_2) \right] \quad (2.23)$$

Onde S_1 e S_2 são referentes a parcela dos dados que foram para o primeiro e para o segundo nó filho, respectivamente.

2.6 *K-Nearest Neighbors (KNN)*

O KNN é um algoritmo de *Machine Learning* que armazena os dados de treinamento de modo que, quando usado para classificar um novo dado, é realizada a comparação de proximidade com aqueles que foram armazenados, possibilitando assim identificar a classe do dado atual com base na classe dos vizinhos mais próximos. Nesse sentido, para realizar essa comparação, são então selecionados uma quantidade K de dados mais próximos, sendo a classe do dado analisado igual à classe da maioria dos K dados selecionados (Taunk *et al.*, 2019; Guo *et al.*, 2003; Ranganathan; Nakai; Schonbach, 2018).

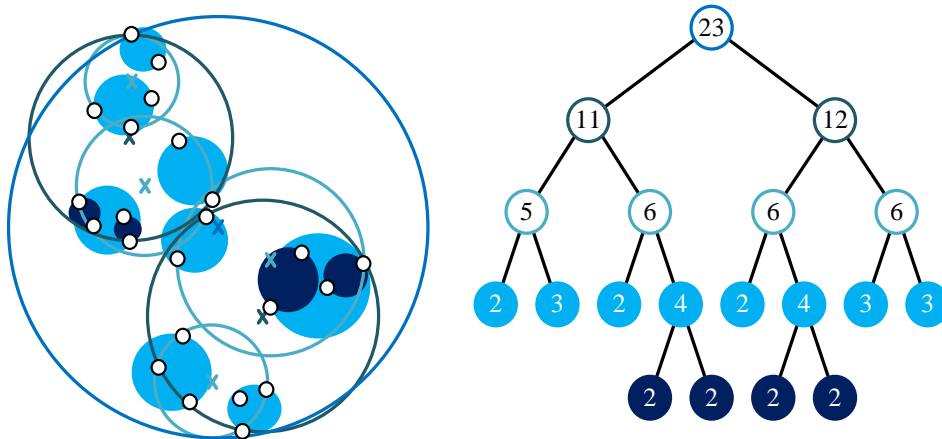
Existem diversas técnicas para realizar a busca pelos dados mais próximos, sendo que essa escolha pode impactar diretamente o tempo de processamento necessário para encontrar os vizinhos próximos. A primeira técnica que pode ser utilizada é realizar a busca por força bruta (BF, do inglês *Brute Force*), o que normalmente é feito calculando a distância euclidiana entre o dado que se deseja classificar e os dados de treinamento. Essa abordagem tem como desvantagem o fato de que se torna ineficiente para grandes conjuntos de dados, e, como vantagem, o fato de que não exige nenhum tipo de pré-processamento (Serrano; Micó; Oncina, 2013).

Além desta, existem outras técnicas que consistem em construir uma estrutura em árvore para facilitar a busca pelos pontos mais próximos, assim diminuindo o processamento e o tempo necessários para encontrar os vizinhos próximos. Uma dessas técnicas é conhecida como *K-Dimensional Tree (KD-tree)*, que é uma estrutura em formato de árvore binária que

recursivamente particiona o espaço de busca de modo a agrupar os dados que estão mais próximos uns dos outros. Assim, esse espaço de buscas particionado simplifica consideravelmente o número de dados que precisam ter suas distâncias analisadas, uma vez que é somente necessário calcular a distância daqueles que estão nos nós mais próximos da árvore (Friedman; Bentley; Finkel, 1977).

Outra abordagem que utiliza uma estrutura em árvore é a *Ball Tree*, a qual divide o espaço em várias hiper-esferas, diferentemente da KD-tree, que divide o espaço de forma paralela aos eixos cartesianos. Dessa forma, cada nó da árvore está relacionado a uma esfera, que, por sua vez, leva a outras esferas menores, como pode ser visto na Figura 19. Esta Figura apresenta a relação do processo descrito com a formação de uma estrutura em forma de árvore (Rajani; McArdle; Dhillon, 2015; Hoyos-Barceló *et al.*, 2017).

Figura 19 – Funcionamento do algoritmo *Ball Tree*.



Fonte: Adaptado de Hoyos-Barceló *et al.* (2017).

2.7 Combinação de Classificadores (Métodos *Ensemble*)

No contexto de ML, o termo *Ensemble* se refere a uma junção de vários modelos em um só, de forma que, ao serem combinadas, as previsões individuais de cada um geram uma previsão única melhorada (González *et al.*, 2020). Essa união pode ser realizada com o mesmo tipo de modelo. Por exemplo, várias árvores de decisão ou com modelos diferentes, por exemplo, combinando *logistic regression*, *support vector machine* e *naive bayes* em um classificador só. Nesse sentido, existem dois tipos de métodos *Ensemble* muito populares: o *Bagging* e o *Boosting* (González *et al.*, 2020), sendo que esses métodos usam modelos homogêneos, ou seja, do mesmo tipo para realizar a previsão.

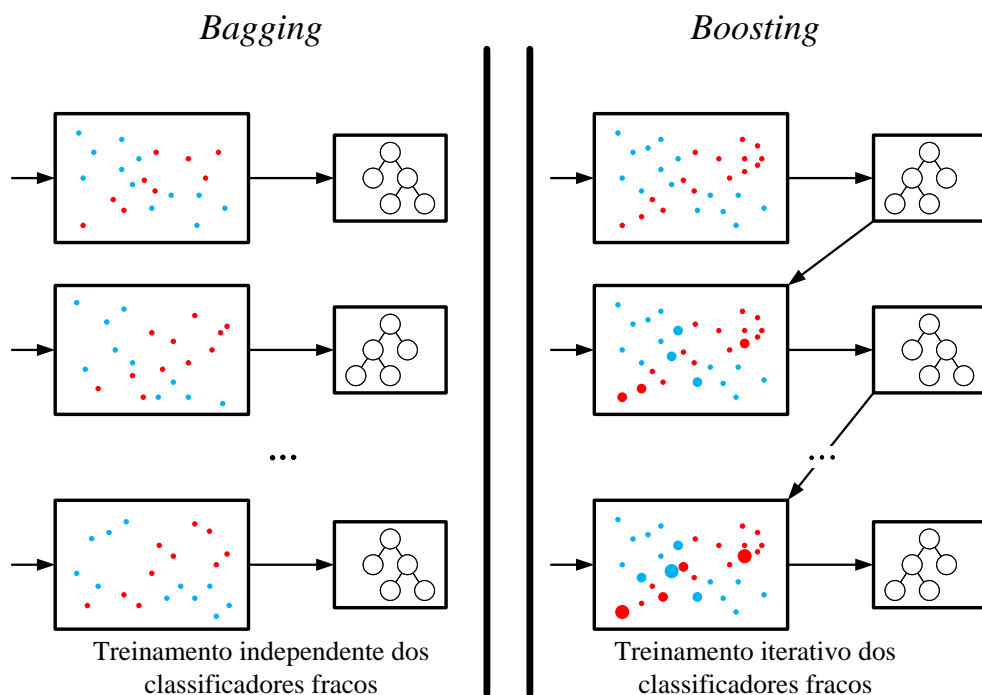
O método conhecido como *Bagging* (*bootstrap aggregation*) é a combinação de vários modelos homogêneos em paralelo, ou seja, o processo de treinamento ocorre individualmente

para cada um e seus resultados são contabilizados de forma a realizar a predição com base no voto da maioria. Além disso, para gerar uma maior diversidade entre cada um dos preditores, cada um deles é treinado com um subconjunto do banco de dados de treinamento usando uma técnica conhecida como *bootstrap* (Yaman; Rattay; Subasi, 2021; Ahmadi *et al.*, 2023). Alguns exemplos de *Bagging* são a *Random Forest* e a *Extremely Randomize Trees*.

Já o *Boosting* é um método o qual treina cada um dos modelos de forma sequencial, onde o treinamento de cada um é realizado atribuindo pesos maiores para os erros dos modelos anteriores (Ahmadi *et al.*, 2023). Alguns exemplos de *Boosting* são o *Adaboost* e o *Gradient Boosting Machine*. A Figura 20 ilustra o funcionamento dessas duas técnicas, onde à esquerda é mostrado o funcionamento do *Bagging* e, à direita, o funcionamento do *Boosting*.

Também é apresentado o conceito de classificador fraco (do original "*weak learner*") que são classificadores com um baixo poder preditivo quando usados individualmente, mas que ao serem usados juntos conseguem ter uma boa capacidade de predição (Freund; Schapire, 1997).

Figura 20 – *Bagging* e *Boosting*.



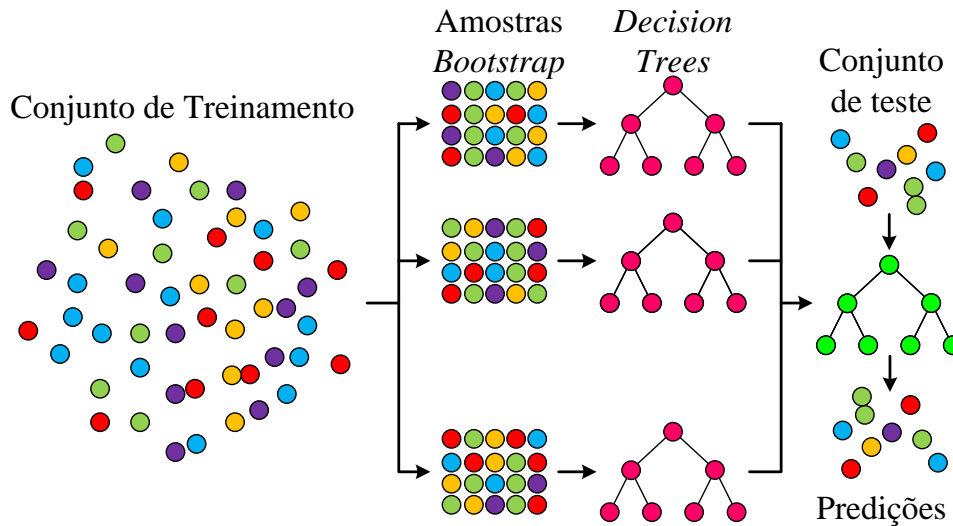
Fonte: Adaptado de Freund e Schapire (1997).

2.7.1 *Random Forest* (RF)

Apresentada por Breiman (2001), a *Random Forest* é um método de *Bagging* composto a partir da combinação de várias árvores de decisão. Esse método realiza a classificação a partir do voto da maioria das árvores, as quais são treinadas a partir de conjuntos formados aleatoriamente dos dados de treinamento. Esses novos *datasets* construídos a partir do conjunto original são conhecidos como dados de treinamento *bootstrap*, onde cada amostra pode ser selecionada mais

de uma vez para formar um mesmo conjunto (amostragem com reposição), conforme a Figura 21 (Ahmadi *et al.*, 2023).

Figura 21 – Treinamento usando amostras *Bootstrap*.



Fonte: Adaptado de Hadjoudj e Pandit (2023).

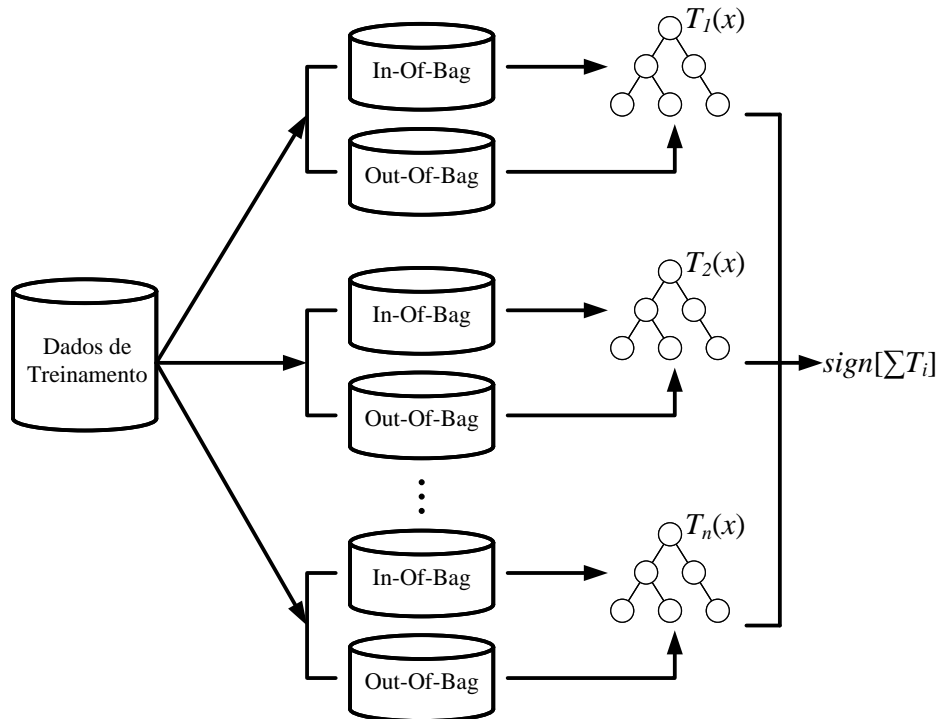
As amostras não escolhidas para treinar as árvores são usadas para a validação. As amostras escolhidas aleatoriamente para formar o subconjunto de treinamento de cada uma das árvores de decisão são conhecidas como amostras *In-of-Bag* e as usadas para a validação são chamadas de amostras *Out-of-Bag* (Wei *et al.*, 2020). A Figura 22 apresenta o funcionamento da *Random Forest*.

2.7.2 *Extremely Randomized Trees* (ET)

Essa variante foi proposta por (Geurts; Ernst; Wehenkel, 2006) e apresenta uma estrutura muito similar à *Random Forest*, exceto pelas seguintes diferenças:

- Ao invés de criar diversos conjuntos alternativos do original através de *bootstrapping* e então treinar cada uma das árvores usando um deles, a *Extremely Randomize Trees* usa o conjunto original para treinar cada uma das árvores;
- A ET não usa um critério para encontrar qual característica (atributo) será usado para separar os dados, essa escolha é feita de forma aleatória (Saeed *et al.*, 2021); e
- Também, é feita aleatoriamente a escolha do ponto de corte dessa característica a qual separará os dados (Acosta *et al.*, 2020).

Os procedimentos listados acima são usados para criar uma diversidade ainda maior nas árvores que formaram o grupo de votação desse preditor.

Figura 22 – Funcionamento da *Random Forest*.

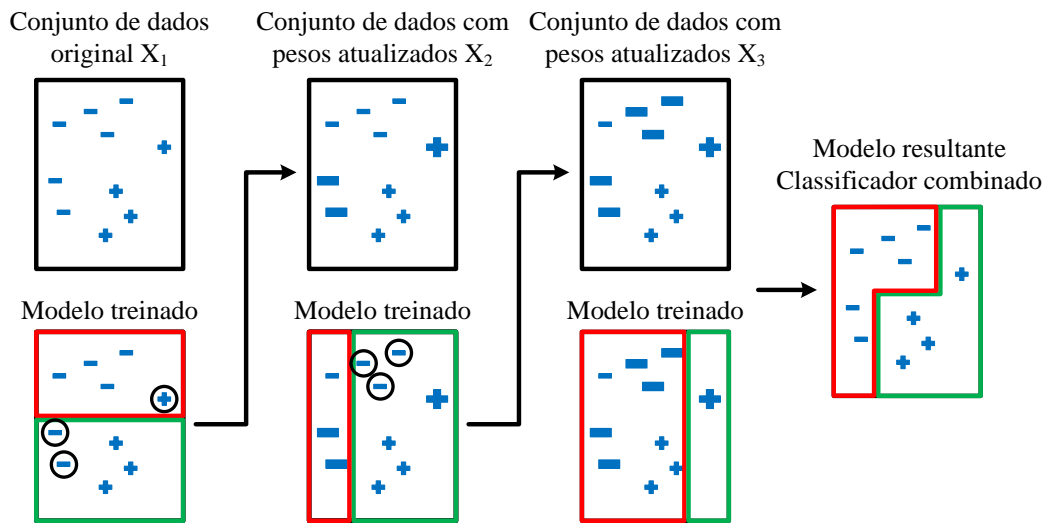
Fonte: Adaptado de Zhang *et al.* (2016).

2.7.3 *Adaboost* (AdaB)

O *Adaboost* ou *adaptive boost* foi proposto por Freund e Schapire (1997) e é um algoritmo composto por vários classificadores em sequência, onde o treinamento ocorre de forma dependente um do outro. Ele inicia treinando um classificador fraco com todo o conjunto de dados e, após isso, atribui pesos que podem ser maiores ou menores dependendo se o dado foi classificado incorretamente ou não. Dessa forma, usando esse conjunto de dados atualizados, é treinada uma cópia desse classificador, o qual irá dar mais importância para as instâncias que o anterior não conseguiu acertar. Esse processo é repetido sucessivamente até que a combinação das previsões de cada classificador resulte em uma boa separação dos dados (Azmi; Baliga, 2020; Wang, 2012). A Figura 23 apresenta as etapas desse algoritmo, onde é representado o processo de atribuição de pesos maiores aos dados classificados incorretamente.

2.7.4 *Gradient Boosting Machine* (GBM)

O *Gradient Boosting Machine* foi apresentado por Friedman (2001) e é um outro método de *boosting*, o qual funciona identificando a limitação dos classificadores fracos a partir de gradientes. Dessa forma, busca-se diminuir o erro do grupo total de preditores ao adicionar novos classificadores iterativamente, reduzindo, assim, a função custo (Singh *et al.*, 2021). O seu objetivo é treinar o novo classificador de forma que o mesmo fique o mais correlacionado possível com o gradiente negativo da função custo relativa a todo o grupo de preditores (Natekin; Knoll, 2013). Esse algoritmo apresenta algumas implementações bem populares, as quais variam

Figura 23 – Funcionamento do *Adaboost*.

Fonte: Adaptado de Wang (2012).

do algoritmo tradicional em termos de performance, do processo de construção das árvores e da busca pela melhor divisão dos dados (González *et al.*, 2020; Ma *et al.*, 2020). Essas variantes são:

- *Extreme Gradient Boosting Machine* (XGBM) proposto em Chen e Guestrin (2016);
- *Light Gradient Boosting Machine* (LGBM) proposto em Ke *et al.* (2017);
- *Categorical Gradient Boosting Machine* (CGBM) proposto em Prokhorenkova *et al.* (2018).

2.8 Métodos *Wrappers* para Seleção de Características

Os métodos *Wrappers* utilizam os algoritmos de ML para avaliar o impacto das características na performance dos mesmos. A vantagem desses modelos é que, ao invés de analisar separadamente cada característica, eles possibilitam analisar a interação das características entre si e como isso afeta o modelo.

Um exemplo de métodos *Wrappers* são aqueles baseados em árvores, os quais escolhem as características com base no quanto elas são capazes de reduzir a impureza, ou seja, separar diferentes classes ao longo de uma árvore de decisão. Além disso, ao grau de contribuição de cada característica para a separação dos dados dá-se o nome de importância de característica, onde, a partir dessa métrica, torna-se possível ranquear quais características foram mais decisivas para o modelo (AlSagri; Ykhlef, 2020).

Um fator interessante descrito em Kohavi e John (1997), é que algoritmos como os de árvores de decisão tendem a ter um decréscimo em sua performance quando alimentados com características que não são necessárias para prever a saída desejada.

Além disso, pode-se usar a importância das características juntamente com outras técnicas como o *cross-validation*, justamente para avaliar a melhora ou não do desempenho

do modelo ao ter um acréscimo ou uma diminuição na quantidade de características usadas. Por exemplo, a técnica conhecida como *Recursive Feature Elimination with Cross-Validation* (RFECV) remove um determinado número de características a cada iteração e vai obtendo os resultados da validação cruzada para esse novo modelo. A quantidade de características que apresentam um melhor desempenho nas métricas escolhidas é aquela que será indicada pela técnica (Wang *et al.*, 2020).

Sobre as técnicas de eliminação recursiva de características, ressalta-se:

- *Recursive feature selection*: envolve apenas remover as características conforme a sua importância para o algoritmo até um estipulado número; e
- *Recursive feature selection with cross-validation*: remove as características conforme a sua importância para o algoritmo, mas também testa a performance do modelo com esse novo conjunto de características, usando validação cruzada.

Além da importância das características, também pode-se usar os coeficientes dos métodos lineares para se obter uma noção de quais são mais importantes para a sua tomada de decisão. Aquelas características que forem multiplicadas por coeficientes maiores apresentam uma contribuição mais significativa para o modelo. É importante ressaltar que esse Trabalho foi realizado com foco no cálculo da importância das características usando árvores de decisão.

2.8.1 Seleção Baseada em Árvores

Uma forma de dividir os dados em uma árvore de decisão é buscar a característica cuja divisão apresente a maior redução na impureza dos conjuntos de dados, ou seja, aquela que torna os conjuntos de dados resultantes da divisão mais homogêneos.

Para medir a quantidade de impureza de um nó da árvore, usa-se o índice Gini, o qual retorna um valor entre 0 e 1, onde um maior valor indica uma maior impureza. Para calcular o índice Gini é usada a Equação 2.22. Entretanto, como no contexto de detecção de falta de alta impedância deseja-se apenas saber se o dado é ou não pertencente a um caso de FAI, então existem apenas duas possibilidades, uma positiva e outra negativa, o que leva a (2.24).

$$Gini = 1 - (P_{pos}^2 + P_{neg}^2) \quad (2.24)$$

Para quantificar a importância de um nó, deve-se calcular o índice Gini para ele e para os seus dois nós subsequentes e, então, fazer a diferença entre eles, obtendo-se assim a redução de impureza resultante dessa ramificação. Esse processo é descrito pela Equação 2.25.

$$C_j = Gini_j - \left(\frac{n_{esq}}{n_j} \cdot Gini_{esq} + \frac{n_{dir}}{n_j} \cdot Gini_{dir} \right) \quad (2.25)$$

Onde C_j é a importância do j -ésimo nó, também chamado de nó pai, n_j se refere ao número de elementos presentes nele e n_{esq} e n_{dir} são o número de elementos presentes nos nós filhos da direita e da esquerda, respectivamente (Zhang *et al.*, 2016).

Concluída essa etapa, para obter a importância relativa à cada característica, basta somar cada um dos nós que usa ela como critério de divisão e, então, dividi-los pelo número total de nós da árvore (AlSagri; Ykhlef, 2020), como apresentado em (2.26).

$$F_i = \frac{\sum_{j \in F_i} C_j}{\sum_{k \in \text{allnodes}} C_k} \quad (2.26)$$

Por fim, o valor normalizado da importância de cada uma das características é obtido por (2.27).

$$\text{Normalized } F_i = \frac{F_i}{\sum_k F_k} \quad (2.27)$$

A partir dessa métrica é possível quantificar a contribuição de cada característica na decisão do algoritmo.

2.9 Considerações Finais

Neste Capítulo, foi apresentada a base teórica sobre a falta de alta impedância, incluindo suas características e os modelos utilizados para sua representação computacional. Além disso, foram discutidos conceitos relativos à aplicação de modelos de machine learning (ML) na tarefa de classificação, que será o foco deste Trabalho para a detecção de FAIs. Foram também introduzidos diferentes algoritmos de ML, cada um com suas particularidades, sendo a maioria dos classificadores utilizados neste estudo baseada em árvores de decisão, com exceção do KNN, o qual se assemelha aos demais pelo fato de também ser um algoritmo não-linear.

3 METODOLOGIA PROPOSTA

3.1 Considerações Gerais

No presente Capítulo, é realizada uma discussão acerca de todos os aspectos envolvidos na implementação da metodologia proposta, a qual tem como base o uso dos modelos de ML. Nesse sentido, primeiramente são apresentados os conceitos relativos ao processo de extração de características, sendo essa uma etapa que desempenha um papel fundamental no bom funcionamento dos classificadores, já que características pouco informativas não possibilitam a correta separação das classes, diminuindo, assim, a performance dos modelos.

Na sequência, é abordada a construção do banco de dados usado nesse Trabalho, o que envolve o processo de simulação dos casos, sua separação em diferentes conjuntos e o seu balanceamento. Além disso, são apresentadas as métricas utilizadas para avaliar os classificadores usados em relação à sua capacidade de corretamente classificar os dados, sejam estes referentes à FAI ou aos demais eventos.

Outra questão discutida nesse Capítulo é a etapa de seleção de características, a qual possibilita realizar interpretações importantes acerca do papel das características inicialmente propostas para o desempenho dos algoritmos e quais destas são realmente importantes para abordar o problema estudado. Ainda, são apresentados os algoritmos que serão analisados nesse Trabalho, juntamente com os dados usados para o treinamento e validação dos mesmos, sendo que cada uma dessas duas etapas é discutida detalhadamente logo em seguida. Por fim, é apresentada a metodologia proposta para a detecção de FAI e a sua operação no tempo.

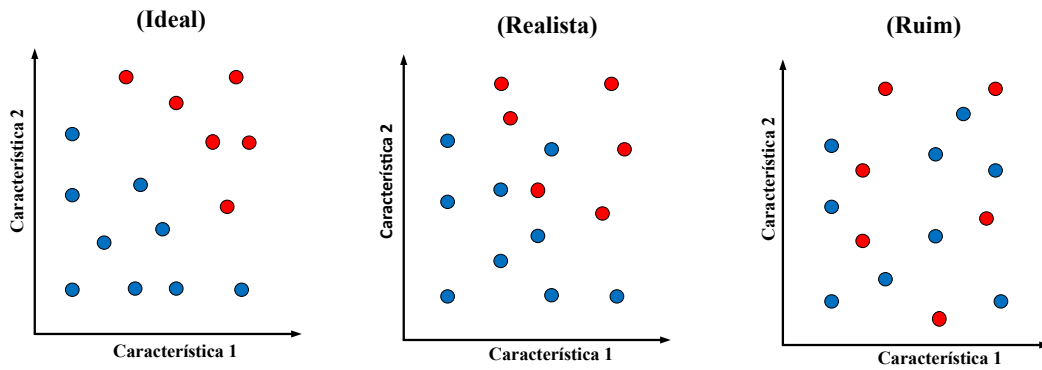
3.2 Extração de Características

A extração de características permite evidenciar detalhes importantes presentes em um sinal, possibilitando, assim, caracterizar diferentes problemas a partir dele. Essa é uma etapa importante no processo de aplicação de ML, uma vez que características mais informativas permitem uma melhor capacidade de classificação dos algoritmos.

Em um contexto de falta de alta impedância, as características extraídas devem possibilitar diferenciar uma situação de FAI dos demais eventos no sistema. Para isso, os espaços criados por essas características devem apresentar a melhor separação possível entre os dados referentes a esse problema, em relação àqueles obtidos para as demais situações, como mostra a Figura 24. Nessa Figura, pode ser visto que os espaços formados pelas características apresentam diferentes graus de separação para uma situação de falta, em vermelho, quando comparado com o que ocorre para os demais eventos do sistema, em azul.

O objetivo dos algoritmos de ML é ajustar os seus parâmetros de forma que eles separem as classes da melhor forma possível, delimitando regiões dentro desse espaço, através do seu respectivo treinamento. Desse modo, durante a sua fase de operação, os dados que adentrarem na região do espaço definida como sendo pertencente a uma determinada classe, serão classificados como pertencentes a esta classe.

Figura 24 – Exemplo de um espaço de características.



Fonte: Adaptado de Grigg (2019).

A Figura 25 exemplifica essas afirmações apresentando as fronteiras de decisão obtidas por nove algoritmos de ML para três espaços de características, onde cada espaço apresenta uma disposição diferentes entre os dados de cada classe.

Assim, é possível perceber que existe uma relação fundamental entre buscar características que melhor separem as situações de FAI dos demais eventos e a aplicação de métodos de ML, os quais sejam capazes de aprender os limites envolvendo essa separação.

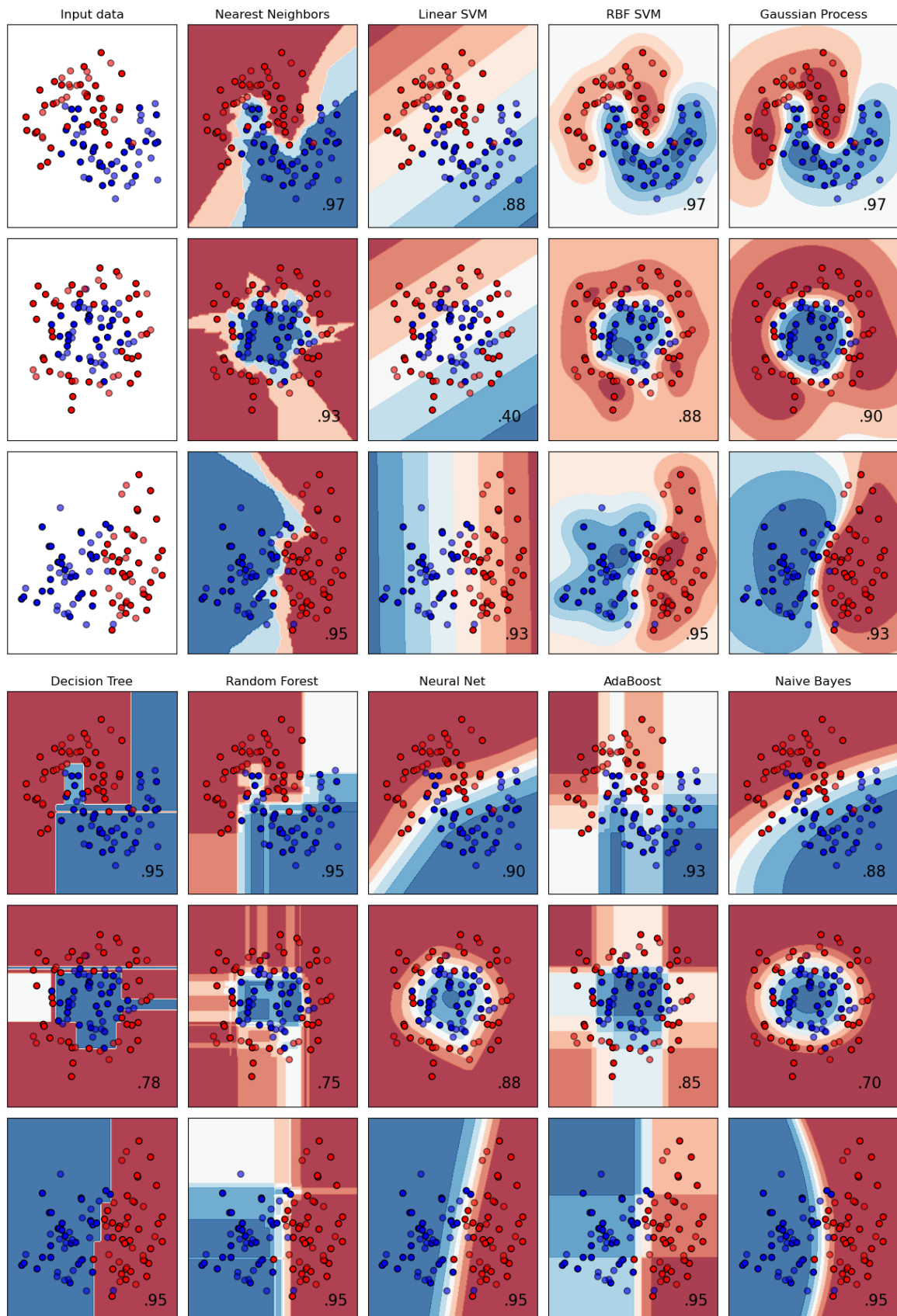
Com base em tudo isso que foi discutido, nessa Seção, são apresentadas as 6 características que tiveram o seu impacto na detecção de FAI analisado. Todas são extraídas do sinal de corrente obtido na subestação, sendo elas:

- Envoltória superior;
- Envoltória inferior;
- Área da envoltória;
- Média;
- Curtose; e
- Assimetria.

3.2.1 Envoltória Superior e Inferior

A Equação 3.1 mostra a janela móvel W_1 , a qual tem o tamanho de um ciclo e que é responsável por armazenar os valores da corrente de uma das fases i vista da subestação.

Figura 25 – Fronteiras de decisão definidas por diferentes algoritmos.



Fonte: Scikit-learn (2024).

$$W_1(k) = [i(k - N + 1) \dots i(k)] \quad (3.1)$$

sendo k a amostra atual e N o número de amostras por ciclo.

Assim, tem-se que as envoltórias superior e inferior podem ser obtidas a partir das Equações 3.2 e 3.3.

$$Es(k) = \max\{W_1(k)\} \quad (3.2)$$

$$Ei(k) = \min\{W_1(k)\} \quad (3.3)$$

Além disso, é realizada a normalização dessas características a partir de seu valor em regime permanente. Em (3.4) e (3.5), são apresentados esses cálculos para a envoltória superior e inferior, respectivamente.

$$Es_{norm}(k) = Es(k)/Es_{regime} \quad (3.4)$$

$$Ei_{norm}(k) = Ei(k)/Ei_{regime} \quad (3.5)$$

Onde Es_{regime} é uma amostra da envoltória superior no instante imediatamente anterior ao início da falta e Ei_{regime} é uma amostra da envoltória inferior também obtida no mesmo referido instante.

