MACILIO DA SILVA FERREIRA

**APLICAÇÃO DE TÉCNICAS DE APRENDIZADO PROFUNDO PARA A
DETECÇÃO E DIAGNÓSTICO DO ESTADO DE
OPERACIONALIDADE DE SISTEMAS: UM ESTUDO DE CASO EM
AMVS FERROVIÁRIOS**

São Paulo

2021

MACILIO DA SILVA FERREIRA

**Aplicação de Técnicas de Aprendizado Profundo para a Detecção e Diagnóstico
do Estado de Operacionalidade de Sistemas: Um Estudo de Caso em AMVS
Ferroviários**

Versão Corrigida

Dissertação apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do
título de Mestre em Ciências.

Área de Concentração: Engenharia de
Computação

Orientador: Prof. Dr. Paulo Sergio Cugnasca

São Paulo
2021

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 29 de Novembro de 2021

Assinatura do autor: Macilio da Silva Ferreira

Assinatura do orientador: Paulo Sérgio Aguiar

FICHA CATALOGRÁFICA

Ferreira, Macilio da Silva

APLICAÇÃO DE TÉCNICAS DE APRENDIZADO PROFUNDO PARA A DETECÇÃO E DIAGNÓSTICO DO ESTADO DE OPERACIONALIDADE DE SISTEMAS: UM ESTUDO DE CASO EM AMVS FERROVIÁRIOS / M. S. Ferreira -- versão corr. -- São Paulo, 2021.
205 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.Manutenção baseada em condição 2.Detecção e Diagnóstico de Falhas 3.Aparelho de Mudança de Via 4.Métodos preditivos 5.Ensemble de classificadores I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II.t.

A Deus;

Aos meus amados pais, Marcia e Manoel;

Ao meu amor, Joyce Nayne;

Aos meus irmãos, Mardson e Magna;

Aos meus sobrinhos, Avylla, Adrian, Júlia e Melissa;

Aos meus sogros, Gorete e José;

E em memória de Raimundo Lopes de Souza e Maria José Lopes da Silva, meus avós, a quem
tenho imensa admiração e saudades inesgotáveis.

AGRADECIMENTOS

Agradeço muito a Deus por todas as bênçãos que me foram dadas, principalmente por me dar forças para superar os momentos difíceis e conseguir concluir esta pesquisa.

Há dois anos sai do interior do Ceará, da minha amada Quixadá, para me dedicar a este mestrado. Neste período, tive a felicidade de conhecer diversas pessoas maravilhosas e contar com o apoio delas. Agradeço a cada uma delas. Em especial, a Ana Klyssia, por toda ajuda em minha chegada a São Paulo e amizade.

À toda a minha família, que apesar dos muitos quilômetros de distância, esteve presente nos momentos mais difíceis dessa jornada, sempre me apoiando, encorajando e compreendendo as minhas inúmeras ausências durante a realização desta pesquisa. Em especial, aos meus amados pais, Marcia e Manoel, muito obrigado por todo o amor, dedicação e bom exemplo de determinação, perseverança, humildade, honestidade e conduta com o qual vocês criaram a mim e meus irmãos, Mardson (muito obrigado pelos ensinamentos, apoio e pelo exemplo de determinação, além dos cuidados e dedicação com a nossa família em minha ausência) e Magna, a quem também sou muito grato. Amo imensamente vocês.

À minha amada esposa, Joyce, primeiramente peço desculpas por não conseguir dar a devida atenção em diversos momentos, privando-a da minha presença, ao longo deste trabalho. Agradeço imensamente por todo apoio, compreensão, incentivo, companheirismo e dedicação à minha família, pois a distância sempre foi um problema e, nos momentos críticos, você os apoiou na minha ausência. Muito obrigado, amo você. Aproveito para agradecer aos meus queridos sogros, Gorete e José, por todo apoio e compreensão. Agradeço de coração.

Meu muito obrigado ao professor Dr. Paulo Sergio Cugnasca, meu orientador, pela excelente orientação. Sempre se dedicando e contribuindo com ideias para melhoria desta pesquisa. O senhor foi fundamental para o desenvolvimento deste trabalho, muito obrigado por toda confiança e ensinamentos. Aproveito para agradecer aos professores Dr. João Batista e Jorge Rady, pelas valiosas contribuições tanto na minha formação quanto na publicação dos resultados desta pesquisa. Aos professores do Departamento de Engenharia de Computação e Sistemas Digitais – PCS, que contribuíram para a minha formação nesse mestrado.

Aos meus queridos amigos do GAS – Grupo de Análise e Segurança da Escola Politécnica da USP – (um local excelente para se trabalhar e conviver, principalmente pelas pessoas maravilhosas que o compõem), meus mais sinceros agradecimentos a vocês. Em especial,

novamente, ao professor Dr. Paulo Sergio Cugnasca, responsável pelo meu ingresso nesse grupo, e aos amigos de maior convivência: Lúcio Vismari, David Oliveira, Caroline Molina e Alexandre Nascimento, meus agradecimentos pelo prazeroso convívio e pelas proveitosas discussões que contribuíram, e muito, para este trabalho. A todos os demais integrantes do GAS, não nomeados aqui (não propositalmente), agradeço de coração.

Meu muito obrigado ao ex-estagiário do GAS Vitor Hugo Perles, cujo trabalho de iniciação científica contribuiu com este trabalho.

À VALE S.A pelo suporte financeiro e institucional que viabilizou esta pesquisa – Acordo de Cooperação Científica e Tecnológica nº 4600043577.

Aos meus amigos do grupo de futebol, aqui representados por Ricardo Franco, responsável pelo meu primeiro contato com o grupo, Maximiliano Oliveira e Oriosvaldo Antônio, a quem sou muito grato por toda ajuda durante a minha adaptação em São Paulo, pela amizade e pelas prazerosas tardes de Sábado repletas de descontração e futebol.

“Even the smallest person can change the course of the future.”

Galadriel, *The Lord of the Rings*

RESUMO

A manutenção representa um processo importante no sistema de transporte ferroviário. A eficiência do processo de manutenção impacta diretamente na disponibilidade da via e, consequentemente, tanto nos custos operacionais quanto no desempenho de qualquer sistema de transporte ferroviário. Os aparelhos de mudança de via são equipamentos importantes no contexto operacional da ferrovia, nos quais a ocorrência de uma falha pode provocar à indisponibilidade operacional ou mesmo acidentes. Portanto, um processo de manutenção eficiente para este ativo se torna essencial para o bom desempenho da ferrovia. Atualmente, em sua maioria, esse processo é baseado em abordagens corretivas e preventivas, que podem levar a longos períodos de indisponibilidade do sistema ou a custos altos desnecessários. Além disso, detectar e diagnosticar, com precisão, os diversos estados de degradação de um equipamento é tido como um desafio para as pesquisas nessa área, principalmente quando os modos de falhas representados pelos estados são desconhecidos ou amplos. Esta pesquisa científica tem como objetivo propor métodos preditivos, baseando-se em técnicas de aprendizado de máquina, capazes de detectar e diagnosticar os estados de operacionalidade (degradação) de um AMV, por intermédio tanto da análise do histórico de dados de operação quanto da análise de dados em tempo real, por meio da ocorrência de anomalias na assinatura elétrica da corrente de operação da *Máquina de Chave* que opera o AMV. Para isso, são analisadas as principais técnicas de aprendizado de máquina utilizadas em aplicações semelhantes e dados reais de máquinas de chave de uma ferrovia no Brasil. Como resultados tem-se: (i) um conjunto de modelos aptos a detectar e diagnosticar os diversos estados de operacionalidade de um AMV, incluindo as causas das falhas; (ii) utilização desses métodos como uma ferramenta de apoio para auxiliar o especialista de domínio na tomada de decisão de manutenção; (iii) geração de um *ranking* com os algoritmos que obtiveram os melhores resultados. Por fim, são apresentadas as principais conclusões decorrentes desses experimentos.

Palavras-chave: Manutenção baseada em condição, detecção e diagnóstico de falhas, aparelho de mudança de via, métodos preditivos, ensemble de classificadores.

ABSTRACT

Maintenance is an important process in the railway system. The efficiency of the maintenance process directly impacts the availability of the track and, consequently, both the operating costs and the performance of any railway transportation system. Track Switch is an important equipment in the operational context of the railway, in which the occurrence of a failure can cause operational unavailability or even accidents. Therefore, an efficient maintenance process for this asset becomes essential for the good performance of the railway. Currently, this process is mostly based on corrective and preventive approaches, which can lead to long periods of unavailability of the system or to unnecessarily high costs. Furthermore, accurately detecting and diagnosing several degradation states of equipment is considered a challenge for research in this area, especially when the failure modes represented by the states are unknown or wide. This scientific research aims to propose predictive methods, based on machine learning techniques, capable of detecting and diagnosing the operational states (degradation) of an AMV, through both the analysis of the history of operation data and the analysis of data in real-time, through the occurrence of anomalies in the electrical signature of the operating current of the Point Machine that operates the AMV. For this, the main machine learning techniques used in similar applications and real data from point machines on a railway in Brazil are analyzed. The results are: (i) a set of models able to detect and diagnose several operational states of an AMV, including the causes of failures; (ii) using these methods as a support tool to assist the domain specialist in making maintenance decisions; (iii) generation of a ranking with the algorithms that obtained the best results. Finally, the main conclusions resulting from these experiments are presented.

Keywords: Condition-based maintenance, fault detection and diagnosis, track switches, predictive methods, ensemble classifier.

LISTA DE FIGURAS

Figura 1. Principais componentes da Via Permanente (Fonte: (KLINCEVICIUS, 2011)).	34
Figura 2. Principais componentes da infraestrutura ferroviária (Fonte: Adaptado de (HAY, 2003)).	34
Figura 3. Componentes do Aparelho de Mudança de Via (AMV), sendo (1) a máquina de chave; (2) encosto da agulha; (3) barra de operação; (4) trilho de proteção (contratrilho); (5) agulhas do AMV; (6) cesto da barra de operação; e (7) jacaré (Fonte: Autor).	39
Figura 4. Configuração Normal do AMV: Simulação do movimento do trem (Fonte: Adaptado de (FUN, 2018)).	39
Figura 5. Configuração Reverso do AMV: Simulação do movimento do trem (Fonte: Adaptado de (FUN, 2018)).	40
Figura 6. Máquina de Chave Elétrica (Fonte: Vale S.A).	40
Figura 7. Exemplificação de um trecho de uma ferrovia, representado no sistema de intertravamento, com um ponto de mudança de via (Fonte: Vale S.A).	42
Figura 8. Relação (causa-efeito) entre falha, erro e disfunção (Fonte: adaptado de (JOHNSON, 1989; ALMEIDA JR, 2003)).	50
Figura 9. Propagação dos defeitos e sua relação com os estados do sistema (Fonte: adaptado de (VISMARI, 2007)).	51
Figura 10. Tipos de Manutenção (Fonte: Adaptado de (MÁRQUEZ, 2007)).	56
Figura 11. Curva característica da vida útil de equipamentos, comumente conhecida como curva da Banheira (Fonte: Adaptado de (JOHNSON, 1989; SILVA NETO, 2014)).	58
Figura 12. Estrutura de um sistema de Detecção e Diagnóstico de Falhas (Fonte: Adaptado de (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000)).	64
Figura 13. Relação entre as distâncias intra-cluster e inter-cluster (Fonte: Adaptado de (TAN; STEINBACH; KUMAR, 2013)).	72

Figura 14. Processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas – FDD (Fonte: Autor).....	94
Figura 15. Sistemática das etapas necessárias para o desenvolvimento dos modelos FDDs (Fonte: Autor).	100
Figura 16. Detecção em camadas do estado de operacionalidade do sistema (Fonte: Autor).	101
Figura 17. Representação dos clusters obtidos com aplicação da técnica não supervisionada no processo de detecção em camadas do estado de operacionalidade do sistema (Fonte: Autor).	103
Figura 18. Sistemática para a interpretação semiautomática de clusters (Fonte: Autor).....	107
Figura 19. Fluxo de avaliação da relevância dos resultados do processo de detecção do estado de operacionalidade do equipamento (Fonte: Autor).	107
Figura 20. Estrutura do classificador ensemble (Fonte: Autor).....	110
Figura 21. Estrutura do modelo de classificação MLP (Fonte: Autor).....	111
Figura 22. Processo de avaliação do desempenho dos modelos de detecção e diagnóstico do estado de operacionalidade de um equipamento (Fonte: Autor).	116
Figura 23. Operadoras ferroviárias do Brasil (Fonte: Adaptado de (KLINCEVICIUS, 2011; CNT, 2018)).	121
Figura 24. Fases da assinatura elétrica durante um movimento de configuração de um AMV (Fonte: Autor).	126
Figura 25. Pré-processamento dos Dados: Primeira etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (Fonte: Autor).	128
Figura 26. Desenvolvimento dos modelos FDD utilizando técnicas de Aprendizado de Máquina: Segunda etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (Fonte: Autor).....	132

Figura 27. Definindo o número ótimo de clusters por meio do critério de validação relativa Largura de Silhueta (Fonte: Autor).	134
Figura 28. Avaliação dos modelos FDDs: Terceira etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (Fonte: Autor).	137
Figura 29. Modelo 1 - Cenário 1: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	140
Figura 30. Modelo 1 - Cenário 1: Matriz de confusão da etapa de detecção de defeitos (Fonte: Autor).	140
Figura 31. Modelo 1 – Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	142
Figura 32. Modelo 1 – Cenário 2: Matriz de confusão da etapa de detecção de defeitos (Fonte: Autor).	143
Figura 33. Modelo 1 – Cenário 2: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	143
Figura 34. Modelo 1 – Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	144
Figura 35. Modelo 1 – Cenário 3: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	145
Figura 36. Modelo 1 – Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).	145
Figura 37. Método 1 – Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	146
Figura 38. Modelo 2 - Cenário 1: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	148
Figura 39. Modelo 2 - Cenário 1: Matriz de confusão da etapa de detecção de defeitos (Fonte: Autor).	149

Figura 40. Modelo 2 - Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	149
Figura 41. Modelo 2 - Cenário 2: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	150
Figura 42. Método 2 - Cenário 2: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).	151
Figura 43. Modelo 2 - Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	151
Figura 44. Modelo 2 - Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).	152
Figura 45. Modelo 2 - Cenário 3: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	153
Figura 46. Modelo 2 - Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	153
Figura 47. Modelo 3 – Cenário 1: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	156
Figura 48. Modelo 3 – Cenário 1: Matriz de confusão da etapa de detecção de defeitos (Fonte: Autor).	156
Figura 49. Modelo 3 - Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	157
Figura 50. Modelo 3 – Cenário 2: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	158
Figura 51. Modelo 3 – Cenário 2: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).	158
Figura 52. Modelo 3 – Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	159

Figura 53. Modelo 3 - Cenário 3: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).	160
Figura 54. Modelo 3 - Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).	160
Figura 55. Modelo 3 - Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).	161
Figura 56. Assinatura elétrica de uma instância de dados detectada incorretamente como defeito pelo detector ensemble (Fonte: Autor).	170
Figura 57. Assinaturas elétricas de instâncias de dados detectadas incorretamente como normal (“Sem Defeito”) (Fonte: Autor).	170
Figura 58. Exemplos de inconsistências de classificação com um cenário idêntico, em relação a máquina de chave, local de instalação (house) e configuração de movimento, de acordo com o segundo fabricante (Fonte: Autor).	171
Figura 59. Exemplos de inconsistências de classificação com um cenário idêntico de acordo com o primeiro fabricante (Fonte: Autor).	171
Figura 60. Exemplos de inconsistências de classificação com configurações de movimento diferentes (Fonte: Autor).	172
Figura 61. Instâncias de dados com mais de um tipo de defeito. Sendo que, a esquerda, tem-se um exemplo de comportamento da classe Atrito. Enquanto que, a direita, tem-se um exemplo de comportamento da classe Problema na Fase 2 (Fonte: Autor).	175
Figura 62. Exemplos de inconsistências e/ou erros de classificação da etapa de diagnóstico dos defeitos (Fonte: Autor).	175
Figura 63. Exemplo de um erro de classificação na etapa de diagnóstico dos defeitos (Fonte: Autor).	176
Figura 64. Comportamento de uma instância de dados da classe Corrente Baixa e Oscilando (Fonte: Autor).	177

Figura 65. Comportamento de uma instância de dados da classe Múltiplos Acionamentos Simultâneos (Fonte: Autor).	177
Figura 66. Comportamento de uma instância da classe Atraso na Partida (Fonte: Autor)....	178
Figura 67. Comportamento de uma instância de dados da classe Desligamento do motor (Fonte: Autor).	178
Figura 68. Comportamento de uma instância de dados da classe Sobrecorrente no Travamento (Fonte: Autor).	179
Figura 69. Exemplo de criação de novos modelos de classificação por meio da combinação dos valores dos atributos da estrutura base (Fonte: Autor).	197
Figura 70. Sistemática de uma tarefa de classificação (Fonte: Autor).	198
Figura 71. Comportamento da função <i>Sigmoide</i> (Fonte: Autor).	201
Figura 72. Comportamento da função <i>Softmax</i> (Fonte: Autor).	202
Figura 73. Comportamento da função <i>ReLu</i> (Fonte: Autor).	203

LISTA DE TABELAS

Tabela 1. Controle da Manutenção de acordo com a forma como o reparo foi realizado (Fonte: (ABRAMAN, 2017)).	26
Tabela 2. Custos de Manutenção (Fonte: (ABRAMAN, 2017)).	27
Tabela 3. Evolução da manutenção (Fonte: adaptado de (KARDEC; NASCIF, 2009)).	54
Tabela 4. Matriz de confusão (C) de um problema de classificação de anomalias, onde C_{ij} representa o número de instâncias de dados da classe i preditas, pelo classificador, como sendo da classe j (Fonte: Adaptado de (TAN; STEINBACH; KUMAR, 2013)).	69
Tabela 5. Comparativo entre o SOS:D ² e a literatura, especificamente dos métodos de detecção e diagnóstico de falhas aplicados no setor ferroviário (Fonte: Autor).	90
Tabela 6. Exemplo de criação de novos atributos (Fonte: Autor).	131
Tabela 7. Criticidade de cada tipo de defeito (Fonte: Autor).	136
Tabela 8. Cenários de teste para a avaliação dos modelos FDDs (Fonte: Autor).	139
Tabela 9. Desempenho do primeiro modelo FDD, referente ao método 1 (combinação das técnicas K-means e SOM), nos três cenários de teste (Fonte: Autor).	147
Tabela 10. Desempenho do segundo modelo FDD, referente ao método 2 (combinação das técnicas Autoencoder e K-means), nos três cenários de teste (Fonte: Autor).	154
Tabela 11. Desempenho do terceiro modelo FDD, referente ao método 3 (combinação das técnicas Autoencoder e SOM), nos três cenários de teste (Fonte: Autor).	163
Tabela 12. Desempenho geral dos modelos de detecção e diagnóstico do estado de operacionalidade do AMV (Fonte: Autor).	167
Tabela 13. Descrição dos classificadores-base utilizados no desenvolvimento do classificador <i>ensemble</i> (Fonte: Autor).	205

LISTA DE ABREVIATURAS E SIGLAS

ABRAMAN	<i>Associação Brasileira de Manutenção e Gestão de Ativos</i>
AMV	<i>Aparelho de Mudança de Via</i>
ANNT	<i>Agência Nacional de Transporte Terrestre</i>
CNAP	<i>Unidade Controladora do Motor</i>
CNT	<i>Confederação Nacional do Transporte</i>
EFC	<i>Estrada de Ferro de Carajás</i>
EFVM	<i>Estrada de Ferro Vitória a Minas</i>
EFPO	<i>Estrada de Ferro Paraná Oeste</i>
FCA	<i>Ferrovias Centro-Atlântica</i>
FDD	<i>Fault Detection Diagnosis</i>
FNSTC	<i>Ferrovias Norte Sul Tramo Central</i>
FNSTN	<i>Ferrovias Norte Sul Tramo Norte</i>
FTC	<i>Ferrovias Tereza Cristina</i>
FTL	<i>Ferrovias Transnordestina Logística</i>
MCH	<i>Máquina de Chave</i>
MRS	<i>MRS Logística</i>
RMN	<i>Rumo Malha Norte</i>
RMO	<i>Rumo Malha Oeste</i>
RMP	<i>Rumo Malha Paulista</i>
RMS	<i>Rumo Malha Sul</i>
SOM	<i>Self-organizing Maps</i>
SOS:D²	<i>System Operational Status Detection and Diagnosis</i>
TKU	<i>Toneladas por Quilômetro Útil</i>
UIC	<i>International Union of Railways</i>

SUMÁRIO

1	INTRODUÇÃO	22
1.1	MOTIVAÇÃO	24
1.2	OBJETIVOS	28
1.3	JUSTIFICATIVA	28
1.4	CONTRIBUIÇÕES	30
1.5	ESTRUTURA DO TRABALHO	30
2	FUNDAMENTAÇÃO TEÓRICA.....	32
2.1	SISTEMA FERROVIÁRIO	32
2.1.1	Sistema de Sinalização Ferroviária	34
2.1.1.1	Aparelhos de Mudança de Via e Máquinas de Chave.....	38
2.2	SISTEMAS CRÍTICOS	42
2.2.1.	Sistemas Tolerantes a falhas	44
2.2.2.	Relação entre Falhas, Erros e Disfunções em Sistemas Críticos	48
2.3	VISÃO GERAL SOBRE MANUTENÇÃO	51
2.3.1	Evolução da Manutenção	53
2.3.2	Tipos de Manutenção	54
2.3.2.1	Manutenção Preventiva	56
2.3.2.1.1	<i>Manutenção preventiva baseada no tempo</i>	<i>57</i>
2.3.2.1.2	<i>Manutenção preventiva baseada em condição.....</i>	<i>59</i>

2.3.2.2	Manutenção Corretiva	61
2.4	DETECÇÃO E DIAGNÓSTICO DE FALHAS	62
2.5	APRENDIZADO DE MÁQUINA: UMA ABORDAGEM SOBRE OS PRINCIPAIS CONCEITOS E TÉCNICAS	64
2.5.1	Classificação	65
2.5.2	Agrupamento de Dados	70
2.5.2.1	O algoritmo <i>K</i> -means	74
2.5.3	Redes Neurais Artificiais.....	76
2.5.3.1	<i>Perceptron</i> Multicamadas	77
2.5.3.2	Mapas Auto-Organizáveis.....	78
2.5.3.3	<i>Autoencoders</i>	80
3	REVISÃO DA LITERATURA.....	82
3.1	VISÃO GERAL SOBRE MANUTENÇÃO BASEADA EM CONDIÇÃO	82
3.2	DETECÇÃO E DIAGNÓSTICO DE FALHAS EM FERROVIAS	83
3.3	COMPARATIVO COM OS TRABALHOS CORRELATOS	87
4	DETECÇÃO E DIAGNÓSTICO DO ESTADO DE OPERACIONALIDADE DE SISTEMAS	91
4.1	A METODOLOGIA PROPOSTA	91
4.1.1	Desenvolvimento dos modelos de Detecção e Diagnóstico de Falhas utilizando técnicas de Aprendizado de Máquina	99
4.1.2	Avaliação dos modelos de Detecção e Diagnóstico de Falhas	115

5	ESTUDO DE CASO	120
5.1	CONTEXTO.....	120
5.1.1	A Companhia Vale S.A	122
5.1.2	Possibilidades de uso dos métodos FDD propostos no sistema de monitoramento dos AMVs da Vale S.A.....	123
5.2	APLICAÇÃO DOS MÉTODO PROPOSTO PARA A DETECÇÃO E DIAGNÓSTICO DO ESTADO OPERACIONAL DOS AMVS DA VALE S.A	124
5.2.1	Conjunto de dados	125
5.2.2	Pré-processamento dos Dados (etapa 1)	127
5.2.3	Desenvolvimento dos Modelos de Detecção e Diagnóstico do Estado de Operacionalidade do AMV (etapa 2)	132
5.2.4	Avaliação dos Modelos de Detecção e Diagnóstico do Estado de Operacionalidade do AMV	137
5.2.4.1	Modelo 1: Obtido a partir do método 1 (combinação das técnicas <i>K</i> -means e SOM)	139
5.2.4.2	Modelo 2: Obtido a partir do método 2 (combinação das técnicas Autoencoder e <i>K</i> -means).....	148
5.2.4.3	Modelo 3: Obtido a partir do método 3 (combinação das técnicas Autoencoder e SOM)	155
5.2.4.4	Avaliação Geral dos Resultados.....	163
5.2.4.5	Resultados emergentes	167
5.2.4.5.1	<i>Inconsistências de Classificação</i>	<i>168</i>

5.2.4.5.2	<i>Diagnóstico de Defeitos: identificação de novas causas</i>	173
6	CONSIDERAÇÕES FINAIS	180
6.1	PRINCIPAIS CONTRIBUIÇÕES	180
6.2	LIMITAÇÕES DO TRABALHO	183
6.3	TRABALHOS FUTUROS	184
7	CONCLUSÃO	186
	REFERÊNCIAS	187
	APÊNDICE A – OTIMIZAÇÃO DOS MODELOS DE CLASSIFICAÇÃO	196
1	Definição da estrutura e/ou dos parâmetros dos modelos	196
2	Busca exaustiva pela melhor combinação dos parâmetros de cada modelo de classificação	198
	APÊNDICE B – COMPORTAMENTO DAS FUNÇÕES DE ATIVAÇÃO E CLASSIFICADORES-BASE UTILIZADOS NO CLASSIFICADOR ENSEMBLE	200
1	Comportamento das Funções de Ativação dos modelos MLP	200
1.1	Função <i>Sigmoide</i>	200
1.2	Função <i>Softmax</i>	201
1.3	Função <i>ReLu</i>	202
2	Classificador <i>Ensemble</i>: Classificadores-base	203

1 INTRODUÇÃO

Ao longo dos últimos anos, devido ao avanço tecnológico, o processo de coleta e armazenamento de dados de aplicações do mundo real tornou-se rápido e prático. Em virtude disso, diariamente, pode-se obter um imenso volume de dados dessas aplicações. Esses dados podem ser utilizados, pela empresa responsável por coletá-los, para auxiliar na tomada de decisão sobre o funcionamento de um sistema e/ou equipamento. Para isto, é fundamental que esses dados sejam transformados, de alguma forma, em informações úteis (conhecimento) sobre esse sistema.

No entanto, apesar da facilidade em adquirir dados de aplicações de domínios reais, tarefas como à análise e extração de conhecimento tornam-se difíceis quando executadas sobre um grande volume de dados. Além disso, a obtenção do conhecimento necessário para compreender o comportamento do sistema e/ou equipamento e, assim, desenvolver algoritmos de aprendizado de máquina eficientes, a partir dos dados coletados, é um processo extremamente custoso – tanto financeiramente quanto em relação ao tempo, principalmente quando realizado manualmente.

Uma forma comum de lidar com esse problema – obtenção de conhecimento – é utilizar as técnicas de aprendizado não supervisionado, as quais aprendem apenas por meio das características dos dados, sem a necessidade de qualquer informação prévia, para extrair informações úteis. Devido a forma como essas técnicas aprendem, podem ser aplicadas tanto em diversos domínios – reais ou artificiais – quanto em diferentes tarefas, tais como detecção de anomalias, detecção de falhas e seleção de atributos (CHANDOLA; BANERJEE; KUMAR, 2009; AGRAWAL; AGRAWAL, 2015; AMRUTHNATH; GUPTA, 2018; LIU et al., 2019). Por essa razão, geralmente, essas técnicas são consideradas como livre de domínio de aplicação.

Por exemplo, no setor industrial, pode-se coletar dados durante o funcionamento de um equipamento e, por meio da análise desses dados, aprimorar o seu desempenho. Inclusive, mediante o uso das técnicas não supervisionadas para a obtenção de informações providas dos dados coletados, é factível prever quando esse equipamento está próximo a ficar indisponível (ou seja, quando se encontra prestes a apresentar uma disfunção), isto é, o momento em que ele não mais cumprirá a função para a qual foi projetado. Atualmente, a manutenção baseada em condição – CBM é uma das estratégias mais comuns de se antever a disfunção de um equipamento (DU; LV; XI, 2012).

Desta forma, na maioria dos casos, é possível substituir o equipamento (realizar a manutenção) de forma programada, com o mínimo impacto operacional, antes que a disfunção efetivamente aconteça. Logo, evita-se a manutenção do equipamento antes do seu desgaste total, caso em que se torna necessário gastos adicionais com a aquisição de um novo equipamento, ou após a disfunção, fato que causa prejuízo devido à indisponibilidade do equipamento ou mesmo pela ocorrência de acidentes.

Todavia, para aplicar a manutenção preditiva (ou CBM), faz-se necessário um processo eficiente de monitoramento das condições de uso dos equipamentos, o qual, normalmente, visa o monitoramento e avaliação de degradação de desempenho para prever e evitar a disfunção do equipamento (DU; LV; XI, 2012). Assim, por meio desses sistemas, é possível utilizar a manutenção preditiva como uma forma de aumentar a disponibilidade e confiabilidade do sistema e/ou equipamento, por meio da detecção e do diagnóstico de falhas, que é realizado por intermédio da análise do registro dessas condições de uso.

Os sistemas de monitoramento das condições de uso de um sistema, inicialmente, tinham como objetivo proporcionar uma forma de analisar os pós-incidentes ou auxiliar no diagnóstico de falhas, de acordo com os dados registrados (CHEN; ROBERTS, 2006). Nesta situação, diz-se que um sistema é tolerante a falha quando o mesmo continua executando suas tarefas, estabelecidas previamente, na presença de falhas no *hardware* ou erros no *software* (JOHNSON, 1989).

Neste contexto, uma falha é qualquer problema físico (imperfeição) que provoque o desvio de desempenho desse equipamento. Isto é, entende-se como uma falha alguma vulnerabilidade em um equipamento ou no próprio *software* do sistema que altera o seu comportamento padrão. A propagação de uma falha pelos componentes de um sistema pode provocar o seu comportamento anormal. Quando esse comportamento se apresenta para os usuários desse sistema de forma incomum, diz-se que ocorreu uma *disfunção* (do inglês *failure*) e denomina-se *erro* (do inglês *error*) a propagação dos efeitos pelo sistema dessa *falha* (do inglês *fault*). Em *dependabilidade*, utiliza-se o termo *disfunção* para representar as situações de mau funcionamento do sistema que, geralmente, podem afetar tanto os seus usuários quanto o ambiente no qual está inserido (J-C et al., 2004).

Além disso, a localização ou identificação de uma falha é o processo de determinar onde aconteceu a falha, objetiva induzir o operador do sistema a atentar-se ao conjunto de variáveis

mais relevantes para o diagnóstico da falha, e a contenção ou isolamento da falha é uma forma de isolar e evitar que os efeitos, causados pela ocorrência da falha, propaguem-se pelo sistema. Sabendo-se disso, a detecção de falhas, como é possível inferir pelo nome, trata-se do procedimento de detectar que uma falha ocorreu no sistema (a detecção prévia da ocorrência de uma falha possibilita executar ações para correção do problema de forma a evitar situações perigosas).

Por outro lado, o diagnóstico de falhas é o processo de identificar o tipo (causa) de falha que aconteceu no sistema, ou seja, apontar o que provocou o mau funcionamento do sistema. Neste caso, é possível evitar, por meio da detecção e o diagnóstico de falhas, que a ocorrência de uma falha, devido a sua manifestação entre os componentes do sistema, ocasione uma disfunção nesse sistema (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000).

Ainda, conforme (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000), em um sistema, o diagnóstico de falhas é fundamental para neutralizá-las ou eliminá-las. Com este propósito, utilizam-se métodos (algoritmos) capazes de detectar e diagnosticar falhas – também chamados de métodos de detecção e diagnóstico de falhas (do inglês *Fault Detection and Diagnosis methods* – FDD), quase sempre de modo prévio à ocorrência de uma disfunção, em equipamentos e/ou sistemas (ISERMANN, 2006). Desta forma, o monitoramento das condições de uso do equipamento, juntamente com esses métodos, pode auxiliar no procedimento de manutenção preditiva do mesmo.

1.1 MOTIVAÇÃO

Devido à quarta revolução industrial ou Indústria 4.0¹, o setor industrial encontra-se em um processo de evolução. Assim, as indústrias precisam participar do desenvolvimento e operação de sistemas autônomos de produção, baseados em sensores e autorregulados. Com isso, poderão usufruir das oportunidades e benefícios dessa abordagem, tais como, produção em massa altamente flexível, coordenação em tempo real e otimização de cadeias de valor e redução de custos de complexidade (HOFMANN; RÜSCH, 2017). No entanto, dentre os desafios dessa nova era está o desenvolvimento de aplicações que utilizam técnicas de tecnologia da informação (TI), automação e controle no processo de produção. Como uma

¹Entende-se como Indústria 4.0 a alteração da lógica de produção, das indústrias existentes, para uma abordagem descentralizada e autorregulada, de modo a auxiliar as empresas a obedecer aos requisitos de produção futuros (HOFMANN; RÜSCH, 2017) .

dessas aplicações, pode-se mencionar a aplicação da *manutenção preditiva* (também chamada de *manutenção baseada em condição – CBM*) em máquinas industriais (BORGI et al., 2017).

Com relação aos custos operacionais em uma indústria, cerca de 15% a 30% dos recursos da empresa são destinados, na média, à manutenção de equipamentos. Além disso, em torno de 5% a 10% de todos os custos operacionais referem-se a manutenções desnecessárias ou indevidas (SILVA NETO, 2014). Logo, esse é um dos motivos pelo qual se faz necessário a busca por alternativas eficientes que visem evitar o desperdício de recursos e/ou a sua melhor utilização. Assim, dado a importância do processo de manutenção, as atividades de manutenção não podem se restringir apenas ao restabelecimento das condições operacionais de um equipamento, mesmo que essas atividades sejam fundamentais. Por isso, além de restaurar o equipamento, as atividades de manutenção necessitam abranger as modificações que devem ser realizadas sobre as condições de uso do equipamento, por meio da introdução de melhorias, para impedir a ocorrência ou reincidência de falhas, diminuir os custos com a manutenção e aumentar a produtividade da empresa (XENOS, 1998).

Desta forma, no contexto industrial, o desenvolvimento de um programa de manutenção preditiva pode proporcionar um aumento da produtividade da empresa, devido ao aumento da confiabilidade e disponibilidade do equipamento, assim como reduz os custos com a manutenção do mesmo, além de evitar as manutenções desnecessárias ou inadequadas. Portanto, a adoção de técnicas de manutenção preditiva pode gerar benefícios financeiros para a empresa. Também pode-se mencionar como melhorias, ocasionadas pela utilização desse programa de manutenção, a redução de estoque de sobressalentes, diminuição do tempo de parada e aumento da vida útil dos equipamentos (SILVA NETO, 2014).

Uma forma de visualizar a redução dos custos com a manutenção no setor industrial, devido às inovações e melhorias nesse processo, é por meio do documento nacional – “A situação da Manutenção no Brasil” – elaborado bienalmente, desde 1985, pela Associação Brasileira de Manutenção e Gestão de Ativos – ABRAMAN. Esse documento apresenta os indicadores de desempenho da “manutenção e gestão de ativos² nos principais setores de

² Segundo Daychoum e Sampaio (2017), ativos, neste caso específico, “são bens que têm apenas uma funcionalidade, não podendo ser reaproveitados”. No contexto ferroviário, ainda conforme Daychoum e Sampaio (2017), o termo ativo compreende “tanto a infraestrutura física da malha ferroviária quanto o material circulante (rodante), tais como, locomotiva, vagões, máquinas de chave, aparelhos de mudança de via, etc.” Sabendo-se disso, no restante do texto, ao utilizar o termo ativo, refere-se a algum desses equipamentos, dependendo diretamente do contexto utilizado.

produtos e serviços” da economia brasileira, assim como representa uma análise detalhada sobre a situação da manutenção em todo o Brasil (ABRAMAN, 2019).

Com base nos resultados obtidos na pesquisa realizada pela ABRAMAN em 2017, como apresentado na *Tabela 1*, é possível notar que, no setor *Mineração e Siderúrgico*, os procedimentos de manutenção corretiva e preventiva representam 83% das atividades realizadas, relacionadas ao controle da manutenção, enquanto que a manutenção preditiva representa apenas 13% dessas atividades. Com relação aos custos da manutenção, na *Tabela 2* observa-se que o setor *Mineração e Siderúrgico* representa apenas 5% do custo de manutenção/faturamento dentre todos os setores analisados na pesquisa (ABRAMAN, 2019).

Tabela 1. Controle da Manutenção de acordo com a forma como o reparo foi realizado (Fonte: (ABRAMAN, 2017)).

Setores	Manutenção Corretiva (%)	Manutenção Preventiva (%)	Manutenção Preditiva (%)	Outras atividades de manutenção (%)
Açúcar e Alcool, Alimentos e Bebidas	42 %	40 %	18 %	0 %
Aeronáutico e Automotivo	42 %	48 %	4 %	6 %
Eletroeletrônicos	28 %	44 %	12 %	16 %
Químico e saneamento	25 %	50 %	20 %	5 %
Mineração e Siderúrgico	60 %	23 %	13 %	4 %
Petróleo e Petroquímico	47 %	39 %	14 %	0 %
Papel, Celulose e Plástico	24 %	31 %	31 %	14 %
Predial e Prestação de Serviços	45 %	44 %	5 %	6 %
Máquinas e Equipamentos - Metalúrgico	25 %	48 %	7 %	20 %

Tabela 2. Custos de Manutenção (Fonte: (ABRAMAN, 2017)).

Setores	Custo Manutenção/ Faturamento (%)	Custo Manutenção/ Valor Imobilizado (%)	Custo Relativo Pessoal Próprio	Custo Relativo Material (%)	Custo Relativo à Contratação (%)	Outros Custos Relativos (%)
Açúcar e Alcool, Alimentos e Bebidas	4 %	6 %	40 %	40 %	20 %	0 %
Aeronáutico e Automotivo	3 %	13 %	54 %	32 %	15 %	0 %
Eletroeletrônicos	7 %	6 %	44 %	29 %	27 %	4 %
Químico e saneamento	5 %	5 %	46 %	38 %	16 %	0 %
Mineração e Siderúrgico	5 %	10 %	50 %	37 %	14 %	0 %
Petróleo e Petroquímico	4 %	15 %	41 %	31 %	28 %	0 %
Papel, Celulose e Plástico	3 %	8 %	39 %	49 %	12 %	6 %
Predial e Prestação de Serviços	4 %	4 %	49 %	23 %	28 %	2 %
Máquinas e Equipamentos - Metalúrgico	3 %	1 %	51 %	37 %	12 %	3 %

Diante dos argumentos apresentados, é possível perceber que um programa de manutenção preditiva pode fornecer diversos benefícios que, geralmente, podem refletir na redução dos custos operacionais das empresas e no aumento da qualidade dos produtos e serviços oferecidos aos seus clientes. Neste contexto, a redução dos custos pode ocorrer tanto de forma direta (relaciona-se com a diminuição dos gastos empresariais, provenientes de fatores internos, tais como, peças, equipamentos e equipe de manutenção) quanto de forma indireta (relaciona-se com a diminuição dos gastos empresariais, oriundos de fatores externos, tais como, pagamentos de multas e ressarcimentos resultantes da valorização global, devido ao aumento da qualidade, dos produtos e dos serviços fornecidos pela empresa) (SILVA NETO, 2014).

Além disso, conforme evidenciado por Bian et al. (2019), “explorar simultaneamente os estados de degradação em diferentes modos de falha e identificá-los com precisão é um problema chave a ser resolvido”. Segundo Carvalho et al. (2019), as pesquisas atuais devem “desenvolver trabalhos que comparem a estratégia de manutenção preditiva proposta a diferentes algoritmos de ML (Aprendizado de Máquina, em inglês *Machine Learning – ML*), além de usar novos algoritmos de ML, como algoritmo de aprendizado profundo (do inglês, *Deep Learning*)”.

Isto posto, o presente trabalho aborda assuntos importantes e atuais para a proposição de algoritmos tanto para a detecção e diagnóstico, de modo eficiente, dos diversos estados de operacionalidade (degradação) de um equipamento – problema de pesquisa abordado – quanto com relação a aplicação da manutenção preditiva sobre um determinado equipamento. Por fim, esses métodos são aplicados em um estudo de caso real.

1.2 OBJETIVOS

O objetivo principal deste trabalho é propor métodos preditivos, baseando-se em técnicas de aprendizado de máquina, capazes de **detectar e diagnosticar** o estado de operacionalidade (degradação) de um sistema e/ou equipamento, por intermédio tanto da análise do histórico de dados de operação quanto da análise de dados em tempo real.

Um método proposto neste trabalho é constituído pela combinação de técnicas de Aprendizado de Máquina. Desta forma, no contexto do setor ferroviário, objetiva-se a proposição de métodos para detectar e diagnosticar o estado de degradação de um Aparelho de Mudança de Via – AMV, por meio da ocorrência de anomalias na assinatura elétrica da corrente de operação da *Máquina de Chave* do AMV, baseando-se em seu histórico de operação. Esses métodos poderão ser utilizados como uma ferramenta, no âmbito de um sistema de sinalização ferroviária, para auxiliar no processo de manutenção desse equipamento, contribuindo com o aumento da produtividade (disponibilidade da via) e da segurança operacional do transporte ferroviário. Neste contexto, o diagnóstico de uma falha restringe-se ao esclarecimento do tipo de falha, visto que não compreende ao escopo do presente trabalho a contenção – de forma direta – da falha que gerou a disfunção do equipamento.

Um objetivo secundário do presente trabalho é apresentar uma análise comparativa entre os métodos propostos. Desta forma, espera-se como resultado desta análise obter um *ranking*, criado de acordo com o desempenho de cada método (algoritmo), que apresente os melhores algoritmos para o processo de detecção e diagnóstico do nível de degradação de um AMV no setor ferroviário.

1.3 JUSTIFICATIVA

Os métodos propostos nesta pesquisa poderiam funcionar como um *componente* de um programa de gerenciamento de manutenção de AMVs, em um sistema de sinalização ferroviária. Nesta situação, poderiam influenciar diretamente na tomada de decisão sobre as

práticas de manutenção preditiva desse equipamento, podendo ser utilizado como uma ferramenta que auxilia no processo de detecção e diagnóstico do nível de degradação (desgaste) do ativo. Desta forma, deseja-se alcançar, como resultado da aplicação desses métodos, os benefícios listados a seguir:

- 1- Diminuição dos custos de manutenção associados ao reparo das disfunções. Além disso, tem-se uma redução dos custos associados aos períodos de indisponibilidade da ferrovia devido a disfunção dos AMVs;
- 2- A antecipação da ocorrência de uma disfunção proporciona uma redução dos chamados de emergência. Assim, pode-se minimizar a probabilidade de incidentes, resultantes do desgaste parcial ou completo do ativo, que causam a indisponibilidade da via;
- 3- A capacidade de detectar, com um alto nível de confiança, o desgaste temporal do ativo antes da ocorrência de uma disfunção, provocando a indisponibilidade na ferrovia, permite otimizar o processo de reparo desse equipamento. Assim, detalhes importantes como a alocação de recursos humanos e financeiros pode ser realizada de forma programada. Além disso, devido ao conhecimento sobre o tipo de falha que provavelmente irá provocar uma disfunção no equipamento, pode-se designar uma equipe de manutenção para a execução do procedimento de acordo com as necessidades identificadas, dimensionando-se adequadamente fatores como a quantidade de integrantes da equipe e suas competências técnicas;
- 4- Melhoria no processo de classificação do estado operacional de um AMV, em que, ao invés de apenas classificá-lo como operacional ou indisponível, especifica-se o que ocasionou o mau funcionamento do ativo. Neste contexto, tem-se três categorias que especificam o estado do ativo. São elas: normal (o ativo está operando conforme o esperado); operacional com degradação (o ativo continua operacional, porém, com desempenho abaixo do esperado); e indisponível (o ativo não é capaz de operar e, conseqüentemente, pode provocar acidentes). Além disso, para cada AMV cujo estado operacional foi classificado como operacional com degradação, tem-se a especificação do tipo de falha que levou a máquina para este estado.

1.4 CONTRIBUIÇÕES

As principais contribuições esperadas do presente trabalho de pesquisa são:

- 1- Levantamento do cenário atual do estado-da-arte em manutenção preditiva (baseada em condição) aplicada, principalmente, no setor industrial;
- 2- Análise e apresentação dos principais tipos de falhas em AMVs;
- 3- Proposição de um conjunto de métodos para auxiliar, como uma ferramenta de apoio ao especialista de domínio no processo de classificação do estado de operacionalidade do ativo, na aplicação da manutenção preditiva em AMVs. Esta abordagem possibilita a alocação da mão de obra necessária de acordo com o tipo de falha. Além disso, pode proporcionar um aumento da segurança operacional, incluindo a operação de manutenção e disponibilidade da ferrovia;
- 4- Apresentação de uma análise comparativa entre métodos de detecção e diagnóstico de falhas, utilizados para identificar o nível de operacionalidade do ativo, aplicados em um estudo de caso real.

1.5 ESTRUTURA DO TRABALHO

O restante desta dissertação está organizado como segue. O Capítulo 2 apresenta os principais conceitos utilizados para o desenvolvimento deste trabalho. Desta forma, inicialmente, apresentam-se os conceitos fundamentais sobre o sistema ferroviário. Em seguida, abordam-se os sistemas críticos e suas características. Por fim, introduz-se o conceito de manutenção, assim como as principais formas de se realizar a manutenção de um equipamento. Além disso, apresenta-se os conceitos relevantes de aprendizado de máquina, incluindo os métodos de detecção e diagnóstico de falhas, abordados no presente trabalho.

No Capítulo 3, mostram-se os resultados obtidos na revisão da literatura. Inicialmente, descrevem-se as pesquisas que abordam a manutenção preditiva, de modo a aprimorar o conhecimento nessa área. Posteriormente, apresentam-se diversos trabalhos correlatos à esta dissertação. Neste caso, buscou-se tanto por pesquisas que objetivam analisar/propor métodos de detecção e diagnóstico de falhas em qualquer domínio de aplicação quanto por trabalhos que visam a aplicação desses métodos sobre o sistema ferroviário. Por fim, baseando-se em suas

características, realiza-se um comparativo entre os métodos aplicados, especificamente, sobre o sistema ferroviário.

No Capítulo 4, apresenta-se a metodologia utilizada para a Detecção e Diagnóstico do estado de Operacionalidade de Sistemas. Primeiramente, mostra-se uma visão geral do processo de detecção e diagnóstico proposto. Em seguida, explora-se a sistemática tanto de desenvolvimento quanto de avaliação dos métodos propostos nesta pesquisa.

O Capítulo 5 analisa os resultados obtidos com a aplicação dos métodos propostos no Capítulo 4 sobre o estudo de caso apresentado. A princípio, descreve-se o contexto de operação de uma ferrovia, incluindo as possibilidades de aplicação desses métodos. Além disso, apresenta-se tanto o conjunto de dados utilizado quanto a aplicação do processo de desenvolvimento e avaliação dos modelos de Detecção e Diagnóstico de Falhas – FDDs.

O Capítulo 6 apresenta as conclusões deste trabalho de investigação científica. Portanto, abordam-se as principais contribuições e limitações deste trabalho, além das possibilidades de linhas de pesquisa para trabalhos futuros.

O Apêndice A apresenta o processo de otimização dos modelos de classificação, complementando o Capítulo 4 deste trabalho. Basicamente, nesse apêndice, descreve-se as duas etapas que constituem esse processo de otimização, a saber, especificação da estrutura e dos parâmetros de cada modelo de classificação e busca exaustiva pela melhor combinação desses parâmetros.

No Apêndice B descreve-se o comportamento das funções de ativação *Sigmode*, *ReLu* e *Sofmax*, as quais foram utilizadas no desenvolvimento dos modelos de classificação baseados na estrutura da rede neural *Multilayer Perceptron* – *MLP*. Além disso, apresentam-se alguns conceitos relevantes sobre o desenvolvimento do classificador *ensemble* apresentado no Capítulo 4, incluindo os classificadores-base utilizados na composição desse classificador.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo define os conceitos teóricos fundamentais para a compreensão e desenvolvimento do presente trabalho. Primeiramente, apresenta-se uma visão geral sobre o sistema ferroviário, especificando as características e os elementos principais de um sistema de sinalização ferroviária. Em seguida, abordam-se os conceitos de sistemas críticos, discutindo sobre as suas principais características e, em especial, o conceito de segurança nesses sistemas. Desta forma, define-se o conceito de *sistemas críticos em segurança (safety)*, assim como os conceitos diretamente relacionados à segurança crítica, tais como confiabilidade, disponibilidade e manutenibilidade. Posteriormente, disserta-se sobre a evolução da manutenção industrial, dado a sua relação com o ambiente no qual o estudo de caso será realizado e, posteriormente, descreve-se os principais tipos de manutenção aplicados na indústria. Por fim, apresenta-se os conceitos relevantes tanto de detecção e diagnóstico de falhas quanto de aprendizado de máquina abordados nesta dissertação.

2.1 SISTEMA FERROVIÁRIO

Dentre as principais características e vantagens do sistema de transporte ferroviário, tem-se à ampla capacidade de transporte tanto de cargas quanto de passageiros e a ótima relação custo-benefício de transporte em trajetos de média e longa distância em relação aos demais meios de transporte terrestre (BEAMON et al., 2002; ANTT, 2019). Em virtude disso, esta subseção descreve a estrutura do sistema ferroviário, incluindo seus principais componentes, tais como equipamentos e sistemas.

