UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ELETRÔNICA E SISTEMAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

RODRIGO DE PAULA MONTEIRO

TREINANDO UM EXTRATOR DE CARACTERÍSTICAS BASEADO EM APRENDIZADO PROFUNDO PARA USO EM DETECÇÃO DE ANOMALIAS

Recife

2021

RODRIGO DE PAULA MONTEIRO

TREINANDO UM EXTRATOR DE CARACTERÍSTICAS BASEADO EM APRENDIZADO PROFUNDO PARA USO EM DETECÇÃO DE ANOMALIAS

Tese submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como requisito parcial para obtenção do título de Doutor em Engenharia Elétrica.

Área de Concentração: Eletrônica.

Orientador: Prof. Dr. Carmelo José Albanez Bastos Filho.

Recife

2021

Catalogação na fonte
Bibliotecária Maria Luiza de Moura Ferreira, CRB-4 / 1469

M775t Monteiro, Rodrigo de Paula.
Treinando um extrator de características baseado em aprendizado profundo para uso em detecção de anomalias / Rodrigo de Paula Monteiro. - 2021.
150 folhas, il.; tab., abr. e sigl.

Orientador: Prof. Dr. Carmelo José Albanez Bastos Filho.

Tese (Doutorado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Elétrica, 2021.
Inclui Referências.

1. Engenharia Elétrica. 2. Detecção de anomalias. 3. Aprendizado profundo. 4. Seleção de protótipos. 5. Aprendizado de máquina. I. Bastos Filho, Carmelo José Albanez (Orientador). II. Título.

UFPE

621.3 CDD (22. ed.) BCTG/2022-07

RODRIGO DE PAULA MONTEIRO

TREINANDO UM EXTRATOR DE CARACTERÍSTICAS BASEADO EM APRENDIZADO PROFUNDO PARA USO EM DETECÇÃO DE ANOMALIAS

Tese apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco, Centro de Tecnologia e Geociências, como requisito para a obtenção do título de doutor em Engenharia Elétrica. Área de concentração: Eletrônica.

Aprovada em: 17/03/2021

BANCA EXAMINADORA

_____

Profº. Dr. Carmelo José Albanez Bastos Filho
(Orientador e Examinador Interno)
Universidade de Pernambuco

_____

Profº. Dr. Manassés Ribeiro (Examinador Externo)
Instituto Federal de Santa Catarina

_____

Profº. Dr. Bruno José Torres Fernandes (Examinador Externo)
Universidade de Pernambuco

_____

Profº. Dr. Anthony Jose da Cunha Carneiro Lins (Examinador Externo)
Universidade Católica de Pernambuco

_____

Profº. Dr. Byron Leite Dantas Bezerra (Examinador Externo)
Universidade de Pernambuco

Dedico este trabalho a minha família.

AGRADECIMENTOS

Aos meus pais, Domingos Sávio Silva Monteiro e Maria das Neves de Paula Monteiro, à minha irmã, Camila de Paula Monteiro, e ao meu avô, Orlando Batista Monteiro, por todo o suporte e confiança que me foram dados.

À minha esposa, Paula Aragão de Souza, por sempre estar ao meu lado, nos momentos fáceis e difíceis, sempre com paciência e compreensão.

Ao meu orientador, Professor Dr. Carmelo José Albanez Bastos Filho, por todo o suporte e orientação desde os tempos de mestrado, pelas oportunidades de colaboração e pelo exemplo de professor, orientador e pesquisador que é.

Aos amigos do PPGES, que sempre estiveram presentes e dispostos a ajudar: Felipe San, Rodrigo Pequeno, Keila, Pedro Malandro, Mega, Marcão, Otávio e tantos outros.

A todos os professores que participaram da minha formação dentro e fora do programa de pós-graduação.

À CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), orgão financiador deste trabalho.

RESUMO

A detecção de anomalias consiste em identificar padrões que divirjam de comportamentos tidos como normais. Ela é uma importante área de estudos, cuja aplicabilidade se estende a diversos domínios, como a segurança de redes de comunicação, a detecção de doenças, a identificação de fraudes, dentre outros. A detecção de anomalias configura-se como uma etapa essencial de processos que envolvem tomadas de decisão, como o planejamento de manutenções em fábricas ou o início do tratamento de doenças graves. Comportamentos anômalos podem ser provocados por erros ou eventos ainda desconhecidos pelo sistema de detecção. A detecção de anomalias apresenta uma série de desafios que a distanciam de problemas de classificação tradicionais. Um deles diz respeito ao desbalanceamento dos dados disponíveis para treinar o modelo de detecção. A inexistência, ou existência em pequenas quantidades, de dados pertencentes às classes anômalas é bastante comum em problemas reais. Isto torna difícil a definição de uma região no espaço amostral que contenha todos os comportamentos normais possíveis, sem compreender os anômalos. Diversas técnicas foram desenvolvidas ao longo dos anos para tratar deste problema. No entanto, um grupo de técnicas ganhou atenção especial da comunidade acadêmica. Tal grupo baseia-se no uso de aprendizado profundo. Este consiste no processamento da informação através de múltiplas camadas, tornando possível a obtenção de representações mais significativas da informação de entrada para um dado problema de classificação ou regressão. Apesar dos avanços obtidos, o uso de aprendizado profundo na detecção de anomalias ainda apresenta algumas dificuldades, principalmente na obtenção de características capazes de representar de forma satisfatória a classe normal, ao mesmo tempo em que a diferencie da classe anômala. Este trabalho apresenta as etapas do desenvolvimento de um sistema para a detecção de anomalias baseado na atuação conjunta de técnicas profundas e tradicionais de aprendizado de máquina. As primeiras abordagens analisadas consistiram de algoritmos treinados de forma supervisionada, supondo-se que todas as classes anômalas eram conhecidas, visando promover um melhor entendimento acerca do problema. Em seguida, partiu-se para as abordagens nas quais o treinamento do algoritmo foi realizado apenas com dados pertencentes à classe normal. Dentre as técnicas propostas na tese, a de resultado mais promissor envolveu o treinamento do extrator de características baseado em aprendizado profundo conjuntamente com um processo de seleção de protótipos. A técnica apresentou valores médios de AUC relativamente altos e estáveis, *e.g.*, acima de 0,95, para um nicho de problemas de detecção de anomalias. Todos os modelos treinados foram avaliados com bases de dados compostas por espectrogramas de sinais sonoros e de vibração, coletados por sensores posicionados em dispositivos eletromecânicos.

Palavras-chave: detecção de anomalias; aprendizado profundo; seleção de protótipos; aprendizado de máquina.

ABSTRACT

The anomaly detection consists of identifying patterns that differ from an expected behavior. It is an important field of study, whose applicability extends to several domains, such as the security of communication networks, the detection of diseases and frauds, among others. The anomaly detection is an essential step in decision-making processes, such as planning the factory maintenance or starting the treatment of serious illnesses. Anomalous behaviors can be caused by errors in the process or by events not known by the detection system. Anomaly detection presents some challenges that makes it different from a traditional classification problem. One of them concerns the unbalance of the data available to train the anomaly detection model. The lack, or even the availability in small quantities of data belonging to the anomalous classes are quite common situations in real-world problems. It makes difficult to define a region in space that contains all possible normal behaviors without comprising the anomalies. Several techniques have been developed over the years to address this problem. However, a group of techniques has gained special attention from the academic community. Such techniques are based on the use of deep learning. They consist of processing information across multiple layers, making it possible to obtain more meaningful representations of the input information for a given classification or regression problem. Despite the advances achieved in this field, using deep learning in anomaly detection tasks still presents some difficulties, especially in obtaining characteristics capable of representing the normal class in a satisfactory way, while simultaneously distinguishing it from the anomalous class. This paper presents the development stages of an anomaly detection system based on the joint operation of deep and traditional machine learning techniques. The first approaches that we analyzed consisted of supervised trained algorithms, assuming that all anomalous classes were known, to promote a better understanding of the problem. Then, we proceeded to the approaches in which the training of the algorithm was performed only by using data belonging to the normal class. Among the techniques proposed in the thesis, the one with the most promising results regarded the training of a deep learning-based feature extractor together with a prototype selection process. The technique presented relatively high and stable mean AUC values, *e.g.*, above 0.95, for a niche of anomaly detection problems. The trained models were evaluated with databases composed of sound and vibration signal spectrograms, collected by sensors placed on electromechanical devices.

Keywords: anomaly detection; deep learning; prototype selection; machine learning.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AAE	*Autoencoder* Adversarial *Adversarial Autoencoder*
AE	*Autoencoder*
ANN	Redes Neural Artificial *Neural Network*
AUC	Área Sob a Curva *Area Under the Curve*
BN	Rede Bayesiana *Bayesian Network*
CANDE	*Context-Aware Novelty Detection*
CIFAR	Instituto Canadense para Pesquisa Avançada *Canadian Institute For Advanced Research*
CNN	Rede Neural Convolucional *Convolutional Neural Network*
DAE	*Autoencoder* Redutor de Ruído *Denoising Autoencoder*
DBN	Rede de Crença Profunda *Deep Belief Network*
DBSCAN	*Density-Based Spatial Clustering of Applications with Noise*
DFN	Rede *Feedforward* Profunda *Deep Feedforward Network*
DFT	Transformada Discreta de Fourier *Dicrete Fourier Transform*
EE	Envelope Elíptico *Elliptic Envelope*
EGBAD	Detecção de Anomalia Eficiente Baseada em GAN *Efficient GAN Based Anomaly Detection*

FFT	Transformada Rápida de Fourier *Fast Fourier Transform*
FT	Transformada de Fourier *Fourier Transform*
GAN	Rede Generativa Adversarial *Generative Adversarial Network*
GPU	Unidade Gráfica de Processamento *Graphical Processing Unit*
GTSRB	*Benchmark* de Reconhecimento de Sinalização de Trânsito Alemã *German Traffic Sign Recognition Benchmark*
IF	Floresta de Isolamento *Isolation Forest*
KNN	K-ésimo Vizinho Mais Próximos *K-th Nearest Neighbor*
LOF	Fator *Outlier* Local *Local Outlier Factor*
LSTM	Memória de Curto e Longo Prazo *Long Short-Term Memory*
MIMII	Investigação e Inspeção de Mau Funcionamento de Máquinas Industriais *Malfunctioning Industrial Machine Investigation and Inspection*
MLP	Percéptron de Múltiplas Camadas *Multilayer Perceptron*
MNIST	Instituto Nacional Modificado de Padrões e Tecnologia *Modified National Institute of Standards and Technology*
MSE	Erro Quadrático Médio *Mean Squared Error*
NN	Vizinhos Mais Próximos *Nearest Neighbors*
OC-NN	Rede Neural de Uma Classe *One-Class Neural Network*
OC-SVM	Máquina de Vetores de Suporte de Uma Classe *One-Class Support Vector Machine*

OEC	Detecção Exposta a *Outlier* *Outlier-Exposed Detection*
PCA	Análise de Componentes Principais *Principal Component Analysis*
RBM	Máquina Restrita de Boltzmann *Restricted Boltzmann Machine*
ReLU	Unidades Lineares Retificadas *Rectified Linear Unit*
RNN	Rede Neural Recorrente *Recurrent Neural Network*
ROC	Característica de Operação do Receptor *Receiver Operating Characteristics*
SAE	Autoencoders empilhados *Stacked Autoencoders*
STFT	Transformada de Fourier de Tempo Curto *Short-Time Fourier Transform*
SURF	*Speeded Up Robust Features*
SVM	Máquina de Vetores de Suporte *Support Vector Machine*
t-SNE	Vizinho Estocástico com Distribuição t *t-Distributed Stochastic Neighbor Embedding*
VAE	Autoencoder Variacional *Variational Autoencoder*

SUMÁRIO

1 INTRODUÇÃO

Nesta seção, faz-se uma breve introdução ao problema da detecção de anomalias, bem como aponta-se os principais desafios na área. Ao final, apresenta-se a proposta do estudo desenvolvido ao longo do doutorado, bem como seus objetivos geral e específicos.