3.2.2 Área da Envoltória Superior

Outra característica utilizada é a área da envoltória superior. Para isso, deve-se antes armazenar os valores dessa envoltória em uma segunda janela móvel W_2 , conforme (3.6), a qual também tem tamanho de um ciclo.

$$W_2(k, n) = [Es(k - N + 1, n) \dots Es(k, n)] \quad (3.6)$$

Onde n representa a posição do valor na janela k . Sendo assim, armazenados esses valores, é então usada regra do trapézio para a realização do cálculo dessa área como apresentado em (3.7).

$$A(k) = \Delta t_s \sum_{n=1}^N \frac{W_2(k, n) + W_2(k, n - 1)}{2} \quad (3.7)$$

Onde Δt_s é o intervalo de amostragem.

Da mesma forma que nas características anteriores, a área da envoltória superior também é normalizada a partir de seu valor no instante imediatamente anterior ao início da falta (A_{regime}), como mostrado na Equação 3.8.

$$A_{norm}(k) = A(k)/A_{regime} \quad (3.8)$$

3.2.3 Média

Assim como a envoltória positiva e negativa, a média pode ser obtida a partir da janela W_1 usando (3.9).

$$m(k) = \frac{1}{N} \sum_{n=0}^N W_1(k, n) \quad (3.9)$$

A sua normalização ocorre de acordo com a Equação 3.10.

$$m_{norm}(k) = m(k) / E s_{regime} \quad (3.10)$$

Como a média em regime permanente é zero ou um valor muito próximo disso, optou-se por normalizar a média utilizando o valor da envoltória superior no instante imediatamente anterior ao início da falta, uma vez que isso permite com que os valores fiquem em um intervalo aceitável sem que haja uma divisão por zero.

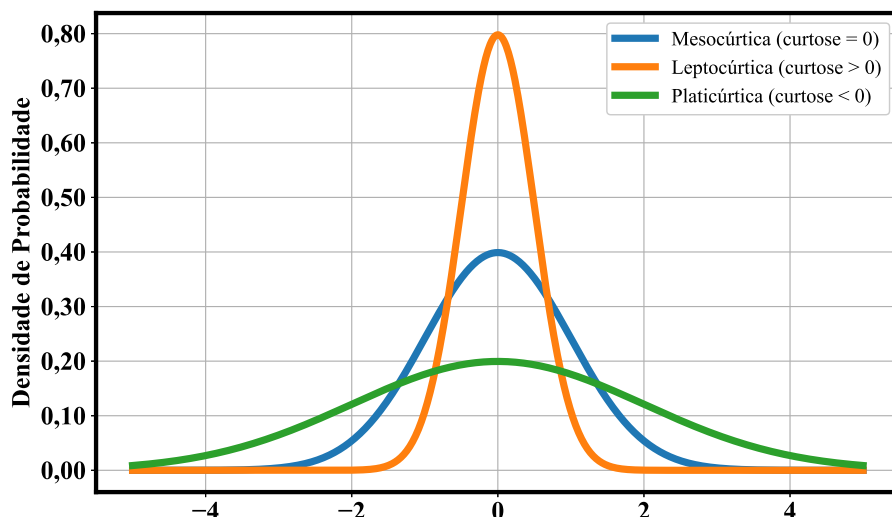
3.2.4 Curtose

Curtose representa o quanto os dados de uma determinada variável se concentram ao longo do pico e das caudas de sua distribuição. Nesse sentido, existem três diferentes formas de curtose, sendo estas:

- Mesocúrtica (Curt = 0): distribuição Normal;
- Leptocúrtica (Curt > 0): Pico mais fino e caudas mais pesadas;
- Platicúrtica (Curt < 0): Pico mais achato e caudas mais leves.

A Figura 26 apresenta três diferentes distribuições, onde cada uma delas se encaixa em uma desses três tipos de curtose.

Figura 26 – Diferentes formas de curtose.



O cálculo da curtose para a janela de dados W_1 é dado por (3.11), onde $\sigma(k)$ é o desvio padrão dos valores da janela $W_1(k)$.

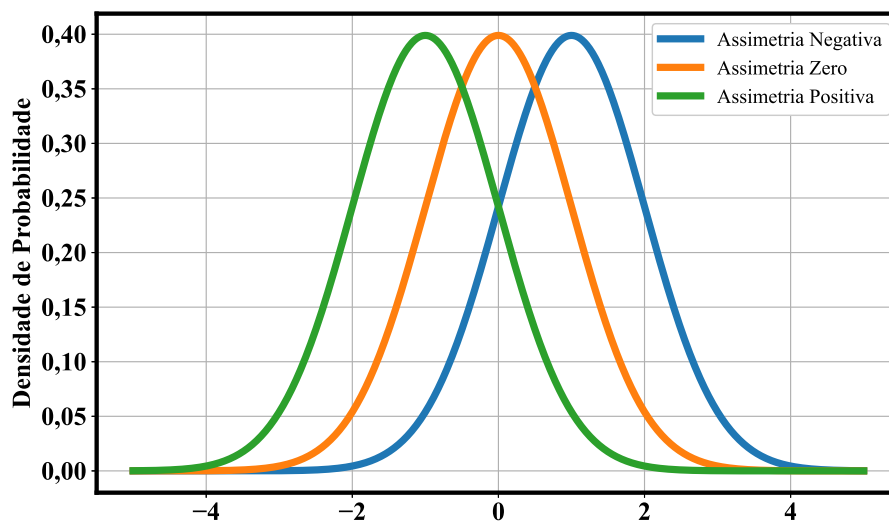
$$Curt(k) = \frac{\sum_{n=0}^N (W_1(k, n) - m(k))^4 / N}{\sigma(k)^4} - 3. \quad (3.11)$$

Como os valores da curtose tendem a não terem uma escala tão diferentes dos valores normalizados das demais grandezas, ele foi mantido sem nenhuma normalização.

3.2.5 Assimetria Amostral

Mede a assimetria da distribuição de uma variável. Nesse sentido, quanto mais próxima essa métrica é de zero, mais próxima a distribuição analisada é da normal. Além disso, a assimetria pode assumir tanto valores positivos quanto negativos, sendo o primeiro um indicativo de que a distribuição tem uma cauda maior à direita do que à esquerda e o segundo representando o oposto. A Figura 27 apresenta os diferentes tipos de assimetria.

Figura 27 – Diferentes tipos de assimetria.



Fonte: Autor.

Seu cálculo é realizado a partir da Equação 3.12.

$$Assm(k) = \frac{\sqrt{N \cdot (N - 1)}}{N - 2} \frac{\sum_{n=0}^N (W_1(k, n) - m(k))^3 / N}{\sigma(k)^3} \quad (3.12)$$

Para o caso da assimetria, a necessidade de normalização foi descartada pelo mesmo motivo apontado para a curtose.

3.3 Banco de Dados

O desempenho dos modelos de ML está diretamente relacionado com os dados utilizados para a realização de seu treinamento, validação e teste. Logo, para o desenvolvimento de uma

metodologia baseada em ML que apresente um comportamento satisfatório para uma determinada tarefa, é necessário que os dados utilizados de fato representem as mais diversas situações daquele problema. Nesse sentido, no contexto de aplicação de algoritmos de ML para a detecção de FAI, mais especificamente para dados obtidos a partir de modelos computacionais, é necessário que sejam realizadas diversas simulações, buscando, assim, representar várias situações distintas que podem ocorrer no mundo real. Com base nisso, para que fosse possível ter um conjunto significativo de simulações, foram criadas várias etapas de modo a gerar diferentes casos e, além disso, rotular os dados contidos nos mesmos, o que foi feito de acordo com a classe e o evento dos quais eles são pertencentes.

3.3.1 Etapas de Criação do Banco de Dados

A primeira etapa consiste na criação de vários arquivos atp a partir de um arquivo original de mesmo formato, o qual vai tendo o seu código copiado, porém com alguns parâmetros modificados, de modo a gerar novas simulações relativas ao evento que está sendo analisado. Para isso, é necessário que esses parâmetros, os quais podem ser, por exemplo, uma resistência, uma tensão ou o tempo de fechamento de chave, sejam declarados no código como sendo variáveis, o que permite que eles sejam facilmente localizados dentro do arquivo atp e, então, alterados. Esse acesso ao arquivo atp pode ser realizado por qualquer linguagem de programação que seja capaz de alterar um arquivo de texto, no caso do presente Trabalho, foi usada uma rotina em Python para acessar e modificar os arquivos de simulação conforme os intervalos desejados.

Uma vez criados os novos arquivos atp, é então usada uma rotina responsável por compilar cada um deles, gerando, assim, os arquivos pl4 que guardam as formas de onda de tensão e corrente de cada um desses casos. Na sequência, os arquivos pl4 são convertidos em arquivos csv para que os dados contidos neles possam ser acessados mais facilmente, resultando, assim, em um arquivo csv para cada caso.

Com as formas de onda de cada caso já devidamente salvas, é então iniciado o processo de rotulação dos dados contidos em cada um desses arquivos. Nesse sentido, são definidos dois tipos de rótulos que serão associados a cada dado, sendo o primeiro relativo à classe que o dado pertence e o segundo referente ao tipo de evento. No primeiro, o objetivo é indicar se o dado pertence ou não à classe de falta de alta impedância, o que é indicado por 1 para os dados que pertencem, e por 0 para os dados que não pertencem. Já o segundo diz respeito ao tipo de chaveamento, o qual pode ser, por exemplo, uma energização de transformadores, chaveamento de banco de capacitores ou a própria falta de alta impedância.

3.3.2 Separação dos Casos

Com a criação do banco de dados finalizada, o número total de casos é então dividido nos conjuntos de treinamento, validação e teste. O primeiro grupo contém 60% dos casos e tem a função de ser empregado para treinar os modelos de ML. Já o segundo, é composto por 20% dos casos e é usado para avaliar se o modelo está apto para ser colocado em operação ou

não, buscando identificar problemas como, por exemplo, *overfitting*. Por fim, também contendo 20% dos casos, o conjunto de teste é usado para obter as métricas finais de desempenho da metodologia proposta.

Como cada conjunto tem uma função específica, a forma como cada um deles é utilizado também varia. Um exemplo disso é que, para fins da presente pesquisa, tanto o conjunto de treinamento quanto o de validação têm os seus casos mesclados em um único bloco de dados, o que acontece depois da etapa de extração de características. Em contrapartida, para o conjunto de teste, a metodologia é aplicada em cada caso separadamente, uma vez que se deseja analisar a operação dela no tempo. Sendo assim, pode-se abordar os conjuntos de treinamento e de validação em termos de sua quantidade de dados, enquanto o conjunto de teste pode ser apresentado em termos de sua quantidade de casos.

3.3.3 Balanceamento dos Dados

Foi realizado o balanceamento tanto dos eventos quanto das classes presentes nos conjuntos de dados de treinamento e validação. Desse modo, foi possível garantir que determinados tipos de amostras não fossem representadas excessivamente mais que outras dentro do conjunto de dados, o deixando menos tendencioso. Para tal, foram usados os rótulos associados a cada um dos dados e o balanceamento foi dividido em duas etapas, sendo elas:

- **Balanceamento de eventos:** No balanceamento de eventos, é calculado o número de dados pertencentes a cada evento, o que pode ser feito a partir da contagem de rótulos. Após isso, usando essas contagens, os eventos com um maior número de elementos vão tendo os seus dados removidos, de modo a terem a mesma quantidade de dados que o evento com o menor número de elementos.
- **Balanceamentos de classes:** Com os eventos de ambas as classes já balanceados, é então iniciado o processo de remoção de elementos da classe com maior número de dados. Para isso, é necessário remover um dado de cada evento da classe majoritária a cada iteração, de modo que esse processo de balanceamento não afete o balanceamento entre os eventos presentes na classe. Logo, se, por exemplo, a classe de dados não pertencentes a FAI for a maior e conter 12 tipos diferentes de eventos, a cada iteração, serão removidos 12 dados da mesma, um de cada evento. Por fim, o processo é finalizado quando a diferença entre as classes for menor que o número de eventos da classe com mais elementos ($N_{maj} - N_{min} < Eventos_{maj}$).

Assim, após a aplicação dessas duas etapas nos conjuntos de treinamento e validação, são obtidos os conjuntos resultantes, os quais apresentam classes com os mesmo número de elementos, sendo que os dados dos eventos dentro de cada classe tem o mesmo número de recorrências entre si.

3.4 Métricas de Desempenho do Classificador em Relação ao Banco de Dados

Para problemas de classificação binária, existem algumas métricas que permitem extrair interpretações importantes a respeito da performance de um determinado classificador (JIAO e DU, 2016; Gong, 2021). Para isso, definem-se quatro contagens que podem ser feitas a respeito da classificação real dos dados e a predição realizada:

- Verdadeiro Positivo (VP): dados de FAI que foram classificados corretamente;
- Verdadeiro Negativo (VN): dados de outros eventos que foram classificados corretamente;
- Falso Positivo (FP): dados de outros eventos que foram classificados como sendo de FAI; e
- Falso Negativo (FN): dados de FAI que foram classificados como sendo pertencentes aos demais eventos.

Com base nas definições acima, podem ser definidas as métricas de desempenho de um classificador. A primeira é a *classifier-accuracy* (*ACC*) e mede a quantidade de acertos em relação a todo o conjunto de dados conforme (3.13).

$$ACC = \frac{VP + VN}{VP + VN + FP + FN} \quad (3.13)$$

Entretanto, nem sempre a *ACC* é uma métrica completamente informativa, uma vez que ela mostra o resultado como um todo. Assim, as vezes é importante quantificar o quanto de certeza um modelo de ML tem acerca de uma predição positiva ou o quanto de casos positivos do total ele é capaz de identificar. Para esses casos que se estabelecem métricas como a *Precision* e o *Recall*.

Precision (*Prec*) quantifica a frequência com que os dados preditos como sendo positivos estavam corretos e é calculado a partir da Equação (3.14).

$$Prec = \frac{VP}{VP + FP} \quad (3.14)$$

Essa métrica é importante em situações em que se precisa saber o quanto de certeza um classificador tem quando define um dado como sendo positivo, ou seja, pertencente a classe analisada.

Já o *Recall* quantifica o quanto do total de dados positivos foram classificados corretamente e é calculado a partir de (3.15).

$$Recall = \frac{VP}{VP + FN} \quad (3.15)$$

A importância dessa métrica está em representar a capacidade que um determinado classificador tem de identificar um dado positivo diante de sua ocorrência.

Por fim, tem-se a métrica conhecida como *F1 score*, a qual faz uma média harmônica entre essas duas métricas e permite ter uma noção geral do classificador nesses dois sentidos. O cálculo do *F1 score* é dado conforme a Equação 3.16.

$$F1\ score = 2 \cdot \frac{Prec \cdot Recall}{Prec + Recall} = \frac{2\ TP}{2\ TP + FP + FN} \quad (3.16)$$

3.5 Seleção de Características

Para avaliar se todas as características escolhidas inicialmente realmente estavam contribuindo com a performance dos classificadores, foi utilizada uma técnica de seleção de características conhecida como *Recursive Feature Elimination with Cross-validation* (RFECV). Nessa técnica, a partir do uso de um algoritmo de ML, são atribuídas importâncias às características de acordo com o papel que elas desempenham na separação das classes do problema analisado. Assim, a característica que for considerada pelo algoritmo como tendo uma importância menor é removida e o modelo é treinado novamente com as que restaram de modo a obter novos valores de importância, sendo esse procedimento repetido até que fique apenas uma. Além disso, o desempenho do algoritmo é avaliado para os diferentes números de características por meio do processo de validação cruzada no conjunto de dados selecionado (Awad; Fraihat, 2023).

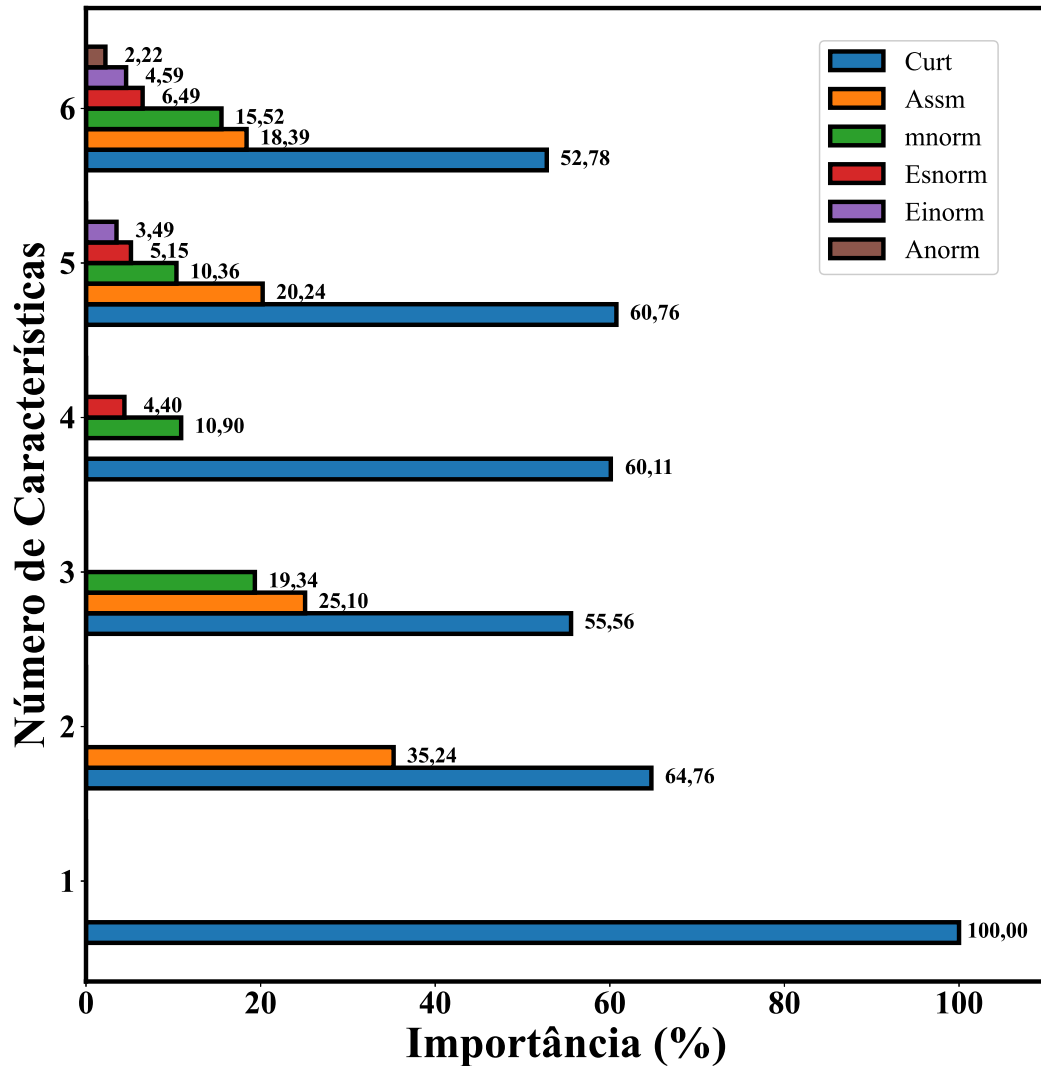
No presente Trabalho, foi escolhido o algoritmo *Random Forest* para atribuir importância às características inicialmente propostas, sendo estas a $E_{S_{norm}}$, $E_{i_{norm}}$, A_{norm} , m_{norm} , $Curt$ e $Assm$, já discutidas anteriormente nesse Capítulo. Além disso, os resultados são referentes à aplicação da técnica de seleção de características no conjunto de dados de treinamento, o qual, por sua vez, teve as suas classes e eventos balanceados.

Na Figura 28, estão as importâncias atribuídas às variáveis em cada etapa da eliminação recursiva de características.

Nesse sentido, é primeiramente realizado o treinamento com todas as características escolhidas para, então, remover aquela que apresentar a menor importância, a qual, para o caso analisado, foi a área normalizada da envoltória superior (A_{norm}). Na sequência, vão sendo removidas as demais características como a envoltória inferior normalizada ($E_{i_{norm}}$), a envoltória superior normalizada ($E_{S_{norm}}$), a média normalizada (m_{norm}) e, por fim, a assimetria amostral ($Assm$).

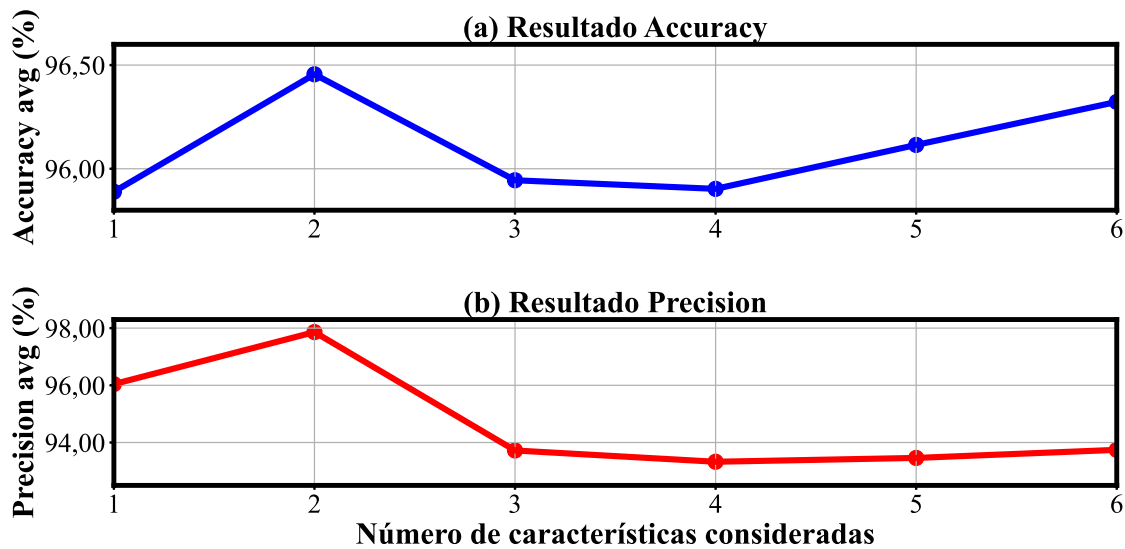
Cada uma dessas etapas é representada na Figura 28 por um grupo de barras, no qual estão apresentadas todas as características restantes. Assim, o grupo com seis barras é referente à etapa onde todas as características foram usadas para treinar o modelo e as importâncias resultantes desse treinamento foram obtidas. Já o com cinco barras é referente à etapa onde uma das características já foi eliminada e foi realizada novamente a obtenção das importâncias. Esses grupos seguem até o que contém apenas a barra referente à última característica, a qual, obviamente, tem uma importância de 100%. As importâncias estão apresentadas em porcentagem, conforme a sua contribuição para a capacidade de separação das classes do algoritmo *Random Forest*.

Figura 28 – Sequência de remoção das características e suas respectivas importâncias.



Fonte: Autor.

Como já discutido anteriormente, além de remover as características conforme a sua importância para o algoritmo, essa técnica também realiza a avaliação das métricas de desempenho em cada uma dessas etapas. As Figuras 29(a) e 29(b) apresentam os resultados do desempenho do modelo de ML para diferentes quantidades de características usando as métricas *Accuracy* e *Precision*, respectivamente. Como pode ser visto, para ambos os casos, o algoritmo apresentou o melhor desempenho usando apenas duas de um total de seis características analisadas. Isso indica que as demais características ou foram pouco informativas ou apresentaram informações redundantes, o que contribuiu para a redução de desempenho do modelo de ML. Essas duas características para as quais o modelo apresentou um melhor resultado são a curtose e a assimetria, conforme já apresentado na Figura 28. Assim, tomando como base o resultado da seleção de características, daqui para a frente serão usadas apenas essas duas características para a realização do treinamento dos modelos e sua operação.

Figura 29 – Desempenho da *Random Forest* para diferentes números de características.

Fonte: Autor.

3.6 Treinamento e Validação dos Modelos de ML

A partir da escolha das características curtose e assimetria como as que serão usadas pela metodologia proposta, foi então dado início ao processo de treinamento dos modelos de ML. Para isso, foram escolhidos nove algoritmos, sendo eles *Decision Tree* (DT), *k-Nearest Neighbors* (KNN), *Random Forest* (RF), *Extremely Randomized Trees* (ET), *Adaptative Boosting* (AdaB), *Gradient Boosting Machine* (GBM), *Extreme Gradient Boosting Machine* (XGBM), *Light Gradient Boosting Machine* (LGBM) e *Categorical Gradient Boosting Machine* (CGBM).

Esses algoritmos foram treinados a partir de 1722 casos, os quais foram mesclados em um único bloco de dados, sendo este balanceado tanto em relação à sua classe quanto em relação aos seus eventos, de modo que o banco de dados resultante fosse o mais representativo possível e não desse demasiada ênfase para determinados eventos em detrimento de outros. Dessa forma, o resultado desse balanceamento está apresentado na Tabela 1, a qual apresenta o número de dados usados para o treinamento separados pelas suas respectivas classes e eventos.

O conjunto de dados relativos aos casos de FAI os quais pertencem a classe positiva 1 é resultado da junção das amostras de sinais de quatro diferentes modelos de arco e apresentam um total de 276.376 dados, onde cada modelo de arco tem 69.094 dados. Esses modelos de arco são definidos da seguinte maneira, conforme a referência de cada um:

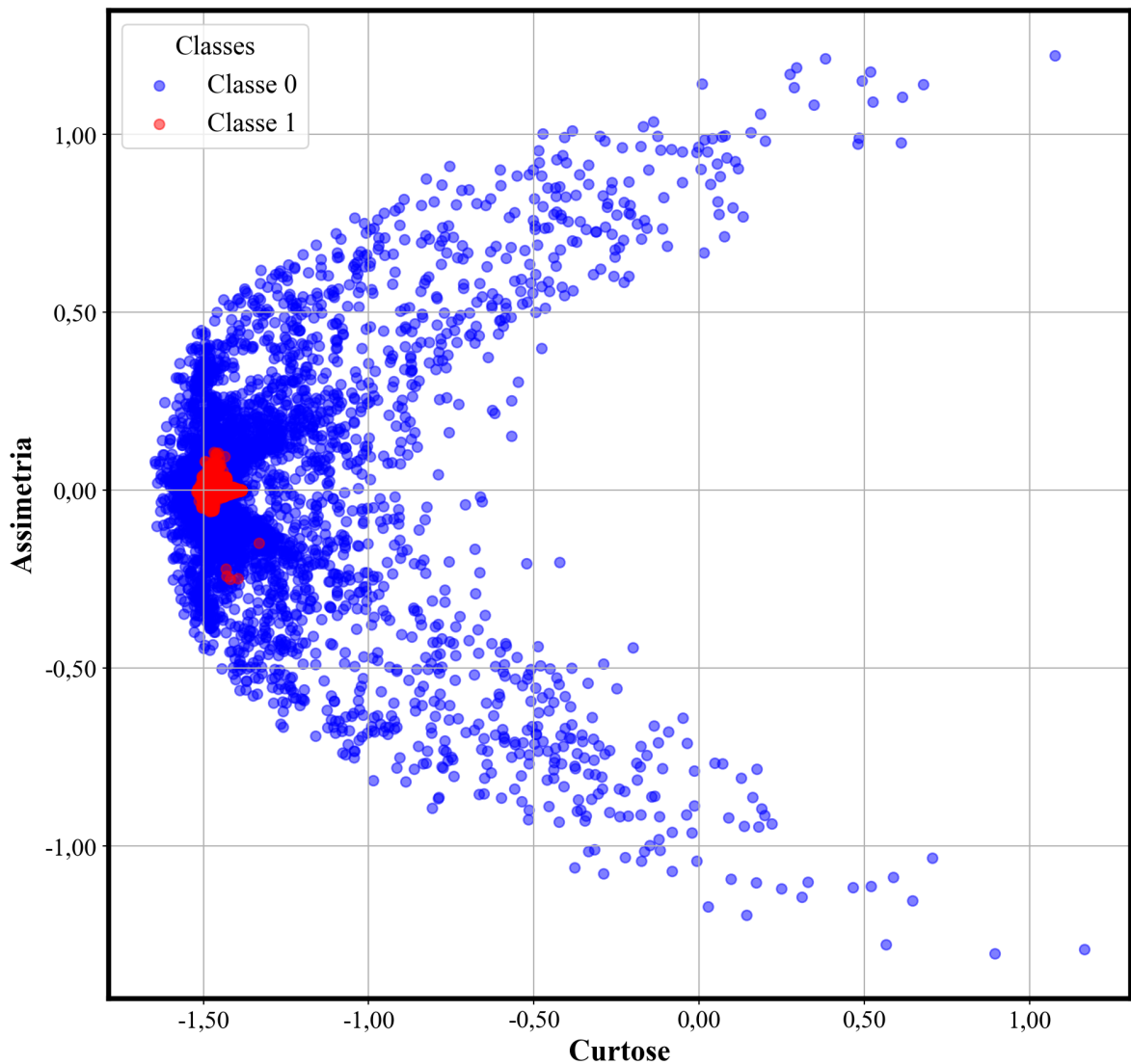
- Modelo de Arco 1: Lai *et al.* (2005);
- Modelo de Arco 2: Gautam e Brahma (2013);
- Modelo de Arco 3: Sedighi (2014); e

- Modelo de Arco 4: Torres (2014).

Em contrapartida, o conjunto de dados relativo aos demais eventos que estão descritos na mesma Tabela 1 e pertencem a classe negativa 0, apresentam um total de 276.374 dados, tendo cada um desses eventos 19.741 dados.

Além disso, uma vez que o conjunto de treinamento é composto por dados de curtose e assimetria amostral extraídos do sinal de corrente, é possível construir um plano a partir dessas duas características. Assim, a Figura 30 apresenta a distribuição dos pontos contidos no bloco mesclados pós-balanceamento dos eventos e das classes. Este é o conjunto que, de fato, é usado para o treinamento dos modelos de ML.

Figura 30 – Dados de Treinamento: Curtose x Assimetria Amostral.



Fonte: Autor.

Na Figura 30, é possível observar o comportamento dos dados pertencentes às situações de FAI, os quais estão em vermelho, e àqueles referentes aos demais tipos de situações, demarcados em azul. Enquanto os pontos azuis estão relativamente bem mais distribuídos ao longo do plano,

os pontos em vermelho tendem a se concentrar em uma determinada região, encontrando-se mais a centro-esquerda do gráfico. Dessa forma, é possível concluir que os classificadores empregados para realizar a separação dessas duas classes devem concentrar as suas fronteiras de classificação ao longo dessa pequena região onde a maioria dos pontos vermelhos estão concentrados.

Tabela 1 – Quantidade de dados usados para o treinamento dos modelos de ML.

Classe	Dados por classe	Evento	Dados por evento
1	276.376	Modelo de Arco 1	69.094
		Modelo de Arco 2	
		Modelo de Arco 3	
		Modelo de Arco 4	
0	276.374	Operação Normal	19.741
		Carga Capacitiva 1f	
		Carga Capacitiva 3f	
		Entrada de Carga 1f	
		Entrada de Carga 3f	
		Carga Indutiva 1f	
		Carga Indutiva 3f	
		Chaveamento Indutivo e Capacitivo	
		Curto-Circuito 1f	
		Curto-Circuito 2f	
		Curto-Circuito 2f-terra	
		Curto-Circuito 3f	
		Curto-Circuito 3f-terra	
Energização de Transformadores 75 kVA			

Fonte: Autor.