Entende-se como ferrovia ou via permanente a estrutura na qual os trens transitam, seja por meio da tração a diesel (locomotiva independente, sem alimentação externa) ou da tração elétrica (locomotiva através da transmissão de potência, realizada remotamente) (PYRGIDIS, 2016). A via permanente, conforme (KLINCEVICIUS, 2011), representa o aglomerado de camadas e elementos que tornam viável a circulação dos trens.

Sabendo disso, o principal objetivo da via férrea é viabilizar, de forma segura e econômica, o transporte dos trens. Por causa disso, é necessário que a ferrovia seja tanto estável quanto alinhada proporcionalmente em ambos os sentidos, isto é, verticalmente e horizontalmente (SELIG, ERNEST T AND WATER, 1995). Consequentemente, o

desempenho da ferrovia, em prover o transporte dos trens, relaciona-se diretamente com o alinhamento do sistema de camadas que a compõe (KLINCEVICIUS, 2011).

No entanto, para que a ferrovia consiga exercer essa função, é essencial que todos os seus componentes funcionem conforme o esperado. Em outras palavras, cada componente da via permanente deve executar sua função sem nenhuma anormalidade, mesmo estando sujeitos a desgastes que podem ser originados tanto pelas condições climáticas, visto que estão expostos ao ambiente, quanto pelo peso das cargas transportadas nos trens, devido ao tráfego constante (SELIG, ERNEST T AND WATER, 1995).

Ainda conforme Selig, Ernest T e Water (1995), esses componentes podem ser divididos em duas categorias: a infraestrutura, composta pela camada de sub-leito e, normalmente, as demais obras realizadas abaixo dessa camada de terraplanagem, isto é, em solo natural, em alguns casos, engloba também as camadas de lastro (KLINCEVICIUS, 2011), e a superestrutura, composta pelas camadas de sub-lastro e lastro, em conjunto com os trilhos, dormentes e as fixações, os quais juntos formam a grade da via (KLINCEVICIUS, 2011).

A *Figura 1*, apresenta os principais componentes da via permanente, tais como os trilhos, dormentes, fixações e placas de apoio, lastro, sub-lastro e sub-leito (SELIG, ERNEST T AND WATER, 1995). Conforme a figura, é possível observar que a via permanente é composta por dois trilhos, alinhados paralelamente e separados por uma distância específica, presos aos dormentes, que garantem a preservação da distância entre os trilhos, por meio das fixações e placas de apoio. Os dormentes são colocados espaçadamente, de acordo com uma distância previamente determinada, sobre a camada de lastro, que proporciona tanto o nivelamento da superfície quanto uma drenagem uniforme, além de manter a distância delimitada para os dormentes e transferir o peso das cargas para uma área mais ampla (camadas inferiores) (HAY, 2003). Por fim, baseando-se em uma análise hierárquica com início na grade da via, os trilhos transmitem o peso da carga para os dormentes, que por sua vez, transferem esse peso para a camada de lastro, e assim sucessivamente. Logo, a camada de sub-leito encarrega-se de transportar o peso total da ferrovia, incluindo o trem que está se movimentando sobre ela (HAY, 2003).

Figura 1. Principais componentes da Via Permanente (**Fonte:** (KLINCEVICIUS, 2011)).

Dito isso, na visão de um sistema de transporte, o sistema ferroviário de transporte é, comumente, constituído por três componentes. São eles: a infraestrutura ferroviária, o material rodante (veículos ferroviários) e a operação ferroviária (HAY, 2003). Neste contexto, ainda conforme (HAY, 2003), a infraestrutura ferroviária representa tanto a via permanente quanto as construções de engenharia civil e sistemas que asseguram o a circulação dos trens. A *Figura 2* apresenta os principais componentes, incluindo alguns de seus elementos da infraestrutura ferroviária.

Figura 2. Principais componentes da infraestrutura ferroviária (**Fonte:** Adaptado de (HAY, 2003)).

Vale ressaltar que, assim como a via permanente, os sistemas são fundamentais no transporte ferroviário, principalmente o de sinalização. Por conta disso, a subseção seguinte aborda esse sistema, especificando seus elementos e características.

2.1.1 Sistema de Sinalização Ferroviária

O sistema ferroviário é relativamente diferente dos demais meios de transporte, e uma das principais razões para isso é a diferença existente entre os veículos utilizados na ferrovia e os veículos convencionais (carros, ônibus e caminhões), pois fatores como o tamanho do veículo afetam tanto a capacidade de aceleração quanto de desaceleração dos trens. Por

exemplo, durante o trajeto realizado por um trem é inviável e, praticamente impossível, que ele desacelere imediatamente para evitar uma colisão, visto que ao frear bruscamente, com o peso advindo da carga transportada, pode-se provocar um descarrilamento.

Portanto, um veículo ferroviário não pode parar de imediato para impedir a colisão com objetos, incluindo equipes de manutenção e/ou outros veículos que estão na via, visto que o maquinista (condutor), normalmente, não consegue visualizar, em tempo hábil de realizar o procedimento de parada, longas extensões da ferrovia. Desta forma, se faz necessário que exista um “supervisor externo”, o qual atua no controle da circulação, que avise antecipadamente, e de modo constante durante a realização do trajeto, ao maquinista se existem ou não objetos ou veículos ao longo do percurso. Neste contexto, uma das funções do sistema de sinalização ferroviário é fornecer autoridade de movimentação ao condutor do trem. Isto é, esse sistema atua como o “supervisor externo” durante a movimentação dos trens (BELLEK, 2013).

Além disso, por questões de segurança, os equipamentos da via devem ser livres de disfunções, ou seja, devem ser mantidos em um estado de boa operacionalidade, pois a ocorrência de uma disfunção em equipamentos, tais como Máquina de Chave – MCH e AMV, pode provocar descarrilamento ou colisões. Devido a isso, a movimentação segura dos trens é o principal objetivo de um sistema de sinalização ferroviária (BELLEK, 2013). Ainda com relação à segurança operacional, e de acordo com as características apresentadas, o sistema de sinalização ferroviária possibilita um aumento na capacidade de operação, pois permite aos trens usufruírem de todo o seu potencial de transporte, assim como otimiza a utilização da via (BELLEK, 2013).

Em alguns casos, a manifestação de uma ou de múltiplas falhas nos componentes do sistema de sinalização ferroviária pode alcançar cenários potencialmente perigosos, tanto para as pessoas que se relacionam com o sistema quanto para o ambiente no qual ele está inserido. Nestes casos, diz-se que o **sistema é crítico em relação à segurança** e, conseqüentemente, a aplicação que utiliza esse sistema é uma **aplicação crítica em relação à segurança** (VISMARI, 2007). Por isso, o sistema ferroviário é considerado um domínio de aplicação crítica, pois utiliza um sistema crítico, neste caso, o *sistema de sinalização ferroviária*.

Sabendo-se disso, o princípio operacional básico do sistema de sinalização é monitorar a movimentação de todos os veículos em via, verificar o estado de operacionalidade de cada equipamento e prover autoridade de movimentação aos trens, de modo a aumentar a segurança

e a capacidade de operação (BELLEK, 2013). De modo geral, alguns dos principais objetivos de um sistema de sinalização, conforme (HAY, 2003), são:

1. Monitorar o movimento dos trens, de tal modo que eles se movimentem, com segurança, na velocidade máxima determinada para o percurso;
2. Preservar a distância, considerada segura, entre dois ou mais trens na via que estejam no mesmo percurso e na mesma direção;
3. Assegurar a segurança dos trens em cruzamentos, incluindo os casos em que os trens devem cruzar vias urbanas, ou rodovias, onde veículos convencionais trafegam;
4. Prover instalações para manobras seguras e eficientes, assim como controlar as chegadas e partidas dos trens nos pátios e/ou estações.

Por fim, os principais componentes de um sistema de sinalização ferroviária são:

Centro de Controle de Operação – CCO. O CCO é o responsável por monitorar e controlar todos os processos, realizados na ferrovia, por um sistema de sinalização ferroviário. Ou seja, as posições e a necessidade ou não de manutenção dos ativos (estado de operacionalidade), de todos os equipamentos da via, tais como MCHs, AMVs e os sinais envolvidos, são monitorados e controlados pelo CCO (DINCEL; ERIS; KURTULAN, 2013).

Vale ressaltar que existe uma relação bidirecional e fundamental entre o CCO e o sistema de intertravamento (descrito a seguir). Esta relação possui uma troca sincronizada de dados (informações). Geralmente, o CCO envia requisições para serem validadas pelo sistema de intertravamento e este, por sua vez, avalia essas requisições e, se forem viáveis, aplica sobre os equipamentos da via. Em seguida, o sistema de intertravamento reporta ao CCO um conjunto de informações, como se fosse um *feedback* com os *status* das requisições, provindas da ferrovia (DINCEL; ERIS; KURTULAN, 2013).

Sistema de Intertravamento. O objetivo principal do sistema de intertravamento é assegurar a segurança dos trens (HAY, 2003). Desta forma, é utilizado no sistema de sinalização ferroviária como uma forma de conectar o CCO com a ferrovia em tempo real. Por isso, geralmente é considerado como uma “ponte entre o CCO e o campo” (DINCEL; ERIS; KURTULAN, 2013).

Neste contexto, como mencionado anteriormente, o sistema de intertravamento analisa as requisições do CCO e, baseando-se nos dados obtidos da ferrovia, tais como a posição de travamento de uma MCH (o significado desse termo é descrito ao longo desta subseção), se existem ou não trens e/ou objetos no percurso, e os sinais atuais apresentados na via, verifica se a aplicação desses comandos é adequada, visando identificar movimentos conflitantes e assegurar uma distância segura entre os trens. Caso seja viável, os comandos solicitados são enviados para serem executados sobre os equipamentos da via. Caso contrário, os comandos são rejeitados e informações relevantes são enviadas, em forma de retorno as requisições, ao CCO (DINCEL; ERIS; KURTULAN, 2013).

Circuitos de Via – CDV. A principal função de um circuito de via (CDV), um componente elétrico acoplado a via, é identificar a ausência de trens ao longo da ferrovia. Geralmente, são espaçadamente distribuídos na ferrovia, por trechos, e separados por juntas isolantes. Esse componente, para detectar a existência de um trem naquele trecho, entra em curto-circuito enquanto o trem passa sobre ele (DINCEL; ERIS; KURTULAN, 2013). Vale ressaltar que o CDV é considerado um componente de falha segura, pois caso o sinal (também chamado de corrente curto-circuito) não seja recebido, leva o trecho ao estado ocupado, e pode ser utilizado tanto para identificar trilhos partidos quanto para “rastrear”, pela ocupação e desocupação dos CDVs, um trem (VALE, 2017).

Detector de Descarrilamento - DD. Um detector de descarrilamento, como o próprio nome indica, é um equipamento de segurança da via responsável por identificar se um descarrilamento ocorreu. Trata-se de uma barra metálica, posta na via, que faz parte de um circuito elétrico. Dessa forma, quando esse circuito é interrompido, um alarme é gerado de modo a informar que, provavelmente, aconteceu um acidente (VALE, 2017).

Rotas. Uma rota pode ser definida como uma sequência ordenada de trechos, que devem ser percorridos pelos trens, que possuem um ponto inicial e final. As rotas normalmente são definidas pelo CCO e possibilitam que os trens circulem, de forma ordenada e não conflitante, entre os pátios e/ou estações (KANSO; MOLLER; SETZER, 2009).

Sinais (ou Sinaleiros). Um sinal (ou sinaleiro) é um equipamento básico da ferrovia que funciona como uma interface entre os equipamentos técnicos e as pessoas. Neste caso, é por meio das luzes de sinalização apresentadas nos sinais que o maquinista do trem ou a própria equipe de manutenção recebem as informações advindas do sistema de sinalização ferroviário

(BELLEK, 2013). Geralmente são informações sobre as próximas rotas, permissão para continuar em direção à rota seguinte e informações sobre a velocidade do trem (DINCEL; ERIS; KURTULAN, 2013).

Outros dois componentes fundamentais no contexto do sistema ferroviário são os *Aparelhos de Mudança de Via – AMVs* e as *Máquinas de Chave – MCHs* que juntos possibilitam a alternância de rotas. Consequentemente, devido a relação direta com o presente trabalho, esses componentes são detalhadamente descritos na subseção a seguir.

2.1.1.1 Aparelhos de Mudança de Via e Máquinas de Chave

O AMV é o ativo (equipamento) da ferrovia responsável por interligar duas vias adjacentes e, assim, permite que o trem mude de percurso. Contudo, apenas com a utilização de um AMV não é possível realizar esse processo e, consequentemente, faz-se necessário utilizar um conjunto de outros equipamentos, dentre eles a MCH, a qual é utilizada para operar o AMV. Desta forma, existe uma relação de dependência entre os AMVs e as MCHS. Nesta relação, o AMV é o ativo dependente. Em alguns casos, esses componentes são considerados como parte de um sistema responsável pela alternância do percurso (VILEINISKIS; REMENYTE-PRESCOTT; RAMA, 2016b).

O AMV, no contexto operacional da ferrovia, tem como objetivo conectar duas vias e possibilitar o direcionamento de um trem em uma mudança de rota (alternância entre vias). A *Figura 3* ilustra, em alto nível, os componentes do AMV, neste caso, incluindo a máquina de chave. De modo geral, considerando a MCH como um componente individual, diz-se que o AMV é composto por três componentes. São eles: (*Figura 3 – Item 5*) Agulhas, exatamente duas, as quais são encarregadas por direcionar o trem para o caminho correspondente a configuração requerida, isto é, guiar o trem para outra via (reverso) ou apenas servir de suporte para a passagem do trem que deve seguir em linha reta (normal); (*Figura 3 – Item 7*) Jacaré, que viabiliza a passagem das rodas dos trens enquanto o mesmo está passando pelo AMV, e corresponde à junção entre os trilhos das vias adjacentes (principal e desvio), e, por fim; (*Figura 3 – Item 4*) Trilho de proteção (contratrilha), um pequeno pedaço de trilho posicionado na lateral dos trilhos externos da via, que serve para auxiliar na passagem das rodas do trem, localizando-se opostas ao Jacaré (HAGA; ARGENTON E; BARBOSA, 2004).

Figura 3. Componentes do Aparelho de Mudança de Via (AMV), sendo (1) a máquina de chave; (2) encosto da agulha; (3) barra de operação; (4) trilho de proteção (contratrilho); (5) agulhas do AMV; (6) cesto da barra de operação; e (7) jacaré (**Fonte:** Autor).

Neste contexto, um AMV pode assumir duas configurações operacionais básicas: reverso, em que um trem viajando sobre o AMV é redirecionado para a via adjacente. Ou seja, quando em configuração reverso, os trilhos (também chamados de agulhas) do AMV são alinhados para que o trem possa ser transferido para a outra via (*Figura 5*); e normal, na qual um trem viajando sobre um AMV permanece na mesma via, isto é, mantém seu caminho pela mesma via em que está posicionado (*Figura 4*). A configuração de um AMV é definida por meio de um motor elétrico comandado pelo sistema de sinalização ferroviária, denominado Máquina de Chave. Assim, a máquina de chave (1) move as agulhas do AMV (5), conectadas pela barra de operação (3), até que elas se alinhem com a direção do movimento, normal ou reverso, solicitada pelo sistema de sinalização ferroviária ou realizada de modo manual por um operador local.

Figura 4. Configuração *Normal* do AMV: Simulação do movimento do trem (**Fonte:** Adaptado de (FUN, 2018)).

Figura 5. Configuração *Reverso* do AMV: Simulação do movimento do trem (**Fonte:** Adaptado de (FUN, 2018)).

Em relação as máquinas de chave, geralmente, são: a) de operação manual (local ou remota por meio de uma manivela); b) elétricas; c) de mola; e d) hidráulicas (estas três últimas operadas pelo sistema de sinalização ferroviária, especificamente, o sistema de intertravamento) (VALE, 2017). A *Figura 6*, apresenta uma MCH elétrica. Normalmente, uma MCH é composta: por um motor, de corrente contínua, que realiza a movimentação das agulhas do AMV; por uma unidade controladora do motor – CNAP, que controla o funcionamento do motor, sendo responsável por determinar o sentido de rotação do motor e encerrar a alimentação do motor (nos casos em que a alimentação do motor ultrapassar o limite de tempo de uma movimentação comum da MCH, normalmente 8 segundos (incluindo 1 segundo de tolerância)); e por um redutor, que possui diversas engrenagens, incluindo uma embreagem eletromagnética, encarregada de realizar o acoplamento entre as engrenagens primárias e secundárias com o pinhão da embreagem (HAGA; ARGENTON E; BARBOSA, 2004).

Figura 6. Máquina de Chave Elétrica (**Fonte:** Vale S.A).

Com relação ao funcionamento de uma MCH, comumente, existem pelo menos três tipos de falhas que podem afetar a sua operação, são elas: abruptas (ocorrem repentinamente, sem qualquer indício prévio); intermitentes (acontecem esporadicamente); e incipientes (ocorrem de forma gradual durante um período de tempo). Como consequência, o surgimento de uma dessas falhas pode ocasionar a disfunção da MCH e, dessa forma, a indisponibilidade desse ativo (ASADA; ROBERTS, 2011).

Tanto as falhas abruptas quanto as intermitentes, devidos às suas características, são difíceis de prever. Por exemplo, a MCH pode estar funcionando perfeitamente e de repente, após a ocorrência de uma falha, pode parar de funcionar sem qualquer indicação prévia de que a mesma poderia falhar (ASADA; ROBERTS, 2011). Note que esse comportamento é característico de uma falha abrupta, geralmente associadas a obstruções físicas no movimento da MCH.

No caso de uma falha intermitente, existe a possibilidade das implicações, que resultaram no surgimento da mesma, também sumirem esporadicamente. Logo, não é possível identificar previamente as causas que levaram ao acontecimento da mesma. Já no caso das falhas incipientes, o desvio de desempenho vai aumentando gradualmente e, por conta disso, é possível antever esse tipo de falha, geralmente efetuado por meio de um processo adequado de monitoramento (ASADA; ROBERTS, 2011).

Dada essa visão geral da relação entre as MCHs e os AMVs, incluindo a descrição de seus componentes, explica-se a seguir o processo de mudança de rota. A *Figura 7* apresenta o cenário comum de um trecho que possui um AMV. Observe-se que, neste pequeno trecho, existem 4 AMVs e, conseqüentemente, 4 MCHs. Essas máquinas de chave, neste caso, são conjugadas em pares (W21A e W21B, W22A e W22B). Desta forma, um único comando aciona as duas MCHs, que operam os AMVs, de forma simultânea.

Por fim, em prol da compreensão do processo de alternância de rotas, considere o seguinte cenário fictício: Suponha que um trem esteja se locomovendo pelo *Trecho D* e tem como destino o *Trecho C* (Rota). Os comandos necessários para a realização dessa rota são enviados ao sistema de intertravamento pelo CCO, que monitora e controla o trem durante todo o percurso, o qual valida os comandos requisitados e, caso seja viável, envia um sinal elétrico informando a posição na qual as agulhas do AMV devem ser posicionadas (comando de chave). A MCH, por sua vez, recebe esse comando e posiciona corretamente as agulhas do AMV na

configuração requerida, Normal ou Reverso (neste caso é reverso), trava mecanicamente o AMV, que assegura o travamento das agulhas até o próximo comando de movimentação, e retorna um sinal com a posição das agulhas para o sistema de intertravamento. Finalmente, o trem realiza esse desvio e consegue completar a rota.

Figura 7. Exemplificação de um trecho de uma ferrovia, representado no sistema de intertravamento, com um ponto de mudança de via (**Fonte:** Vale S.A).

2.2 SISTEMAS CRÍTICOS

Quando um sistema não consegue funcionar conforme o esperado, isto é, não realiza corretamente a função para a qual foi projetado, diz-se que esse sistema apresentou um comportamento atípico (não atendeu aos requisitos definidos). Nessas ocasiões, o funcionamento anormal do sistema pode provocar, nas pessoas envolvidas com o sistema ou no ambiente no qual está inserido, consequências tanto intoleráveis quanto toleráveis (de proporções menores) pela sociedade.

Qualquer sistema, seja natural ou artificial – sistema desenvolvido pelo homem –, expõe o ambiente ao qual está inserido a algum tipo de risco inerente (conforme o tipo de aplicação compreendida pelo sistema) (ALMEIDA JR, 2003). Em razão disso, ainda conforme (ALMEIDA JR, 2003), o risco representado por um sistema é também definido pela sociedade, “que em última análise, é quem define o conceito de risco aceitável”.

Isto posto, faz-se necessária a realização de uma análise dos benefícios gerados pelo sistema e do grau de risco que a sociedade está disposta a correr por esses benefícios. Inclusive, uma parte significativa dos sistemas existentes foi desenvolvida visando reduzir o risco inerente das aplicações críticas, nas quais o risco está diretamente relacionado à segurança dos envolvidos com a aplicação (pessoas, propriedades, ambiente), a um nível de risco aceitável pela sociedade.

O risco relaciona-se diretamente com a ocorrência frequente de eventos considerados perigosos e com a severidade das consequências que podem ser geradas pelo sistema (ALMEIDA JR, 2003). Em alguns casos, o sistema não consegue desempenhar suas funções de forma apropriada, na maioria das vezes, devido ao fato de ter ocorrido uma falha em pelo menos um de seus componentes. Por essa razão, os riscos apresentados pelo sistema estão diretamente relacionados com essas situações de prováveis acidentes (SILVA NETO, 2014).

Os sistemas definidos como *Sistemas Críticos*, nos quais a ocorrência de um funcionamento anormal pode causar consequências socialmente inaceitáveis, prejudicam tanto as pessoas que, de alguma forma, interagem com o sistema, quanto o ambiente em que estão inseridos (ALMEIDA JR, 2003). Portanto, um sistema crítico é um tipo de sistema que foi projetado para garantir a segurança da operação no ambiente em que está inserido (BOZZANO; VILLAFIORITA, 2010).

Vale ressaltar que, dadas suas características individuais, nem todos os sistemas críticos provocam as mesmas consequências na ocorrência de uma disfunção. Por esse motivo, diz-se que esses sistemas não são igualmente críticos quando apresentam uma disfunção. Uma forma de analisar os diferentes níveis de criticidade de sistemas críticos é observar as consequências relacionadas ao funcionamento anormal (disfunção) desses sistemas (BOZZANO; VILLAFIORITA, 2010).

Os três tipos principais de *sistemas críticos*, baseando-se no nível de criticidade desses sistemas, são descritos a seguir (SOMMERVILLE, 2011):

- *Sistemas Críticos em Negócios (Business Critical Systems)*: sistemas nos quais o funcionamento atípico pode ocasionar prejuízos financeiros aos negócios;
- *Sistemas Críticos em Missão (Mission Critical Systems)*: sistemas nos quais o funcionamento atípico pode provocar o não cumprimento de uma missão; e
- *Sistemas Críticos em Segurança (Safety Critical Systems)*: sistemas nos quais o funcionamento atípico pode ocasionar danos significativos à vida humana, tais como morte ou ferimentos em pessoas, ou ao meio ambiente.

Baseando-se na definição apresentada para os três principais tipos de sistemas críticos, evidencia-se que esses sistemas, via de regra, podem afetar apenas as pessoas que interagem de alguma forma com estes sistemas (*sistema críticos em negócio e missão*) ou podem afetar todo

o ambiente no qual está inserido, inclusive, as pessoas envolvidas com o sistema (*sistemas críticos em segurança*) (VISMARI, 2007).

A severidade das aplicações que utilizam sistemas críticos (denominadas *aplicações críticas*), cuja disfunção pode provocar acidentes que, geralmente, causam sérios danos materiais, ambientais ou expõem à vida humana a diversos perigos, torna imprescindível a utilização de técnicas que tem como objetivo evitar que acidentes aconteçam. Todavia, tem-se uma relação proporcional entre a gravidade das perdas ou prejuízos resultantes da disfunção de um sistema crítico e os recursos investidos, incluindo os esforços necessários para a aplicação da técnica, na prevenção de acidentes (ALMEIDA JR, 2003).

Em razão disso, uma das principais propriedades, que deve ser garantida por esses sistemas, é a segurança (*safety*). Uma vez que essa propriedade garante que, caso o sistema apresente uma disfunção, o funcionamento atípico do sistema não resultará em prejuízos para à vida humana ou para o ambiente no qual está inserido (VISMARI, 2007). Devido à ascensão do desenvolvimento de *Sistemas Críticos*, conseqüentemente aumenta-se a possibilidade de ocasionar problemas cada vez mais graves, torna-se fundamental o cumprimento de um conjunto de atividades destinadas a prevenção de acidentes e/ou a diminuição das conseqüências causadas pela ocorrência de uma disfunção desses sistemas (ALMEIDA JR, 2003).

Com esse propósito, o processo de desenvolvimento de sistemas críticos em segurança é, via de regra, controlado por órgãos reguladores. Esses órgãos determinam os critérios de certificação, específico de cada área, dos sistemas. Logo, existe uma relação entre os critérios, estabelecidos pelos órgãos, relacionados à garantia da segurança, também chamada de *Segurança Crítica*, e o *Nível de Integridade de Segurança* – SIL (*Safety Integrity Level*) do sistema (ALMEIDA JR, 2003).

2.2.1. Sistemas Tolerantes a falhas

Em sistemas críticos, devido à severidade das conseqüências que podem ser ocasionadas por uma disfunção, faz-se necessário evitar que o sistema ou algum componente do sistema opere de forma incorreta na presença de falhas (J-C et al., 2004). Neste contexto, a tolerância a falhas é uma qualidade ou atributo do sistema que permite alcançar a operação tolerante a

falhas, ou seja, o sistema ou o componente do sistema continua a operar normalmente mesmo na presença de uma disfunção no *hardware* ou erros de *software* (JOHNSON, 1989).

A tolerância a falhas é uma forma de reduzir a probabilidade de ocorrência de situações consideradas perigosas, cenários indesejados, que podem ocasionar acidentes devido a disfunção de sistemas críticos (SILVA NETO, 2014). Desta forma, um sistema tolerante a falhas possui a habilidade de continuar operando normalmente, ou seja, executando corretamente suas funções previamente definidas, mesmo nos casos em que os componentes de *hardware* funcionam de modo anormal (disfunção) ou erros de *software* ocorreram. Em outras palavras, um erro (“bug”) no *software* ou o funcionamento incomum do *hardware* de um sistema tolerante a falha não afeta a habilidade do sistema de funcionar corretamente (JOHNSON, 1989).

No entanto, para que um sistema consiga alcançar o estado de operação tolerante a falhas, é fundamental que esse sistema possua diversas propriedades (requisitos não funcionais), tais como, confiabilidade, disponibilidade, segurança, funcionabilidade, manutenibilidade, testabilidade e dependabilidade. Baseando-se nesses requisitos não funcionais do sistema, pode-se determinar o grau de tolerância a falhas de um sistema crítico (JOHNSON, 1989). A seguir explica-se cada um desses requisitos.

- a) **Confiabilidade (*Reliability*):** A confiabilidade é um atributo do sistema que garante, por um determinado período de tempo, a operação normal desse sistema. Ou seja, assegura que o sistema exercerá suas funções sem erros por um dado período de tempo (BOZZANO; VILLAFIORITA, 2010). De modo geral, utiliza-se a confiabilidade tanto com a finalidade de se evidenciar o grau de tolerância a falhas de sistemas críticos quanto para estimar a probabilidade de um sistema continuar a desempenhar corretamente suas funções, por um período de funcionamento, sem a realização de qualquer forma de manutenção (SILVA NETO, 2014).
- b) **Disponibilidade (*Availability*):** A disponibilidade avalia a probabilidade de um sistema desempenhar corretamente suas funções em um determinado instante de tempo (BOZZANO; VILLAFIORITA, 2010). Por esse motivo, representa o segmento de tempo, no tocante ao tempo total de operação do sistema, no qual o sistema continua completamente operacional (SILVA NETO, 2014).

Vale salientar que, apesar da aparente similaridade, existe uma diferença entre a confiabilidade e disponibilidade. A confiabilidade especifica a probabilidade de funcionamento correto do sistema durante um intervalo de tempo, enquanto que a disponibilidade se preocupa apenas com um instante de tempo, em outras palavras, a disponibilidade é medida em um determinado momento e reflete a probabilidade do sistema estar operando corretamente e disponível para exercer suas funcionalidades nesse período (JOHNSON, 1989).

Além disso, a disponibilidade é afetada tanto pela frequência de indisponibilidade do sistema, quanto pelo quão rápido, no tocante ao tempo, o processo de manutenção do sistema (reparo) pode ser realizado. Desta forma, mesmo sistemas que estão frequentemente indisponíveis, consequentemente poucos confiáveis (baixa confiabilidade), podem possuir alta disponibilidade desde que o tempo gasto com o processo de manutenção do sistema seja desprezível (JOHNSON, 1989).

- c) **Segurança (*Safety*):** A segurança de um sistema crítico, também chamada de segurança crítica, é uma propriedade fundamental desse tipo de sistema, principalmente de um sistema crítico em segurança, pois assegura que, no caso de funcionamento anormal do sistema, nenhum dano à vida humana e nem ao ambiente são provocados pelo sistema. Devido às graves consequências que esse sistema pode provocar em caso de falha, a *segurança é uma propriedade que deve ser garantida* por um sistema crítico em segurança (VISMARI, 2007). Portanto, a segurança de um sistema crítico, refere-se à característica do sistema de não colocar em risco, na ocorrência de uma disfunção, vidas humanas ou o próprio ambiente no qual está inserido (BOZZANO; VILLAFIORITA, 2010).

Considerando a ocorrência de uma disfunção, um sistema crítico em segurança pode alcançar tanto um **estado seguro** (*safe state*), no qual a disfunção apresentada pelo sistema o leva para um estado onde os requisitos de segurança ainda são satisfeitos, quanto um **estado inseguro ou perigoso** (*unsafe state*), no qual a disfunção apresentada pelo sistema o leva para um estado em que o sistema não satisfaz os requisitos de segurança. Nesse caso, quando o sistema se encontra em um estado inseguro, geralmente acontecem acidentes, acontecimentos que ocasionam mortes, ferimentos e perdas tanto materiais

quantos ambientais que sucederam-se de forma não intencional (VISMARI, 2007). Em função disso, quando um sistema, após a ocorrência de uma disfunção, alcança um estado seguro, diz-se que o sistema falhou de forma segura. Por outro lado, quando o sistema alcança um estado inseguro, diz-se que o sistema falhou de forma insegura.

- d) **Funcionabilidade (*Performability*):** A funcionabilidade de um sistema, especificada por $P(L, t)$, é uma função do tempo definida como a probabilidade do sistema desempenhar suas funcionalidades, no instante de tempo t , com um nível de desempenho L (JOHNSON, 1989). Esse nível de desempenho L , também chamado de nível de integridade, representa a capacidade do sistema de permanecer funcionando corretamente com uma redução de desempenho (operação degradada), tendo em vista que algumas funcionalidades do sistema estão comprometidas (SILVA NETO, 2014).
- e) **Mantenabilidade (*Maintainability*):** Manutenção é a ação realizada com o objetivo de manter ou restaurar um determinado sistema às condições de operação projetadas (STOREY, 1996). Em razão disso, dada a ocorrência de uma disfunção em um sistema, a manutenibilidade é uma medida relacionada a capacidade desse sistema ser reparado (JOHNSON, 1989). Em outras palavras, entende-se como manutenibilidade a habilidade de um sistema ser preservado (STOREY, 1996).

O processo de manutenção do sistema inclui desde a localização do problema, conseqüentemente, aplicando-se a etapa de reparação física do problema, até a restauração do mesmo para um estado operacional. Portanto, a manutenibilidade é uma característica essencial em qualquer sistema, mas torna-se ainda mais importante nos casos em que à vida humana, equipamentos ou o ambiente no qual o sistema está inserido podem estar expostos a riscos durante o processo de reparo do sistema (JOHNSON, 1989).

Em casos com maior nível de criticidade, tal como em sistemas ferroviários, pode-se utilizar técnicas, fundamentais para a obtenção da operação tolerante a falha em um sistema, para detectar e localizar vulnerabilidades nesse sistema visando a manutenção (JOHNSON, 1989). Diante disso, técnicas de diagnóstico automático de falhas podem ser utilizadas como uma forma de aprimorar os

índices de manutenibilidade de um sistema, pois o tempo total gasto na etapa de localização da origem de uma falha, etapa mais custosa no tocante ao tempo na execução de uma manutenção manual, é minimizado (JOHNSON, 1989).

- f) **Testabilidade (*Testability*):** A etapa de teste é uma etapa fundamental do processo de desenvolvimento de qualquer sistema crítico, pois essa etapa tem como objetivo descobrir erros no sistema (SOMMERVILLE, 2011). Neste contexto, um teste é uma forma de se analisar ou verificar a existência e a qualidade dos atributos do sistema. Logo, a testabilidade de um sistema refere-se à habilidade de avaliar os atributos de um sistema por meio de testes (JOHNSON, 1989).
- g) **Dependabilidade (*Dependability*):** A Dependabilidade “é uma propriedade de um sistema que justifica confiar nele” (STOREY, 1996). Sabendo disso, a dependabilidade define propriedades importantes de um sistema, tais como todos os requisitos descritos anteriormente (confiabilidade, disponibilidade, segurança, funcionabilidade, manutenibilidade, testabilidade). Devido a isso, é comumente avaliada qualitativamente por meio desses requisitos (JOHNSON, 1989; J-C et al., 2004; BOZZANO; VILLAFIORITA, 2010).

2.2.2. Relação entre Falhas, Erros e Disfunções em Sistemas Críticos

Na subseção anterior apresentou-se o conceito de sistemas tolerantes a falhas. Esta subseção apresenta a relação entre falhas, erros e disfunções (três conceitos essenciais no projeto de tolerância a falha) em sistemas críticos que, via de regra, são tolerantes a falhas.

No desenvolvimento de um sistema, geralmente, especifica-se a forma e o tipo dos dados que serão fornecidos como entrada para o sistema que está sendo desenvolvido. Com este propósito, são estabelecidas as condições normais necessárias para que o sistema exerça corretamente suas funcionalidades conforme o projetado. Logo, em qualquer sistema e independentemente de sua aplicação, é possível que um ou mais componentes do sistema falhe, ou seja, todo sistema está exposto à ocorrência de falhas em seus componentes.

Ademais, existe a possibilidade de que, em algum momento, durante o funcionamento do sistema os dados de entrada adquiridos sejam inadequados – isto é, dados provavelmente incompletos e, assim, diferentes das exigências especificadas no projeto do sistema. Neste caso,

em Sistemas Críticos em Segurança, faz-se necessário que essas situações sejam tratadas de modo que o sistema entre em um estado considerado seguro, no qual não exista a possibilidade de sua exposição a um cenário potencialmente perigoso, tanto para os indivíduos que interagem com o sistema, quanto para o ambiente no qual o sistema está inserido.

Uma forma de se averiguar o funcionamento normal ou incomum de um sistema é por meio de uma análise dos níveis de desempenho, em que os níveis atuais e mínimos (previamente especificados) do sistema são comparados. Essa análise viabiliza a monitoração da degradação, em relação ao desempenho, de um sistema ao longo do tempo. Normalmente, essa degradação relaciona-se com a existência de *defeitos* no sistema, os quais inviabilizam o desempenho de suas funcionalidades conforme os requisitos estabelecidos. Consequentemente, o acontecimento de uma disfunção em um sistema está vinculada à degradação do nível de desempenho desse sistema a valores menores do que os níveis mínimos, considerados aceitáveis, definidos previamente (VISMARI, 2007).

O conceito de *defeito* pode ser visto como uma relação de causa-consequência em três níveis distintos: *falha*, *erro* e *disfunção*. A distinção entre esses níveis pode ser feita, baseando-se em um modelo de três universos (*universo físico*, *universo da informação* e *universo externo ou do usuário*), conforme o universo ao qual o defeito pertence (JOHNSON, 1989; VISMARI, 2007). A *Figura 8* apresenta uma visão hierárquica da relação (causa-efeito) entre esses conceitos, sendo que no topo dessa hierarquia está a disfunção (mau funcionamento do sistema que afeta os seus usuários) e em baixo tem-se a falha (problema em algum componente do sistema) (JOHNSON, 1989). Ainda, conforme (JOHNSON, 1989), define-se:

- *Falha (Fault)*: defeito físico ou imperfeição originado por alguma debilidade em pelo menos um componente – de *hardware* ou de *software* – do sistema (defeito no *universo físico*). Geralmente, a vulnerabilidade de um componente de *software* origina-se em erros de especificação e implementação (VISMARI, 2007), enquanto que a vulnerabilidade de um componente de *hardware* origina-se por componentes defeituosos e distúrbios/fatores externos (JOHNSON, 1989);
- *Erro (Error)*: manifestação de uma *falha* física. Um erro representa a degradação do desempenho do sistema. “Especificamente, um *erro* é o desvio da acurácia” de alguma funcionalidade do sistema (defeito no *universo da informação* do

sistema) (VISMARI, 2007). Isto é, o sistema continua a desempenhar a funcionalidade, porém, abaixo do nível padrão de desempenho;

- *Disfunção (Failure)*: manifestação de um *erro* que provoca, no sistema, a execução incorreta de alguma funcionalidade, ou seja, o nível de desempenho do sistema é inferior aos níveis exigidos (defeito no *universo externo* ao sistema - usuários) (VISMARI, 2007). Devido a isso, o sistema apresenta um funcionamento anormal aos seus usuários. De modo geral, o funcionamento do sistema é interrompido, ou seja, o sistema deixa de operar.

Figura 8. Relação (causa-efeito) entre falha, erro e disfunção (**Fonte:** adaptado de (JOHNSON, 1989; ALMEIDA JR, 2003)).

Como evidencia a *Figura 8*, as falhas originam-se no universo físico do sistema, e devido a isso os defeitos gerados nesse universo se propagam pelos demais universos do sistema. O processo de propagação de um defeito (veja a *Figura 9*), que resulta na manifestação de uma disfunção na saída do sistema, dá-se como segue: a ocorrência de uma *falha* em um determinado componente, causada por problemas mecânicos ou elétricos no equipamento (ALMEIDA JR, 2003), provoca uma alteração no comportamento do módulo funcional do sistema que contém o componente falho. Denomina-se essa alteração comportamental local de *erro*, pois retrata a manifestação da falha do componente sobre a operação do sistema. Desta forma, e de acordo com a expressividade dessa alteração comportamental local, podem acontecer duas situações: (1) O efeito (defeito) provocado pela mudança de comportamento local é insuficiente (expressividade baixa) para impactar na operação global do sistema, logo as saídas geradas pelo sistema não sofrem nenhuma alteração, e; (2) O efeito provocado pelo *erro* é expressivo o suficiente para impactar na operação global do sistema, assim, comprometem as funcionalidades do sistema de modo que as saídas globais do sistema podem ser alteradas. Consequentemente, provocam uma alteração no nível de desempenho do sistema,

levando-o a valores inferiores aos níveis mínimos estabelecidos. Neste caso, diz-se que uma *disfunção*, provocada pela manifestação do *erro* (propagação do defeito até o universo externo), manifestou-se (SILVA NETO, 2014).

A *Figura 9* ilustra o processo de propagação dos defeitos, incluindo, sua relação com os estados de um sistema crítico – seguro e inseguro – especificados na subseção anterior. Observe que um estado inseguro (ou perigoso) do sistema, representado na figura pela *Zona insegura*, possui todas as premissas necessárias para a ocorrência de um acidente. Desta forma, dado que um acidente infringe o requisito de “segurança” do sistema, sistemas críticos em segurança possuem como principal objetivo, no tocante a segurança crítica, minimizar a níveis aceitáveis as situações que contribuem para a ocorrência de acidentes (VISMARI, 2007).

Figura 9. Propagação dos defeitos e sua relação com os estados do sistema (**Fonte:** adaptado de (VISMARI, 2007)).

2.3 VISÃO GERAL SOBRE MANUTENÇÃO

Atualmente, independentemente do setor, a competitividade imposta pelo mercado e as exigências dos consumidores fazem com que as empresas, que desejam manter-se ou serem relevantes no mercado, busquem por alternativas de melhorar o seu processo produtivo, tanto com relação à eficiência na produção, quanto na realização de reparos dos equipamentos. Neste contexto, segundo Xenos (1998), “qualquer degradação das condições físicas do equipamento poderá afetar também seu desempenho, sua produtividade e a qualidade dos produtos e serviços da empresa”. Por exemplo, considerando o caso de crescimento da demanda pelos produtos de uma determinada empresa – uma situação crucial para o aumento do número de vendas de seus produtos, e consequentemente, o aumento do lucro geral obtido através das vendas – torna-se inadmissível que seus equipamentos parem de produzir, devido à ocorrência de uma falha, ou produzam com desempenho abaixo dos níveis estabelecidos no projeto desses equipamentos.

Em vista disso, a manutenção surge como uma forma de evitar a degradação dos equipamentos, provocada tanto pelo desgaste natural ao longo do tempo quanto pelo uso. Existem diversas formas de se identificar a degradação de um dado equipamento, tais como aparência externa ruim, desempenho abaixo do normal, tempo de interrupção da produção, fabricação de produtos defeituosos e poluição ambiental. Desta forma, a manutenção dos equipamentos pode exercer uma função importante na melhoria da produtividade da empresa (XENOS, 1998).

A manutenção, segundo Xenos (1998), é definida como “a combinação de ações técnicas e administrativas, incluindo as de supervisão, destinadas a *manter* ou recolocar um item em um estado no qual possa desempenhar uma função requerida”. Sendo que se entende por *manter* o processo de garantir, independentemente das ações necessárias, que um equipamento permaneça a exercer suas funções, conforme o projetado, em um nível de desempenho considerado aceitável.

No entanto, a manutenção, dependendo do contexto de aplicação, não deve apenas visar a preservação do equipamento. Tem-se, ainda, que em alguns casos somente manter o equipamento operacional é insuficiente para as empresas que dependem diretamente do equipamento durante a produção dos produtos e serviços oferecidos pela empresa, caso no qual a falha de um equipamento é vista como um desastre e acarreta prejuízos. Todavia, o tratamento de falhas pode ser um excelente negócio para as empresas que trabalham apenas com a assistência técnica, onde se fornece peças de reposição e mão-de-obra à outras empresas (XENOS, 1998).

Kardec e Nascif (2009) afirmam que a manutenção, além de reestabelecer as condições originais de um equipamento, deve “garantir a confiabilidade e disponibilidade da função dos equipamentos e instalações de modo a atender a um processo de produção ou de serviço, com segurança, preservação do meio ambiente e custo adequados. ”

Desta forma, conforme Xenos (1998), pode-se classificar as atividades de manutenção, realizada em qualquer empresa, em dois tipos:

- *Atividades de Manutenção*: Objetivam, por meio do reparo das deteriorações nas condições originais, preservar o estado original de operação e o desempenho de um equipamento. Assim, esse tipo de atividade restringe-se apenas ao

reestabelecimento das condições originais de desempenho e confiabilidade intrínseca do equipamento. Portanto, visam preservar as propriedades e capacidade dos equipamentos ao longo do tempo (conceito mais comum de manutenção) (XENOS, 1998);

- *Atividades de Melhoria*: Objetivam, por meio da incorporação de modificações das condições originais, melhorar o estado original de operação, desempenho e confiabilidade intrínseca de um equipamento. Além disso, visam aumentar a produtividade por meio da redução dos custos de manutenção (XENOS, 1998).

Portanto, de modo geral, pode-se dizer que um dos objetivos da manutenção, devido ao fato de restringir-se ao processo de manter ou restaurar as condições físicas de um equipamento, é manter as capacidades funcionais dos equipamentos. Neste caso, além de preservar a essência do equipamento, preserva-se a capacidade com a qual o equipamento opera. Assim, ao realizar a manutenção da condição física do equipamento, indiretamente, realiza-se a manutenção da qualidade e segurança do produto (XENOS, 1998).

2.3.1 Evolução da Manutenção

A manutenção, nos últimos anos, sofreu diversas alterações, principalmente, decorrentes do aumento frenético do número e da diversidade de itens físicos – equipamentos, instalações e edificações – que devem ser mantidos, da complexidade dos novos projetos, do surgimento de novas técnicas de manutenção, assim como dos novos enfoques sobre a organização da manutenção e suas responsabilidades, e, por fim, da conscientização – por partes das pessoas da manutenção – sobre a importância da manutenção como uma função estratégica que visa a melhoria dos resultados do negócio e a competitividade das organizações (KARDEC; NASCIF, 2009).

Segundo Kardec e Nascif (2009), a evolução da manutenção, a partir de 1930, pode ser dividida em quatro gerações. Na *Tabela 3* apresenta-se uma visão geral da evolução da manutenção industrial ao longo dos anos, desde 1930 até os dias atuais. As colunas da tabela representam as quatro gerações descritas anteriormente. Da mesma forma, cada linha da tabela descreve, respectivamente, aumento das expectativas em relação à manutenção, a visão quanto à falha dos equipamentos e, por fim, as mudanças nos métodos de manutenção (KARDEC; NASCIF, 2009). Vale ressaltar, como é possível observar na tabela, que o processo de manutenção industrial iniciou considerando apenas que os equipamentos, em algum momento,

iriam falhar e, assim, seria realizada a manutenção corretiva não planejada somente para restaurar o estado operacional do equipamento. Atualmente, tem-se diversas formas de se realizar a manutenção dos equipamentos e, inclusive, é possível monitorar as suas condições de uso e conseguir efetuar a manutenção desse equipamento antes que uma disfunção efetivamente aconteça.

Tabela 3. Evolução da manutenção (**Fonte:** adaptado de (KARDEC; NASCIF, 2009)).

Evolução da Manutenção				
	Primeira Geração	Segunda Geração	Terceira Geração	Quarta Geração
Aumento das expectativas em relação à manutenção	<ul style="list-style-type: none"> ▪ Conserto após a falha 	<ul style="list-style-type: none"> ▪ Disponibilidade crescente ▪ Maior vida útil do equipamento 	<ul style="list-style-type: none"> ▪ Maior confiabilidade ▪ Maior disponibilidade ▪ Melhor relação custo-benefício ▪ Preservação do meio ambiente 	<ul style="list-style-type: none"> ▪ Maior confiabilidade ▪ Maior disponibilidade ▪ Preservação do meio ambiente ▪ Segurança ▪ Influir nos resultados do negócio ▪ Gerenciar os ativos
Visão quanto à falha do equipamento	<ul style="list-style-type: none"> ▪ Todos os equipamentos se desgastam com a idade e, por isso, falham 	<ul style="list-style-type: none"> ▪ Todos os equipamentos eletrônicos se comportam de acordo com a curva da banheira* 	<ul style="list-style-type: none"> ▪ Existência de padrões de falhas 	<ul style="list-style-type: none"> ▪ Reduzir drasticamente falhas prematuras.
Mudança nas técnicas de manutenção	<ul style="list-style-type: none"> ▪ Habilidades voltadas para o reparo 	<ul style="list-style-type: none"> ▪ Planejamento manual da manutenção ▪ Computadores grandes e lentos ▪ Manutenção preventiva (por tempo) 	<ul style="list-style-type: none"> ▪ Monitoramento da condição ▪ Manutenção preditiva ▪ Análise de risco ▪ Computadores pequenos e rápidos ▪ <i>Softwares</i> potentes ▪ Grupos de trabalhos multidisciplinares ▪ Projetos voltados para a confiabilidade ▪ Contratação por mão de obra e serviços 	<ul style="list-style-type: none"> ▪ Aumento da manutenção preditiva e monitoramento da condição ▪ Minimização nas manutenções preventiva e corretiva não planejadas ▪ Análise de falhas ▪ Técnicas de confiabilidade ▪ Manutenibilidade ▪ Engenharia de manutenção ▪ Projetos voltados para confiabilidade e custo do ciclo de vida ▪ Contratação por resultados

* Descrito a seguir.

2.3.2 Tipos de Manutenção

A subseção anterior apresentou tanto o conceito de manutenção quanto a sua evolução ao longo dos anos. Contudo, o conceito de *Gestão da Manutenção* não foi explicado. Por isso, antes de apresentar os diversos tipos de manutenção, faz-se necessário definir e compreender esse termo. Com este propósito, nesta subseção, apresenta-se o conceito de gestão da

manutenção. Posteriormente, descrevem-se os principais tipos de manutenção que podem ser aplicados no processo de manutenção industrial, visando a especificar a diferença entre cada forma de manutenção.

Entende-se como *gestão* o processo de liderar e administrar toda ou apenas uma parte de uma organização, geralmente de negócios, por meio da introdução e do controle dos recursos dessa organização (MÁRQUEZ, 2007). Dessa forma, pode-se definir a *Gestão da Manutenção* como todas as atividades de gestão, realizadas pelo departamento de gerenciamento e manutenção, que especificam os objetivos e/ou prioridades de manutenção (metas), estratégias (método de gerenciamento para atingir os objetivos de manutenção) e responsabilidades. Essas atividades são implementadas por meio do planejamento de manutenção (no qual determina-se o tipo de manutenção que será aplicado) e do controle e supervisão de manutenção. Além disso, entre essas atividades, tem-se a proposta de métodos que visam a melhoria, incluindo aspectos econômicos, do processo de manutenção (MÁRQUEZ, 2007).