1.1 DETECÇÃO DE ANOMALIAS

A detecção de anomalias é uma importante área de estudos, cuja aplicabilidade se estende a diversos domínios (CHALAPATHY; CHAWLA, 2019), *e.g.* segurança de redes (TIAN; LI; LIU, 2019), detecção de fraudes (RAZA; QAYYUM, 2019), detecção de falhas em dispositivos mecânicos (MONTEIRO; BASTOS-FILHO, 2019b), detecção de problemas em vias (LUO; LU; GUO, 2020) e detecção de avarias em produtos (JIANG; WANG; ZHAO, 2019). Ela consiste na identificação de padrões não conformes com aqueles ditos normais ou esperados para uma determinada aplicação (CHANDOLA; BANERJEE; KUMAR, 2009). A detecção de anomalias é uma etapa essencial para processos que envolvem tomadas de decisões, como o planejamento de manutenções em fábricas ou o início do tratamento de doenças graves. Sob a óptica dos sistemas de detecção de anomalias, os comportamentos anômalos podem resultar de erros, defeitos, ou mesmo de eventos ainda desconhecidos pelo sistema (CHALAPATHY; CHAWLA, 2019).

A detecção de *outliers* e de novidades são duas áreas intimamente relacionadas à detecção de anomalias, sendo frequentemente confundidas. No entanto, existem algumas diferenças teóricas. A detecção de novidades, por exemplo, consiste na identificação de padrões não observados previamente, mas que podem ser incorporados posteriormente ao conjunto dos padrões ditos normais. A de *outliers*, por outro lado, consiste na identificação de dados com baixa probabilidade de ocorrência e que possam prejudicar a análise estatística de um determinado conjunto de informações (CHANDOLA; BANERJEE; KUMAR, 2009).

1.2 PRINCIPAIS DESAFIOS DA DETECÇÃO DE ANOMALIAS COM APRENDIZADO PROFUNDO

Em seu trabalho, Chandola *et al*. (CHANDOLA; BANERJEE; KUMAR, 2009) enumeraram uma série de desafios para o desenvolvimento de algoritmos de detecção de anomalias. O primeiro deles é definir uma região do espaço amostral que contenha todos os comportamentos normais possíveis, sem compreender os anômalos. Determinar as fronteiras que separam as instâncias normais das anômalas de modo preciso não é uma tarefa trivial, principalmente quando dados anômalos e normais compartilham algumas características, ou quando todos os cenários anômalos possíveis não são conhecidos.

Outro desafio envolve lidar com anomalias adaptativas, *i.e.* anomalias que emulam o

comportamento de amostras normais. Este é o caso típico da detecção de intrusos em redes de comunicação. A principal dificuldade imposta por tais anomalias é estabelecer um padrão normal para o conjunto de dados, uma vez que as anomalias procuram imitar este padrão.

A mudança do conceito de padrão normal ao longo do tempo também se mostra uma questão desafiadora. A frequência cardíaca do ser humano é um exemplo disso. Quando o ser humano é um recém nascido, a sua frequência cardíaca varia de 120 a 140 batimentos por minuto. Para adultos, ela varia de 60 a 90 batimentos por minuto (REDACAO, 2019). Desta forma, o padrão de frequências utilizado para identificar doenças cardíacas em seres humanos, isto é, as anomalias, altera-se com o tempo. Isto torna impeditivo o estabelecimento de um padrão único para a análise da frequência cardíaca em seres humanos ao longo de toda a sua vida.

A noção de anomalia também pode variar de acordo com a aplicação. Por exemplo, um valor de temperatura pode ser considerado anômalo tendo-se como referência a temperatura corporal do ser humano. Em contrapartida, o mesmo valor pode ser considerado normal para um dado processo industrial. Isto dificulta o estabelecimento de técnicas universais para a detecção de anomalias. Ressalta-se ainda que presença de ruídos também configura um desafio à detecção de anomalias. Estes podem fazer dados normais apresentarem características de dados anômalos, bem como dados anômalos apresentarem características de dados normais, dificultando a distinção entre um grupo e outro. Tal problema torna necessário o uso de técnicas para tratar os dados ou algoritmos mais robustos para realizar a detecção de anomalias.

Algumas destas questões são particularmente desafiadoras para algoritmos baseados em aprendizado profundo. A definição de regiões no espaço amostral que contenham todos os comportamentos normais, mas não os anômalos, por exemplo, é um dos principais desafios. Como se sabe, algoritmos de aprendizado profundo modelam diferentes abstrações dos dados de entrada através de múltiplas camadas de processamento (GOODFELLOW; BENGIO; COURVILLE, 2016). Em problemas de classificação com múltiplas classes, por exemplo, o processo de modelagem dessas abstrações, *i.e.*, treinamento do modelo, é guiado pelas informações presentes nas diferentes classes. Desta forma, tais classificadores podem aprender representações de alto nível das classes ao analisar as semelhanças e diferenças entre os dados pertencentes às mesmas.

Por outro lado, em problemas de detecção de anomalias, frequentemente modelados como problemas de classificação de uma classe, é comum que a única informação disponível para se treinar os detectores esteja relacionada à classe normal. A falta de informações sobre as classes anômalas dificulta o aprendizado de representações de alto nível que discriminem amostras normais e anômalas de forma adequada (CHALAPATHY; CHAWLA, 2019). No Capítulo 2, são apresentados alguns grupos de técnicas de aprendizado profundo aplicadas à detecção de anomalias, bem como as formas que tais grupos lidam com a carência ou inexistência de amostras anômalas para treinar os detectores.

Apesar da variedade algoritmos de detecção de anomalias baseados em aprendizado profundo, existe uma carência de técnicas cujo processo de treinamento objetive o aprendizado

de representações de alto nível para a detecção de anomalias. Para uma quantidade significativa de técnicas, o treinamento dos detectores de anomalias visa atingir objetivos secundários, como reduzir o erro de reconstrução da informação de entrada, ou mesmo o erro na predição de uma série temporal. Por conta disso, esse processo de aprendizado é sub-ótimo por definição (CHALAPATHY; CHAWLA, 2019).

1.3 PROPOSTA

Ciente do cenário apontado em 1.2, este trabalho foca em detectores de anomalias que mesclam técnicas de aprendizado de máquina tradicional e profundo, *i.e.*, sistemas híbridos. Tais sistemas são constituídos por dois estágios: o extrator de características, normalmente baseado em aprendizado profundo, e o detector de anomalias propriamente dito, baseado em aprendizado de máquina tradicional. Este último é o responsável pela classificação da instância de teste como normal ou anômala, de acordo com uma avaliação das características extraídas pelo extrator.

A detecção de anomalias é abordada neste trabalho como um problema de classificação de uma classe. Em outras palavras, os classificadores são treinados com dados pertencentes apenas à classe normal, tendo como objetivo identificar se uma instância de teste pertence ou não à mesma classe dos dados utilizados para treinar o classificador. Esta forma de abordar o problema remete a um cenário real onde apenas se tenha informações sobre o comportamento normal de uma dada aplicação, *e.g.*, identificar falhas num dispositivo novo que ainda não tenha apresentado defeitos.

A principal contribuição deste trabalho está no treinamento do extrator de características, baseado em redes neurais convolucionais. O treinamento deve ser realizado objetivando melhorar o desempenho da detecção de anomalias, e não de objetivos secundários, *e.g.*, reconstrução da informação de entrada. Para isto, propõe-se um processo de treinamento para o extrator de características em conjunto com seletores de protótipos. A hipótese é que a seleção de protótipos ajude o extrator de características a mapear os dados pertencentes à classe normal para uma região mais restrita do espaço de características. Isto facilitaria a separação destas em relação àquelas pertencentes a dados anômalos, não utilizados para treinar o extrator, por parte de algoritmos de detecção de anomalias baseados em aprendizado de máquina.

Bases de dados relacionadas à detecção de defeitos e/ou diagnose de falhas em sistemas eletromecânicos são utilizadas para avaliar o desempenho da solução proposta, sendo este também comparado ao de técnicas de detecção de anomalias consolidadas da literatura, *e.g.*, *autoencoders* e máquinas de vetores de suporte de uma classe.

1.4 OBJETIVOS

Nesta seção estão listados os objetivos geral e específicos do estudo desenvolvido no doutorado.

Objetivo geral:

Desenvolver uma metodologia de treinamento para extratores de características baseados em aprendizado profundo, visando o mapeamento de características associadas a instâncias normais para regiões mais restritas do espaço de características, melhorando a separabilidade destas em relação a características associadas a instâncias anômalas.

Objetivos específicos:

Os objetivos específicos deste trabalho são:

1. Definir uma arquitetura para o extrator de características baseada em aprendizado profundo, de modo a condensar os dados de entrada em vetores de características de menor dimensionalidade, porém mantendo a relação com os mesmos;
2. Definir uma técnica de detecção de anomalias para identificar padrões anômalos nas características extraídas pelo extrator;
3. Definir um seletor de protótipos para auxiliar no treinamento do extrator de características;
4. Definir uma metodologia de treinamento conjunto do extrator de características com um seletor de protótipos, de modo que o processo de extração das características mapeie a informação de entrada para regiões mais restritas do espaço de características;
5. Testar e avaliar a metodologia de treinamento proposta em bases de dados de detecção de defeitos e diagnóstico de falhas em dispositivos industriais.