Os algoritmos utilizados apresentam vários parâmetros, os quais podem ser modificados conforme a necessidade do problema abordado. Para o caso do presente Trabalho, onde os classificadores são empregados para a detecção de faltas de alta impedância, os parâmetros ajustados para cada um dos modelos de ML são os seguintes:

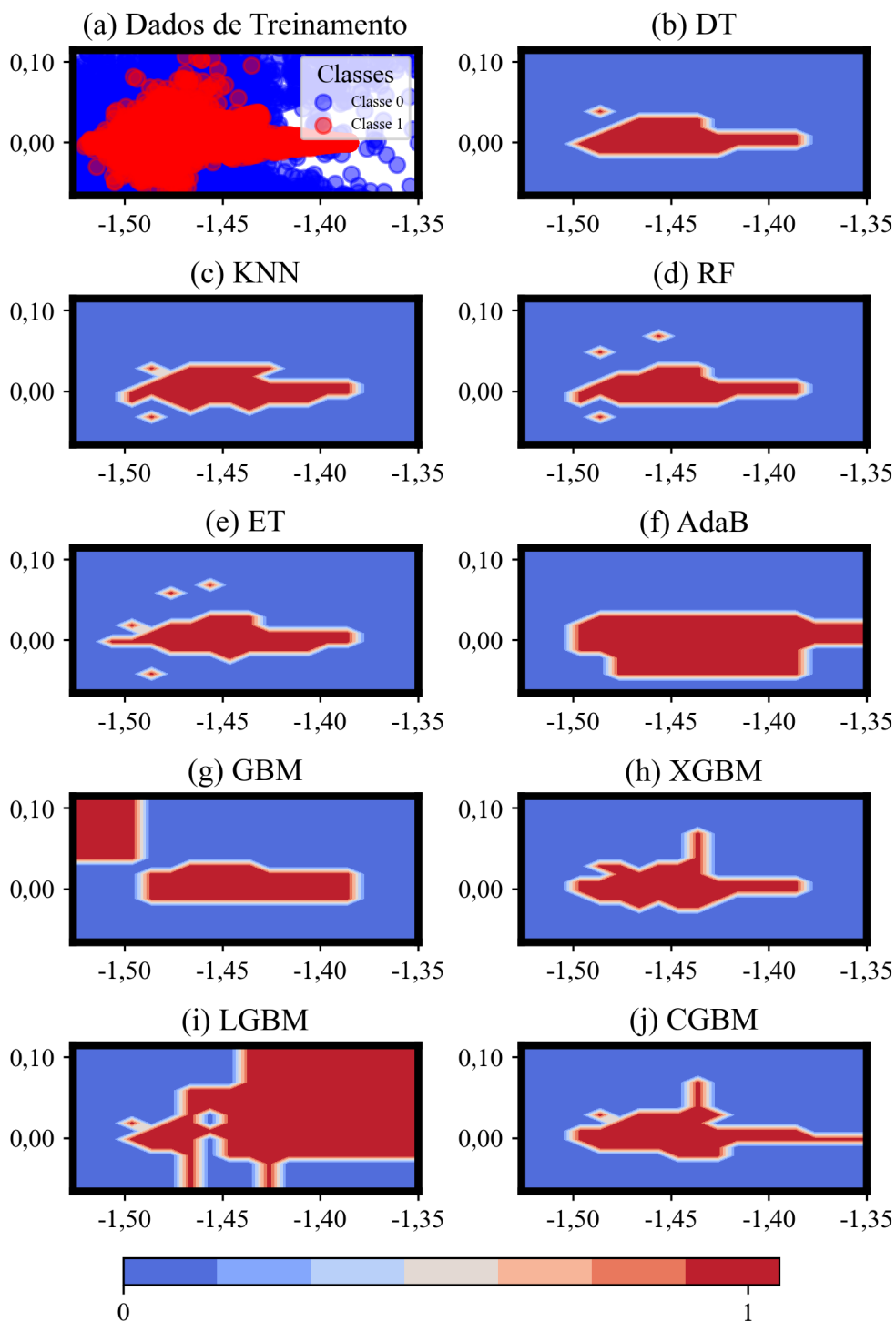
- DT
 - Profundidade máxima (max_depth): 10;
 - Critério (criterion): Gini.
- KNN
 - Número de vizinhos (n_neighbors): 5.
- RF
 - Número de estimadores (n_estimator): 20;

- Critério (criterion): Gini.
- ET
 - Número de estimadores (n_estimator): 20;
 - Critério (criterion): Gini.
- AdaB
 - Número de estimadores (n_estimator): 20;
 - Taxa de aprendizado (learning_rate): 1,0.
- GBM
 - Número de estimadores (n_estimator): 100;
 - Taxa de aprendizado (learning_rate): 1,0;
 - Profundidade máxima (max_depth): 3.
- GBM
 - Número de estimadores (n_estimator): 100;
 - Taxa de aprendizado (learning_rate): 1,0;
 - Profundidade máxima (max_depth): 3.
- XGBM
 - Número de estimadores (n_estimator): 100;
 - Taxa de aprendizado (learning_rate): 1,0;
 - Profundidade máxima (max_depth): 6.
- LGBM
 - Número de estimadores (n_estimator): 100;
 - Taxa de aprendizado (learning_rate): 1,0.
- CGBM
 - Número de estimadores (n_estimator): 100;
 - Taxa de aprendizado (learning_rate): 1,0.

Os parâmetros usados ou foram atribuídos conforme os ajustes padrão de cada algoritmo, ou conforme aqueles que possibilitaram uma maior velocidade de treinamento e operação sem perder a capacidade de classificação do algoritmo.

Definidos os algoritmos que serão utilizados, ajustados os seus parâmetros e balanceados os dados que serão utilizados, é dado início então à etapa de treinamento. O resultado do treinamento de cada um dos modelos é mostrado na Figura 31, na qual é apresentada graficamente a diferença de cada um dos modelos treinados a partir das regiões delimitadas por eles como sendo pertencente a FAI ou não.

Figura 31 – Fronteiras de decisão delimitadas pelo treinamento de cada modelo de ML.



Primeiramente, na Figura 31(a), é apresentada a região onde se encontra a maioria dos pontos referentes à FAI no plano formado pelos valores de curtose e assimetria. Já nas Figuras 31(b) a 31(j), são apresentadas as fronteiras de cada um dos modelos de ML treinados, ilustrando assim as diferentes formas com que os classificadores separaram as classes e que regiões eles consideram como tendo uma probabilidade maior de se encontrarem dados pertencentes à FAI. Assim, as regiões avermelhadas indicam as posições nas quais os dados são considerados como sendo de FAI e as regiões em azul, as posições relativas a todos os demais eventos.

Após isso, com esses classificadores já treinados, foi realizada a sua aplicação nos dados de validação para a identificação de possíveis problemas. De modo a alcançar essa finalidade e fazer com que as métricas fossem afetadas da forma mais uniforme possível pelos diferentes tipos de eventos simulados, foi realizado também o processo de mesclagem dos sinais de validação simulados e, na sequência, o seu balanceamento tanto em relação às classes quanto em relação aos eventos.

Esse balanceamento é necessário, uma vez que existem tipos de dados que são muito mais numerosos do que os demais, como exemplo os dados de operação normal do sistema que aparece em praticamente todas as simulações antes do chaveamento que marca o início dos eventos. Caso um algoritmo apresentasse facilidade em classificar corretamente esse tipo específico de situação, ele automaticamente já teria boas métricas devido à quantidade muito maior desse tipo de dado. Assim, é realizado o balanceamento de modo que cada cenário possa ter o mesmo nível de impacto nos valores finais das métricas de desempenho em relação aos demais eventos pertencentes a sua mesma classe. O resultado desse balanceamento em termos de dados é apresentado na Tabela 2.

Para que fosse possível avaliar o desempenho dos modelos recém-treinados diante de um conjunto de dados ainda não visto, além de usar simplesmente os dados de validação, também foi adicionado diferentes níveis de ruído nos casos pertencentes a esse conjunto. Dessa forma, seria possível avaliar como o desempenho de cada um dos algoritmos cairia, conforme apresentados à sinais com níveis maiores de ruído. Esse processo de adição de ruído foi realizado nas formas de onda dos sinais de corrente de cada um dos casos, sendo que, após isso, foram realizados os mesmos processos de mesclagem e balanceamento de dados já discutidos anteriormente, tendo como resultado o mesmo número de dados presentes na Tabela 2.

Ademais, foram usadas diferentes métricas de desempenho como *Accuracy*, *Precision*, *Recall* e *F1 score*, sendo essa performance medida em relação a diferentes conjuntos de dados, iniciado pelos dados de treinamento. Com isso, foi possível verificar o quão bem o algoritmo conseguiu se ajustar em relação aos dados usados para obtenção de seus parâmetros internos.

Na sequência, foi avaliado o conjunto de dados de validação, para detectar possíveis quedas significativas de desempenho, o que poderia indicar possíveis ocorrências de *overfitting*. Por fim, foram analisados diferentes níveis de ruído adicionados aos sinais dos casos de validação, variando a relação sinal-ruído (SNR, do inglês *Signal-to-Noise ratio*) de 100 até 20 dB através de um ruído branco gaussiano.

Tabela 2 – Quantidade de dados usados para a validação modelos de ML.

Classe	Dados por classe	Evento	Dados por evento
1	93.460	Modelo de Arco 1	23.365
		Modelo de Arco 2	
		Modelo de Arco 3	
		Modelo de Arco 4	
0	93.450	Operação Normal	6.675
		Carga Capacitiva 1f	
		Carga Capacitiva 3f	
		Entrada de Carga 1f	
		Entrada de Carga 3f	
		Carga Indutiva 1f	
		Carga Indutiva 3f	
		Chaveamento Indutivo e Capacitivo	
		Curto-Circuito 1f	
		Curto-Circuito 2f	
		Curto-Circuito 2f-terra	
		Curto-Circuito 3f	
		Curto-Circuito 3f-terra	
Energização de Transformadores 75 kVA			

Fonte: Autor.

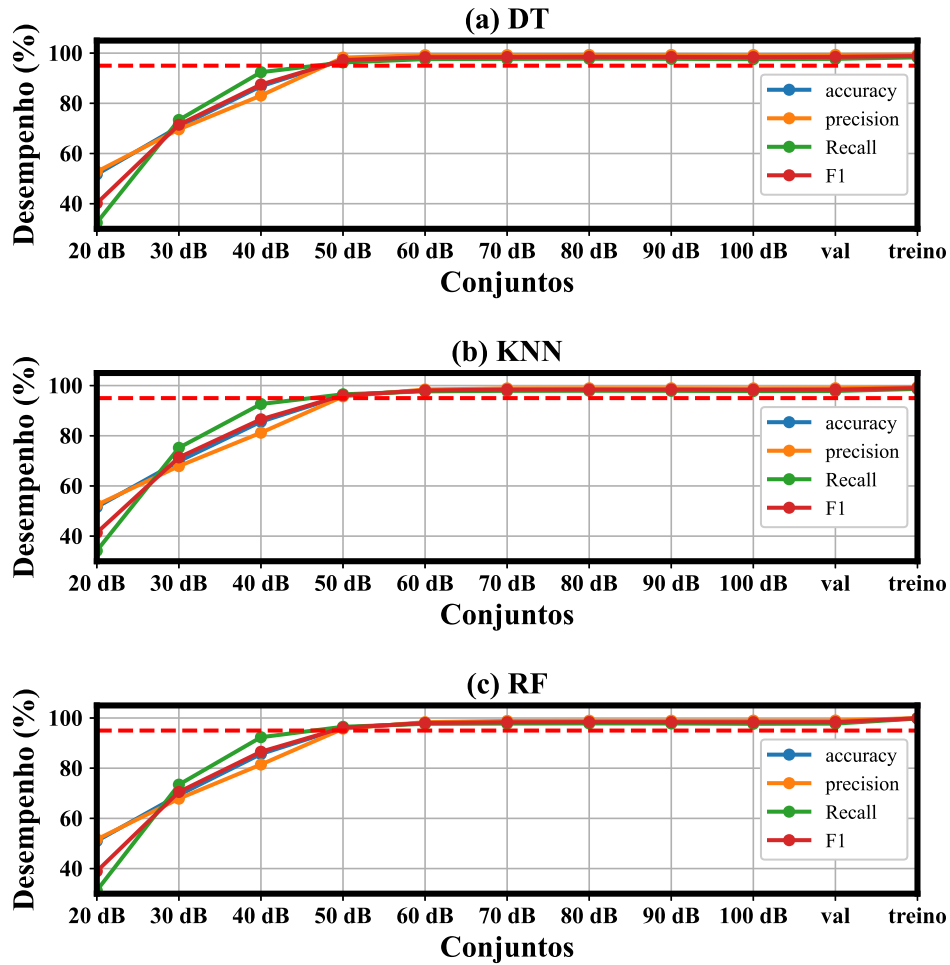
Na Figura 32(a), são apresentadas as métricas de desempenho da DT para os conjuntos de treinamento e de validação, sem e com diferentes níveis de ruído, sendo que esse último apresenta valores de SNR que variam entre 20 e 100 dB.

Além disso, foi traçada uma reta pontilhada vermelha em 95% de desempenho para uma melhor localização dos pontos dentro do gráfico. Dessa forma, observa-se que a DT apresenta valores acima da reta pontilhada para todas as suas métricas até 50 dB, tendo um decaimento significativo para valores abaixo desse. Ainda sobre as métricas, a *Recall* foi a que apresentou um maior decaimento conforme o nível de ruído foi aumentando, enquanto a métrica *Precision* teve uma queda consideravelmente menor. Essa análise vale também para o caso da Figura 32(b) e 32(c), apresentando os algoritmos KNN e RF, respectivamente, uma vez que todos apresentam um comportamento bastante semelhante.

Já em relação ao Figura 33, existem algumas diferenças que valem a pena ser consideradas. Pois, embora 33(a) e 33(c), tenham um comportamento muito semelhante ao analisado para os casos anteriores, a 33(b), a qual é referente ao AdaB, apresenta algumas diferenças no comportamento de suas métricas.

Uma dessas mudanças está na forma como os valores da *Recall* e *Precision* decaem com o nível de ruído, já que agora a segunda tem uma diminuição maior dos seus valores do que a primeira, o oposto do que acontecia antes. Além disso, outra diferença é que, em 20 dB, esse modelo apresentou um desempenho um pouco melhor que os demais, o que não necessariamente indica que ele seja melhor quando usado juntamente a metodologia proposta.

Figura 32 – Análise dos modelos treinados para DT, KNN e RF.

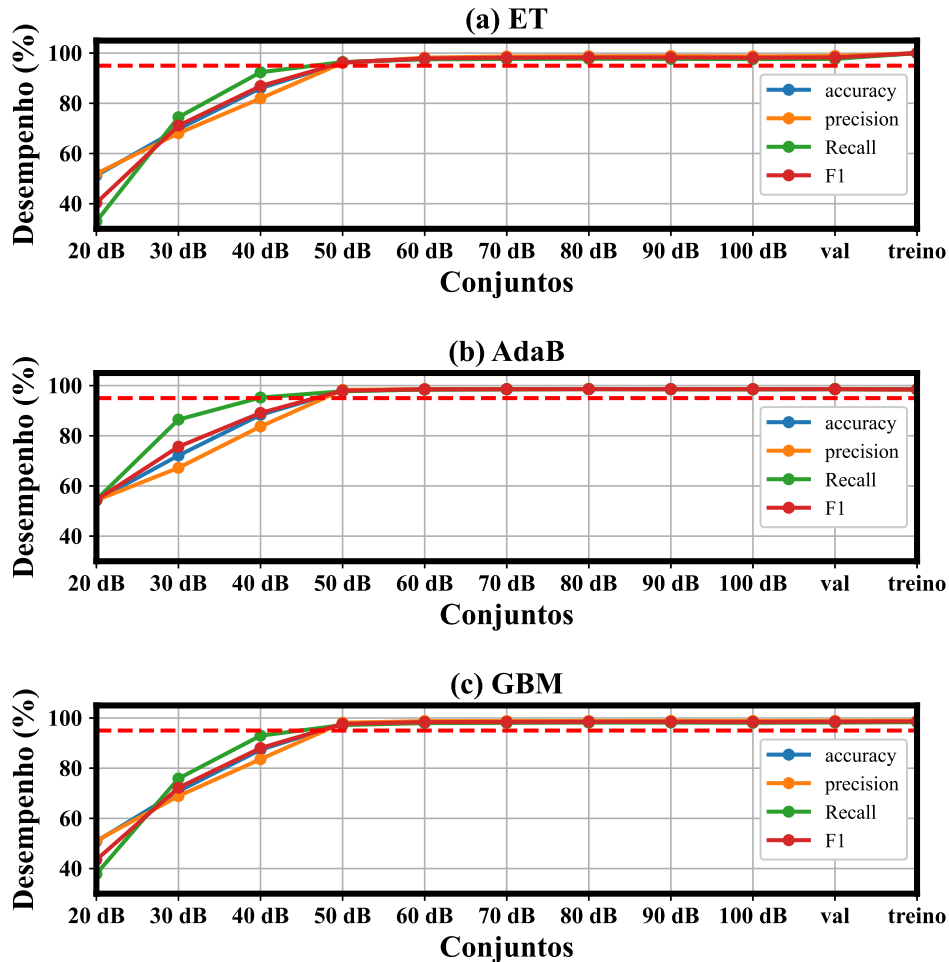


Fonte: Autor.

Por último, a Figura 34 apresenta o comportamento das métricas referentes aos modelos XGBM, LGBM e CGBM. Esses algoritmos têm um comportamento muito semelhante aos dos casos anteriores, com destaque apenas para o LGBM, o qual apresentou um comportamento com algumas semelhanças em relação ao AdaB e sendo o que obteve as melhores métricas para o caso de 20 dB.

Com base na discussão realizada acerca de cada uma das figuras, é importante ressaltar que, de forma geral, todas alcançaram métricas acima de 95% até 50 dB. Dessa forma, demonstrando uma boa adaptabilidade a diferentes tipos de dados. Ademais, considerando que, dependendo do nível de ruído, a tarefa de classificar corretamente as amostras do sinal em algum momento ficaria consideravelmente difícil, é esperado que houvesse um decréscimo em algum ponto, ficando bem perceptível abaixo de 40 dB. Entretanto, o fato dos modelos terem lidado consideravelmente bem para todos os valores acima de 50 dB e não ter havido uma queda considerável do conjunto de treinamento para o de validação, isso leva à conclusão de que os modelos estão aptos para serem colocados em operação.

Figura 33 – Análise dos modelos treinados para ET, AdaB e GBM.



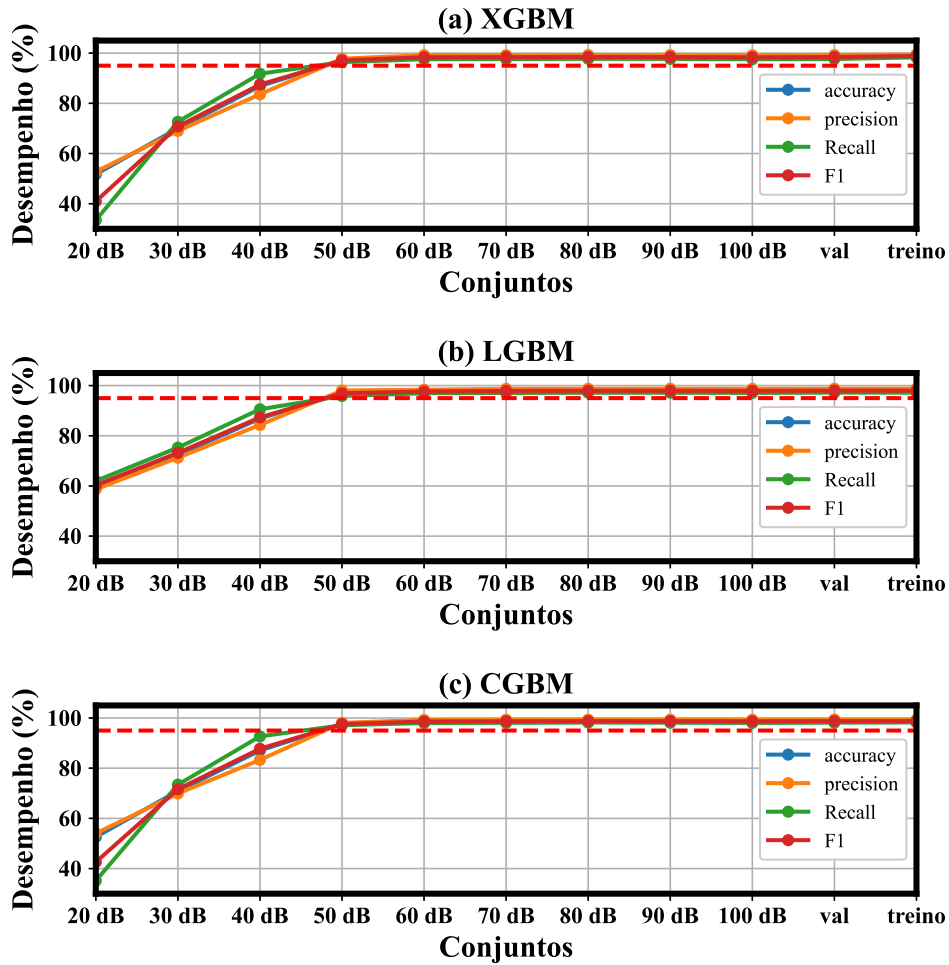
Fonte: Autor.

3.7 Lógica de Operação

Após a escolha e avaliação dos modelos de ML utilizados nesse Trabalho, a etapa seguinte é aplicá-los no banco de casos de teste. Para isso, é necessário definir a forma como isso deve ser feito, uma vez que, nesta etapa, diferentemente das anteriores, nas quais os dados eram mesclados em um único bloco de dados, a análise ocorrerá por cada caso individualmente. Logo, a classificação dos dados presentes dentro de um determinado caso se torna apenas um dos aspectos da operação da lógica que se deseja construir, já que essa classificação deve satisfazer um determinado critério para que a detecção seja realizada.

Desse modo, um fator do qual pode ser tirado proveito é a sequencialidade das amostras no tempo. Já que, durante a ocorrência de uma falta de alta impedância, por exemplo, é de se esperar que os dados de FAI sejam detectados consecutivamente, o que pode ser usado como um critério para aumentar a robustez da metodologia a falsos positivos. Por esse motivo que a detecção da FAI se dará após um determinado número de amostras ser classificado ininterruptamente como sendo pertencente à FAI.

Figura 34 – Análise dos modelos treinados para XGBM, LGBM e CGBM.



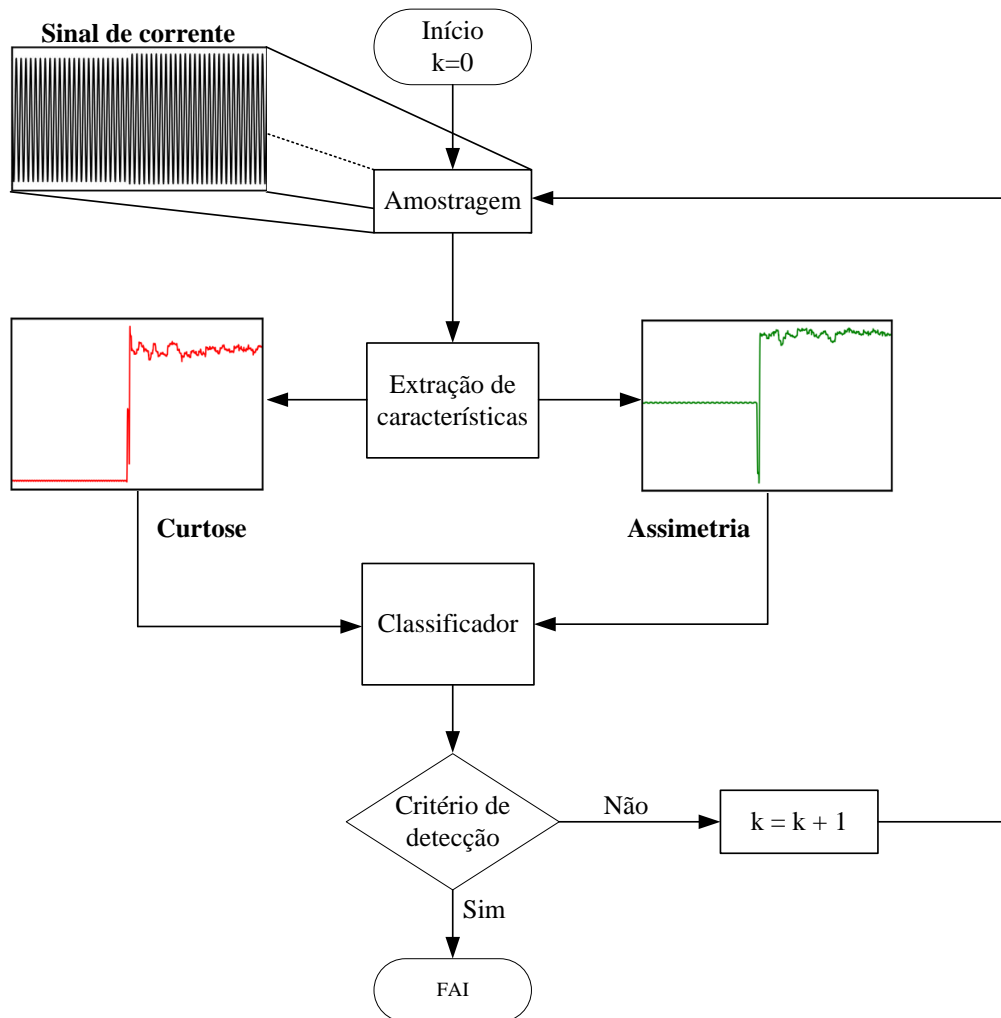
Fonte: Autor.

Com base no que foi discutido até então, pode-se definir a metodologia proposta como sendo dividida em três etapas principais:

- **Extração de características do sinal de corrente da subestação:** a partir da corrente de fase obtida do TC, a qual é armazenada em uma janela de um ciclo, a lógica realiza os cálculos necessários para obter o valor de curtose e de assimetria amostral dos dados presentes nessa janela;
- **Classificação das características pelo modelo de ML:** O valor obtido de ambas as características é, então, usado como entrada de um classificador, já previamente treinado. Basicamente, o que será avaliado é se o ponto formado pelos valores de curtose e assimetria amostral se encontra na região do plano de características pertence à FAI ou não, sendo esta regiões delimitadas durante o treinamento dos modelos;
- **Critério de detecção:** Para aumentar a robustez da detecção da FAI, a metodologia emite o alerta somente após um número específico de dados positivos consecutivos.

Cada uma dessas etapas desempenha um papel importante para o correto funcionamento da metodologia e estão apresentadas na Figura 35.

Figura 35 – Fluxograma da metodologia proposta.



Fonte: Autor.

A partir dessa Figura, é possível visualizar a relação que cada um dos procedimentos tem entre si, bem como a sequência de cada um. Além disso, a Figura 36 apresenta a metodologia funcionando para um caso de falta de alta impedância. Nesse caso, a incidência da falta ocorre no instante demarcado pela linha vertical pontilhada, a qual divide o sinal de corrente entre a operação normal do sistema e após o início do arco elétrico. Pode-se observar, em (a), a forma de onda de corrente vista da subestação, sendo as características extraídas apresentadas em 36(b) e 36(c). Além disso, na Figura 36(d), é mostrada a resposta do classificador para essas duas características, onde o nível lógico 1 é referente aos dados que o modelo classifica como sendo pertencentes à FAI e 0 como não sendo pertencentes.

Por fim, o ponto vermelho indica o instante onde ocorre a detecção da falta de alta impedância. Para esse exemplo apresentado, o critério de decisão escolhido para que houvesse

a detecção foi a classificação ininterrupta dos dados como sendo pertencentes à FAI, por um intervalo de tempo equivalente a 3 ciclos consecutivos da senóide de frequência fundamental do sistema.

3.8 Escolha do Critério de Detecção

Uma vez apresentada a metodologia, resta ainda definir o número de ciclos usados no critério de detecção. Nesse sentido, o contador tem que alcançar o número de amostras relativo à quantidade de ciclos escolhida, sendo que isso deve ser feito de forma ininterrupta, caso contrário, o contador é reiniciado do zero. Assim, para escolher o número de ciclos a ser utilizado nesse Trabalho, foram avaliadas diferentes possibilidades sendo estas o uso de meio ciclo, um ciclo, dois, três e quatro ciclos. O algoritmo utilizado para avaliar esses diferentes critérios de detecção foi o *Random Forest*, o qual foi aplicado no conjunto de validação. Dessa forma, para que fosse possível analisar o desempenho da metodologia, foi computado o número de casos corretamente classificados pela lógica para diferentes níveis de ruído, buscando, assim, analisar como que cada critério se comportaria conforme a quantidade de ruído no sinal aumentasse.

A Figura 37 apresenta um gráfico com os dados obtidos para cada um dos números de ciclos utilizados, considerando diferentes valores de relação sinal ruído.

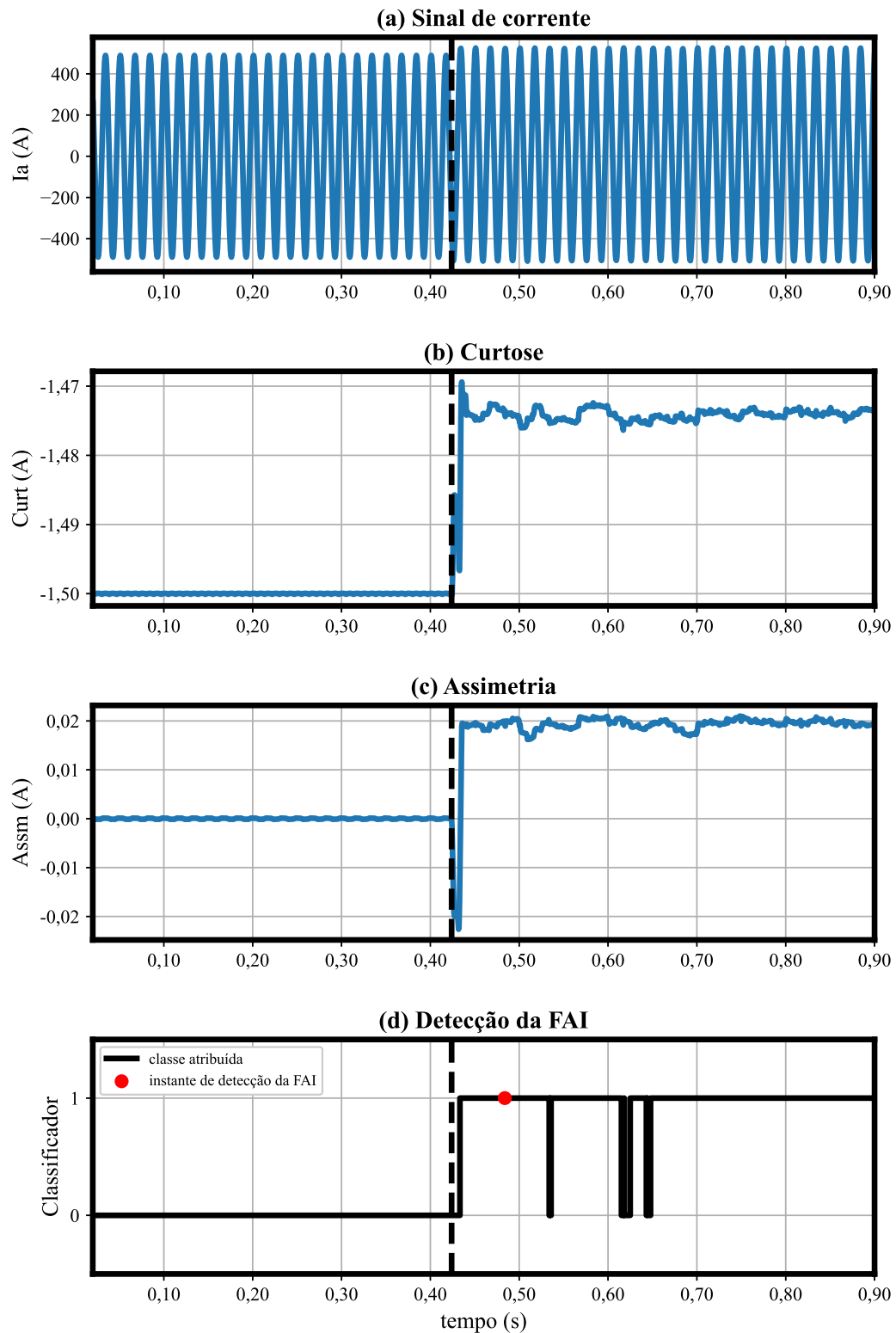
Nesse gráfico, pode ser observado que, embora os critérios apresentem comportamentos parecidos de queda conforme o valor de dB diminui, ainda assim existem critérios que performam bem abaixo dos demais, como é o caso do uso de meio ciclo e um ciclo, os quais apresentam quedas consideravelmente abruptas de seu desempenho a partir dos 55 dB. Entretanto, para os demais casos, principalmente os que são relativos aos uso de dois e três ciclos, essa percepção já não é tão fácil de ser obtida, uma vez que existem intervalos onde o uso de dois ciclos apresenta uma porcentagem de acerto maior, como também existem casos onde isso ocorre com o uso de três ciclos.

Assim, para que fosse possível avaliar de forma quantitativa qual seria o melhor critério de detecção a ser utilizado, foi usado o cálculo da área sob cada uma das curvas da Figura 37 para realizar essa decisão. Nesse sentido, para realizar esse cálculo usando os dados que foram obtidos durante a aplicação da *Random Forest* no conjunto de validação, foi usada a regra do trapézio. Além disso, de modo a facilitar a visualização dos resultados e a interpretação dos mesmos, o valor de cada uma dessas áreas foi normalizado pelo valor da maior área.