Com relação aos tipos de manutenção, na literatura, é possível observar que não existe um consenso sobre uma categorização, comumente aceita, que englobe todas as formas de manutenção. Por exemplo, (SUSTO et al., 2015) definem que os tipos de manutenção podem ser categorizados em ordem crescente, baseando-se em sua complexidade e eficiência, em três categorias, a saber: Executar até a falha (do inglês, *Run-to-Failure*), Preventiva e Preditiva. Ainda, (MOBLEY, 2002) diz que, normalmente, as indústrias empregam apenas as duas primeiras abordagens. Neste caso, considera que a manutenção preditiva, por diversos autores também chamada de *manutenção baseada em condição*, é uma variação da manutenção preventiva e, assim, não a conceitua como uma outra categoria de manutenção.

Além disso, observou-se que o tipo de manutenção aplicado sobre os equipamentos de uma indústria, comumente, reflete a forma de gerenciamento dos equipamentos dessa indústria. Consequentemente, a ampla variedade de denominações para o tipo de manutenção aplicado na indústria, em alguns casos, dificulta o entendimento sobre a forma de gestão da manutenção utilizada (KARDEC; NASCIF, 2009).

Antes de descrever os principais tipos de manutenção e suas características, faz-se necessário esclarecer os termos *ação de manutenção* e *políticas de manutenção*. Conforme exposto na norma NBR-5462 ABNT (1994), compreende-se como uma ação de manutenção a “sequência de atividades elementares de manutenção” realizadas com um determinado objetivo.

As atividades, mais comuns, de manutenção em um equipamento são inspeção, monitoramento, manutenção de rotina, revisão, reconstrução e reparo (MÁRQUEZ, 2007). Enquanto que o termo *política de manutenção* representa as “inter-relações entre os escalões de manutenção (posição dentro da organização onde os níveis de manutenção determinados são realizados sobre um item), os níveis de intervenção (fragmentação de um item sobre o qual são efetuadas as ações de manutenção) e os níveis de manutenção (conjunto de ações de manutenção a serem realizadas em um nível de intervenção especificado) a serem aplicados para a manutenção de um item”. Sobretudo, conforme (MÁRQUEZ, 2007), os tipos de manutenção resumem-se a um conjunto de atividades de manutenção executadas em uma determinada ordem.

A *Figura 10* apresenta os diversos tipos de manutenção. Observe que, de modo geral, os tipos de manutenção podem ser divididos, baseando-se no tipo de ação de manutenção utilizado, em preventivos (objetivam preservar o equipamento em um estado operacional) ou corretivos (visam o reestabelecimento das condições originais/operacionais do equipamento). A seguir, descreve-se cada uma das formas de manutenção apresentadas nesta figura.

Figura 10. Tipos de Manutenção (Fonte: Adaptado de (MÁRQUEZ, 2007))

2.3.2.1 Manutenção Preventiva

A manutenção preventiva pode ser definida como um tipo de manutenção que é aplicada baseando-se em períodos de tempo ou em critérios, ambos previamente estabelecidos. Desta forma, objetiva diminuir tanto a probabilidade de falha quanto o nível de degradação, em relação ao desempenho, de um equipamento (NBR-5462 ABNT, 1994). Segundo Márquez (2007), a manutenção preventiva pode ser periódica, comumente chamada de manutenção baseada no tempo (do inglês, *Time-Based Maintenance – TBM*), ou baseada em condição (do inglês, *Condition-Based Maintenance – CBM*). A seguir, descreve-se brevemente cada uma delas.

2.3.2.1.1 Manutenção preventiva baseada no tempo

A forma tradicional de se realizar a manutenção preventiva em um equipamento é por meio da manutenção baseada no tempo, comumente conhecida como manutenção periódica (AHMAD; KAMARUDDIN, 2012). A TBM utiliza-se do tempo para efetuar a manutenção, geralmente lubrificação, ajustes ou pequenos reparos, de um equipamento (MOBLEY, 2002). Assim, as decisões de manutenção, tais como a definição – em meses ou em horas de operação – dos intervalos de tempo para a realização da manutenção em um equipamento, são estabelecidas de acordo com a análise do tempo de falha (do inglês, *failure time*) desse equipamento. Ou seja, baseando-se em dados de tempo de falha, estima-se o tempo de vida útil do equipamento (AHMAD; KAMARUDDIN, 2012).

A TBM, devido à relação com a vida útil do equipamento, pressupõe que, dentro de um determinado período de tempo, ocorrerá o processo de degradação desse equipamento (MOBLEY, 2002). Consequentemente, assume que é possível predizer o comportamento do equipamento durante o seu ciclo de vida, incluindo o tempo até a falha desse equipamento, baseando-se na quantidade de falhas – do equipamento – nesse intervalo de tempo (AHMAD; KAMARUDDIN, 2012). Neste contexto, supõe-se que a previsão de falhas, baseando-se na manifestação das falhas no decorrer da vida útil do equipamento, é um processo viável. Assim, sugere-se um intervalo fixo de tempo para a realização da manutenção, de modo constante (por exemplo, a cada 3 meses), durante o ciclo de vida do equipamento. Essa suposição baseia-se na função de taxa de falha. Segundo (JOHNSON, 1989), entende-se como taxa de falha (*failure rate*) o número de falhas de um equipamento ou sistema que podem acontecer em um dado intervalo de tempo.

No caso de componentes eletrônicos, ou sistemas elétricos, eletrônicos e programáveis (SILVA NETO, 2014), a curva da banheira, ilustrada na *Figura 11*, é comumente aceita como a relação entre a função de taxa de falhas ao longo do tempo (JOHNSON, 1989). Como é possível observar na figura, a curva da banheira divide-se em três fases, *Período de Mortalidade Infantil*, *Período de Vida Útil* e *Período de Desgaste*. Basicamente, a técnica TBM baseia-se na ideia da curva da banheira, e, assim, assume que no início do ciclo de vida dos equipamentos a taxa de falhas é decrescente (Mortalidade infantil), em seguida, passam por um longo período de vida útil, onde a taxa de falha torna-se constante e possui valor igual a λ (*lambda*), e, por

fim, no final do ciclo de vida dos equipamentos, a frequência das falhas aumenta, ou seja, a taxa de falhas é crescente (Desgaste) (AHMAD; KAMARUDDIN, 2012).

Figura 11. Curva característica da vida útil de equipamentos, comumente conhecida como curva da Banheira (**Fonte:** Adaptado de (JOHNSON, 1989; SILVA NETO, 2014)).

A curva da banheira considera que no início da vida útil de um sistema, frequentemente, ocorrem disfunções. Essas disfunções são ocasionadas normalmente por componentes cujo desempenho está abaixo do normal (padrão) ou devido a alguma vulnerabilidade de um componente. Em vista disso, no final da vida útil do sistema, dado que ele esteve funcionando por um longo período de tempo, tem-se um aumento no número de falhas apresentadas pelo sistema, ocasionadas pelo desgaste físico dos componentes eletrônicos ou mecânicos. Por isso, é durante a vida útil do sistema que ele fornece tanto o melhor serviço, ou seja, desempenha melhor as suas funções quanto o mais presumível, em relação a compreensão do seu funcionamento, aos seus usuários (JOHNSON, 1989).

Sabendo-se disso, após a obtenção de um conjunto de dados de tempo de falha, realiza-se uma análise por meio de uma modelagem estatística ou de confiabilidade, visando apontar um conjunto de atributos sobre o comportamento do equipamento, tais como o tempo médio até a falha (do inglês, *Mean-Time-To-Failure* – MTTF) e a tendência da taxa de falha do equipamento de acordo com a curva da banheira (AHMAD; KAMARUDDIN, 2012).

Desta forma, acredita-se que é possível antever a falha de um equipamento por intermédio do cálculo do MTTF (MOBLEY, 2002). Esse atributo representa o tempo previsto de operação de um determinado sistema até que a primeira falha aconteça, neste caso uma disfunção ocasionada por uma falha em pelo menos um componente desse sistema (JOHNSON, 1989). Vale ressaltar que, na TBM, não se verifica previamente a condição de uso do equipamento (MÁRQUEZ, 2007).

2.3.2.1.2 *Manutenção preventiva baseada em condição*

A manutenção baseada em condição, também conhecida pelos termos “*Manutenção Preditiva*”, “*monitoramento online*” e “*manutenção baseada em risco*” (HASHEMIAN; BEAN, 2011), é um tipo de manutenção contemporânea e amplamente estudado na literatura nos últimos anos (AHMAD; KAMARUDDIN, 2012). Vale ressaltar que, na literatura, pode-se visualizar a junção dos termos CBM e manutenção preditiva como um único tipo de manutenção (MOBLEY, 2002; HASHEMIAN; BEAN, 2011), assim como em formas similares de manutenção, mas diferentes, onde a manutenção preditiva é uma subdivisão da CBM (MÁRQUEZ, 2007).

Shin e Jun (2015) apresentam diversas definições para o termo manutenção baseada em condição, inclusive, baseando-se nessas definições, afirma que a CBM é um tipo de manutenção que realiza as ações de manutenção antes da ocorrência de uma falha, por meio do monitoramento das condições – incluindo o ambiente operacional – do equipamento, além de prever, em tempo real, o risco de falha desse equipamento com base nos dados coletados.

Márquez (2007) define a CBM como um tipo de manutenção preventiva que se baseia no desempenho e/ou no monitoramento de determinados parâmetros e nas ações decorrentes desse processo, sendo que o monitoramento, via de regra, é realizado de forma programada, contínua ou por meio de requisição. Neste contexto, entende-se por monitoramento o processo de analisar o estado (real) de operacionalidade de um equipamento (AHMAD; KAMARUDDIN, 2012).

Consequentemente, de um modo geral, pode-se compreender a CBM como uma forma de manutenção cujas decisões sobre a realização das ações de manutenção são efetuadas com base nas informações oriundas do monitoramento do equipamento (AHMAD; KAMARUDDIN, 2012). Neste contexto, a vida útil do equipamento é monitorada por meio do seu estado de operacionalidade, normalmente, representado por parâmetros de monitoramento, tais como a vibração, temperatura, corrente elétrica e níveis de óleo lubrificante (AHMAD; KAMARUDDIN, 2012).

Desta forma, diferentemente da manutenção corretiva e da manutenção preventiva periódica, a CBM objetiva tanto a detecção e diagnóstico de falhas em um equipamento (ou sistema) quanto o monitoramento da degradação desse equipamento, além da predição de falhas

(SHIN; JUN, 2015). Em vista disso, frequentemente, é utilizada como uma forma de diminuir a incerteza sobre as atividades de manutenção, pois é executada de acordo com a condição do equipamento, a qual é obtida por meio dos parâmetros analisados (SHIN; JUN, 2015).

Por isso, diz-se que ao utilizar a CBM é possível prolongar a vida útil de um equipamento o máximo possível, otimizando o processo de produção e aumentando a produtividade, sem a ocorrência de uma falha (SHIN; JUN, 2015). Neste caso, as ações de manutenção, tais como substituição, reparos ou ajustes, são efetuados somente quando necessários ou antes que a falha efetivamente ocorra, ambos de forma programada. Portanto, a CBM visa, principalmente, avaliar em tempo real o estado de operacionalidade do equipamento, com o propósito de executar decisões sobre as ações de manutenção, de modo a diminuir a quantidade de ações de manutenção desnecessárias e, conseqüentemente, os custos operacionais relacionados a essas atividades (AHMAD; KAMARUDDIN, 2012).

A degradação de um equipamento pode ser identificada por meio da variação dos parâmetros analisados, isto é, caso esses parâmetros alcancem valores anormais, em relação aos valores limites, previamente definidos, que representam um estado operacional, pode-se proceder com a manutenção do equipamento e antever a falha. Devido a isso, consegue-se “alertar”, com antecedência, a falha eminente do equipamento monitorado. Logo, aumenta-se a confiabilidade (o sistema conseguirá executar sem falhas por extensos períodos de tempo) e disponibilidade (o tempo de reparo, por ser programado, é menor e, conseqüentemente, o sistema estará, em um curto intervalo, novamente à disposição para executar suas funções) do sistema.

No entanto, em geral, os custos advindos do investimento em um programa de manutenção baseada em condição, devido aos recursos necessários para sua implantação, são altos. Visto que, para implementar um programa de CBM, faz-se necessário instalar sensores para monitorar o equipamento, além de definir uma estratégia de tomada de decisão para a manutenção do mesmo. Logo, tanto a aquisição do *hardware* apropriado quanto o treinamento dos funcionários responsáveis por gerenciar o programa geram altos custos e, em alguns casos, inviabilizam a implantação desse programa (SHIN; JUN, 2015).

No presente trabalho, dado as principais características da CBM (descritas nesta subseção), considera-se que a CBM engloba todas as características da manutenção preditiva e, assim, entende-se como um único tipo de manutenção. Logo, qualquer um dos termos,

previamente mencionados, referem-se à mesma forma de manutenção. Porém, vale ressaltar que a (ISO 13372, 2012) considera o termo manutenção preditiva obsoleto. Por fim, no restante desta dissertação, adotou-se o termo *manutenção baseada em condição* (sugerido pela (ISO 13372, 2012)).

2.3.2.2 Manutenção Corretiva

A manutenção corretiva é aplicada para corrigir uma falha ou restaurar o desempenho de um equipamento a níveis satisfatórios (KARDEC; NASCIF, 2009). Assim, objetiva recolocar o equipamento em um nível de desempenho no qual ele possa exercer, conforme o esperado, uma determinada função (NBR-5462 ABNT, 1994).

Neste contexto, geralmente, existem duas situações em que se faz necessário efetuar especificamente a manutenção corretiva em um equipamento. São elas: (1) após uma falha ocorrer, onde as ações de manutenção são realizadas se, e somente se, a máquina apresentar uma falha (disfunção), ou seja, de modo emergencial e; (2) quando se deseja restaurar um equipamento defeituoso, ou que esteja apresentando um desempenho insatisfatório, a um estado de operação aceitável (KARDEC; NASCIF, 2009).

Desta forma, a manutenção corretiva pode ser dividida em **Manutenção Corretiva Emergencial** (também conhecida como Não-Planejada) ou **Manutenção Corretiva Planejada** (KARDEC; NASCIF, 2009). No primeiro caso, realiza-se a manutenção apenas após a ocorrência da falha e não se realiza previamente o planejamento de execução da ação de manutenção. Ou seja, efetua-se o reparo da falha de modo emergencial, sem preparação prévia para a execução do serviço, tais como equipe de manutenção apropriada e equipamentos necessários (KARDEC; NASCIF, 2009). Assim, devido às suas características, trata-se de uma forma de manutenção reativa, visto que se deixa o equipamento parar de funcionar para realizar a manutenção do mesmo (em alguns casos, utiliza-se o termo *executar até a falha*) (MOBLEY, 2002).

Por isso, diz-se que, baseando-se no conceito de manutenção apresentado anteriormente, nenhuma outra forma de manutenção é aplicada sobre o equipamento, isto é, apenas ações de manutenção corretiva são realizadas. Além disso, faz-se necessário reagir a todos os possíveis modos de falha do equipamento. Logo, dado essas características, é considerada um tipo de manutenção primitiva e cara de se realizar na manutenção de equipamentos (MOBLEY, 2002).

No caso da manutenção corretiva planejada, o reparo da falha é realizado de forma planejada e de acordo com uma decisão gerencial. Normalmente, essa decisão provém da análise de um conjunto de variáveis, obtidas por meio do monitoramento do equipamento, que representam o desempenho do equipamento em um determinado período. Assim, baseando-se na variação dessas variáveis, a manutenção do equipamento pode ser planejada, inclusive pode-se optar por deixar o equipamento executar até a falha, mas, mesmo nesse caso, trata-se de uma decisão gerencial, pois tem-se um planejamento para quando a falha ocorrer (KARDEC; NASCIF, 2009).

Finalmente, após se apresentar os diversos tipos de manutenção e suas principais características, vale ressaltar que as políticas de manutenção visam o desenvolvimento desse conceito. Assim, de acordo com (KARDEC; NASCIF, 2009), as políticas de manutenção possibilitam a aplicação dos diversos tipos de manutenção e, segundo (AHMAD; KAMARUDDIN, 2012), expressam “a maneira como uma empresa pensa sobre o papel da manutenção como uma função operacional”. Dentre as principais políticas de manutenção estão a Manutenção Centrada em Confiabilidade (em inglês, *Reliability-Centred Maintenance – RCM*) e a Manutenção Produtiva Total (em inglês, *Total-Productive Maintenance – TPM*) (AHMAD; KAMARUDDIN, 2012).

2.4 DETECÇÃO E DIAGNÓSTICO DE FALHAS

Desde o surgimento da indústria, época na qual o ser humano começou a construir equipamentos mecânicos e elétricos, existe uma preocupação com as condições de uso dos equipamentos. Neste período, utilizavam-se apenas os conhecimentos do operador, oriundos da experiência de trabalho, para realizar a supervisão da condição de uso do equipamento. Neste contexto, geralmente, as decisões sobre as ações de manutenção baseavam-se nos sentidos biológicos do supervisor. Assim, procurava-se compreender a origem e o tipo de falha por meio, por exemplo, da audição (supervisor ouve sons incomuns provindos do equipamento) e do tato (supervisor sente, ao tocar no equipamento, excesso de calor e/ou vibração) (GERTLER, 2017).

Recentemente, com a implantação dos sistemas de monitoramento das condições de uso dos equipamentos, permitindo a obtenção de dados sobre o estado de degradação do equipamento, tornou-se possível o desenvolvimento de algoritmos capazes de realizar a tarefa de supervisão dos equipamentos, antes feita manualmente, de modo automático e em tempo real (CHEN; ROBERTS, 2006). Neste contexto, ainda conforme (CHEN; ROBERTS, 2006),

os sistemas de detecção e diagnóstico de falhas (do inglês *Fault Detection and Diagnosis – FDD*) permitem a detecção de falhas antes da sua manifestação ocasionar uma disfunção no sistema, além de prover informações sobre a sua localização e criticidade por meio do diagnóstico dessas falhas.

Sabendo-se disso, uma tarefa de supervisão objetiva analisar o estado atual de um equipamento de modo a evitar que o mesmo alcance um nível de degradação, geralmente ocasionado por uma falha, indesejado e/ou não permitido. Com este propósito, baseando-se em conjunto de variáveis monitoradas, realizam-se as ações adequadas para impedir a ocorrência de acidentes (ISERMANN, 2006).

Geralmente, em um processo de monitoramento da condição de uso de um equipamento, tem-se a análise contínua de pelo menos uma variável. Essas variáveis, advindas, em alguns casos, de cálculos estatísticos ou da experiência (*expertise*) do especialista sobre o equipamento, representam o estado de operacionalidade e/ou comportamento do equipamento. Nesse contexto, por meio dos dados coletados em tempo real, que são transformados em medidas significativas, é possível auxiliar o especialista de domínio (que possui conhecimento especializado sobre o equipamento) na identificação do desgaste do equipamento/sistema monitorado e, em alguns casos, no diagnóstico das falhas (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000).

Além disso, é comum se definir um valor limite de desempenho, o qual não causa nenhum dano ao sistema, para uma ou mais variáveis. Normalmente, utiliza-se essa técnica para a detecção de falhas, dado que no momento em que a variável monitorada ultrapassar esse limite, considera-se que uma falha ocorreu. Devido a isso, assume-se que as variáveis usadas definem o comportamento do equipamento e, assim, podem ser utilizadas para identificar se o mesmo encontra-se dentro ou fora das condições de funcionamento estabelecidas (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000).

Sabendo-se dessa relação entre as variáveis observadas e o comportamento do sistema, é possível identificar, por intermédio da comparação dessas propriedades, qual a variável mais afetada pela ocorrência de uma falha. Neste caso, torna-se viável relacionar o acontecimento de uma falha a uma variável específica e, assim, aumentar o entendimento do operador sobre os tipos de falha do equipamento, além das suas características comportamentais em estado de falha (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000).

A *Figura 12*, apresenta a estrutura básica de um sistema FDD. Observe que, nesses sistemas, tem-se quatro processos consecutivos e interligados (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000): (i) a detecção de falhas, responsável por detectar que uma falha ocorreu no sistema; (ii) a identificação de falhas, encarregada de especificar o tamanho da falha; (iii) o diagnóstico de falhas, o qual fornece informações sobre a falha e, por fim; (iv) a restauração do equipamento e/ou sistema, responsável por restaurar o sistema a um estado operacional. Vale ressaltar, que em alguns casos, a identificação de falhas é omitida, pois ela é considerada como uma etapa do processo de diagnóstico da falha. Finalmente, um método de detecção e diagnóstico de falhas é um algoritmo que implementa as tarefas de detecção e de diagnóstico descritas anteriormente (GERTLER, 2015).

Figura 12. Estrutura de um sistema de Detecção e Diagnóstico de Falhas (**Fonte:** Adaptado de (CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, 2000)).

2.5 APRENDIZADO DE MÁQUINA: UMA ABORDAGEM SOBRE OS PRINCIPAIS CONCEITOS E TÉCNICAS

Esta seção apresenta os principais conceitos relacionados ao aprendizado de máquinas (do inglês, *Machine Learning* – *ML*). Primeiramente, descrevem-se alguns conceitos importantes em ML. Em seguida, apresentam-se algumas das diversas formas de aplicação deste conceito, tais como as tarefas de agrupamento de dados e classificação abordadas neste trabalho científico. Além disso, cada uma das técnicas utilizadas nesta dissertação é brevemente descrita.

Entende-se como aprendizado de máquina uma subárea da Inteligência Artificial, o processo de aprender informações significativas sobre um conjunto de dados com o mínimo ou nenhum suporte adicional (externo). Neste contexto, um algoritmo de ML, sequência de ações/instruções que visam realizar uma determinada tarefa, consegue aprender somente por meio da análise dos dados (ALPAYDIN, 2009).

Geralmente, o aprendizado de máquina pode ser utilizado para a resolução de problemas em diferentes domínios, tais como negócios, indústrias, medicina e telecomunicações

(ALPAYDIN, 2009). Independentemente do domínio de aplicação, um algoritmo de aprendizado de máquina é comumente classificado de acordo com a forma pela qual ele obtém informações do conjunto de dados (AMRUTHNATH; GUPTA, 2018).

Normalmente, as técnicas de aprendizado de máquina são classificadas em três categorias (classes). São elas: (1) *Aprendizado supervisionado*, no qual existe o conceito de um “*supervisor externo*”. Esse supervisor conhece a classe de cada instância do conjunto de dados. Assim, nesta categoria, sabe-se a classe (rótulo) de cada exemplo do conjunto de dados; (2) *Aprendizado não supervisionado*, no qual não existe o conceito do “supervisor externo”. Logo, não se sabe a classe das instâncias do conjunto de dados. Devido a isso, diz-se que as técnicas desta categoria não utilizam qualquer informação sobre as classes dos dados; e (3) *Aprendizado semi-supervisionado*, na qual sabe-se a classe de uma parte das instâncias do conjunto de dados. Desta forma, utiliza-se esse conhecimento para identificar a classe das demais instâncias (ALPAYDIN, 2009; SARAVANAN; SUJATHA, 2018; DAWOUD; SHAHRISTANI; RAUN, 2019).

No entanto, além dessas formas de aprendizado, tem-se também o *aprendizado por reforço*, no qual um agente deve aprender a se comportar, por meio de interações, com o ambiente no qual está inserido. Geralmente, o agente aprende por técnicas de “força bruta” (tentativa e erro). Neste contexto, para cada ação realizada pelo agente existe a possibilidade de ele ser recompensado (ação correta) ou punido (ação errada) (AMRUTHNATH; GUPTA, 2018).

Sabendo-se disso, nas próximas subseções, apresentam-se os conceitos de classificação e agrupamento de dados em aprendizado de máquina. Desta forma, descrevem-se as etapas necessárias para a resolução de um problema nessas subáreas de ML. Posteriormente, apresentam-se as redes neurais artificiais, as quais são amplamente utilizadas em problemas de ML.

2.5.1 Classificação

De modo geral, problemas do mundo real sofrem com a necessidade de se realizar uma distinção entre objetos. Esses objetos, por sua vez, podem ser tanto heterogêneos quanto homogêneos, e comumente são representados por suas características (propriedades). O termo objeto é amplo e, devido a isso, depende diretamente do contexto no qual está sendo utilizado,

ou seja, do seu domínio de aplicação. Desta forma, o termo objeto pode representar uma pessoa, um animal, uma planta ou um equipamento.

Em um cenário bancário, suponha que um agente de empréstimo necessita decidir se é viável (ou não) aceitar a solicitação de empréstimo realizada por um determinado cliente. Neste caso, precisa-se categorizar o cliente como “Seguro” (não traz riscos para o banco) ou “Inseguro” (pode trazer riscos para o banco). Para isso, faz-se necessário analisar o perfil desse cliente e, baseando-se em seus dados, categorizá-lo. Finalmente, pode-se concluir essa ação aceitando ou recusando a solicitação do cliente.

Sabendo-se disso, denomina-se *classificação* a tarefa de analisar dados conhecidos e atribuir um objeto (no exemplo anterior, o cliente seria um “objeto”) à pelo menos uma das diversas classes (categorias) predefinidas (TAN; STEINBACH; KUMAR, 2013). Assim, a classificação é uma forma de analisar os dados de modo a extrair um modelo (função) que, por meio da compreensão de informações sobre os dados, consegue distinguir essas classes predefinidas (HAN; KAMBER; PEI, 2011).

Dessa forma, a classificação é uma técnica amplamente utilizada em aprendizado supervisionado na qual deve-se conhecer a classe de cada instância do conjunto de dados analisado, para a resolução de diferentes problemas em diversos domínios de aplicação (HAN; KAMBER; PEI, 2011). Formalmente, a classificação pode ser definida como a tarefa de aprender uma função alvo (f), ou objetivo, que determina para uma nova instância de dados (x) um dos rótulos das classes predefinidas (y) (TAN; STEINBACH; KUMAR, 2013).

Informalmente, a função alvo é comumente chamada de modelo de classificação (TAN; STEINBACH; KUMAR, 2013). Normalmente, um modelo de classificação é utilizado tanto para distinguir um objeto específico em um conjunto de objetos, chamada de modelagem descritiva, quanto para prever a classe, baseando-se em classes predefinidas, de instâncias (exemplos) de dados desconhecidas (novas). Vale salientar que essas instâncias não são utilizadas durante o processo de aprendizado do modelo e, por conta disso, diz-se que são desconhecidas pelo modelo (TAN; STEINBACH; KUMAR, 2013).

Segundo Tan, Steinbach e Kumar (2013), uma técnica de classificação (ou classificador) é “uma abordagem sistemática para construir modelos de classificação a partir de um conjunto de dados de entrada”. Um classificador, obtido como resultado do treinamento de um modelo,

aprende uma generalização (regra preditiva geral) do conjunto de dados de entrada por meio de exemplos (RASCHKA, 2015).

Sabendo-se disso, cada uma dessas técnicas de classificação utiliza um *algoritmo de aprendizado*, tais como redes neurais artificiais, árvores de decisão e máquinas de vetores de suporte, para obter um modelo em que se ajusta a relação entre os atributos e o seu respectivo rótulo de classe no conjunto de dados analisado. Neste contexto, esse modelo deve se ajustar bem aos dados de entrada de modo a conseguir prever, com uma boa precisão, os rótulos de classe das instâncias desconhecidas (TAN; STEINBACH; KUMAR, 2013). Devido a isso, diz-se que o principal objetivo de um algoritmo de aprendizado é criar modelos com alta capacidade de generalização, isto é, com uma boa regra preditiva geral, para classificar os rótulos de classe de novos exemplos (TAN; STEINBACH; KUMAR, 2013).

Uma tarefa de classificação consiste em duas etapas, treinamento e teste. Em vista disso, é uma prática comum dividir o conjunto de dados de entrada em dois subconjuntos, um subconjunto de treinamento e outro de teste. Vale ressaltar que cada subconjunto é utilizado apenas em uma das etapas. Logo, utiliza-se o subconjunto de treinamento na etapa de treinamento do modelo de classificação e o subconjunto de teste na etapa de avaliação do desempenho do classificador (HAN; KAMBER; PEI, 2011).

Durante o processo de treinamento, baseando-se no subconjunto de dados de treinamento, o classificador identifica um conjunto de classes de acordo com o rótulo (classe) de cada instância de dados. Em virtude disso, é possível utilizar o classificador para classificar novas instâncias de dados em uma dessas classes. Por isso, diz-se que na etapa de treinamento o classificador aprende, baseando-se nos dados rotulados, como classificar novas instâncias de dados (CHANDOLA; BANERJEE; KUMAR, 2009).

Ainda com relação à etapa de treinamento, é possível avaliar o desempenho do classificador durante o processo de aprendizado. Para isso, utiliza-se a validação, a qual pode ser entendida como uma subetapa do processo de treinamento do classificador. Geralmente, se utiliza a validação para analisar a perda de aprendizado, refere-se à inconsistência entre o tipo de classe predito pelo classificador e o real de cada instância de dados, e as métricas definidas na implementação do modelo a cada iteração de treinamento, também chamada de “época” (KOHAVI, 1995).

Além disso, a validação é uma forma de identificar que o classificador está com sobreajuste (do inglês, *overfitting*), um problema comum em um classificador (NITISH et al., 2014). Basicamente, diz-se que um classificador apresenta sobreajuste quando ele obtém um bom desempenho no processo de treinamento (validação), mas as características aprendidas sobre o conjunto de dados referem-se a comportamentos individuais dos dados. Consequentemente, o classificador não obtém um bom desempenho na classificação de novas instâncias de dados (DIETTERICH, 1995). Segundo (NITISH et al., 2014), realizar cortes pontuais, por meio da inserção de camadas de “corte” (do inglês, *dropout*) é uma forma de se contornar esse problema.

O desempenho de um classificador na classificação de novas instâncias de dados é obtido na etapa de teste. Desta forma, por meio do subconjunto de teste, é possível analisar o desempenho de um classificador antes de aplicá-lo sobre dados cujo rótulo não é conhecido. Geralmente, avalia-se o desempenho de um classificador por meio de um conjunto de métricas, chamadas *métricas de desempenho* (TAN; STEINBACH; KUMAR, 2013).

Neste contexto, a etapa de teste de um classificador é responsável por avaliar o seu desempenho em um ambiente conhecido e controlado. Assim, baseando-se nas métricas de desempenho calculadas nessa etapa, é possível decidir sobre a utilização do classificador em um ambiente real, ou seja, cujos rótulos dos objetos (instâncias de dados) não são previamente conhecidos (HAN; KAMBER; PEI, 2011). Analisando essa etapa de uma tarefa de classificação, é possível visualizar o cálculo de desempenho em duas subetapas: a *classificação do subconjunto de teste* e o *cálculo da métricas de performance*.

Primeiramente, o subconjunto de teste é dado como entrada para o classificador. Consequentemente, o classificador define uma nova classe para cada instância de dados desse subconjunto. Posteriormente, baseando-se nessa classificação, calcula-se as métricas de desempenho, tais como *acurácia*, *precisão*, *sensibilidade (recall)* e *média-F1* (descritas a seguir) (LEE et al., 2018). Para isto, basicamente, computa-se a quantidade de instâncias de dados do subconjunto de teste preditas corretamente e incorretamente pelo classificador. Em geral, em um problema de classificação binária (apenas duas classes), tal como a classificação de anomalias (com duas classes – “anomalia” ou “normal”), organiza-se essas informações em uma matriz, chamada *matriz de confusão* (veja a *Tabela 4*) (TAN; STEINBACH; KUMAR, 2013).

Tabela 4. Matriz de confusão (C) de um problema de classificação de anomalias, onde C_{ij} representa o número de instâncias de dados da classe i previstas, pelo classificador, como sendo da classe j (**Fonte:** Adaptado de (TAN; STEINBACH; KUMAR, 2013)).

		Classe predita	
		Classe 0	Classe 1
Classe real	Classe 0	C_{00}	C_{01}
	Classe 1	C_{10}	C_{11}

Os elementos C_{ij} de uma matriz de confusão (Tabela 4), são comumente chamados de verdadeiro negativo – C_{00} (TN, do inglês *True Negative*), falso negativo – C_{10} (FN, do inglês *False Negative*), falso positivo – C_{01} (FP, do inglês *False Positive*) e verdadeiro positivo – C_{11} (TP, do inglês *True Positive*). No contexto da detecção de anomalias, TN representa o número de registros previstos corretamente pelo classificador como anomalias. Enquanto que FN representa o número de registros previstos incorretamente como anomalias. Consequentemente, TP representa o número de registros previstos corretamente como normais, assim como TN representa o número de registros previstos incorretamente como normais.

De acordo com Tan, Steinbach e Kumar (2013), uma matriz de confusão possui as informações necessárias para avaliar o desempenho de um classificador. Assim, por meio dos valores de TN, TP, FN e FP é possível calcular as métricas de desempenho (LEE et al., 2018). Em outras palavras, transforma-se as informações dispostas na matriz de confusão em números de modo a tornar mais conveniente a comparação de desempenho entre classificadores (TAN; STEINBACH; KUMAR, 2013).

Por fim, ainda no caso da detecção de anomalias, a *acurácia* refere-se à quantidade de instâncias corretamente classificadas, dentre todas as instâncias de dados. A *precisão* refere-se à eficiência do classificador na detecção de anomalias, assim, representa a fração de todas as instâncias corretamente classificadas como anomalias dentre as instâncias previstas como anomalias. Enquanto que a *sensibilidade* se refere à taxa de acerto do classificador sobre as instâncias de dados anômalas corretamente detectadas (SELIYA; KHOSHGOFTAAR; VAN HULSE, 2009).

Observe que a precisão e a sensibilidade são complementares, assim quando combinadas possibilitam atingir uma alta qualidade na predição de anomalias. Em vista disso, a *média-F1*, média harmônica entre precisão e sensibilidade, define o “equilíbrio” entre as

classes. Essas métricas podem ser calculadas da seguinte forma (SELIYA; KHOSHGOFTAAR; VAN HULSE, 2009):

$$acurácia = \frac{TP + TN}{N}$$

$$precisão = \frac{TP}{(TP + FP)}$$

$$sensibilidade = \frac{TP}{(TP + FN)}$$

$$média-F1 = \frac{2 \times (precisão \times sensibilidade)}{(precisão + sensibilidade)},$$

em que N é o número total de instâncias do subconjunto de teste.

2.5.2 Agrupamento de Dados

Como apresentado na subseção anterior, em um problema de classificação a classe de cada instância de dados é conhecida previamente. Essa característica diferencia um problema de classificação de um problema de agrupamento de dados ou simplesmente de clusterização (do inglês, *clustering*), pois, neste caso, não se tem conhecimento prévio sobre a classe dos dados. Devido a isso, geralmente, o agrupamento de dados é visto como uma forma de realizar uma análise exploratória dos dados (AGRAWAL; AGRAWAL, 2015).

Atualmente, o objetivo de diversas aplicações do mundo real, utilizadas em domínios como biologia, medicina, negócios e recuperação de informação, é a obtenção de informações relevantes a partir de uma imensa quantidade de dados coletada. Isto é, coletar e armazenar dados é um processo relativamente rápido e simples hoje em dia, mas extrair conhecimento desses dados coletados continua sendo um desafio (HAN; KAMBER; PEI, 2011). Para isso, normalmente, utiliza-se o agrupamento de dados, visto que não se tem conhecimento sobre a existência de classes no conjunto de dados, além do fato de que a classificação ou análise manual é inviável, tanto pelo tempo necessário, quanto pelo custo relativo à mão de obra necessária.

O agrupamento de dados, assim como a classificação, é uma técnica de *Mineração de Dados*, a qual diz respeito à aplicação de técnicas específicas para a identificação e extração de padrões existentes em grandes volumes de dados (KATAYAMA et al., 1994), e relaciona-se diretamente com o conceito de aprendizado não supervisionado de aprendizado de máquina

(HAN; KAMBER; PEI, 2011). Desta forma, uma técnica de agrupamento analisa apenas as informações contidas nos dados recebidos como entrada, isto é, as características que descrevem os objetos desse conjunto de dados e os relacionamentos entre eles, tais como a similaridade e/ou dissimilaridade entre esses objetos (TAN; STEINBACH; KUMAR, 2013).

Em uma tarefa de agrupamento de dados, o conjunto de dados de entrada é dividido em um conjunto de grupos (*clusters*), sendo que esses *clusters* são gerados baseando-se na similaridade entre os objetos do conjunto de dados de entrada. Consequentemente, instâncias de dados do mesmo *clusters* são similares (possuem mais características em comum) entre si, mas dissimilares (diferentes) em relação as instâncias de dados em outros clusters (HAN; KAMBER; PEI, 2011).

Assim, espera-se que uma determinada instância de dados seja mais semelhante as demais instâncias do *cluster* em que ela se encontra. Por conseguinte, essa instância deve ser diferente das instâncias de outros *clusters*. Isto posto, a utilização do conhecimento extraído desses dados depende do contexto de aplicação de uma técnica de agrupamento de dados, pois esses *clusters* podem ser tanto significativos (representam a disposição natural dos dados) quanto úteis (fornecem informações úteis sobre o conjunto de dados para a realização de outras tarefas) (TAN; STEINBACH; KUMAR, 2013).

Sabendo-se disso, faz-se necessário compreender os conceitos de *similaridade* e *dissimilaridade* utilizados pelos algoritmos de agrupamento de dados. A similaridade entre dois objetos, representada por um número positivo s e geralmente $s \in [0,1]$, determina o nível de semelhança desses objetos. Por esse motivo, diz-se que s é proporcional ao grau de semelhança entre os objetos, ou seja, objetos com alto grau de semelhança possuem alta similaridade (o valor de s aproxima-se da similaridade máxima, ou seja, $s = 1$) e objetos com baixo grau de semelhança possuem baixa similaridade (o valor de s aproxima-se de 0, isto é, não existe similaridade) (TAN; STEINBACH; KUMAR, 2013).

A dissimilaridade entre dois objetos, também representada por um número (d) e comumente no mesmo intervalo da similaridade ($d \in [0,1]$), define a diferença desses objetos. Da mesma forma que s , d relaciona-se ao nível de diferença dos objetos. Logo, quanto mais diferentes, maior o valor de d . No caso de uma relação linear, é possível obter diretamente a dissimilaridade a partir da similaridade e vice-versa. Entretanto, em alguns casos, a dissimilaridade assume um intervalo diferente do anterior, neste caso $d \in [0,\infty]$, e isto

inviabiliza a relação linear, fazendo-se necessário uma transformação não linear (TAN; STEINBACH; KUMAR, 2013).

Conforme Tan, Steinbach e Kumar (2013), geralmente, utiliza-se o termo *distância* como um sinônimo de dissimilaridade entre objetos, sendo que esses objetos podem ser tanto uma instância de dados quanto um *cluster*. Além disso, costuma-se usar o termo *proximidade* para referenciar a similaridade e/ou dissimilaridade, mantendo-se o padrão de relacionamento com o nível de semelhança, entre objetos. Vale salientar que, mesmo que seja violada em alguns casos, o termo distância, quando usado como sinônimo de dissimilaridade, representa uma classe específica de dissimilaridades.

Dessa forma, compreende-se como uma medida de distância, tal como a distância Euclidiana, uma medida de dissimilaridade que possui propriedades como: positividade e reflexividade (distância entre os objetos p e q , representada por $d(p, q)$, tem que ser maior ou igual a zero para quaisquer valores de p e q , porém a distância entre dois objetos é igual a zero apenas se os objetos forem iguais); simetria (distância entre os objetos p e q é igual a distância entre os objetos q e p); e desigualdade triangular (calcula-se a distância de um objeto p ao q baseando-se na distância a objetos adjacentes a q). A medida de distância que possui essas características é denominada de métrica (TAN; STEINBACH; KUMAR, 2013).

A Figura 13 ilustra o objetivo geral de um algoritmo de agrupamento de dados. Suponha que o conjunto de dados possui um conjunto de atributos que representam figuras geométricas e que não se tem nenhuma informação sobre o tipo de figura, tais como, círculo, quadrado ou triângulo. Ou seja, baseando-se apenas nas características desses dados, objetiva-se distribuí-los em um conjunto de *clusters* de modo que as figuras mais semelhantes estejam no mesmo *cluster*, além de serem diferentes das figuras dos outros *clusters*.

Figura 13. Relação entre as distâncias *intra-cluster* e *inter-cluster* (Fonte: Adaptado de (TAN; STEINBACH; KUMAR, 2013)).

Em vista disso, e baseando-se nos conceitos de similaridade e dissimilaridade descritos anteriormente, entende-se como *distância intra-cluster* a distância (ou dissimilaridade) entre as instâncias de dados que se encontram no mesmo *cluster*. Da mesma forma que se entende como *distância inter-cluster* a distância entre os diversos *clusters* que representam a distribuição dos dados. Em outras palavras, em relação ao *cluster*, visa-se obter um valor alto de similaridade internamente e baixa externamente (TAN; STEINBACH; KUMAR, 2013).

A análise de agrupamento pode ser utilizada em diversos contextos. Quando objetiva-se a compreensão dos dados por meio de um técnica de agrupamento, obtém-se como resultado da aplicação do algoritmo um conjunto de *clusters* conceitualmente significativos, cujos elementos (instâncias de dados) possuem diversas características em comum e, conseqüentemente, possuem um nível elevado de similaridade (TAN; STEINBACH; KUMAR, 2013). Neste caso, esses *clusters* são considerados como eventuais classes e a análise de agrupamento pode ser definida como o estudo de técnicas capazes de identificar essas classes de forma automática (TAN; STEINBACH; KUMAR, 2013).

Algumas técnicas de agrupamento de dados visam à identificação dos elementos mais representativos em cada um dos *clusters*. Desta forma, a análise de agrupamento possibilita avaliar a representatividade dos objetos individualmente e decidir qual objeto é mais representativo. Além disso, torna possível aplicar técnicas para validar os resultados obtidos pelo algoritmo, baseando-se tanto em métricas de avaliação da literatura quanto no conhecimento do especialista do domínio de aplicação. Neste cenário, a análise de agrupamento é definida como o estudo de técnicas capazes de identificar esses elementos, também chamados de protótipos de *cluster* (TAN; STEINBACH; KUMAR, 2013).

O agrupamento do conjunto de dados, independentemente da abordagem utilizada, varia de acordo com a técnica de agrupamento aplicada sobre os dados. Assim, para um mesmo conjunto de dados, pode-se obter diferentes formas de agrupamentos desses dados. Dentre as formas mais comuns, estão o agrupamento hierárquico (aninhado) e particional (não aninhado), sendo então esses os métodos de agrupamento de dados mais utilizados (TAN; STEINBACH; KUMAR, 2013).

Basicamente, em um agrupamento particional, o conjunto de dados fornecido como entrada para o algoritmo é particionado (dividido) em k *clusters* (grupos), assim k determina o número de *clusters* que deve ser gerado pelo algoritmo e geralmente é um atributo requerido

por algoritmos particionais (HAN; KAMBER; PEI, 2011). Vale salientar que existe uma relação de um-para-um entre uma instância de dados e um *cluster*, isto é, não é possível que uma instância de dados faça parte de dois ou mais *clusters*. Por esse motivo, diz-se que neste tipo de agrupamento não existe sobreposição de objetos, consequentemente, não existe *clusters* sobrepostos (TAN; STEINBACH; KUMAR, 2013).

O agrupamento hierárquico é uma variação de um agrupamento particional, no qual é possível a sobreposição de *clusters*. Por esse motivo, existe o conceito de *sub-cluster*, ou seja, um *cluster* pode ser dividido em *sub-clusters*. Desta forma, tem-se um conjunto de *clusters* aninhados. Além disso, devido a essa característica deste tipo de agrupamento, os *clusters* são organizados de modo semelhante à uma árvore, chamada de *Dendrograma*. Nessa analogia, os nós (*clusters*) da árvore são formados pela união de seus filhos (*sub-clusters*), as folhas representam os objetos individualmente e a raiz da árvore representa um *cluster* que abriga todos os objetos do conjunto de dados (TAN; STEINBACH; KUMAR, 2013).

Com base na argumentação apresentada nesta subseção, é possível observar que as técnicas de agrupamento de dados são utilizadas principalmente em aplicações cuja informação referente a classe dos dados é inexistente. Além disso, é vista como um desafio em aprendizado de máquina, principalmente em mineração de dados (HAN; KAMBER; PEI, 2011). A subjetividade também é uma característica do agrupamento de dados, pois os resultados obtidos podem variar tanto com a aplicação de técnicas diferentes sobre o mesmo conjunto de dados quanto com a mudança na configuração inicial dos algoritmos (HAN; KAMBER; PEI, 2011; TAN; STEINBACH; KUMAR, 2013).

2.5.2.1 O algoritmo *K*-means

Existem diversos algoritmos que podem ser aplicados para a resolução de problemas no contexto do agrupamento de dados, mas, dentre eles, o *K*-means (ou *K*-médias) geralmente é utilizado. Nesta subseção, apresenta-se os conceitos necessários para a compreensão do *K*-means, assim como também aborda-se algumas particularidades do algoritmo, incluindo seu funcionamento.

O *K*-means, uma técnica de aprendizado de máquina, é amplamente aplicado para a solução de problemas de agrupamento de dados. Assim como outras técnicas de aprendizado não supervisionado, o *K*-means aprende as características dos dados baseando-se apenas nas

informações que representam esses dados, isto é, o algoritmo atua de modo não supervisionado (MACQUEEN, 1967).

Trata-se de um algoritmo particional, por isso visa particionar uma população de n dimensões em k *clusters*, sendo que k determina o número de *clusters* e por causa disso tem-se o nome *K-means* (MACQUEEN, 1967). Também se baseia na similaridade entre os objetos do conjunto de dados, logo, utiliza uma medida de distância, geralmente a Euclidiana, para calcular a proximidade entre dois objetos (HAN; KAMBER; PEI, 2011).

O algoritmo funciona da seguinte forma. Inicialmente, o usuário (ou analista de dados) define o número de *clusters* (k), nos quais os dados de entrada devem ser divididos. Em seguida, de modo aleatório, escolhe-se k protótipos (objetos mais representativos dos *clusters*), que neste caso são os centroides. Posteriormente, atribui-se cada uma das instâncias de dados a um dos k *clusters*, de acordo com a proximidade entre a instância e o centroide. Após esse processo, os centroides são recalculados, sendo que cada centroide é calculado baseando-se na média dos elementos de cada *cluster*. Esse processo de distribuição dos objetos nos *clusters*, o cálculo e a “movimentação” dos centroides é realizado até que um critério de convergência do algoritmo seja alcançado, por exemplo, limite de iterações ou inércia de movimentação dos centroides (TAN; STEINBACH; KUMAR, 2013).

Observe que a cada iteração o algoritmo busca movimentar os centroides de modo a diminuir a distância entre eles e os objetos do *cluster*. Devido a isso, diz-se que a função objetivo desse método é diminuir a distância *intra-cluster* (alta similaridade entre os objetos do mesmo *cluster*) e aumentar a distância *inter-cluster* (baixa similaridade entre objetos de *clusters* diferentes) (HAN; KAMBER; PEI, 2011). Vale ressaltar que uma má escolha dos centroides iniciais, acontecimento possível devido a inicialização aleatória, pode afetar diretamente na obtenção dos centroides finais, utilizados para a distribuição final dos objetos. Consequentemente, pode afetar o resultado do algoritmo (ABHILASH; ROHITAKSHA; BIRADAR, 2014).

Uma forma usual de evitar esse problema é aplicar um algoritmo hierárquico sobre uma amostra estatística do conjunto de dados, calcular os centroides e passá-los como entrada para o algoritmo (TAN; STEINBACH; KUMAR, 2013). Ainda, conforme (TAN; STEINBACH; KUMAR, 2013), realizar execuções múltiplas com diferentes conjuntos de centroides, calculados aleatoriamente, e escolher o conjunto de *cluster* (entre todas as execuções) que

minimiza a soma do erro quadrático, também é uma forma de se solucionar o problema de inicialização do *K*-means.

2.5.3 Redes Neurais Artificiais

As redes neurais artificiais (RNAs) foram desenvolvidas tendo como inspiração a estrutura e o funcionamento do sistema nervoso dos seres vivos. Neste contexto, tenta-se reproduzir, por meio de sistemas computacionais, a capacidade de aprendizado do cérebro humano, tanto na aquisição de conhecimento, quanto na propagação e armazenamento do conhecimento adquirido (FACELI et al., 2011).

Basicamente, uma RNA é um sistema computacional composto por neurônios artificiais, unidades (da rede) de processamento simples e responsáveis pelo cálculo de funções matemáticas, densamente interconectados. Geralmente, os neurônios artificiais são distribuídos em diversas camadas da rede, interligadas por um elevado número de conexões (comumente unidirecionais) (FACELI et al., 2011).

Normalmente, essas conexões possuem pesos sinápticos, visto que essas conexões simulam a sinapse biológica, os quais representam a ponderação das instâncias de dados recebidas pelos neurônios da rede (FACELI et al., 2011). Os pesos sinápticos representam a força/expressão dessas conexões e, geralmente, são utilizados para armazenar as informações adquiridas sobre os dados de entrada (conhecimento) (HAYKIN, 2001).