1.5 ORGANIZAÇÃO DO TRABALHO

O documento inclui uma série de trabalhos desenvolvidos ao longo do doutorado, visando atingir o objetivo geral. Tais trabalhos auxiliaram no entendimento dos assuntos estudados e na realização de escolhas, *e.g.*, técnicas, organização dos dados, etc. A organização dos demais capítulos deste documento é vista a seguir:

- Capítulo 2 – É realizado um levantamento das técnicas do estado da arte de detecção de anomalias. Também discute-se suas forças e fraquezas;
- Capítulo 3 - É realizado um estudo sobre a seleção de dados oriundos de sensores diferentes com o auxílio de medidas de dissimilaridade e testes estatísticos. Também se avalia a influência de diferentes formatos desses dados na acurácia dos classificadores treinados com os mesmos;
- Capítulo 4 – É apresentada a primeira etapa de desenvolvimento do sistema para detectar anomalias. Esta consistiu no uso de redes neurais convolucionais para diagnosticar falhas em sinais de vibração;

- Capítulo 5 – Nesta etapa, analisou-se a eficácia de um estágio de decisão para análise das saídas do classificador baseado em redes neurais convolucionais desenvolvido na primeira etapa;
- Capítulo 6 – É apresentada a terceira etapa do desenvolvimento do detector de anomalias. Esta consistiu em testar um novo modelo de treinamento para o extrator de características que, em conjunto com o OC-SVM, detecta anomalias em sinais de vibração;
- Capítulo 7 – É apresentada a quarta etapa do desenvolvimento do detector de anomalias. Esta consistiu em aprimorar o processo de aprendizado do modelo apresentado no Capítulo 6, que desta vez ocorre em conjunto com um seletor de protótipos;
- Capítulo 8 - Neste capítulo, realiza-se uma análise comparativa da solução proposta com técnicas e bases de dados da literatura, bem como discussões acerca do nível de desempenho atingido e a outros aspectos relacionados;
- Capítulo 9 – Apresenta a Conclusão.

2 ESTADO DA ARTE

A detecção de anomalias não é algo recente. Ela é objeto de estudo frequente em trabalhos da literatura mesmo antes do advento do aprendizado profundo (CHANDOLA; BANERJEE; KUMAR, 2009). As técnicas de detecção de anomalias baseiam-se nos mais variados princípios, *e.g.* aprendizados de máquina tradicional e profundo; aprendizados supervisionado, semi e não supervisionado, classificadores tradicionais e de uma classe, dentre outros.

2.1 DETECÇÃO DE ANOMALIAS COM TÉCNICAS TRADICIONAIS

Dentre as técnicas tradicionalmente utilizadas na detecção de anomalias, seis grupos de abordagens podem ser identificados (CHANDOLA; BANERJEE; KUMAR, 2009). O primeiro deles é constituído por técnicas baseadas em classificadores, *e.g.* redes neurais artificiais (ANN, do inglês *neural network*) (BISHOP et al., 1995), redes bayesianas (BN, do inglês *bayesian network*) (FRIEDMAN; GEIGER; GOLDSZMIDT, 1997) e máquinas de vetores de suporte (SVM, do inglês *support vector machine*) (HSU et al., 2003). Tais técnicas envolvem o treinamento de modelos com dados rotulados, que identificam a classe de uma determinada instância. Os problemas abordados podem apresentar duas ou múltiplas classes. Detectores de anomalias baseados em classificadores tendem a utilizar algoritmos mais complexos se comparados a outros tipos de abordagens. Por outro lado, eles são relativamente rápidos, uma vez que a amostra testada é comparada apenas ao modelo treinado. A principal desvantagem deste tipo de abordagem é a necessidade de dados rotulados de maneira precisa, a fim de garantir a confiabilidade dos modelos treinados. Seguindo a linha do uso de classificadores na detecção de anomalias, o trabalho de Xiang *et al.* (XIANG; LIU; ZHANG, 2020) utiliza o SVM de uma classe (OC-SVM, do inglês *One-Class Support Vector Machine*) para detectar inconsistências nos relatórios de inspeção de transformadores. Outro exemplo é o trabalho de Mulongo *et al.* (MULONGO et al., 2020), que utiliza o Percéptron de Múltiplas Camadas (MLP, do inglês *Multilayer Perceptron*) e outros algoritmos para detectar anomalias no consumo de combustível em usinas de geração de energia.

O segundo grupo de abordagens se baseia na distância, ou similaridade, entre as amostras de uma determinada distribuição, *e.g.* k-ésimo vizinho mais próximo (KNN, do inglês *K-th Nearest Neighbor*) (RAMASWAMY; RASTOGI; SHIM, 2000) e fator *outlier* local (LOF, do inglês *local outlier factor*) (BREUNIG et al., 2000). Tal grupo pressupõe que as instâncias normais são encontradas em vizinhanças densamente povoadas, enquanto as anômalas se encontram isoladas. As técnicas requerem o uso de uma medida para aferir a distância ou similaridade entre duas instâncias, *e.g.* a distância euclidiana. O treinamento deste grupo de técnicas costuma ser semi ou não supervisionado, sendo ele puramente orientado aos dados. As técnicas deste grupo são facilmente adaptáveis a diferentes tipos de dados. Por outro lado, uma das principais

desvantagens apresentadas por este grupo de técnicas é a complexidade computacional, pois o teste de uma nova amostra normalmente envolve o cálculo da distância desta a todas as amostras de treino e/ou outras amostras de teste. O desempenho das técnicas também depende da medida de distância utilizada. Dois trabalhos que exemplificam o uso de técnicas pertencentes a este grupo são (SARMADI; KARAMODIN, 2020) e (YU et al., 2020). No primeiro, Yu *et al.* utilizam filtros baseados em LOF para detectar anomalias em dados hiperespectrais. Em contrapartida, Sarmadi *et al.* utilizam o KNN associado à distância quadrática de Mahalanobis no monitoramento das condições de estruturas físicas, *e.g.*, pontes de madeira. O trabalho desenvolvido por (VIJAYAKUMAR et al.,) aborda o uso de florestas de isolamento (IF, do inglês *isolation forest*) e LOF na detecção de fraudes em cartão de crédito.

Um terceiro grupo de abordagens baseia-se em algoritmos de agrupamento, ou clusterização, para detectar anomalias. Tais técnicas são utilizadas para agrupar instâncias de dados que apresentam características semelhantes. A detecção de anomalias pode ocorrer de acordo com diferentes estratégias. Uma delas pressupõe que os dados normais pertencem a *clusters*, ou agrupamentos, enquanto dados anômalos não pertencem. Esta abordagem requer o uso de técnicas que permitam que partículas não pertençam a nenhum dos agrupamentos formados, *e.g.* DBSCAN (do inglês *density-based spatial clustering of applications with noise*) (ESTER et al., 1996). Outra estratégia pressupõe que dados normais encontram-se próximos aos centróides dos *clusters* aos quais pertencem, enquanto dados anômalos estão localizadas nas periferias dos mesmos. Esta abordagem, ao contrário da anterior, permite o uso de técnicas que forcem todas as partículas a pertencerem a pelo menos um *cluster*, como o k-médias (LLOYD, 1982). O trabalho de Sheridan *et al.* (SHERIDAN et al., 2020) exemplifica o uso deste tipo de abordagem na detecção de anomalias. Os autores utilizaram o DBSCAN e clusterização hierárquica na identificação de voos com características anômalas. Outro trabalho que utiliza o DBSCAN na detecção de anomalias é o de Dokuz *et al.* (DOKUZ; ÇELIK; ECEMIŞ, 2020), porém em variações de preços de Bitcoins. O trabalho desenvolvido por Wang *et al.* (WANG; ZHOU; LI, 2020), por outro lado, baseia-se no uso do k-médias para identificar anomalias em dados na forma de *streaming* e lotes. Dentre as vantagens de se utilizar técnicas baseadas em clusterização para detectar anomalias, pode-se citar: prescindibilidade de rótulos, facilidade de adaptação a diferentes tipos de distribuições de dados e rapidez em determinar se uma nova instância pertence ou não ao conjunto de instâncias normais. Dentre as desvantagens, existem: falta de técnicas otimizadas para a detecção de anomalias, custo computacional elevado para treinamento dos modelos e a possibilidade de dados anômalos formarem agrupamentos com outros dados anômalos.

Técnicas estatísticas também são utilizadas na detecção de anomalias, configurando um quarto grupo de abordagens. Elas consistem no estabelecimento de um modelo estatístico (MCCULLAGH, 2002) para a distribuição de dados. Tais técnicas partem do pressuposto que instâncias normais ocorrem nas regiões de alta probabilidade de um modelo estocástico, enquanto instâncias anômalas ocorrem nas regiões de baixa probabilidade. As técnicas estatísticas podem

ou não ser paramétricas. As primeiras assumem a existência de um conhecimento prévio sobre a estrutura do modelo que gera as amostras normais, *e.g.* modelos gaussianos e modelos de regressão. Por outro lado, as técnicas não-paramétricas assumem que o conhecimento prévio não existe, sendo o modelo construído a partir da própria distribuição, *e.g.* modelos baseados em histogramas. O *score* de anomalia deste grupo de abordagens encontra-se associado a um intervalo de confiança, que pode ser utilizado como informação adicional em tomadas de decisões. A principal desvantagem das técnicas estatísticas é que elas assumem que dados são gerados a partir de determinadas distribuições, o que nem sempre se concretiza. Este problema ganha importância com o aumento da dimensionalidade dos dados. A captura de interações entre as variáveis também pode se tornar um problema, principalmente considerando as técnicas não paramétricas. Um exemplo de pesquisa que emprega modelos estatísticos na detecção de anomalias é a de Shylendra *et al.* (SHYLENDRA et al., 2020). Eles utilizaram estimativas de densidades não paramétricas baseadas em *kernels* para gerar modelos estatísticos em tempo real de dados de sensores. Tais modelos foram utilizados na identificação de *outliers* a partir do desvio padrão do *kernel* e do limiar de probabilidade.

O quinto grupo de abordagens compreende o emprego de teoria da informação na detecção de anomalias. Neste grupo, assume-se que as anomalias induzem irregularidades na informação contida nos conjuntos de dados. Desta forma, métricas como entropia, entropia relativa e complexidade de Kolmogorov (BORDA, 2011), podem ser utilizadas para analisar a informação contida nos dados e inferir a presença ou ausência de comportamentos anômalos. O trabalho de Nguyen-An *et al.* (NGUYEN-AN et al., 2020) exemplifica o uso deste tipo de abordagem na detecção de anomalias. Eles calculam a entropia dos parâmetros de tráfego de dispositivos de Internet das coisas, identificando anomalias através da comparação dos valores de entropia obtidos por sistemas reais e gerados de forma sintética. Gu *et al.* (GU; CHOW; SHAO, 2020) também utilizam a entropia para este fim, porém associada a algoritmos que codificam esparsamente a informação analisada. O emprego de teoria da informação na detecção de anomalias não necessita de dados rotulados, podendo operar de maneira não supervisionada. Ela também não faz suposições acerca da distribuição estatística dos mesmos. Dentre os problemas apresentados por este tipo de abordagem, estão: desempenho altamente dependente da métrica utilizada e dificuldade de relacionar o *score* de anomalia à instância de teste por meio de técnicas da teoria da informação.