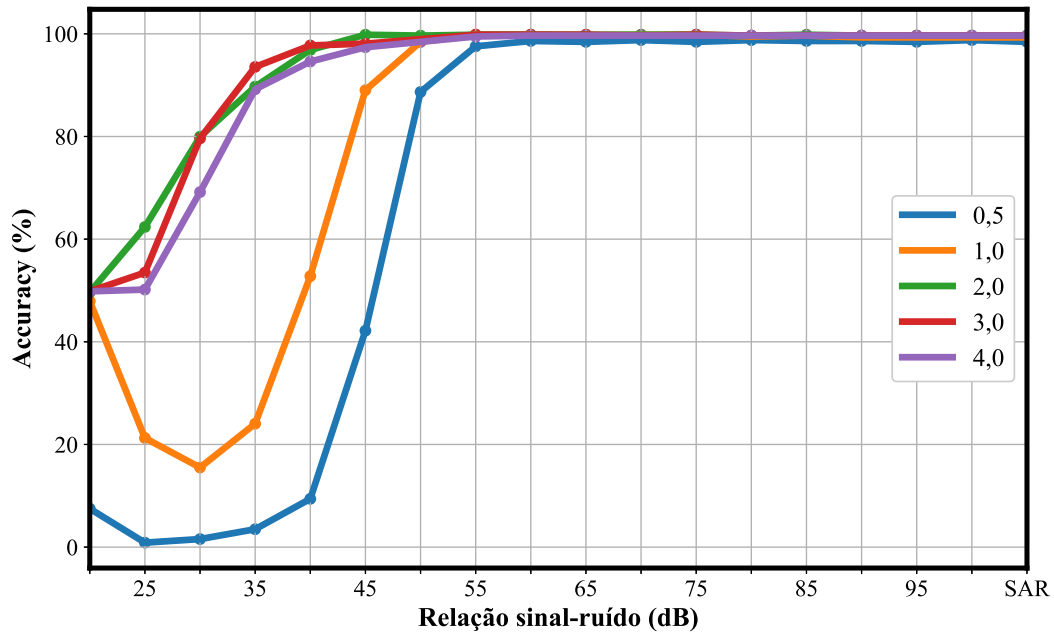
Com base em tudo que foi dito até então, o número de ciclos escolhido foi aquele que apresenta a maior área ao longo do eixo dos dBs, sendo este representado na Tabela 3 pelo valor 1, devido à normalização realizada.

Dessa forma, a partir da análise da Tabela 3, é possível observar que o uso de 2 ciclos como critério de detecção foi o que apresentou a melhor área. Portanto, daqui para a frente, a metodologia contará 2 ciclos consecutivos de dados sendo classificados como sendo de FAI para realizar a detecção.

Figura 36 – Exemplo de operação da lógica para um caso de FAI.



Fonte: Autor.

Figura 37 – Avaliação de diferentes critérios de detecção com a *Random Forest*.

Fonte: Autor.

Tabela 3 – Resultado do cálculo da área usando a regra do trapézio.

Critério	Área (normalizada)
meio ciclo	0,73980952
1 ciclo	0,85632653
2 ciclos	1
3 ciclos	0,99548299
4 ciclos	0,98089796

Fonte: Autor.

3.9 Considerações Finais

Neste Capítulo, foi realizada uma discussão acerca de cada um dos aspectos envolvidos na implementação de modelos de ML para a detecção de FAI nesse Trabalho. Esses aspectos envolvem desde os detalhes a respeito dos dados usados para a validação e treinamento desses modelos, até quais características foram mais decisivas para a correta classificação dos dados referentes ao problema abordado. A partir disso, foi decidido pelo uso das características curtose e assimetria amostral para abordar o problema em questão. Além disso, foram apresentados os algoritmos usados e as informações referentes ao seu processo de treinamento. Com base no que foi discutido nesse assunto, foi concluído que os modelos treinados não apresentaram nenhum tipo de *overfitting* ou qualquer outro tipo de problema, indicando, assim, que os mesmos estão aptos para serem colocados em operação no conjunto de teste.

4 SIMULAÇÕES E RESULTADOS

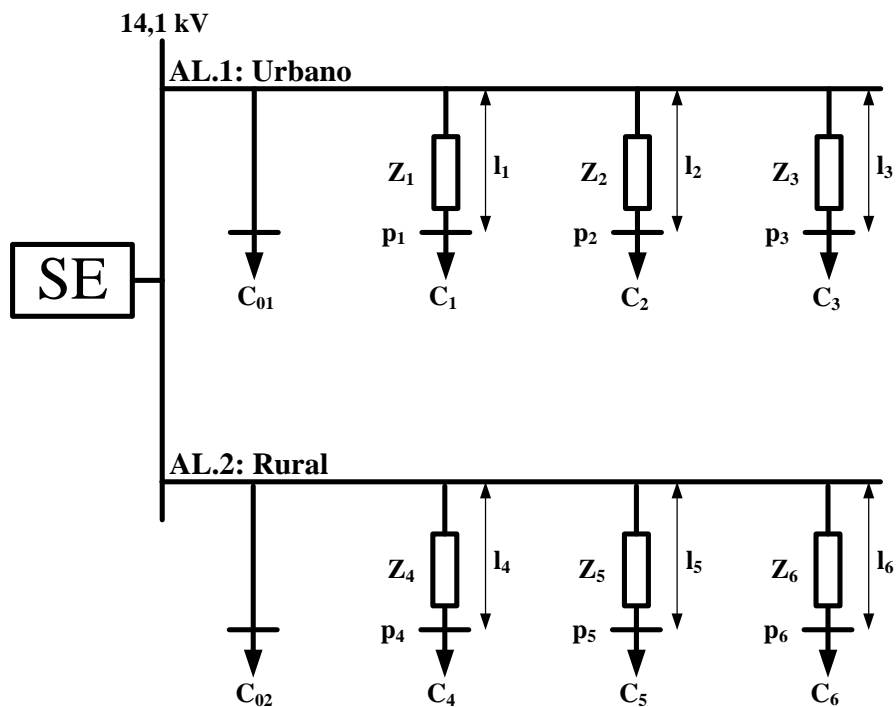
4.1 Considerações Gerais

Este Capítulo apresenta o sistema de teste no qual foram simulados os casos utilizados neste Trabalho. Em seguida, são detalhados os eventos simulados, incluindo os intervalos de variação de cada parâmetro para a geração dos casos. Posteriormente, foram selecionados 9 casos para realizar a análise temporal da metodologia, demonstrando como a resposta dos classificadores varia conforme os diferentes tipos de eventos estudados. Após essa análise, são apresentadas as métricas gerais para cada um dos classificadores, permitindo avaliar a performance da metodologia em cada caso. Finalmente, é analisado o desempenho da metodologia em relação aos valores de SNR, variando de 100 dB a 20 dB, para entender o impacto do aumento do nível de ruído no sinal de corrente sobre as métricas de desempenho da metodologia.

4.2 Sistema Teste

Nesse Trabalho, o sistema teste usado para gerar os diversos casos de transitórios eletromagnéticos é o mesmo presente em Stein (2019), sendo este construído a partir da ferramenta ATPDraw. Esse sistema é mostrado na Figura 38 e é composto por uma subestação rebaixadora de 69/13,8 kV, da qual saem dois alimentadores, AL.1 e AL.2, sendo o primeiro um alimentador urbano e o segundo um alimentador rural.

Figura 38 – Sistema teste utilizado.



Fonte: Autor.

A frequência dessa rede de distribuição é 60 Hz, sendo os sinais de corrente de fase amostrados a uma taxa de 32 amostras/ciclo. Os pontos de p₁ até p₆, além de serem onde estão

localizadas as cargas do sistema, também são os locais nos quais são realizados os chaveamentos dos modelos de arco elétrico de FAI e dos demais eventos. Além disso, o TAP do transformador está ajustado para 14,1 kV e cada alimentador tem três ramificações, as quais, por sua vez, possuem um determinado valor de impedância de linha acumulada por trecho até suas cargas, sendo estes apresentados na Tabela 4.

Tabela 4 – Valores de impedância acumulados por trecho.

Trecho	$R_A=R_B=R_C$ (Ω)	$X_A=X_B=X_C$ (Ω)
Z ₁	1,66	7,34
Z ₂	3,083	5,85
Z ₃	2,33	9,08
Z ₄	3,09	6,2
Z ₅	9,25	8,46
Z ₆	7,04	9,87

Fonte: Autor.

Outro detalhe importante é o fato de o alimentador rural apresentar distâncias maiores da subestação até as suas cargas C₄, C₅ e C₆, as quais se encontram, respectivamente, a 12, 15 e 17 km da subestação. Já o alimentador urbano apresenta cargas mais próximas da subestação, sendo estas C₁, C₂ e C₃ com distâncias de 7, 5 e 9 km, respectivamente. A relação entre as distâncias mencionadas e sua respectiva nomenclatura no sistema é apresentado na Tabela 5.

Tabela 5 – Distâncias de cada ponto do sistema até a subestação.

Nome	Distância da Subestação (km)
l ₁	7
l ₂	5
l ₃	9
l ₄	12
l ₅	15
l ₆	17

Fonte: Autor.

Ainda, além das cargas que se encontram após um determinado trecho, cada alimentador possui uma carga adicional, a qual é C₀₁, para o AL.1, e C₀₂, para o AL.2. Essas cargas se encontram diretamente conectadas no barramento da subestação.

A Tabela 6 apresenta as informações das cargas utilizadas no sistema. Nesta Tabela estão os valores de fator de potência usados para cada uma das cargas, as quais são todas indutivas, além de sua potência aparente.

Em relação a essas cargas, também é indicada a tensão no ponto do sistema no qual elas se encontram. É importante mencionar que o sistema teste em questão é balanceado e, portanto, o valor das suas impedâncias de linha e de suas cargas é igual para cada uma de suas fases.

Tabela 6 – Características de cada uma das cargas.

Alimentadores	Cargas	FP	Potência Aparente (MVA)	Tensão (kV)
AL.1	C ₀₁	0,95	4,8	14,1
	C ₁	0,93	2,35	12,9
	C ₂	0,96	0,3	13,9
	C ₃	0,93	0,7	13,7
AL.2	C ₀₂	0,98	5,37	14,1
	C ₄	0,96	0,9	13,7
	C ₅	0,97	0,4	13,8
	C ₆	0,93	0,075	14

Fonte: Autor.

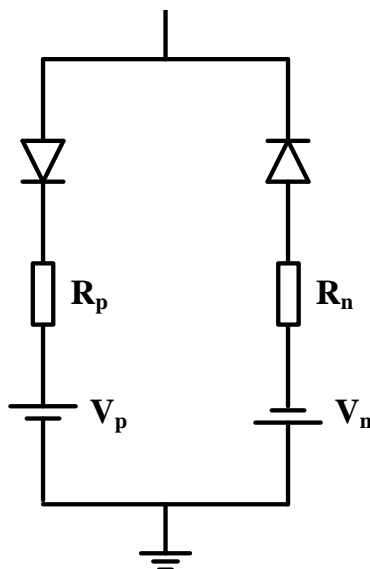
4.2.1 Simulações de FAI

Apresentadas as características do sistema teste de forma geral, agora são apresentadas como foram realizadas as simulações de FAI. Assim, de modo que fosse possível extrair diferentes comportamentos da forma de onda do sinal de corrente, foram utilizados quatro modelos diferentes de arco elétrico, que foram definidos na Seção 3.6. Nesse sentido, cada um dos modelos computacionais de arco elétrico conseguem representar diferentes características normalmente presentes durante o fenômeno físico de FAI em situações reais. Ao total foram simulados 1440 casos de FAI, dos quais 864 foram separados para serem usados no treinamento dos classificadores de ML, 288 para a validação e 288 para o teste da metodologia.

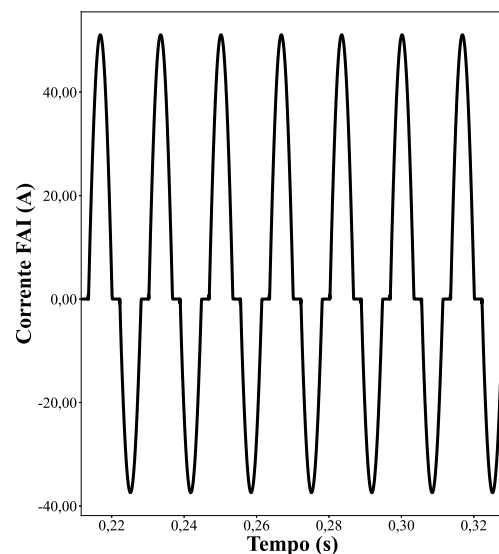
Nas Figuras 39(a) e 40(a), são ilustrados os esquemas referentes ao modelos de arco 1 e 2, respectivamente, e nas Figuras 39(b) e 40(b), são apresentadas as correntes resultantes do chaveamento desses arcos no sistema.

Figura 39 – Aspectos do Modelo de Arco 1.

(a) Modelo de Arco 1



(b) Corrente Resultante

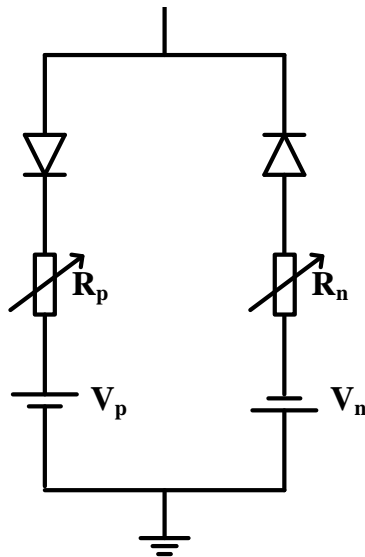


Fonte: Autor.

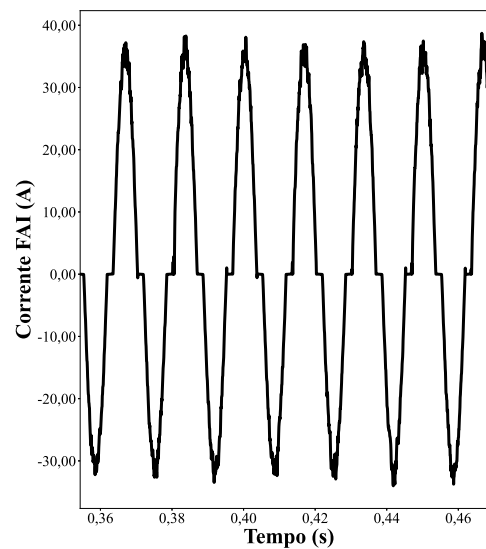
Para o modelo de arco 1 é utilizado uma fonte de tensão V_p em série com um diodo e uma resistência R_p , ambos em antiparalelo, com outra fonte de tensão V_n em série com um outro diodo e uma outra resistência R_n . Já o modelo de arco 2, embora apresente muitas semelhanças com o primeiro, principalmente em relação ao arranjo dos elementos utilizados, difere-se pelo comportamento de suas resistências R_p e R_n . Isto, pois, para esse modelo, essas resistências variam aleatoriamente a cada passo de simulação, possibilitando, assim, emular o comportamento estocástico muitas vezes visto durante a ocorrência de FAI. O fechamento de cada uma dessas chaves se dá uma após a outra, sendo esperado no máximo um tempo de um ciclo depois do chaveamento da anterior para a realização do fechamento da chave atual, o que é escolhido de forma aleatória.

Figura 40 – Aspectos do Modelo de Arco 2.

(a) Modelo de Arco 2



(b) Corrente Resultante



Fonte: Autor.

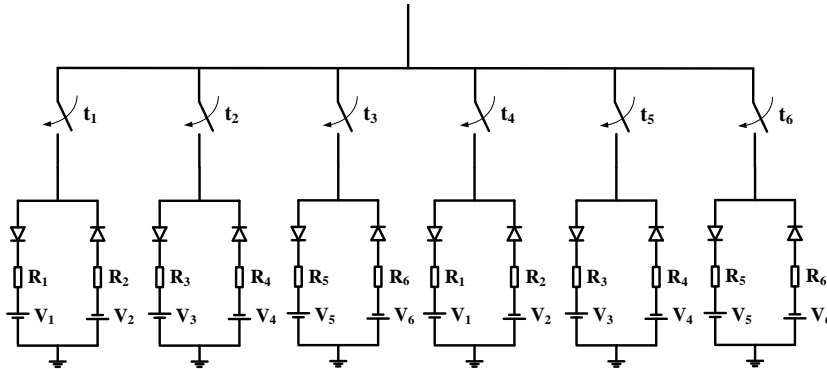
O modelo de arco 3 é composto por um conjunto de modelos de arco 1 em paralelo. Nesse sentido, para fins de facilitar a variação dos parâmetros referentes a esse arco elétrico, foram realizadas algumas modificações, principalmente em relação às fontes de tensão e às resistências pertencentes ao circuito do arco elétrico, porém, sem retirar a ideia central por trás do modelo. Mais especificamente, as tensões V_1 , V_3 e V_5 são referentes às tensões de ignição do semiciclo positivo, e as tensões V_2 , V_4 e V_6 , dos semiciclos negativos. Entretanto, de modo a utilizar 6 arcos em paralelo o padrão é repetido conforme indicado na Figura 41(a). A ideia principal é diminuir a quantidade de parâmetros necessários para realizar as simulações, facilitando assim o controle e reduzindo significativamente os erros que poderiam ocorrer durante a simulação. A corrente resultante desse arco elétrico vai crescendo gradualmente conforme cada uma das chaves t_1 até t_6 vai se fechando, sendo o resultado desse processo apresentado na Figura 41(b).

Já o modelo de arco 4 foi simulado utilizando uma resistência variável (R_f) controlada por TACS, conforme apresentado na Figura 42(a). Nesse modelo, os comportamentos associados

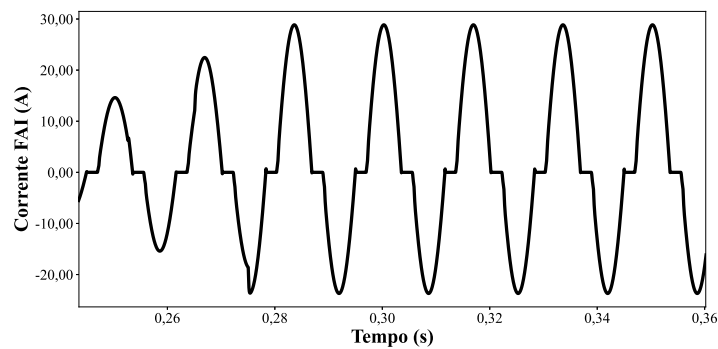
a FAI eram obtidos através da modelagem matemática do comportamento dinâmico do arco elétrico. Assim, a resistência vai variando no tempo, de modo a resultar nas distorções comumente encontradas nesse tipo de problema, além da própria etapa de *Buildup*, o que pode ser visto na Figura 42(b).

Figura 41 – Aspectos do Modelo de Arco 3.

(a) Modelo de Arco 3



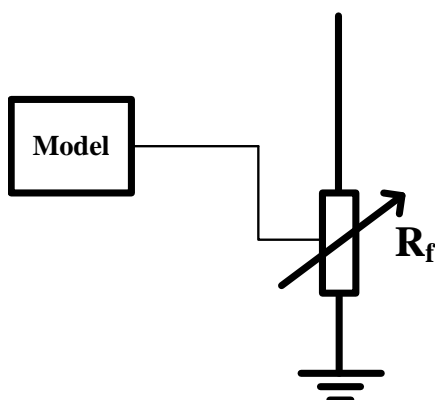
(b) Corrente Resultante



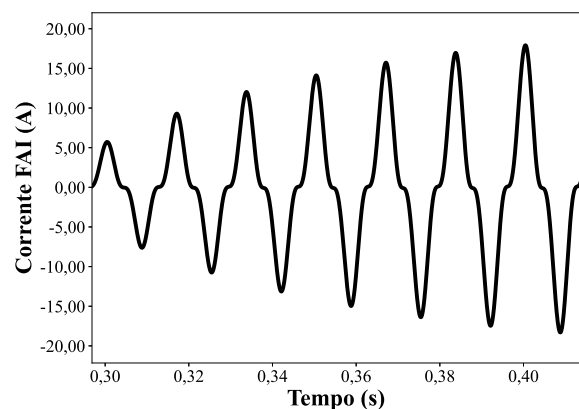
Fonte: Autor.

Figura 42 – Aspectos do Modelo de Arco 4.

(a) Modelo de Arco 4



(b) Corrente Resultante



Fonte: Autor.

Com base no que foi discutido até então, é possível perceber que mesmo nos modelos de arco que apresentam *buildup*, a envoltória que o acompanha é diferente, uma vez que, em um dos arcos, esse processo se dá através da introdução consecutiva de diodos e fonte de tensão em antiparalelo no sistema teste e, no outro, esse processo se dá a partir de uma modelagem matemática, a qual é usada para controlar uma resistência não-linear. Dessa forma, a partir do uso de diferentes modelos é possível analisar a resposta do classificador de ML em cenários que apresentam diferentes números de características de FAI sendo usadas no sinal, ou em cenários onde essas características se diferem um pouco de um modelo para o outro. Isto é interessante, uma vez que, em cenários reais, é possível que não só diferentes quantidades de características estejam presentes, como também estas se apresentem das mais diversas formas. Além disso, vale ressaltar que no tocante a simulações realizadas, todos os quatro modelos de arco tiveram o mesmo número de casos simulados e foram divididos igualmente dentre os conjuntos de treinamento, validação e teste, permitindo, assim, a representação mais homogênea possível de cada um destes nos bancos de dados utilizados. Por fim, pode-se resumir as características de FAI no sinal de corrente representados por cada um dos arcos elétricos da seguinte forma:

- **Modelo de arco 1:** Não-linearidade e Assimetria.
- **Modelo de arco 2:** Não-linearidade, Assimetria e Aleatoriedade.
- **Modelo de arco 3:** Não-linearidade, Assimetria, *Buildup* e *Shoulder*.
- **Modelo de arco 4:** Não-linearidade, *Buildup* e *Shoulder*.

Discutidos os aspectos gerais envolvendo os modelos de arco utilizados nesse Trabalho, agora é interessante definir os intervalos nos quais cada um deles foi variado. Os casos simulados foram gerados variando diferentes parâmetros dentro de um intervalo específico, sendo cada um desses intervalos apresentados na Tabela 7.

Tabela 7 – Intervalos nos quais os parâmetros dos arcos elétricos foram variados à cada simulação.

Modelo de Arco	Parâmetros variados	Intervalos
Arco 1 e Arco 2	Início	0,15 até 0,45 s
	V_p	3000 até 6000 V
	V_n	$1,2 \cdot V_p$ até $1,4 \cdot V_p$
	R_p	90 até 800 Ω
	R_n	R_p até $R_p + 0,2 \cdot R_p$
Arco 3	Início	0,15 até 0,45 s
	$V_{p=1,3,5}$	3000 até 6000 V
	$V_{n=2,4,6}$	$1,2 \cdot V_p$ até $1,4 \cdot V_p$
	R_1 até R_6	800 até 2000 Ω
Arco 4	Início	0,15 até 0,45 s
	τ	0,03 até 0,1 s
	G_0	0,0015 até 0,01 S

Fonte: Autor.

Outro fator analisado nesse estudo foi o impacto do rompimento do cabo no sinal de corrente visto da subestação. Para isso, metade das simulações de FAI realizadas apresentam o rompimento de cabo. Dessa forma, foi possível estudar se a queda de corrente resultante da desconexão de um determinado trecho do sistema interferiu no desempenho da metodologia proposta. As porcentagens de queda de corrente devido ao rompimento de condutor em diferentes pontos do sistema teste são apresentadas na Tabela 8.

Um exemplo de como fica a forma de onda do sinal de corrente durante a ocorrência desse rompimento pode ser visto na Figura 43, a qual ilustra como seria o comportamento da corrente diante de uma situação onde houvesse a desconexão de carga seguida pelo surgimento de um arco elétrico.

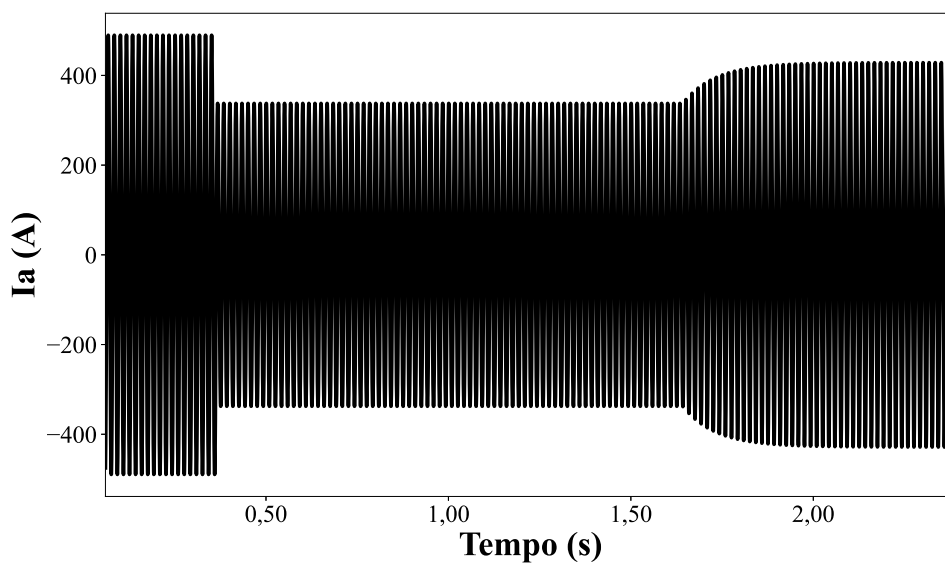
Tabela 8 – Porcentagem de queda de corrente resultante da desconexão de um determinado trecho.

Carga Removida (Fase A)	Queda de Corrente (%)
C ₁	31,06
C ₂	3,62
C ₃	8,59
C ₄	13,7
C ₅	6,02
C ₆	1,11

Fonte: Autor.

Para esse Trabalho, é realizada a consideração de que o cabo rompido cai de uma distância de 8 metros, com uma aceleração gravitacional de $9,8 \text{ m/s}^2$ e resistividade do ar desprezada. Logo, o tempo de queda do condutor é de aproximadamente 1,278 s.

Figura 43 – Exemplo de situação envolvendo rompimento do condutor seguido por arco elétrico.

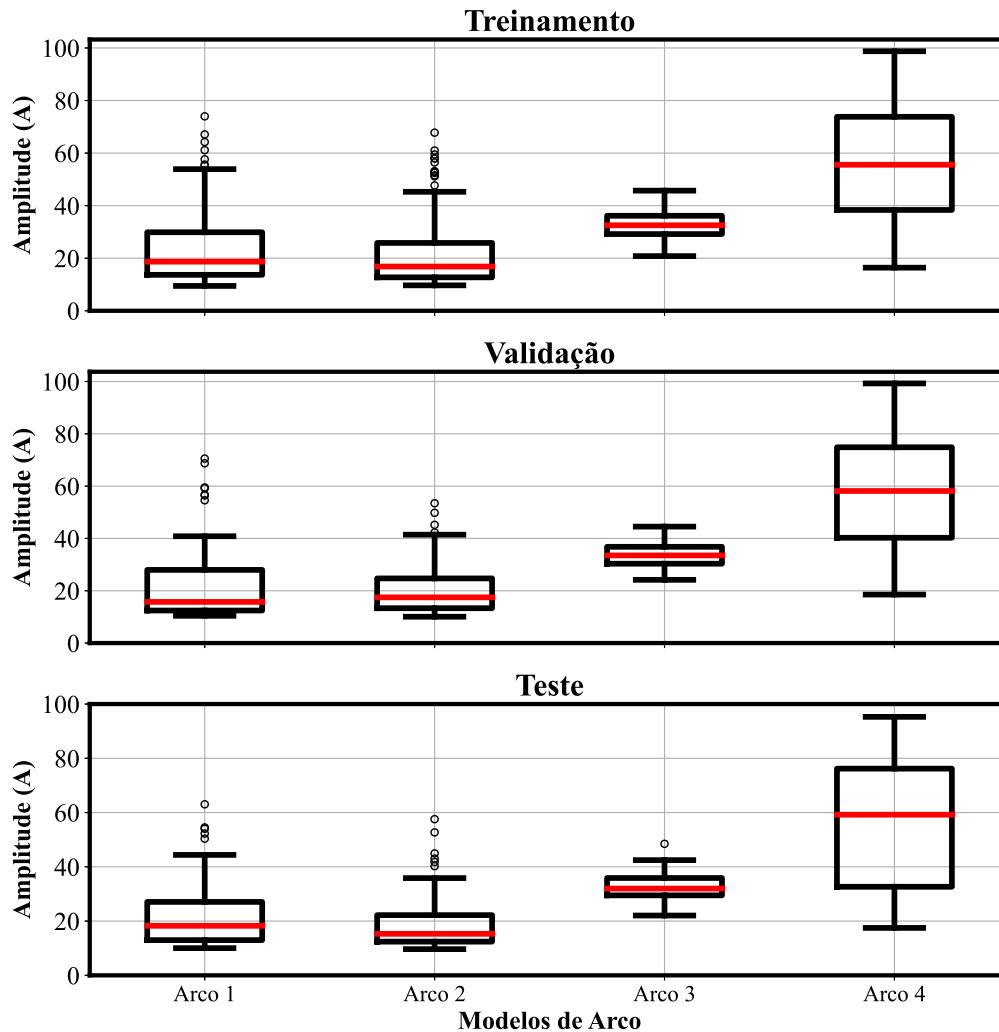


Fonte: Autor.

Discutidas as questões gerais envolvendo as simulações de FAI, é interessante apresentar as magnitudes das corrente resultantes do chaveamento dos arcos elétricos. Isso é importante

para estabelecer um entendimento geral de em que níveis de corrente de FAI os modelos foram treinados e validados e em quais níveis de corrente a metodologia foi testada. Dessa forma, foram calculadas as correntes de pico da FAI já em regime para cada um dos casos simulados. Os resultados são apresentados na Figura 44, que apresenta a distribuição desses valores de corrente para cada um dos conjuntos na forma de *boxplot*.

Figura 44 – Distribuição dos Valores de Pico de Regime das Correntes de FAI Simuladas.



Fonte: Autor.

4.2.2 Tipos de Eventos Simulados

Além dos modelos de falta de alta impedância, foram também simulados diferentes tipos de transitórios, de modo a verificar a robustez da metodologia frente à situações que poderiam ser confundidas com FAI. Para isso, foram realizadas 1430 simulações de chaveamentos comuns no sistema elétrico de distribuição, dos quais 858 foram separados para treinamento, 286 para validação e 286 para teste.

Dentre os eventos simulados, têm-se casos de: chaveamentos de banco de capacitores trifásicos e monofásicos, entradas de cargas indutivas trifásicas e monofásicas, bem como de

cargas puramente resistivas trifásicas e monofásicas, curtos-circuitos trifásicos, trifásicos-terra, bifásicos, bifásicos-terra e monofásicos; simulação híbrida chaveando no mesmo caso uma carga indutiva e uma capacitiva, ambas trifásicas e energização de transformadores. Para cada um desses eventos, foram realizadas 110 simulações em diferentes pontos do sistema. Os intervalos nos quais os parâmetros de cada simulação foram variados são apresentados na Tabela 9.

Os chaveamentos envolvendo cargas capacitivas e indutivas apresentam tanto o parâmetro reativo (B_C ou X_L) quanto o parâmetro resistivo R . Com isso, é possível controlar o nível de DC (do inglês, *Decaying component*) resultante do chaveamento de alguma desses dois tipos de carga. Essa variação também ocorre nos casos em que esses dois chaveamentos acontecem na mesma simulação, sendo que estes casos tem o objetivo de gerar distorções significativas nas formas de onda, além de possibilitar uma pluralidade maior de distorções nos sinais de corrente. As simulações de energização de transformadores são realizadas usando transformadores de 75 kVA, os quais normalmente estão presentes nas redes de distribuição de energia elétrica.

Tabela 9 – Intervalos utilizados para as simulações de cada tipo de transitório.

Nome do tipo de evento	Parâmetros variados	Intervalos de variação
Entrada de carga resistiva 1f, 3f	Início	0,15 até 0,45 s
	R	50 até 300 Ω
Entrada de carga capacitiva 1f, 3f	Início	0,15 até 0,45 s
	B_C	1000 até 3000 μS
	R_C	0,1 até 10 Ω
Entrada de carga Indutiva 1f, 3f	Início	0,15 até 0,45 s
	X_L	20 até 200 Ω
	R_L	0,02 até 20 Ω
Entradas de carga indutiva e capacitiva	$Início_C, Início_L$	0,15 até 0,45 s
	B_C	1000 até 3000 μS
	X_L	20 até 200 Ω
	R_C	0,1 até 10 Ω
	R_L	0,02 até 20 Ω
Curto-Circuito 1f, 2f, 2f-terra, 3f, 3f-terra	Início	0,15 até 0,45 s
	R	5 até 10 Ω
Energização de trafo	Início N_{trafo}	0,15 até 0,45 s 1 até 15 transformadores

Fonte: Autor.