Desta forma, em um processo de aprendizado (também chamado de algoritmo de aprendizado), os valores dos pesos sinápticos são constantemente ajustados e, por esse motivo, codificam o conhecimento obtido pela rede (FACELI et al., 2011). Em vista disso, diz-se que algoritmo de aprendizado visa, de forma ordenada, ajustar os pesos sinápticos da rede em prol de um objetivo, especificado no projeto do algoritmo (HAYKIN, 2001).

Em vista disso, uma rede neural artificial é comumente descrita por meio da arquitetura da rede e do algoritmo de aprendizado utilizado. Sendo que a arquitetura se refere a quantidade de camadas, tipo da rede e o número de neurônios (ou nós, no caso de uma representação por meio de um grafo), assim como a forma de conexão desses neurônios. O algoritmo de aprendizado representa tanto a forma como os pesos sinápticos são ajustados (regras) quanto a informação utilizada com esta finalidade pelas regras implementadas (FACELI et al., 2011).

Vale salientar que os neurônios são essenciais no contexto das RNAs. Basicamente, todas as instâncias do conjunto de dados são passadas como entrada para os neurônios. Cada extremidade de entrada de um neurônio recebe um atributo da instância de dados. Neste caso, cada neurônio da camada, por intermédio de uma função matemática (chamada função de ativação), pondera os valores recebidos. Posteriormente, esses valores são combinados pela mesma função de ativação do neurônio, cuja saída representa a resposta do neurônio para aquela entrada de dados (FACELI et al., 2011).

Em outras palavras, realiza-se um somatório da resposta dada pela função de ativação em cada neurônio, no qual utiliza-se tanto o atributo quanto o peso sináptico atribuído a cada atributo da instância de dados. Por esse motivo, diz-se que a saída de um neurônio é determinada pela aplicação de uma função de ativação sobre a entrada total e, conseqüentemente, a saída de um neurônio representa a sua resposta para aquela entrada de dados (FACELI et al., 2011).

2.5.3.1 *Perceptron* Multicamadas

A rede neural artificial *Perceptron*, juntamente com a rede neural Adaline, foi uma das primeiras RNAs a ser implementada. Trata-se de uma RNA simples que, em sua arquitetura, possui apenas um neurônio e uma função de ativação do tipo *limiar*. Além disso, considera-se que a rede *Perceptron* foi a percussora do processo de treinamento de redes neurais. Essa rede utiliza em seu treinamento um algoritmo de correção de erro. Esse algoritmo objetiva ajustar os pesos da rede de modo a encontrar o hiperplano de separação das classes. Para isto, utiliza um conjunto de dados de treinamento, o qual possui uma série de exemplos de ambas as classes (FACELI et al., 2011).

Apesar de sua simplicidade, essa rede obtém bons resultados em problemas de classificação linearmente separáveis. Todavia, possui a limitação de não conseguir classificar objetos que não possuem essa característica e, por essa razão, não é eficaz em problemas de classificação em um espaço n -dimensional. Nesses problemas tem-se objetos de n -dimensões e não-linearmente separáveis (FACELI et al., 2011).

Geralmente, para suprir a limitação das RNAs de camada única, adiciona-se uma ou mais camadas ocultas (intermediárias). Esse modelo (nomenclatura comum para a arquitetura de uma RNA) é semelhante ao modelo apresentado em (CYBENKO, 1989a). Conforme (CYBENKO, 1989b), uma rede neural com apenas uma camada interna e não-linearidade

sigmoidal contínua e arbitrária é suficiente para aproximar qualquer função contínua. Assim, esse modelo pode “implementar qualquer função contínua” (FACELI et al., 2011). Sendo que, ainda de acordo com (FACELI et al., 2011), é possível executar a aproximação de qualquer função contínua com duas camadas internas.

Sabendo-se disso, redes neurais artificiais do tipo *Perceptron Multicamadas (Multilayer Perceptron – MLP*, em inglês), geralmente, possuem camadas auxiliares entre as camadas de entrada e saída, comumente chamadas camadas ocultas ou intermediárias. Essas camadas são totalmente conectadas entre si, ou seja, os neurônios de uma camada são conectados a todos os neurônios da seguinte (GARDNER; DORLING, 1998).

Uma MLP, por ser um tipo de RNA *feedforward* (BEBIS; GEORGIOPOULOS, 1994), possui um fluxo de transmissão de informação de sentido único, neste caso, sempre para a camada seguinte da rede (“frente”). Neste fluxo, a informação flui desde a camada de entrada dos dados até os neurônios da camada de saída (FACELI et al., 2011). Uma MLP executa um mapeamento não linear entre entrada e saída, de modo que a saída corresponda ao valor esperado para uma determinada entrada (GARDNER; DORLING, 1998).

Por fim, vale ressaltar que as camadas de entrada e saída podem ser representadas por meio de vetores. Devido a isso, a camada de entrada não desempenha nenhuma função computacional e, por isso, é usada apenas para introduzir os valores dos dados de entrada na rede (GARDNER; DORLING, 1998). Além disso, uma MLP consegue aproximar funções extremamente não lineares. Por conseguinte, aumentar o número de camadas intermediárias do modelo aumenta o seu nível de não-linearidade e, assim, é possível obter bons resultados em problemas complexos (GARDNER; DORLING, 1998).

2.5.3.2 Mapas Auto-Organizáveis

Mapas auto-organizáveis (*Self-Organizing Maps – SOM*, em inglês) (KOHONEN, 1990) consistem em uma rede neural artificial não supervisionada (FACELI et al., 2011). A rede SOM, vista como uma técnica de aprendizado não supervisionado, é amplamente utilizada em tarefas de agrupamento e visualização de dados (FACELI et al., 2011). Essa rede, em diversas áreas de pesquisas, tais como indústria, medicina e negócios, também é utilizada na análise exploratória dos dados, cujo um dos objetivos é a compreensão dos dados coletados (KOHONEN, 2013).

Por ser uma RNA, normalmente com apenas uma camada computacional, a rede SOM possui um conjunto finito de neurônios, os quais se conectam a todas as instâncias de dados (entrada) da rede (FACELI et al., 2011). A rede neural artificial SOM, por intermédio desses neurônios, comumente representa a distribuição do conjunto de dados de entrada – instâncias de dados – em uma matriz uni ou bidimensional (FAUSETT; OTHERS, 1994).

Os neurônios da rede possuem duas propriedades básicas: posição, a qual refere-se à posição do neurônio na matriz, e conexões com os neurônios adjacentes a ele, comumente chamados de vizinhos. A rede SOM, assim como o algoritmo *K-means* apresentado anteriormente, baseia-se na similaridade entre as instâncias de dados. Por causa disso, os neurônios com um elevado nível de similaridade associam-se automaticamente com os seus vizinhos, enquanto que neurônios com um baixo nível de similaridade tendem a ficar distantes uns dos outros na matriz (KOHONEN, 2013)

Essa característica da rede SOM proporciona uma compreensão das relações topográficas entre os dados, principalmente quando as instâncias de dados possuem um número elevado de dimensões. Desta forma, grandes volumes de dados, multidimensionais e complexos, podem ser representados em um espaço bidimensional, ou seja, de menor dimensão (QIANG; CHENG; LI, 2010; KOHONEN, 2013). Em vista disso, essa técnica de aprendizado não supervisionado é comumente utilizada para a redução de dimensionalidade dos dados (QIANG; CHENG; LI, 2010).

A rede SOM, assim como as demais técnicas de aprendizado não supervisionado – exemplos comuns de aprendizado competitivo –, aprende por meio do aprendizado competitivo. Isto é, a rede de modo independente (por conta própria), assimila as informações sobre o conjunto de dados de forma não supervisionada e competitiva (QIANG; CHENG; LI, 2010). Com relação às propriedades básicas de um neurônio, durante o processo de treinamento dessa RNA, as conexões entre os neurônios permanecem inalteradas. Contudo, devido a um processo de atualização característico da rede SOM, a posição de um neurônio na matriz muda conforme as iterações de treinamento.

A RNA SOM aprende da seguinte forma: inicialmente, antes de iniciar o treinamento da rede, define-se tanto as conexões entre neurônios e seus vizinhos quanto as suas posições, as quais são normalmente inicializadas de modo aleatório. A cada iteração de treinamento, uma nova instância de dados é adicionada à rede (QIANG; CHENG; LI, 2010; KOHONEN, 2013).

Em vista disso, os neurônios calculam os seus respectivos valores de ativação para essa nova instância, e isso faz com que diferentes regiões da matriz sejam ativadas durante o treinamento (FACELI et al., 2011).

Posteriormente, inicia-se um processo de competição entre os neurônios para encontrar o neurônio mais próximo da instância de dados inserida. Normalmente, utiliza-se a distância euclidiana como função de ativação dos neurônios. O neurônio mais próximo da instância de dados adicionada, ou seja, com o maior valor de ativação, é chamado de *neurônio vencedor*. Ao final de cada iteração, atualiza-se o neurônio vencedor, as posições dos seus vizinhos e (em relação a topologia dos neurônios da rede ou estrutura do mapa) ajusta-se os pesos de todos esses neurônios (QIANG; CHENG; LI, 2010; KOHONEN, 2013). Esse processo de ajuste dos pesos aumenta o nível de similaridade entre o neurônio, juntamente com os seus vizinhos, e novas instâncias de dados. Além disso, a quantidade de vizinhos de um neurônio é gradativamente reduzida (FACELI et al., 2011).

Por fim, geralmente ao utilizar a rede SOM, objetiva-se tanto obter um conjunto de vetores de referência quanto vincular, baseando-se no vetor de referência mais próximo, as instâncias do conjunto de dados a um desses vetores. Consequentemente, obtém-se o agrupamento dos dados de entrada. Observe que, fazendo uma analogia ao algoritmo *K-means*, esses vetores de referência desempenham a função dos centroides. Portanto, a rede SOM também depende diretamente da inicialização desses vetores (FACELI et al., 2011).

2.5.3.3 *Autoencoders*

Um *Autoencoder* (BALDI, 2012) é uma rede neural artificial não supervisionada similar a rede *Perceptron* Multicamadas – MLP. Em vista disso, por mais que possua uma arquitetura semelhante a de uma MLP, utiliza uma abordagem de aprendizado não supervisionado, assim, é capaz de aprender automaticamente diversas características (atributos) a partir de um conjunto de dados não rotulado (NG; OTHERS, 2011).

Trata-se de uma técnica amplamente aplicada, devido a suas características, em diversos contextos, tais como agrupamento de dados, detecção de anomalia, transferência de aprendizado, redução de dimensionalidade e extração de atributos (NG; OTHERS, 2011; GOODFELLOW; BENGIO; COURVILLE, 2016). Geralmente, a arquitetura de um *autoencoder* é composta por duas partes: uma função de codificação (*encoder*), cujo objetivo é

aprender uma representação do conjunto de dados de entrada, e uma função de decodificação (*decoder*), cujo objetivo é executar a reconstrução desses dados (GOODFELLOW; BENGIO; COURVILLE, 2016).

Desta forma, tem-se uma camada de codificação, a qual comprime a informação contida nos dados de entrada, e decodificação, que realiza o processo reverso e reconstrói a informação original, interligadas por uma camada de representação compacta (chamada de codificador, *code* em inglês). Vale ressaltar que, assim como a rede MLP, o *autoencoder* é do tipo *feedforward* e pode conter uma ou mais camadas ocultas. Observe que para cada camada de codificação inserida, deve-se inserir também uma de decodificação (GOODFELLOW; BENGIO; COURVILLE, 2016).

Um *autoencoder* aprende como representar os dados recebidos como entrada. Por esse motivo, visa transformar as instâncias de dados, recebidas como entrada, em saídas de dados. Esse processo de reprodução dos dados de entrada na camada de saída deve ser realizado com uma quantidade mínima, caso seja possível, de discrepância entre os dados originais e os de saída (BALDI, 2012). Basicamente, no processo de treinamento de um *autoencoder*, estimula-se a função treinada na camada de codificação, por meio da tarefa de reproduzir entradas nas saídas, a aprender características (propriedades) úteis sobre os dados de entrada (NG; OTHERS, 2011; GOODFELLOW; BENGIO; COURVILLE, 2016).

3 REVISÃO DA LITERATURA

Neste capítulo apresenta-se a revisão da literatura, por meio de diversos trabalhos científicos, realizada visando analisar as pesquisas que abordam os conceitos apresentados no capítulo anterior e que são fundamentais para o desenvolvimento desta dissertação. Neste contexto, dentre os resultados da revisão, tem-se uma visão geral das pesquisas que apresentam, especificamente, a manutenção baseada em condição, um conjunto de algoritmos de detecção e diagnóstico de falhas, desenvolvidos tanto para a aplicação em um domínio do mundo real quanto para incrementar o conhecimento dessa área em aprendizado de máquina. Por fim, descreve-se as principais semelhanças e diferenças entre as pesquisas encontradas na literatura e o presente trabalho.

3.1 VISÃO GERAL SOBRE MANUTENÇÃO BASEADA EM CONDIÇÃO

Esta subseção apresenta pesquisas que abordam especificamente a *manutenção baseada em condição*. Desta forma, descreve-se a seguir, os estudos que objetivam o desenvolvimento científico dessa área, incluindo o seu estado da arte.

O estudo apresentado em Hasheiman e Bean (2011) realiza um levantamento sobre o estado-da-arte da área de manutenção preditiva. Sabendo-se disso, descreve as três principais técnicas de manutenção preditiva – sensores, teste de sensores e teste de sinais – que podem ser aplicadas em diversos domínios do mundo real, tais como usinas nucleares e o setor industrial. Também apresenta o comportamento das funções, ou curvas, que representam as probabilidades de falha de diversas categorias de equipamento ao longo do tempo. Além disso, afirma que a integração dessas técnicas de manutenção com a tecnologia dos sensores, possibilita às indústrias e/ou usinas a evitar a troca desnecessária de equipamentos, a reduzir custos e a aprimorar a segurança, disponibilidade e a eficiência do processo de produção industrial.

Uma visão geral da manutenção baseada em condição, também chamada por diversos autores de manutenção preditiva, pode ser encontrada no trabalho proposto por (AHMAD; KAMARUDDIN, 2012). Neste contexto, apresenta-se também uma visão geral sobre a manutenção periódica. Com este propósito, descreve-se diversos artigos que realizam a aplicação de alguma dessas técnicas nos mais variados domínios de aplicação. Além disso, discute-se como essas formas de manutenção funcionam no processo de tomada de decisão de manutenção. Por fim, realiza-se uma comparação entre os desafios para a implementação de

cada uma dessas técnicas, principalmente com relação à aquisição de dados, incluindo a sua análise e/ou modelagem, e a tomada de decisão. Sabendo-se disso, e baseando-se nesses fatores, esse trabalho conclui que a aplicação da manutenção baseada em condição é mais realista do que a aplicação da manutenção periódica.

No estudo exposto em Shin e Jun (2015) discute-se diversos aspectos da manutenção baseada em condição, principalmente, a sua definição em vários pontos de vista, vantagens e desvantagens, e as técnicas que podem ser aplicadas sobre um conjunto de dados com o intuito de se implementar uma política de manutenção baseada em condição (preditiva). Assim, também é apresentado, incluindo um breve detalhamento das etapas, o procedimento para a implantação dessa abordagem. Além disso, descreve-se brevemente diversos estudos de caso que exemplificam a aplicação dessa política em diferentes equipamentos.

Geralmente, devido à possibilidade de aquisição de grandes quantidades de dados, de modo rápido e contínuo, proporcionada pela Indústria 4.0, aplicam-se técnicas de aprendizado de máquina sobre os dados coletados para realizar a manutenção preventiva de um determinado equipamento. Por isso, os métodos de aprendizado de máquina são considerados como uma ferramenta, emergente e promissora, para aplicação da manutenção preditiva. No estudo apresentado em (CARVALHO et al., 2019), por meio de uma revisão da literatura, apresenta-se o estado da arte das técnicas de aprendizado de máquina aplicadas em manutenção preditiva. Desta forma, descreve-se brevemente tanto os principais métodos utilizados, tais como redes neurais artificiais, *K*-means, e máquinas de vetores de suporte, com o propósito quanto às etapas necessárias para o desenvolvimento de um método de aprendizado de máquina para a aplicação da manutenção preditiva.

3.2 DETECÇÃO E DIAGNÓSTICO DE FALHAS EM FERROVIAS

Atualmente, diversas pesquisas objetivam analisar a origem da falha de um equipamento por intermédio dos métodos FDDs. Em geral, abordam a aplicação desses métodos sobre domínios do mundo real, tais como o setor ferroviário. Portanto, as pesquisas apresentadas nesta subseção visam a aplicação de algoritmos de aprendizado de máquina para a detecção e diagnóstico de falhas em ativos da via permanente.

Neste contexto, uma revisão sucinta sobre o estado da arte em métodos preditivos para a detecção e diagnóstico de falhas para ativos do sistema ferroviário, tais como as máquinas de

chave e aparelhos de mudança de via, é apresentada em (CHEN; ROBERTS, 2006). Neste caso, apresentam práticas comumente utilizadas na época para o monitoramento das condições de uso desses equipamentos. Além disso, propõem um método *neuro-fuzzy* de detecção e diagnóstico de falhas, utilizado em um estudo de caso sobre os dados de circuitos de via – CDVs, o qual conseguiu diagnosticar corretamente mais de 98% dos dez tipos mais comuns de falha.

Em Asada, Roberts e Koseki (2013) é proposto um método para a detecção e o diagnóstico de falhas em máquinas de chave. Com este propósito, esse estudo utiliza técnicas como *Discrete Wavelet Transform – DWT*, para seleção de atributos, e o C-SVM, uma variação de Máquinas de Vetores Suporte (*Support Vector Machines - SVMs*), para o diagnóstico das falhas. Os atributos monitorados, calculados de acordo com os dados de corrente e tensão elétrica do motor, são coletados por meio de sensores acessíveis e de baixo custo. Esses atributos formam a base de dados utilizada em seus experimentos, além de representar os modos de falhas analisados.

O trabalho proposto por Sun e Zhao (2013) também faz a proposição de um algoritmo, para o diagnóstico de falhas, que utiliza SVM, porém o método visa a classificação de várias classes (*multi-class*) e baseia-se na estratégia um-contra-um (*one-versus-one*). Além disso, aplica-se esse método sobre os dados de um ativo diferente da via permanente, neste caso, dos circuitos de via (especificamente, sobre um componente-chave desse ativo chamado junta de separação elétrica (*Electrical Separation Joint – ESJ*)). Os atributos, provindos do sinal (também conhecido como corrente de curto-circuito) enviado pelo circuito de via, representam as características das falhas e são utilizados no treinamento do modelo de diagnóstico. Por fim, realiza-se uma comparação entre as funções polinomiais e a *Gaussian RBF* utilizadas como a função de classificação do algoritmo.

Ainda no cenário da detecção e do diagnóstico de falhas, algumas pesquisas utilizam-se da análise de séries temporais. Por exemplo, De Bruin, Verbert e Babuska (2017) analisam os dados, gerados por intermédio de um método (utiliza tanto o conhecimento de domínio, especificamente, sobre o comportamento das falhas quanto dados reais de uma via) que baseia-se na corrente de curto-circuito (sinal), de diversos circuitos de via em uma determinada área da ferrovia, para diagnosticar falhas nesse ativo a partir de suas dependências espaciais e temporais. Além disso, realizam uma comparação entre uma rede neural recorrente do tipo *LSTM (LSTM-RNN)*, a qual conseguiu classificar corretamente 99,7% do conjunto de teste, sem

falsos positivos na etapa de detecção de falhas, e uma rede neural convolucional (CNN). Por fim, como resultado da comparação, esse trabalho conclui que a arquitetura da RNN é melhor, na tarefa de detecção e diagnóstico de falhas, do que a CNN.

Em Atamarudov et al. (2009) é proposto um sistema especialista que, baseando-se na análise de séries temporais, realiza a detecção e diagnóstico de falhas em um cenário cuja separação dos dados não é tão visível. Em outras palavras, existe uma alta similaridade entre os dados. Nesta situação, utiliza-se a corrente elétrica da Máquina de Chave – MCH para representar os estados defeituosos e íntegros (“saudáveis”). Sabendo-se disso, a progressão de uma falha, entre os modos de falha estudados, é calculada por meio de uma média ponderada baseando-se nos dados das séries temporais que representam a variação da corrente de um estado sem falha para um estado falho (algum modo de falha). Por fim, para definir os limites entre os modos de falha e identificar a transição entre os estados, utiliza-se um conjunto de valores (*thresholds*), extraídos da corrente elétrica da MCH, definidos por uma série de regras de domínio usados pelo sistema especialista na etapa de diagnóstico de falha.

Jin et al. (2015) propõem um método não-intrusivo para analisar a degradação de desempenho de uma máquina de chave, utilizando como dado apenas os sinais de potência do motor. Além disso, realizam uma comparação entre os algoritmos $^3SOM-MQE$ e $PCA-T^2$, usados para detectar os modos de degradação das máquinas de chave e avaliar o seu respectivo nível de severidade, no processo de monitoramento do desempenho. Em seguida, por meio da rede SOM, realizam o diagnóstico da falha. Por fim, os experimentos são realizados com base em três diferentes modos de falhas, desalinhamento, obstáculo e travamento, previamente definidos. Vale ressaltar que eles também desenvolvem um conjunto de ferramentas para o monitoramento a bordo das condições de operação da MCH, visando a detecção precoce de falhas.

Em Bian et al. (2019) também é proposto um algoritmo que visa, por meio do uso de dados normais, distinguir e identificar os estados de degradação (percorridos por um equipamento até a ocorrência de uma falha) dos diferentes modo de falhas de uma máquina de chave. Em razão disso, é realizado o processamento dos dados sem falhas por intermédio de uma estratégia que combina um conjunto de técnicas comumente usadas no processamento de

³ Mapas auto organizáveis – Erro mínimo de quantização (Self-Organizing Map – Minimum Quantization Error – SOM-MQE) e Análise dos componentes principais – T^2 (Principal Component Analysis – PCA- T^2).

dados. Vale ressaltar que esses atributos são obtidos a partir da análise da corrente elétrica de uma MCH. Assim, por intermédio dessa combinação de técnicas, tem-se um conjunto de atributos que representam o estado de degradação da MCH. Em seguida, para a obtenção dos estados de degradação que descrevem as diferentes condições do ativo, aplica-se a rede *SOM* sobre o conjunto de dados. Por fim, eles usam um algoritmo implementado por meio da combinação entre o *PSO* e o *SVM*, chamado *PSO-SVM*, para identificar os diferentes estados de degradação encontrados pelo *SOM*.

Dada a severidade dessas atividades, incluindo as suas dificuldades, existem pesquisas que visam o desenvolvimento de algoritmos, que incrementem o conhecimento da área ou da técnica utilizada, específicos para a detecção ou diagnóstico de falhas. Por exemplo, (VILEINISKIS; REMENYTE-PRESCOTT; RAMA, 2016a) propõem uma nova abordagem, e consequentemente um algoritmo, para a detecção de falhas em máquinas de chave. Neste contexto, a detecção de uma falha é vista como uma tarefa de classificação. Com este propósito, utilizam o algoritmo *OCSVM* (*One Class Support Vector Machines*) e a *ERP* (*Edit distance with Real Penalties*) como mediada de similaridade. Para o treinamento do modelo *OCSVM*, utiliza-se um conjunto de dados, o qual é constituído por um número específico de atributos obtidos a partir de dados reais da corrente elétrica de uma MCH.

Em todos esses estudos faz-se necessário o conhecimento prévio dos prováveis tipos de falhas que podem ocorrer no equipamento. Assim, baseando-se nesses tipos de falhas, os autores propõem métodos capazes de diagnosticá-las de forma supervisionada. Porém, em diversas aplicações do mundo real, nem sempre é possível, de antemão, obter essas informações. Neste caso, pode-se utilizar algoritmos que objetivam, de forma não supervisionada, descobrir essas informações por intermédio da exploração do conjunto de dados e do reconhecimento de padrões. Neste contexto, Amruthnath e Gupta (2018) realizam uma análise de desempenho entre diversos algoritmos de agrupamento, tais como *K-means*, *SOM* e *Fuzzy C-Means*, capazes de detectar previamente falhas, baseando-se em dados de vibração, de um determinado equipamento.

Sabendo-se disso, um sistema pode ser considerado inteligente mediante a sua capacidade de analisar uma imensa quantidade de dados e realizar decisões eficientes. Por exemplo, no caso da manutenção preditiva, o sistema, dada as condições atuais de funcionamento do equipamento, deve ser capaz de decidir quando executar ou não a manutenção do mesmo.

Essa decisão pode ser tomada com base no conhecimento adquirido por meio de uma análise exploratória dos dados, isso depende exclusivamente da robustez do programa de manutenção preditiva aplicado. Esse tipo de análise, por conta da imensa quantidade de dados, é geralmente efetuado por intermédio da aplicação de algoritmos de *clusterização* sobre os dados.

Em vista disso, o estudo apresentado em (DATTA; MAVROIDIS; HOSEK, 2008) discute a função desses algoritmos de *clusterização* no processo de tomada de decisões. Desta forma, esse estudo revisa diversos métodos não supervisionados existentes, tais como mapas auto-organizáveis (SOM), algoritmos hierárquicos e limiar adaptativo de qualidade (*QT*), métodos de validação do número ótimo de *clusters*, e propõe um método (*SOMQT*) baseado na combinação dos algoritmos *SOM* e *QT*. Por fim, esse trabalho apresenta uma estrutura geral para o desenvolvimento de algoritmos de detecção e diagnósticos de falhas, incluindo o envolvimento dos algoritmos de *clusterização* no processo de tomada de decisão.

3.3 COMPARATIVO COM OS TRABALHOS CORRELATOS

Na subseção anterior foram apresentados diversos trabalhos que têm como objetivo a detecção e diagnóstico de falhas, principalmente pesquisas que aplicam esses métodos sobre os ativos da via permanente, tais como circuitos de via e aparelhos de mudança de via. Desta forma, de modo a distinguir os métodos que serão propostos nesta dissertação dos disponíveis na literatura, descreve-se a seguir as principais diferenças e semelhanças identificadas durante a análise da literatura, em relação aos métodos que serão propostos nesta dissertação.

Assim, as principais características que constituem as semelhanças e as diferenças entre o presente trabalho e a literatura revisada, a qual foi brevemente apresentada por intermédio das pesquisas descritas anteriormente, são listadas a seguir:

1. Em relação aos dados utilizados para a realização dos experimentos, a maioria das pesquisas também utiliza a assinatura elétrica da corrente de acionamento da Máquina de Chave de um AMV. Vale ressaltar que em alguns casos, com o mesmo propósito, usam a corrente de curto-circuito (CDV) e a potência do motor (MCH). Além disso, apenas algumas pesquisas também abordam temas como a utilização de dados não balanceados, um problema comum em bases de dados ferroviárias, e dados com alto grau de similaridade entre os dados;

2. A maioria dos trabalhos apresentados visam a detecção e o diagnóstico de falhas em um determinado ativo, isto é, atuam com o propósito de identificar falhas previamente. Porém, não é claro o intuito de ser protagonista, como uma ferramenta auxiliar, na implantação de um programa de manutenção preditiva desse equipamento (por mais que auxiliem na manutenção do ativo). Além disso, nos trabalhos descritos, realizar uma comparação de desempenho entre diversos métodos FDDs não é um objetivo secundário;
3. O modo como o conjunto de dados é coletado, bem como a sua utilização, são fatores fundamentais para a proposição de um algoritmo FDD. Neste contexto, em sua maioria, essas pesquisas obtêm os resultados apresentados por meio de experimentos em laboratórios. Em outras, os dados são obtidos por intermédio de uma medição controlada e monitorada desses equipamentos em laboratórios. Assim, apenas alguns trabalhos utilizam dados reais, oriundos do campo (ferrovia). Além disso, em todos os trabalhos descritos, o diagnóstico é realizado com base em um conjunto de modo de falha previamente definido. Em outras palavras, os tipos de falhas que podem acontecer no ativo são conhecidos, seja pela definição de um valor limite de operação (*thresholds*), comumente aplicado, ou pelo conhecimento do especialista do domínio. Desta forma, não se obtém como resultado da aplicação do algoritmo, em conjunto com o conhecimento do especialista de domínio, os principais tipos de falhas e, conseqüentemente suas causas, que ocorrem no ativo;
4. Devido ao fato de os tipos de falha serem previamente definidos, as demais pesquisas abordam principalmente técnicas de aprendizado supervisionado. Os métodos propostos no presente trabalho utilizam, principalmente, técnicas de aprendizado não supervisionado;
5. De modo geral, técnicas como os Mapas Auto-organizáveis (SOM) e o *K-means* são utilizadas em algumas das pesquisas mencionadas na subseção anterior. No entanto, não foi encontrado na SLR realizada um trabalho que utilize essas técnicas, tanto na tarefa de detecção quanto no diagnóstico de falhas. Este trabalho também utiliza, além do SOM e *K-means*, *Auto-encoders*. No próximo capítulo descreve-se a forma como cada uma dessas técnicas é utilizada;
6. Vale ressaltar que, em sua minoria, os métodos propostos na literatura são utilizados em um ambiente real. Por mais que sejam aplicados sobre os dados de um ativo da via, não são implantados no mundo real.

Por fim, baseando-se na argumentação apresentada nesta subseção, a *Tabela 5* apresenta uma comparação entre o método proposto pelo presente trabalho, denominado como **SOS:D²** (*System Operational Status Detection and Diagnosis*), e a literatura. A *Tabela 5* é composta tanto pelas propriedades (colunas), que representam as principais características analisadas quanto pelos trabalhos (linhas). A seguir, define-se cada uma das propriedades (**PROPs**) consideradas:

1. **Ativo (PROP1):** Especifica o ativo sobre o qual o método FDD é aplicado;
2. **Sinal (PROP2):** Determina a grandeza pela qual os dados são obtidos e, consequentemente, como o equipamento é monitorado;
3. **Ambiente de aplicação (PROP3):** Representa o ambiente cujos métodos FDDs foram aplicados. Neste caso, especifica se o método foi proposto utilizando dados reais, capturados em campo, de uma ferrovia, ou se foi proposto utilizando dados simulados obtidos em laboratório;
4. **Ferramenta auxiliar em um programa de manutenção (PROP4):** Ao se propor um método FDD, tem-se também, indiretamente, como objetivo, a manutenção do equipamento. No entanto, este atributo refere-se especificamente ao auxílio na tomada de decisão do operador, atuando como uma ferramenta, em um programa de manutenção preditiva de uma ferrovia;
5. **Módulo de falhas (PROP5):** Especifica se os tipos de falha do equipamento analisados na pesquisa são previamente conhecidos. Isto é, se técnicas de aprendizado não supervisionado são utilizadas no processo de definição dos modos de falha a serem analisados.

Tabela 5. Comparativo entre o SOS:D² e a literatura, especificamente dos métodos de detecção e diagnóstico de falhas aplicados no setor ferroviário (**Fonte:** Autor).

Trabalho	PROP1	PROP2	PROP3	PROP4	PROP5
[Chen, 2016]	CDV	Corrente de curto-circuito	Real	Não	Conhecido
[Asada, 2013]	AMV	Tensão e corrente elétrica do motor	Simulado	Não	Conhecido
[Sun, 2013]	CDV	Corrente de curto-circuito	Simulado	Não	Conhecido
[De Bruin, 2017]	CDV	Corrente de curto-circuito	Simulado	Não	Conhecido
[Atamuradov, 2009]	CDV	Corrente de curto-circuito	Simulado	Não	Conhecido
[Jin, 2015]	AMV	Sinais da potência do motor	Real	Não	Conhecido
[Bian, 2019]	AMV	Corrente elétrica da MCH	Real	Não	Conhecido
[Vileiniskis, 2016]	AMV	Corrente elétrica da MCH	Real	Não	Conhecido
SOS:D²	AMV	Corrente elétrica da MCH	Real	Sim	Desconhecido

Finalmente, conforme a *Tabela 5*, o SOS:D² diferencia-se da literatura, principalmente, por abordar a tarefa de detecção e diagnóstico de falhas de uma forma não supervisionada. Neste contexto, visa tanto a identificação dos principais modos de falhas quanto auxiliar o operador ferroviário no processo de tomada de decisão, como uma ferramenta, em um programa de manutenção preditiva.

4 DETECÇÃO E DIAGNÓSTICO DO ESTADO DE OPERACIONALIDADE DE SISTEMAS

Este capítulo tem o objetivo de apresentar a metodologia proposta para a *Deteção e Diagnóstico do Estado Operacional de um Sistema*, denominada como “***System Operational Status Detection and Diagnosis***” (***SOS:D²***). Primeiramente, apresenta-se uma visão geral do processo desenvolvido para identificar o estado de operacionalidade do sistema. Em seguida, descreve-se as etapas necessárias para o diagnóstico do nível de degradação do sistema e/ou equipamento, incluindo a forma como as técnicas são aplicadas em cada etapa do processo.

4.1 A METODOLOGIA PROPOSTA

Geralmente, um sistema e/ou equipamento está sujeito a um processo natural de degradação, no qual, primeiramente, tem-se o início da deterioração do equipamento, provocada pelo surgimento de uma falha – problema físico ou imperfeição. A manifestação dessa falha – erro – ocasiona o desgaste desse equipamento. Esse desgaste leva o equipamento para um estado intermediário no qual ele continua exercendo sua função, mas com um desempenho abaixo do esperado. Existem diversos estados intermediários de degradação, os quais representam os diferentes níveis de operacionalidade do equipamento, entre o início da degradação e o momento em que o nível de degradação do equipamento está acima dos valores máximos estabelecidos, o que pode causar a disfunção do equipamento (incapacidade de operar).

Portanto, quando o nível de operabilidade do equipamento encontra-se abaixo do mínimo permitido, existe uma alta probabilidade de ocorrência de uma disfunção e, consequentemente, interrupção do serviço prestado por esse equipamento. Todavia, considerando esse processo natural de degradação, é possível mensurar o nível operacional do equipamento por meio da análise dos estados intermediários de degradação (BIAN et al., 2019). Para isso, é conveniente definir a criticidade de cada estado.

Neste contexto, por meio da identificação do nível de criticidade (degradação) de um estado intermediário, o qual pode ser categorizado em escalas de baixa, média e alta severidade, é factível antever a disfunção de um equipamento e, assim, aplicar a manutenção corretiva programada para restaurá-lo a um bom nível operacional. Devido a isso, considera-se que um estado intermediário representa uma falha incipiente e que esse procedimento de identificação

do estado de operacionalidade pode ser realizado por meio do diagnóstico prévio dessa falha (ASADA; ROBERTS, 2011).

Entende-se por uma falha incipiente o desgaste do equipamento que ocorre gradativamente, isto é, o seu nível de operacionalidade vai se reduzindo, por causa de uma falha, ao longo do tempo. Em função disso, diagnosticar previamente essas falhas possibilita acionar a equipe de manutenção em tempo hábil para o planejamento do reparo do equipamento, proporcionando eficiência na produção (redução das paradas de emergência, ocasionadas pela interrupção abrupta do serviço prestado pelo equipamento) com viés de segurança e disponibilidade operacional.

Resumidamente, o processo de diagnóstico do estado de degradação (operacionalidade) de um sistema e/ou equipamento pode ser visualizado em duas etapas. São elas: a *detecção*, responsável por detectar o estado operacional, e o *diagnóstico*, o qual especifica o que ocasionou o desgaste (caso seja identificado na etapa de detecção). Cabe destacar que essas etapas constituem os métodos de detecção e diagnóstico de falhas (*Fault Detection and Diagnosys – FDD*).

Neste contexto, os métodos FDDs – implementados no SOS:D² por meio da combinação de diversas técnicas de aprendizado de máquina – serão utilizados para o diagnóstico prévio de falhas incipientes. Dessa forma, para cada estado de degradação do ativo tem-se um provável tipo (modo) de falha. Além disso, é factível realizar uma pressuposição do seu grau de criticidade baseando-se em sua discrepância em relação ao estado operacional ideal.

Vale ressaltar que esses métodos podem ser utilizados como uma ferramenta que auxilia o operador durante o monitoramento das condições de uso de um equipamento. Por esse motivo, podem ser vistos como um componente de um programa de manutenção preditiva (*Predictive Maintenance – PdM*) que atua na etapa de diagnóstico. O diagnóstico de falhas é uma etapa importante desse processo, visto que é fundamental para a estimativa do tempo restante de vida útil do equipamento (etapa de prognóstico) (SHIN; JUN, 2015).

Em vista disso, faz-se necessário compreender o processo de desenvolvimento de um algoritmo, desenvolvido no contexto de aprendizado de máquina, que visa a aplicação da manutenção preditiva. Segundo (CARVALHO et al., 2019), esse processo envolve uma série

de etapas, são elas: a *aquisição dos dados*, o *pré-processamento de dados*, a *seleção do modelo*, o *treinamento e validação do modelo*, e, finalmente, a *manutenção do modelo*.

Na etapa inicial do desenvolvimento do algoritmo (aquisição dos dados), definem-se os dados que serão coletados e, conseqüentemente, utilizados para o seu desenvolvimento. Em seguida, na etapa de pré-processamento, realiza-se um processo que visa a transformação dos dados coletados em um novo conjunto de dados. Geralmente, nesse processo de transformação dos dados, executa-se atividades como a *normalização e limpeza dos dados*, *remoção de outliers* e *redução de atributos*. Em virtude disso, esse novo conjunto de dados pode ser utilizado de forma mais eficiente pelo modelo de aprendizado de máquina escolhido (CARVALHO et al., 2019).

No caso da PdM, utiliza-se essa etapa para compreender, analisar e interpretar melhor os dados coletados. Posteriormente, efetua-se o desenvolvimento do modelo de aprendizado de máquina que será utilizado. Após a escolha do modelo, executa-se o seu treinamento e avaliação. Essa etapa possibilita tanto treinar o modelo para que ele aprenda, por meio de exemplos, algumas características do conjunto de dados, quanto avaliar o quão bem o modelo aprendeu essas propriedades. Por fim, tem-se a manutenção do modelo implementado. Nesta etapa objetiva-se manter o desempenho do modelo desenvolvido ao longo do tempo (CARVALHO et al., 2019). Cabe enfatizar que a manutenção do modelo não faz parte do escopo do SOS:D².

Dito isto, baseando-se nas etapas descritas, apresenta-se um processo geral, ilustrado em alto nível na *Figura 14*, para o desenvolvimento e avaliação dos métodos de detecção e diagnóstico de falhas propostos no SOS:D². Observe que, neste caso, considera-se que a etapa de aquisição dos dados foi devidamente realizada, representada pela etapa 1 do processo (pré-processamento dos dados). Por conseguinte, tem-se o conjunto de dados do equipamento, com as suas principais características, que deve ser analisado.

Figura 14. Processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas – FDD (Fonte: Autor).

A sistemática (metodologia) proposta, que objetiva o desenvolvimento de modelos baseados em aprendizado de máquina para aplicação da manutenção preditiva, constitui-se de três etapas: **(1.) Pré-processamento dos dados**; **(2.) Desenvolvimento dos modelos de Detecção e Diagnóstico de Falhas utilizando técnicas de Aprendizado de Máquina**; e **(3.) Avaliação dos modelos de Detecção e Diagnóstico de Falhas**. A seguir, descreve-se cada uma dessas etapas.

1. Pré-processamento dos dados

Geralmente, é necessário realizar um processo de transformação (pré-processamento) sobre os dados provindos de um ambiente real. Esses dados, devido as suas características, possuem uma baixa representatividade do seu domínio de aplicação e, por isso, não costumam ser utilizados diretamente pelos algoritmos.

Por exemplo, no setor ferroviário – domínio de aplicação deste trabalho – a coleta dos dados é comumente feita por meio de sensores de corrente, os quais são propensos a interferências e erros de leitura. Por causa disso, em alguns casos, tem-se dados ruidosos e que podem afetar diretamente as atividades

executadas sobre esses dados. Logo, o pré-processamento dos dados é utilizado como uma forma de aprimorar a qualidade (e representatividade) dos dados, além de evitar o uso de dados ruidosos.

No contexto do SOS:D², nesta etapa, o conjunto de dados brutos é pré-processado. As instâncias desse conjunto de dados representam, por intermédio de diversos atributos – propriedades –, o comportamento da corrente elétrica (assinatura) de acionamento das *máquinas de chave (MCHs)* durante a sua movimentação. Essas instâncias foram extraídas, pelo operador da ferrovia, pela amostragem da assinatura dessas MCHs durante a configuração dos AMVs.

Com este propósito, removem-se os dados com erros de leitura e/ou amostragem do conjunto de dados original e realiza-se a normalização desses dados, baseando-se na informação de quantis. Desse modo, cada atributo do conjunto de dados é transformado de modo a seguir uma distribuição normal. Assim, a normalização dos dados objetiva, para cada atributo, transformar os dados de modo a espalhar os valores mais frequentes e reduzir o impacto dos valores discrepantes (*outliers*).

Além disso, baseando-se nos atributos mais relevantes do conjunto de dados original – obtidos por meio de um processo físico de leitura da assinatura da corrente elétrica das MCHs –, realiza-se um processo de geração de novos atributos sobre os dados brutos. Neste cenário, a relevância de um atributo pode ser definida de duas formas: (i.) investigação manual, executada por meio da análise dos atributos da base de dados em conjunto com o especialista de domínio, e (ii.) semiautomática, efetuada por intermédio de técnicas de aprendizado de máquina que, após sua execução, reportam a relevância de cada atributo para a tarefa executada. Por exemplo, neste caso o classificador *Random Forest* foi utilizado – por causa da sua ampla utilização para o cálculo da importância dos atributos de conjuntos de dados de alta dimensão (MENZE et al., 2009).

Ao final dessas atividades, obtém-se um novo conjunto de dados com (atributos) dimensão ampliada, sem erros de leitura e/ou amostragem e com uma maior representatividade do domínio de aplicação. Finalmente, divide-se esse

novo conjunto de dados em dois subconjuntos: *subconjunto de dados não rotulado*, o qual é composto pelos dados que não possuem nenhuma informação sobre o estado de operacionalidade do equipamento, e *subconjunto de dados rotulados (conjunto de dados independente)*, o qual foi manualmente rotulado pelo operador ferroviário (utilizado no processo de geração de atributos).

Dado que o conjunto de dados original possui n atributos. Ambos os subconjuntos, após a atividade de geração de atributos, possuem $n + m$ atributos. Ou seja, m refere-se aos atributos que foram criados, por meio da combinação dos atributos mais importantes – realizada manualmente –, obtidos no processo de geração de atributos. Desta forma, esse novo conjunto de dados, composto pelos subconjuntos descritos, pode ser utilizado nas etapas seguintes.

2. Desenvolvimento dos modelos de Detecção e Diagnóstico de Falhas utilizando técnicas de Aprendizado de Máquina

Essa etapa é essencial para a análise do desempenho dos modelos FDDs. Neste contexto, um método é definido por uma combinação de técnicas que possibilitam a execução de uma tarefa de detecção e diagnóstico do estado de degradação do equipamento de forma não supervisionada. Cada método é aplicado sobre o conjunto de dados obtido na etapa anterior conforme o ilustrado na *Figura 14*. A forma de aplicação é igual para todos os métodos utilizados, isto é, a estrutura de aplicação é a mesma, contudo mudam-se as técnicas usadas na composição de cada método.

Vale ressaltar que, dado o esquema apresentado na *Figura 14*, a forma como os métodos (algoritmos) serão aplicados influencia diretamente no desempenho (performance) desses algoritmos na tarefa de identificação e definição do estado de degradação do equipamento. Em virtude disso, a etapa de desenvolvimento dos modelos é fundamental.

Portanto, o desempenho de um modelo FDD é afetado diretamente pela combinação das técnicas (método) de aprendizado de máquina aplicadas em seu desenvolvimento. No contexto do SOS:D², um modelo FDD (principal resultado desta etapa) pode ser entendido como o resultado da execução de um método

sobre o conjunto de dados. Logo, gera-se um modelo FDD para cada método proposto.

Os métodos aplicados neste trabalho científico são descritos a seguir. De modo geral, cada método é composto pela combinação de duas técnicas de aprendizado não supervisionado. As técnicas são combinadas de modo sequencial (uma após a outra), sendo a primeira técnica aplicada na detecção do estado de operacionalidade e, a segunda, no diagnóstico do estado de degradação (operacionalidade) do equipamento.

No contexto do SOS:D², o Autoencoder obtém os melhores resultados como a técnica não supervisionada aplicada na detecção de disfunções (anomalias) e falhas. Isto posto, as combinações das técnicas não supervisionadas (métodos) propostas objetivam unir essa técnica com outras técnicas amplamente utilizadas como o *K-means* e SOM (JIN et al., 2015; AMRUTHNATH; GUPTA, 2018; BIAN et al., 2019). Inclusive, uma das combinações engloba o caso em que o Autoencoder não é utilizado no processo de detecção de falhas.

Método 1 (*K-means* & SOM): Este método constitui-se da combinação dos algoritmos comumente aplicados em tarefas de agrupamento de dados. Neste caso, o algoritmo *K-means* é utilizado no processo de detecção do estado de operacionalidade do ativo, isto é, se o ativo se encontra em estado operacional normal ou indisponível (disfuncional). Posteriormente, aplica-se a rede *SOM*, sobre o resultado da etapa de detecção, para a identificação dos estados intermediários (degradados), referente aos prováveis tipos (modos) de falha.

Método 2 (Autoencoder & *K-means*): Neste método, combina-se a rede Autoencoder, uma das técnicas mais utilizadas atualmente devido a sua eficiência na detecção de falhas (GUO et al., 2018), e o algoritmo *K-means*. Ou seja, o método é composto pela combinação de uma técnica atual e eficiente e uma técnica tradicional.

Método 3 (Autoencoder & SOM): Semelhante ao método anterior. Porém, o processo de diagnóstico do estado de degradação utiliza a rede

SOM ao invés do algoritmo *K*-means. Neste caso, tanto o *Método 1* quanto este estão sujeitos a alterações nos hiperparâmetros⁴ da rede de forma a aprimorar os resultados obtidos pela rede SOM, sem enviesar os experimentos.

3. Avaliação dos modelos de Detecção e Diagnóstico de Falhas

Após o desenvolvimento dos modelos FDDs (etapa anterior), é possível inferir o estado de operacionalidade do equipamento. Nesta etapa, cada modelo FDD implementado na etapa anterior, será aplicado sobre o conjunto de dados independente (manualmente rotulado pelo especialista de domínio, especificado na *Figura 14* como “Dados manualmente rotulados”). Logo, ao final da execução de cada modelo, tem-se tanto um conjunto de prováveis causas (falhas) que caracterizam o comportamento abaixo do esperado desse equipamento quanto as informações (métricas de desempenho) referentes ao desempenho do método (algoritmo) utilizado no desenvolvimento do modelo. Desta forma, é possível realizar uma análise de causa e consequência dos principais modos de falha.

Além disso, a partir do desempenho dos algoritmos (métodos) – utilizados no desenvolvimento do modelo FDD –, de acordo com um conjunto de métricas de desempenho, cria-se um *ranking*. Assim, o desempenho (performance) de cada algoritmo será avaliado tanto na tarefa de detecção quanto no diagnóstico de falhas do equipamento. Cabe ressaltar que é viável ponderar essas tarefas de acordo com a sua criticidade/dificuldade. Neste contexto, tem-se métricas gerais calculadas a partir das métricas parciais de cada etapa, sendo que o desempenho do modelo de detecção e diagnóstico de falhas é definido pelas métricas gerais.

Por exemplo, pode acontecer a seguinte situação: o método (algoritmo) A obteve o melhor desempenho na tarefa de detecção de falhas, mas não manteve o mesmo desempenho no diagnóstico de falhas. Neste cenário, também é

⁴ Entende-se como hiperparâmetros os valores fornecidos como entrada para os algoritmos. De modo geral, esses valores são definidos previamente ao treinamento desses algoritmos.

possível que o algoritmo A seja melhor no diagnóstico e não na detecção de falhas. Logo, pode-se identificar o melhor algoritmo para cada etapa.

Finalmente, propõe-se que o desempenho de cada método no processo de detecção e diagnóstico de falhas seja calculado da seguinte forma:

$$m = p_1 * m_{Det} + p_2 * m_{Diag} ,$$

em que m (denominada métrica geral) equivale a cada uma das métricas analisadas, p_1 e p_2 referem-se, respectivamente, aos pesos dados às tarefas (detecção ou diagnóstico) cuja soma é igual a 1, m_{Det} refere-se a métrica de desempenho do algoritmo na etapa de detecção e m_{Diag} refere-se a métrica de desempenho do algoritmo na etapa de diagnóstico.

4.1.1 Desenvolvimento dos modelos de Detecção e Diagnóstico de Falhas utilizando técnicas de Aprendizado de Máquina

Nesta subseção, descreve-se o procedimento utilizado para o desenvolvimento dos modelos de detecção e diagnóstico de falhas, representados pela combinação de técnicas (métodos) de aprendizado de Máquina. A metodologia apresentada na *Figura 15*, baseada na estrutura utilizada em (DATTA; MAVROIDIS; HOSEK, 2008), especifica os passos necessários para o desenvolvimento dos modelos de Detecção e Diagnóstico ilustrados na etapa 2 da *Figura 14*. Como ilustrado na *Figura 15*, a sistemática utilizada consiste em seis etapas. A seguir, descreve-se cada uma dessas etapas.

Figura 15. Sistemática das etapas necessárias para o desenvolvimento dos modelos FDDs (Fonte: Autor).