O último grupo de abordagens tradicionais baseia-se em técnicas espectrais. Tais técnicas buscam encontrar uma aproximação dos dados originais por meio de uma combinação de atributos que capturem a maior parte da variabilidade dos dados. Técnicas espectrais partem do pressuposto que os dados podem ser representados em subespaços de menor dimensionalidade, nos quais as instâncias normais e anômalas difiram significativamente. Um exemplo comumente encontrado em trabalhos da literatura é a análise de componentes principais (PCA, do inglês *principal component analysis*) (WOLD; ESBENSEN; GELADI, 1987; DEFREITAS et al., 2020). Este grupo de técnicas pode ser utilizado de forma não-supervisionada, não sendo obrigatória a

existência de rótulos. Também são bastante utilizados como uma etapa de preprocessamento de dados em problemas de alta dimensionalidade. Por outro lado, tais técnicas tendem a apresentar alta complexidade computacional e são úteis somente quando as instâncias normais e anômalas são separáveis no subespaço com dimensionalidade reduzida.

Vale ressaltar que as técnicas tradicionais continuam sendo empregadas numa grande variedade de problemas relacionados à detecção de anomalias, mesmo com a ascensão das abordagens de arquiteturas profundas. Porém, quando não associadas a outros tipos de algoritmos, a sua aplicação é normalmente restrita a problemas de baixa dimensionalidade, com dados em menores quantidades e distribuições de menor complexidade (CHALAPATHY; CHAWLA, 2019). Tal restrição tende a ser um empecilho a muitas aplicações no cenário atual, dada a quantidade de informações disponíveis e geradas a todo momento.

2.2 DETECÇÃO DE ANOMALIAS COM TÉCNICAS PROFUNDAS

Testemunhou-se na última década a expansão do aprendizado profundo para diversos domínios de aplicação, *e.g.* classificação de imagens (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), processamento de linguagem natural (YOUNG et al., 2018), predição de séries temporais (KUREMOTO et al., 2014), dentre outros (CHALAPATHY; CHAWLA, 2019). O aprendizado profundo é uma subárea do aprendizado de máquina, caracterizado pelo processamento da informação através de modelos com múltiplas camadas. Modelos de aprendizado profundo mostram-se capazes de obter representações implícitas das informações de entrada ao longo de suas camadas internas. Desta forma, eles são capazes de gerar diferentes níveis de abstração de um dado problema sem a necessidade de interferência humana (GOODFELLOW; BENGIO; COURVILLE, 2016). O desempenho de técnicas baseadas em aprendizado profundo mostrou-se superior ao de abordagens tradicionais de aprendizado de máquina, como as máquinas de vetores de suporte e redes bayesianas, principalmente em problemas relacionados a imagens e grandes volumes de dados (CHALAPATHY; CHAWLA, 2019; GOODFELLOW; BENGIO; COURVILLE, 2016).

Apesar do aprendizado profundo ter melhorado substancialmente o desempenho de aplicações baseadas em aprendizado de máquina, o seu emprego na detecção de anomalias ainda não se encontra tão difundido quanto em aplicações como a detecção de objetos e a regressão, por exemplo. A principal razão está na dificuldade de se treinar modelos discriminativos o suficiente quando toda a informação disponível pertence apenas a uma das classes, ou quando o nível de desbalanceamento das classes é alto (CHALAPATHY; CHAWLA, 2019).

Analisando as técnicas comumente encontradas na literatura, considerando apenas aquelas que utilizam o aprendizado profundo para a detecção de anomalias, pode-se destacar alguns grupos de soluções.

O primeiro deles utiliza modelos treinados com dados de apenas uma classe, ou mesmo

de forma não supervisionada, para reconstruir a informação fornecida na entrada. Desta forma, a anomalia é detectada por meio da análise do erro de reconstrução (RAZA; QAYYUM, 2019; KOIZUMI et al., 2018; NAKAZAWA; KULKARNI, 2019; TULUPTCEVA et al., 2020). O *autoencoder* (AE) tradicional e suas variações são exemplos de técnicas utilizadas para este fim. Esta abordagem parte do pressuposto que a maioria ou totalidade dos dados disponíveis para treinar o modelo pertencem à classe normal. Portanto, o modelo tende a aprender de forma precisa apenas a reconstrução dos dados pertencentes à classe normal, de modo que o erro da reconstrução destes seja baixo. Por outro lado, o erro de reconstrução de amostras pertencentes às classes anômalas tende a ser alto. A definição de um limiar adequado para o erro de reconstrução é essencial para que este tipo de solução obtenha sucesso na detecção de anomalias. Uma estratégia bastante comum para a obtenção do limiar consiste em fixar um percentil, *e.g.* 95, para a distribuição obtida com os erros de reconstrução dos dados de treinamento.

Um exemplo de trabalho recente baseado neste tipo de abordagem foi desenvolvido por Raza e Qayyum (RAZA; QAYYUM, 2019). Eles propuseram o uso de *autoencoders* variacionais (VAE, do inglês *variational autoencoder*) baseados em redes neurais artificiais para detectar fraudes em cartões de crédito. Os *autoencoders* foram treinados apenas com dados normais, de modo que aprendessem apenas a reconstruí-los. Desta forma, o erro de reconstrução para dados anômalos tendeu a ser maior. Eles também realizaram comparações com outros classificadores clássicos, como: árvores de decisão, máquinas de vetores de suporte, dentre outros. O modelo proposto apresentou resultados promissores de *recall*, porém as demais técnicas apresentaram resultados significativamente melhores em termos de precisão.

Outra pesquisa que segue esta abordagem foi desenvolvida por Koizumi *et al*. (KOIZUMI et al., 2018). Eles propuseram uma técnica não supervisionada para treinar sistemas de detecção de sons anômalos. O modelo utilizado como base também foi um *autoencoder*. As anomalias são detectadas através do erro de reconstrução dos dados. Eles definiram uma função objetivo baseada no lema de Neyman-Pearson, considerando a detecção de sons anômalos como sendo um teste estatístico de hipótese. O *autoencoder* é treinado para maximizar a taxa de verdadeiros positivos, dada uma baixa condição de falsos negativos, arbitrariamente definida. Os sons anômalos são simulados com o uso de uma segunda rede neural e de amostragem por rejeição. A técnica proposta foi capaz de melhorar o desempenho do *autoencoder*, bem como superar o de técnicas como o VAE. O sistema também foi capaz de funcionar na detecção de sons anômalos em ambientes reais.

Um terceiro exemplo é o trabalho desenvolvido por Tuluptceva *et al.* (TULUPTCEVA et al., 2020). Eles redesenharam o *pipeline* de treinamento do *autoencoder* tradicional para detectar anomalias em imagens médicas de alta resolução, *e.g.*, raios-X e imagens de microscópio. Um dos pontos principais do trabalho é o uso de uma função de perda que foca na reconstrução de padrões ou características extraídas das imagens, e não na reconstrução destas últimas propriamente. As características são acrescentadas gradualmente ao processo de treinamento de acordo com

a sua profundidade. O segundo ponto é o uso de imagens anômalas em pequenas quantidades no processo de treinamento, de modo a permitir um melhor ajuste dos parâmetros do detector de anomalias. A solução proposta foi avaliada em bases de dados com imagens médicas e com imagens normais, superando o desempenho de algoritmos comuns à área, como o Deep GEO (GOLAN; EL-YANIV, 2018) e o Deep IF (OUARDINI et al., 2019).

Apesar desta ser uma abordagem frequente em trabalhos da literatura, ela pode falhar quando dados normais e anômalos compartilham semelhanças. Isto ocorre porque, sendo as características de ambas as classes semelhantes, o erro de reconstrução dos dados anômalos também será baixo. Desta forma, torna-se difícil estabelecer um limiar para o erro de reconstrução, de modo que as classes normal e anômala sejam separadas de maneira satisfatória (MONTEIRO; BASTOS-FILHO, 2019b).

Uma lógica semelhante é utilizada na detecção de eventos anômalos em séries temporais. Neste tipo de abordagem, os modelos são treinados para realizar a predição de valores futuros da série (RUSHE; NAMEE, 2019; MUNIR et al., 2018; MUNOZ-ORGANERO, 2019; PARK; PARK; CHOI, 2020). As redes neurais recorrentes (RNN, do inglês *recurrent neural networks*) e redes neurais com memórias de curto e longo prazo (LSTM, do inglês *long-short time memory*) são exemplos de técnicas utilizadas para este fim. Esta solução também parte da hipótese que a maioria ou totalidade dos dados disponíveis para o treinamento do modelo pertence à classe normal. Desta forma, o modelo treinado tende a predizer corretamente apenas valores de eventos pertencentes à classe normal. O erro de predição é o critério utilizado para identificar se um determinado evento é ou não anômalo. Eventos anômalos tendem a apresentar erros de predição maiores em relação aos eventos normais. Assim como para abordagens relacionadas ao erro de reconstrução, a definição de um limiar adequado é fundamental para o desempenho do modelo.

O trabalho desenvolvido por Rushe e Namee (RUSHE; NAMEE, 2019) promoveu a adaptação da WaveNet (OORD et al., 2016) para o problema de detecção de anomalias em áudios. A WaveNet é um modelo autorregressivo convolucional originalmente destinado à geração de áudios. O detector de anomalias foi treinado para prever a amostra seguinte de uma sequência utilizando apenas dados normais, fazendo com que a rede aprendesse uma distribuição condicional sobre as sequências normais. As sequências anômalas, portanto, não seguiam essa distribuição, resultando em predições mais distantes do valor real da amostra. Essa distância, calculada em termos de erro quadrático médio (MSE, do inglês *mean squared error*), é utilizada por Rushee e Namee para identificar se a sequência é normal ou anômala. Experimentos realizados em 17 bases de dados mostraram que o modelo proposto obteve ganhos de desempenho em relação a técnicas robustas e amplamente utilizadas na detecção de anomalias, como os *autoencoders* convolucionais.

Um segundo trabalho baseado nesta abordagem foi apresentado por Munir *et al*. (MUNIR et al., 2018). Eles apresentaram uma solução baseada em aprendizado profundo para a detecção de anomalias em séries temporais, a qual chamaram DeepAnT. A técnica proposta é capaz de

detectar diferentes tipos de anomalias, *e.g.* anomalias contextuais e pontuais, utilizando dados não rotulados. O sistema é composto por dois módulos: o preditor de séries temporais, que utiliza uma rede neural convolucional para predizer o valor seguinte da série, e um detector de anomalias, que classifica esse valor predito como normal ou anômalo ao compará-lo com o valor real da série, tendo como base a distância euclidiana. A escolha do limiar baseou-se no desvio padrão e na distribuição de densidades dos valores preditos. A técnica mostrou desempenho equivalente ou superior a outras técnicas do estado da arte, *e.g.* redes LSTMs, nas bases de dados utilizadas.

Outro exemplo de estudo baseado neste tipo de abordagem foi desenvolvido por Park *et al.* (PARK; PARK; CHOI, 2020). Eles utilizaram redes neurais do tipo LSTM na detecção de intrusos em aplicações de Internet das coisas. Tais redes foram treinadas para realizar a predição de pacotes de informações, que eram comparados a pacotes, reais objetivando identificar as anomalias. Esta comparação foi realizada com métricas de distância, como a distância euclidiana e a similaridade por cosseno. Os resultados apresentados pela solução proposta foram superiores ao de técnicas tradicionalmente utilizadas em mineração de dados, como o XGBoost e a floresta aleatória.