Essas simulações são feitas usando diferentes quantidades desses transformadores, uma vez que geralmente, quando um trecho é desconectado e conectado novamente, vários desses transformadores são energizados simultaneamente. Dessa forma, para simular situações onde trechos com diferentes números de transformadores são energizados em um sistema de distribuição, são simulados casos com diferentes quantidade de trafos.

Outro grupo de simulações realizadas nesse Trabalho é o de curtos-circuitos, os quais variam entre monofásico, bifásico, bifásico-terra, trifásico e trifásico-terra. Para esse tipo de simulação, o parâmetro variado é a resistência de falta, a qual é utilizada como sendo a mesma tanto para o contato entre fases quanto o contato entre fases e a terra. Além disso, fora os

curtos-circuitos envolvendo todas as três fases, as fases que foram envolvidas no problema são escolhidas de forma aleatória a cada simulação.

4.3 Métricas de Desempenho da Metodologia em Relação aos Casos de Teste

Segundo Ghaderi (2014), para lidar com um problema de falta de alta impedância é necessário um conjunto de critérios para que possa ser possível tomar decisões em relação a esse problema. Uma vez que, embora desconectar uma possível FAI aumente a segurança das pessoas daquela região, isso pode também significar desconectar cargas importantes como hospitais, luzes de tráfego e elevadores.

Dessa forma, é necessário não somente saber a taxa de acertos de uma determinada metodologia de proteção, mas também saber se ela tem tendências a detectar incorretamente eventos normais do sistema como faltas de alta impedâncias ou se, embora sempre indique uma FAI corretamente, costuma não ser capaz de detectar a ocorrência de FAI com a frequência desejada.

As quatro contagens que podem ser definidas para o comportamento da metodologia em relação aos casos são:

- *CD*: casos de FAI detectados;
- *ID*: demais eventos incorretamente identificados como FAI;
- *ND*: casos de FAI não detectados; e
- *DE*: demais eventos para os quais a metodologia se comportou corretamente e não atuou.

Sendo assim, pode ser construída a matriz de confusão (*CM*, do inglês *confusion matrix*), conforme Ghaderi *et al.* (2014), onde, na sua diagonal principal, estão as contagens dos casos que a metodologia se comportou corretamente e, na diagonal secundária, estão os casos que a atuação não foi como desejada. Essa matriz é uma abordagem muito utilizada para uma melhor visualização dos resultados e é apresentada conforme a Equação 4.1.

$$CM = \begin{bmatrix} CD & ND \\ ID & DE \end{bmatrix} \quad (4.1)$$

Com base no que é apresentado em (4.1), podem-se obter diferentes métricas que possibilitam a análise do desempenho da metodologia em diferentes aspectos de seu funcionamento.

A primeira métrica apresentada é a *Relay Accuracy (A)*, a qual quantifica os acertos da metodologia em seu aspecto geral, dividindo os casos em que o relé se comportou corretamente pelo total de casos avaliados. Essa métrica é apresentada na Equação 4.2.

$$A = \frac{\text{Casos que Operou Corretamente}}{\text{Todos os Casos}} = \frac{CD + DE}{CD + ND + ID + DE} \quad (4.2)$$

A *Dependability* (D) é uma outra métrica, a qual quantifica a frequência com que a metodologia foi capaz de detectar uma falta de alta impedância em todas as vezes que ela ocorreu. Logo, essa métrica nos informa o quão sensível é o método à ocorrência da FAI e seu cálculo é apresentado na Equação 4.3.

$$D = \frac{\text{Casos de FAI Detectados pelo Relé}}{\text{Todos os Casos de FAI}} = \frac{CD}{CD + ND} \quad (4.3)$$

Já *Security* (S) quantifica a frequência com que a metodologia conseguiu distinguir algum outro evento como não sendo FAI e, portanto, não atuando incorretamente. É representada por (4.4).

$$S = \frac{\text{Casos Sem Atuação Indevida do Relé}}{\text{Todos os Casos dos Demais Eventos}} = \frac{DE}{DE + ID} \quad (4.4)$$

Safety (SF) é uma métrica que quantifica a probabilidade de um evento que não sensibilizou o relé não ser uma FAI (4.5).

$$SF = \frac{\text{Casos Sem Atuação Indevida do Relé}}{\text{Todos os Casos Sem Atuação do Relé}} = \frac{DE}{DE + ND} \quad (4.5)$$

Por fim, *Sensibility* (SN) quantifica a probabilidade da detecção realizada pela lógica ser de fato uma FAI. Seu cálculo é representado pela Equação 4.6.

$$SN = \frac{\text{Casos de FAI Detectados pelo Relé}}{\text{Todos os Casos que Fizeram o Relé Atuar}} = \frac{CD}{CD + ID} \quad (4.6)$$

4.4 Resultados

Após realizadas as etapas de simulação dos casos, juntamente com o treinamento e validação dos modelos de ML, é feita a etapa de aplicação desses modelos no banco de casos de teste. Nesse sentido, como já mencionado anteriormente, 288 casos de FAI foram separados para serem usados como teste, enquanto para os demais eventos esse valor foi de 286 casos, resultando em um total de 574 casos. Dessa forma, é aplicada a metodologia usando os modelos de ML, já previamente treinados, para avaliar como que o uso de diferentes algoritmos pode afetar o desempenho da classificação dos eventos simulados.

O objetivo dessa aplicação é analisar o quão bem os algoritmos de ML, mais especificamente os modelos não-lineares, conseguem performar na tarefa de detectar corretamente casos de FAI para o banco de casos simulados, quando usados juntamente com um critério de detecção. Para o presente Trabalho, esse critério é a classificação consecutiva de amostras como sendo pertencentes a uma ocorrência de FAI durante o tempo de dois ciclos.

Além disso, de modo que seja possível analisar diferentes aspectos dessa aplicação, são realizadas várias abordagens, como, por exemplo, a apresentação do comportamento da metodologia no tempo para alguns dos casos simulados, possibilitando, assim, uma melhor compreensão de como funciona a resposta dos classificadores frente a diferentes comportamentos

das características extraídas. Também, são analisadas as métricas gerais do desempenho dessa metodologia no banco de teste como um todo, o que possibilita ter uma noção mais abrangente e conclusiva da eficácia dos algoritmos de ML para essa tarefa. Além disso, são analisados os tempos de detecção de cada classificador e se existe alguma relação da eficácia de detecção e do tempo de detecção de cada um dos modelos. Por fim, ainda é discutido o impacto do ruído nas métricas de desempenho da metodologia de cada um dos casos.

4.4.1 Operação da Metodologia no Tempo

Para analisar a resposta dos classificadores e da metodologia em relação a casos de FAI e de eventos recorrentes no sistema elétrico de distribuição, foram escolhidos 9 casos, sendo 4 de FAI e 5 de eventos como entrada carga resistiva, entrada de carga capacitiva, entrada de carga indutiva, energização de transformadores e curto-circuito. Os parâmetros relativos à simulação de cada um desses casos, tais como o tipo de evento, local, instante de incidência e os demais parâmetros que caracterizam o respectivo evento, bem como os resultados obtidos pela atuação da metodologia em cada caso são apresentados nas Subseções a seguir.

4.4.1.1 Caso de FAI com o modelo de arco 1

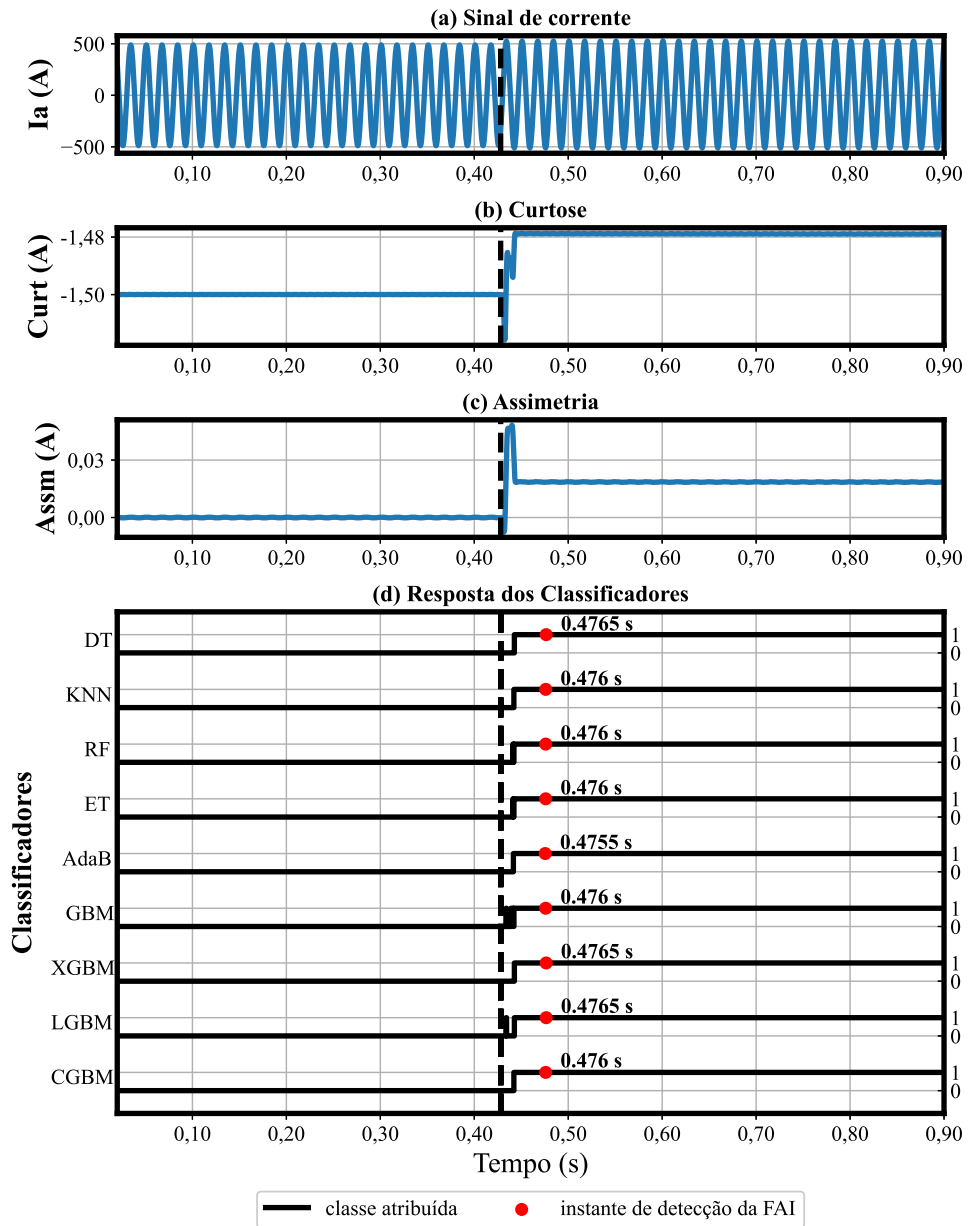
Detalhes da simulação: alimentador urbano, ponto p_2 , incidência em 0,42786 segundo, $V_p = 4828$ V, $R_p = 159$ Ω , $V_n = 6748, 1091$ V, $R_n = 186, 1598$ Ω .

O modelo de arco 1 apresenta as características de não-linearidade e assimetria entre os semiciclos. Além disso, vale ressaltar que o sinal apresentado é aquele visto da subestação, sendo este, portanto, resultado da soma da corrente do arco elétrico da FAI com a corrente drenada pelas demais cargas do sistema, o que explica a pouca diferença entre o sinal de pré-falta e pós-falta vista na Figura 45(a). A linha vertical pontilhada, representa o instante de chaveamento do modelo de arco elétrico.

Nas Figuras 45(b) e 45(c), são apresentados os comportamentos da curtose e da assimetria da distribuição estatística da corrente, ou assimetria amostral, conforme a janela móvel se movimenta ao longo do sinal. Assim, observando essas duas características, é perceptível a diferença de ambas quando comparado o seu comportamento de pós-falta com o de pré-falta, uma vez que o valor de ambas tende a crescer um pouco e permanecer acima do valor inicial. Com isso, é interessante o fato de que, mesmo que as características da FAI como não-linearidade e assimetria dos semiciclos ficando relativamente ocultadas no sinal de corrente visto da subestação, as características como curtose e assimetria amostral da janela móvel ainda assim apresentam uma diferença perceptível.

A Figura 45(d) apresenta a resposta de cada um dos classificadores para cada uma das amostras de curtose e assimetria amostral calculadas. Nesse sentido, todos os modelos apresentaram uma resposta bem consistente, tendo bem definidos os instantes em que o seu estado lógico é zero, o que ocorre antes do início da FAI, e em que seu estado lógico é um, o que ocorre após transcorrido um dado intervalo de tempo desde o início da falta. É interessante

Figura 45 – Operação no tempo para o caso com modelo de arco 1.



Fonte: Autor.

destacar que o estado lógico da resposta de cada um dos classificadores, na sua maioria, não muda instantaneamente após o início da falta, sendo esse comportamento provavelmente resultado do tempo de convergência dos filtros responsáveis pelo cálculo dessas características, uma vez que os mesmos apresentam uma breve oscilação durante a ocorrência do transitório.

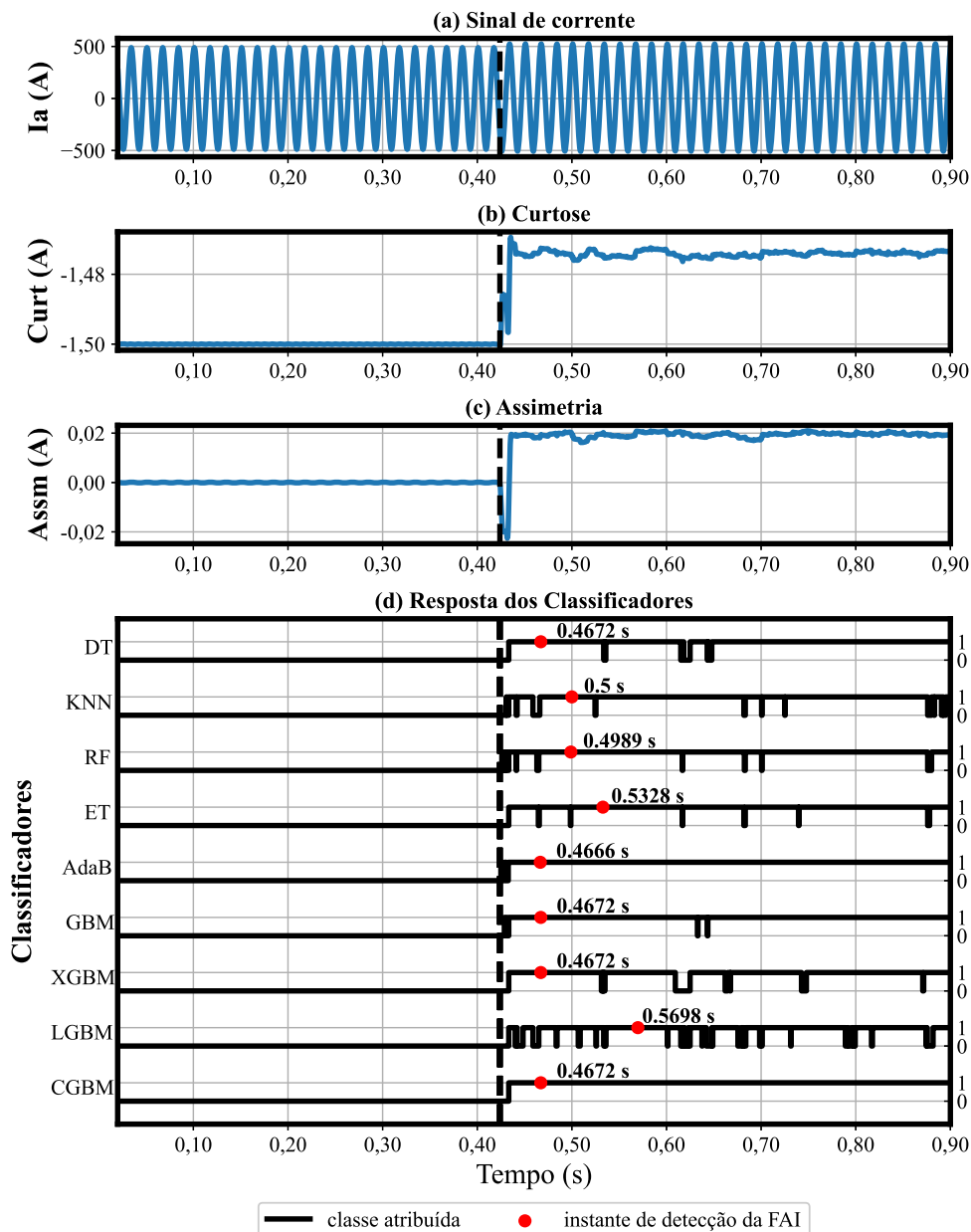
Além da resposta dos classificadores, na Figura, também é apresentado o instante em que houve a detecção da FAI pela metodologia, o qual é representado por um ponto vermelho. Esse instante de detecção da FAI obrigatoriamente deve ocorrer após passados dois ciclos com o estado lógico da resposta dos classificadores como sendo um, o que explica o fato de ainda levar um intervalo de tempo para a ocorrência da detecção, mesmo com a resposta dos modelos de ML se estabilizando.

4.4.1.2 Caso de FAI com o modelo de arco 2

Detalhes da simulação: alimentador urbano, ponto p_3 , incidência em 0,42382 segundo, $V_p = 5824$ V, $R_p = 114 \Omega$, $V_n = 8128,3873$ V, $R_n = 122,6569 \Omega$.

O segundo caso usado para avaliar a operação no tempo da metodologia proposta é a simulação de uma FAI com o modelo de arco 2. Destaca-se que esse modelo de arco se assemelha muito com o anterior, porém simula também o comportamento aleatório que é comumente encontrado na corrente de FAI. Dito isto, a Figura 46(a) apresenta a forma de onda do sinal de corrente para a fase na qual ocorre essa falta.

Figura 46 – Operação no tempo para o modelo de arco 2.



Assim, como pode ser observado, o seu comportamento é bastante semelhante com o já visto para o caso anterior, o que, por sua vez, também pode ser explicado pela soma da corrente de FAI com a corrente drenada pelas demais cargas no sistema, tornando assim menos evidente as características do evento.

Em contrapartida, nas Figuras 46(b) e 46(c), as quais apresentam, respectivamente, o comportamento da curtose e da assimetria amostral, essa diferença é mais perceptível. Isso acontece porque, enquanto no caso anterior a principal diferença era o aumento nos valores desses atributos, no caso atual é bastante perceptível também o comportamento mais oscilatório dos mesmos quando a falta entra em regime.

Esse comportamento mais oscilatório torna a tarefa de classificar as amostras desse modelo de arco ainda mais desafiadora, uma vez que não é fácil delimitar regiões para as classes quando o comportamento do arco varia de forma aleatória. Desse modo, como pode ser visto na Figura 46(d), as respostas dos classificadores em relação a esse caso de FAI foram, no geral, bem menos consistentes do que no caso anterior. Isso pois, mesmo que modelos de ML como DT, AdaB, GBM, XGBM e CGBM tenham permitido que a metodologia detectasse rapidamente a ocorrência da FAI, outros classificadores como KNN, RF, ET e LGBM apresentaram uma resposta bem menos consistente, levando a metodologia à uma demora maior para a realização da detecção da ocorrência da falta.

Nesse sentido, para o caso em questão, o LGBM foi nitidamente o que apresentou o pior comportamento, apresentando uma resposta bastante instável durante toda a ocorrência da falta. Já o oposto é percebido no desempenho do CGBM, o qual classificou os dados corretamente durante quase toda a simulação, sem nenhuma instabilidade em sua resposta.

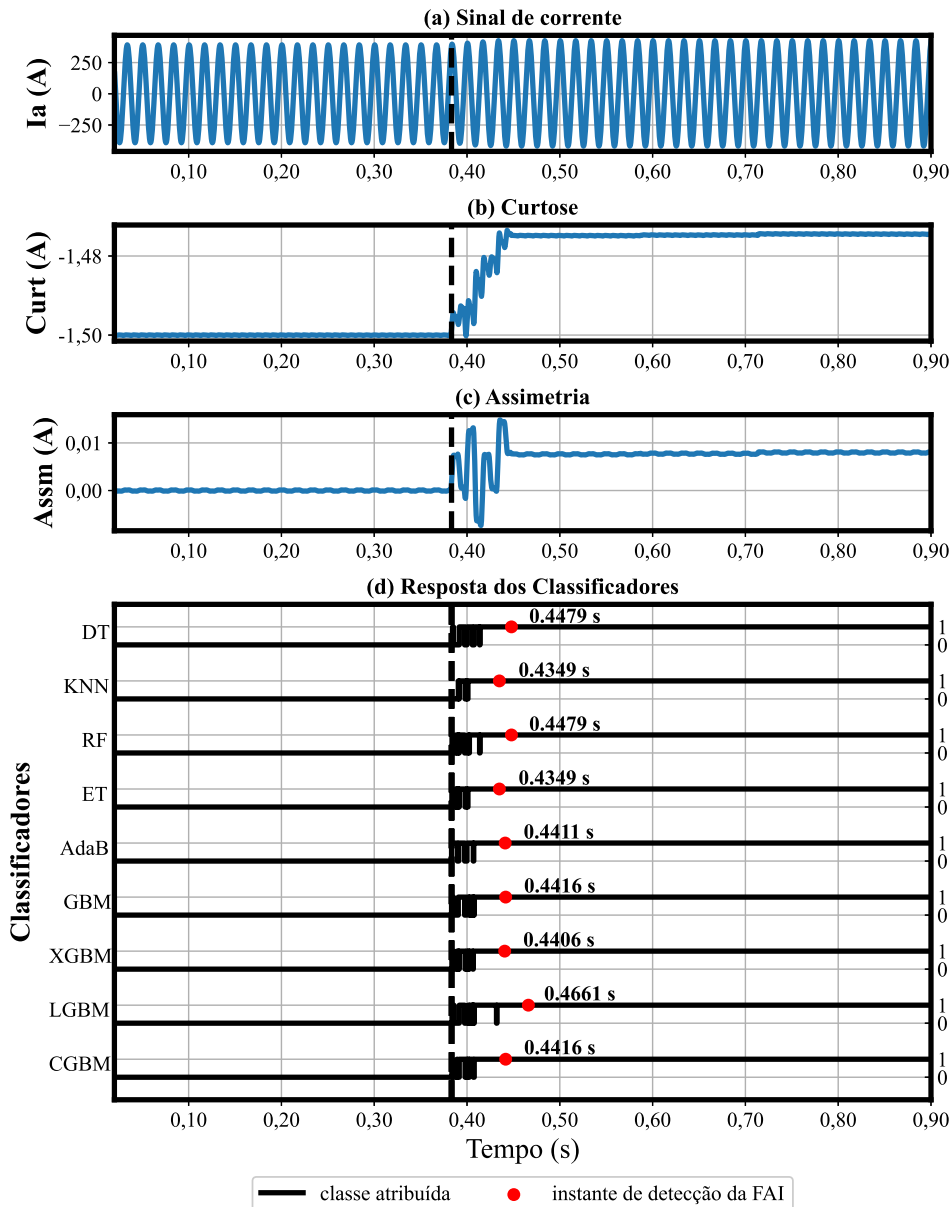
4.4.1.3 Caso de FAI com o modelo de arco 3

Detalhes da simulação: alimentador rural, ponto p_5 , incidência em 0,38287 segundo, $V_1 = 3414$ V, $V_2 = 4552$ V, $V_3 = 4220$ V, $V_4 = 5249$ V, $V_5 = 5190$ V, $V_6 = 6410$ V, $R_1 = 1275$ Ω , $R_2 = 913$ Ω , $R_3 = 1267$ Ω , $R_4 = 1774$ Ω , $R_5 = 1779$ Ω , $R_6 = 1191$ Ω .

No terceiro caso, é usado o modelo de arco 3 para realizar a simulação da FAI. Esse modelo consegue representar algumas características da corrente de FAI já vista nos arcos anteriores, como, por exemplo, assimetria entre os semiciclos e não-linearidade. Entretanto, diferentemente dos casos anteriores, esse modelo de arco consegue representar, também, o comportamento de *buildup* e *shoulder* a partir do chaveamento em sequência de vários modelos de arco 1, os quais fazem parte de sua composição. Dessa forma, a corrente de falta desse arco elétrico apresenta um crescimento gradual de seus valores durante um dado intervalo de tempo, o qual pode ser visto de uma forma bem sucinta na Figura 47(a). Além disso, como pode ser observado nas Figuras 47(b) e 47(c), o comportamento da curtose e da assimetria amostral tende a apresentar um intervalo mais longo de transitório quando comparadas com as dos outros casos, sendo que agora esses dois atributos apresentam muito mais oscilações, cada uma referente a entrada de um dos arcos pertencentes ao modelo, resultando em uma demora maior para que

ambos os filtros converjam. Ademais, o comportamento durante o regime da falta segue bastante semelhante com aquele visualizado nos demais casos.

Figura 47 – Operação no tempo para o caso com o modelo de arco 3.



Fonte: Autor.

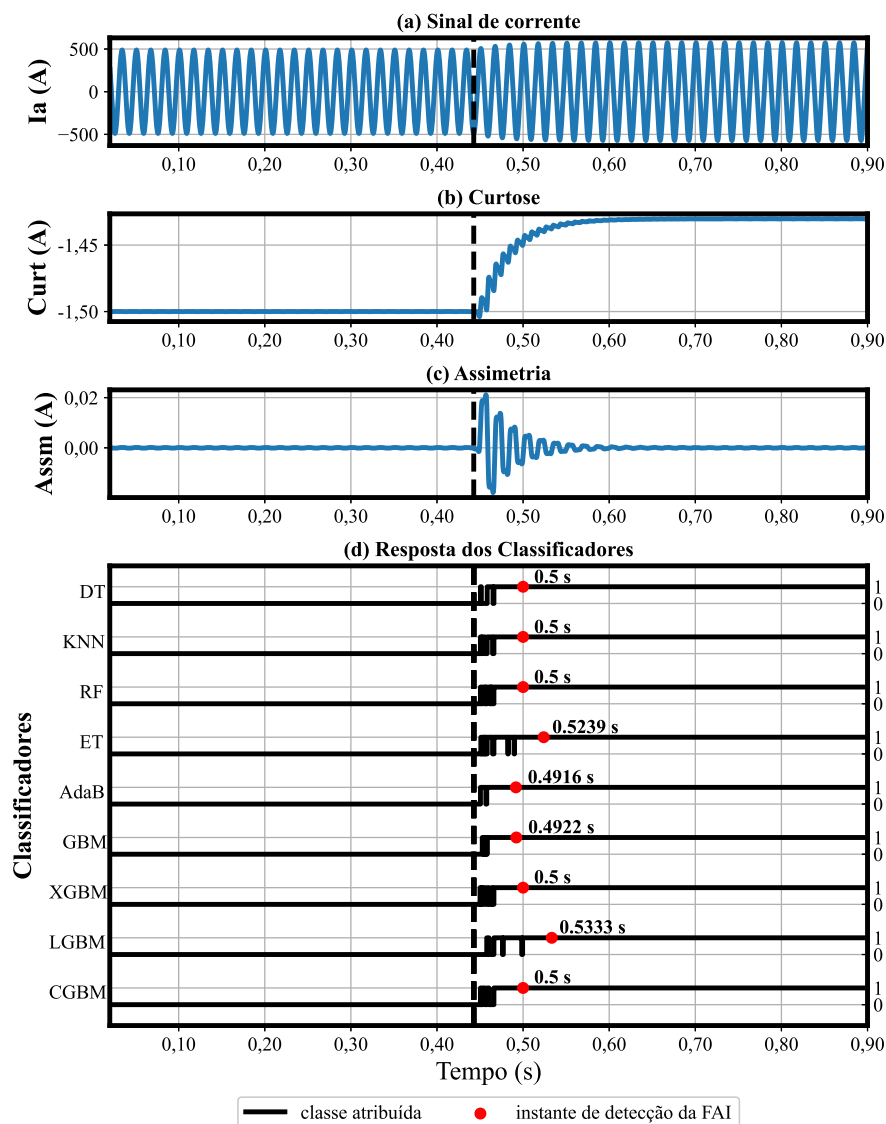
Em relação ao comportamento dos classificadores, todos apresentaram um desempenho de classificação bastante interessante, com uma resposta mais oscilatória apenas durante o intervalo de transitório dos filtros, conforme a Figura 47(d). Nesse caso, todos os classificadores tiveram resultados bastante semelhantes entre si, sendo o LGBM, novamente, aquele que apresentou um desempenho um pouco abaixo dos demais. Além disso, os classificadores KNN e ET foram os que detectaram mais rapidamente a ocorrência de FAI no sistema, apresentando uma estabilização de sua resposta antes mesmo do transitório do filtro terminar.

4.4.1.4 Caso de FAI com o modelo de arco 4

Detalhes da simulação: alimentador rural, ponto p_2 , incidência em 0,44246 segundo, $G_0 = 0,00873$ S, $\tau = 0,03627$ s.

O último caso de FAI a ser apresentado em sua operação no tempo é aquele que usa o modelo de arco 4 para representar o comportamento da falta. Como já discutido anteriormente, esse modelo computacional se diferencia dos demais por não ser resultado do uso de um esquema elétrico baseado em diodos e fontes de tensão em antiparalelo e, sim, de uma modelagem matemática. Nesse contexto, na Figura 48(a), é possível analisar como esse modelo de arco afeta o sinal de corrente do sistema, sendo o seu *buildup* bem perceptível. Além disso, uma das diferenças do *buildup* desse modelo de arco para o 3, é o formato mais arredondado de sua envoltória, em contrapartida ao do caso anterior, o qual apresentava uma forma mais achatada em seu crescimento.

Figura 48 – Operação no tempo para o caso com modelo de arco 4.



Ainda, nas Figuras 48(b) e 48(c), é perceptível a diferença no comportamento da curtose e da assimetria amostral para esse caso, uma vez que, como já citado, a envoltória de seu *buildup* é mais arredondado, gerando assim um comportamento de curtose que segue um padrão semelhante. Já em relação a assimetria amostral, nota-se que, após passar por um período de oscilação durante o transitório do filtro, a mesma tende a voltar para os valores de zero. Esse comportamento pode ser explicado pela ausência da diferença de tamanho entre os semiciclos positivos e negativos para esse modelo de arco elétrico, o que, por sua vez, afeta a distribuição dos valores de corrente presente na janela móvel usada para calcular esse atributo.

Por fim, a Figura 48(d) apresenta o comportamento dos classificadores para esse caso. De forma geral, eles apresentam um comportamento bastante semelhante com aquele já relatado nos demais casos, apresentando um breve intervalo de incerteza durante o transitório do filtro, tendendo a se estabilizar após a convergência dos valores de curtose e assimetria. Além disso, para esse caso, o LGBM, assim como nos anteriores, foi o classificador que resultou em uma maior demora de detecção da FAI pela metodologia. Entretanto, o que detectou no menor intervalo de tempo foi o modelo AdaB. Ademais, os outros classificadores apresentaram performances bastante semelhantes, com tempos de detecção consideravelmente próximos.