1. *Aplicação de Técnicas de aprendizado não supervisionado para a detecção de falhas*

Nesta etapa, o subconjunto de dados não rotulado, resultante do pré-processamento dos dados (etapa 1 do processo ilustrado na *Figura 14*), e um número finito de grupos (*clusters*), requerido pelos algoritmos utilizados e representado comumente por k , é dado como entrada para cada uma das técnicas de aprendizado de máquina utilizadas. Essas técnicas aprendem de forma não supervisionada e, conseqüentemente, são comumente chamadas de *técnicas de aprendizado não supervisionado*.

Em Mineração de Dados, define-se como uma anomalia (*outlier*) uma instância de dados que apresenta características anômalas, ou discrepantes, em relação às demais instâncias de dados (CHANDOLA; BANERJEE; KUMAR, 2009; AGRAWAL; AGRAWAL, 2015; ABGHARI et al., 2019). De modo geral, a detecção de anomalias é a atividade de Mineração de Dados, utilizada principalmente em problemas de classificação ou de agrupamento de dados, comumente aplicada ao realizar-se uma análise exploratória dos dados.

Uma tarefa de detecção de anomalias tem como objetivo descobrir padrões anômalos no conjunto de dados (AGRAWAL; AGRAWAL, 2015). No contexto do SOS:D², uma anomalia representa a ocorrência de uma disfunção do sistema. Isto é, um comportamento

incomum desse sistema, no qual o serviço prestado é interrompido, que ocorre de forma abrupta (sem indicação prévia).

Isto posto, a detecção de falhas é fundamental em um programa de manutenção preditiva. Além disso, pode ser definida como “um processo de identificação do comportamento anormal de um componente ou subsistema de um sistema”, ou seja, é o processo de detectar a ocorrência de falhas incipientes no sistema (BOZZANO; VILLAFIORITA, 2010; AMRUTHNATH; GUPTA, 2018), sendo que essa anormalidade de comportamento não afeta a capacidade de operar do equipamento, diferentemente de uma disfunção.

Por conta disso, além do fato de o foco desta etapa ser a detecção de falhas, as instâncias de dados que representam disfunções (anomalias) são removidas previamente à detecção não supervisionada de falhas. Para isso, inicialmente, executa-se uma atividade de detecção de anomalias. Em seguida, realiza-se a detecção de falhas. Essas tarefas são realizadas nesta etapa.

Portanto, o objetivo desta etapa, além de detectar a ocorrência de uma disfunção, é a detecção da presença de uma falha incipiente no sistema de uma forma não supervisionada, visto que não existe o conhecimento prévio sobre o comportamento dos dados de um equipamento falho. Em razão disso, a detecção do estado de operacionalidade do sistema é realizada em camadas, como ilustrado na *Figura 16*.

Figura 16. Detecção em camadas do estado de operacionalidade do sistema (**Fonte:** Autor).

Vale ressaltar que, em cada método, apenas a técnica não supervisionada definida para a detecção do estado de operacionalidade do equipamento é utilizada no processo apresentado na *Figura 16*. Todavia, criam-se modelos específicos para cada camada, incluindo o número ótimo de clusters (grupos). O número ótimo de clusters, k , é definido como o valor que maximiza o coeficiente de *Largura de Silhueta* – LS (critério de validação relativa de *clusters*), o qual baseia-se nos conceitos de compactação *intra-cluster* e separação *inter-cluster* – veja a

Figura 13 – (TAN; STEINBACH; KUMAR, 2013). Basicamente, para se calcular o coeficiente de silhueta executou-se o algoritmo *K*-means sobre o conjunto de dados não rotulado $k-2$ vezes (considerou-se 2 como o número mínimo de clusters), ou seja, a quantidade de possíveis *clusters* varia de 2 até k .

Na primeira camada executa-se uma tarefa de *Deteção de Anomalias*. Ou seja, a técnica não supervisionada é aplicada sobre o novo conjunto de dados visando a detecção de instâncias anômalas. Como resultado, tem-se um conjunto de *clusters* que representam dados normais (incluindo instâncias falhas) e anômalos (disfunções) (veja a *Figura 17* – *Deteção de Anomalias*). Assume-se que as instâncias de dados normais tendem a ficar próximas aos centroides dos *clusters* considerados normais pelo algoritmo, assim como as instâncias de dados anormais tendem a ficar distante desses centroides e, conseqüentemente, próximas aos centroides dos *clusters* considerados anômalos (CHANDOLA; BANERJEE; KUMAR, 2009).

Vale ressaltar que essa suposição é geral para as técnicas utilizadas, mas particular dessa tarefa. No entanto, para cada técnica, mantém-se as suas características e os parâmetros individuais requeridos. Em vista disso, os *clusters* resultantes da tarefa de detecção de anomalias são separados em normal ou anomalia com base na seguinte afirmação:

“Instâncias de dados normais pertencem a clusters grandes e densos, enquanto anomalias pertencem a clusters pequenos ou esparsos. ” (CHANDOLA; BANERJEE; KUMAR, 2009)

Neste contexto, a primeira camada funciona como um filtro de anomalias, o que possibilita a segunda camada lidar diretamente com identificação das falhas incipientes. Devido a isso, na tarefa de *Deteção de Falhas* (segunda camada), a técnica não supervisionada é utilizada para identificar as instâncias de dados que, provavelmente, representam falhas incipientes. Isto é, instâncias de dados que foram pseudo-rotuladas como normais no processo de detecção de anomalias.

Figura 17. Representação dos *clusters* obtidos com aplicação da técnica não supervisionada no processo de detecção em camadas do estado de operacionalidade do sistema (**Fonte: Autor**).

A Figura 17 mostra, em alto nível, uma distribuição ilustrativa dos *clusters* que podem ser obtidos por meio do processo apresentado na Figura 16. Como pode ser observado nesta última figura, no processo de identificação das falhas incipientes (segunda camada), diferentemente da detecção de anomalias (primeira camada), os dados possuem um maior grau de similaridade. Isto ocorre pelo fato de o comportamento da falha incipiente ser diferente de uma disfunção.

Como o nível de criticidade – deterioração do equipamento – da falha incipiente vai se agravando ao longo do tempo, é difícil visualizar/identificar a diferença entre os comportamentos de dados normais e falhos no início do acontecimento da falha. Por outro lado, quanto mais próximo de uma disfunção o comportamento estiver, mais fácil é a sua visualização/identificação e, portanto, maior o seu nível de degradação (criticidade).

Por essa razão, assim como na primeira camada, na segunda camada do processo ilustrado pela Figura 16, realiza-se a definição do número ótimo de *clusters*, k , também de acordo com o coeficiente de *Largura de Silhueta* – LS. Posteriormente, o número ótimo de *clusters* e o subconjunto de dados composto apenas pelas instâncias consideradas como operacionais, isto é, tanto dados normais (boas condições operacionais) quanto dados falhos (condições operacionais degradadas), são dados como entrada para a mesma técnica não supervisionada aplicada na detecção de anomalias.

É importante salientar que os *clusters* resultantes da aplicação da técnica são categorizados, em normal ou falha, por meio das seguintes considerações: (1.) Assume-se que existem mais instâncias de dados normais porque, a princípio, falhas são eventos de baixa frequência; (2.) Baseando-se nessa afirmação, considera-se que *clusters* grandes representam dados normais e, conseqüentemente, *clusters* pequenos representam dados falhos (falhas incipientes).

Por fim, após a execução das tarefas mencionadas, tem-se um conjunto de *clusters* constituído por: (i.) *clusters* anômalos, resultantes da detecção de anomalias e representam disfunções; (ii.) *clusters* falhos, resultantes da detecção de falhas e representam as falhas incipientes e; (iii.) *clusters* normais, também resultantes da detecção de falhas, mas representam o comportamento normal do equipamento (veja a *Figura 17*).

Até aqui, a interpretação dos *clusters* é realizada baseando-se apenas na densidade e tamanhos dos *clusters*. No entanto, a interpretação dos *clusters* é uma etapa essencial em uma tarefa de agrupamento e, por isso, é interessante empenhar-se na mesma. Inclusive, nessa etapa, a opinião do especialista de domínio é fundamental para a validação do agrupamento obtido por meio da aplicação da técnica não supervisionada (MARTINS, 2003). Geralmente, a participação do especialista de domínio nessa etapa pode aprimorar os resultados.

Devido a isso, na próxima etapa é realizado um processo de interpretação semiautomática de *clusters*. Em função disso, nesta etapa, é gerado o conjunto de dados agrupados (*clusters*) que, na próxima etapa, será utilizado para gerar o conjunto de dados de treinamento, por meio desse processo de interpretação. Para isto, na etapa 2, utiliza-se um algoritmo semiautomático de interpretação de *clusters*.

Esse algoritmo foi desenvolvido com base no conhecimento do especialista, representado pelo conjunto de dados independente (manualmente rotulado). Após a aplicação do algoritmo, a organização dos elementos de um *clusters* pode ser alterada, conseqüentemente, o pseudo-rótulo de uma instância de dados do conjunto de dados de treinamento pode ser modificada.

2. Interpretação semiautomática dos clusters

Como resultado da etapa anterior, obtém-se um conjunto de *clusters* a partir de uma base de dados não rotulada. Esses *clusters* representam os possíveis estados operacionais do

equipamento, ou seja, estado operacional (normal), estado operacional com degradação (falha) e indisponível (disfunção). Sabendo-se disso, essa etapa objetiva interpretar cada *cluster* gerado na etapa anterior e definir, com base no conhecimento do especialista do domínio de aplicação, o comportamento representado por um *cluster*. Desta forma, o comportamento descoberto em conjunto com o especialista de domínio e o definido para o *cluster* são analisados. Por fim, caso seja necessário, pode-se alterar tanto a classe quanto o pseudo-rótulo definido para um *cluster*.

A interpretação dos *clusters* em uma tarefa de agrupamento de dados é comumente realizada de forma manual pelo usuário em conjunto com o especialista do domínio. Dada as suas características, existe uma alta subjetividade no agrupamento dos dados. Visto que, normalmente, o conjunto de dados é agrupado de acordo com alguma medida de similaridade. Desta forma, o mesmo conjunto de dados pode ser agrupado de diversas formas distintas, onde a aplicação do mesmo algoritmo pode resultar em grupos diferentes (MARTINS, 2003).

Por conseguinte, não existe um algoritmo capaz de solucionar todo e qualquer problema relacionado ao agrupamento de dados. Por conta disso, diz-se que o processo de agrupamento de dados é uma tarefa difícil e para obter o algoritmo adequado para o problema estudado é necessário verificar as características de cada algoritmo e o domínio de aplicação do problema. Por outro lado, por meio do agrupamento conceitual, os *clusters* podem representar conceitos, uma vez que o agrupamento conceitual, além de gerar os *clusters*, produz descrições que caracterizam os clusters (MARTINS, 2003).

Apesar do agrupamento conceitual aprimorar a representatividade dos *clusters*, o processo de descrição conceitual não é uma tarefa simples. Inclusive, pode ser mais difícil do que a geração dos *clusters*. Obter as descrições das categorias, por meio de indução sobre os dados, representadas pelos *clusters* é uma tarefa inferencial complexa (MARTINS, 2003).

Além dos argumentos apresentados, em uma aplicação sobre um domínio do mundo real, a interpretação dos *clusters* não pode depender, em excesso, do especialista de domínio, considerando que dificilmente é possível obter informações em um curto período de tempo. Devido a todos esses fatores, neste trabalho utiliza-se uma sistemática que usa o subconjunto de dados manualmente rotulado pelo especialista de domínio – também chamado de conjunto de dados independente ou de referência (*Ground-Truth*) – para tornar o processo de interpretação dos clusters semiautomático.

A *Figura 18* apresenta a sistemática utilizada para a interpretação semiautomática de *clusters*. A entrada do processo é o conjunto de *clusters* resultantes da etapa anterior. Basicamente, a sistemática é composta por duas etapas, são elas, *Algoritmo de Interpretação de Clusters* e *Interpretação Conceitual dos Clusters*:

- **Algoritmo de Interpretação de *Clusters*:** O algoritmo desenvolvido, o qual tem como objetivo realizar o papel do especialista de domínio, é aplicado sobre os *clusters*. Esse algoritmo é baseado no conhecimento do domínio do especialista. Assim, visa-se obter, por meio da análise da base de dados manualmente rotulada, um conjunto de regras de domínio que possam auxiliar na descrição (categorização) dos *clusters*. Ou seja, dado um conjunto de classes definidas pelo especialista, o algoritmo tenta identificar características individuais dessas classes. Posteriormente, utiliza-se dessas características para definir a categoria (classe) representada por cada *cluster*.
- **Interpretação Conceitual dos *Clusters*:** Após a aplicação do algoritmo, dependendo dos grupos “finais” encontrados, o especialista do domínio pode decidir repetir (ou não) todo o processo de agrupamento de dados ou apenas uma etapa específica, podendo:
 - Alternar a técnica de aprendizado não supervisionado utilizada ou apenas variar os parâmetros de entrada do modelo atual, tal como, a medida de similaridade;
 - Unir em um único grupo, dois ou mais grupos distintos cujo algoritmo identificou que possuem comportamentos idênticos e/ou similares; e
 - Realocar instâncias alocadas indevidamente a um grupo.

Observe que, devido à forma como o algoritmo aprende as características individuais e, em seguida, as transforma em regras de domínio, o resultado dessa etapa pode ser afetado pela presença de inconsistências na base de dados que representa o conhecimento do especialista. Neste contexto, uma inconsistência ocorre quando duas ou mais instâncias de dados possuem comportamentos idênticos, mas classes distintas.

É importante notar que o usuário participa, praticamente, de todo o processo de agrupamento de dados. De fato, apesar da autonomia do algoritmo em interpretar os *clusters*, o usuário deve analisar os resultados obtidos com a finalidade de verificar se o agrupamento está de acordo com os seus objetivos. Além disso, deve-se verificar se os grupos apresentam uma descrição conceitual que possa ser associada a uma “classe”, visto que uma técnica não supervisionada sempre encontra grupos, mesmo que eles não

tenham significado prático, isto é, não é possível relacionar os comportamentos identificados pela técnica (representados pelos *clusters*) com o domínio de aplicação.

Figura 18. Sistemática para a interpretação semiautomática de clusters (**Fonte:** Autor).

Após a explicação do processo de interpretação dos *clusters* (grupos) é importante explicar como essa metodologia é aplicada sobre os grupos oriundos da etapa anterior. Cabe salientar que o algoritmo de interpretação é implementado em módulos, isto é, existem módulos específicos para cada tipo de tarefa. Existe um módulo para a detecção de anomalias (disfunções), outro para falhas, e outro para os tipos de falhas (explicado na subseção referente a etapa de diagnóstico).

Sabendo-se disso, inicialmente, interpreta-se os grupos anômalos. Em seguida, os grupos falhos são interpretados. Por fim, os demais grupos, considerados normais, são interpretados. Como ilustrado na *Figura 19*, caso exista a necessidade de refinar apenas o modelo de detecção de falhas, somente essa tarefa é executada novamente. Contudo, caso o modelo de detecção de anomalias também precise ser refinado, todo o processo de detecção do estado de operacionalidade do equipamento deve ser executado novamente. Uma vez que o resultado da etapa de detecção de anomalias é relevante para a detecção de falhas.

Figura 19. Fluxo de avaliação da relevância dos resultados do processo de detecção do estado de operacionalidade do equipamento (**Fonte:** Autor).

No entanto, caso não seja necessário refinar o modelo (isto é, faz-se necessário apenas agrupar *clusters* ou realocar instâncias), modificam-se somente os pseudo-rótulos dos *clusters*, ou de instâncias de dados específicas, no conjunto de treinamento (gerado nesta etapa). Por conta disso, de modo geral, possíveis erros de agrupamento são removidos nesta etapa, onde uma instância alocada indevidamente a um grupo pode ser realocada a um novo grupo cujo comportamento, segundo as regras de domínio, possui um grau maior de similaridade com essa instância.

Como ilustrado na *Figura 15*, nesta etapa gera-se o conjunto de dados de treinamento que será utilizado na etapa de criação do classificador *ensemble* (próxima etapa). Neste caso, para cada técnica aplicada sobre o subconjunto de dados não rotulado (etapa anterior), gera-se um novo conjunto de dados que, conseqüentemente, será usado na criação de um novo classificador *ensemble*.

Com este propósito, para gerar o conjunto de dados de treinamento, cria-se um novo atributo. Esse atributo representa a classe – normal, falha ou disfunção (anomalia) – a que uma instância de dados pertence. Baseando-se nos *clusters* (gerados pela técnica usada em cada tarefa de agrupamento) resultantes do processo de interpretação, um pseudo-rótulo é dado para cada uma das instâncias de dados. Assim, é gerado um conjunto de dados com uma dimensão extra, em que o rótulo é definido como o atributo “classe” dos dados.

Isto posto, os dados do conjunto não rotulado são rotulados (pseudo-classificação) da seguinte forma: de acordo com o rótulo (pseudo-rótulo) do *cluster* determinando pelo algoritmo de interpretação de *clusters*, o rótulo de uma instância é definido como segue: (1.) para cada instância de dados que pertence a um grupo considerado normal, atribui-se o rótulo “Normal” (isto é, sem falha); (2.) atribui-se o rótulo “Falha” para as instâncias que pertencem a um grupo que representa as falhas incipientes; e, por fim, (3.) o rótulo “Anomalia” (ou seja, disfunção) é atribuído às instâncias que pertencem a um grupo considerado anômalo. Observe que o resultado deste procedimento de pseudo-classificação é a geração de dois conjuntos de dados de treinamento (“Conjunto de treinamento para a Detecção de Falhas” e “Instâncias de dados classificados como falhas”), objetivo principal desta etapa, a serem utilizados pelas etapas 3 e 4 do desenvolvimento dos modelos FDD.

3. Criação do classificador *ensemble* para a detecção de falhas

Dado a severidade do processo de tomada de decisão em um programa de manutenção preditiva, o desenvolvimento de modelos robustos e eficientes é um desafio na detecção de falhas. Principalmente quando esses algoritmos devem ser utilizados para auxiliar nas decisões de manutenção (AMRUTHNATH; GUPTA, 2018). Isto posto, tendo como objetivo a detecção de falhas de modo supervisionado – em que utiliza-se pelo menos um classificador –, para aprimorar a eficiência e robustez de predição do classificador, o SOS:D² utiliza um conjunto de classificadores para a classificação de novas falhas incipientes.

O classificador *ensemble* gerado nesta etapa é um componente fundamental de um modelo FDD, pois, por intermédio do mesmo, pode-se aplicar o conhecimento extraído pela técnica não supervisionada. Note que, dada as suas características, faz-se necessário utilizar técnicas que aprendem de forma supervisionada. Levando em consideração a eficiência e robustez do classificador, cinco modelos de classificação são utilizados como modelos-base para a criação de um classificador *ensemble*: a rede neural artificial *Perceptron* Multicamadas (MLP), *Random Forest*, *AdaBoost*, *XGBoost* e SVM⁵.

Cada modelo de classificação é criado com base na melhor combinação possível para a tarefa de detecção de falhas. Essa combinação é definida previamente por meio de um procedimento de otimização dos modelos de classificação utilizados, e consequentemente de seus hiperparâmetros, sobre o conjunto de dados independente (manualmente rotulado pelo especialista de domínio)⁶. Após esse processo, cria-se um classificador *ensemble* – objetivo principal desta etapa – por meio da combinação dos modelos otimizados.

Neste caso, os classificadores-base otimizados são combinados por meio de voto majoritário para formar o classificador *ensemble*. Os classificadores-base foram escolhidos com base tanto em diversos cenários de teste quando na diversidade de cada modelo, além da sua ampla utilização em problemas de classificação que objetivam a detecção e diagnóstico de falhas. De fato, uma forma de aprimorar a eficiência de predição de um classificador *ensemble* é aumentar a diversidade entre seus classificadores-base, uma vez que isso proporciona classificações não correlacionadas (TIAN et al., 2017).

⁵ Os classificadores *Random Forest*, *AdaBoost*, *XGBoost* e SVM são descritos no Apêndice B. Esses classificadores tradicionais são amplamente aplicados em tarefas de detecção de falhas e/ou anomalias, como pode ser observado na subseção de trabalhos correlatos.

⁶ Veja o Apêndice A para mais detalhes sobre o processo de otimização dos modelos de classificação.

O classificador *ensemble* criado determina a classe de uma nova instância de dados baseando-se na classe predita por cada um dos seus classificadores-base. Essa análise individual, ilustrado na *Figura 20*, é efetuada pelo algoritmo de votação (votador), o qual é responsável por definir a classe final da instância. O votador, individualmente, analisa a classe predita (voto) por cada classificador-base e, em seguida, a classe final é definida por meio de voto majoritário (classe mais predita). Por causa disso, normalmente, um classificador *ensemble* com voto majoritário possui um número ímpar de classificadores-base.

Figura 20. Estrutura do classificador *ensemble* (Fonte: Autor).

O modelo baseado na arquitetura da rede neural artificial *perceptron* multicamadas (do inglês, *Multilayer Perceptron* – MLP), apresentado na *Figura 21*, é composto por duas camadas ocultas, as quais possuem 256 nós (neurônios) e a função *ReLU*⁷ como função de ativação, a camada de entrada dos dados, em que $x_1, x_2, x_3, \dots, x_n$ representam as n instâncias de dados, e a camada de saída. A camada de saída é composta por um único neurônio, visto que este modelo objetiva uma classificação binária. Esse neurônio tem a função *Sigmoid*⁸, como função de ativação. Por fim, foi observado que essa configuração aprimora a capacidade da rede MLP de identificar anomalias e/ou falhas (NWANKPA et al., 2018).

Um problema sério, porém comum, que afeta diretamente o desempenho dos classificadores é o sobreajuste (do inglês, *Overfitting*) (NITISH et al., 2014). Em redes neurais artificiais, diz-se que um classificador (um modelo treinado que aprende uma generalização sobre um conjunto de dados a partir de exemplos (RASCHKA, 2015)) está com sobreajuste

⁷ O Apêndice B apresenta tanto o comportamento quanto uma descrição dessas funções de ativação.

⁸ O Apêndice B apresenta tanto o comportamento quanto uma descrição dessas funções de ativação.

quando tenta excessivamente se ajustar aos dados de treinamento. Desta forma, aprende diversas características individuais desses dados ao invés de encontrar uma boa generalização, muitas vezes chamada de regra preditiva geral, desses dados (DIETTERICH, 1995).

Figura 21. Estrutura do modelo de classificação MLP (**Fonte:** Autor).

Devido a isso é possível que o modelo obtenha um bom desempenho na etapa de treinamento, mas seja ineficaz na classificação de novos dados. Em outras palavras, a rede “memoriza” bem os dados de treinamento, no entanto, não é assegurado que essa “memorização” funcione em dados não vistos, assim existe uma diferença entre a precisão durante o treinamento e durante a avaliação do desempenho (XIE et al., 2017).

Deste modo, para evitar que o modelo fique com sobreajuste, faz-se necessário a implementação de uma outra camada, chamada de “corte” (do inglês, *dropout*), que desativa aleatoriamente uma fração de neurônios durante o processo de treinamento. Assim, o termo *dropout* “refere-se à remoção de unidades (ocultas e visível) em uma rede neural” (NITISH et al., 2014).

Neste contexto, uma camada de “corte” é responsável por, a cada iteração de treinamento (época), eliminar uma porcentagem específica da generalização aprendida pelo modelo. Logo, a generalização obtida na etapa de treinamento torna-se mais realista, aumentando a probabilidade de ser aplicável sobre o conjunto de dados de teste. Por isso, durante a implementação do modelo MLP proposto, uma camada de “corte” foi adicionada.

Sabendo-se disso, os modelos de classificação são treinados com o conjunto de dados de treinamento gerado na etapa anterior (um para cada técnica). Consequentemente, a partir do processo de treinamento do modelo, obtém-se um classificador-base. Ao final desta etapa, para

cada modelo FDD, tem-se cinco classificadores-base, os quais foram treinados com o conjunto de dados de treinamento gerado pela técnica não supervisionada usada no modelo FDD. O classificador *ensemble* formado por esses classificadores é utilizado na próxima etapa para a classificação de novas instâncias de dados. Vale ressaltar que o processo de treinamento do modelo é geral. Assim, apenas mudam-se os dados de treinamento.

No processo de treinamento dos modelos utiliza-se a validação cruzada (KOHAVI, 1995). Primeiramente, divide-se aleatoriamente o conjunto de dados de treinamento, gerando-se subconjunto de dados (também chamados de grupos). Neste contexto, baseando-se na métrica de precisão do classificador *ensemble*, identifica-se o melhor grupo de dados de treinamento. Posteriormente, treina-se o classificador com esse grupo de dados e realiza-se a classificação de novas instâncias de dados (KOHAVI, 1995).

Vale ressaltar que a porcentagem dos dados de cada classe, em relação a todo o conjunto de treinamento, é mantida em cada um dos grupos. Desta forma, no treinamento do classificador *ensemble*, utilizam-se todos os grupos. Inicialmente, de modo aleatório, escolhe-se o primeiro grupo. Desse grupo retira-se uma parte para realizar a validação e, conseqüentemente, para calcular a precisão do classificador. Finalmente, cada grupo é usado apenas uma vez, como conjunto de treinamento e validação, durante o treinamento do modelo. Além disso, utiliza-se somente um grupo por época (iteração de treinamento) (KOHAVI, 1995).

Todo esse processo de criação de um classificador *ensemble* é realizado conforme o tipo de tarefa de classificação a ser realizada. Por exemplo, caso seja uma tarefa de classificação de anomalias, todo esse processo seria efetuado com foco na identificação de instâncias anômalas. Ou seja, os modelos de classificação seriam otimizados visando esse tipo de tarefa. Isto posto, é importante mencionar que o classificador gerado nessa etapa é composto por dois classificadores *ensemble*.

Desta forma, primeiramente realiza-se o processo de treinamento de um classificador *ensemble* com foco na detecção de anomalias. Posteriormente, o mesmo processo, porém com modelos de classificação diferentes – por conta da mudança de objetivo –, é aplicado com foco na detecção de falhas. Portanto, ao final desta etapa, tem-se um classificador geral composto por dois classificadores *ensemble* que podem ser utilizados em processo de detecção de falhas em camadas.

4. *Aplicação de Técnicas de aprendizado não supervisionado para o diagnóstico do estado de degradação do ativo*

Esta etapa é semelhante à etapa 1. Porém, o número ótimo de *clusters* é definido em conjunto com o especialista de domínio e não apenas pelo método *Largura de Silhueta - LS*. Desta forma, assume-se que cada *cluster* corresponde a um tipo específico de comportamento do equipamento que, conseqüentemente, reflete o seu grau de operacionalidade. Além disso, objetiva-se que os dados com um alto grau de similaridade estejam no mesmo *cluster*.

Encontrar o número ótimo de grupos (*clusters*), k , para se particionar o conjunto de dados é uma tarefa factível, sendo que, com a identificação do número ideal de *clusters*, é possível particionar o conjunto de dados de modo a representar corretamente esses dados e compreender os diversos estados existentes. Perceba que se tem somente dados considerados como falhas (veja a *Figura 15*). Assim, nesta etapa, torna-se possível entender os estados intermediários do conjunto de dados (AMRUTHNATH; GUPTA, 2018).

Existem diversas técnicas que podem ser aplicadas sobre o conjunto de dados ou, geralmente, sobre uma amostra desses dados, tais como o método de Elbow, Critério de Inferência Bayesiana, Escala multidimensional e técnicas de validação de *clusters* (como o *Largura de Silhueta*) (DATTA; MAVROIDIS; HOSEK, 2008; AMRUTHNATH; GUPTA, 2018). Nesta etapa, utiliza-se o mesmo método utilizado na etapa 1, porém dá-se um peso maior para a opinião especialista de domínio, uma vez que isso pode direcionar o agrupamento dos dados e reduzir a subjetividade do processo.

Nesta etapa, posteriormente à definição do número ótimo de *clusters*, a técnica não supervisionada utilizada para o diagnóstico do estado de degradação do equipamento é aplicada sobre o subconjunto de dados de instâncias falhas, obtidos na etapa 2. Como pode ser observado na *Figura 15*, após a criação do conjunto de dados de treinamento e da etapa de interpretação de *clusters* para a detecção de falhas, as instâncias de dados pseudo-rótuladas como “Falha” são passadas para esta etapa com a finalidade de serem usadas no processo de geração do conjunto de dados de treinamento para o diagnóstico dessas falhas.

O conjunto de dados de treinamento para o diagnóstico de falhas é criado da seguinte forma: Inicialmente, adiciona-se um novo atributo ao conjunto de dados, referente ao modo de falha das instâncias de dados (“classe”). O pseudo-rótulo de cada modo de falha é definido

conforme o valor de k . Suponha que o número ótimo de *clusters* seja dez ($k=10$). Neste caso, tem-se dez modos de falha. Logo, as “classes” 1,2,3, ..., 10 são atribuídas às instâncias de dados de acordo com o *clusters* em que elas foram alocadas pela técnica não supervisionada.

Observe que nessa etapa tem-se um conjunto de *clusters*, ou seja, prováveis “classes” que representam o comportamento de cada modo de falha. Contudo, sem o processo de interpretação de *clusters*, não é possível inferir o tipo de falha que cada *cluster* representa. Isto posto, na próxima etapa, por intermédio do processo de interpretação dos *clusters*, o mapeamento entre os modo de falhas (tipos) existentes no conjunto de dados original e os *clusters* obtidos é realizado de acordo com o conhecimento do especialista de domínio. Cabe ressaltar que, assim como na etapa anterior, a distribuição dos *clusters* e o conjunto de dados de treinamento criado nesta etapa podem ser modificados (caso seja viável).

5. *Interpretação semiautomática dos clusters*

Assim como na etapa 2, nesta etapa aplica-se o algoritmo de interpretação dos *clusters* sobre o resultado da etapa anterior. Neste caso, utiliza-se especificamente o módulo referente ao processo de diagnóstico do estado de operacionalidade do equipamento. O processo de interpretação dos *clusters* é o mesmo da etapa 2, ilustrado na *Figura 18*. Porém o algoritmo de interpretação ao invés de identificar características referentes ao estado operacional do equipamento, trata do estado de desgaste do equipamento.

Desta forma, realiza um mapeamento entre os diversos tipos de falhas (especificados pelo especialista do domínio) e os *clusters* encontrados pelas técnicas de aprendizado não supervisionado. Essencialmente, define-se um conjunto de regras do domínio de aplicação que representam cada modo de falha. Sendo que, em alguns casos, um único modo de falha é representado por mais de uma regra.

É importante notar que, assim como na etapa 2, também é possível realizar alterações no processo de agrupamento dos dados. Consequentemente, pode-se modificar os conjuntos de dados de treinamento provenientes de cada uma das técnicas de aprendizado não supervisionado aplicadas. Vale salientar que esses conjuntos de dados de treinamento possuem instâncias de dados de todas as possíveis causas de degradação do equipamento.

Os modos de falhas são definidos por meio do processo de interpretação dos *clusters*. Logo, é possível ampliar a dimensão do conjunto de dados de modo que o atributo referente ao

rótulo da “classe” seja definido pelo modo de falha representado pelo *cluster*. Em virtude das características da metodologia abordada, nesta etapa, tanto pode-se utilizar as mesmas técnicas do processo de detecção quanto uma nova técnica. Consequentemente, é possível realizar uma combinação entre técnicas de aprendizado não supervisionado.

6. Criação de um classificador ensemble de acordo com os estados intermediários

Esta etapa também utiliza as técnicas de classificação descritos na etapa 3. De fato, essa etapa tem o objetivo de criar um classificador *ensemble*, mas específico para o diagnóstico do estado de operacionalidade do equipamento. Por esse motivo, os modelos implementados possuem na camada de saída uma quantidade de neurônios equivalente ao número de modos (tipos) de falha. Em outras palavras, existe uma relação de equivalência entre o número de tipos de falha e a quantidade de neurônios da camada de saída dos modelos. Por exemplo, no caso do modelo MLP, a camada de saída possui a função *Softmax*⁹ como função de ativação ao invés da *Sigmoid*, visto que essa função é geralmente utilizada em tarefas de classificação multiclasse (NWANKPA et al., 2018).

Vale ressaltar que a única diferença entre esta etapa e a etapa 3 é a alteração do tipo de classificação (mudança de binária para multiclasse). Portanto, também se utiliza tanto o processo de otimização¹⁰ dos modelos de classificação para a implementação dos classificadores base quanto a validação cruzada no treinamento do classificador *ensemble*. Da mesma forma, ao final desta etapa tem-se diversos classificadores *ensemble* (um para cada técnica aplicada), que serão utilizados posteriormente para a classificação de novas instâncias de dados. Portanto, transforma-se o diagnóstico de falhas do equipamento em uma tarefa de classificação.

4.1.2 Avaliação dos modelos de Detecção e Diagnóstico de Falhas

Como mostra a *Figura 14*, a avaliação do desempenho dos modelos de detecção e diagnóstico do estado de operacionalidade (degradação) de um equipamento é efetuada após o desenvolvimento desses modelos (descrito na subseção anterior). Com base na argumentação apresentada na subseção anterior, as tarefas de detecção e diagnóstico, nesse momento, podem

⁹ O Apêndice B apresenta tanto o comportamento quanto uma descrição dessas funções de ativação.

¹⁰ Veja o Apêndice A para mais detalhes sobre o processo de otimização dos modelos de classificação.

ser consideradas como tarefas de classificação. Por conta disso, o desempenho de cada modelo é analisado por meio de uma tarefa de classificação.

Com este propósito, o conjunto de dados independente, manualmente rotulado pelo especialista de domínio, é utilizado para analisar o desempenho do modelo FDD em cada tarefa de classificação. Esse conjunto de dados possui informações referentes tanto ao estado operacional do equipamento – *Normal* (operacional), *Falha* (operacional, mas com degradação) ou *Indisponível* (disfunção) – quanto ao estado de degradação do equipamento – modos de falha.

Figura 22. Processo de avaliação do desempenho dos modelos de detecção e diagnóstico do estado de operacionalidade de um equipamento (**Fonte:** Autor).

A Figura 22 apresenta as etapas do processo de avaliação do desempenho dos modelos proposto neste trabalho, descrito nesta subseção. Esse procedimento é composto por três etapas, que são: **(1.) Detecção de falhas por meio do classificador *ensemble***; **(2.) Realização do Diagnóstico do estado de degradação por meio do classificador *ensemble***; e **(3.) Análise de desempenho do modelo FDD**. Cada uma dessas etapas é descrita a seguir.

1. Detecção de falhas por meio do classificador *ensemble*

O conjunto de dados independente, manualmente rotulado pelo especialista de domínio, é utilizado para analisar o desempenho do classificador *ensemble*, obtido na etapa 3 da Figura 15 e descrito na subseção anterior, na tarefa de detecção de falhas.

Nesta etapa, utiliza-se a classificação como uma técnica de detecção de falhas. O conjunto de dados independente é utilizado para avaliar o desempenho do classificador *ensemble*, obtido na tarefa de criação do detector *ensemble* de falhas (etapa 3 da *Figura 15*) e que faz parte do modelo FDD avaliado, na identificação do estado de operacionalidade do equipamento (no contexto do SOS:D², do AMV). Devido a isso, as novas instâncias de dados são classificadas em normal (operacional, sem falha e/ou degradação), falha (operacional, mas com degradação) e disfunção (indisponível, anomalia).

Vale salientar que, previamente à tarefa de detecção de falhas (objetivo principal desta etapa), executa-se uma tarefa de detecção de anomalias (disfunções). Visto que, ao remover as anomalias do conjunto de dados, pode-se tratar diretamente da identificação de falhas. Inclusive, a equipe de manutenção pode ser acionada para solucionar o problema das instâncias de dados classificadas como anômalas. Isto é, como disfunções, pois o equipamento encontra-se indisponível.

Posteriormente, realiza-se uma tarefa de detecção de falhas sobre as instâncias de dados que representam um comportamento operacional do equipamento. Em virtude desse processo, tem-se duas tarefas de classificação binária que resultam, de certa forma, em uma classificação multiclasse. Observe que, apesar da importância da detecção das disfunções, o SOS:D² avalia o desempenho do classificador apenas na tarefa de detecção de falhas. Assim, a detecção de anomalias é uma etapa essencial, mas não é considerada na avaliação do desempenho.

Desta forma, após o treinamento do modelo (realizado na etapa 3 da *Figura 15*), o conjunto de dados manualmente rotulado é usado para avaliar o desempenho do classificador *ensemble*. Essa análise de desempenho é feita por meio de métricas, tais como a *acurácia*, *precisão* e sensibilidade (*recall*). Vale salientar que essas métricas são calculadas a partir da comparação, para cada instância classificada, entre o resultado obtido pelo classificador e o valor real do rótulo definido pelo especialista de domínio.

Após aplicar o classificador sobre o conjunto de dados e avaliar o seu desempenho (por meio das métricas), observa-se especificamente o rótulo definido por ele para cada instância de dados. Realiza-se essa análise para separar as instâncias de dados classificadas como falhas das consideradas normais. Em seguida, cria-se um novo conjunto de dados composto apenas pelas instâncias de dados falhas. Esse novo conjunto de dados é utilizado no processo de diagnóstico do estado de degradação do AMV (próxima etapa – etapa 2 da *Figura 22*).

2. Realização do Diagnóstico do estado de degradação por meio do classificador *ensemble*

Esta etapa é semelhante a anterior, isto é, utiliza-se o conjunto de dados independente, manualmente rotulado pelo especialista, para calcular as métricas especificadas anteriormente. Porém, não existe a execução de mais de uma tarefa de classificação. O objetivo desta etapa é especificar, tendo como premissa o fato de a instância ser uma falha incipiente, o que ocasionou esse desvio no comportamento do equipamento. Portanto, uma instância de dados será classificada em um dos modos de falhas definidos no treinamento do classificador *ensemble* (etapa 6 da *Figura 15*).

3. Análise de desempenho do Modelo FDD

Nesta etapa os desempenhos dos classificadores obtidos por meio da utilização das técnicas de aprendizado não supervisionado, nas etapas 3 e 6 do processo de desenvolvimento dos modelos de detecção e diagnóstico do estado de degradação do AMV (apresentado na subseção anterior) são analisados. Neste contexto, avalia-se o desempenho do detector *ensemble* na tarefa de detectar se o equipamento se encontra em um estado operacional com degradação ou operacional sem degradação. Como mencionado anteriormente, esse desempenho é determinado por intermédio das métricas calculadas em ambas tarefas de classificação do processo descrito nesta subseção.

Também se avalia os desempenhos dos classificadores na tarefa de diagnóstico do estado de operacionalidade do equipamento. Em outras palavras, analisa-se os desempenhos dos classificadores na identificação dos prováveis tipos de falhas que provocam o seu desgaste. Vale ressaltar que nesta etapa calcula-se as métricas gerais de acordo com o apresentado anteriormente na subseção referente ao passo 3 da *Figura 14*.

Além disso, devido ao fato de o conjunto de dados utilizado como teste dos classificadores ter sido manualmente rotulado, é possível que os métodos propostos no SOS:D² encontrem uma nova classificação para algumas instâncias de dados. Por exemplo, pode acontecer de uma instância ser classificada como do tipo 1 pelo especialista, mas o método classificou como 3. Nestes casos, faz-se necessário uma discussão sobre os resultados obtidos com o especialista de domínio para a obtenção de um consenso sobre esses casos de inconsistências de classificação, principalmente em relação a classificação em normal

(operacional, sem degradação) e falha (operacional, mas com degradação), pois, em alguns casos, a similaridade entre os comportamentos do AMV gera uma subjetividade na classificação, que varia de acordo com quem classificou manualmente a instância de dados. No caso do diagnóstico de falhas, inconsistências de classificação podem revelar novos comportamentos para o comportamento do AMV. Todavia, esses comportamentos precisam ser validados pelo especialista de domínio.

Portanto, pode-se realizar uma análise de causa-consequência entre as diversas causas encontradas por este processo e as manualmente identificadas. Desta forma, provavelmente, pode-se agrupar ou descobrir novos tipos de falhas. Por fim, gera-se um *ranking* baseando-se no desempenho geral dos métodos aplicados no processo de detecção e diagnóstico do nível de degradação do equipamento.

5 ESTUDO DE CASO

Este capítulo apresenta o estudo de caso realizado como uma forma de validar os resultados obtidos nesta dissertação. A seguir, apresenta-se o contexto de operação de uma ferrovia brasileira – neste caso a Vale S.A, incluindo um problema que a operadora objetiva solucionar e a forma como é realizado, atualmente, o processo de manutenção dos seus equipamentos na via. Além disso, descrevem-se as formas como os métodos FDDs propostos neste trabalho poderiam ser aplicados nesse domínio. Posteriormente, apresentam-se resultados, incluindo a descrição do conjunto de dados e a análise experimental.

5.1 CONTEXTO

Atualmente, a malha ferroviária brasileira, utilizada principalmente para o transporte de cargas, possui 29.074 km de extensão (CNT, 2018). A malha concedida, por intermédio das concessões, é operada e mantida principalmente por empresas privadas (12 operadoras ferroviárias) e estatais (apenas a Estrada de Ferro Paraná Oeste – EFPO) (KLINCEVICIUS, 2011), reguladas pela Agência Nacional de Transporte Terrestre – ANTT.

A *Figura 23* apresenta, em ordem decrescente e de acordo com o tamanho da malha (em km), as concessionárias ferroviárias brasileiras baseando-se na extensão da malha concedida para cada operadora. De acordo com o gráfico é possível observar que a Rumo Malha Sul – RMS, operada pela Rumo S.A, e a Ferrovia Centro-Atlântico – FCA são as maiores ferrovias do país em relação a extensão da malha ferroviária, sendo de apenas 1 Km (quilômetro) a diferença entre elas. Cabe ressaltar que a Estrada de Ferro Carajás – EFC, operada pela Vale S.A, é a ferrovia de transporte de carga mais eficiente do país em relação ao índice de toneladas transportadas por quilômetro útil (TKU) (ASSIS; DALTO; FERREIRA, 2017). A EFC é responsável pelo transporte do minério de ferro produzido em Carajás, maior mina de minério de ferro a céu aberto do mundo, que representa a maior produção de minério de ferro da operadora (VALE, 2020).

No contexto do sistema ferroviário, tanto a disponibilidade quanto a segurança crítica (*safety*) da operação da ferrovia são itens críticos do processo logístico. A disfunção de um equipamento da ferrovia pode ocasionar tanto sua indisponibilidade operacional – produzindo grandes perdas financeiras – quanto acidentes, como descarrilamentos e colisões entre trens ou com objetos em via (incluindo equipamentos e equipes de manutenção), afetando a produção e

causando danos ambientais, materiais (produtividade), imateriais (reputação da operadora) e à vida humana.

Figura 23. Operadoras ferroviárias do Brasil (**Fonte:** Adaptado de (KLINCEVICIUS, 2011; CNT, 2018)).

Assim, e devido a sua grande representatividade, o sistema ferroviário deve possuir um alto nível de confiabilidade, disponibilidade e segurança operacional. Desta forma, na etapa de definição de uma rota, considera-se que os equipamentos e sistemas irão funcionar corretamente, isto é, tanto os ativos da via quanto o sistema de sinalização devem desempenhar corretamente as funções para as quais foram projetados. No entanto, nem sempre isto acontece, pois esses equipamentos estão sujeitos a disfunções.

Neste caso, quando um equipamento apresenta uma disfunção, faz-se necessário efetuar a manutenção desse equipamento de forma eficiente, visando o menor impacto operacional (indisponibilidade) possível. A manutenção dos equipamentos da via representa um importante processo no sistema de transporte ferroviário, envolvendo questões como o custo operacional e a disponibilidade da ferrovia. Neste contexto, a disfunção de uma *Máquina de Chave* pode afetar consideravelmente as ações do trem (ASADA; ROBERTS; KOSEKI, 2013), causar a interdição da via e, no pior caso, provocar acidentes.

A manutenção baseada em condições – CBM (também chamada de manutenção preditiva) pode ser utilizada para evitar essas situações, pois objetiva realizar a manutenção do equipamento previamente a ocorrência de uma disfunção, baseando-se em suas condições

operacionais. Para isso, efetua-se a detecção e o diagnóstico precoce de falhas do equipamento. Isto posto, os métodos propostos no SOS:D² podem evitar a situação mencionada no parágrafo anterior.

5.1.1 A Companhia Vale S.A

A Vale S.A. (VALE) é uma das maiores mineradoras do mundo, possui um valor de mercado de aproximadamente (US\$) 71 bilhões de dólares e em torno de 219.754 acionistas em todos os continentes. A sede da mineradora, atualmente, está estabelecida na cidade do Rio de Janeiro-RJ, Brasil. No entanto, a Vale está presente em diversos países ao redor do mundo. Além disto, é a produtora mundial de minério de ferro, pelotas de minério de ferro e níquel (VALE, 2019). No Brasil, é a operadora responsável pelas estradas de ferro Estrada de Ferro Vitória à Minas – EFVM e Estrada de Ferro de Carajás – EFC.

A Vale S.A, devido a sua representatividade global, busca evoluir simultaneamente com o cenário mundial do setor ferroviário. Por isso, visa ao desenvolvimento de pesquisas internas à empresa, visto que as empresas, via de regra, optam por soluções ‘de prateleira’ e ofertadas por empresas estrangeiras (oferecidas por desenvolvedores e fornecedores internacionais) cujos gastos poderiam ser aplicados no desenvolvimento interno – nacional – de soluções. Desta forma, surgiu a necessidade de criar, com conhecimento nacional, um projeto de sinalização ferroviário futurista, propondo soluções tecnológicas inovadoras para a modernização dos sistemas de sinalização, telecomunicações e energia ferroviários da mineradora.

Como parte desse projeto, tem-se o desenvolvimento de soluções de manutenção preditiva para os ativos (equipamentos) da empresa, incluindo os aparelhos de mudança de via da EFVM. Desta forma, como apresentado anteriormente, este trabalho objetiva auxiliar no processo de desenvolvimento de um programa de manutenção preditiva para esse ativo, por meio da proposição de um conjunto de métodos capazes de monitorar o estado de degradação dos aparelhos de mudança de via – AMVs operados por Máquinas de Chave – MCHs.

Hoje em dia, o processo de manutenção na Vale S.A é realizado tanto de forma corretiva quanto preventiva. Na abordagem corretiva, após a ocorrência de uma disfunção no AMV, a equipe de manutenção é acionada para realizar o reparo do ativo. Já na abordagem preventiva, geralmente, a cada intervalo de tempo, realiza-se a manutenção em um conjunto específico de MCHs e/ou AMVs.

Como primeiro passo para a realização de uma abordagem preditiva, um processo de monitoramento em tempo real das condições desse equipamento foi implantado. A leitura da corrente de acionamento das MCH dos AMV é obtida automaticamente. No entanto, ainda se faz necessário que um especialista de domínio, de forma manual, avalie essas leituras, interprete a assinatura de corrente de acionamento produzida pelas MCHs dos AMVs, classifique o estado de operacionalidade e degradação de cada AMV e analise se é necessário ou não o reparo do ativo.

5.1.2 Possibilidades de uso dos métodos FDD propostos no sistema de monitoramento dos AMVs da Vale S.A.

No contexto do programa de manutenção baseada em condição que a Vale vem implementando, encontra-se a aplicação desse tipo de manutenção sobre os AMVs. Para isso, faz-se necessário obter algoritmos de aprendizado de máquina capazes de executar essa tarefa com um grau adequado de precisão e robustez. Esses algoritmos, no sistema de monitoramento de sinalização que está sendo utilizado, fazem parte do componente *Machine Learning (ML) Algorithms* e auxiliam o especialista de domínio no processo de detecção e diagnóstico do estado de operacionalidade do AMV.

No contexto operacional da Vale S.A, um AMV possui três estados operacionais possíveis: *Normal* (operacional, sem degradação), *Defeito* (operacional, mas com degradação) e *Falha* (indisponível). Por isso, neste capítulo, considera-se essa forma de classificação, denominada “NDF”. Observe que essa forma de classificação difere da definição apresentada em capítulos anteriores para o termo falha e disfunção. Todavia, por questões de adaptação ao domínio de aplicação deste trabalho, a classificação NDF, a partir deste ponto, será adotada para representar o estado de operacionalidade do AMV.

Sabendo disso, existem duas formas de se aplicar os métodos propostos no SOS:D² sobre os dados de assinatura das MCHs da EFVM:

- (i.) *Implantação dos métodos propostos no sistema utilizado na Vale S.A., no qual o método de detecção e diagnóstico do estado de degradação do AMV com o melhor desempenho poderá ser testado no sistema da Vale S.A, em tempo real, por meio da inserção desse método como uma “instância” do*

- componente de algoritmos de ML. Desta forma, o método pode ser aplicado sobre os dados providos da via e fornecer uma classificação preliminar para essa instância de dados e auxiliar no processo de análise do especialista de domínio; e
- (ii.) *Disponibilização de dados para testes locais*, neste caso, por meio de um novo conjunto de dados disponibilizado pela Vale S.A, baseando-se no padrão dos dados já fornecidos, realiza-se o processo de avaliação dos métodos FDDs propostos neste trabalho científico. Assim, devido ao fato de não ser em tempo real, é possível avaliar todos os métodos, além de realizar ajustes de implementação.