Embora seja uma abordagem adequada à detecção de anomalias em séries temporais, ela pode apresentar problemas semelhantes aos da abordagem que utiliza o erro de reconstrução. Em outras palavras, quando dados normais e anômalos apresentam semelhanças, o erro de predição de dados anômalos pode ser próximo ao de dados normais. Isto torna difícil o estabelecimento de um limiar adequado para o erro de predição, de modo a separar as classes normal e anômala.

Uma terceira abordagem, também frequentemente encontrada em trabalhos da literatura (TIAN; LI; LIU, 2019; SINGH; MOHAN, 2018; VARTOUNI; TESHNEHLAB; KASHI, 2019; PEREIRA; SILVEIRA, 2019; MÜLLER et al., 2020), consiste em utilizar modelos híbridos para realizar a detecção de anomalias. Tais modelos normalmente são constituídos por um extrator de características baseado em aprendizado profundo e um detector de anomalias tradicional. O extrator de características, *e.g.* uma máquina restrita de Boltzmann (RBM, do inglês *restricted Boltzmann machine*), é treinando de forma não supervisionada para reduzir a dimensionalidade da informação de entrada, *e.g.* imagens, com o intuito de fornecer informações resumidas e relevantes para um detector de anomalias propriamente dito, como a máquina de vetores de suporte de uma classe.

Singh e Mohan (SINGH; MOHAN, 2018) propuseram um dos trabalhos que seguem esta abordagem. Trata-se de um *framework* para a detecção de acidentes rodoviários em vídeos de monitoramento. Os acidentes são considerados eventos anômalos neste *framework*. Uma parte do processo consiste em extrair representações profundas de vídeos de tráfego normal com o uso de *autoencoders* redutores de ruídos (DAE, do inglês *denoising autoencoders)*. A possibilidade de acidente é determinada pelo erro de reconstrução dos *frames*, utilizando os *autoencoders*, e pela distribuição das representações obtidas. Para se determinar a possibilidade de anomalia através

das representações, utilizou-se o OC-SVM, treinado apenas com representações pertencentes a vídeos normais. A fim de reduzir a quantidade de alarmes falsos, utilizou-se a análise das trajetórias dos veículos. Os resultados obtidos mostraram-se eficazes na detecção de acidentes, que foi avaliada em bases de dados contendo vídeos reais de acidentes.

Vartouni *et al.* (VARTOUNI; TESHNEHLAB; KASHI, 2019) propuseram o uso de técnicas baseadas em redes neurais profundas e aprendizado de máquina tradicional para a detecção de intrusos em redes *web*. Dentre as técnicas utilizadas no aprendizado de características e na redução de dimensionalidade, estão os *autoencoders* empilhados (SAE, do inglês *stacked autoencoders*) e as redes de crença profunda (DBN, do inglês *deep belief network*). As características aprendidas são utilizadas por algoritmos como o OC-SVM, a floresta de isolamento e o envelope elíptico (EE, do inglês *elliptic envelope*) para a detecção de anomalias. O sistema foi treinado apenas com dados pertencentes à classe normal. Os resultados demonstram que o aprendizado hierárquico e a fusão de características contribuem para a melhoria da acurácia e da generalização do modelo de detecção de anomalias, que permanece operando em tempo hábil.

O trabalho de Müller *et al.* (MÜLLER et al., 2020) também utiliza esta abordagem. No entanto, a extração de características é realizada por redes neurais convolucionais pré-treinadas para a classificação de imagens, como a ResNet (HE et al., 2016), a AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) e a SqueezeNet (IANDOLA et al., 2016). Todos estes modelos foram treinados na base ImageNet (DENG et al., 2009). Tais características são utilizadas no treinamento de algoritmos de detecção de anomalias, *e.g.*, OC-SVM e floresta de isolamento. Os resultados sugerem uma melhoria no desempenho da detecção de anomalias em máquinas industriais em relação ao uso de *autoencoders* convolucionais na extração de características.

A principal desvantagem deste tipo de abordagem é que, mesmo apresentando bons resultados em relação a outros grupos de técnicas, ela é sub-ótima por definição. Isto ocorre porque os detectores de anomalias propriamente ditos não influenciam no aprendizado das representações em camadas internas dos extratores de características, dado que o treinamento do extrator de características e do detector de anomalias é realizado separadamente. Em outras palavras, o processo de extração de características não é projetado para a detecção de anomalias, mas sim para outros objetivos, *e.g.*, a reconstrução da informação de entrada.

Classificadores treinados de forma supervisionada também são utilizados na detecção de anomalias (XU; WU; BIE, 2018; SRIVASTAVA et al., 2019; LUO; LU; GUO, 2020). Diferentemente das abordagens vistas anteriormente, esta exige algum conhecimento prévio das classes anômalas, *i.e.* todo o conjunto de dados deve ser rotulado. Neste cenário, cada classe corresponde a uma modalidade de dado normal ou anômalo. Um exemplo de técnica pertencente a este grupo é a rede neural convolucional (CNN, do inglês *convolutional neural network*). A principal vantagem deste tipo de abordagem é a maior facilidade para se aprender características relevantes ao processo classificatório, dado que o treinamento do modelo conta com informações de todas as classes envolvidas.

Seguindo esta abordagem, Xu *et al*. (XU; WU; BIE, 2018) propuseram um *framework* baseado em redes neurais convolucionais hierárquicas para a detecção de padrões anômalos em imagens de raio-X peitoral. A base de dados contém imagens de raio-X com a presença ou ausência de doenças. O desempenho de tal solução foi superior ao de técnicas que envolvem *fine tuning*, em termos de desempenho e compactação. Eles também apresentaram uma nova função de perda: a *sin-loss*. Esta permite um melhor aprendizado de informações discriminativas em imagens erroneamente classificadas ou indistinguíveis por meio de um parâmetro autoadaptável.

Srivastava *et al*. (SRIVASTAVA et al., 2019) empregaram redes neurais convolucionais na detecção de doenças intestinais em imagens de biópsias de alta resolução. O processo foi tratado como um problema de classificação comum, no qual três classes foram definidas. Uma classe correspondeu ao cenário normal, enquanto as demais estavam relacionadas às doenças a serem detectadas. A quantidade limitada de imagens tornou necessário o uso de técnicas de *data augmentation*, como: rotação, espelhamento, etc. O uso de CNN mostrou-se eficaz para a aplicação proposta, atingindo uma acurácia superior a 96% no processo de classificação.

Luo *et al.* (LUO; LU; GUO, 2020) também estudaram a detecção de anomalias sob a óptica do aprendizado supervisionado. Em seu trabalho, eles modelaram a detecção de anomalias em rodovias como um problema de classificação, no qual os classificadores foram treinados com dados rotulados. Três modelos de aprendizado profundo foram estudados: a CNN, a rede *feedforward* profunda (DFN, do inglês *deep feedforward network*) e a RNN. Tais modelos foram treinados com dados coletados pelos sensores de um veículo de testes, dirigido em rodovias com diferentes condições de anomalias. Os autores também analisaram a influência de dados de diferentes quantidades de sensores no aprendizado dos classificadores. As redes neurais recorrentes apresentaram os melhores desempenhos, atingindo valores de acurácias superiores a 99% quando treinados com dados de todos os sensores disponíveis.

Conforme mencionado anteriormente, uma limitação deste tipo de abordagem é a necessidade do uso de dados rotulados, uma vez que o treinamento do modelo de classificação é realizado de forma supervisionada. O balanceamento das classes é outra condição necessária, de modo a evitar que o modelo apresente viés. Tais problemas limitam o uso desta abordagem na detecção de anomalias, dados os desafios apresentados na subseção 1.2.

Outro tipo de abordagem, com desenvolvimento relativamente recente, é o das redes neurais de uma classe (OC-NN, do inglês *one-class neural network*) (CHALAPATHY; MENON; CHAWLA, 2018; OZA; PATEL, 2018). O princípio que rege o funcionamento de tal abordagem costuma variar de modelo para modelo. Um exemplo é o trabalho de Chalapathy *et al*. (CHALAPATHY; MENON; CHAWLA, 2018), que propôs uma rede neural de uma classe para a detecção de anomalias. Ela combina a habilidade de redes neurais profundas para a extração de características, com uma função objetivo de uma classe para criar uma superfície de decisão em torno dos dados normais. O diferencial desta solução é a representação da informação ao longo das camadas ocultas, cujo aprendizado é guiado pelo objetivo de uma classe e, consequentemente,

customizado para a detecção de anomalias. O desempenho da rede foi avaliado em bases de dados como a CIFAR (do inglês *Canadian Institute For Advanced Research*) (KRIZHEVSKY; NAIR; HINTON,) e a GTSRB (do inglês *German Traffic Sign Benchmarks*) (STALLKAMP et al., 2012), sendo semelhante ao de técnicas do estado da arte e superando o de técnicas com arquitetura rasa em alguns cenários, como o OC-SVM.

Outro exemplo desta abordagem foi apresentado por Oza e Patel (OZA; PATEL, 2018). Eles propuseram uma rede neural convolucional adaptada ao problema de classificação de uma classe. O treinamento da rede utiliza ruído gaussiano centrado em zero como conjunto de características pertencentes a uma classe pseudo-negativa. As características da classe positiva são extraídas pela base convolucional da CNN. Tal abordagem permite a utilização de bases convolucionais extraídas de CNN previamente treinadas. O desempenho do modelo proposto foi avaliado em uma variedade de conjuntos de dados que remetem a problemas de uma classe, como autenticação de usuários e detecção de novidades.

As redes generativas adversariais (GAN, do inglês *generative adversarial networks*) (GOODFELLOW et al., 2014) também passaram a ser utilizadas recentemente na detecção de anomalias. A forma de treinamento competitivo entre o modelo gerador e o discriminador aumenta a capacidade das GANs de aprender distribuições de dados pertencentes aos mais variados domínios, *e.g.*, imagens e sons. Este fato justifica o aumento do uso de GANs em diversas aplicações de aprendizado profundo, sendo particularmente interessante para problemas com desbalanceamento de classes ou ausência de rótulos, como a detecção de anomalias.

A forma como as GANs são utilizadas na detecção de anomalias pode variar de aplicação para aplicação. Algumas abordagens utilizam estas redes para aumentar a quantidade de dados normais disponíveis para o treinamento de modelos de detecção de anomalias, como a desenvolvida por Lim *et al*. (LIM et al., 2018). Eles utilizam *autoencoders* adversariais (AAE, do inglês *adversarial autoencoders*), uma variação das GANs, para gerar imagens normais de baixa probabilidade, responsáveis por falsos positivos na detecção de anomalias. Por outro lado, outras abordagens utilizam as próprias GANs para realizar a detecção de anomalias. Li *et al*. (LI et al., 2018) propuseram um modelo para detectar anomalias baseado em dois aspectos: a diferença entre a saída do gerador e a imagem real, e a perda do discriminador. Este conceito é semelhante ao utilizado por Schlegl *et al.* (SCHLEGL et al., 2017). Em vez da perda do discriminador, este último trabalho compara as características das imagens geradas e das imagens reais, observadas numa determinada camada do discriminador, *e.g.*, a última camada oculta.