4.4.1.5 Caso de entrada de carga resistiva

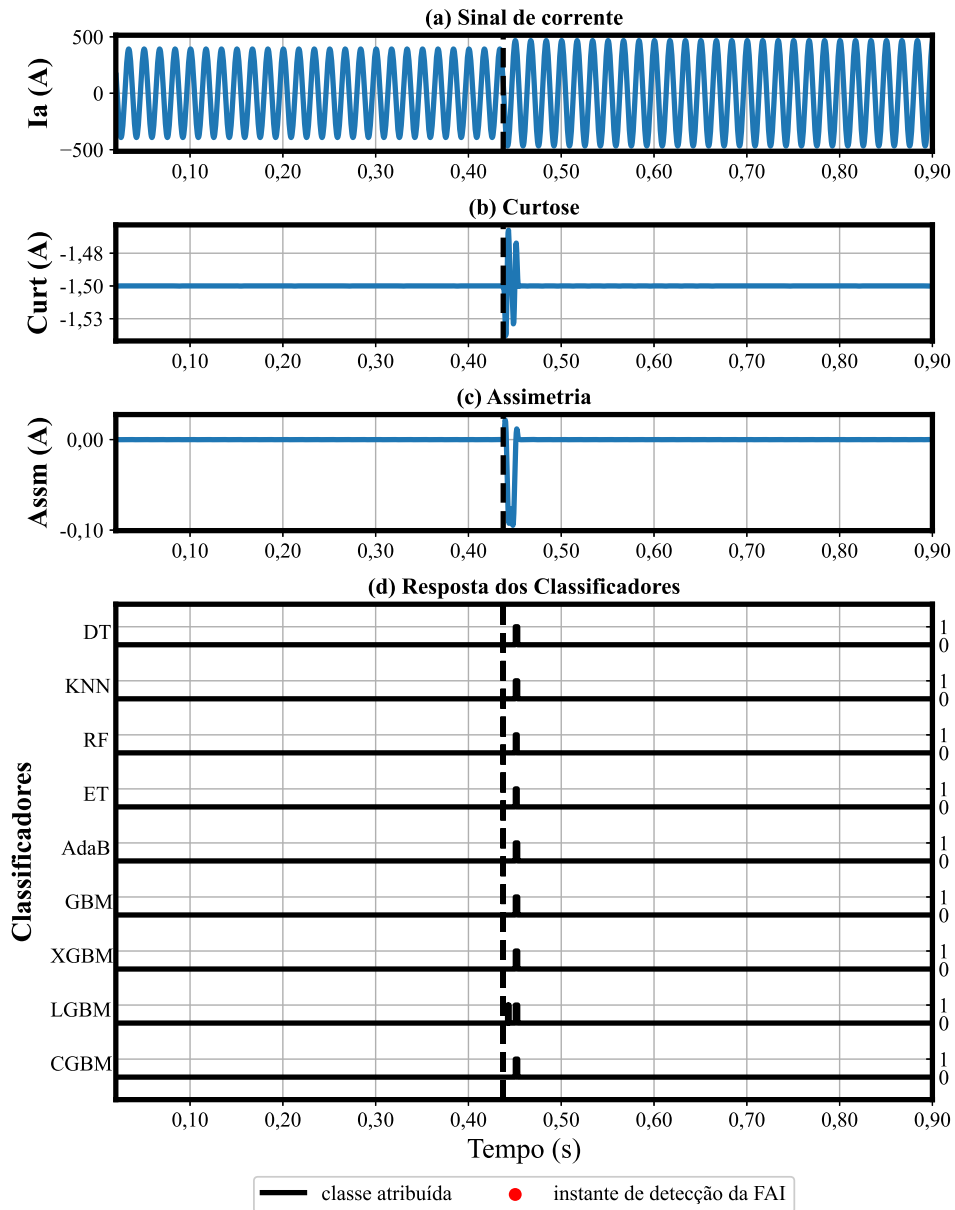
Detalhes da simulação: alimentador rural, ponto p_4 , incidência em 0,43713 segundo, resistência de carga de 142,562 Ω .

O quinto caso analisado envolve a entrada de uma carga resistiva no sistema elétrico de distribuição. A Figura 49(a) apresenta a forma de onda do sinal de corrente da fase onde esse evento acontece. Nesse sentido, assim como nos demais casos analisados, a linha pontilhada vertical representa o instante em que ocorre o chaveamento do evento.

Ressalta-se que esse é um caso interessante porque, visualmente, ele é bastante parecido com a forma de onda vista para os casos dos modelos de FAI os quais não envolviam a presença de *buildup*, como os modelos de arco 1 e 2. Entretanto, embora não fique perceptível na forma de onda tanto desse caso, quanto dos que envolvem a ocorrência de FAI, a entrada de carga não envolve nenhum tipo de distorção da forma de onda do sinal de corrente em regime permanente e nem apresenta assimetria entre os seus semiciclos, o que obviamente afeta o comportamento dos atributos extraídos desse sinal. Esse fato é comprovado nas Figuras 49(b) e 49(c), as quais apresentam a curtose e a assimetria amostral antes e depois do chaveamento dessa carga. Nesse contexto, observa-se que ambas as características apresentam uma rápida oscilação em seus comportamentos durante o transitório, rapidamente voltando para os mesmos valores de pré-falta após essa ocorrência.

Desse modo, se existe um momento em que os classificadores podem realizar uma classificação errada, esse instante é durante o transitório dessas duas características. Essa resposta dos classificadores pode ser vista na Figura 49(d), na qual, como já considerado, os classificadores apresentam uma dificuldade em classificar corretamente os dados somente nos instantes iniciais

Figura 49 – Operação no tempo para o chaveamento de carga resistiva.



Fonte: Autor.

do chaveamento. Entretanto, como esse intervalo é consideravelmente curto, o número de amostras classificadas inadequadamente não é o suficiente para que haja uma detecção errada por parte da metodologia.

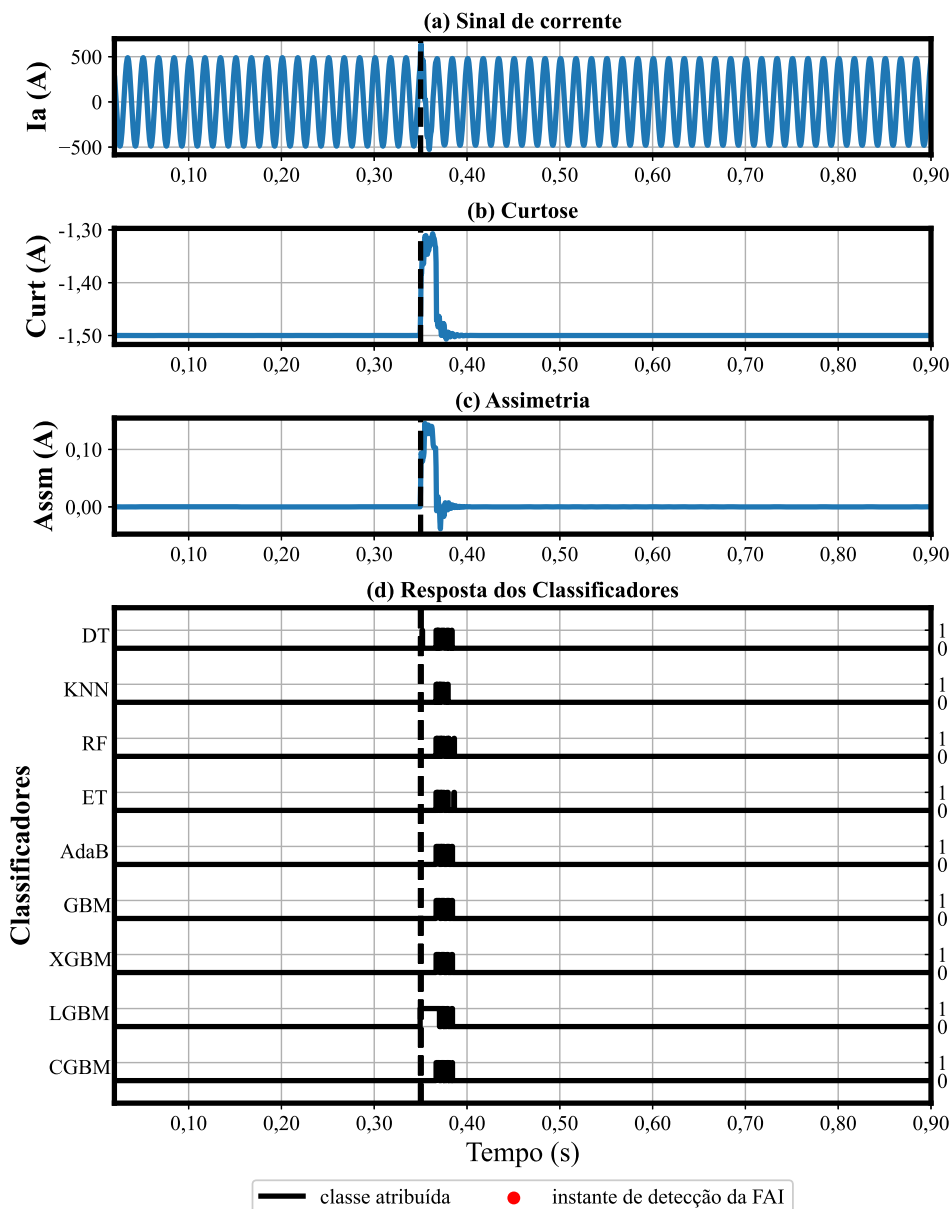
4.4.1.6 Caso de entrada de carga capacitiva

Detalhes da simulação: alimentador urbano, ponto p_3 , incidência em 0,34965 segundo, susceptância da carga de $2815 \mu S$, resistência associada de $2,98351 \Omega$.

O sexto caso é referente a entrada de uma carga capacitiva no sistema de distribuição. Nesse contexto, percebe-se, na Figura 50(a), que o efeito visível na forma de onda do sinal de corrente dura apenas alguns instantes, sendo o seu resultado apenas uma deformação nesse

sinal durante esse intervalo de tempo. Esse comportamento é um pouco diferente daquele visto na entrada de carga puramente resistiva no sistema, cuja a mudança nos valores de amplitude da corrente permaneciam mesmo depois de ocorrido o transitório, sendo que, aqui, os efeitos permanecem, porém, não de forma tão visível como o aumento de carga do caso anterior, por exemplo. Além disso, analisando as Figuras 50(b) e 50(c), é possível observar o rápido crescimento dos valores de curtose e assimetria obtidos, o qual acontece de forma bastante semelhante para ambas as características, seguido de uma queda até os atributos voltarem aos valores antes dos chaveamento. Entretanto, embora o comportamento para esse caso também tenha um efeito temporário na curtose e assimetria amostral, o tempo de duração desse efeito é maior do que o visto para a entrada de carga resistiva.

Figura 50 – Operação no tempo para o chaveamento de carga capacitiva.



Dessa forma, observando as respostas de cada um dos classificadores apresentadas na Figura 50(d), constata-se que os mesmos erram a classificação de um número maior de amostras para esse caso, com destaque para o comportamento da classificador LGBM, o qual chega a classificar as amostras como sendo de FAI ininterruptamente por um tempo considerável. Mesmo assim, independente da dificuldade relativamente maior de classificar as amostras do sinal de corrente desse caso em relação ao outro, ainda assim não foi o suficiente para que houvesse uma atuação equivocada da metodologia em relação a esse caso, o que continua sendo um indicador positivo a respeito do funcionamento da metodologia.

Além disso, é importante destacar que, para todos os classificadores, exceto o LGBM, embora tenha ocorrido um número considerável de erros a mais em comparação com o caso anterior, esses erros foram intermitentes. Ou seja, os dados classificados erroneamente eram seguidos por dados corretamente classificados, o que gerou oscilações nas respostas dos classificadores. Dessa forma, a contagem era constantemente reiniciada, impedindo que os contadores alcançassem valores equivalentes aos dois ciclos consecutivos de dados classificados como FAI, necessários para a detecção. Com isso, pode-se observar a robustez da metodologia frente a cenários mais desafiadores, sendo essa robustez atribuída ao critério de classificações sucessivas de amostras de FAI. Isso ocorre porque, assim como nesse caso de chaveamento de capacitores, embora algumas amostras possam invadir a região de classificação de FAI, isso não acontece de forma contínua, permitindo distinguir esses casos mais complexos dos de FAI.

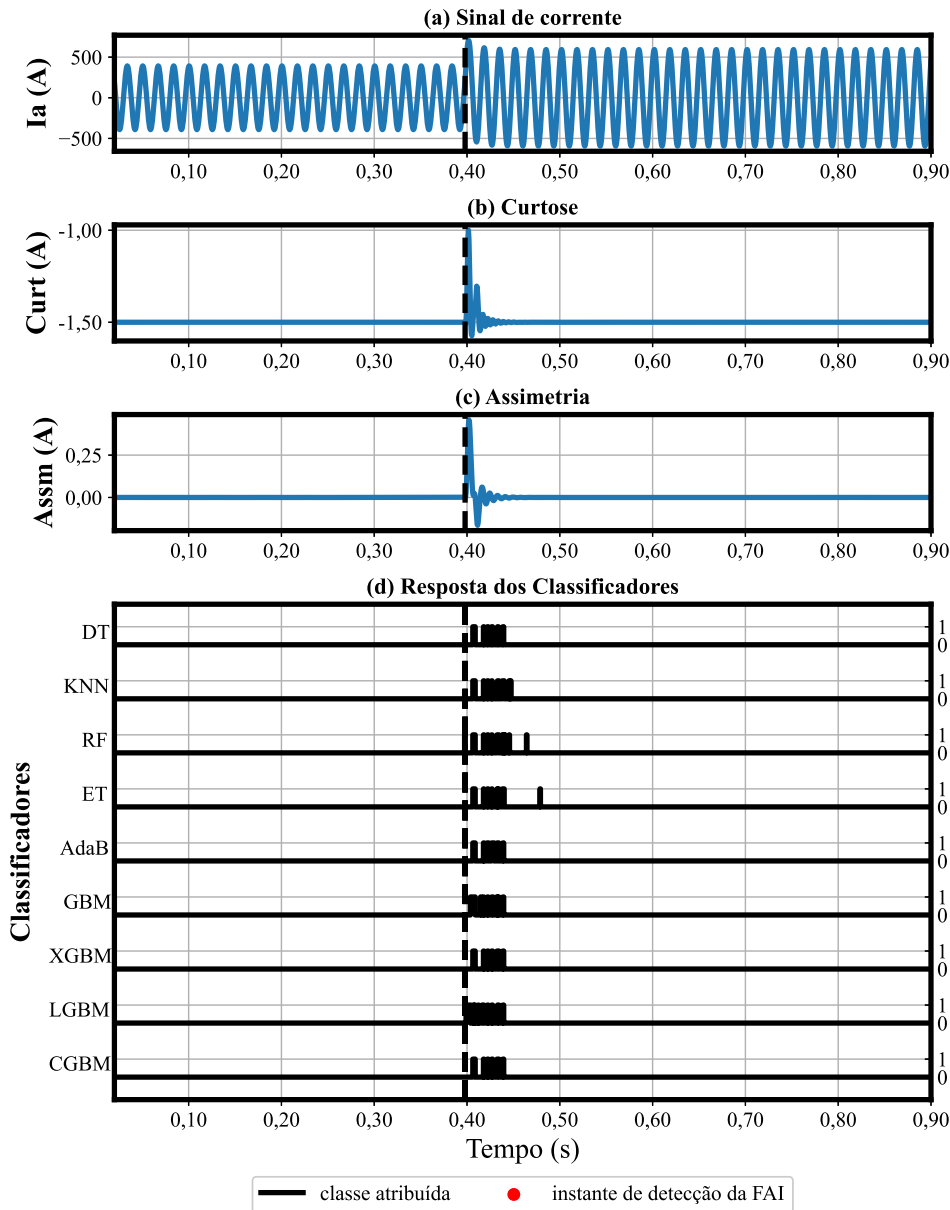
4.4.1.7 Caso de entrada de carga indutiva

Detalhes da simulação: alimentador rural, ponto p_6 , incidência em 0,39782 segundo, reatância indutiva de 26,584 Ω , resistência associada de 2,25698 Ω .

Já o sétimo caso é relativo a entrada de uma carga indutiva no sistema analisado. A Figura 51 apresenta o comportamento da forma de onda do sinal de corrente antes e após o chaveamento dessa carga. Assim, rapidamente é possível identificar que esse evento apresenta envoltórias consideravelmente diferentes daquelas vistas até então, uma vez que a corrente resultante do mesmo apresenta uma componente exponencial, com decaimento proporcional à constante X/R do sistema após a entrada dessa carga. Entretanto, apesar do comportamento distinto desse evento, nas Figuras 51(b) e 51(c), o comportamento relativo a curtose e assimetria amostral se demonstrou bastante semelhante com aquele visto para os demais casos. Porém, com a diferença de ter um decaimento mais oscilatório, ou seja, transcorrido o instante inicial do chaveamento, o decaimento não é abrupto, ocorrendo a partir de oscilações cada vez menores para ambos os atributos analisados.

Já em relação a resposta dos classificadores, observa-se, na Figura 51(d), que os mesmos apresentam ainda mais erros do que nos dois casos anteriores. No entanto, como já foi discutido no caso anterior, embora a sequência de amostras classificadas erradamente seja maior do que o visto anteriormente, isso ocorreu de forma intervalada, impedindo, assim, que a contagem conseguisse avançar o suficiente para que houvesse qualquer tipo de falsa detecção de FAI.

Figura 51 – Operação no tempo para o chaveamento de carga indutiva.



Fonte: Autor.

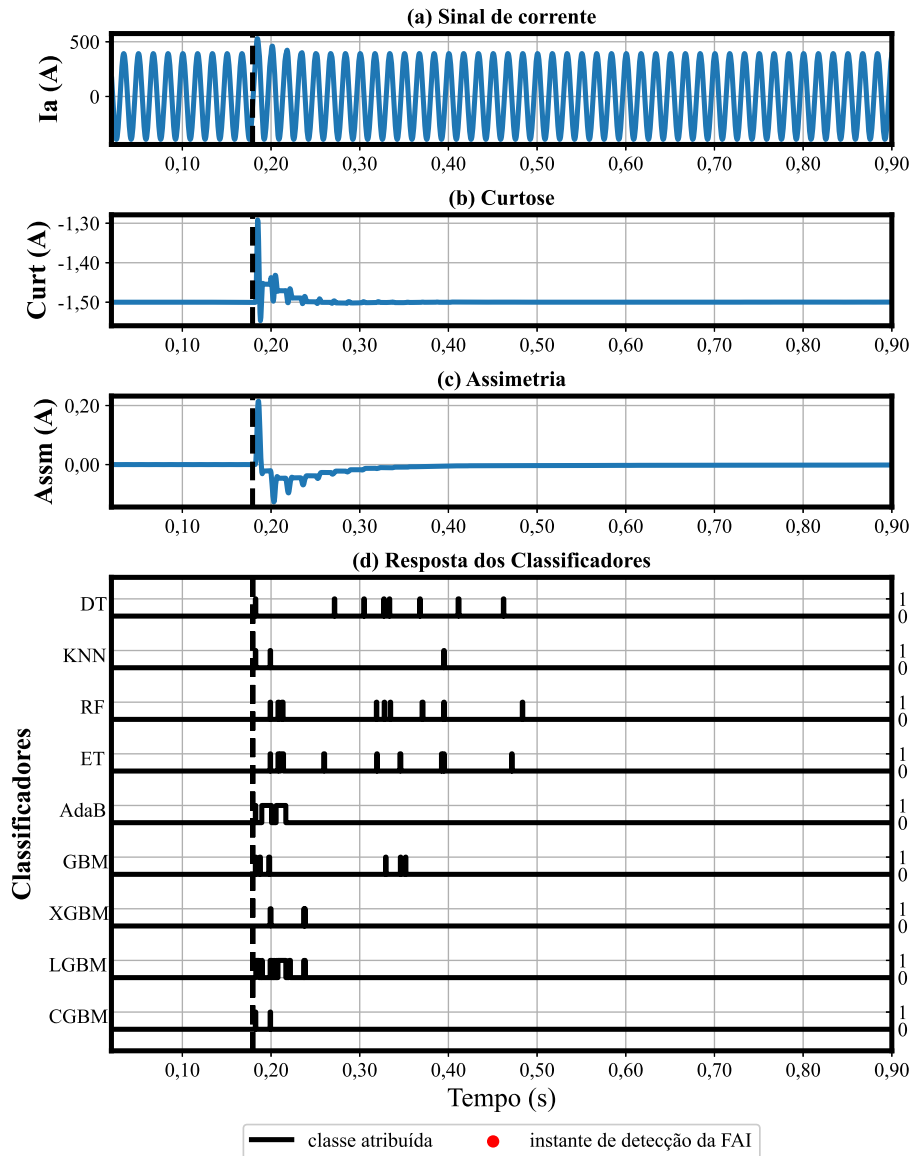
4.4.1.8 Caso de energização de transformador

Detalhes da simulação: alimentador rural, ponto p_6 , incidência em 0,1787 segundo, número de transformadores energizados no trecho igual a 11 transformadores.

O oitavo caso é relativo à uma situação de energização de vários transformadores em um determinado trecho do sistema analisado. Esse caso apresenta um comportamento bastante semelhante com aquele visto no caso anterior, como apresentado na Figura 52(a), uma vez que a energização de transformadores é um tipo de carga indutiva sendo chaveada no sistema elétrico de potência, com a diferença que, devido à questões construtivas desse tipo de elemento, o processo de energização de transformadores apresenta um conteúdo harmônico um pouco

diferente daquele visto para o caso anterior.

Figura 52 – Operação no tempo para o caso de energização de transformadores.



Fonte: Autor.

Já as Figuras 52(b) e 52(c) mostram que a curtose e assimetria se diferem bastante daquelas vistas no chaveamento de carga indutiva, uma vez que, aqui, esse decaimento tende a não se dar de uma forma tão oscilatória, além de ter rápidos crescimentos e decrescimentos nesses valores conforme esses atributos tendem a se estabilizar, o que normalmente acontece quando a curtose e assimetria voltam para os valores encontrados antes da energização dos transformadores. Ademais, em relação ao comportamento da resposta dos classificadores para as diferentes amostras desse evento, a Figura 52(d), mostra como cada modelo se comportou diante desse cenário. Nesse caso, ao contrário dos demais, as classificações erradas se deram de forma bem mais espaçada e em uma quantidade bem menor do que a aquelas vistas anteriormente, com ressalvas apenas em relação às respostas do AdaB e do LGBM, as quais apresentaram alguns

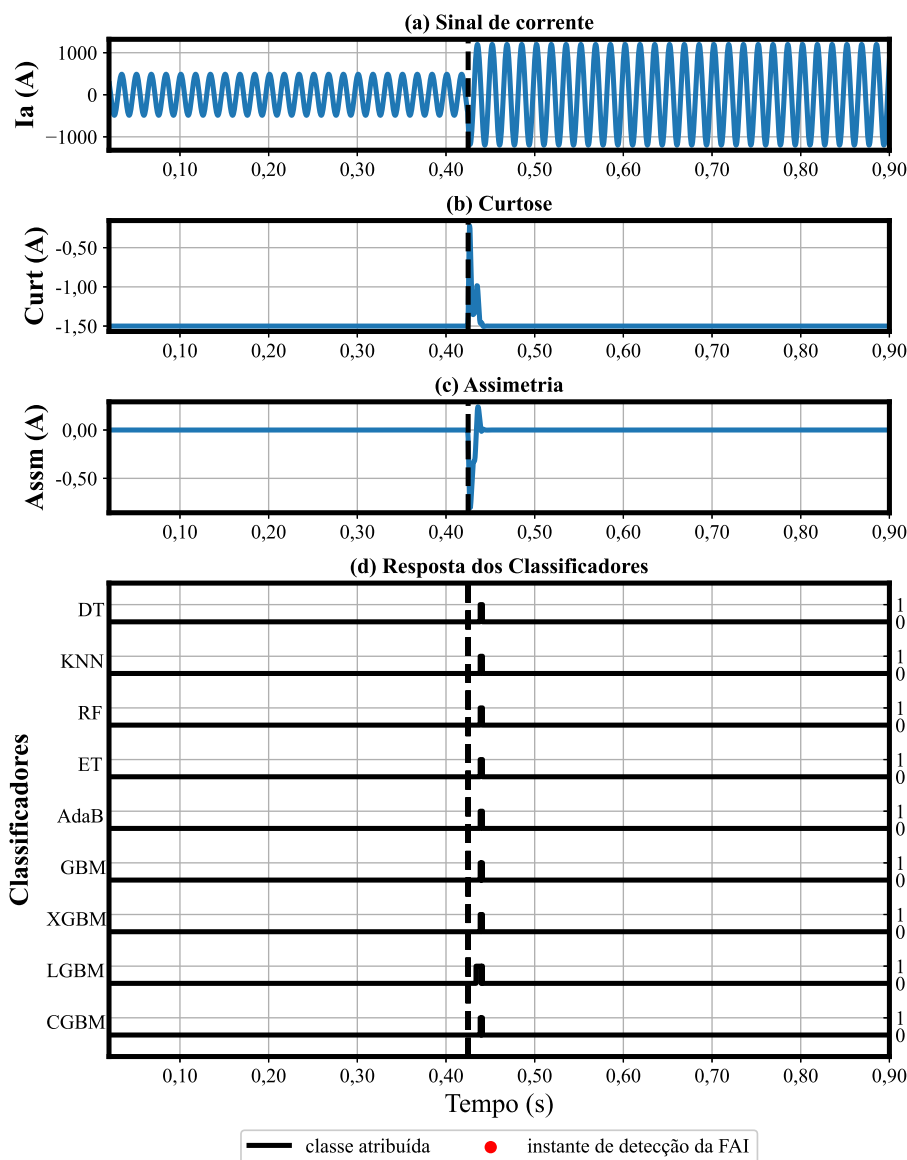
intervalos de classificações ininterruptas de dados como sendo de FAI. Entretanto, assim como visto no demais casos, não houve nenhuma detecção errada de FAI por parte da metodologia.

4.4.1.9 Caso de curto-circuito

Detalhes da simulação: alimentador urbano, ponto p_2 , incidência em 0,42454 segundo, curto-circuito bifásico-terra entre as fases C e A, resistência de falta de 9,58192 Ω .

As respostas do modelo podem ser vista na Figura 53, onde, em (a), é apresentada a forma de onda resultante desta falta para a corrente da fase A.

Figura 53 – Operação no tempo para o caso de curto-circuito bifásico-terra.



Fonte: Autor.

Esse caso se assemelha bastante com o comportamento visto no caso que envolvia o chaveamento de uma carga puramente resistiva, com a diferença que a resistência desse tipo de problema é consideravelmente menor do que a vista anteriormente, resultando em valores bem

maiores de corrente de pós-falta quando comparada com os valores de pré-falta. O comportamento da curtose e da assimetria pode ser visto nas Figuras 53(b) e 53(c) e são basicamente picos no sentido positivo, no caso da curtose, e picos no sentido negativo, no caso da assimetria amostral.

Por fim, a resposta dos classificadores para esse caso é bastante semelhante com aquela vista para a entrada de carga, uma vez que as saídas altas dos classificadores ocorrem em um intervalo específico, o qual se dá durante os picos já mencionados, e passam longe de serem o suficientes para que ocorra a falsa detecção de um FAI pela metodologia.

4.4.2 Métricas Gerais

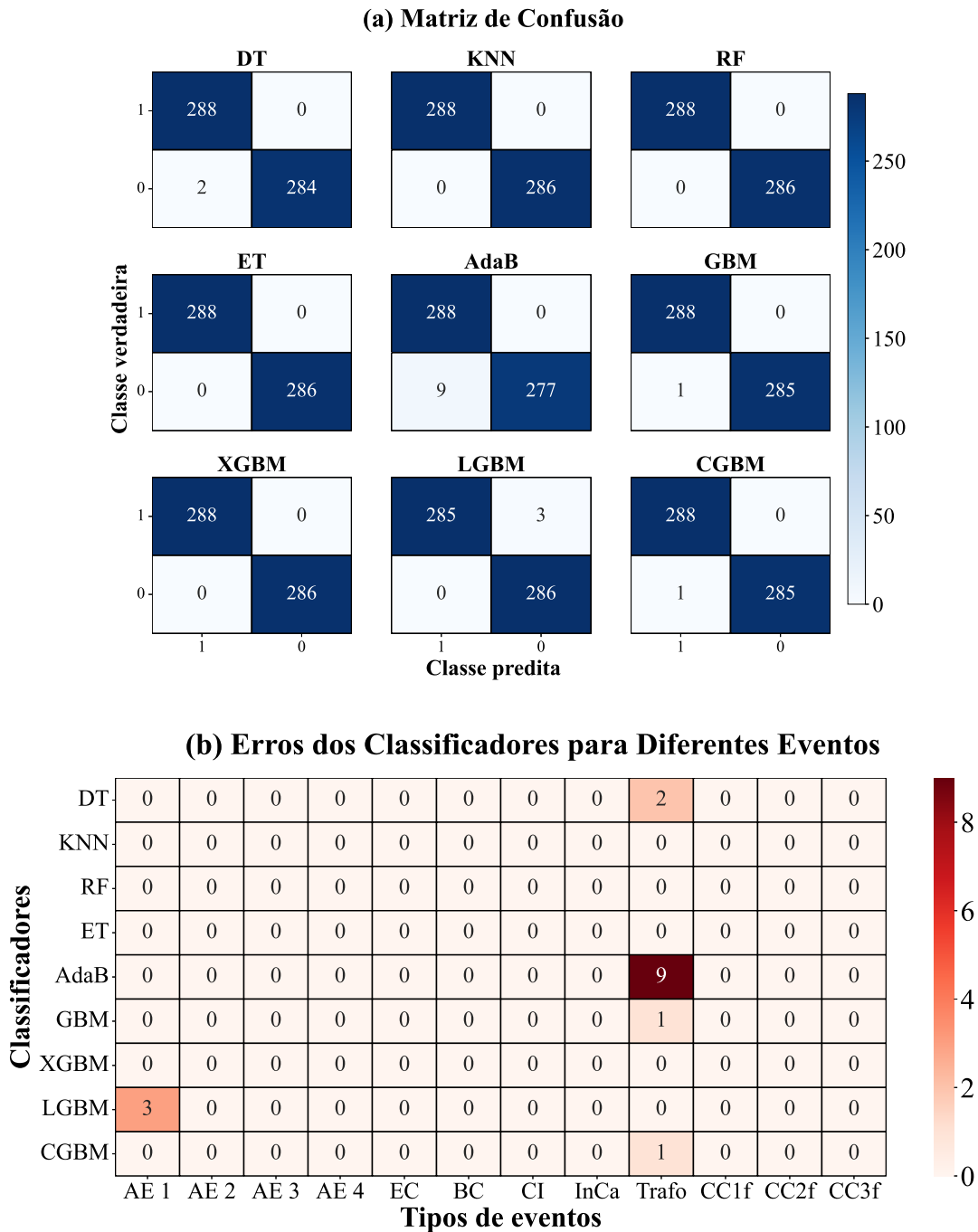
Após o funcionamento da metodologia ter sido analisado em detalhe para os nove casos apresentados, agora é apresentada a análise do desempenho da mesma para todo o banco de teste. Para isso, a mesma lógica apresentada anteriormente foi aplicada em cada um dos casos que compõem o banco de teste e os resultados de sua atuação foram armazenados a partir de quatro contadores, os quais já foram discutidos mais a fundo nesse mesmo Capítulo, sendo esses: *CD* (casos de FAI detectados); *ID* (demais eventos incorretamente identificados como FAI); *ND* (caso de FAI não detectados) e *DE* (demais eventos para os quais a metodologia se comportou corretamente e não atuou). A partir do uso destas contagens, é possível formar a matriz de confusão, a qual nada mais é que um recurso gráfico para facilitar a análise do desempenho da metodologia com base nesses parâmetros.

As matrizes de confusão referentes a aplicação da metodologia com cada um dos classificadores no banco de casos de teste são apresentadas na Figura 54(a). Nesse sentido, vale lembrar que o número de casos de FAI presentes no conjunto de teste é de 288 casos e de demais eventos do sistema elétrico de distribuição é de 286 casos. Assim, como pode ser observado na referida Figura, a metodologia alcançou as métricas máximas em seu desempenho quando a mesma foi utilizada com base na classificação de 4 modelos de ML, sendo estes KNN, RF, ET e XGBM. Outros classificadores apresentaram contagens um pouco inferiores, com a DT apresentando dois casos de *ID*, o GBM juntamente com o CGBM apresentando 1 caso de *ID* e o LGBM com 3 casos de *ND*. Além destes, o classificador AdaB se destacou por apresentar 9 casos de *ID*, demonstrando que a mesma foi a que mais confundiu eventos comuns do sistema de distribuição como sendo FAI.

De modo que fosse possível observar de forma mais detalhada quais eventos causaram mais dificuldade para a metodologia, a Figura 54(b) apresenta o número de erros relativos a cada um dos tipos de eventos simulados. Dessa forma, como pode ser visto, os casos envolvendo energização de transformadores foram os responsáveis por todos os casos de *ID* apresentados na matriz de confusão. Além deste evento, outro que se destaca é a FAI utilizando o modelo de arco 1, o qual apresentou 3 casos de *ND*, quando a metodologia tomava como base as classificações de dados do LGBM para realizar a sua detecção.

A partir das contagens apresentadas anteriormente, é possível calcular as métricas de desempenho da metodologia para diferentes classificadores. Dessa forma, a Tabela 10 apresenta

Figura 54 – Matriz de confusão e erros dos classificadores.



Fonte: Autor.

o resultado do cálculo das métricas *Accuracy (A)*, *Dependability (D)*, *Security (S)*, *Safety (SF)* e *Sensibility (SN)*, conforme já explicado na Seção 4.3.

Analisando os resultados nessa Tabela, percebe-se que todos os classificadores apresentaram métricas acima de 96%, o que indica que todos tiveram um bom desempenho diante do problema proposto. Além disso, com base nos resultados observados, é possível destacar os seguintes pontos:

Tabela 10 – Métricas de desempenho dos classificadores.