Neste estudo de caso, foram disponibilizados pela Vale S.A. um conjunto de dados para a realização de testes locais. Ou seja, os métodos FDDs propostos foram aplicados utilizando a segunda forma de aplicação (Disponibilização de dados para testes locais).

5.2 APLICAÇÃO DOS MÉTODOS PROPOSTOS PARA A DETECÇÃO E DIAGNÓSTICO DO ESTADO OPERACIONAL DOS AMVS DA VALE S.A

Nesta subseção apresenta-se a aplicação do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas - FDDs (*Figura 14*), contribuição principal deste trabalho de investigação científica, sobre um conjunto de dados proveniente de uma aplicação real, fornecido pela Vale S.A.

Como apresentado na seção 4.1 do Capítulo 4 (vide *Figura 14*), esse processo é composto por 3 etapas, cuja entrada é um conjunto de dados da aplicação, e cuja saída são os três (3) modelos de Detecção e Diagnóstico de Falhas quantitativamente ranqueados. Assim, inicialmente, a seção 5.3.1 apresenta o conjunto de dados utilizados nos experimentos. Em seguida, a seção 5.3.2 apresenta o pré-processamento dos dados (etapa 1 da *Figura 14*). Em seguida, a seção 5.3.3 apresenta o desenvolvimento dos modelos de detecção e diagnóstico (etapa 2 da *Figura 14*), na qual as particularidades de cada tarefa – detecção ou diagnóstico – e/ou técnica não supervisionada são descritas. Por fim, a seção 5.3.4 apresenta a avaliação de cada um dos métodos propostos (etapa 3 da *Figura 14*), desenvolvidos na etapa 2, em três cenários de teste.

Nesse experimento, utilizou-se como recursos operacionais um computador com processador Intel Core i5 7ª Geração, 8.0 GB de memória RAM e sistema operacional Linux 16.04 LTS de 64 *bits*. Com relação ao ambiente computacional, utilizou-se o Jupyter Notebook e a linguagem de programação Python. No desenvolvimento dos classificadores foram utilizadas as bibliotecas, Keras e Tensorflow (Perceptron Multicamadas – MLP) e Scikit-learn (Random Forest, AdaBoost, XGBoost e SVM) e para o processamento, manipulação e visualização dos dados foram usadas, respectivamente, pandas, numpy e matplotlib.

5.2.1 Conjunto de dados

O conjunto de dados utilizado no experimento é composto por amostras de corrente elétrica da *Máquina de Chave – MCH*. Esses dados operacionais (amostras elétricas) representam o comportamento dos AMVs, assim como das MCHs, durante o processo de configuração (operação) do AMV (de Normal-para-Reverso ou de Reverso-para-Normal). Essas amostras referem-se à corrente elétrica da MCH durante a configuração de um AMV. Neste caso, a cada operação do AMV, tanto de normal para reverso quanto de reverso para normal, uma nova amostra de dados é obtida.

Como mencionado anteriormente, esses dados operacionais são reais e provenientes de uma ferroviária operada pela Vale S.A, especificamente a Estrada de Ferro Vitória à Minas – EFVM. Esses dados foram coletados, em dois trechos específicos da EFVM, entre agosto de 2019 e julho de 2020. Portanto, representam o estado de operacionalidade desses ativos durante esse período de tempo. Sabendo-se disso, ao todo, tem-se um conjunto de dados com cerca de 29 MB da operação de AMVs. Especificamente, isso equivale a aproximadamente 59.250 assinaturas de corrente elétrica das MCHs referentes às movimentações de configuração (operação) de um AMV de normal para reverso e vice-versa.

A assinatura de um movimento de operação de um AMV (assinatura elétrica da Máquina de Chave) foi dividida em três fases consecutivas, respectivamente; o **acionamento do motor**, o **movimento livre da barra de operação** e a **movimentação do conjunto de frente** (também chamada de movimentação da cesta). A *Figura 24* apresenta cada uma dessas fases. Primeiramente, aciona-se o motor elétrico da MCH, que deve superar o estado estático de suas partes mecânicas e iniciar a movimentação da barra de operação. Em seguida, a barra de operação move-se livremente (quase que sem resistência) até que ela se acople ao cesto das agulhas do AMV (por isso o nome de movimento livre da barra de operação). Finalmente, após

a acoplagem da barra ao cesto das agulhas, a barra de operação empurra e/ou puxa as agulhas até que o alinhamento com os demais trilhos seja finalizado. Neste momento, tem-se o fim do evento padrão de um movimento de configuração de um AMV. Por isso, interrompe-se a corrente elétrica do motor da MCH e considera-se que o AMV foi configurado de acordo com a configuração requerida (garantida por sensores de posição de fim de curso e outros procedimentos).

A leitura da corrente – unidade de medida é ampere (A) –, é realizada por meio de sensores, os quais são implantados nas ‘houses’ (gabinete localizado à margem da via). As ‘houses’ ficam estrategicamente localizadas em pontos de alternância de via (desvios ou bifurcações). A partir de cada amostra de assinatura elétrica da Máquina de Chave, como na *Figura 24* obtém-se diversas características, principalmente estatísticas, dessa assinatura. No contexto da Vale S.A, existe um algoritmo de extração de característica que extrai e armazena essas informações em uma base de dados (no caso deste Estudo de Caso, uma planilha).

Figura 24. Fases da assinatura elétrica durante um movimento de configuração de um AMV (Fonte: Autor).

Portanto, a Vale S.A realiza uma parametrização do comportamento da assinatura elétrica padrão de um movimento de configuração de um AMV em um conjunto de atributos. Esses atributos representam de forma estatística esse comportamento. Em virtude disso, uma linha da planilha representa uma amostra de movimento de configuração do AMV lida da via, assim como as colunas (atributos) representam as características extraídas.

Isto posto, o conjunto de dados original utilizado neste trabalho é composto por 60 (sessenta) atributos. Dentre esses atributos estão a duração das fases (em segundos), o tempo

total do evento, a porcentagem de tempo de cada fase em relação ao tempo total do evento, a área total da curva do evento, a área total da curva em cada fase, a média e mediana da corrente de cada fase e o valor máximo da corrente.

Cabe salientar que das 59.250 instâncias de dados que constituem o conjunto de dados, aproximadamente 5.749 foram manualmente rotuladas pelo operador ferroviário, neste caso, por um especialista de domínio da Vale S.A. Com relação a essa rotulação, utilizou-se a classificação normal (operacional e sem degradação de desempenho), defeito (operacional, mas com degradação de desempenho) e falha (referente ao conceito de disfunção), chamada NDF. Assim, dessas 5.749 instâncias tem-se, 3.691 classificadas como defeituosas, 1.994 como normais e apenas 64 como falha.

Como já comentado anteriormente, essa classificação difere da relação entre falha, erro e disfunção definida, justificada e adotada nos capítulos anteriores deste trabalho. Por questões de concordância com o contexto de aplicação, a classificação considerada no restante deste trabalho é a NDF. Por fim, ressalva-se que o conjunto de dados é dividido e composto por dois subconjuntos: o subconjunto de dados não rotulados, o qual possui uma ampla representatividade dos dados operacionais (90,3%), e o subconjunto de dados manualmente rotulados pela Vale S.A (também chamado de conjunto de dados independente), o qual representa apenas 9,7% desses dados.

5.2.2 Pré-processamento dos Dados (etapa 1)

O diagnóstico do estado de operacionalidade de um AMV consiste em duas tarefas, a saber, a detecção de falhas e o diagnóstico do estado de degradação do equipamento (vide Figura 15). Todavia, faz-se necessário efetuar o pré-processamento dos dados previamente a execução dessas tarefas, como ilustra a Figura 25. A figura destaca essa etapa no escopo do processo apresentado na Figura 14.

Ao todo, na etapa de pré-processamento dos dados, seis atividades foram executadas sobre o conjunto de dados bruto da via, são elas; *seleção de atributos*, *normalização*, *remoção de erros de tipo*, *padronização dos dados*, *geração de atributos* e *filtragem de dados ruidosos*. A seguir apresenta-se uma descrição de cada uma dessas atividades, em ordem de execução.

Figura 25. *Pré-processamento dos Dados*: Primeira etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (**Fonte:** Autor).

1. *Padronização dos Dados*

Essa atividade visa corrigir prováveis erros de intervalo dos dados. Dada a forma como o conjunto de dados é gerado, em alguns casos, existe uma discrepância entre os valores de um atributo. Isto é, existem valores que poderiam ser considerados como *outliers* (considerando apenas os dados desse atributo), mas apenas estão em um intervalo (casas decimais) diferente. Por conta disso, realiza-se a padronização dos dados.

Em cada atributo, todos os valores são analisados. Caso um valor esteja em um intervalo diferente, o qual é identificado por meio de um valor limite definido por inspeção manual, é realizada uma divisão por mil (1000). Por exemplo, considere o caso do atributo que representa a duração da Fase 1 em segundos. Geralmente, tem-se valores entre 0 e 1 segundos. Desta forma, durante a análise desse atributo, valores como 450, 780 e 850 são transformados, respectivamente, em 0,45, 0,78 e 0,85 segundos. Cabe salientar que casos como os exemplificados ocorrem esporadicamente, mas podem atrapalhar no desenvolvimento dos modelos FDDs.

2. *Análise de dados ausentes*

Um problema comum quando se lida com dados de um domínio de aplicação real é a presença de alguns valores ausentes no conjunto de dados analisado (problema conhecido como ausência de dados). Em virtude disso, torna-se necessário realizar algum procedimento, seja de remoção ou de substituição desses dados. Em alguns casos, os dados ausentes são substituídos pela média dos demais dados do atributo (uma coluna da planilha) em que foi identificado a ausência. Neste trabalho, desconsidera-se uma instância (uma linha da planilha) que apresenta ausência de valores. Portanto, essa instância é removida do conjunto de dados. Nesta atividade, além da remoção dos dados ausentes, efetua-se o ajuste de tipo dos dados. Em outras palavras, considerando como exemplo a linguagem *Python*, evita-se que um número (*int* / *float*) seja representado como texto (*string*). Isso pode acontecer durante a leitura dos dados de uma planilha que, por alguma razão, teve alguns dados corrompidos.

3. *Filtragem de dados ruidosos*

Posteriormente à padronização dos dados e à remoção de valores ausentes, executa-se a filtragem de ruídos, isto é, valores aleatórios que não representam informações úteis. Assim, prováveis erros de leitura e/ou amostragem dos dados são identificados. Por exemplo, em conjunto com o especialista de domínio, observou-se que os dados dos meses de agosto e setembro de 2019 podem ser considerados como ruidosos – nesse período um novo processo de leitura da corrente da MCH foi implantado na Vale S.A e, assim, foram excluídos do conjunto de dados nessa atividade.

4. *Seleção de Atributos*

Essa atividade objetiva identificar os atributos mais relevantes do conjunto (base) de dados. Com este propósito, o conjunto de dados manualmente rotulado foi utilizado em uma tarefa de classificação. Nesta tarefa, utilizou-se o classificador *Random Forest* – comumente utilizado para seleção de atributos em conjunto de dados com alta dimensão (MENZE et al., 2009). Assim, apesar do baixo volume de dados utilizado, é possível inferir a relevância de um atributo para a classificação de uma nova instância de dados. Desta forma, os atributos mais relevantes são aqueles que foram importantes na classificação do estado de operacionalidade do AMV.

Basicamente, o conjunto de dados manualmente rotulado foi dividido em treinamento e teste, seguindo proporções comuns, 80% para treinamento e 20% para teste. Após a etapa de teste, por meio de um método do classificador *Random Forest* que calcula a relevância de um atributo na tarefa de classificação, a saber, *feature importance*, selecionam-se os atributos mais relevantes. No caso, os 10 (dez) atributos mais relevantes foram utilizados para gerar novos atributos e, assim, aumentar a representatividade da base de dados.

Os atributos, em ordem decrescente de relevância, selecionados foram:

- *Area3'div.Mediana F2* (área da Fase 3' – lê-se Fase3 linha, uma variação da Fase 3 – dividida pela mediana de corrente da Fase 2);
- *CoeffVariacaoF2* (coeficiente de variação da Fase 2);
- *PerOcAreaF2eF3* (Percentual de ocupação da Fase 2 em relação a Fase 3);
- *RangeF3'* (range da Fase 3');
- *VMax3div.VMaxEvento* (valor máximo de corrente da Fase 3 dividido pelo valor máximo de tempo do evento);
- *VMaximoF3* (valor máximo de corrente da Fase 3);
- *RangeF2* (range da Fase 2);
- *VMax3divMedia2* (valor máximo de corrente da Fase 3 dividido pela média de corrente da Fase 2);
- *DuracaoF3'* (duração em segundo da Fase 3');
- *Media3divMedia2* (média de corrente da Fase 3 dividido pela média de corrente da Fase 2).

5. Geração de Atributos

A partir dos atributos selecionados na atividade anterior, diversos novos atributos são criados e, consequentemente, adicionados ao conjunto de dados original. Ao todo, 20 (vinte) novos atributos foram gerados baseando-se na relação entre esses atributos. Neste contexto, diversas hipóteses foram testadas, tais como a combinação dos atributos que se destacaram na tarefa de classificação mencionada anteriormente.

O objetivo dessa atividade é aumentar a representatividade do conjunto de dados por intermédio dessas combinações dos atributos mais relevantes. Esse processo de geração de atributo foi realizado manualmente. Basicamente, baseando-se nos atributos selecionados na atividade anterior, realiza-se operações matemáticas e/ou cálculos estatísticos com esses atributos.

A *Tabela 6* apresenta o resultado obtido com a geração de atributos. De acordo com os atributos mais relevantes do conjunto de dados original, representados na tabela pelos atributos *V. Max. F3* (valor máximo de corrente da Fase 3), *Area3' div. Mediana F2* (área da Fase 3' – lê-se Fase3 linha, uma variação da Fase 3 – dividida pela mediana de corrente da Fase 2) e *V. MaxF3. div. MediaF3* (valor máximo de corrente dividido pela média de corrente da Fase 3), e os gerados nesta atividade, representados pelos atributos 1 e 2, os quais são gerados por meio de combinação. Por exemplo, os dois novos atributos (1 e 2) são gerados por meio da soma do atributo *V. Max. F3* com o atributo *Area3' div. Mediana 2 (Atributo1)* e com o atributo *V. MaxF3. div. MediaF3 (Atributo2)*, respectivamente.

Tabela 6. Exemplo de criação de novos atributos (**Fonte:** Autor).

Atributos do Conjunto de Dados original			Atributos Criados	
V. Max. F3	Area3' div Mediana F2	V. MaxF3 div MediaF3	Atribut1	Atributo2
10,229	6,715	1,491	16,944	11,79
11,363	8,372	1,432	19,735	12,795
...	...	...	...	...
10,999	7,694	1,444	18,693	12,443

6. Normalização dos Dados

Por fim, os dados são normalizados. Diversas formas de normalização foram analisadas, tais como: a) Mínimo e Máximo: os dados de cada atributo são transformados de modo a pertencer ao intervalo $[0,1]$; b) Z-score: baseia-se na média e no desvio padrão de cada atributo; especificamente, o escore-padrão (Z-score) define o número de desvios padrão que um determinado valor está da média; e c) a transformação por quantil: realiza a transformação dos dados por meio de quantis (medidas de posição), os quais dividem os valores – ordenados – dos atributos em partes iguais.

A transformação por quantis foi escolhida, visto que as técnicas usadas neste trabalho obtiveram um desempenho melhor ao utilizar essa forma de normalização.

5.2.3 Desenvolvimento dos Modelos de Detecção e Diagnóstico do Estado de Operacionalidade do AMV (etapa 2)

Esta subseção refere-se à segunda etapa da sistemática proposta, apresentada na *Figura 14* e em destaque na *Figura 26*, para o desenvolvimento e avaliação dos modelos FDDs. O processo de desenvolvimento dos métodos FDD é constituído por diversas etapas, como ilustra a *Figura 15*. As atividades realizadas nas etapas do processo geral de desenvolvimento desses métodos são descritas nesta subseção. Vale ressaltar que esse procedimento é aplicado individualmente para cada combinação de técnicas (método). Como resultado da execução desse procedimento, tem-se um modelo FDD. Desta forma, esse procedimento foi realizado três vezes (uma para cada combinação – descritas anteriormente) para criar os métodos FDDs. Consequentemente, três modelos FDDs foram obtidos.

Figura 26. *Desenvolvimento dos modelos FDD utilizando técnicas de Aprendizado de Máquina*: Segunda etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (**Fonte:** Autor).

Primeiramente, objetiva-se a detecção do estado de operacionalidade do AMV. Isto é, identificar se o equipamento (AMV) está operacional ou indisponível. Assim, as técnicas utilizadas na proposição dos métodos FDDs, tais como *K-means*, Autoencoder e SOM, foram aplicadas na tarefa de detecção de falhas (AMV operacional ou indisponível) e defeitos (AMV operacional com ou sem degradação). Desta forma, baseando-se nas etapas do processo

apresentado na *Figura 15*, especificamente, da etapa 1 a etapa 3, implementa-se o módulo de detecção de defeitos em um AMVs.

Como mencionado na subseção de desenvolvimento dos modelos FDD do Capítulo 4, para aplicar essas técnicas, especificamente o K -means e o SOM, é necessário definir o número de *clusters* (grupos) que devem ser gerados. Logo, a técnica aplicada deve particionar o conjunto de dados em k grupos. Para determinar o valor ótimo de k , o conjunto de dados não rotulado foi dado como entrada para o algoritmo K -means 19 (dezenove) vezes, em que o valor de k variou de 2 a 20 (veja a *Figura 27*).

No caso do Autoencoder, isso não se faz necessário, uma vez que essa técnica realiza o agrupamento dos dados por meio da definição de um valor limite (*threshold*), o qual é utilizado pela técnica para particionar os dados de entrada. Como dito na subseção de redes neurais do Capítulo 2, um Autoencoder aprende uma representação dos dados recebidos como entrada. Para isto, tenta transformar os dados de entrada em saídas de dados, com uma quantidade mínima de discrepância entre os dados originais e os de saída.

No processo de agrupamento dos dados, o valor limite é comparado com o valor de discrepância de cada instância de dados de entrada, obtido após o treinamento do Autoencoder. Desta forma, valores inferiores ao limite pertencem a um *cluster* e valores superiores ao limite pertencem a outro *cluster*. Esse valor limite é definido como segue: previamente, analisa-se o conjunto de dados manualmente rotulado para identificar a classe majoritária no contexto de cada tarefa não supervisionada. Ou seja, identifica-se a porcentagem da classe falha e defeito em relação a classe normal (atentando-se às particularidades de cada tarefa). Em seguida, considerando a porcentagem obtida, calcula-se o erro quadrático médio dos dados de entrada e define-se o valor obtido como limite para o agrupamento dos dados.

Isto posto, a *Figura 27* ilustra o processo de definição do valor de k para a tarefa de detecção de defeitos para um cenário de teste com, aproximadamente, 53 mil instâncias de dados. Observe que, de acordo com a *Figura 27*, o número ótimo de clusters é 2 (dois). Visto que, baseando-se na Largura de Silhueta, tanto o grau de similaridade entre os elementos de um mesmo grupo (compactação) quanto a distância entre os grupos (separação) são otimizados quando o conjunto de dados é particionado em apenas dois grupos. No entanto, geralmente,

existem vários grupos com comportamentos normais e/ou defeituosos em um domínio de aplicação real.

Figura 27. Definindo o número ótimo de clusters por meio do critério de validação relativa Largura de Silhueta (**Fonte:** Autor).

No caso do SOS:D², isso pode acontecer em decorrência do fato das Máquinas de Chave possuírem modelos diferentes. Dessa forma, esse número ótimo *clusters* pode representar o agrupamento dos dados de acordo com os modelos das MCHs ao invés dos comportamentos normais e defeituosos das MCHs. Uma forma de contornar essa situação é à aplicação do procedimento de definição do número ótimo de *clusters* de forma específica para cada modelo. Por causa da metodologia proposta neste trabalho, esse procedimento individual não é englobado, por isso, o número ótimo de *clusters* da etapa de detecção de defeitos é 2 (dois).

Após a definição do número ótimo de *clusters*, tem-se à aplicação das técnicas não supervisionadas sobre o conjunto de dados não rotulado (etapa 1 da Figura 15). Em seguida, realize-se a interpretação dos *clusters* gerados (etapa 2 da Figura 15). É importante mencionar que a interpretação é efetuada por meio do *algoritmo de interpretação de clusters* (descrito na subseção 4.1 do Capítulo 4). Por conta disso, apesar de o número de grupos, *k*, ser o mínimo aceitável, o conjunto dos dados agrupados (*clusters*) possui representatividade satisfatória das relações existentes no conjunto de dados original.

Isso é possível por causa do algoritmo de interpretação de clusters. Durante a execução desse algoritmo, para cada instância de dados de um grupo, existe a possibilidade de realocação dessa instância e, por consequência, à alteração do pseudo-rótulo definido pela técnica de agrupamento em questão. Desta forma, devido ao fato desse algoritmo exercer a função do especialista de domínio, os *clusters* resultantes dessa etapa (os quais são utilizados na criação do conjunto de dados de treinamento) possuem representatividade dos comportamentos –

representados pelas classes definidas previamente pelo especialista – do AMV no contexto operacional da ferrovia.

Observe que o resultado da etapa de interpretação dos *clusters* é o conjunto de dados de treinamento. Esses dados são utilizados para treinar os diversos modelos de classificação que compõem o classificador *ensemble*. Isto é, para cada conjunto de dados de treinamento resultante das etapas 1 e 2 da *Figura 15* – realizadas para cada combinação de técnicas não supervisionadas –, treina-se um modelo de classificação baseado nas estruturas da MLP, *Random Forest*, *Ada Boost*, *XGBoost* e SVM. O treinamento desses modelos resulta em cinco classificadores-base que são utilizados para criar o classificador *ensemble*, o qual pode ser utilizado para detectar o estado operacional de novas instâncias de dados.

Com relação a etapa de diagnóstico (etapa 4 a 6 da *Figura 15*), o número ótimo de *clusters* obtido pelo procedimento da *Figura 27* é idêntico ao da etapa de detecção, neste caso, dois clusters. Contudo, por meio da análise do conjunto de dados manualmente rotulado – que representa o conhecimento do especialista –, foram definidos 9 (nove) modos de falhas, neste contexto, os tipos de defeitos. São eles: *Atrito*, *Atraso na Partida*, *Corrente Baixa e Oscilando*, *Desligamento do Motor*, *Múltiplos Acionamentos Simultâneos*, *Problema de Medição*, *Problema na Fase 2*, *Sobrecorrente no Destravamento* e *Sobrecorrente no Travamento*. Por conta disso, na etapa de diagnóstico dos defeitos, as instâncias defeituosas são divididas em 9 (nove) grupos.

Além disso, também por intermédio da análise do conjunto de dados manualmente rotulado, definiu-se a criticidade de cada tipo de defeito. A *Tabela 7* mostra a relação de causa-criticidade dos tipos de defeito e as suas criticidades (severidades). As colunas da tabela especificam, respectivamente, o tipo de defeito, o nível de criticidade e a prioridade. Nas linhas tem-se os tipos de defeito identificados por meio da análise do conjunto de dados manualmente rotulado.

O nível de criticidade de um defeito auxilia no processo de manutenção preditiva do AMV. Pois, de acordo com essa criticidade, é possível priorizar o reparo dos ativos que apresentam uma prioridade maior. O fator prioridade, para cada tipo de defeito apresentado na *Tabela 7*, segue uma ordem crescente. Portanto, quanto maior a severidade do defeito, mais urgente deve ser realizado o reparo do ativo.

Neste contexto (modelo NDF), uma falha possui criticidade máxima, a qual pode ser categorizada como gravíssima ou urgente. De fato, uma falha (interrupção do serviço prestado pelo equipamento), no contexto operacional, é mais crítica do que qualquer defeito. Cabe salientar que o conceito de prioridade não se aplica às instâncias de dados consideradas normais, pois nenhuma ação de manutenção ou monitoramento é necessária. Por exemplo, baseando-se nos valores de prioridade listados na *Tabela 7*, pode-se definir um valor 4 (quatro) para a prioridade de uma falha e 0 (zero) para a prioridade de uma instância normal.

Tabela 7. Criticidade de cada tipo de defeito (Fonte: Autor).

Tipo de Defeito	Criticidade	Prioridade
Atrito	Alta	3
Atraso na Partida	Baixa	1
Corrente Baixa e Oscilando	Média	2
Desligamento do Motor	Alta	3
Múltiplos Acionamentos Simultâneos	Média	2
Problema de Medição	Média	2
Problema na Fase 2	Média	2
Sobrecorrente no Destravamento	Alta	3
Sobrecorrente no Travamento	Alta	3

Sabendo disso, diferentemente da tarefa de detecção dos defeitos, no processo de diagnóstico dos defeitos o valor de k , dado como entrada para as técnicas não supervisionadas, foi 9 (nove). Desta forma, à aplicação de uma dessas técnicas gera nove *clusters*. Esses clusters são interpretados pelo algoritmo de interpretação de clusters – por meio de uma análise relativamente diferente da aplicada na detecção dos defeitos. Note que neste caso, analisa-se diretamente a relação dos tipos de defeito e não apenas o estado operacional do AMV – e o resultado desse procedimento é usado para gerar o conjunto de dados de treinamento do processo de diagnóstico dos defeitos.

Esse conjunto de dados é utilizado para criar o classificador *ensemble* (etapa 6 da *Figura 15*). O processo de treinamento dos modelos de classificação que compõem esse classificador difere do processo da detecção dos defeitos apenas no tocante ao tipo de classificação. Sendo que no diagnóstico realiza-se uma classificação multiclasse ao invés de uma classificação binária. Por fim, após o desenvolvimento dos métodos, e consequentemente dos modelos FDD, é possível avaliar o desempenho dos modelos por intermédio de uma tarefa de classificação, utilizando-se como dados de teste o conjunto de dados manualmente rotulado. A avaliação desses modelos FDD (etapa 3 da *Figura 14*) é realizada na próxima subseção.

5.2.4 Avaliação dos Modelos de Detecção e Diagnóstico do Estado de Operacionalidade do AMV

Os modelos FDDs, obtidos como resultado da implementação dos métodos FDDs, desenvolvidos por intermédio da realização das atividades apresentadas na subseção anterior – a qual refere-se a segunda etapa da *Figura 26* –, foram avaliados em **três cenários de testes**. Neste caso, devido a forma como implementa-se o método, o processo de detecção e diagnóstico de falhas pode ser compreendido como uma tarefa de classificação, a qual é dividida em duas etapas, a saber, detecção e diagnóstico de falhas – defeitos, no contexto Vale. Em vista disso, nesta subseção avalia-se o desempenho dos modelos FDDs (como destaca a *Figura 28*) em cada um dos cenários de teste. Esses cenários (casos) foram criados por meio da divisão do conjunto de dados não rotulado original em subconjuntos de dados de tamanhos distintos.

Figura 28. *Avaliação dos modelos FDDs*: Terceira etapa do processo de desenvolvimento e avaliação dos Modelos de Detecção e Diagnóstico de Falhas (**Fonte:** Autor).

Para isto, dada a quantidade de instâncias de dados disponível no conjunto de dados bruto, definiu-se os valores máximos de instâncias para cada subconjunto de dados como sendo, respectivamente, 20 mil e 30 mil para o primeiro e segundo cenários de teste. Esses valores foram definidos visando uma análise do desempenho dos modelos FDDs de modo gradual no tocante à quantidade de dados bruto utilizados no processo de desenvolvimento dos mesmos.

Esses subconjuntos de dados são gerados como segue: o primeiro subconjunto é formado pelas primeiras 20 mil instâncias do conjunto de dados bruto – assumindo que esses dados estão em uma planilha, considera-se as primeiras 20 mil linhas. Nas mesmas condições, o segundo subconjunto é constituído pelas primeiras 30 mil instâncias dos dados brutos. Ou seja, o segundo subconjunto possui um acréscimo de 10 mil instâncias. Desta forma, a partir de cada subconjunto gerado, tem-se um cenário de teste. Logo, esses subconjuntos resultam em dois casos de teste. O terceiro cenário de teste é formado pelo conjunto de dados original, sem qualquer divisão.

Esses cenários de teste são mostrados na *Tabela 8*. Cada coluna da tabela especifica uma propriedade do cenário, respectivamente, a identificação, descrição, quantidade de instâncias de dados inicial e, por fim, quantidade de instâncias de dados do novo conjunto de dados, resultante da etapa de pré-processamento dos dados brutos. Com viés de comparação de resultados, em cada cenário, o conjunto de dados manualmente rotulados foi utilizado no teste de performance dos modelos FDD.

Neste caso, utilizou-se o conjunto de dados independente (manualmente rotulado pelo operador) para avaliar o desempenho de cada modelo no processo de detecção e diagnóstico do estado de operacionalidade do AMV – procedimento referente a etapa 3 da *Figura 14* e destacado na *Figura 28* –. Assim, os conjuntos de dados, obtidos pela aplicação das técnicas – conforme o método – sobre o subconjunto de dados não rotulados, são utilizados na etapa de treinamento dos classificadores *ensemble*, tanto na tarefa de detecção quanto na de diagnóstico.

Por fim, durante a análise do desempenho do classificador *ensemble* na tarefa de detecção do estado de operacionalidade do AMV, as principais métricas analisadas são a **precisão** e a **sensibilidade** obtidas pelo classificador (LEE et al., 2018). No caso do diagnóstico de falhas, considera-se como métrica principal a **acurácia**. Ao todo, nove desempenhos serão avaliados. Esses desempenhos dizem respeito à aplicação dos três modelos sobre os três cenários de teste descritos. Sabendo-se disso, em seguida, descrevem-se os resultados de cada modelo FDD por cenário de avaliação. No final desta seção, apresenta-se uma visão geral desses resultados.

Tabela 8. Cenários de teste para a avaliação dos modelos FDDs (**Fonte:** Autor).

Cenários de Teste			
Cenários	Descrição	Tamanho original do conjunto de dados (instâncias)	Tamanho do conjunto de dados após a etapa de pré-processamento
Cenário 1	O conjunto de dados bruto é constituído por 20 mil instâncias de dados não rotuladas, advindas do conjunto de dados original.	20.000 instâncias	15.058 instâncias
Cenário 2	O conjunto de dados bruto é constituído por 30 mil instâncias de dados não rotuladas, advindas do conjunto de dados original.	30.000 instâncias	25.045 instâncias
Cenário 3	O conjunto de dados bruto utilizado neste cenário é o próprio conjunto de dados original, sem particionamento.	53.501 instâncias	48.543 instâncias

5.2.4.1 Modelo 1: Obtido a partir do método 1 (combinação das técnicas *K*-means e SOM)

De acordo com a estrutura dos métodos, no primeiro método (*K*-means & SOM), o conjunto de dados utilizado para o treinamento do detector ensemble na detecção de defeitos provém da aplicação do algoritmo *K-means* sobre o conjunto de dados pré-processado. Do mesmo modo que, na especificação dos defeitos, o conjunto de dados de treinamento provém da aplicação da rede SOM sobre o subconjunto das instâncias de dados defeituosas (conforme o processo de desenvolvimento ilustrado na *Figura 15*).

As *Figuras 29* e *30* apresentam, respectivamente, a matriz de confusão do detector *ensemble* de falhas e defeitos no contexto do **primeiro cenário** de teste. O detector *ensemble* na tarefa de detecção de falhas conseguiu identificar corretamente todas as falhas existentes no conjunto de dados de teste. Isto é, obteve uma sensibilidade de 100% na identificação das falhas. No entanto, o detector *ensemble* cometeu alguns erros de classificação, visto que classificou algumas instâncias como sendo falhas, mas essas instâncias são consideradas normais pelo especialista de domínio.

Por conta disso, a precisão do detector *ensemble* foi de 50%. Vale salientar que, devido à propriedade de desbalanceamento do conjunto de dados, durante a análise de desempenho do detector priorizou-se a classificação, de forma correta, da classe minoritária, isto é, as falhas.

Por exemplo, de acordo com a *Figura 29* e com a descrição das métricas de desempenho descritas na subsecção 2.5, a precisão é calculada como segue:

$$precisão = \frac{TP}{(TP+FP)} \Rightarrow \frac{35}{(35+35)} \Rightarrow \frac{35}{70} = 0,5 \text{ ou } 50\%, \text{ em que TP (verdadeiro}$$

positivo) é o número de instâncias corretamente classificadas como falhas e FP (falso positivo) é o número de instâncias incorretamente classificadas como falhas.

Figura 29. Modelo 1 - Cenário 1: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

Figura 30. Modelo 1 - Cenário 1: Matriz de confusão da etapa de detecção de defeitos (**Fonte:** Autor).

De modo geral, priorizar a identificação das falhas é uma forma de assegurar à análise não enviesada do detector de falhas em um contexto com classes desbalanceadas, problema comum em cenários do mundo real. Sabendo disso, o mesmo acontece na tarefa de detecção de defeitos. Isto é, prioriza-se a identificação dos defeitos ao invés dos dados normais, visto que, teoricamente, essa deve ser a classe minoritária e mais relevante para o operador ferroviário. No entanto, como pode ser observado na *Figura 30*, no domínio de aplicação (*Figura 1*) deste

trabalho, essa relação não é válida. Uma vez que a classe defeituosa é a classe majoritária. Apesar disso, ela continua sendo a classe priorizada durante o processo de análise de desempenhos. Os resultados do detector *ensemble* de defeitos, no âmbito do primeiro caso de teste, são apresentados na primeira linha da *Tabela 9*, na coluna referente a detecção de defeitos.

Na tarefa de detecção de defeitos, o detector *ensemble* é avaliado com todas as instâncias de dados normais do conjunto de dados de teste usado na tarefa de classificação de falhas, incluindo as instâncias que foram preditas incorretamente como falhas. É importante informar que, caso o detector fosse aplicado em tempo real, caso em que não é possível identificar previamente o rótulo de uma nova instância de dados, as instâncias classificadas incorretamente como falhas poderiam ser repassadas para o processo de manutenção sem grandes perdas de produtividade.

Essas instâncias, após a classificação, podem passar por uma análise para confirmar a necessidade de realização ou não da manutenção do ativo. Observe que, no contexto operacional da via, um alarme falso é menos crítico do que informar que uma máquina de chave está operacional, quando na verdade ela está indisponível. Contudo, alarmes falsos não são desejáveis.

Posteriormente a detecção dos defeitos, tem-se a especificação do que ocasionou esse desvio de comportamento em cada instância. Conforme o primeiro método (*K-means* & SOM), o conjunto de dados utilizado para o treinamento do classificador *ensemble* na etapa de diagnóstico provém da aplicação da rede SOM sobre um subconjunto dos dados de treinamento da etapa de detecção, formado pelas instâncias de dados pseudo-classificadas como defeitos. A *Figura 31* apresenta, por meio de uma matriz de confusão, a distribuição das instâncias defeituosas nas nove classes definidas em conjunto com o especialista de domínio.

Observe que o classificador *ensemble* do primeiro modelo, quando aplicado sobre os dados do primeiro cenário de teste, na tarefa de especificação do tipo de defeito, conseguiu um bom desempenho sobre três das nove classes definidas. Ou seja, o classificador *ensemble*, durante o seu treinamento, aprendeu uma boa generalização para as classes *Atrito*, *Problema de Medição* e *Problema na Fase 2*. Porém, o mesmo não aconteceu para as demais classes.

Por exemplo, com relação as classes *Corrente Baixa e Oscilando*, *Múltiplos Acionamentos Simultâneos* e *Sobrecorrente no Destravamento* apenas algumas instâncias foram corretamente classificadas. O classificador *ensemble* não obteve uma boa generalização para as classes *Desligamento do Motor*, *Atraso na Partida* e *Sobrecorrente no Travamento*. Os

valores das métricas de desempenho da etapa de diagnóstico, no cenário de teste em evidência, do primeiro modelo são apresentados na primeira linha da *Tabela 9*, na coluna referente ao diagnóstico dos defeitos.

Figura 31. Modelo 1 – Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

No âmbito do **segundo cenário** (caso) de teste, cuja a principal diferença em relação ao primeiro cenário é a quantidade de instâncias de dados do subconjunto de dados não rotulado – tem-se um acréscimo de 10 mil instâncias –, o detector *ensemble* de falhas do primeiro modelo também conseguiu identificar corretamente todas as instâncias falhas. Por meio da análise das *Figuras 29 e 32* pode-se observar que o detector gerou um número maior de alarmes falsos (falso positivo) no caso de teste mencionado. No entanto, apesar de afetar o desempenho geral do detector de falhas, isso não afetou a sua capacidade de identificar novas falhas.

Com relação a identificação dos defeitos, o detector *ensemble* apresentou um aprimoramento em seu desempenho. Pois, quando comparado ao primeiro cenário, o desempenho do detector no segundo cenário foi superior. Principalmente no tocante a identificação das instâncias normais, pertencentes a classe minoritária. Observando as *Figuras 30 e 33*, é possível inferir que o primeiro modelo, no contexto do segundo cenário de teste, aprimorou a sua taxa de acertos em 42 instâncias de dados. Dessas instâncias, metade foram detectadas corretamente como defeito e as demais como normal. Perceba que o percentual de

acertos do detector em cada classe aumentou de forma proporcional. No entanto, o ganho de precisão foi superior ao ganho de sensibilidade, em relação ao desempenho do detector *ensemble* de defeitos. Isso acontece devido a priorização da classe defeituosa – veja as duas primeiras linhas da *Tabela 9*–.

Figura 33. Modelo 1 – Cenário 2: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).

Figura 32. Modelo 1 – Cenário 2: Matriz de confusão da etapa de detecção de defeitos (Fonte: Autor).

Ainda com relação ao segundo cenário de teste, no processo de especificação dos eventos que provocaram os defeitos (diagnóstico), o desempenho do classificador *ensemble* – do modelo em questão – foi superior à do primeiro caso de teste. Como pode ser evidenciado pela comparação das matrizes de confusão desse classificador, representadas pelas *Figuras 31* e *34*, a porcentagem de acertos do primeiro modelo foi maior no segundo cenário de teste. Principalmente, em relação ao percentual de acertos por tipo de defeito. Outra forma de se evidenciar essa superioridade de desempenho é por meio das métricas calculadas em cada situação. Essas métricas são apresentadas nas duas primeiras linhas da *Tabela 9*, na coluna referente a etapa de diagnóstico.

Figura 34. Modelo 1 – Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Isto posto, vale ressaltar que esses cenários de teste – primeiro e segundo – foram criados por meio da divisão do conjunto de dados original em subconjuntos, conforme descrito anteriormente. No terceiro caso de teste, todo o conjunto de dados original é utilizado no desenvolvimento dos métodos FDDs. Isso impacta diretamente na quantidade de instâncias de dados úteis – instâncias pré-processadas – usadas na criação desses métodos e, consequentemente, dos modelos FDDs. Com relação aos demais cenários, como foi exposto na *Tabela 8*, o terceiro caso de teste possui o triplo de instâncias do primeiro e quase duas vezes mais do que o segundo.

O detector *ensemble* de falhas do primeiro modelo FDD quando aplicado sobre os dados do **terceiro cenário** de teste, assim como nos cenários anteriores, cometeu alguns erros de classificação. Porém, com aumento da quantidade de instâncias analisadas, o detector não manteve a excelência na detecção de novas instâncias falhas. Isto é, a sensibilidade do modelo caiu. Dado que, em relação aos cenários anteriores, o detector não conseguiu identificar corretamente todas instâncias falhas (veja a *Figura 35*), além de gerar um maior número de falsos positivos.

Figura 35. Modelo 1 – Cenário 3: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).

Figura 36. Modelo 1 – Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).

Ainda com relação a sensibilidade dos detectores *ensemble*, diferentemente do detector de falhas, o detector de defeitos apresentou um aumento significativo na identificação, de maneira correta, dos defeitos. Se comparado ao cenário anterior, como pode ser observado nas Figuras 33 e 36, o detector reduziu pela metade o número de falsos negativos. No entanto, o detector perdeu precisão. Isto é, houve uma redução da capacidade de identificar instâncias normais. Quando comparamos os resultados obtidos por esse detector em todos os cenários de teste (veja a Tabela 9), apenas no contexto do modelo FDD em questão, é possível perceber que o mesmo, no terceiro caso de teste, obteve a melhor sensibilidade e a pior precisão na tarefa de detecção do estado de operacional do AMV.

O desempenho do classificador *ensemble*, do modelo em evidência, na etapa de diagnóstico foi inferior as performances apresentadas, por esse mesmo modelo, nos demais casos de teste. Apesar disso, esse classificador excede em desempenho os demais

classificadores, nas condições do primeiro modelo, na porcentagem de acertos sobre as classes *Atrito*, *Atraso na Partida* e *Problema na Fase 2* (veja a *Figura 37*). Para as outras classes o seu desempenho foi equivalente ou inferior.

Figura 37. Método 1 – Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Ademais, é importante mencionar que em todos os modelos FDD utilizou-se o Autoencoder – como técnica – no processo de detecção não supervisionada de falhas. Isso porque, conforme mencionado anteriormente, é a melhor técnica para essa tarefa no contexto deste trabalho. Consequentemente, o desempenho dos classificadores *ensemble* nessa etapa não possuem variações de performance consideráveis. Essas variações ocorrem porque, em cada modelo, o classificador é treinado novamente com o conjunto de treinamento gerado durante o processo de detecção não supervisionada do estado de operacionalidade do AMV, respeitando a metodologia proposta. Por conta disso, embora seja uma atividade essencial do processo, os resultados da etapa de detecção de falhas não foram incorporados na avaliação geral dos resultados dos modelos FDD propostos.

Sabendo disso, a *Tabela 9* resume os resultados obtidos pelo primeiro modelo FDD no processo de detecção e diagnóstico do estado de operacionalidade do AMV em cada cenário de teste. As linhas da tabela representam, em ordem crescente, os referidos cenários de teste. As

colunas especificam tanto a tarefa de classificação realizada (detecção ou diagnóstico) quanto as métricas de desempenho dos classificadores *ensemble* nessa tarefa.

Tabela 9. Desempenho do primeiro modelo FDD, referente ao método 1 (combinação das técnicas K-means e SOM), nos três cenários de teste (**Fonte:** Autor).

Cenários de Teste	Métricas de Desempenho no processo de Detecção e Diagnóstico de Defeitos							
	Detecção de Defeitos				Diagnóstico de Defeitos			
	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Acurácia</i>	<i>Média-F1</i>	<i>Acurácia</i>	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Média-F1</i>
Cenário 1	93.24%	90.44%	88.00%	91.82%	76.73%	75.81%	76.73%	71.00%
Cenário 2	94.91%	92.06%	90.41%	93.41%	77.55%	76.35%	77.55%	72.84%
Cenário 3	91.35%	95.30%	89.78%	93.28%	75.32%	70.05%	75.32%	69.88%

Por fim, baseando-se na argumentação exposta e nos resultados exibidos, pode-se concluir sobre o primeiro modelo FDD que:

1. Em todos os cenários de teste, o detector *ensemble* de falhas gerou falsos positivos. Sendo que a quantidade de falsos positivos aumentou de forma gradual com a aplicação do modelo em cada cenário. No terceiro cenário de teste esse detector identificou incorretamente uma instância falha como normal. Portanto, o mesmo não manteve a excelente sensibilidade na detecção de falhas em todos os cenários de teste, apenas no primeiro e segundo cenário.
2. Nos três cenários de teste, o desempenho do detector *ensemble* de defeitos foi satisfatório, apesar da ocorrência de falsos positivos e negativos. Sendo que a melhor performance do detector *ensemble* de defeitos, em relação a identificação correta das instâncias de dados defeituosas – refere-se à sensibilidade do detector –, ocorreu no contexto do terceiro cenário. Todavia, nas condições do segundo caso de teste, o detector alcançou o seu melhor desempenho na identificação, de modo correto, de instâncias de dados normais – refere-se à precisão do detector –.
3. Assim como no caso da detecção, em todos os casos de teste, o desempenho do classificador *ensemble* da etapa de diagnóstico dos defeitos foi satisfatório, apesar de não manter uma boa taxa de acertos em todos os tipos de defeitos identificados. Esse classificador *ensemble* obteve a melhor performance, na etapa de diagnóstico de falhas, quando aplicado sobre os dados do segundo cenário.

5.2.4.2 Modelo 2: Obtido a partir do método 2 (combinação das técnicas Autoencoder e *K*-means)

Neste modelo, conforme a estrutura dos métodos, o conjunto de dados de treinamento do detector *ensemble* na detecção de defeitos provém da aplicação do algoritmo Autoencoder sobre o conjunto de dados pré-processado. Assim como na especificação dos defeitos, o conjunto de dados de treinamento advém da aplicação do algoritmo *K*-means sobre o subconjunto das instâncias de dados defeituosas.

As Figuras 38 e 39 mostram a matriz de confusão das etapas de detecção de falhas e defeitos, respectivamente, do segundo modelo FDD (Autoencoder & *K*-means). Essas matrizes foram obtidas como um dos resultados da aplicação desse modelo sobre os dados do **primeiro cenário** de teste. Observe que, apesar da quantidade de falsos positivos, o detector *ensemble* de falhas desse modelo conseguiu identificar corretamente todas as instâncias de dados falhas. Ou seja, o detector alcançou sensibilidade máxima – 100% – na tarefa de detecção de falhas.

Na tarefa de detecção de defeitos, que se refere à identificação do estado de operacionalidade do AMV, o detector *ensemble* de defeitos do modelo em questão obteve um desempenho satisfatório. Todavia, apesar da boa generalização aprendida, o detector gerou tanto falsos negativos quanto falsos positivos. Uma vez que identificou incorretamente como defeituosas 87 (oitenta e sete) instâncias normais – falsos positivos –. Assim como 111 (cento e onze) instâncias defeituosas foram incorretamente identificadas como normais – falsos negativos –.

Figura 38. Modelo 2 - Cenário 1: Matriz de confusão da etapa de detecção de falhas (Fonte: Autor).

Figura 39. Modelo 2 - Cenário 1: Matriz de confusão da etapa de detecção de defeitos (Fonte:

Com relação ao diagnóstico dos defeitos, o classificador *ensemble* do segundo modelo FDD obteve um desempenho satisfatório. Como pode ser observado na *Figura 40*, esse classificador conseguiu um bom desempenho sobre três das nove classes definidas. Portanto, durante o seu treinamento, o classificador *ensemble* aprendeu uma boa generalização para as classes *Atrito*, *Problema de Medição* e *Problema na Fase 2*.

Figura 40. Modelo 2 - Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Entretanto, a generalização aprendida para as demais classes não foi eficaz. Visto que, em alguns casos, o classificador *ensemble* não conseguiu classificar corretamente nenhuma

instância de dados. Como exemplo temos as classes *Desligamento do Motor*, *Atraso na Partida*, *Múltiplos Acionamentos Simultâneos* e *Sobrecorrente no Travamento*. Além disso, aconteceram casos em que o classificador classificou corretamente apenas algumas instâncias de algumas classes, tais como *Corrente Baixa e Oscilando* e *Sobrecorrente no Destravamento*. As métricas de desempenho desse classificador na etapa de diagnóstico, em relação ao primeiro cenário de teste e segundo modelo FDD, são apresentados na primeira linha da *Tabela 10*, na coluna referente ao diagnóstico dos defeitos.

Com a aplicação do modelo em evidência no **segundo cenário** de teste, na tarefa de detecção de falhas, o detector *ensemble* de falhas também obteve um desempenho satisfatório. No entanto, gerou um número maior de falsos positivos se comparado a seu desempenho no primeiro cenário de teste (veja as *Figura 38* e *Figura 41*). Na tarefa de detecção dos defeitos, o detector *ensemble* de defeitos alcançou um desempenho excedente o desempenho desse mesmo detector no contexto do primeiro cenário de teste. De fato, embora sutil, é perceptível a diferença de performance entre esses detectores ao analisar tanto as matrizes de confusão, ilustradas nas *Figura 39* e *Figura 42*, quanto as métricas de desempenho mostradas nas duas primeiras linhas da *Tabela 10*, na coluna relacionada a detecção de defeitos.

Figura 41. Modelo 2 - Cenário 2: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

Na etapa de diagnóstico dos defeitos, diferentemente do processo de detecção dos mesmos, o classificador *ensemble* não conseguiu exceder os resultados obtidos no primeiro cenário de teste. De fato, como é possível observar nas *Figura 40* e *Figura 43* e nas duas linhas iniciais da *Tabela 10*, o mesmo alcançou um desempenho relativamente inferior. Por conta disso, o desempenho do classificador *ensemble*, apesar de regular, não apresentou nenhum aprimoramento com o aumento do subconjunto de dados não rotulado. Portanto, no caso do

segundo modelo, uma maior quantidade de dados não rotulados auxilia o processo de detecção dos defeitos, mas não influencia consideravelmente no diagnóstico.

Figura 42. Método 2 - Cenário 2: Matriz de confusão da etapa de detecção dos defeitos (Fonte: Autor).

Ainda conforme as *Figura 40* e *Figura 43*, o classificador *ensemble* na etapa de diagnóstico manteve o desempenho regular sobre as mesmas três classes, a saber, *Atrito*, *Problema de Medição* e *Problema na Fase 2* do primeiro cenário de teste. Ou seja, tanto no primeiro cenário quanto no segundo cenário, o classificador *ensemble* do segundo modelo FDD obteve um desempenho regular sobre essas classes.