Além desses dois aspectos, *i.e.*, o erro associado ao gerador e a perda associada ao discriminador, existem abordagens que também utilizam a representação dos dados num espaço latente de características não associado a nenhuma das duas estruturas. Este é o caso da GANomaly (AKCAY; ATAPOUR-ABARGHOUEI; BRECKON, 2018). Os autores deste estudo propuseram o treinamento da GAN em conjunto com um modelo de codificação (*encoder*), de modo que este seja capaz de criar uma representação dos dados que sirva tanto para a definição

de um *score* de anomalias quanto para auxiliar o modelo gerador na construção de informações mais realistas. Num estudo comparativo realizado pelos autores, a GANomaly aprensentou os melhores resultados em bases de dados como MNIST (do inglês *Modified National Institute of Standards and Technology*) (LECUN; CORTES, 2010) e CIFAR, superando outros modelos baseados em GANs como a AnoGAN (SCHLEGL et al., 2017) e a EGBAD (do inglês *efficient gan-based anomaly detection*) (ZENATI et al., 2018).

Assim como os demais grupos de soluções, o trabalho com GANs pode apresentar algumas dificuldades. Um dos problemas comumente associados a GANs é a sensibilidade aos hiperparâmetros selecionados, que por sua vez pode resultar em problemas de convergência. Outro problema que pode ocorrer é o desbalanceamento entre discriminador e gerador, que pode resultar em *overfitting* ou na redução do gradiente do gerador a zero, impossibilitando o processo de aprendizado por parte do gerador.

3 IDENTIFICANDO DADOS PROMISSORES PARA O DIAGNÓSTICO DE FALHAS E DETECÇÃO DE ANOMALIAS

A análise de sinais no diagnóstico de falhas em máquinas industriais já é conhecida na literatura. Este tipo de abordagem, bastante utilizada por empresas na manutenção de equipamentos (HASHEMIAN, 2010), desempenha um papel fundamental em questões financeiras e de segurança (CHINNIAH, 2015). Um diagnóstico precoce e correto da integridade de equipamentos melhora a gestão da manutenção e também a segurança dos sistemas monitorados, *e.g.*, máquinas rotativas industriais (HASHEMIAN, 2010) e turbinas eólicas (JIANG et al., 2017).

Neste contexto, o aprendizado de máquina, seja ele tradicional ou profundo, tornou-se uma ferramenta importante e amplamente utilizada (LIU et al., 2018). Os algoritmos de aprendizado de máquina podem fornecer diagnósticos relativamente rápidos e precisos, se comparados a profissionais humanos. Isto ocorre porque os mesmos não estão sujeitos a problemas como o cansaço e a falta de atenção, possibilitando o monitoramento simultâneo e em tempo real de vários equipamentos na linha de produção. Os algoritmos de aprendizado de máquina são treinados para identificar padrões em dados referentes às diferentes condições de operação das máquinas, como os sinais coletados por sensores. Estes sinais, por sua vez, podem estar representados nos domínios do tempo (CABRERA et al., 2020), da frequência (MONTEIRO; BASTOS-FILHO, 2019a), ou mesmo tempo-frequência (MONTEIRO et al., 2019).

No entanto, para que os modelos de aprendizado de máquina atendam aos objetivos, *e.g.*, diagnóstico de falhas, é necessário que os dados utilizados sejam apropriados (BACK; TRAPPENBERG, 1999). Em outras palavras, os dados utilizados devem conter informações discriminativas o suficiente para caracterizar de forma única cada módulo de falha. Selecionar as informações mais relevantes para treinar estes modelos, seja para a detecção de anomalias ou diagnóstico de falhas, não é algo trivial. O mesmo vale para a forma como tais informações são apresentadas, *e.g.*, o domínio do sinal e o tamanho dos vetores de dados que alimentam os algoritmos. Por outro lado, existem formas de contornar esse problema que podem ser encontradas na literatura.

Para o cenário no qual diferentes sensores fornecem informações acerca da operação de máquinas rotativas, o uso da informação mais apropriada ao diagnóstico de falhas pode ocorrer de 4 formas. A primeira é treinar modelos com os dados de cada sensor, e escolher aquele atrelado ao melhor desempenho. A segunda forma é estabelecer uma medida de qualidade para a informação e selecionar os dados mais promissores baseando-se nesta medida (ERIKSSON; FRISK; KRYSANDER, 2013). A terceira é treinar modelos com os dados de cada sensor, para então formar um comitê com os modelos treinados com dados de sensores diferentes (SHARKEY; CHANDROTH; SHARKEY, 2000). Os resultados dos modelos são combinados de acordo com alguma estratégia predefinida, *e.g.*, votação majoritária, para realizar o diagnóstico de falhas. A

última forma é treinar modelos para realizar o diagnóstico de falhas com a fusão de informações multimodais, *i.e.*, de diferentes sensores (HE et al., 2019).

A seleção do conjunto de dados mais apropriado para uma determinada tarefa permanece importante, independente da existência de comitês ou classificadores multimodais capazes de lidar com dados de diferentes modalidades. Ao identificar os dados que agregam mais informações úteis ao sistema, os demais dados podem ser descartados. Desta forma, reduz-se a quantidade de informação utilizada, o tempo de treinamento dos modelos, bem como os tamanhos e complexidades, dentre outros aspectos.

Neste contexto, propõe-se o uso de medidas como a divergência de Kullback-Leibler (KL) (KULLBACK, 1997) e o teste de Kolmogorov-Smirnov (KS) (NEUHAUSER, 2011) para selecionar os dados dos sensores mais promissores ao diagnóstico de falhas. A divergência de KL é uma medida de dissimilaridade. Ela mede quão diferentes são duas distribuições de probabilidades entre si. Esta diferença, também conhecida como divergência, é calculada em termos de uma medida de informação, *e.g.* entropia. Para duas distribuições P_1 e P_2 de uma variável contínua x, a divergência de Kullback-Leibler é definida pela equação (3.1).

$$D_{KL}(P_1(x)||P_2(x)) = \int_{-\infty}^{\infty} p_1(x) \log\left(\frac{p_1(x)}{p_2(x)}\right) dx \tag{3.1}$$

na qual p_1 e p_2 denotam as densidades de probabilidades de P_1 e P_2, respectivamente. Por outro lado, o teste de KS é um teste estatístico não paramétrico que também compara duas distribuições de probabilidades. De acordo com a hipótese nula, os dados comparados são oriundos de distribuições de probabilidade estatisticamente idênticas. O teste estatístico considera o valor máximo da diferença absoluta entre duas funções de distribuições empíricas, conforme apresentado na equação (3.2).

$$D_{n_1,n_2} = \max_t |F_{1,n_1}(t) - F_{2,n_2}(t)| \tag{3.2}$$

na qual F_1 e F_2 são duas funções de distribuições empíricas. n_1 e n_1 são os tamanhos das duas distribuições, respectivamente. Para conjuntos grandes, a hipótese nula é rejeitada para um dado nível de significância α, *e.g.* 5%, se as Equações (3.3) e (3.4) forem verdadeiras.

$$D_{n_1,n_2} > c(\alpha)\sqrt{\frac{n_1 + n_2}{n_1 n_2}} \tag{3.3}$$

$$c\left(\alpha\right) = \sqrt{-\frac{1}{2}\ln\alpha} \tag{3.4}$$

Desta forma, considerando que as informações de entrada do problema, *e.g.*, espectros de frequência, são distribuições de probabilidade, utiliza-se essas medidas na identificação de dados que apresentam potencial para treinar modelos mais precisos.

A contribuição está na forma como se usa a divergência de KL e o teste de KS para este propósito. Para exemplificar o processo, tem-se a seguinte situação: dados dois conjuntos de dados, sendo cada conjunto relacionado a um sensor diferente, calcula-se a razão entre as médias das dissimilaridades interclasse e intraclasse de cada um. Então, seleciona-se o conjunto de dados mais promissor ao comparar-se os valores dessas razões. Vale ressaltar que esta análise comparativa não se limita a pares de conjuntos de dados, podendo envolver uma quantidade maior de conjuntos, caso necessário. Ao longo do estudo, notou-se que os conjuntos de dados que apresentavam as maiores razões estava relacionadas aos modelos mais precisos.

Esta técnica também foi utilizada na análise da informação contida em diferentes formatos dos dados de entrada. Em outras palavras, foi avaliado o potencial de espectros de frequência e espectrogramas com diferentes dimensões para treinar modelos precisos, baseando-se nas medidas descritas.

3.1 METODOLOGIA

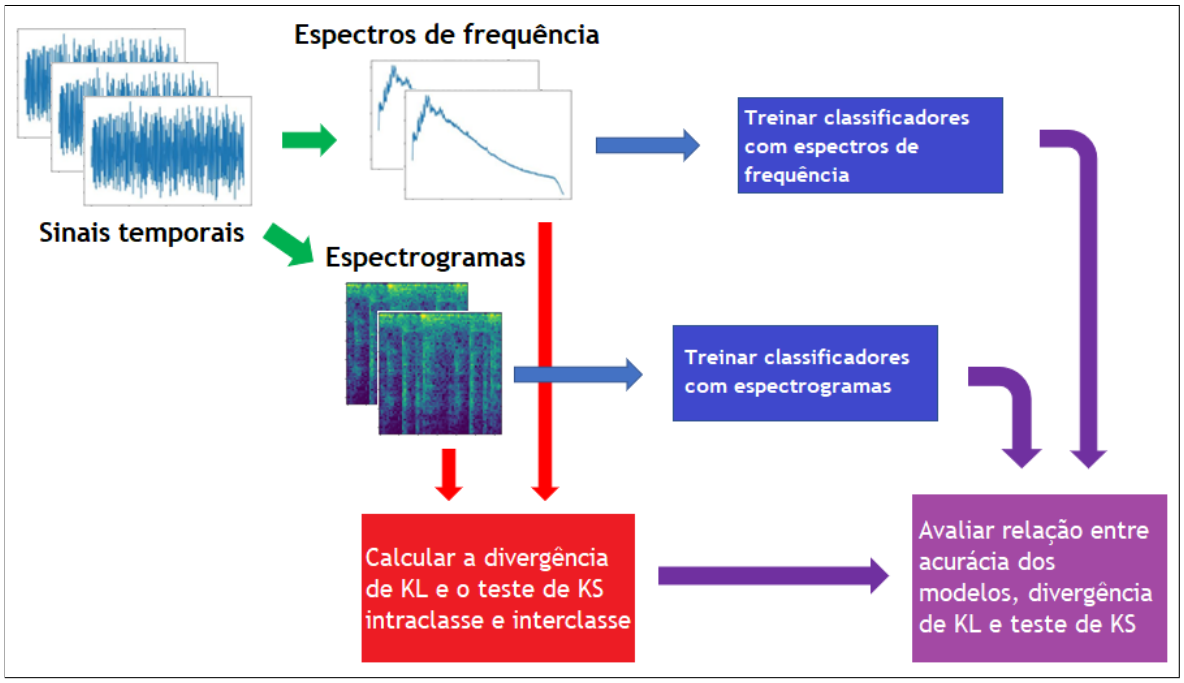
Para explicar como o estudo foi desenvolvido, toma-se como referência o cenário no qual foram analisados diferentes formatos de dados de entrada. Este processo é ilustrado no diagrama da Figura 1. O primeiro passo foi obter espectros de frequência e espectrogramas de diferentes dimensões a partir dos sinais no tempo, por meio da transformada rápida de Fourier (FTT, do inglês *Fast Fourier Transform*) (RAO; KIM; HWANG, 2011) e da Transformada de Fourier de Tempo Curto (STFT, do inglês *Short Time Fourier Transform*) (RAO; KIM; HWANG, 2011), respectivamente. Os espectros de frequência e espectrogramas de diferentes tamanhos, bem como seus respectivos rótulos, foram utilizados para treinar os modelos de classificação, *i.e.*, diagnóstico de falhas. Além disso, o teste de KS e a divergência de KL foram aplicados aos diferentes tipos e dimensões de dados de entrada, de modo a obter medidas de dissimilaridades entre dados pertencentes à mesma classe e a classes diferentes. Ao fim, tais medidas foram analisadas em conjunto com a acurácia dos classificadores treinados.