Classificador	A(%)	D(%)	S(%)	SF(%)	SN(%)
DT	99,652	100	99,301	100	99,310
KNN	100	100	100	100	100
RF	100	100	100	100	100
ET	100	100	100	100	100
AdaBoost	98,432	100	96,853	100	96,970
GBM	99,826	100	99,650	100	99,654
XGBM	100	100	100	100	100
LGBM	99,477	98,958	100	98,962	100
CGBM	99,826	100	99,650	100	99,654

Fonte: Autor.

- KNN, RF, ET e XGBM apresentaram 100% de desempenho em todas as métricas;
- GBM e CGBM compõe o segundo grupo de classificadores que melhor performaram em termos de métricas em relação ao banco de dados analisado, com métricas acima de 99,6%;
- DT encontra-se na terceira posição do ranking de melhores classificadores, obtendo 99,301 % de desempenho para a métricas *Security (S)*;
- O LGBM é o único que apresentou dificuldade em detectar casos de FAI, mas, por outro lado, não realizou nenhuma detecção incorreta. Assim, apresentou queda nas métricas *Dependability (D)* e *Safety (SF)* para até 98,958 %, colocando o mesmo na quarta posição dentre os melhores classificadores; e
- AdaB foi o classificador que resultou nas piores métricas da metodologia para o banco de dados de teste quando comparado com os demais, tendo um desempenho de 96,853 % para a métrica *Security (S)* .

Pode-se calcular a média dos resultados para cada uma das métricas, de modo que seja possível analisar o desempenho dos classificadores não-lineares como um todo para a tarefa de detecção de FAI. Os resultados desses cálculos são apresentados em porcentagem na Tabela 11.

Tabela 11 – Métricas de desempenho médias englobando todos os classificadores.

Accuracy	Dependability	Security	Safety	Sensibility
99,6903	99,8843	99,4949	99,8847	99,5098

Fonte: Autor.

Os resultados presentes na Tabela 11 demonstram que a métrica que mais foi afetada por erros dos classificadores foi a *Security*, a qual está diretamente relacionada com a capacidade da metodologia de não atuar dado um evento no sistema elétrico de potência que não seja uma FAI. A segunda métrica mais afetada para essa situação é a *Sensibility*, responsável por quantificar o quão confiável é uma detecção de FAI dada pela metodologia. Entretanto, é importante ressaltar

que essa é uma observação geral, sendo que, como já foi discutido antes, existem classificadores que alcançaram o desempenho máximo em todas as métricas.

Analisado o desempenho da metodologia com base na resposta de diferentes classificadores, outra questão importante de ser observada é o tempo de detecção da mesma. Dessa forma, na Figura 55(a), é apresentado o número de casos detectados em relação ao intervalo de tempo transcorrido após a incidência do transitório. Nessa Figura, cada uma das curvas é relativa a um dos classificadores usados, sendo a mesma atravessada por duas linhas pontilhadas, uma vertical e outra horizontal. Assim, a linha vertical é responsável por demarcar o tempo mínimo de detecção, ou seja, o tempo que ocorre exatamente dois ciclos após o início do transitório, o qual só poderia ser satisfeito por um caso onde já, desde o início da falta, seus dados fossem classificados como sendo pertencentes à FAI de maneira ininterrupta. Entretanto, essa situação descrita dificilmente acontece, uma vez que existe um intervalo de convergência do filtro durante o transitório, dificultando bastante o trabalho dos modelos de ML em classificarem corretamente os dados. Logo, o mais esperado é que o número de detecções comece a crescer a partir de já passado um tempo depois desse intervalo de convergência, como pode ser visto na Figura 55.

Já a linha horizontal é responsável por demarcar o número máximo de detecções realizada pelo classificador que mais conseguiu detectar corretamente os casos de FAI. Como apresentado na referida Figura, esse número é de 288 casos e é alcançado pela maioria dos classificadores. Além disso, em relação ao comportamento das curvas apresentadas, observa-se que elas vão crescendo gradualmente conforme o tempo vai passando, sendo este um comportamento já esperado, uma vez que quanto maior o tempo, mais chance a metodologia tem de conseguir identificar a ocorrência de uma falta de alta impedância.

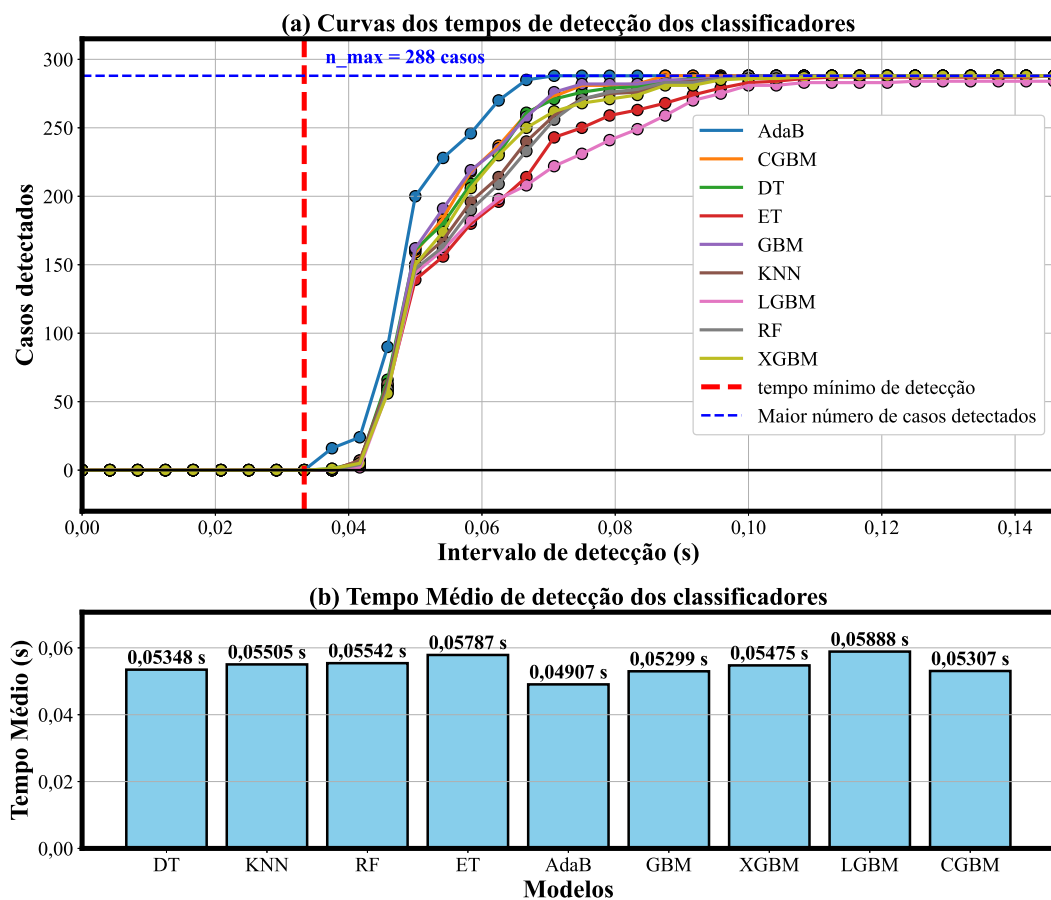
Outra questão importante, é que esse gráfico também é capaz de representar quais dos classificadores estão conseguindo detectar mais ocorrências de FAI em menos tempo, uma vez que, quanto mais rápido for o classificador em detectar os casos de FAI, mais acentuada é a inclinação da sua curva. Assim, com base nesse fato, é possível comparar os classificadores entre si e acompanhar como foi o histórico de detecções conforme o tempo foi passando após a falta. Com isso, algumas curvas se destacam em meio a todas as apresentadas no gráfico, sendo elas a do AdaB, LGBM e ET. No caso do classificador AdaB, o que se destaca é a sua capacidade de detectar, no mesmo intervalo de tempo, mais casos do que os demais classificadores analisados. Já as curvas do LGBM e do ET se destacam por apresentarem uma desaceleração de suas detecções a partir dos 0,05 segundos após iniciada a falta, tornando-os mais lentos em suas detecções caso já passado uma certa quantidade de casos.

A Figura 55(b) apresenta o tempo médio de detecção de FAI pela metodologia para cada um dos classificadores analisados. Essa figura tem o objetivo de complementar as informações apresentadas no gráfico anterior, mas agora, ao invés de apresentar o comportamento do número de detecções ao longo do tempo, essa Figura fornece uma noção mais direta da performance geral dos classificadores. Assim, uma primeira percepção que pode ser obtida é que os tempos de detecção dos classificadores são muito próximos entre si. Dessa forma, mesmo que um seja mais

lento ou mais rápido, essa diferença tende a ser mínima. Mesmo assim, ainda é possível comparar os classificadores e definir, em termos médios, qual é mais rápido e mais lento em detectar a FAI. Nesse sentido, como pode ser observado na Figura, a sequência dos classificadores do mais rápido para o mais lento é: AdaB, GBM, CGBM, DT, XGBM, KNN, RF, ET e LGBM.

Com isso, analisados os resultados envolvendo as métricas de desempenho dos classificadores e também o comportamento do tempo de detecção, é possível construir a Tabela 12, a qual apresenta o número de casos errados pela metodologia e a posição no *ranking* dos tempos médios.

Figura 55 – Tempos de detecção dos classificadores para os casos de FAI.



Fonte: Autor.

Assim, existem algumas informações importantes que podem ser extraídas da Tabela 12. A primeira é que os classificadores que não apresentaram nenhum erro no banco de teste, também demoraram mais para realizar as detecções. Outra informação é que o LGBM foi um dos que mais errou casos e também foi o que pior performou na média dos tempos de detecção das FAIs. Por sua vez, o AdaB realizou as detecções mais rapidamente, mas também foi o que mais apresentou erros na sua detecção. Por fim, os classificadores GBM e CGBM não conseguiram ser os primeiros em nenhum dos quesitos, porém, conseguiram ir bem em ambos os itens apresentados nessa Tabela.

Tabela 12 – Número de classificações erradas e *ranking* dos melhores tempos médios de detecção das FAIs.

Classificadores	Número de Erros	Posição
DT	2	4
KNN	0	6
RF	0	7
ET	0	8
AdaB	9	1
GBM	1	2
XGBM	0	5
LGBM	3	9
CGBM	1	3

Fonte: Autor.

4.5 Métricas Gerais do Banco de Teste com Adição de Ruído

As análises realizadas até então foram feitas em cima dos resultados obtidos pela aplicação da metodologia proposta no banco de teste sem ruído. Entretanto, para a aplicações no mundo real, é importante considerar a influência de diferentes níveis de ruído no desempenho da metodologia para os diferentes classificadores utilizados pela mesma. Assim, de modo a compreender como que a performance da metodologia decai conforme o ruído no conjunto de teste aumenta, são analisados diferentes valores de SNR. Esses valores analisados são de 20 dB até 100 dB, com passos de 5 dB. Para isso, a metodologia foi aplicada nos cenários de ruído mencionados e os resultados relativos a diferentes parâmetros observados foram obtidos.

A primeira análise a ser feita é relativa ao efeito que o aumento dos níveis de ruído tem na quantidade de erros da metodologia. Sendo assim, a Figura 56(a) apresenta os erros de forma geral, considerando tanto aqueles que ocorrem em relação ao casos de FAI, quanto aqueles que envolvem os demais tipos de eventos simulados no sistema teste. Nesse sentido, é possível observar que, ao longo do intervalo de 50 dB até 100 dB, não é constatado nenhum aumento considerável na quantidade de erros da metodologia, para qualquer um dos classificadores analisados. Entretanto, para valores de SNR menores ou iguais a 45 dB, o número de erros da metodologia começa a aumentar consideravelmente, acontecendo praticamente de forma linear para a maioria dos classificadores. Nesse sentido, um classificador que apresenta um comportamento distinto dos demais é o AdaB, o qual, além de resultar em maior número de erros da metodologia quando utilizado com a mesma, também apresenta um intervalo de crescimento dos seus erros, seguido por uma queda logo em seguida, sendo que essa transição acontece no o valor de 30 dB.

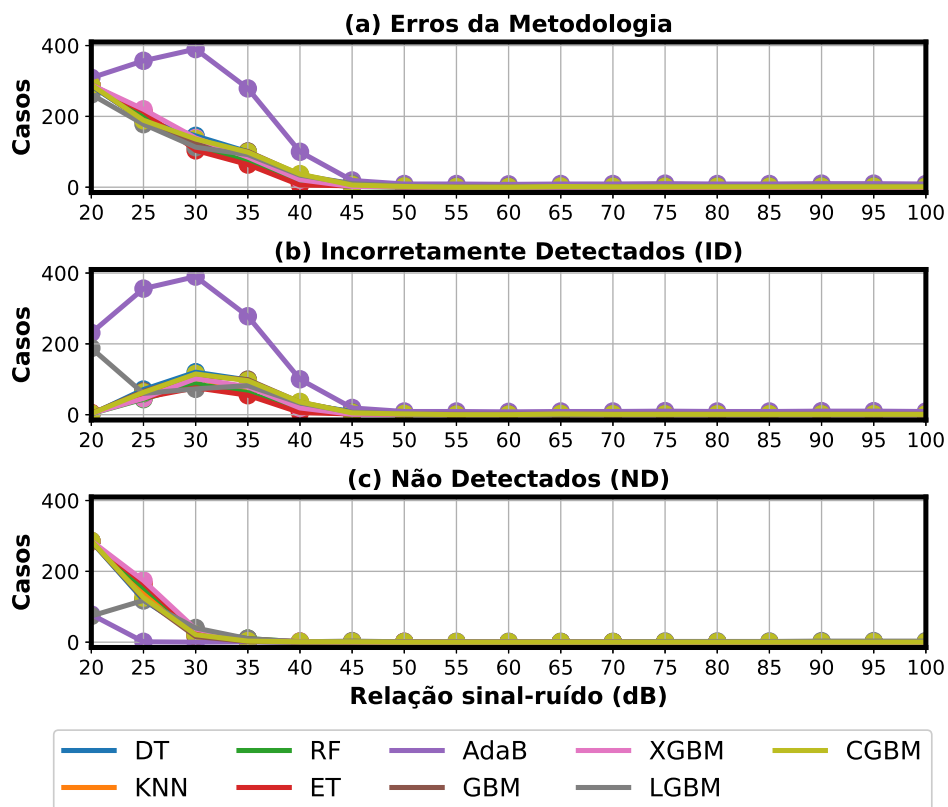
Na Figura 56(b), é apresentado o número de casos em que a metodologia identificou o caso incorretamente como sendo uma ocorrência de FAI. Nessa Figura, assim como na anterior, é possível perceber que a metodologia tende a não apresentar um número consideravelmente pequeno de erros até 45 dB, sendo que, abaixo desse valor, o número de casos classificados incorretamente como sendo de FAI cresce significativamente, onde o AdaB é novamente aquele

que apresenta o pior desempenho dentre os classificadores utilizados.

Já a Figura 56(c) apresenta somente o número de casos de FAI não detectados pela metodologia. Com base nisso, é possível observar que, diferentemente da Figura anterior, a qual começa a apresentar um certo crescimento de sua quantidade de erros a partir de 45 dB, para esse tipo de erro, os valores só começam a crescer a partir de 35 dB. Além disso, também diferente do ocorrido no caso anterior, onde o AdaB apresentou o maior número de erros, o seu desempenho acabou sendo oposto, inclusive com este sendo o classificador que apresentou a menor quantidade de erros quando analisado apenas os casos de FAI não detectados.

Agora, analisando os comportamentos dos erros da metodologia para cada um dos classificadores, é possível perceber que o AdaB, por exemplo, alcançou valores acima de 286 casos, chegando inclusive perto de 400.

Figura 56 – Comportamento dos erros de detecção da metodologia para os diferentes classificadores em diversos níveis de ruído.



Fonte: Autor.

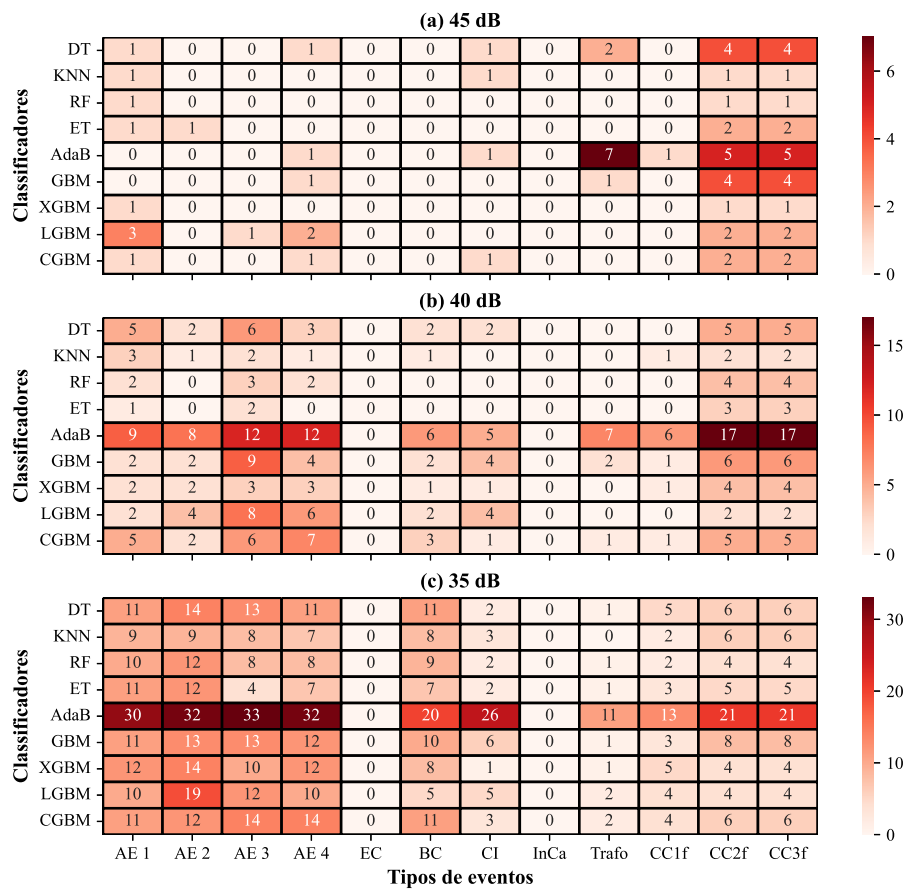
Isso ocorre porque a lógica utilizada para definir um determinado caso com sendo ID considera também detecções que ocorrem dentro de um caso de FAI, mas que ocorrem antes do arco elétrico se iniciar. Assim, caso o ruído distorça a forma de onda do sinal de corrente, durante a operação normal do sistema, ao ponto da mesma ser confundida com uma falta de alta impedância antes mesmo que qualquer evento seja chaveado, pode acontecer de ocorrerem detecções incorretas dentro do próprio caso de FAI. Isso explica o porque de ser possível haverem

valores acima de 286.

Dessa forma, é possível que boa parte dos erros que compõem a contagem de *ID* sejam provenientes dos próprios casos de FAI. Com base nisso, para melhor entender essa questão juntamente com quais tipos de eventos foram mais desafiadores para a metodologia, a Figura 57 apresenta como que o número de erros de cada um dos classificadores para cada evento aumenta com a diminuição do valor de dB, sendo analisados os valores de 45, 40 e 35 dB.

Como pode ser observado na Figura 57(a), os casos que envolvem a maioria dos erros dos classificadores são os curto-circuitos bifásicos e trifásicos, sendo que, nesse nível de dB, todos os classificadores apresentaram pelo menos um erro nesse tipo de evento.

Figura 57 – Erros de detecção por tipo de evento para níveis de SNR de 45, 40 e 35 dB.



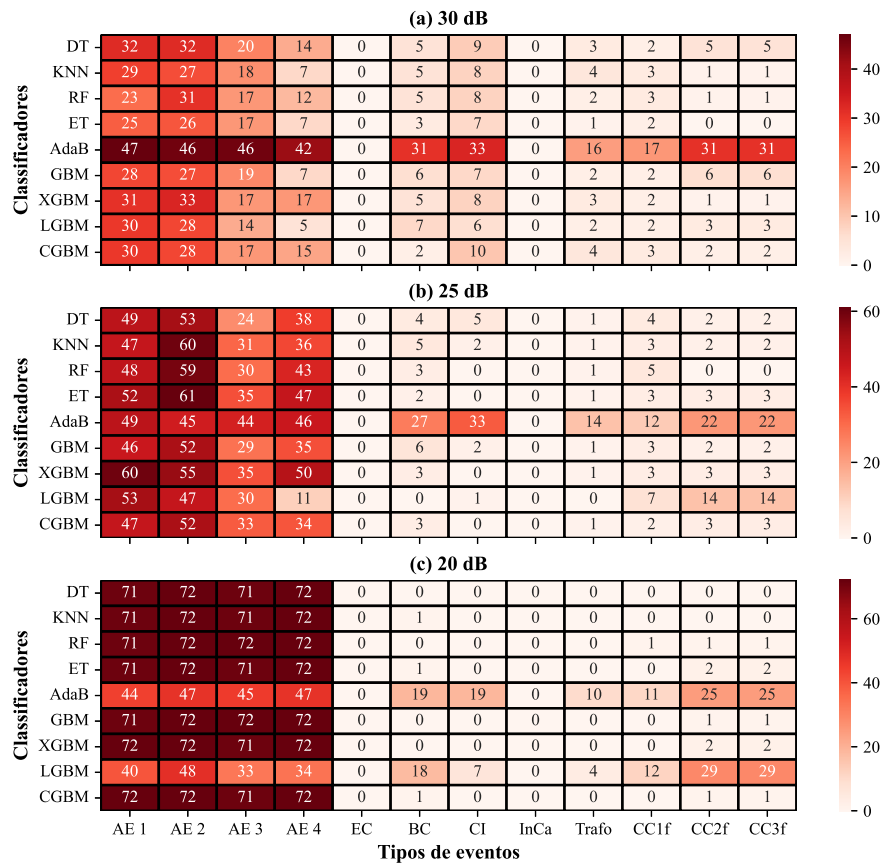
Fonte: Autor.

Além destes, houveram alguns erros relativos à energização de transformadores e carga indutiva. Outra informação interessante que pode ser obtida dessa Figura é que cada um dos classificadores cometeu pelo menos um erro em relação aos casos de FAI. Já na Figura 57(b), é possível perceber um aumento na quantidade de casos de FAI sendo classificados incorretamente, sendo que os seus valores começam a se comparar com aqueles vistos para os erros relativos aos demais casos. Nesse sentido, é importante notar que os tipos de eventos que lideraram os erros de classificação foram novamente os curtos-circuitos bifásicos e trifásicos. Porém, essa diferença em relação aos casos de falta de alta impedância já não é mais tão significativa como visto até então.

Para a Figura 57(c), a qual apresenta o desempenho da metodologia e seus diferentes classificadores para 35 dB, é possível perceber que o número de erros de detecção para os casos envolvendo FAI já ultrapassa o número obtido para os demais casos. Isso demonstra uma tendência de crescimento do número de erros para os casos de FAI conforme o nível de ruído aumenta.

A tendência dos erros relativos aos casos de FAI aumentarem conforme se aumenta o nível de ruído no sinal fica ainda mais nítida quando analisadas as Figuras 58(a), 58(b) e 58(c), as quais apresentam o valor de erros da metodologia para os casos de FAI com 30 dB, 25 dB e 20 dB, respectivamente. Nesse sentido, percebe-se que, a partir de 30 dB, já começa a existir uma predominância dos erros da metodologia para casos de FAI, sendo que, em 20 dB, a maioria dos classificadores levam a um valor de 72 casos erroneamente classificados, com exceção do AdaB e do LGBM, os quais apresentam números de erros bastante semelhantes tanto para casos de FAI, quanto para os demais eventos simulados.

Figura 58 – Erros de detecção por tipo de evento para níveis de SNR de 30, 25 e 20 dB.



Fonte: Autor.

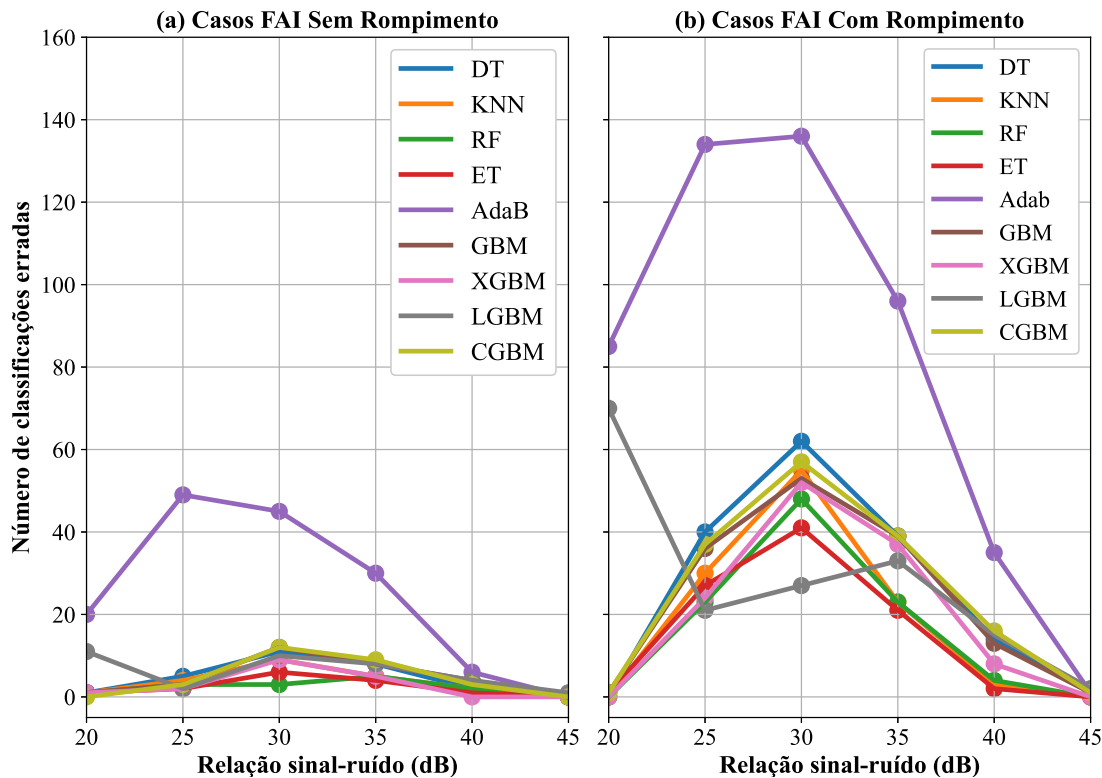
Se os resultados obtidos nas Figuras 57 e 58 forem comparados com aqueles vistos na Figura 56, é possível perceber que parece haver uma certa incongruência entre os resultados, já que entre 30 dB e 45 dB há um crescimento nos casos de ID, seguido uma queda em 25 dB e 20 dB. Entretanto, se for observado no gráfico dos erros por tipo de evento, como já discutido antes, tem-se um crescimento na quantidade de casos de FAI nos quais a metodologia cometeu algum

erro. Dessa forma, é estranho que *ID*, a qual é uma métrica que se refere ao erro de atuação da metodologia para outros eventos do sistemas, aconteça no mesmo intervalo que o número de classificações erradas de FAI aumenta. A explicação para esse fenômeno se dá pelo fato de que, caso a metodologia atue antes mesmo de começar o arco elétrico, esse caso mesmo sendo de FAI é considerado como uma falso positivo, uma vez que a falta propriamente dita não teve início ainda.

Sendo assim, o aumento na contagem de *ID* se dá tanto pelos erros de detecção da metodologia em relação aos demais casos dos sistema como também para os próprios casos de falta de alta impedância. Além disso, abaixo de 30 dB, percebe-se um crescimento significativo da contagem de *ND*, ou seja, casos de FAI que não foram identificados pela metodologia. Assim, como isso também é contado como um erro da metodologia em relação aos casos de FAI, então os mesmos também são computados no mapa de calor, o que explica também a continuidade do crescimentos dos erros para as simulações de arco, mesmo quando o valor de *ID* decresce.

Para analisar se o rompimento do cabo exerce alguma influência no número de detecções de FAI antes do início do arco elétrico, a Figura 59 apresenta a quantidade de casos incorretamente identificados, comparando as situações em que houve rompimento do cabo com aquelas em que não houve.

Figura 59 – Comportamento dos erros para FAI com e sem rompimento do condutor para diferentes níveis de ruído.



Fonte: Autor.

Assim, como pode ser observado analisando as Figuras 59(a) e 59(b), os casos com rompimento do cabo compõem uma parcela significativa dos casos computados como sendo *ID*.

Logo, existe uma correlação entre o rompimento do cabo e as detecções antes do início da FAI para valores de SNR abaixo de 45 dB.

Após realizada a discussão sobre o comportamento de *ID* e *ND* conforme o ruído aumenta, são então discutidas as métricas gerais de desempenho da metodologia, as quais foram apresentadas na Seção 4.3. O resultado do uso de diferentes classificadores é apresentado para os níveis de SNR entre 45 dB e 20 dB, uma vez que, acima disso, os valores se mantêm semelhantes ao que foi apresentado na análise sem ruído. A Tabela 13 traz os resultados para os classificadores tanto em relação as suas métricas de desempenho, as quais são apresentadas em porcentagem, como também em relação ao tempo médio de detecção das ocorrências de FAI pela metodologia (t_m), o qual é apresentado em segundos.

Tabela 13 – Métricas Gerais dos Classificadores.

	SNR	DT	KNN	RF	ET	AdaB	GBM	XGBM	LGBM	CGBM
<i>A</i>	45	98,258	99,477	99,652	99,303	96,69	98,955	99,652	98,258	98,955
	40	94,599	97,909	97,735	98,955	82,578	93,902	96,341	94,077	93,554
	35	82,578	88,502	87,456	88,85	51,394	82,404	85,192	83,972	82,753
	30	74,739	78,571	79,268	81,882	32,056	77,875	75,784	80,314	76,307
	25	66,899	65,505	65,505	62,718	37,805	67,073	61,672	68,99	67,073
	20	50,174	49,826	49,652	49,652	46,167	49,826	49,652	54,355	49,652
<i>D</i>	45	99,652	99,653	99,653	99,306	100,0	100,0	99,653	98,947	99,652
	40	100,0	99,293	99,645	100,0	100,0	100,0	99,286	99,628	99,628
	35	99,17	98,077	96,154	96,578	99,383	99,17	97,561	95,951	98,75
	30	88,372	92,411	86,498	88,382	100,0	92,825	83,7	84,064	90,411
	25	51,029	44,882	41,221	35,907	99,048	50,602	33,588	55,472	49,194
	20	0,697	0,697	0,348	0,699	57,377	0,35	0,348	64,251	0,348
<i>S</i>	45	96,864	99,301	99,65	99,301	93,38	97,909	99,65	97,578	98,258
	40	89,735	96,564	95,89	97,924	69,419	88,449	93,537	89,18	88,197
	35	70,571	80,573	80,255	82,315	32,524	70,27	75,915	74,924	71,257
	30	66,574	69,714	74,184	77,177	16,488	68,376	70,605	77,399	67,606
	25	78,55	81,875	85,897	84,762	24,094	79,692	85,256	80,583	80,675
	20	99,652	98,955	98,955	98,264	40,921	98,958	98,955	48,774	98,955
<i>SF</i>	45	99,642	99,649	99,65	99,301	100,0	100,0	99,65	98,947	99,647
	40	100,0	99,293	99,644	100,0	100,0	100,0	99,278	99,634	99,63
	35	99,156	98,062	96,183	96,604	99,259	99,153	97,647	96,078	98,755
	30	90,53	93,487	88,652	90,175	100,0	93,75	86,879	86,207	91,954
	25	68,602	65,174	63,507	61,663	99,123	67,801	60,455	67,847	67,609
	20	50,088	49,912	49,825	49,912	67,227	50,0	49,825	70,751	49,825
<i>SN</i>	45	96,949	99,308	99,653	99,306	93,791	97,952	99,653	97,578	98,282
	40	89,769	96,564	95,904	97,938	71,182	88,562	93,603	89,037	88,158
	35	70,92	80,696	80,128	82,201	36,674	70,71	75,235	74,295	71,171
	30	61,29	66,134	70,205	73,702	21,529	65,094	65,068	74,296	63,259
	25	63,59	66,279	71,053	65,957	22,609	65,625	65,672	71,014	65,946
	20	66,667	40,0	25,0	28,571	31,25	25,0	25,0	41,433	25,0
t_m	45	0,061	0,066	0,071	0,074	0,052	0,059	0,068	0,07	0,06
	40	0,083	0,091	0,092	0,101	0,065	0,076	0,095	0,103	0,083
	35	0,1	0,108	0,107	0,124	0,064	0,101	0,109	0,116	0,102
	30	0,179	0,186	0,192	0,207	0,088	0,173	0,195	0,197	0,173
	25	0,372	0,366	0,4	0,405	0,152	0,356	0,403	0,303	0,364
	20	0,451	0,341	0,231	0,45	0,391	0,096	0,451	0,266	0,451

Fonte: Autor.