Figura 43. Modelo 2 - Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Entretanto, no segundo cenário, a porcentagem de acertos na classe *Atrito* aumentou, enquanto que nas classes *Problema de Medição* e *Problema na Fase 2*, diminuiu. Além dessas classes, o classificador *ensemble* também não foi eficiente sobre as classes *Corrente Baixa e Oscilando* e *Sobrecorrente no Destravamento*, caso em que perdeu a capacidade de generalização e, assim, não conseguiu classificar corretamente nenhuma instância dessa classe. Além disso, o classificador permaneceu com um desempenho insatisfatório sobre as classes *Desligamento do Motor*, *Atraso na Partida*, *Múltiplos Acionamentos Simultâneos* e *Sobrecorrente no Travamento*.

No **terceiro cenário** de teste, em relação a detecção correta de instâncias defeituosas, o desempenho do detector *ensemble* de defeitos do segundo modelo excede o desempenho do mesmo nos cenários anteriores. Por conta disso, quando aplicado sobre os dados do terceiro caso de teste, o detector *ensemble* conseguiu obter a maior sensibilidade na detecção das instâncias pertencentes a classe *Defeito*.

No entanto, apesar do ganho em sensibilidade do detector por causa do aumento do número de instâncias identificadas corretamente como defeituosas (observe a *Figura 44*), houve uma perda de precisão desse detector – sendo a menor em relação aos demais cenários –. Isso significa que a generalização aprendida, no treinamento do detector, possibilita identificar melhor a classe *Defeito*, mas o mesmo não acontece com a classe *Normal*.

Figura 44. Modelo 2 - Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (**Fonte:** Autor).

Porém, esse aumento de desempenho não foi observado na detecção de falhas. De fato, como evidencia a *Figura 45*, o detector apresentou resultados inferiores aos resultados obtidos com a aplicação do segundo modelo nos demais cenários de teste. Observe que o detector, neste cenário, gerou uma maior quantidade de falsos positivos. Além disso, diferentemente dos outros

casos, não conseguiu identificar corretamente todas as instâncias falhas do conjunto de teste. Isso caracteriza a perda de sensibilidade desse detector na tarefa detecção de falhas.

Figura 45. Modelo 2 - Cenário 3: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

A *Figura 46* apresenta, por meio de uma matriz de confusão, a quantidade de acertos por classe do classificador *ensemble* do segundo modelo no processo de diagnóstico dos defeitos. Por meio da análise dessa matriz de confusão pode-se observar, apesar da taxa de acertos do classificador ser relativamente inferior as taxas de acerto desse mesmo classificador nos demais cenários, a taxa de acertos por classe foi a maior, do modelo FDD em questão nos três casos de teste, em relação a quantidade de classes em que pelo menos uma instância foi classificada corretamente.

Figura 46. Modelo 2 - Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (**Fonte:** Autor).

Inclusive, no terceiro caso de teste, o classificador *ensemble* conseguiu classificar corretamente algumas instâncias das classes *Atraso na Partida* e *Múltiplos Acionamentos Simultâneos*. Sendo que nos primeiros cenários não conseguiu classificar, de forma correta, nenhuma instância dessas classes. Porém, devido a boa quantidade de acertos nas demais classes, por exemplo *Atrito*, a taxa de acerto nos casos de teste iniciais torna-se superior.

Isto posto, a *Tabela 10* resume os resultados obtidos pelo segundo modelo FDD no processo de detecção e diagnóstico do estado de operacionalidade do AMV em cada cenário de teste. Essa tabela possui a mesma estrutura da *Tabela 9*, logo, as linhas representam os cenários de teste e as colunas representam tanto a tarefa na qual o modelo foi avaliado quanto as métricas de desempenho dos classificadores e/ou detectores *ensembles* na tarefa executada.

Tabela 10. Desempenho do segundo modelo FDD, referente ao método 2 (combinação das técnicas Autoencoder e *K*-means), nos três cenários de teste (**Fonte:** Autor).

Cenário de teste	Métricas de Desempenho no processo de Detecção e Diagnóstico de Defeitos							
	Detecção de Defeitos				Diagnóstico de Defeitos			
	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Acurácia</i>	<i>Média-F1</i>	<i>Acurácia</i>	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Média-F1</i>
Cenário 1	93.17%	91.44%	88.63%	92.30%	77.23%	76.08%	77.23%	72.49%
Cenário 2	95.22%	92.21%	90.75%	93.69%	76.84%	72.68%	76.84%	71.87%
Cenário 3	91.51%	95.53%	90.06%	93.47%	74.50%	73.96%	74.50%	69.81%

Finalmente, de acordo com os resultados apresentados, pode-se concluir sobre o segundo modelo que:

1. No processo de detecção de falhas, em cada cenário, a quantidade de falsos positivos gerados pelo detector *ensemble* aumentou gradualmente. Nos dois primeiros cenários de teste – primeiro e segundo –, conseguiu identificar corretamente todas as falhas. Entretanto, no terceiro cenário de teste identificou incorretamente uma instância falha como normal. Por conta disso, o detector *ensemble* de falhas não manteve a excelente sensibilidade na detecção de falhas em todos os cenários de teste.
2. Nos três cenários de teste, o desempenho do detector *ensemble* de defeitos foi satisfatório, independentemente da quantidade de falsos positivos e negativos gerados pelo detector. Os erros de classificação, os quais representam os falsos

positivos e negativos, foram reduzindo gradualmente conforme a aplicação do modelo em questão nos primeiros cenários de teste. No terceiro cenário de teste o número de falsos positivos teve um aumento significativo, enquanto que o número de falsos negativos diminuiu também de forma significativa. Por conta disso, dada a priorização da identificação de instâncias defeituosas, a melhor performance do detector *ensemble* de defeitos, em relação a identificação correta das instâncias defeituosas, ocorreu no contexto do terceiro cenário. Porém, quando aplicado sobre os dados do segundo caso de teste, o detector alcançou o seu melhor desempenho na identificação, de forma correta, das instâncias normais.

3. Na etapa de diagnóstico dos defeitos, em todos os cenários, o desempenho do classificador *ensemble* foi regular, principalmente para três classes (tipos de defeito), a saber, *Atrito*, *Problema de Medição* e *Problema na Fase 2*. Além disso, foi observado uma redução de desempenho desse classificador ao longo dos cenários de teste. Sendo que no terceiro cenário, esse classificador obteve a sua pior performance. Entretanto, foi nesse cenário em que o classificador conseguiu classificar corretamente instâncias de um maior número de classes. Por fim, a melhor performance desse classificador ocorreu no primeiro cenário de teste.

5.2.4.3 Modelo 3: Obtido a partir do método 3 (combinação das técnicas Autoencoder e SOM)

Neste modelo, assim como no segundo modelo FDD, o conjunto de dados de treinamento do detector *ensemble* na detecção de defeitos provém da aplicação do algoritmo Autoencoder sobre o conjunto de dados pré-processado. Todavia, na especificação dos defeitos, o conjunto de dados de treinamento advém da aplicação da rede SOM sobre o subconjunto das instâncias de dados defeituosas. Uma vez que esse modelo é referente ao terceiro método.

Conforme apresenta a *Figura 47*, a aplicação do modelo em evidência sobre os dados do **primeiro cenário** de teste possibilitou ao detector *ensemble* de falhas alcançar uma excelente sensibilidade – 100% – na detecção de novas falhas. Porém, o mesmo não é válido para as instâncias normais. Uma vez que o detector cometeu alguns erros de classificação, isto é, gerou alarmes falsos sobre a ocorrência de uma falha.

Figura 47. Modelo 3 – Cenário 1: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

Além disso, ainda no contexto do primeiro cenário de teste, o detector *ensemble* de defeitos obteve um bom desempenho no processo de detecção dos defeitos. Consequentemente, a generalização aprendida pelo detector para as classes analisadas – *Normal* e *Defeito* – foi razoável. Como pode ser observado na *Figura 48*, esse detector gerou uma alta quantidade de falsos positivos e negativos, sendo o segundo caso superior ao primeiro. Neste caso, 100 (cem) instâncias foram identificadas incorretamente como defeituosas – falsos positivos – e 108 (cento e oito) instâncias foram identificadas incorretamente como normais – falsos negativos –.

Figura 48. Modelo 3 – Cenário 1: Matriz de confusão da etapa de detecção de defeitos (**Fonte:** Autor).

No processo de diagnóstico dos defeitos, como ilustra a *Figura 49*, o classificador *ensemble* não conseguiu classificar de forma correta nenhuma instância de dados de cinco classes entre as nove analisadas. Consequentemente, o desempenho do classificador sobre essas classes foi insatisfatório. Dentre essas classes temos, *Atraso na Partida*, *Desligamento do Motor*, *Múltiplos Acionamentos Simultâneos*, *Sobrecorrente no Destravamento* e *Travamento*. Contudo, o classificador obteve uma taxa de acertos satisfatória para as classes *Atrito*, *Problema de Medição* e *Problema na Fase 2*. No caso da classe *Corrente Baixa e Oscilando*, apenas algumas instâncias foram corretamente classificadas. Por fim, de modo geral, a taxa de acertos por classe do classificador foi baixa e, portanto, insatisfatória.

Figura 49. Modelo 3 - Cenário 1: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Os resultados obtidos com a aplicação do terceiro modelo FDD no **segundo cenário** de teste foram similares aos resultados do primeiro cenário de teste, apresentados anteriormente. O aumento da quantidade de instâncias de dados não rotuladas não afetou a capacidade do detector *ensemble* de falhas de identificar corretamente novas instâncias falhas. Visto que, no segundo cenário, o detector de falhas também obteve sensibilidade máxima na identificação dessas instâncias no processo de detecção de falhas. Porém, o mesmo gerou um número maior de falsos positivos (observe as *Figura 47* e *Figura 50*).

Figura 50. Modelo 3 – Cenário 2: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

A *Figura 51* apresenta a matriz de confusão do detector *ensemble* de defeitos na tarefa de identificação dos mesmos. Observe que, no cenário em questão, o detector gerou um menor número de falsos positivos e, conseqüentemente, obteve uma melhor precisão nessa etapa em relação ao primeiro cenário de teste. Além disso, o ganho de sensibilidade do detector não foi tão significativo quanto o ganho de precisão, como é possível perceber pela comparação das duas primeiras linhas da *Tabela 11* na coluna referente a detecção dos defeitos. Essa diferença de aprimoramento de desempenho também pode ser constatada por meio da análise das *Figura 48* e *Figura 51*. Note que, a quantidade de falsos positivos foi reduzida em 45 (quarenta e cinco) instâncias. Por outro lado, a quantidade de falsos negativos foi reduzida em apenas duas instâncias.

Figura 51. Modelo 3 – Cenário 2: Matriz de confusão da etapa de detecção dos defeitos (**Fonte:** Autor).

No entanto, na tarefa de diagnóstico dos defeitos, não foi observado nenhum ganho de desempenho em relação a taxa de acertos do classificador, sem considerar a taxa de acertos por classe. Essa diferença de desempenho pode ser percebida por intermédio tanto da análise da

matriz de confusão do classificador *ensemble* (veja as *Figura 49* e *Figura 52*) quanto das métricas de desempenho, mostradas na terceira linha da *Tabela 11*.

Figura 52. Modelo 3 – Cenário 2: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Ao se considerar a taxa de acertos por classe, é possível perceber que essa diferença de performance se deve ao fato de, no segundo cenário de teste, o classificador *ensemble* ter classificado corretamente algumas instâncias da classe *Corrente Baixa e Oscilando*. Do mesmo modo que classificou menos instâncias da classe *Problema na Fase 2*. Todavia, classificou com êxito algumas instâncias como pertencentes a classe *Sobrecorrente no Des travamento*, mas isso não foi suficiente para igualar ou superar o desempenho obtido no primeiro cenário de teste.

Apesar do desempenho inferior, a eficiência sobre as classes *Atrito*, *Problema de Medição* e *Problema na Fase 2* não foi afetada significativamente. Por fim, tanto no primeiro quanto no segundo cenário, o classificador *ensemble* de defeitos não conseguiu êxito sobre as classes *Sobrecorrente no Travamento*, *Múltiplos Acionamentos Simultâneos*, *Atraso na Partida* e *Desligamento do Motor*.

Assim como no segundo cenário de teste, no **terceiro cenário** – constituído por todas as instâncias de dados não rotuladas do conjunto original –, a significativa ampliação da

quantidade de dados utilizada no desenvolvimento do método FDD não prejudicou a competência do detector *ensemble* de falhas em identificar instâncias falhas. Como mostrado na *Figura 53*, todas essas instâncias foram identificadas com êxito. Porém, novamente, o detector gerou uma quantidade considerável de falsos positivos, sendo a maior entre todos os cenários de teste.

Figura 53. Modelo 3 - Cenário 3: Matriz de confusão da etapa de detecção de falhas (**Fonte:** Autor).

Na detecção dos defeitos, diferentemente do segundo cenário, o detector de defeitos apresentou, de forma significativa, tanto um ganho de sensibilidade quanto uma perda de precisão no terceiro caso de teste. Note que ocorreu uma inversão no ganho de desempenho se comparado ao segundo cenário. Conforme ilustra a *Figura 54*, ao todo 165 (cento e sessenta e cinco) instâncias foram classificadas incorretamente. Dessas instâncias, apenas 54 (cinquenta e quatro) foram identificadas de forma incorreta como normais – falsos negativos–, mas 111 (cento e onze) foram identificadas incorretamente como defeituosas – falsos positivos–. Em razão disso, no terceiro cenário, o detector *ensemble* de defeito obteve a melhor sensibilidade e pior precisão no processo de detecção dos defeitos, considerando todos os resultados obtidos com a aplicação do terceiro método em todos os cenários de teste.

Figura 54. Modelo 3 - Cenário 3: Matriz de confusão da etapa de detecção dos defeitos (**Fonte:** Autor).

No diagnóstico dos defeitos (veja a *Figura 55*), se comparado aos resultados do terceiro modelo no primeiro e segundo cenário de teste, o classificador *ensemble* obteve a melhor taxa de acertos para as classes *Atrito* e *Problema na Fase 2*. Para a classe *Problema de Medição*, uma das três classes cujo o classificador alcançou uma porcentagem de acertos regular, a taxa de acertos foi igual. Esse classificador conseguiu classificar corretamente pelo menos uma instância de dados das classes *Atraso na Partida*, *Múltiplos Acionamentos Simultâneos* e *Sobrecorrente no Destravamento*. Porém, apresentou uma redução na taxa de acerto sobre a classe *Corrente Baixa e Oscilando*. Além disso, o desempenho insatisfatório sobre as classes *Desligamento do Motor* e *Sobrecorrente no Travamento* permaneceu. Apesar do bom desempenho sobre algumas classes, o classificador *ensemble* de defeitos obteve o seu pior desempenho, como é possível observar na *Tabela 11*, na coluna referente ao diagnóstico dos defeitos.

Figura 55. Modelo 3 - Cenário 3: Matriz de confusão da etapa de diagnóstico dos defeitos (Fonte: Autor).

Por fim, a *Tabela 11* sumariza os resultados do terceiro modelo FDD em todos os casos de teste. Nesta tabela, assim como nas anteriores – *Tabela 9* e *Tabela 10* –, as linhas equivalem aos cenários de teste e as colunas refletem tanto a tarefa na qual o modelo FDD foi avaliado quanto as métricas de desempenho dos classificadores e/ou detectores *ensembles* na tarefa

efetuada. Baseando-se tanto nos resultados mostrados na tabela quanto na discussão evidenciada, pode-se inferir sobre o terceiro modelo que:

1. No procedimento de identificação de falhas, o detector *ensemble* de falhas manteve a excelência de sensibilidade na identificação de novas falhas em todos os cenários de teste, apesar da quantidade de falsos positivos gerados.
2. Nos três casos de teste, na tarefa de detecção dos defeitos, o desempenho do detector *ensemble* foi satisfatório. Sendo que esse detector alcançou o melhor desempenho, com relação a sensibilidade e precisão, quando aplicado sobre os dados do terceiro e segundo cenário de teste, respectivamente. De fato, como apresentado anteriormente, conforme à aplicação desse modelo sobre os cenários de teste a quantidade de falsos negativos –gerados pelo detector– diminuiu gradualmente. Já o número de falsos positivos oscilou consideravelmente, alcançando a maior quantidade no terceiro caso de teste.

Desta forma, os detectores de defeitos desse modelo FDD passaram a ser mais eficientes na detecção de novas instâncias defeituosas do que na detecção de instâncias normais. Por conta disso, foi observado que, no segundo e terceiro cenário, o detector apresentou um ganho de sensibilidade. Por outro lado, no terceiro cenário, mostrou uma perda de precisão. Essa situação não aconteceu nos outros cenários de teste.

3. O classificador *ensemble* na tarefa de diagnóstico dos defeitos, no terceiro caso de teste, obteve um desempenho inferior em relação as performances de sua aplicação nos cenários iniciais de teste. Por isso, pode-se dizer que, no cenário mencionado, esse classificou obteve o seu pior desempenho nessa tarefa. A melhor performance desse classificador ocorreu quando o mesmo foi aplicado sobre os dados do primeiro cenário de teste. De modo geral, o desempenho do classificador *ensemble* de defeitos foi satisfatório, apesar de não ter apresentado uma boa taxa de acertos sobre todos os tipos de defeitos –classe analisadas– em todos os cenários de teste.

Tabela 11. Desempenho do terceiro modelo FDD, referente ao método 3 (combinação das técnicas Autoencoder e SOM), nos três cenários de teste (**Fonte:** Autor).

Cenário de Teste	Métricas de Desempenho no processo de Detecção e Diagnóstico de Defeitos							
	Detecção de Defeitos				Diagnóstico de Defeitos			
	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Acurácia</i>	<i>Média-F1</i>	<i>Acurácia</i>	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Média-F1</i>
Cenário 1	92.24%	91.67%	88.05%	91.96%	78.05%	71.79%	78.05%	72.05%
Cenário 2	95.59%	91.83%	90.75%	93.67%	76.96%	70.51%	76.96%	71.21%
Cenário 3	91.80%	95.84%	90.52%	93.78%	75.86%	75.18%	75.86%	70.15%

5.2.4.4 Avaliação Geral dos Resultados

Nas subseções anteriores os resultados dos modelos de detecção e diagnóstico do estado de operacionalidade do AMV foram discutidos. Inicialmente, de forma individual, argumentou-se sobre o desempenho desses modelos em cada um dos três cenários de teste analisados neste trabalho científico. Nesta subseção, apresenta-se uma visão geral desses resultados, principalmente em relação a eficiência e robustez desses modelos nas tarefas de detecção e diagnóstico do estado de operacionalidade do AMV. Ao final desta subseção mostra-se o *ranking* desses modelos FDDs, criado com base em seus desempenhos em cada tarefa de classificação, propostos neste trabalho.

Conforme explicado anteriormente, a detecção do estado de operacional do AMV é realizada em duas etapas. Devido a isso, embora não influencie diretamente no cálculo de desempenho do modelo FDD, a identificação correta das instâncias falhas – casos em que o AMV se encontra indisponível – é fundamental no contexto do processo apresentado. Sabendo disso, o desempenho do detector *ensemble* de falhas foi satisfatório na maioria dos cenários de teste.

Em todos esses casos de teste, os classificadores-base do detector de falhas foram treinados com os dados gerados pela aplicação do Autoencoder sobre o conjunto de dados não rotulado, visto que essa foi a técnica escolhida para esse procedimento – por conta do seu ótimo desempenho nessa etapa. Por conta disso, não existe uma variação significativa de performance dos modelos FDDs nessa tarefa. Em outras palavras, tem-se apenas diversas aplicações dessa técnica em contextos diferentes.

De fato, apenas em um caso de teste esse detector apresentou uma redução de sensibilidade na identificação de instâncias falhas, caso no qual teve-se a ocorrência de falsos negativos. Isto é, instâncias identificadas incorretamente como falhas. Todavia, a identificação incorreta de algumas instâncias como normais foi um evento recorrente desse detector.

Posteriormente a identificação dos casos de indisponibilidade do AMV, tem-se a identificação das instâncias que estão operacionais sem degradação (normais) e com degradação (defeitos). De modo geral, independentemente do modelo FDD ou cenário de teste, o detector *ensemble* de defeitos foi eficiente na identificação correta dessas situações. Inclusive, em todos os cenários e em cada modelo FDD, esse detector obteve pelos menos 91% de precisão e sensibilidade. Isto é, conseguiu identificar corretamente pelos menos 91% das instâncias de cada classe, apesar da existência de diversos casos de inconsistência de classificação no conjunto de dados de teste (alguns desses casos serão exibidos na próxima seção).

Além disso, foi observado que existe uma relação entre precisão e sensibilidade do detector *ensemble* de defeitos. Nos cenários de teste analisados, quando a precisão desse detector aumenta, a sua sensibilidade diminui. Por exemplo, em todos os modelos FDD nos dois primeiros casos de teste, o detector foi mais eficiente na identificação de instâncias normais – isto é, a precisão excede a sensibilidade –. Do mesmo modo que no terceiro caso de teste, o detector foi mais eficiente na identificação de instâncias defeituosas – ou seja, a sensibilidade foi superior a precisão –.

De modo geral, independentemente dessa relação, **o Autoencoder foi a melhor técnica não supervisionada, dentre as utilizadas neste trabalho, no processo não supervisionado de detecção dos defeitos (etapa de agrupamento dos dados)**. De fato, em todos os modelos FDD avaliados, o detector *ensemble* de defeitos alcança a melhor performance quando utiliza no treinamento dos seus classificadores-base o conjunto de dados resultante da aplicação dessa técnica na etapa de agrupamento dos dados.

Por fim, **a rede SOM foi a melhor técnica não supervisionada na tarefa de agrupamento de dados da etapa de diagnóstico dos defeitos**. Como foi evidenciado nos cenários de teste, o classificador *ensemble* dessa etapa alcança o seu melhor desempenho ao usar no treinamento dos seus classificadores-base o conjunto de dados obtido pela aplicação da rede SOM sobre as instâncias defeituosas.

No que diz respeito a eficiência do classificador, os resultados obtidos no processo de especificação dos defeitos foram inferiores aos resultados da detecção. Dado que a maior taxa de acertos dessa etapa não atingiu 80% de classificações corretas das instâncias do conjunto de dados de teste. Todavia, além das inconsistências de classificação, foi observado que pode existir mais de uma causa (defeito) para o comportamento defeituoso de uma instância. Isto posto, o classificador *ensemble* conseguiu identificar a presença de mais de um tipo de defeito em diversas instâncias (algum desses casos serão apresentados na próxima seção). Ademais, em alguns desses casos, a classe definida pelo classificador difere da classe especificada pela Vale S.A, o que afeta diretamente no desempenho.

A seguir, considerando todos os modelos FDD e cenários de teste, apresenta-se uma relação da taxa de acertos (desempenho) do classificador *ensemble* para cada tipo de defeito:

1. **Atrito:** tipo de defeito em que a taxa de acertos do classificador foi alta, independentemente do modelo e/ou cenário de teste. Além disso, diversas instâncias especificadas previamente como sendo de outras classes foram classificadas como atrito. Isto é, o classificador definiu que, além do comportamento original, essas instâncias possuem também esse tipo de defeito.
2. **Atraso na Partida:** a taxa de acertos do classificador para este tipo de defeito se resume a poucas instâncias diagnosticadas corretamente nas condições do terceiro cenário de teste.
3. **Corrente Baixa e Oscilando:** o desempenho do classificador *ensemble* oscilou bastante para este tipo de defeito, principalmente no contexto do terceiro modelo. No qual, diferentemente dos outros cenários, teve o menor percentual de acertos do classificador para essa causa.
4. **Desligamento do Motor:** o classificador *ensemble* foi relativamente ineficiente para este tipo de defeito. Visto que conseguiu identificar corretamente apenas uma instância entre três possíveis do conjunto de dados de teste. Todavia, essa causa não possui uma quantidade enorme de instâncias no conjunto de dados de treinamento.
5. **Múltiplos Acionamentos Simultâneos:** Assim como para a causa atraso na partida, o classificador *ensemble* conseguiu diagnosticar corretamente apenas algumas instâncias que apresentam esse comportamento. Entretanto, isso deu-se praticamente em todos os cenários de teste. Baseando-se nos resultados, existe

uma entre os comportamentos das classes Atrito Múltiplos Acionamentos Simultâneos.

6. **Problema de Medição:** a taxa de acerto do classificador *ensemble* neste tipo de defeito excede as demais. De fato, em todos os modelos FDD dos cenários analisados, esse classificador conseguiu diagnosticar corretamente pelo menos 98% das instâncias dessa causa nos dados de teste.
7. **Problema na Fase 2:** essa causa possui a terceira maior taxa de acertos do classificador no processo de especificação dos defeitos. Inclusive, pelo menos 73% das instâncias defeituosas foram corretamente diagnosticadas como pertencentes a essa classe em todos os cenários de teste.
8. **Sobrecorrente no Destravamento:** neste tipo de defeito, assim como no caso de atraso na partida, o classificador foi ineficiente no diagnóstico das instâncias que apresentam o comportamento defeituoso representado por esta causa. Sendo que a maior taxa de acerto para este defeito não passa de 10%.
9. **Sobrecorrente no Travamento:** Do mesmo modo que para a causa *Desligamento do Motor*, para este defeito o classificador foi ineficiente. Inclusive, em quase todos os casos, o classificador *ensemble* não conseguiu diagnosticar de modo correto nenhuma instância defeituosa, cujo o comportamento é representado por esta causa.

O *ranking* dos modelos FDD foi gerado por intermédio do desempenho geral de cada modelo, calculado de acordo com a fórmula proposta na terceira etapa da subseção 4.1.2 – seção 4.1 do Capítulo 4. No cálculo do desempenho de cada modelo, devido aos diversos casos de inconsistência e descoberta de novas causas, considerou-se o mesmo valor de peso para cada etapa do processo de detecção e diagnóstico do estado de operacionalidade do AMV.

Neste caso, o desempenho tanto do detector quando do classificador dos defeitos representam 50% (p_1 e p_2 iguais a 0,5) do desempenho total do modelo FDD. Para obter as métricas que compõem esse ranking, considerou-se as métricas parciais do terceiro caso de teste. Dado que esse cenário contempla todo o conjunto de dados não rotulado. Logo, as métricas parciais de desempenho são combinadas baseando-se na fórmula apresentada anteriormente. Por exemplo, a métrica acurácia geral do primeiro modelo é calculada como segue:

$$\begin{aligned}
 \text{acuraciaGeral} &= p1 \times mDet + p2 \times mDig \Rightarrow 0,5 \times 89,78 + 0,5 \times 75,32 \\
 &\Rightarrow 44,89 + 37,66 = 82,55
 \end{aligned}$$

em que, *mDet* e *mDig* representam, respectivamente, a acurácia do detector *ensemble* de defeitos e a acurácia do classificador *ensemble* dos defeitos nas etapas de detecção e diagnóstico.

Finalmente, a *Tabela 12* apresenta o ranking dos modelos FDD. Nessa tabela as colunas correspondem as métricas de desempenho e as linhas aos modelos. Esses modelos FDD foram expostos na tabela em ordem decrescente de desempenho geral. Ou seja, o melhor modelo se encontra na primeira linha. Para a definição do melhor modelo avaliou-se tanto a acurácia quanto a sensibilidade geral de cada modelo. Uma vez que essas métricas, no contexto deste trabalho, representam, respectivamente, a eficiência do classificador *ensemble* na especificação dos defeitos e a sensibilidade do modelo em identificar novas instâncias defeituosas. Portanto, como é possível observar na tabela, o terceiro modelo FDD foi o melhor modelo dentre os analisados neste trabalho científico. Logo, a combinação das técnicas Autoencoder e SOM, que constituem o terceiro método proposto (descrito na subseção 4.1.1 do Capítulo 4), possibilita obter um bom desempenho no processo de detecção e diagnóstico do estado de operacionalidade do AMV.

Tabela 12. Desempenho geral dos modelos de detecção e diagnóstico do estado de operacionalidade do AMV (Fonte: Autor).

Modelos de Detecção e Diagnóstico de Defeitos - FDDs	Métricas de Desempenho gerais no processo de Detecção e Diagnóstico de Defeitos			
	<i>Acurácia</i>	<i>Precisão</i>	<i>Sensibilidade</i>	<i>Média- F1</i>
M3 - Autoencoder & SOM	83.19%	83.49%	85.85%	81.96%
M1 - K-means & SOM	82.55%	80.70%	85.31%	81.58%
M2 - Autoencoder & K-means	82.28%	82.73%	85.01%	81.64%

5.2.4.5 Resultados emergentes

Esta seção apresenta os resultados emergentes da aplicação da metodologia proposta – apresentados anteriormente– desta pesquisa de investigação científica. Isto é, resultados não esperados, mas que foram observados. Primeiramente, aborda-se o problema de inconsistência

de classificação, o qual pode ocorrer tanto na detecção quanto no diagnóstico. Posteriormente, discute-se sobre a descoberta de novas causas.

5.2.4.5.1 *Inconsistências de Classificação*

Na tarefa de detecção de defeitos, posteriormente a identificação que o ativo está operacional, tem-se a atribuição da instância de dados a uma das classes pré-definidas previamente que representam o estado operacional do AMV, a saber, Normal (operacional sem degradação) ou Defeito (operacional, com degradação), de acordo com o rótulo predito para a mesma pelo detector *ensemble* de defeitos. Essa atribuição ocorre durante o procedimento de classificação de uma nova instância de dados.

No contexto de uma tarefa de classificação que objetiva a avaliação de desempenho de um classificador, por meio dos dados de teste, é possível analisar os erros de atribuição (ou classificação) desse classificador nessa tarefa. Geralmente, esses erros são apresentados em uma matriz de confusão e categorizados em falsos positivos ou negativos. Isto posto, esta subseção apresenta uma análise sobre os erros do detector *ensemble* de defeitos no processo de detecção e diagnóstico de falhas, proposto neste trabalho. Vale salientar que essa análise surgiu como um aditivo, resultante das observações das atividades da metodologia proposta. Por isso, se encontra fora do contexto de aplicação da mesma.

Os erros de classificação analisados nesta etapa provêm do detector *ensemble* de defeitos do modelo com a melhor performance nessa tarefa, em relação a todos os cenários de teste descritos anteriormente. Neste caso, selecionou-se o terceiro modelo (referente ao método composto pela combinação das técnicas Autoencoder e SOM) do segundo cenário de teste. Pois, como mostrado previamente, foi o modelo com o melhor desempenho no contexto deste trabalho.

Com relação aos erros de classificação analisados, de acordo com a *Figura 51*, 55 instâncias foram incorretamente classificadas como defeito (falsos positivos). Do mesmo modo que 106 instâncias foram incorretamente classificadas como normal (falsos negativos). Sabendo disso, a análise dessas instâncias foi realizada por intermédio de um processo de duas etapas:

- (i.) *plotagem da assinatura elétrica da Máquina de Chave – MCH referente a instância*, por meio dos dados de leitura da corrente elétrica da MCH – esses dados também foram

disponibilizados pela Vale, mas não relacionam-se diretamente com o conjunto de dados utilizado neste trabalho – gera-se um gráfico (semelhante à *Figura 24*), o qual representa o comportamento dessa corrente durante a configuração do AMV, e;

- (ii.) *avaliação do especialista de domínio*, o especialista de domínio analisa o comportamento da instância e atribui a mesma uma das classes mencionadas (*Normal* ou *Defeito*).

Esse procedimento é realizado individualmente para cada instância de dados. De modo a otimizar o tempo gasto com a análise de cada instância, previamente, selecionou-se algumas instâncias que, provavelmente, podem ser compreendidas como inconsistências de classificação. Neste contexto, entende-se como uma inconsistência de classificação o caso em que instâncias que possuem comportamentos semelhantes e/ou idênticos foram atribuídas a classes diferentes por um especialista do domínio de aplicação.

É importante mencionar que o conjunto de dados de teste foi manualmente rotulado por um especialista de domínio. No entanto, de forma não simultânea, pelo menos dois especialistas foram responsáveis pela classificação desses dados. Isso possibilita tanto a existência das inconsistências quanto eleva a subjetividade desse processo, uma vez que o especialista de domínio efetua essa classificação baseando-se na sua experiência em campo. Dito isto, posteriormente a separação dos prováveis casos de inconsistência, tem-se a execução do processo de duas etapas mencionado anteriormente.

A *Figura 56* apresenta a assinatura elétrica de uma instância de dados detectada incorretamente como defeito, conseqüentemente, um falso positivo. Neste caso, o detector *ensemble* de defeitos atribuiu essa instância à classe defeito. Contudo, o rótulo definido pela Vale, nos dados de teste, é normal (“*Sem Defeito*”). Baseando-se nisso, pode-se assumir que o detector *ensemble* cometeu um erro.

No entanto, conforme a *Figura 57* e a avaliação do especialista do domínio, o caso de falso positivo ilustrado na *Figura 56* pode ser considerado como uma inconsistência de classificação, que corresponde a um erro de classificação do detector *ensemble* e que pode refletir um problema de integridade do conjunto de dados de teste. De fato, diversas instâncias classificadas incorretamente como normais – falsos negativos –, apresentam um

comportamento semelhante a essa instância. Pois não existe uma diferença significativa entre as assinaturas dessas MCHs.

Figura 56. Assinatura elétrica de uma instância de dados detectada incorretamente como defeito pelo detector *ensemble* (Fonte: Autor).

Figura 57. Assinaturas elétricas de instâncias de dados detectadas incorretamente como normal (“Sem Defeito”) (Fonte: Autor).

Outros exemplos de inconsistências de classificação podem ser observados nas *Figura 58* e *Figura 59*. Assim como no caso anterior, essas instâncias possuem o mesmo fabricante, local de instalação e são referentes a mesma MCH. Porém, são eventos distintos. Em casos como apresentados aqui, é difícil identificar um ponto de distinção entre as instâncias até mesmo para o especialista de domínio. Pois, devido a classificação com base na experiência –

a qual pode ser prejudicada por extensos períodos de trabalho—, ao avaliar essas assinaturas com dois especialistas de domínio, de forma individual, observou-se uma divergência em suas respostas. Vale ressaltar que não teve qualquer interação entre os especialistas. Portanto,

Figura 58. Exemplos de inconsistências de classificação com um cenário idêntico, em relação a máquina de chave, local de instalação (house) e configuração de movimento, de acordo com o segundo fabricante (**Fonte:** Autor).

Figura 59. Exemplos de inconsistências de classificação com um cenário idêntico de acordo com o primeiro fabricante (**Fonte:** Autor).

nenhuma informação foi repassada ao segundo especialista sobre a avaliação do primeiro e vice-versa.

Ademais, foi observado que, em alguns casos, eventos com configuração de movimento diferentes podem ser considerados como uma inconsistência de classificação. A *Figura 60* ilustra essa situação. Na figura, a primeira instância (localizada na parte superior) foi

classificada por um especialista de domínio da Vale como defeito. Essa instância representa um movimento de configuração do AMV de normal para reverso. As demais instâncias (localizadas na parte inferior) possuem um comportamento relativamente similar a primeira, porém, representam movimentos de configuração de reverso para normal. De fato, conforme apresentado na seção 2.1 do capítulo 2, existe uma distinção no contexto ferroviário entre esses movimentos de configuração. No entanto, em relação ao detector *ensemble* de defeitos, o movimento de configuração não influencia na capacidade do detector em identificar corretamente o estado operacional de uma nova instância. Por conta disso, podem ser consideradas como uma inconsistência de classificação.

Figura 60. Exemplos de inconsistências de classificação com configurações de movimento diferentes (Fonte: Autor).

Finalmente, baseando-se nos pontos discutidos nesta subseção, pode-se concluir que, em alguns casos, o detector *ensemble* de defeitos não necessariamente comete um erro ao detectar o estado operacional de uma nova instância de dados como normal ou defeituoso, visto que existem inconsistências de classificação no conjunto de dados de teste. Dito isso, de modo a solucionar esse problema, pode-se:

1. Avaliar todos os casos de inconsistência do conjunto de dados de teste. Para isto, faz-se necessário que apenas um especialista de domínio realize a

reclassificação. Visto que isso diminuiria a subjetividade dessa tarefa. Todavia, trata-se de uma tarefa que demanda muito tempo e, normalmente no contexto de uma ferrovia, não é possível delegar um especialista por tanto tempo em uma mesma atividade.

Posteriormente ao procedimento de reclassificação das inconsistências dos dados de teste, pode-se aplicar novamente o detector *ensemble* de defeitos com a finalidade de analisar o desempenho do mesmo em um cenário livre de inconsistências. É plausível considerar que, até atingir um estado livre de inconsistências, diversas iterações de análise e reclassificação de inconsistências serão realizadas.

Desse modo, tem-se tanto o refinamento do detector *ensemble* quanto dos dados de teste. Vale salientar que esse processo de refinamento não foi realizado neste trabalho, pois, além de demandar a participação continua do especialista, exige um elevado nível de conhecimento de domínio. Uma vez que a reclassificação é fundamental durante o refinamento.

2. Excluir todos os casos de inconsistências. Ao contrário do item anterior, neste caso, remove-se dos dados de teste todas as instâncias que podem ser consideradas como uma inconsistência de classificação. Note que, embora a quantidade de instâncias de teste seja reduzida, o novo conjunto de dados será livre de inconsistências (assim como no item anterior).

5.2.4.5.2 *Diagnóstico de Defeitos: identificação de novas causas*

Assim como no caso da detecção do estado operacional do AMV, no diagnóstico de operacionalidade desse ativo existe a possibilidade de ocorrência de erros de classificação, visto que, dada a metodologia proposta, realiza-se o diagnóstico por meio de uma tarefa de classificação multiclasse. Devido a isso, do mesmo modo da detecção, faz-se necessário analisar as instâncias classificadas incorretamente – novamente, esse tipo de análise não se encontra no contexto da metodologia –.

Com este propósito, utiliza-se o processo de duas etapas explicado anteriormente. Todavia, ao invés de avaliar somente as inconsistências, também objetiva-se definir se a instância possui somente um tipo de defeito. Portanto, nesta subseção mostra-se alguns

resultados dessa análise. Primeiramente, é factível introduzir a hipótese de uma instância apresentar mais de um tipo de defeito. Neste trabalho, utiliza-se o termo *descoberta de uma nova causa* para representar os casos em que o classificador *ensemble* descobre esses comportamentos.

Note que na etapa de avaliação de desempenho dos modelos FDDs, em nenhum momento, abordou-se esse termo. De fato, a princípio, considerou-se apenas que o classificador *ensemble* não conseguiu obter uma boa generalização de todas as classes analisadas e, por consequência, não obteve um bom desempenho geral sobre essas classes. No entanto, após uma avaliação em conjunto com o especialista de domínio, constatou-se a possibilidade de uma instância apresentar vários tipos de defeito.

Desta forma, durante o diagnóstico, caso o rótulo predito pelo classificador *ensemble* seja diferente do rótulo definido – pelo especialista – no conjunto de dados de teste tem-se duas situações possíveis, a saber:

- (i.) *erro de classificação*, não existe correlação entre o comportamento da instância de dados (verificado por intermédio da plotagem do gráfico da assinatura da MCH) e a classe predita, e;
- (ii.) *descoberta de uma nova causa*, existe correlação entre comportamentos, isto é, a instância de dados possui o comportamento descrito pela classe predita.

A *Figura 61* ilustra dois casos da segunda situação, ou seja, de descoberta de um novo tipo (causa) de defeito. Observe que em ambas instâncias, além de possuírem comportamentos similares, a classe predita pelo classificador *ensemble* difere da classe definida – pela Vale – no conjunto de dados de teste. Conforme a classificação da Vale S.A, essas instâncias pertencem, respectivamente, à classe *Atrito* – atrito ao movimentar o conjunto de frente – e *Problema na Fase 2* – variação da corrente durante a movimentação livre da barra de operação da MCH –.

A classe predita pelo classificador *ensemble* para a primeira instância (lado esquerdo da figura) foi Problema na Fase 2. No caso da segunda instância (lado direito da figura), a classe predita foi atrito. Após a avaliação em conjunto com um especialista do domínio, concluiu-se que essas instâncias, de fato, além dos comportamentos especificados pela Vale S.A,

apresentam os comportamentos preditos pelo classificador *ensemble*. Ou seja, pelo menos um comportamento foi diagnosticado corretamente pelo classificador, embora não seja o especificado pela Vale S.A.

Figura 61. Instâncias de dados com mais de um tipo de defeito. Sendo que, a esquerda, tem-se um exemplo de comportamento da classe *Atrito*. Enquanto que, a direita, tem-se um exemplo de comportamento da classe *Problema na Fase 2* (**Fonte:** Autor).

A primeira situação, isto é, um provável erro de classificação ocorre em casos como os mostrados na *Figura 62*. Entretanto, percebe-se que, neste caso, pode-se considerar como uma inconsistência de classificação. Pois ambas instâncias deveriam ser consideradas como um problema de medição – como o próprio nome indica, a leitura dos dados da via não foi realizada corretamente, consequentemente, os dados gerados a partir desses casos estão, de certa forma, incoerentes –.

Figura 62. Exemplos de inconsistências e/ou erros de classificação da etapa de diagnóstico dos defeitos (**Fonte:** Autor).

Portanto, no caso exposto, o classificador acertou ao diagnosticar a primeira instância (lado esquerdo da *Figura 62*) como um problema de medição. Assim como cometeu um erro ao diagnosticar a segunda instância (lado direito da *Figura 62*) como atrito. Uma possível razão para o erro de classificação do segundo caso é o fato de o algoritmo de interpretação de clusters extrair as regras de domínio, de cada classe, a partir do conjunto de dados manualmente rotulado

pelo especialista (também chamado de conjunto de dados independente). Logo, caso exista inconsistência de classificação entre os diversos tipos de defeito, essas inconsistências podem se propagar para as regras de domínio e, por consequência, tem-se os erros de classificação como o caso ilustrado na figura.

No entanto, existem erros de classificação que não podem ser considerados como inconsistências, nesses casos diz-se que o classificador *ensemble*, de fato, cometeu erro. Como pode ser observado no exemplo mostrado na *Figura 63*, a classe definida para a instância é atrito, porém, o classificador atribuiu a mesma à classe *Corrente Baixa e Oscilando* – valores de corrente da MCH abaixo do esperado com oscilações –. Diferentemente dos casos apresentados anteriormente, essa instância não apresenta o comportamento predito pelo classificador.

Figura 63. Exemplo de um erro de classificação na etapa de diagnóstico dos defeitos (**Fonte: Autor**).

De fato, por meio da comparação das *Figura 64* e *Figura 63*, pode-se concluir que o classificador errou ao predizer a classe corrente baixa e oscilando para a instância ilustrada na *Figura 63*. Além disso, a classe predita para a instância da *Figura 64* foi *Atrito*, note que existe uma variação de corrente durante a movimentação do conjunto de frente (Fase 3 do movimento de configuração do AMV, veja a *Figura 24*), que pode ser considerada como *Sobrecorrente no Destravamento* quanto atrito. Essa variação pode ser causada por um atrito entre algum objeto, tal como uma pedra, e os roletes que auxiliam na movimentação das agulhas do AMV.

Em casos como esse, definir se o comportamento corresponde ou não a uma determinada classe pode ser difícil até mesmo para o especialista de domínio. Visto que o comportamento da instância pode ser representado por mais de uma causa. Por exemplo, classificá-la como *Sobrecorrente no Destravamento* não seria incorreto, pois apresenta um pico de corrente na

transição da Fase 2 para a Fase 3 – característica principal dessa classe –. Do mesmo modo que classificá-la como atrito também não estaria incorreto, neste caso, depende do padrão definido como normal pelo especialista. Uma vez que essa instância poderia estar dentro do limite estabelecido para se considerar uma instância de dados como operacional e sem degradação, isto é, normal. Por fim, o comportamento mais nítido na instância é a corrente baixa e oscilando.

Figura 64. Comportamento de uma instância de dados da classe *Corrente Baixa e Oscilando* (Fonte: Autor).

Portanto, a mesma apresenta todos os três tipos de defeitos mencionados.

De modo a ilustrar o comportamento de todas as classes analisadas, a seguir, apresenta-se outros casos de descoberta de novas causas. A *Figura 65* apresenta um caso típico da classe *Múltiplos Acionamentos Simultâneos*. Esse tipo de defeito representa um comportamento da máquina de chave no qual ela é acionada diversas vezes durante a configuração do AMV. Como é mostrado na figura, é como se a máquina fosse ligada mais de uma vez. Ainda conforme a *Figura 65*, como especificado pelo classificador *ensemble* neste caso, essa instância possui pelo menos mais um tipo de defeito, a saber, corrente baixa e oscilando.

Figura 65. Comportamento de uma instância de dados da classe *Múltiplos Acionamentos Simultâneos* (Fonte: Autor).

As *Figura 66*, *Figura 67* e *Figura 68* apresentam casos de descoberta de novos tipos de defeitos envolvendo as demais classes, respectivamente, *Atraso na partida*, *Desligamento do motor* e *Sobrecorrente no Travamento*. Em um acontecimento de atraso na partida, como pode ser observado na *Figura 66*, tem-se uma variação de tempo, em relação ao tempo comum de início da máquina, para ligar o motor da máquina de chave – MCH. Por exemplo, no caso mencionado houve uma demora de quase três segundos. Logo, pode-se dizer que ocorreu um atraso na partida da MCH.

Figura 66. Comportamento de uma instância da classe *Atraso na Partida* (**Fonte:** Autor).

No caso do desligamento do motor, durante a movimentação da barra de operação da MCH para realizar a configuração do AMV, tem-se uma queda de corrente provocada pelo desligamento do motor. Todavia, em questões de milissegundos, o motor é liga novamente e o evento é concluído. Note que em ambas instâncias, ilustradas nas *Figura 66* e *Figura 67*, apresentam problemas na segunda fase.

Figura 67. Comportamento de uma instância de dados da classe *Desligamento do motor* (**Fonte:** Autor).

Finalmente, no tocante ao comportamento, a classe sobrecorrente no destravamento e travamento são parecidas. Porém, a sobrecorrente no travamento é caracterizada por um pico

de corrente no final da movimentação do conjunto de frente. Consequentemente, apesar de tratarem de um problema de sobrecorrente, essas classes são distintas e representam situações diferentes.

Figura 68. Comportamento de uma instância de dados da classe *Sobrecorrente no Travamento* (Fonte: Autor).

Baseando-se nos resultados apresentados, pode-se concluir que o classificador *ensemble* da etapa de diagnóstico dos defeitos não necessariamente comete um erro ao diagnosticar uma nova instância de dados. Em alguns casos, o classificador descobre a existência de pelo menos mais um tipo de defeito cujo o comportamento corresponde ao dessa instância. Além disso, pode acontecer uma inconsistência de classificação. Por último, tem-se o erro de classificação, o qual é resultante de um equívoco do classificador *ensemble*.

Isto posto, de forma a eliminar as inconsistências e lidar com os casos de descoberta de novas causas é factível:

1. Excluir ou avaliar todas as inconsistências, assim como na etapa de detecção do estado operacional do AMV.
2. Considerar um processo de classificação por camadas. Inicialmente, avaliam-se os casos mais críticos, tais como Atrito e Sobrecorrente. Posteriormente, os demais casos são avaliados. A classificação em camadas permite que após a última camada de classificação tenha-se a relação dos defeitos apresentados pela instância. Observe que a capacidade de inferir a criticidade de uma instância não é afetada.

6 CONSIDERAÇÕES FINAIS

Este capítulo apresenta as conclusões deste trabalho de investigação científica. Baseando-se nos resultados e argumentações, apresenta-se algumas conclusões sobre os modelos de detecção e diagnóstico do estado de operabilidade de um sistema/equipamento propostos. Para isto, discutisse os resultados obtidos com os modelos FDD, resultantes da implementação desses métodos, incluindo a metodologia utilizada para a obtenção dos mesmos.

A aplicação da manutenção baseada em condição possibilita realizar a manutenção de forma mais eficiente e com viés de segurança e produtividade operacional. Sabendo disso, o SOS:D² teve como objetivo a proposição de um conjunto de métodos para a detecção e diagnóstico do estado de operacionalidade (degradação) de equipamentos ferroviários, especificamente, aparelhos de mudança de via – AMV.

Para isto, apresentou-se a estrutura desenvolvida para o processo de desenvolvimento e avaliação dos modelos de Detecção e Diagnóstico de Falhas – FDD. Em seguida, abordou-se detalhadamente a metodologia de desenvolvimento dos métodos FDDs. A execução dessa metodologia resulta nos modelos FDDs, uma vez que um modelo FDD é o resultado obtido com o desenvolvimento de um método FDD (combinação de técnicas não supervisionadas). Finalmente, realizou-se experimentos com um conjunto de dados real, fornecidos pela Vale S.A. Esses dados operacionais, coletados durante a operação (movimentação para configuração de Normal ou Reverso) de um AMV – realizado pela máquina de chave –, representam as condições de uso dos AMVs em um determinado período de tempo. O desempenho desses modelos FDD, no processo de detecção e diagnóstico do estado de operacionalidade dos AMVs, foram avaliados conforme o procedimento de avaliação de performance proposto.