A análise dos dados de diferentes sensores é feita de maneira similar. A diferença é que esta análise compreende apenas espectros de frequência calculados a partir dos sinais temporais obtidos por diferentes sensores, *e.g.*, acelerômetros, sensores de corrente e microfones. As demais etapas são repetidas.

3.1.1 A razão entre as médias das dissimilaridades interclasse e intraclasse.

Para cada conjunto de dados, *e.g.*, espectros de frequência obtidos a partir dos sinais de um dos acelerômetros, calcula-se a razão (R) entre os valores médios de divergência de KL

Figura 1 – Metodologia da análise dos dados de entrada de diferentes formatos.



Fonte: O autor (2020).

interclasse e intraclasse. Esta razão é utilizada para indicar os dados que resultam nos modelos mais precisos. Esta análise é comparativa, isto é, compara-se as razões obtidas de cada conjunto de dados e escolhe-se aquele supostamente mais adequado para realizar o diagnóstico de falhas. Esta comparação pode ser feita considerando 2 ou mais conjuntos de dados.

Se a razão tende a um, tem-se que as médias das divergências interclasse e intraclasse são próximas. Desta forma, espera-se que exista uma maior dificuldade para se treinar modelos precisos, pois isto significa que a dissimilaridade entre dados de mesma classe e de classes diferentes é semelhante. Em contrapartida, quão maior for o valor da razão, tem-se que maior é a divergência média entre dados de classes diferentes em relação a de dados pertencentes à mesma classe, o que provavelmente simplificaria o processo de aprendizado dos modelos. Esta mesma lógica é utilizada nas análises que envolvem o teste de KS.

Os valores de divergência de Kullback-Leibler e do teste de Kolmogorov-Smirnov são calculados entre pares de espectros de frequência ou entre pares de espectrogramas. Estes valores, considerando todos os pares de espectros de frequência ou espectrogramas do conjunto de dados, são utilizados na obtenção dos valores médios intraclasse e interclasse da medida de dissimilaridade em questão, e estes últimos são utilizados no cálculo da razão. No caso dos espectrogramas, é preciso reorganizá-los numa disposição linear antes do cálculo da divergência de KL e do teste de KS.

3.1.2 Base de dados

Neste capítulo, foram utilizadas 2 bases de dados na realização dos experimentos. A primeira delas trata de falhas em mancais de compressores. A segunda também trata de falhas em compressores, mas combina as falhas de mancais e válvulas.

3.1.2.1 Falhas de mancais em compressores

Este conjunto de dados faz referência à diagnose de falhas de mancais em compressores. Ele apresenta quatro classes relacionadas a diferentes condições de falhas, indicadas na Tabela 1. A classe P1 representa a condição normal de operação do compressor, enquanto às demais classes estão relacionadas a três modalidades de falhas. Os dados consistem de sinais temporais coletados por diferentes tipos de sensores.

Tabela 1 – As quatro classes referentes ao conjunto de dados de falhas de mancais em compressores.

Código da Falha	**Tipo da Falha**
P1	Sem falha
P2	Rachadura no canal interno
P3	Rachadura no elemento de rolagem
P4	Rachadura no canal externo

Fonte: O autor (2020).

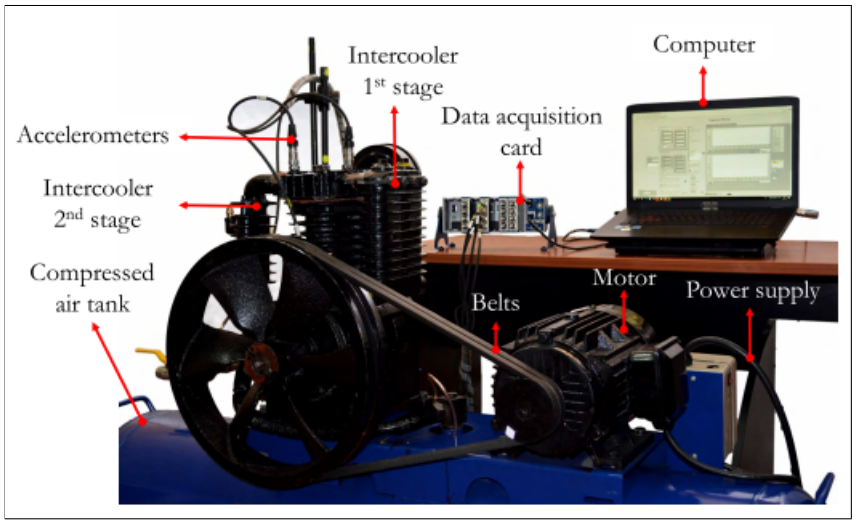
Os sinais foram coletados de acordo com o arranjo experimental ilustrado na Figura 2, no laboratório do grupo de pesquisa GIDTEC, da Universidade Politécnica Salesiana (Cuenca, Equador). O seu principal componente é um compressor alternativo de dois estágios EBG250, 5 HP. Um motor trifásico proporciona potência ao compressor por meio de um sistema de transmissão de correia. A frequência de rotação do motor é 57,7 Hz, resultando numa frequência de rotação de 12,8 Hz para o eixo do compressor.

A medição do sinal de vibração foi realizada com um acelerômetro *IMI SENSORS* 603C01, de sensibilidade 100mV/g, e posicionado verticalmente sobre a câmara de compressão. Os sinais analógicos oriundos desta medição foram transferidos à placa de aquisição de dados *National Instrument* NI9234, e convertidos em sinais digitais a uma taxa de amostragem de 50.000 amostras/s. A placa foi montada num chassis NI9188, também da *National Instrument*, a fim de se transmitir os dados para armazenamento num computador.

3.1.2.2 Falhas de mancais e válvulas em compressores

Este conjunto de dados faz referência à diagnose de falhas combinadas de mancais e válvulas em compressores. Ele apresenta treze classes relacionadas a diferentes condições de falhas, indicadas na Tabela 2. A classe P1 representa a condição normal de operação do compressor, enquanto às demais classes estão relacionadas a doze modalidades de falhas. Os

Figura 2 – Arranjo experimental para a coleta dos sinais de vibração.



Fonte: (CABRERA et al., 2019).

dados também consistem de sinais temporais coletados por diferentes tipos de sensores, de acordo com um arranjo experimental semelhante ao apresentado na Figura 2.

3.1.3 Organizando os dados de entrada a partir dos sinais temporais

A partir deste ponto, as subseções de Metodologia e Resultados serão divididas em duas partes: **Cenário 1** e **Cenário 2**. Os estudos relacionados aos dois cenários foram feitos paralelamente e referem-se ao uso da razão R, descrita em 3.1.1. Enquanto o Cenário 1 trata da comparação entre sinais obtidos por sensores diferentes, o Cenário 2 trata da comparação entre diferentes formatos de dados de entrada, porém obtidos a partir de sinais coletados por um mesmo sensor.

3.1.3.1 Cenário 1: Comparando dados de diferentes sensores

Neste cenário, os dados utilizados são espectros de frequência. Eles foram obtidos pela aplicação da FFT em medições realizadas por nove sensores, que se encontram listados na Tabela 3. Os sinais obtidos pelos sensores, no domínio do tempo, foram divididos em seções menores, cada uma com duração de 0,1 segundo. Isto resultou em 1.500 sinais de vibração para cada classe, considerando ambas as bases de dados. Consequentemente, tem-se 1.500 espectros de frequência para cada classe.

Os espectros estão representados por vetores contendo 1.024 elementos. Este tamanho de vetor for escolhido após uma análise preliminar, na qual foram comparados os desempenhos de classificadores treinados com vetores de diferentes tamanhos, *e.g.*, 128, 256, 512, 1.024, 2.048, etc. Os vetores com 1.024 elementos apresentaram a melhor combinação de acurácia (valor médio próximo a 95%) e tempo de treinamento por modelo (inferior a 5 minutos).

As duas bases de dados apresentadas em 3.1.2 foram utilizadas no estudo. Cada uma

Tabela 2 – As treze classes referentes ao conjunto de dados com falhas combinadas de mancais e válvulas em compressores.

Código da Falha	Tipo da Falha
P1	Sem falha
P2	Rachadura no canal interno/ Desgaste da sede da válvula
P3	Rachadura no canal interno/ Corrosão da placa da válvula
P4	Rachadura no canal interno/ Fratura da placa da válvula
P5	Rachadura no canal interno/ Quebra da mola
P6	Rachadura no elemento de rolagem/ Desgaste da sede da válvula
P7	Rachadura no elemento de rolagem/ Corrosão da placa da válvula
P8	Rachadura no elemento de rolagem/ Fratura da placa da válvula
P9	Rachadura no elemento de rolagem/ Quebra da mola
P10	Rachadura no canal externo/ Desgaste da sede da válvula
P11	Rachadura no canal externo/ Corrosão da placa da válvula
P12	Rachadura no canal externo/ Fratura da placa da válvula
P13	Rachadura no canal externo/ Quebra da mola

Fonte: O autor (2020).

Tabela 3 – Lista de sensores.

Código	Sensor
MC1	Microfone
MC2	Microfone
CVC1	Sensor de corrente
CVC2	Sensor de corrente
CVC3	Sensor de corrente
A1	Acelerômetro
A2	Acelerômetro
A3	Acelerômetro
A4	Acelerômetro

Fonte: O autor (2020).

contém dados coletados por nove sensores diferentes. Considerando a primeira base, tem-se quatro classes, sendo uma relacionada à operação normal do compressor e as demais relacionadas às falhas. Além disso, tem-se 1.500 espectros de frequência por classe para cada sensor. Desta

forma, considerando-se quatro classes e nove sensores, o número de espectros de frequência resultantes na primeira base é 54.000.