Como já evidenciado anteriormente, o classificador AdaB foi o que apresentou o maior número de detecções incorretas de FAI dentre os classificadores analisados. Nesse sentido, se o mesmo for levado em consideração na hora de analisar o desempenho da metodologia para métricas que dependem da contagem *ID*, como *Accuracy*, *Security*, *Sensibility*, obviamente ele apresentaria os piores resultados devido a sua discrepância em relação aos outros algoritmos. Dessa forma, para as próximas análises, ele será considerado como um *outlier* e será desprezado. Isso é importante principalmente para não enviesar a análise relativa aos intervalos de máximos e mínimos das métricas obtidas nesse estudo.

A Tabela 14 apresenta os resultados das métricas em termos dos menores e maiores valores das métricas obtidas pela metodologia com diferentes classificadores.

Tabela 14 – Melhores e piores resultados de cada uma das métricas para diferentes níveis de ruído, desconsiderando o AdaB.

Métrica	SNR	Mínimo	Máximo	Δ
A	45	98,258 (DT, LGBM)	99,652 (RF, XGBM)	1,394
	40	93,554 (CGBM)	98,955 (ET)	5,401
	35	82,404 (GBM)	88,85 (ET)	6,446
	30	74,739 (DT)	81,882 (ET)	7,143
	25	61,672 (XGBM)	68,99 (LGBM)	7,318
	20	49,652 (RF, ET, XGBM, CGBM)	54,355 (LGBM)	4,703
D	45	98,947 (LGBM)	100,0 (GBM)	1,053
	40	99,286 (XGBM)	100,0 (DT, ET, GBM)	0,714
	35	95,951 (LGBM)	99,17 (DT, GBM)	3,219
	30	83,7 (XGBM)	92,825 (GBM)	9,125
	25	33,588 (XGBM)	55,472 (LGBM)	21,884
	20	0,348 (RF, XGBM, CGBM)	64,251 (LGBM)	63,903
S	45	96,864 (DT)	99,65 (RF, XGBM)	2,786
	40	88,197 (CGBM)	97,924 (ET)	9,727
	35	70,27 (GBM)	82,315 (ET)	12,045
	30	66,574 (DT)	77,399 (LGBM)	10,825
	25	78,55 (DT)	85,897 (RF)	7,347
	20	48,774 (LGBM)	99,652 (DT)	50,878
SF	45	98,947 (LGBM)	100,0 (GBM)	1,053
	40	99,278 (XGBM)	100,0 (DT, ET, GBM)	0,722
	35	96,078 (LGBM)	99,156 (DT)	3,078
	30	86,207 (LGBM)	93,75 (GBM)	7,543
	25	60,455 (XGBM)	68,602 (DT)	8,147
	20	49,825 (RF, XGBM, CGBM)	70,751 (LGBM)	20,926
SN	45	96,949 (DT)	99,653 (RF, XGBM)	2,704
	40	88,158 (CGBM)	97,938 (ET)	9,78
	35	70,71 (GBM)	82,201 (ET)	11,491
	30	61,29 (DT)	74,296 (LGBM)	13,006
	25	63,59 (DT)	71,053 (RF)	7,463
	20	25,0 (RF, GBM, XGBM, CGBM)	66,667 (DT)	41,667

Além disso, é apresentada a diferença entre o pior e o melhor desempenho (Δ), de modo a evidenciar o quão consistente é o desempenho dos classificadores entre si e o quanto isso se degrada conforme o ruído no sinal aumenta. Todos esses resultados são apresentados em porcentagem, assim como na Tabela anterior. A partir dessa Tabela, é possível avaliar com que frequência os classificadores aparecem como sendo os piores ou os melhores dentro do grupo de algoritmos analisados. Essa frequência é contabilizada abaixo para cada um dos modelos de ML utilizados:

- DT: 8 mínimos e 7 máximos;
- KNN: 0 mínimos e 0 máximos;
- RF: 4 mínimos e 5 máximos;
- ET: 1 mínimo e 9 máximos;
- GBM: 4 mínimos e 7 máximos;
- XGBM: 10 mínimos e 3 máximos;
- LGBM: 7 mínimos e 7 máximos; e
- CGBM: 7 mínimos e 0 máximos.

Com base nas contagens realizadas acima, é possível dividir os algoritmos em três grupos. O primeiro é composto pelos algoritmos que performaram abaixo dos demais, sendo esses CGBM e XGBM, para os quais houveram mais performances negativas do que positivas. O segundo grupo é relativo aos que apresentaram um desempenho moderado, ou seja, a frequência com que apresentaram uma menor performance foi igual ou muito próxima da com que apresentaram a melhor. Esse grupo é composto pelos algoritmos DT, KNN, RF, GBM e LGBM. Por fim, o terceiro grupo diz respeito aos modelos de ML que apresentaram um bom desempenho, estando dentre os melhores resultados com maior frequência do que nos piores resultados. O único algoritmo que apresentou tal comportamento para esse estudo foi o ET. Assim, de forma geral, tem-se a seguinte ordem dos desempenhos relativos:

- Desempenho abaixo dos demais: CGBM e XGBM;
- Desempenho moderado: DT, KNN, RF, GBM e LGBM; e
- Melhor desempenho: ET.

Vale ressaltar que essas comparações tem o objetivo de avaliar o desempenho dos classificadores entre si, uma vez que no geral todos performaram bem para o banco de dados analisado. Além disso, para níveis de SNR acima de 45 dB os valores de Δ estão abaixo de 3% e as métricas com valores acima de 95%. Entretanto, para níveis de SNR abaixo disso, é possível

perceber uma queda considerável nos valores das métricas de desempenho e um aumento da diferença de performance da metodologia para diferentes tipos de classificadores. Esse padrão se repete para todas as métricas, com exceção de A que apresenta uma diminuição de Δ em 20 dB. Porém, para as demais métricas, acontece um salto considerável nos valores de Δ , onde o mesmo passa a variar entre 20% até 63%.

A Tabela 15 apresenta os menores e maiores tempos médios de detecção e a diferença de tempo entre estes (Δt), todos em segundos. Essa Tabela traz dados importantes para este estudo, uma vez que ela mostra que classificadores de performance moderada como DT, GBM e LGBM, apresentaram os menores tempos médios de detecção. Em contrapartida, embora apareçam os classificadores DT e LGBM também nessa lista, o classificador ET, o qual apresentou os melhores desempenhos nas métricas, foi o mais demorado para vários valores de SNR.

Tabela 15 – Melhores e piores resultados dos tempos médios de detecção, desconsiderando o AdaB.

SNR	Menor tempo	Maior tempo	Δt
45	0,059 (GBM)	0,074 (ET)	0,015
40	0,076 (GBM)	0,103 (LGBM)	0,027
35	0,1 (DT)	0,124 (ET)	0,024
30	0,173 (GBM)	0,207 (ET)	0,034
25	0,303 (LGBM)	0,405 (ET)	0,102
20	0,096 (GBM)	0,451 (DT)	0,355

Fonte: Autor.

Por fim, outro fator interessante a ser analisado é que a discrepância entre os tempos médios de detecção de cada um dos classificadores cresce conforme o SNR diminui, demonstrando que níveis maiores de ruído dificultam a estabilização da resposta dos classificadores para cada um dos dados analisados, resultando em uma demora maior para alcançar o critério de 64 amostras consecutivas classificadas como FAI.

4.6 Considerações Finais

Neste Capítulo, foram apresentados os resultados da aplicação da metodologia no banco de teste usando diferentes modelos de ML. Esta aplicação foi realizada primeiramente considerando os casos sem a adição de ruído, permitindo ter uma noção inicial do desempenho dos classificadores na tarefa de realizar a detecção das FAIs. Nesse contexto, verificou-se que a metodologia não cometia nenhum erro para esse banco de dados quando os classificadores utilizados eram KNN, RF, ET e XGBM. Entretanto, também foi com a utilização desses modelos que a metodologia apresentou os maiores tempos até a detecção dos casos de FAI. Por outro lado, o uso da metodologia com o AdaB apresentou os menores tempos transcorridos para a detecção da falta, porém, isso resultou na maior quantidade de erros dentre os classificadores.

Na sequência, foi analisado como diferentes níveis de ruído impactavam na performance da metodologia dependendo de qual algoritmo de ML era utilizado. O resultado desse estudo foi

que o algoritmo que apresentava uma melhor performance diante dos diferentes cenários de ruído foi o ET, o qual também foi o algoritmo que pior performou nas métricas de tempo de detecção, seguido por algoritmos como DT, KNN, RF, GBM e LGBM. Além disso, outras informações importantes extraídas dos estudos com ruído foram que as métricas começavam a piorar apenas para valores de SNR abaixo de 50 dB e que, em situações que envolviam o rompimento de cabo, a probabilidade de ocorrer um falso positivo era maior do que em casos que não envolviam esse tipo de problema.

Por fim, o próximo Capítulo apresenta as conclusões finais deste Trabalho e as sugestões para trabalhos futuros.

5 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

Este Trabalho apresenta um estudo da aplicação dos modelos de *Machine Learning* (ML) para a detecção de FAI. Para isso, foi realizada a simulação de diferentes eventos do sistema elétrico de distribuição juntamente com simulações de FAIs a partir de quatro modelos de arco elétrico da literatura. Com isso, foi possível obter o comportamento da corrente do sistema na subestação para esses diferentes cenários. Além disso, foram analisados diferentes parâmetros extraídos de uma janela móvel de um ciclo com amostras do sinal de corrente de fase, das quais a curtose e a assimetria estatística foram escolhidas com base na técnica de seleção de características conhecida como RFECV.

Utilizando essas duas características, nove modelos de ML foram treinados dentre eles DT, KNN, RF, ET, AdaB, GBM, XGBM, LGBM e LGBM. Esses classificadores foram então avaliados e, em seguida, empregados no banco de dados de teste. Nesse sentido, para esse Trabalho, foi realizado tanto um estudo do desempenho dos algoritmos para casos que não envolvem ruído, quanto para casos com diferentes valores de SNR variando entre 100 db e 20 db. Com base nisso, foram obtidas as seguintes conclusões a respeito da aplicação desses modelos para o banco de teste simulado, levando em consideração os desempenhos obtidos para as métricas *Accuracy*, *Dependability*, *Safety*, *Security* e *Sensibility*.

- Resultados para a análise sem ruído:
 1. KNN, RF, ET e XGBM apresentaram 100% de desempenho em todas as métricas;
 2. GBM e CGBM apresentaram métricas abaixo de 100%, com os seus piores resultados na métrica *Security* com 99,650%;
 3. DT apresentou seu pior resultado em *Security* com 99,301%;
 4. LGBM foi o único que apresentou dificuldades em detectar casos de FAI para os testes sem ruído, mas por outro lado não realizou nenhuma detecção incorreta, sua pior métrica foi *Dependability* com 98,958%;
 5. AdaB foi o classificador que apresentou as piores métricas para essa análise, tendo um desempenho de 96,853% em *Security*.
 6. Para a análise dos tempos de detecção de cada um dos classificadores foi concluído que o comportamento dos classificadores do mais rápido para o mais lento foi AdaB, GBM, DT, XGBM, KNN, RF, ET e LGBM.

- Resultados para a análise com ruído:
 1. A quantidade de erros da metodologia começa a crescer significativamente a partir de 45 db;
 2. Casos de FAI envolvendo rompimento do cabo apresentam um número significativamente maior de casos de ID;

3. A discrepância entre o desempenho dos classificadores aumenta conforme o SNR diminui;
4. Adab se distancia significativamente dos resultados apresentados pelos demais classificadores, o que levou a desconsiderar o impacto do mesmo para os resultados finais;
5. Desempenho abaixo dos demais: CGBM e XGBM;
6. Desempenho moderado: DT, KNN, RF, GBM e LGBM;
7. Melhor Desempenho: ET;
8. Os classificadores que apresentaram os menores tempos médios de detecção pertencem ao grupo de Desempenho moderado;
9. O ET foi o que apresentou a maior demora para a maioria dos valores de SNR analisados; e
10. A diferença entre o tempo de detecção médio entre os classificadores cresce conforme o SNR diminui.

Com base nisso, conclui-se que a técnica que contribuiu melhor para o desempenho da metodologia proposta foi o ET, permitindo afirmar que a mesma pode contribuir para o aumento da confiabilidade e robustez dos sistemas de proteção diante da ocorrência de FAIs em redes elétricas de distribuição de energia.

5.1 Sugestões para Trabalhos Futuros

As sugestões para trabalhos futuros são:

- Testar a metodologia para diferentes bancos de dados, como, por exemplo, simulações obtidas no sistema IEEE 13 Barras e os dados de FAI dos ensaios realizados na Austrália;
- Testar outros algoritmos de ML, principalmente os de *Deep Learning*, como *Multilayer Perceptron* (MLP), *Convolutional Neural Network* (CNN), *Long Short-Term Memory Recurrent Neural Network* (LSTM-RNN), etc;
- Buscar outras características que podem ser extraídas da forma de onda de corrente da FAI e testar o seu impacto nas métricas dos algoritmos de ML;
- Buscar na literatura outros modelos de arco elétrico e implementá-los computacionalmente, treinando/testando os classificadores de ML com os dados resultantes da simulação destes;
- Mesclar modelos de arco elétrico de forma a criar novas configurações de características na forma de onda de corrente; e

- Buscar dados reais de FAI e aplicar os modelos de ML treinados a partir de modelos computacionais de arco elétrico nesses dados, analisando assim o quão confiáveis são os modelos computacionais para treinar classificadores que atuarão em cenários reais.

REFERÊNCIAS

- ACOSTA, M. R. C. *et al.* Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks. **IEEE access**, IEEE, v. 8, p. 19921–19933, 2020. Citado na página 48.
- AHMADI, A. *et al.* Decomposition-based stacked bagging boosting ensemble for dynamic line rating forecasting. **IEEE Transactions on Power Delivery**, IEEE, v. 38, n. 5, p. 2987–2997, 2023. Citado 2 vezes nas páginas 47 e 48.
- ALSAGRI, H.; YKHLEF, M. Quantifying feature importance for detecting depression using random forest. **International Journal of Advanced Computer Science and Applications**, Science and Information (SAI) Organization Limited, v. 11, n. 5, 2020. Citado 2 vezes nas páginas 50 e 52.
- ANING, S.; PRZYBYŁA-KASPEREK, M. Comparative study of twoing and entropy criterion for decision tree classification of dispersed data. **Procedia Computer Science**, Elsevier, v. 207, p. 2434–2443, 2022. Citado 2 vezes nas páginas 44 e 45.
- AZMI, S. S.; BALIGA, S. An overview of boosting decision tree algorithms utilizing adaboost and xgboost boosting strategies. **Int. Res. J. Eng. Technol**, v. 7, n. 5, p. 6867–6870, 2020. Citado na página 49.
- BHANDIA, R. *et al.* High impedance fault detection using advanced distortion detection technique. **IEEE Transactions on Power Delivery**, IEEE, v. 35, n. 6, p. 2598–2611, 2020. Citado na página 17.
- BREIMAN, L. Random forests. **Machine learning**, Springer, v. 45, p. 5–32, 2001. Citado na página 47.
- 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*. [S.l.: s.n.], 2016. p. 785–794. Citado na página 50.
- CHINCHALI, R. P. **A parametric digital signal processing algorithm for arcing high-impedance fault detection**. [S.l.: s.n.]: Texas A&M University, 1988. Citado na página 18.
- COSTA, F. B. *et al.* Real-time detection of transients induced by high-impedance faults based on the boundary wavelet transform. **IEEE Transactions on Industry Applications**, IEEE, v. 51, n. 6, p. 5312–5323, 2015. Citado na página 28.
- CZERWINSKA, U. Interpretability of machine learning models: How can one explain machine learning models? *In: Applied Data Science in Tourism: Interdisciplinary Approaches, Methodologies, and Applications*. [S.l.: s.n.]: Springer, 2022. p. 275–303. Citado 2 vezes nas páginas 41 e 42.
- EMANUEL, A. *et al.* High impedance fault arcing on sandy soil in 15 kv distribution feeders: contributions to the evaluation of the low frequency spectrum. **IEEE Transactions on Power Delivery**, IEEE, v. 5, n. 2, p. 676–686, 1990. Citado 4 vezes nas páginas 30, 31, 32 e 38.
- FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. **Journal of computer and system sciences**, Elsevier, v. 55, n. 1, p. 119–139, 1997. Citado 2 vezes nas páginas 47 e 49.

- FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. **Annals of statistics**, JSTOR, p. 1189–1232, 2001. Citado na página 49.
- FRIEDMAN, J. H.; BENTLEY, J. L.; FINKEL, R. A. An algorithm for finding best matches in logarithmic expected time. **ACM Transactions on Mathematical Software (TOMS)**, ACM New York, NY, USA, v. 3, n. 3, p. 209–226, 1977. Citado na página 46.
- GADANAYAK, D. A.; MALLICK, R. K. Interharmonics based high impedance fault detection in distribution systems using maximum overlap wavelet packet transform and a modified empirical mode decomposition. **International Journal of Electrical Power & Energy Systems**, Elsevier, v. 112, p. 282–293, 2019. Citado na página 21.
- GAO, J. *et al.* A high-impedance fault detection method for distribution systems based on empirical wavelet transform and differential faulty energy. **IEEE Transactions on Smart Grid**, IEEE, v. 13, n. 2, p. 900–912, 2021. Citado na página 21.
- GAUTAM, S.; BRAHMA, S. M. Detection of high impedance fault in power distribution systems using mathematical morphology. **IEEE Transactions on Power Systems**, IEEE, v. 28, n. 2, p. 1226–1234, 2012. Citado 4 vezes nas páginas 19, 34, 35 e 38.
- GEURTS, P.; ERNST, D.; WEHENKEL, L. Extremely randomized trees. **Machine learning**, Springer, v. 63, p. 3–42, 2006. Citado na página 48.
- GHADERI, A.; III, H. L. G.; MOHAMMADPOUR, H. A. High impedance fault detection: A review. **Electric power systems research**, Elsevier, v. 143, p. 376–388, 2017. Citado 3 vezes nas páginas 17, 26 e 29.
- GOMES, D. P.; OZANSOY, C.; ULHAQ, A. High-sensitivity vegetation high-impedance fault detection based on signal’s high-frequency contents. **IEEE Transactions on Power Delivery**, IEEE, v. 33, n. 3, p. 1398–1407, 2018. Citado na página 20.
- GONZÁLEZ, S. *et al.* A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities. **Information Fusion**, Elsevier, v. 64, p. 205–237, 2020. Citado 2 vezes nas páginas 46 e 50.
- GRIGG, T. **Concept learning and feature space**. 2019. Disponível em: <https://towardsdatascience.com/concept-learning-and-feature-spaces-45cee19e49db>. Acesso em: 29 fev. 2024. Disponível em: <https://towardsdatascience.com/concept-learning-and-feature-spaces-45cee19e49db>. Citado na página 54.
- GUO, G. *et al.* Knn model-based approach in classification. *In: SPRINGER. On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings.* [S.l.: s.n.], 2003. p. 986–996. Citado 2 vezes nas páginas 22 e 45.
- HADJOU DJ, Y.; PANDIT, R. K. Improving o&m decision tools for offshore wind farm vessel routing by incorporating weather uncertainty. **IET Renewable Power Generation**, Wiley Online Library, v. 17, n. 6, p. 1488–1499, 2023. Citado na página 48.
- HOYOS-BARCELÓ, C. *et al.* Efficient k-nn implementation for real-time detection of cough events in smartphones. **IEEE journal of biomedical and health informatics**, IEEE, v. 22, n. 5, p. 1662–1671, 2017. Citado na página 46.

- HSSINA, B. *et al.* A comparative study of decision tree id3 and c4. 5. **International Journal of Advanced Computer Science and Applications**, v. 4, n. 2, p. 13–19, 2014. Citado na página 42.
- JEERINGS, D.; LINDERS, J. Ground resistance-revisited. **IEEE Transactions on Power Delivery**, IEEE, v. 4, n. 2, p. 949–956, 1989. Citado na página 28.
- KAVI, M.; MISHRA, Y.; VILATHGAMUWA, M. D. High-impedance fault detection and classification in power system distribution networks using morphological fault detector algorithm. **IET Generation, Transmission & Distribution**, Wiley Online Library, v. 12, n. 15, p. 3699–3710, 2018. Citado na página 20.
- KE, G. *et al.* Lightgbm: A highly efficient gradient boosting decision tree. **Advances in neural information processing systems**, v. 30, 2017. Citado na página 50.
- KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. **Artificial intelligence**, Elsevier, v. 97, n. 1-2, p. 273–324, 1997. Citado na página 50.
- KOLLURI, J. *et al.* Reducing overfitting problem in machine learning using novel l1/4 regularization method. *In: IEEE. 2020 4th international conference on trends in electronics and informatics (ICOEI)(48184)*. [S.l.: s.n.], 2020. p. 934–938. Citado na página 40.
- LAI, T. *et al.* High-impedance fault detection using discrete wavelet transform and frequency range and rms conversion. **IEEE Transactions on Power Delivery**, IEEE, v. 20, n. 1, p. 397–407, 2005. Citado na página 19.
- LIEN, K.-Y. *et al.* Energy variance criterion and threshold tuning scheme for high impedance fault detection. **IEEE Transactions on Power Delivery**, IEEE, v. 14, n. 3, p. 810–817, 1999. Citado na página 19.
- LIMA, É. M.; BRITO, N. S. D.; SOUZA, B. A. de. High impedance fault detection based on stockwell transform and third harmonic current phase angle. **Electric Power Systems Research**, Elsevier, v. 175, p. 105931, 2019. Citado na página 21.
- LIMA, É. M. *et al.* High impedance fault detection method for distribution networks under non-linear conditions. **International Journal of Electrical Power & Energy Systems**, Elsevier, v. 131, p. 107041, 2021. Citado na página 20.
- LOPES, G. N. *et al.* Analysis of signal processing techniques for high impedance fault detection in distribution systems. **IEEE Transactions on Power Delivery**, IEEE, v. 36, n. 6, p. 3438–3447, 2020. Citado 2 vezes nas páginas 17 e 25.
- LÓPEZ, O. A. M.; LÓPEZ, A. M.; CROSSA, J. **Multivariate statistical machine learning methods for genomic prediction**. [S.l.: s.n.]: Springer Nature, 2022. Citado na página 40.
- MA, B. *et al.* Diagnostic classification of cancers using extreme gradient boosting algorithm and multi-omics data. **Computers in biology and medicine**, Elsevier, v. 121, p. 103761, 2020. Citado na página 50.
- MISHRA, M.; PANIGRAHI, R. R. Taxonomy of high impedance fault detection algorithm. **Measurement**, Elsevier, v. 148, p. 106955, 2019. Citado na página 25.

- MOLOI, K.; DAVIDSON, I. High impedance fault detection protection scheme for power distribution systems. **Mathematics**, MDPI, v. 10, n. 22, p. 4298, 2022. Citado 2 vezes nas páginas 18 e 21.
- NAM, S. *et al.* A modeling method of a high impedance fault in a distribution system using two series time-varying resistances in emtp. *In: IEEE. 2001 Power Engineering Society Summer Meeting. Conference Proceedings (Cat. No. 01CH37262)*. [S.l.: s.n.], 2001. v. 2, p. 1175–1180. Citado 5 vezes nas páginas 28, 29, 31, 32 e 33.
- NATEKIN, A.; KNOLL, A. Frontiers in neurorobotics. **Gradient boosting machines, a tutorial**, 2013. Citado na página 49.
- PROKHORENKOVA, L. *et al.* Catboost: unbiased boosting with categorical features. **Advances in neural information processing systems**, v. 31, 2018. Citado na página 50.
- PRUSTY, S.; PATNAIK, S.; DASH, S. K. Skcv: Stratified k-fold cross-validation on ml classifiers for predicting cervical cancer. **Frontiers in Nanotechnology**, Frontiers Media SA, v. 4, p. 972421, 2022. Citado na página 41.
- QUINLAN, J. R. Induction of decision trees. **Machine learning**, Springer, v. 1, p. 81–106, 1986. Citado na página 42.
- QUINLAN, J. R. **C4. 5: programs for machine learning**. [S.l.: s.n.]: Elsevier, 2014. Citado na página 44.
- RAJANI, N.; MCARDLE, K.; DHILLON, I. S. Parallel k nearest neighbor graph construction using tree-based data structures. *In: BARCELONA SUPERCOMPUTING CENTER. 1st high performance graph mining workshop, sydney, 10 august 2015*. [S.l.: s.n.], 2015. Citado na página 46.
- RANGANATHAN, S.; NAKAI, K.; SCHONBACH, C. **Encyclopedia of bioinformatics and computational biology: ABC of bioinformatics**. [S.l.: s.n.]: Elsevier, 2018. Citado 4 vezes nas páginas 39, 40, 42 e 45.
- ROKACH, L.; MAIMON, O. Decision trees. **Data mining and knowledge discovery handbook**, Springer, p. 165–192, 2005. Citado 2 vezes nas páginas 42 e 43.
- ROY, S.; DEBNATH, S. Psd based high impedance fault detection and classification in distribution system. **Measurement**, Elsevier, v. 169, p. 108366, 2021. Citado na página 21.
- RUTKOWSKI, L. *et al.* The cart decision tree for mining data streams. **Information Sciences**, Elsevier, v. 266, p. 1–15, 2014. Citado na página 45.
- SAEED, U. *et al.* Fault diagnosis based on extremely randomized trees in wireless sensor networks. **Reliability engineering & system safety**, Elsevier, v. 205, p. 107284, 2021. Citado na página 48.
- SANCHEZ-MARTINEZ, S. *et al.* Machine learning for clinical decision-making: challenges and opportunities in cardiovascular imaging. **Frontiers in cardiovascular medicine**, Frontiers Media SA, v. 8, p. 765693, 2022. Citado na página 38.
- SANTOS, W. *et al.* High-impedance fault identification on distribution networks. **IEEE Transactions on Power Delivery**, IEEE, v. 32, n. 1, p. 23–32, 2016. Citado na página 17.

SARKER, I. H. Machine learning: Algorithms, real-world applications and research directions. **SN computer science**, Springer, v. 2, n. 3, p. 160, 2021. Citado 3 vezes nas páginas 38, 39 e 41.

SARWAGYA, K.; DE, S.; NAYAK, P. K. High-impedance fault detection in electrical power distribution systems using moving sum approach. **IET Science, Measurement & Technology**, Wiley Online Library, v. 12, n. 1, p. 1–8, 2018. Citado na página 20.

SCIKIT-LEARN. **Classifier Comparison**. 2024. Disponível em: https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html#sphx-glr-auto-examples-classification-plot-classifier-comparison-py. Acesso em: 29 fev. 2024. Disponível em: https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html#sphx-glr-auto-examples-classification-plot-classifier-comparison-py. Citado na página 55.

SEDIGHI, A. A new model for high impedance fault in electrical distribution systems. **International Journal of Scientific Research in Computer Science and Engineering**, v. 2, n. 4, p. 6–12, 2014. Citado 3 vezes nas páginas 35, 36 e 38.

SERRANO, A.; MICÓ, L.; ONCINA, J. Which fast nearest neighbour search algorithm to use? *In: SPRINGER. Pattern Recognition and Image Analysis: 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5-7, 2013. Proceedings 6*. [S.l.: s.n.], 2013. p. 567–574. Citado na página 45.

SHANNON, C. Claude shannon. **Information Theory**, eM Publications, v. 3, p. 224, 1948. Citado na página 43.

SHENG, Y.; ROVNYAK, S. M. Decision tree-based methodology for high impedance fault detection. **IEEE Transactions on Power Delivery**, IEEE, v. 19, n. 2, p. 533–536, 2004. Citado na página 19.

SILVA, S. *et al.* High impedance fault detection in power distribution systems using wavelet transform and evolving neural network. **Electric power systems research**, Elsevier, v. 154, p. 474–483, 2018. Citado na página 20.

SINGH, U. *et al.* A machine learning-based gradient boosting regression approach for wind power production forecasting: A step towards smart grid environments. **Energies**, MDPI, v. 14, n. 16, p. 5196, 2021. Citado na página 49.

SULTAN, A.; SWIFT, G.; FEDIRCHUK, D. Detecting arcing downed-wires using fault current flicker and half-cycle asymmetry. **IEEE Transactions on Power Delivery**, IEEE, v. 9, n. 1, p. 461–470, 1994. Citado 4 vezes nas páginas 19, 26, 27 e 28.

TALUKDER, M. A. *et al.* Machine learning-based lung and colon cancer detection using deep feature extraction and ensemble learning. **Expert Systems with Applications**, Elsevier, v. 205, p. 117695, 2022. Citado na página 41.

TAUNK, K. *et al.* A brief review of nearest neighbor algorithm for learning and classification. *In: IEEE. 2019 international conference on intelligent computing and control systems (ICCS)*. [S.l.: s.n.], 2019. p. 1255–1260. Citado na página 45.

THERON, J. J.; PAL, A.; VARGHESE, A. Tutorial on high impedance fault detection. *In: IEEE. 2018 71st Annual Conference for Protective Relay Engineers (CPRE)*. [S.l.: s.n.], 2018. p. 1–23. Citado na página 25.

- THOMAS, M. S.; BHASKAR, N.; PRAKASH, A. Voltage based detection method for high impedance fault in a distribution system. **Journal of The Institution of Engineers (India): Series B**, Springer, v. 97, p. 413–423, 2016. Citado na página 20.
- TIMOFEEV, R. Classification and regression trees (cart) theory and applications. **Humboldt University, Berlin**, v. 54, p. 48, 2004. Citado na página 42.
- TORRES, V. *et al.* Modeling and detection of high impedance faults. **International Journal of Electrical Power & Energy Systems**, Elsevier, v. 61, p. 163–172, 2014. Citado 2 vezes nas páginas 37 e 38.
- VEERASAMY, V. *et al.* Lstm recurrent neural network classifier for high impedance fault detection in solar pv integrated power system. **IEEE access**, IEEE, v. 9, p. 32672–32687, 2021. Citado na página 21.
- VEERASAMY, V. *et al.* High-impedance fault detection in medium-voltage distribution network using computational intelligence-based classifiers. **Neural Computing and Applications**, Springer, v. 31, p. 9127–9143, 2019. Citado na página 21.
- WANG, C. *et al.* Indicator element selection and geochemical anomaly mapping using recursive feature elimination and random forest methods in the jingdezhen region of jiangxi province, south china. **Applied Geochemistry**, Elsevier, v. 122, p. 104760, 2020. Citado na página 51.
- WANG, R. Adaboost for feature selection, classification and its relation with svm, a review. **Physics Procedia**, Elsevier, v. 25, p. 800–807, 2012. Citado 2 vezes nas páginas 49 e 50.
- WANG, S.; DEHGHANIAN, P. On the use of artificial intelligence for high impedance fault detection and electrical safety. **IEEE Transactions on Industry Applications**, IEEE, v. 56, n. 6, p. 7208–7216, 2020. Citado 3 vezes nas páginas 17, 18 e 21.
- WEI, M. *et al.* Distortion-based detection of high impedance fault in distribution systems. **IEEE Transactions on Power Delivery**, IEEE, v. 36, n. 3, p. 1603–1618, 2020. Citado 2 vezes nas páginas 17 e 48.
- WONTROBA, A. *et al.* High-impedance fault detection on downed conductor in overhead distribution networks. **Electric Power Systems Research**, Elsevier, v. 211, p. 108216, 2022. Citado na página 22.
- WOODMAN, R. J.; MANGONI, A. A. A comprehensive review of machine learning algorithms and their application in geriatric medicine: present and future. **Aging Clinical and Experimental Research**, Springer, v. 35, n. 11, p. 2363–2397, 2023. Citado na página 38.
- YAMAN, M. A.; RATTAY, F.; SUBASI, A. Comparison of bagging and boosting ensemble machine learning methods for face recognition. **Procedia Computer Science**, Elsevier, v. 194, p. 202–209, 2021. Citado na página 47.
- ZAMANAN, N.; SYKULSKI, J. K. Modelling arcing high impedances faults in relation to the physical processes in the electric arc. **WSEAS Transactions on power systems**, v. 1, n. 8, p. 1507–1512, 2006. Citado na página 26.
- ZHANG, C. *et al.* Feature selection of power system transient stability assessment based on random forest and recursive feature elimination. *In*: IEEE. **2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)**. [S.l.: s.n.], 2016. p. 1264–1268. Citado 2 vezes nas páginas 49 e 51.