6.1 PRINCIPAIS CONTRIBUIÇÕES

Neste trabalho científico foram pesquisados diversos métodos de detecção e diagnóstico de falhas – FDD, os quais auxiliaram como base de conhecimento na proposição dos métodos FDD. Diversos desafios estão envolvidos com o processo de detecção e diagnóstico de falhas, principalmente quando esses métodos são aplicados em um domínio de aplicação real. Sabendo disso, os objetivos desta pesquisa relacionados a proposição dos métodos FDD, foram:

1. Explorar simultaneamente os vários estados de degradação de um sistema por meio de diferentes modos de falha e identificá-los com precisão, além de relacionar esses estados com níveis de criticidade.
2. Realizar uma análise comparativa com os métodos propostos, de modo a determinar o melhor método FDD do estudo de caso, o qual foi realizado em conjunto com a Vale S.A.

Para o **primeiro objetivo**, a princípio, apresentou-se o estado-da-arte dos métodos de detecção e diagnóstico de falhas. Com este propósito, foram identificadas e avaliadas diversas pesquisas que objetivam tanto o desenvolvimento científico dessa área quanto a aplicação desses métodos em algum domínio de aplicação, por exemplo, o sistema de transporte ferroviário. Além disso, também foi mostrado uma visão geral sobre o estado-da-arte da manutenção preditiva, aplicada principalmente no setor industrial.

Posteriormente, foi proposta uma metodologia para o desenvolvimento e avaliação dos métodos de detecção e diagnóstico do estado de operacionalidade de um equipamento/sistema (considerando, como estudo de caso, de um AMV). No contexto dessa metodologia, o uso de técnicas não supervisionadas na composição dos métodos possibilitou a exploração do espaço de estados de degradação do equipamento (etapa de diagnóstico). Esses estados de degradação, representados por um conjunto de *clusters*, representam os modos de falha do equipamento.

Desta forma, por meio de um procedimento de interpretação de *clusters*, obteve-se os diversos modos de falha. Neste caso, nove modos de falha foram identificados, são eles, *Atrito*, *Atraso na Partida*, *Corrente Baixa e Oscilando*, *Desligamento do Motor*, *Múltiplos Acionamentos Simultâneos*, *Problema de Medição*, *Problema na Fase 2*, *Sobrecorrente no Destravamento* e *Sobrecorrente no Travamento*. Além disso, para cada um desses modos de falha foi apresentado o seu nível de criticidade, variando entre baixa, média e alta, respectivamente.

No caso da detecção de falhas, as técnicas não supervisionadas foram utilizadas para particionar os dados de acordo com os estados operacionais do equipamento. Dessa forma, também por meio de um procedimento de interpretação de *clusters*, é plausível analisar os três estados operacionais desse equipamento. São eles: (1.) *Indisponível*, o equipamento não consegue exercer a função para qual foi projetado; (2.) *Operacional sem degradação*, o equipamento está operando conforme o esperado, sem desvio de desempenho, (3.) *Operacional*

com degradação, o equipamento continua operacional, porém, com desempenho abaixo do esperado.

Sabendo disso, as técnicas não supervisionadas *K*-means, Mapas Auto-organizáveis e Autoencoder foram combinadas visando a criação dos métodos FDD. Ao todo, três métodos FDD foram propostos, Método 1 (*K*-means & SOM), Método 2 (Autoencoder & *K*-menos) e Método 3 (Autoencoder & SOM). Durante a execução de cada método, como uma sub-etapa da detecção e previamente ao diagnóstico de falhas, realiza-se um procedimento de detecção de disfunções.

Cabe ressaltar que esses métodos foram propostos para auxiliar, como uma ferramenta de apoio ao especialista de domínio no processo de classificação do estado de operacionalidade do ativo, na aplicação da manutenção preditiva em AMVs. Portanto, podem contribuir com a formação (suporte ao treinamento) de recursos humanos (especialistas de domínio), reduzindo a subjetividade deste processo e variabilidade de critério de decisão entre membros da equipe. A utilização desses métodos possibilita a alocação da mão-de-obra necessária de acordo com o modo de falha do equipamento. Além disso, pode proporcionar um aumento da segurança operacional, incluindo a operação de manutenção e disponibilidade da ferrovia.

No contexto do programa de manutenção baseada em condição que está sendo desenvolvido na Vale S.A, esses algoritmos, no sistema de monitoramento de sinalização em uso, podem ser utilizados no componente *Machine Learning (ML) Algorithms* para auxiliar o especialista de domínio no processo de detecção e diagnóstico do estado de operacionalidade do AMV com um grau adequado de precisão e robustez.

Para o **segundo objetivo**, os desempenhos dos modelos FDD (obtidos por meio da implementação dos métodos FDD propostos) foram avaliados, seguindo o processo proposto, em diversos cenários de teste. De modo geral, os resultados dos métodos foram satisfatórios, os quais constituem os modelos e por isso podem afetar os seus desempenhos, principalmente na detecção do estado operacional do AMV. Em todos os cenários, o detector *ensemble* de falhas – treinado com os dados obtidos pela aplicação da respectiva técnica não supervisionada – conseguiu alcançar uma precisão e sensibilidade de pelo menos 90%, em relação a classificação correta das instâncias defeituosas do conjunto de dados de teste.

Com relação ao diagnóstico do estado de operacionalidade do AMV, as performances desses métodos também foram satisfatórias, mas abaixo do desempenho geral apresentado na etapa de detecção. No caso da etapa de diagnóstico, o classificador *ensemble* de falhas – assim como no caso anterior, é treinado com os dados obtidos pela aplicação da respectiva técnica não supervisionada – conseguiu obter uma boa generalização sobre três modos de falhas (*Atrito*, *Problema de Medição* e *Problema na Fase 2*), dentre nove possíveis.

Como o principal resultado da análise comparativa realizada, obteve-se um *ranking*. Esse *ranking* apresenta as melhores técnicas não supervisionadas para o processo de detecção e diagnóstico do estado de operacionalidade de um AMV. Como evidenciado nos experimentos realizados, o algoritmo Autoencoder obteve o melhor desempenho na detecção não supervisionada de falhas. Assim como a rede SOM obteve o melhor desempenho no diagnóstico não supervisionado de falhas. Baseando-se nisso e nos resultados apresentados na *Tabela 12*, o **Método 3 (Autoencoder & SOM)** foi o melhor método FDD, dentre os métodos avaliados neste trabalho científico.

6.2 LIMITAÇÕES DO TRABALHO

Um problema comum ao se trabalhar com dados de um domínio de aplicação real é a dificuldade em avaliar a qualidade da solução encontrada, principalmente quando se objetiva encontrar padrões por meio de técnicas de agrupamento de dados. Por exemplo, como saber se os grupos encontrados representam, de fato, comportamentos de um equipamento do domínio de aplicação. Ou seja, como saber se o agrupamento realizado é o mais adequado para os dados utilizados?

Por conta disso, a metodologia proposta utiliza o conhecimento prévio do especialista de domínio, representado por um conjunto de dados manualmente rotulado pelo mesmo, para superar esse problema e otimizar, tanto em relação a mão-de-obra quanto ao tempo gasto, o processo de interpretação dos grupos gerados em uma tarefa de agrupamento de dados. Devido a isso, para utilização dessa metodologia, faz-se necessário a existência desse conjunto de dados de referência. Sendo essa uma das limitações deste trabalho.

Observe que, nessa metodologia, o especialista do domínio tem participação indireta na etapa de interpretação dos *clusters*, uma vez que o mesmo fornece apenas um conjunto de dados manualmente rotulado. Todavia, o procedimento de extração das regras de domínio independe

da participação do especialista. Portanto, o mesmo não tem participação constante (algo comum no estado-da-arte) no processo de interpretação dos *clusters*.

Além disso, na etapa de diagnóstico de falhas, considerou-se que existe uma relação um-para-um entre uma instância de dados defeituosa e um tipo de defeito (causa). Ou seja, assumiu-se que uma instância de dados possui apenas uma causa para o comportamento apresentado. Todavia, como discutido anteriormente, observou-se que isso não acontece para todos os casos avaliados neste trabalho. Portanto, dado que uma instância de dados defeituosa pode possuir mais uma causa para o comportamento apresentado, essa também é uma limitação da metodologia proposta.

6.3 TRABALHOS FUTUROS

A seguir, são apresentados como possíveis trabalhos futuros alguns refinamentos e extensões plausíveis, relacionados a esta pesquisa de investigação científica.

1. **Análise das inconsistências de classificação:** A existência de inconsistências de classificação no conjunto de dados de referência pode afetar tanto a extração de informações do conhecimento do especialista, representado pelo mesmo, quanto o desempenho dos métodos FDD no processo de detecção e diagnóstico de operacionalidade de um AMV. Desta forma, os desempenhos dos métodos FDD propostos, em um contexto de aplicação sem inconsistências no conjunto de dados de referência, podem ser superiores aos apresentados neste trabalho.
2. **Existência de diversas causas para um comportamento defeituoso:** Na etapa de diagnóstico de falhas, seria interessante analisar as performances desses métodos em um cenário cuja existência de mais de uma causa, para o comportamento defeituoso de uma instância de dados, fosse possível. Isso é, considerar uma relação um-para-muitos ao invés de uma relação um-para-um.
3. **Análise em camadas dos modos de falha:** Uma alternativa para a tarefa de classificação multiclasse executada na etapa de diagnóstico de falhas, a qual analisa todos os modos de falha simultaneamente, seria aplicar um processo em camadas de análise dos mesmos. A princípio, os modos de falhas mais críticos poderiam ser mantidos conforme apresentado neste trabalho. Porém, caso uma instância seja classificada como pertencente a um desses modos críticos, pode-se adicionar mais uma camada de classificação. Essa nova camada é composta por graus intermediários, de

modo a elevar o nível de conhecimento sobre a criticidade dessa instância. Por exemplo, considerando o modo de falha *Atrito*, pode-se dividir esse modo de falhas em dois, a saber, *Médio Atrito* e *Alto Atrito*. Contudo, os modos de falha para os quais não se deseja obter mais informações não precisam ser divididos.

4. **Interpretação semiautomática de *clusters*:** O método de interpretação semiautomática de *clusters* pode ser aprimorado por meio da união do conhecimento do especialista de domínio com as especificações do fabricante do equipamento (as quais podem ser tabuladas para uma planilha do Excel e/ou arquivo CSV). Desta forma, no processo de geração e/ou extração das regras de domínio seriam considerados ambas formas de conhecimento, deixando essa etapa mais robusta.
5. **Prognóstico de falhas:** Uma extensão direta desta pesquisa é a predição do tempo restante de funcionamento de um equipamento. Considerando o contexto do estudo de caso, seria possível determinar até quando uma máquina de chave poderia funcionar corretamente sem ocasionar a disfunção de um AMV. Neste caso, pode-se determinar esse tempo restante de funcionamento por meio da quantidade de dias ou de movimentações até a ocorrência da disfunção.
6. **Combinação de métodos FDD visando a diversidade na etapa de Diagnóstico de Falhas:** No processo de diagnóstico de falhas, pode-se combinar dois métodos FDD para aumentar a diversidade do classificador *ensemble* utilizado no diagnóstico dessas falhas. A combinação desses métodos objetiva maximizar a taxa de acertos por modo de falhas. Essa combinação pode ser realizada de duas formas: (1.) *Ensemble* de técnicas não supervisionadas na geração do conjunto de dados de treinamento da etapa de diagnóstico; e (2.) Combinação das predições dos classificadores *ensemble* de cada modelo FDD. Observe que essa combinação pode ser realizada de forma majoritária. Ou seja, tem-se um classificador *ensemble*, com um grau maior de robustez, que é composto por pelo menos dois classificadores *ensemble*.
7. **Manutenção dos modelos FDD:** O processo de manutenção dos modelos FDDs propostos neste trabalho científico pode ser executado por meio do monitoramento do desempenho dos mesmos. Para isso, faz-se necessário avaliar o desempenho dos mesmos periodicamente e decidir, baseando-se em premissas de performance previamente definidas, quando retreinar o modelo – isso caracteriza a “manutenção” do modelo.

7 CONCLUSÃO

Os modelos propostos no presente trabalho científico podem ser utilizados de diversas formas no contexto de um programa de manutenção preditiva. Por exemplo, inicialmente, podem auxiliar o especialista de domínio tanto na obtenção de conhecimento sobre o comportamento do equipamento analisado quanto na definição de padrões de performance, visando a manutenção dos modelos. Além disso, periodicamente, pode-se treinar novamente o modelo FDD – considerando apenas o módulo supervisionado (que visa a classificação de novas instâncias) – com os dados coletados com o uso do mesmo e validados (pelo especialista de domínio).

Sabendo-se disso, em uma primeira aplicação desses modelos de detecção e diagnóstico de falhas, é recomendado utilizar tanto o módulo não supervisionado quando o módulo supervisionado. Uma vez que o objetivo principal da abordagem proposta no desenvolvimento dos modelos é conseguir aprender informações úteis a partir apenas dos dados e aplicar o conhecimento adquirido em eventos futuros.

Ao surgir a necessidade de realizar a manutenção do modelo FDD, a qual pode ser definida por meio do monitoramento do seu desempenho, pode-se trabalhar diretamente com o módulo supervisionado. Para isso, considerando a existência de uma base de dados coletada a partir do uso do modelo em questão e validada pelo especialista de domínio, o classificador *ensemble* deve ser treinado com os dados provindos dessa base – em outras palavras, os classificadores-base devem ser treinados. Esse novo processo de treinamento do classificador *ensemble* possibilita uma redução da subjetividade do procedimento de geração da base de dados. Portanto, essa atualização e/ou manutenção do modelo FDD pode impactar positivamente em seu desempenho.

REFERÊNCIAS

- ABGHARI, S. et al. A Minimum Spanning Tree Clustering Approach for Outlier Detection in Event Sequences. **Proceedings - 17th IEEE International Conference on Machine Learning and Applications, ICMLA 2018**, p. 1123–1130, 2019.
- ABHILASH, C. B.; ROHITAKSHA, K.; BIRADAR, S. A comparative analysis of data sets using Machine Learning techniques. **Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014**, p. 24–29, 2014.
- ABRAMAN. Documento Nacional - A Situação da Manutenção no Brasil 2017. p. 1–14, 2017. Disponível em: <<https://docplayer.com.br/105100568-Resultado-do-documento-nacional-2017.html>>.
- ABRAMAN. **Mensagem do presidente da Abramam**. Disponível em: <<http://www.abraman.org.br/sidebar/documento-nacional/mensagem-do-presidente>>. Acesso em: 6 jul. 2019.
- AGRAWAL, S.; AGRAWAL, J. Survey on anomaly detection using data mining techniques. **Procedia Computer Science**, v. 60, n. 1, p. 708–713, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.procs.2015.08.220>>.
- AHMAD, R.; KAMARUDDIN, S. An overview of time-based and condition-based maintenance in industrial application. **Computers and Industrial Engineering**, v. 63, n. 1, p. 135–149, 2012. Disponível em: <<http://dx.doi.org/10.1016/j.cie.2012.02.002>>.
- ALMEIDA JR, J. **Segurança em Sistemas Críticos e em Sistemas de Informação - Um Estudo Comparativo**. 2003. Escola Politécnica da Universidade de São Paulo, 2003.
- ALPAYDIN, E. **Introduction to machine learning**. Second ed. [s.l.] MIT press, 2009.
- AMRUTHNATH, N.; GUPTA, T. A research study on unsupervised machine learning algorithms for early fault detection in predictive maintenance 2018 5th International Conference on Industrial Engineering and Applications, ICIEA 2018, 2018. .
- ANTT. **Anuário Estatístico do Setor Ferroviário**. Disponível em: <http://www.antt.gov.br/ferrovias/arquivos/Anuario_Estatistico.html>. Acesso em: 16 jun. 2019.
- ASADA, T.; ROBERTS, C. Development of an effective condition monitoring system for AC point machines. **IET Conference Publications**, v. 2011, n. 581 CP, 2011.

- ASADA, T.; ROBERTS, C.; KOSEKI, T. An algorithm for improved performance of railway condition monitoring equipment: Alternating-current point machine case study. **Transportation Research Part C: Emerging Technologies**, v. 30, p. 81–92, 2013.
- ASSIS, A. C. V.; DALTO, E. J.; FERREIRA, M. A. Ferrovias De Carga Brasileiras : Uma Análise Setorial. **BNDES Setorial** 46, n. 46, p. 79–126, 2017.
- ATAMURADOV, V. et al. Failure diagnostics for railway point machines using expert systems. In: 2009 IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, SDEMPED 2009, **Anais...**2009.
- BALDI, P. Autoencoders, Unsupervised Learning, and Deep Architectures. p. 37–50, 2012. Disponível em: <<http://proceedings.mlr.press/v27/baldi12a/baldi12a.pdf>>.
- BEAMON, B. M. et al. Logística E Transportes : Uma Discussão Sobre Os Modais De Transporte E O Panorama Brasileiro . **Transportation Science**, v. 32, n. 3, p. 128, 2002. Disponível em: <<http://dx.doi.org/10.3141/2238-04%5Cnhttp://dx.doi.org/10.1108/14636681211239755%5Cnhttp://www.ourenergypolicy.org/freight-transportation-modal-shares-scenarios-for-a-low-carbon-future/%5Cnhttp://dx.doi.org/10.1016/j.apm.2009.08.004%5Cnhttp://onlinelibrary>>.
- BEBIS, G.; GEORGIOPOULOS, M. Feed-forward neural networks. **IEEE Potentials**, v. 13, n. 4, p. 27–31, 1994.
- BELLEK, M. **DESIGN AND RAMS ANALYSIS OF RAILWAY INTERLOCKING SYSTEMS USING FORMAL METHODS**. 2013. İstanbul Technical University, 2013.
- BENTÉJAC, C.; CSÖRGÖ, A.; MARTÍNEZ-MUÑOZ, G. A comparative analysis of gradient boosting algorithms. **Artificial Intelligence Review**, v. 54, n. 3, p. 1937–1967, 2021. Disponível em: <<https://doi.org/10.1007/s10462-020-09896-5>>.
- BIAN, C. et al. Degradation state mining and identification for railway point machines. **Reliability Engineering and System Safety**, v. 188, p. 432–443, 2019.
- BORGI, T. et al. Data analytics for predictive maintenance of industrial robots. In: 2017 International Conference on Advanced Systems and Electric Technologies (IC_ASET), **Anais...**2017.
- BOZZANO, M.; VILLAFIORITA, A. **Design and safety assessment of critical systems**. [s.l.] Auerbach Publications, 2010.
- BREIMAN, L. Bagging predictors. **Machine learning**, v. 24, n. 2, p. 123–140, 1996.

- BREIMAN, L. Random forests. **Machine learning**, v. 45, n. 1, p. 5–32, 2001.
- CARVALHO, T. P. et al. **A systematic literature review of machine learning methods applied to predictive maintenance** *Computers & Industrial Engineering*, 2019. .
- CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Survey of Anomaly Detection. **ACM Computing Survey**, n. September, p. 1–72, 2009. Disponível em: <<http://cucis.ece.northwestern.edu/projects/DMS/publications/AnomalyDetection.pdf>>.
- CHEN, J.; ROBERTS, C. Effective condition monitoring of line side assets. 2006.
- 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, *Anais...*2016.
- CHIANG, LEO H AND RUSSELL, EVAN L AND BRAATZ, R. D. **Fault detection and diagnosis in industrial systems**. [s.l.] Springer Science & Business Media, 2000.
- CNT. Anuário CNT do Transporte. Estadistíias consolidadas. p. 229, 2018. Disponível em: <www.cnt.org.br>.
- CYBENKO, G. Approximation by superpositions of a sigmoidal function. **Mathematics of control, signals and systems**, v. 2, n. 4, p. 303–314, 1989a.
- CYBENKO, G. Approximation by superpositions of a sigmoidal function. **Mathematics of control, signals and systems**, v. 2, n. 4, p. 303–314, 1989b.
- DATTA, A.; MAVROIDIS, C.; HOSEK, M. A role of unsupervised clustering for intelligent fault diagnosis. **ASME International Mechanical Engineering Congress and Exposition, Proceedings**, v. 9 PART A, p. 687–695, 2008.
- DAWOUD, A.; SHAHRISTANI, S.; RAUN, C. Deep learning for network anomalies detection. **Proceedings - International Conference on Machine Learning and Data Engineering, iCMLDE 2018**, p. 117–120, 2019.
- DAYCHOUM, M. T.; SAMPAIO, P. R. P. **Regulação e Concorrência no Setor Ferroviário**. [s.l.: s.n.]
- DE BRUIN, T.; VERBERT, K.; BABUSKA, R. **Railway Track Circuit Fault Diagnosis Using Recurrent Neural Networks** *IEEE Transactions on Neural Networks and Learning Systems*, 2017. .
- DIETTERICH, T. Overfitting and undercomputing in machine learning. **ACM Computing**

Surveys, v. 27, n. 3, p. 326–327, 1 set. 1995. Disponível em: <<http://portal.acm.org/citation.cfm?doid=212094.212114>>.

DINCEL, E.; ERIS, O.; KURTULAN, S. Automata-based railway signaling and interlocking system design. **IEEE Antennas and Propagation Magazine**, v. 55, n. 4, p. 308–319, 2013.

DU, S.; LV, J.; XI, L. Degradation process prediction for rotational machinery based on hybrid intelligent model. **Robotics and Computer-Integrated Manufacturing**, v. 28, n. 2, p. 190–207, 2012. Disponível em: <<http://dx.doi.org/10.1016/j.rcim.2011.08.006>>.

FACELI, K. et al. **Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina**Ltc, 2011. .

FAUSETT, L. V; OTHERS. **Fundamentals of neural networks: architectures, algorithms, and applications**. [s.l.] prentice-Hall Englewood Cliffs, 1994. v. 3

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. **Journal of computer and system sciences**, v. 55, n. 1, p. 119–139, 1997.

FUN, S. **How Train Changes/Switches Track at Crossing | Solidworks Motion Analysis - YouTube**. Disponível em: <<https://www.youtube.com/watch?v=3NELFCMl3co&t=4s>>. Acesso em: 20 out. 2020.

GARDNER, M. W.; DORLING, S. R. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. **Atmospheric Environment**, v. 32, n. 14, p. 2627–2636, 1998. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1352231097004470>>.

GERTLER, J. Fault Detection and Diagnosis. In: BAILLIEUL, J.; SAMAD, T. (Ed.). **Encyclopedia of Systems and Control**. London: Springer London, 2015. p. 417–422.

GERTLER, J. **Fault detection and diagnosis in engineering systems**. [s.l.] Routledge, 2017.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [s.l.] MIT Press, 2016.

GUO, Z. et al. A fault detection method for railway point machine operations based on stacked autoencoders. **ICAC 2018 - 2018 24th IEEE International Conference on Automation and Computing: Improving Productivity through Automation and Computing**, n. September, p. 1–6, 2018.

HAGA, A.; ARGENTON E, L.; BARBOSA, S. Viabilidade da Aplicação do Monitoramento Contínuo nos Sistemas de Máquina de Chave e Proteção da Sobretensão do Negativo-Terra.

2004.

HAN, J.; KAMBER, M.; PEI, J. **Data mining concepts and techniques third edition**. Third Edit ed. [s.l: s.n.]

HAN, J.; MORAGA, C. The influence of the sigmoid function parameters on the speed of backpropagation learning. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **Anais...** Springer Verlag, 1995. Disponível em: <https://link.springer.com/chapter/10.1007/3-540-59497-3_175>. Acesso em: 15 fev. 2021.

HASHEMIAN, H. M.; BEAN, W. C. State-of-the-art predictive maintenance techniques. **IEEE Transactions on Instrumentation and measurement**, v. 60, n. 10, p. 3480–3492, 2011.

HAY, W. W. **Railway Engineering**. [s.l: s.n.]

HAYKIN, S. **Redes Neurais Artificiais: Princípios e Práticas**. 2ª edição ed. [s.l.] Editora Bookman, Porto Alegre, 900p, 2001.

HEARST, M. A. et al. Support vector machines. **IEEE Intelligent Systems and their applications**, v. 13, n. 4, p. 18–28, 1998.

HOFMANN, E.; RÜSCH, M. Industry 4.0 and the current status as well as future prospects on logistics. **Computers in Industry**, v. 89, p. 23–34, 2017.

ISERMANN, R. **Fault-diagnosis systems: an introduction from fault detection to fault tolerance**. [s.l.] Springer Science & Business Media, 2006.

ISO 13372. International Standard - Condition monitoring and diagnostics of machines — Vocabulary. 2012.

J-C, L. et al. Basic Concepts and Taxonomy of Dependable and Secure Computing. **IEEE Transactions on Dependable and Secure Computing**, v. 01, n. 1, p. 11–33, 2004.

JIN, W. et al. Development and evaluation of health monitoring techniques for railway point machines. **2015 IEEE Conference on Prognostics and Health Management: Enhancing Safety, Efficiency, Availability, and Effectiveness of Systems Through PHAf Technology and Application, PHM 2015**, p. 1–11, 2015.

JOHNSON, B. W. **Design and analysis of fault-tolerant digital systems**. [s.l.] Addison-Wesley Reading, MA, 1989. v. 218

KANSO, K.; MOLLER, F.; SETZER, A. Automated Verification of Signalling Principles in

Railway Interlocking Systems. **Electronic Notes in Theoretical Computer Science**, v. 250, n. 2, p. 19–31, 2009. Disponível em: <<http://dx.doi.org/10.1016/j.entcs.2009.08.015>>.

KARDEC, A.; NASCIF, J. **Manutenção-função estratégica**. 3ª edição ed. [s.l.] Qualitymark Editora Ltda, 2009.

KARLIK, B.; VEHBI, A. Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks. **International Journal of Artificial Intelligence And Expert Systems (IJAE)**, v. 1, n. 4, 2011.

KATAYAMA, N. et al. Attempt to isolate mast-cell precursors based on the differential sensitivity to UV-B and X-irradiation. **Toxic Substances Journal**, v. 13, n. 2, p. 85–95, 1994.

KLINCEVICIUS, M. G. Y. **Estudo de propriedades, de tensões e do comportamento mecânico de lastros ferroviários**. 2011. 2011.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. **Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2**, v. 2, n. 0, p. 1137–1143, 1995.

KOHONEN, T. The Self-organizing Map. **Proceedings of the IEEE**, v. 78, n. 9, p. 1464–1480, 1990.

KOHONEN, T. Essentials of the self-organizing map. **Neural Networks**, 2013.

LEE, T. J. et al. Precision and recall for range-based anomaly detection. **arXiv preprint arXiv:1801.03175**, 2018.

LIU, J. et al. Anomaly Detection in Manufacturing Systems Using Structured Neural Networks. **Proceedings of the World Congress on Intelligent Control and Automation (WCICA)**, v. 2018- July, p. 175–180, 2019.

MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: In 5-th Berkeley Symposium on Mathematical Statistics and Probability, **Anais...**1967.

MÁRQUEZ, A. C. **The maintenance management framework: models and methods for complex systems maintenance**. [s.l.] Springer Science & Business Media, 2007.

MARTINS, C. A. Uma abordagem para pré-processamento de dados textuais em algoritmos de aprendizado. p. 208, 2003.

MENZE, B. H. et al. A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. **BMC**

Bioinformatics, v. 10, p. 1–16, 2009.

MOBLEY, R. K. **An Introduction to Predictive Maintenance**. [s.l.] Elsevier, 2002.

NBR-5462 ABNT. NBR 5462 Confiabilidade e manutenibilidade. **Associação Brasileira De Normas Técnicas**, n. 1, p. 37, 1994.

NG, A.; OTHERS. Sparse autoencoder. 2011.

NITISH, S. et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. **Journal of Machine Learning Research**, 2014.

NWANKPA, C. et al. Activation Functions: Comparison of trends in Practice and Research for Deep Learning. p. 1–20, 2018. Disponível em: <<http://arxiv.org/abs/1811.03378>>.

PYRGIDIS, C. N. **Railway Transportation Systems**. [s.l.: s.n.]

QIANG, X.; CHENG, G.; LI, Z. A survey of some classic Self-Organizing Maps with incremental learning. **ICSPS 2010 - Proceedings of the 2010 2nd International Conference on Signal Processing Systems**, v. 1, p. V1-804-V1-809, 2010.

RASCHKA, S. **Python machine learning**. [s.l.] Packt Publishing Ltd, 2015.

SARAVANAN, R.; SUJATHA, P. Algorithms : A Perspective of Supervised Learning Approaches in Data Classification. **2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)**, n. Iciccs, p. 945–949, 2018.

SCHAPIRE, R. E. The strength of weak learnability. **Machine learning**, v. 5, n. 2, p. 197–227, 1990.

SCHAPIRE, R. E.; FREUND, Y. **Boosting: Foundations and Algorithms**. [s.l.] The MIT Press, 2014.

SELIG, ERNEST T AND WATER, J. M. Track geotechnology. p. 12–19, 1995.

SELIYA, N.; KHOSHGOFTAAR, T. M.; VAN HULSE, J. A study on the relationships of classifier performance metrics. **Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI**, p. 59–66, 2009.

SHIN, J. H.; JUN, H. B. On condition based maintenance policy. **Journal of Computational Design and Engineering**, v. 2, n. 2, p. 119–127, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.jcde.2014.12.006>>.

SILVA NETO, A. V. da. Modelo de predição de falhas baseado em processos estocásticos e filtragem Kalman para suporte à manutenção preditiva de sistemas elétricos, eletrônicos e

programáveis. p. 302, 2014. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3141/tde-19032015-160659/>.

SOMMERVILLE, I. Engenharia de Software, 9a. **São Palo, SP, Brasil**, 2011.

STOREY, N. R. **Safety critical computer systems**. [s.l.] Addison-Wesley Longman Publishing Co., Inc., 1996.

SUSTO, G. A. et al. Machine learning for predictive maintenance: A multiple classifier approach. **IEEE Transactions on Industrial Informatics**, v. 11, n. 3, p. 812–820, 2015.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining (New International Editon)**. [s.l: s.n.]

TIAN, J. et al. An ensemble learning-based fault diagnosis method for rotating machinery. In: 2017 Prognostics and System Health Management Conference (PHM-Harbin), **Anais...**2017.

VALE. Visão Geral Sinalização. 2017.

VALE. **Vale - Fact Sheet**. [s.l: s.n.]. Disponível em: moz-extension://a6f9897f-5180-4b6a-ad81-4b0b26d43ff5/enhanced-reader.html?openApp&pdf=http%3A%2F%2Fwww.vale.com%2FPT%2Finvestors%2Fcompany%2Ffact-sheet%2FDocuments%2Ffactsheet2Q19_p.pdf. Acesso em: 23 jun. 2021.

VALE. **Estrada de Ferro Carajás é considerada ferrovia mais eficiente do país**. Disponível em: <http://www.vale.com/brasil/PT/aboutvale/news/Paginas/estrada-de-ferro-carajas-e-considerada-ferrovia-mais-eficiente-do-pais.aspx>. Acesso em: 16 set. 2020.

VILEINISKIS, M.; REMENYTE-PRESCOTT, R.; RAMA, D. A fault detection method for railway point systems. **Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit**, v. 230, n. 3, p. 852–865, 2016a.

VILEINISKIS, M.; REMENYTE-PRESCOTT, R.; RAMA, D. A fault detection method for railway point systems. **Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit**, v. 230, n. 3, p. 852–865, 9 mar. 2016b. Disponível em: <http://journals.sagepub.com/doi/10.1177/0954409714567487>.

VISMARI, L. F. **Vigilância dependente automática no controle de tráfego aéreo: avaliação de risco baseada em modelagem em redes de Petri fluidas e Estocásticas**. 2007. 2007.

XENOS, H. G. Gerenciando a manutenção produtiva. **Belo Horizonte: Editora de Desenvolvimento Gerencial**, v. 171, 1998.

XIE, R. et al. A predictive model of gene expression using a deep learning framework. **Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016**, p. 676–681, 2017.

APÊNDICE A – OTIMIZAÇÃO DOS MODELOS DE CLASSIFICAÇÃO

Neste APÊNDICE apresenta-se o processo de otimização dos modelos de classificação utilizados para criar os *ensembles* de classificadores aplicados na tarefa de detecção e diagnóstico do estado de operacionalidade do AMV. Esse processo é constituído de duas etapas, são elas, definição da estrutura e/ou dos parâmetros de cada modelo e a busca exaustiva pela melhor combinação dos parâmetros definidos. As informações aqui contidas foram usadas para modelar cada classificador-base implementado.

1 Definição da estrutura e/ou dos parâmetros dos modelos

Esta etapa do processo de otimização é responsável por definir o espaço de atuação do algoritmo de busca exaustiva (descrito na próxima etapa). Com esta finalidade, para cada modelo de classificação, define-se um conjunto de valores possíveis para cada atributo (hiperparâmetro) do modelo que será otimizado. Esses valores são especificados por meio de uma estrutura chave – valor, neste caso, atributo - lista de valores.

O espaço de busca é proporcional a quantidade de atributos a serem otimizados e seus respectivos valores. Ou seja, quanto maior o número de atributos analisados, incluindo o tamanho da lista de valores de cada atributo, maior será o espaço de busca. Consequentemente, o sucesso do algoritmo de busca exaustiva em encontrar a melhor combinação dos hiperparâmetros depende do espaço de busca. Pois, em alguns, pode ocorrer um problema de explosão do número de estados, isto é, da quantidade de combinações analisadas.

Após a definição dos atributos que serão otimizados, incluindo a estrutura base do modelo de classificação, a criação de um novo modelo é realizada por meio de uma combinação dos valores de cada atributo da estrutura base definida. A *Figura 69* apresenta, em alto nível, os componentes necessários para a criação desse modelo. No lado esquerdo superior da figura apresenta-se um exemplo da relação atributo – hiperparâmetros (lista de valores), assim como do lado direito superior apresenta-se a estrutura base definida (MLP simples com apenas duas camadas ocultas e um neurônio na camada de saída – utiliza *Sigmoide* como função de ativação –).

No caso do exemplo da *Figura 69* os atributos a serem otimizados são: a quantidade de iterações de treinamento (*epochs*), número de neurônios das camadas ocultas (*neurons*), a forma de inicialização (*init*), a quantidade de instâncias que serão passadas como entrada, por vez, na rede MLP durante o treinamento do modelo (*batch_size*), o algoritmo de otimização (*optimizer*), a porcentagem de “corte” a cada época (*dropout*) e, por fim, a função de ativação aplicada nas camadas intermediárias (*activation*).

Figura 69. Exemplo de criação de novos modelos de classificação por meio da combinação dos valores dos atributos da estrutura base (**Fonte:** Autor).

Dito isso, a criação de um novo modelo pode ser efetuada como segue: (1.) Para cada atributo definido, escolhe-se um valor de hiperparâmetro (combinação); (2.) Crie uma instância da estrutura base (modelo) com essa combinação (veja os exemplos de modelos mostrados na *Figura 69*). O número total de combinações – logo, de modelos – é calculado por meio da multiplicação da quantidade de hiperparâmetros dos atributos. Por exemplo, com relação aos atributos definidos na *Figura 69*, tem-se um total de 3.240 combinações ($5 \times 2 \times 3 \times 6 \times 2 \times 3 \times 3 = 3240$). Por isso, a definição dos atributos e hiperparâmetros é uma etapa fundamental do processo de otimização dos modelos de classificação.

2 Busca exaustiva pela melhor combinação dos parâmetros de cada modelo de classificação

O algoritmo de busca exaustiva visa obter a melhor combinação dos hiperparâmetros, os quais foram especificados na etapa anterior. Consequentemente, como resultado desse algoritmo, tem-se o melhor modelo de classificação. Devido a isso, o algoritmo analisa todas as combinações possíveis, o que pode ser tanto computacionalmente inviável quanto custoso. Uma forma de superar esse problema é por meio de uma boa definição do conjunto de atributos analisados.

A obtenção do melhor modelo de classificação é realizada por meio da utilização de cada novo modelo em uma tarefa de classificação (veja a *Figura 70*). O conjunto de dados independente (manualmente rotulado pelo especialista de domínio) – possui menos de 2 mil instâncias de dados – é utilizado nessa tarefa tanto para treinar quanto avaliar o desempenho do classificador (obtido pelo treinamento do modelo). Por fim, a “qualidade” do modelo é representada pelo desempenho desse classificador nessa tarefa de classificação. Observe que para cada modelo analisado pelo algoritmo de busca exaustiva, executa-se uma determinada tarefa de classificação e guarda-se o resultado (desempenho) obtido.

Figura 70. Sistemática de uma tarefa de classificação (**Fonte:** Autor).

Essa tarefa de classificação depende do contexto de utilização do algoritmo, no caso do SOS:D², tem-se tanto uma tarefa de detecção quanto de diagnóstico de falhas. Vale salientar que essas tarefas são executadas separadamente. Observe que, dada as suas características, apesar de usar a mesma estrutura, os modelos resultantes de cada tarefa podem ser diferentes. Por causa disso, como discutido nos capítulos anteriores, as métricas calculadas têm pesos diferentes em cada tarefa. Por exemplo, precisão e sensibilidade são as métricas principais da detecção de falhas, do mesmo modo que, a acurácia é a métrica principal na etapa de diagnóstico.

Como ilustrado na *Figura 70*, o conjunto de dados independente é dividido em treinamento e teste – 80% e 20%, respectivamente, de todas as instâncias de dados –. Cada modelo, referente a cada combinação, é treinado com os exemplos de treinamento, inclusive usa-se a validação cruzada durante o treinamento do modelo – note que o número de modelos é multiplicado pela quantidade de subconjuntos utilizado–. Em seguida, o classificador obtido com o treinamento do modelo, é avaliado com exemplos de teste. Finalmente, as métricas são calculadas para mensurar a eficiência desse classificador.

Portanto, a melhor combinação de hiperparâmetros para cada modelo é aquela cujo o classificador obteve os melhores resultados, de acordo com as métricas mencionadas, na tarefa de classificação realizada.

APÊNDICE B – COMPORTAMENTO DAS FUNÇÕES DE ATIVAÇÃO E CLASSIFICADORES-BASE UTILIZADOS NO CLASSIFICADOR ENSEMBLE

Este APÊNDICE apresenta o comportamento das funções de ativação utilizadas nos modelos de classificação baseado na estrutura da MLP e os demais classificadores que constituem o classificador *ensemble* usado neste trabalho científico. Primeiramente, descreve-se o comportamento de cada função de ativação usada nos modelos MLP. Posteriormente, apresenta-se uma descrição dos classificadores-base.

1 Comportamento das Funções de Ativação dos modelos MLP

Em redes neurais artificiais, tal como a MLP, utiliza-se uma função de ativação para definir se um determinado neurônio de uma camada da rede deve ser ativado ou não. Para isto essa função computa a soma ponderada das entradas e vieses do neurônio. Portanto, uma função de ativação realiza operações com os dados de entrada para gerar uma saída para a rede neural (ou para um neurônio em uma camada intermediária) (NWANKPA et al., 2018).

Observe que a função de ativação é responsável por controlar as saídas geradas pela rede neural. É importante mencionar que, dependendo do contexto de aplicação, uma função de ativação pode não conseguir desempenhar eficientemente o seu papel. É por isso que, para algumas situações específicas, existem funções de ativações amplamente aceitas na literatura. Por exemplo, a função de ativação *Sigmoide* é comumente utilizada em problemas de classificação binária, porém, o mesmo não acontece para problemas de classificação multiclasse (NWANKPA et al., 2018).

1.1 Função *Sigmoide*

A função *Sigmoide*, usualmente também chamada de função logística, é uma função não linear amplamente utilizada como função de ativação em diversos modelos de aprendizado profundo, principalmente em redes neurais artificiais baseadas na estrutura *feedforward* (NWANKPA et al., 2018) (não possui recorrência no fluxo da rede neural, por isso a alimentação da rede dar-se “para frente”).

A função *Sigmoide* é uma função diferenciável, definida para o domínio dos números reais (por isso é dita como uma função real), que possui derivada positiva para qualquer valor no intervalo de $[-\infty, +\infty]$ e algum grau de suavidade (HAN; MORAGA, 1995). Essa função

mapeia os valores de entrada para o intervalo $[0,1]$. Por conta disso, a imagem dessa função é limitada ao domínio dos números reais (isto é, a *Sigmoide* é uma função limitada). Além disso, também por conta dessa limitação, o seu comportamento possui um formato de “S” (KARLIK; VEHBI, 2011). Conforme Nwankpa et al. (2018) essa função é dada pela expressão a seguir:

$$\text{sigmoide}(x) = \left\{ \frac{1}{1 + e^{-x}} \right\}$$

A *Figura 71* apresenta de forma gráfica o comportamento da função *Sigmoide* (logística). Por fim, geralmente, utiliza-se essa função na camada de saída de uma rede neural de aprendizado profundo para prever a probabilidade para uma determinada classe (NWANKPA et al., 2018). Desta forma, a probabilidade de uma nova instância de dados pertencer a uma das classes analisadas é proporcional a sua proximidade aos extremos do intervalo $[0,1]$.

Figura 71. Comportamento da função *Sigmoide* (Fonte: Autor).

1.2 Função *Softmax*

A função *Softmax* é outro tipo de função de ativação comumente usada, em redes neurais artificiais, para computação neural (NWANKPA et al., 2018). Essa função é “frequentemente usada para prever as probabilidades associadas a uma distribuição *Multinoulli* – também chamada de distribuição categórica – ” (GOODFELLOW; BENGIO; COURVILLE, 2016).

Isto posto, a função *Softmax* é amplamente utilizada em casos que um determinado evento pode assumir um número limitado de possibilidades. A saída gerada pela aplicação dessa função, assim como a *Sigmoide*, é um número entre 0 e 1. Porém, no caso dessa função, um número nesse intervalo é gerado para cada possibilidade. Sendo que a soma dessas probabilidades é 1. Devido a esses fatores, geralmente, é aplicada em modelos multiclasse.

Neste caso, essa função retorna à probabilidade do evento analisado para cada uma das classes definidas. Assim, a classe predita para o evento é aquela com o maior valor de probabilidade (NWANKPA et al., 2018). Segundo Goodfellow, Bengio e Courville (2016) a função *Softmax* pode ser definida como a seguir:

$$\text{softmax}(x)_i = \left\{ \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \right\}$$

A Figura 72 ilustra em alto nível o comportamento da função *Softmax*. Finalmente, a principal diferença entre as funções *Sigmoide* e *Softmax* diz respeito a sua aplicação, uma vez que ambas podem ser utilizadas por modelos, baseados em redes neurais, em problemas de classificação. Porém, a primeira é comumente utilizada em uma tarefa de classificação binária enquanto que a segunda é usada em uma tarefa de classificação multiclasse (NWANKPA et al., 2018). Outra diferença entre essas funções é o fato da função *Softmax* poder ser utilizada nas camadas intermediárias (caso raro) e não apenas na camada de saída da rede (caso da *Sigmoide*) (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 72. Comportamento da função *Softmax* (Fonte: Autor).

1.3 Função *ReLU*

A função *ReLU* (*Rectified Linear Unit*) é a função de ativação padrão recomendada para diversas redes neurais artificiais baseadas na estrutura *feedforward* (GOODFELLOW; BENGIO; COURVILLE, 2016). Por conta disso, é amplamente utilizada em modelos de aprendizado profundo. Essa função, quando aplicada à saída de uma transformação linear, produz uma transformação não linear. Porém, “a função permanece muito próxima de linear, no sentido de que é uma função linear por partes com duas peças lineares” (veja a Figura 73) (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 73. Comportamento da função *ReLU* (Fonte: Autor).

Isso possibilita a função *ReLU* preservar, por meio das unidades lineares retificadas – as quais são quase lineares –, diversas propriedades de uma função linear. Desta forma, são mantidas as propriedades que tornam os modelos lineares fáceis de otimizar, com métodos baseados em gradiente, e de obter uma boa generalização (GOODFELLOW; BENGIO; COURVILLE, 2016).

A função de ativação *ReLU* realiza um mapeamento sobre os dados de entrada. Neste caso, por meio de um threshold, define como zero todos os valores de entrada menores do que zero. Ou seja, limita em zero os valores de entrada negativos (NWANKPA et al., 2018). Ainda conforme Nwankpa et al. (2018) essa função pode ser definida como a seguir:

$$relu(x) = \max(0, x) = \begin{cases} x_i, & \text{if } x_i \geq 0 \\ 0, & \text{if } x_i < 0 \end{cases}$$

Em relação as funções anteriores, a saber *Sigmoide* e *Softmax*, a *ReLU* efetua a computação neural de forma mais rápida. Visto que não é necessário calcular expoentes e nem divisões para todos os valores de entrada. Por outro lado, é mais suscetível ao sobre-ajuste. Sendo essa uma limitação desta função (NWANKPA et al., 2018).

2 Classificador *Ensemble*: Classificadores-base

O objetivo principal de um classificador baseado em uma metodologia *ensemble* é combinar um conjunto de classificadores-base, geralmente por meio de voto majoritário, para obter um classificador geral, comumente chamado de classificador *ensemble*, cuja performance seja igual ou superior a performance individual desses classificadores. No contexto de um classificador *ensemble* uma predição é realizada por meio de uma votação majoritária sobre a ponderação dos resultados dos classificadores-base (TIAN et al., 2017).

Isto posto, por intermédio da metodologia *ensemble*, é possível obter como resultado da combinação de diversos classificadores com baixa performance um classificador robusto e com performance satisfatória. Uma vez que esses classificadores-base contribuem, apesar da baixa performance, no aprendizado do classificador geral (TIAN et al., 2017). Dentre as principais estratégias da metodologia (abordagem) *ensemble* tem-se *Bagging* e *Boosting*.

Na abordagem *Bagging*, proposto por Breiman (1996), um modelo de classificação é treinado em k diferentes subconjuntos, gerados a partir do conjunto de dados de treinamento. Esses subconjuntos de treinamento, gerados aleatoriamente, possuem amostragens diferentes. Contudo, existe a reposição de instâncias de dados na geração de um subconjunto de treinamento, o qual possui o mesmo tamanho da base de dados de treinamento original. Por fim, a predição é realizada por meio de voto majoritário.

A abordagem *Boosting*, proposto por Scharpire (1990), baseia-se na suposição de aprendizagem fraca (do inglês, *Weak Learning Assumption*), na qual considera-se que um classificador com baixa performance (geralmente chamado de classificador fraco) aprende uma generalização dos dados de treinamento relativamente superior a um classificador baseado em predição aleatória (isto é, cada valor de saída é predito por meio de um palpite aleatório) (SCHAPIRE; FREUND, 2014).

Em outras palavras, apesar da generalização aprendida pelo classificador ser insatisfatória, supõe-se que a taxa de erros desse classificador é relativamente superior à de um classificador cuja cada predição é uma suposição aleatória. Portanto, assumisse-se que classificadores fracos não são completamente triviais. Devido a isso, a ideia principal do *Boosting* é utilizar no treinamento de um modelo de classificação conjuntos de dados resultantes da aplicação desse modelo em iterações anteriores (aprendizado sucessível). Apesar da baixa performance, isso impõe que esse classificador aprenda algo útil a cada iteração (SCHAPIRE; FREUND, 2014).

Sabendo disso, a *Tabela 13* sumariza os classificadores-base usados na composição do classificador *ensemble* utilizado neste trabalho científico. Nesta tabela, as linhas equivalem aos quatro classificadores utilizados (*Random Forest*, *Ada Boost*, *XGBoost* e *SVM*) e as colunas representam propriedades desses classificadores. Basicamente, a tabela apresenta uma descrição sucinta para cada classificador, além de informações sobre características *ensemble* e seus respectivos autores.

Tabela 13. Descrição dos classificadores-base utilizados no desenvolvimento do classificador *ensemble* (Fonte: Autor).

Classificador	Modelos de classificação			
	Descrição	Ensemble	Metodologia Ensemble	Referência
<i>Random Forest</i>	<i>Ensemble</i> de árvores de decisão que utiliza <i>bagging</i> e seleção aleatória de atributos para tentar criar uma floresta de árvores não correlacionada. Por conseguinte, aumenta a diversidade do classificador ensemble. Geralmente, esse aumento da diversidade aprimora o seu desempenho.	Sim	<i>Bagging</i>	Breiman (2001)
<i>Ada Boost</i>	Treina diversos classificadores-base de forma sucessiva. Cada classificador base é treinado com um conjunto de dados ponderado. Esse conjunto de dados é gerado, a cada sucessão, por meio dos coeficientes de ponderação, calculados com base nos erros de classificação do classificador anterior. Desta forma, cada novo conjunto de dados objetiva complementar o aprendizado.	Sim	<i>Boosting</i>	Freund e Schapire (1997)
<i>XGBoost</i>	<i>Ensemble</i> de árvores de decisão baseado em <i>Gradient Boosting</i> ¹¹ . O <i>XGBoost</i> é um modelo de classificação altamente escalável e eficiente. Para isso, incorpora técnicas para evitar sobre-ajuste, encontrar a melhor divisão para uma árvore de decisão durante a sua construção e aprimorar a velocidade de treinamento do modelo.	Sim	<i>Boosting</i>	Chen e Guestrin (2016)
<i>SVM</i>	Baseado na teoria de aprendizagem estatística. Por meio do mapeamento não linear dos dados de treinamento em um espaço ampliado dos atributos e de uma função <i>kernel</i> , define o hiperplano ótimo de separação das classes. Como resultado desse processo tem-se um limite de decisão não linear no espaço de entrada.	Não	-----	Hearst et al. (1998)

¹¹ Para mais detalhes sobre o *Gradient Boosting* veja (BENTÉJAC; CSÖRGÖ; MARTÍNEZ-MUÑOZ, 2021).