Com segunda base de dados ocorre algo semelhante, estando a diferença no número de classes, *i.e.*, treze. Desta forma, tem-se um número de espectros de frequência resultante igual 175.000, *i.e.*, tem-se 1.500 espectros de frequência por classe para cada sensor, treze classes e nove sensores.

3.1.3.2 Cenário 2: Comparando diferentes formatos de dados

Os sinais no domínio do tempo foram divididos em seções menores, cada uma com duração de 0,1 segundo. Isto resultou em 1.500 sinais de vibração para cada classe. Neste estudo, foram utilizadas duas formas de representação dos sinais: espectros de frequência e espectrogramas. O primeiro trata de uma forma de representação dos sinais por meio das frequências que os compõem, enquanto o segundo permite visualizar como estes componentes de frequência de cada sinal variam no tempo.

Os espectros de frequência foram obtidos por meio da FFT, como mencionado em 3.1. Nesta etapa, avaliou-se a influência na acurácia do modelo da representação dos sinais em diferentes resoluções de frequência. Desta forma, foram testados vetores de entrada para o sistema de classificação, *i.e.*, espectros de frequência, com 8, 16, 32, 64, 128, 256, 512 e 1.024 elementos.

Por outro lado, os espectrogramas foram obtidos pela STFT, conforme também mencionado em 3.1. Eles carregam mais informações, se comparados aos espectros de frequência, uma vez que mostram como as componentes de frequência de um sinal variam ao longo do tempo. No entanto, este tipo de representação aumenta o custo computacional dos processos de treinamento e classificação, considerando-se os espectros de frequência e os espectrogramas com a mesma resolução em frequência. Neste cenário, 8 resoluções em frequência diferentes foram testadas, considerando o sinal temporal de tamanho fixo. Tais resoluções incluíram representações em frequência com: 8, 16, 24, 32, 40, 48, 56 e 64 elementos. Isto resultou em espectrogramas de formato: 8 x 357, 16 x 178, 24 x 118, 32 x 89, 40 x 71, 48 x 59, 56 x 50 e 64 x 44. A sobreposição das janelas foi de 12,5%.

3.1.4 Os modelos de classificação

Nesta subseção, discute-se sobre os modelos de classificação utilizados no estudo.

3.1.4.1 Cenário 1: Comparando dados de diferentes sensores

Os classificadores utilizados para realizar o diagnóstico de falhas foram dois tipos de redes neurais: MLPs e CNNs. Com relação à MLP, foram utilizadas arquiteturas com apenas 1

camada oculta, mas com diferentes quantidades de neurônios nesta camada, *e.g.*, 512, 1.024, ou 2.048 neurônios.

Quanto à CNN, foram utilizadas arquiteturas com camadas convolucionais, de *max-pooling*, de linearização e densas. Variou-se o número de camadas convolucionais e de *max-pooling*, bem como a quantidade de neurônios nas camadas densas, a fim de se melhorar a acurácia do processo de classificação. A Tabela 4 dá mais detalhes acerca da arquitetura das CNNs.

As funções de ativação utilizadas em ambos os tipos de redes foram a *softmax*, para os neurônios de saída, e a ReLU (do inglês *Rectified Linear Unit*), para os demais neurônios. A não linearidade da função ReLU permite que redes neurais aprendam padrões não lineares. Além disso, ela aprimora o processo de aprendizado das redes, pois seu gradiente não apresenta problemas de saturação da mesma forma que ocorre em funções como a sigmóide e a tangente hiperbólica. A função *softmax* transforma as saídas das redes em valores de probabilidade, proporcionais às exponenciais das próprias saídas da rede (GOODFELLOW; BENGIO; COURVILLE, 2016).

É válido ressaltar que esta parte do estudo busca apenas verificar se o comportamento esperado, *i.e.*, a relação descrita em 3.1.1 entre razão e acurácia, ocorre com modelos de complexidades diferentes.

Tabela 4 – Camadas das CNNs

Camadas	**Quantidade**
Camada convolucional com filtros 3 x 1 Camada de max-pooling com filtros 2 x 1	Modelos com 1, 2, 3, e 4 pares
Camada de linearização	1
Camada densa	Modelos de 1 camada com 32, 64, 128, e 256 neurônios
Camada de saída	4 neurônios na base 1, e 13 neurônios na base 2

Fonte: O autor (2020).

3.1.4.2 Cenário 2: Comparando diferentes formatos de dados

Redes neurais convolucionais foram utilizadas para realizar o processo de classificação, *e.g.*, o diagnóstico de falhas. Assim como no cenário anterior, foram adotadas arquiteturas contendo camadas convolucionais, de *max-pooling*, de linearização dos dados e densas. A Tabela 5 lista essas camadas com maior detalhe. Dado que o objetivo deste estudo é analisar a relação entre diferentes tipos de entrada, medidas de dissimilaridade e o desempenho dos modelos de classificação, a escolha dos parâmetros do modelo não constituem um aspecto crítico para os experimentos.

Tabela 5 – Configuração das CNNs.

Camadas
Camada convolucional com 9 filtros (3 x 3 para espectrogramas, 3 x 1 para espectros de frequência)
Camada de *max-pooling* (2 x 2 para espectrogramas, 2 x 1 espectros de frequência)
Camada de linearização
Camada densa (32 neurônios)
Camada de saída (4 neurônios na base 1 e 13 neurônios na base)

Fonte: O autor (2020).

3.1.5 Treinando os modelos de diagnóstico de falhas

Nesta subseção, discute-se sobre os processos de treinamento dos modelos de classificação utilizados no estudo.

3.1.5.1 Cenário 1: Comparando dados de diferentes sensores

Os espectros de frequência foram divididos em grupos de treinamento e de teste. Enquanto o grupo de treinamento continha 80% dos espectrogramas, o grupo de teste continha 20%. Para cada combinação de configuração de modelo e conjunto de dados, 15 classificadores foram treinados, todos por 50 épocas.

O computador utilizado para realizar o treinamento dos modelos apresentou a seguinte configuração: OS Windows 10 Home, 64 bits, Memória (RAM) 15.9 GB, Processador Intel® Core™ i7-6500 CPU @ 2.50 GHz x 2, AMD Radeon™ T5 M330 (sem suporte CUDA). Todos os códigos foram escritos em Python (VANROSSUM; DRAKE, 2010) 3.7, no Jetbrains PyCharm (ISLAM, 2015) Community Edition 2019.2.

3.1.5.2 Cenário 2: Comparando diferentes formatos de dados

Conforme explicado em 3.1.2, os dados foram divididos em quatro classes, considerando a base de dados com falhas em mancais, e treze classes, considerando a base de dados com falhas combinadas em mancais e válvulas. Cada classe continha 1.500 sinais. Os sinais referentes a cada classe foram divididos em 10 subconjuntos, combinados de forma aleatória para formar os conjuntos de treino e de teste. 80% dos sinais, *i.e.*, 1.200 ou 8 subconjuntos, compunham o conjunto de treino de cada classe, enquanto os 20% restantes, *i.e.*, 300 ou 2 subconjuntos, compunham os conjuntos de teste. Cada modelo treinado contou com uma combinação diferente de subconjuntos para treino e teste. Neste cenário, apenas os sinais referentes a um dos acelerômetros foram utilizados, considerando ambos os cenários de falhas.

15 modelos foram treinados para cada combinação de formato de entrada e conjunto de

dados. O número de épocas para o treinamento de cada modelo também foi 50. O computador utilizado para realizar o treinamento dos modelos apresentou a seguinte configuração: OS Windows 10 Home, 64 bits, Memória (RAM) 15.9 GB, Processador Intel® Core™ i7-6500 CPU @ 2.50 GHz x 2, AMD Radeon™ T5 M330 (sem suporte CUDA). Todos os códigos foram escritos em Python (VANROSSUM; DRAKE, 2010) 3.7, no Jetbrains PyCharm (ISLAM, 2015) Community Edition 2019.2.

3.2 RESULTADOS E DISCUSSÕES

Nesta seção, os resultados do estudo desenvolvido ao longo deste capítulo são apresentados e discutidos.

3.2.1 Cenário 1: Comparando dados de diferentes sensores

Nesta subseção, são avaliados os desempenhos dos modelos de classificação treinados com sinais coletados por diferentes sensores.

3.2.1.1 Resultados para o Percéptron de Múltiplas Camadas

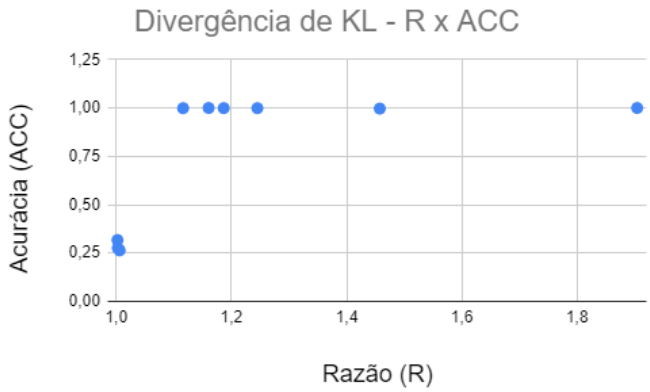
Os primeiros experimentos foram realizados com os dados de falhas em mancais, cujas classes estão indicadas na Tabela 1. A Figura 3 mostra a relação da acurácia média dos modelos treinados com a razão (R) entre a média das divergências de KL interclasse e intraclasse, para cada sensor. A Figura 4 também mostra essa relação entre acurácia e R, porém a razão foi obtida por meio do teste de KS. Estes resultados pertencem à configuração de MLP com melhor desempenho em termos de acurácia, *i.e.*, a configuração com 1024 neurônios na camada oculta. Cada ponto nas Figuras 3 e 4 se refere ao resultado médio obtido por 15 modelos treinados com os dados relacionados a um único sensor. Desta forma, tem-se nove pontos no gráfico, dado que tem-se dados obtidos por nove sensores diferentes. Os dados de cada sensor estão associados a um único valor da razão R.

Ressalta-se que não há rótulos indicando os sensores representados por cada ponto, pois o objetivo desta análise é verificar se de fato existe uma relação entre R e a acurácia dos modelos treinados, e não qual sensor apresentou o melhor ou pior resultado.

Pode-se observar nas Figuras 3 e 4 que os sensores atrelados aos maiores valores de R foram aqueles que apresentaram o melhores desempenhos em termos de acurácias. Para o cenário onde R foi obtido por meio da divergência de KL, vê-se que os sensores atrelados a valores da razão acima de 1,1 apresentaram acurácia média próxima a 1, *i.e.*, 100%. Algo semelhante ocorre para R obtido por meio do teste de KS. Em contrapartida, os valores de R mais próximos a 1 resultaram em modelos de baixa acurácia.

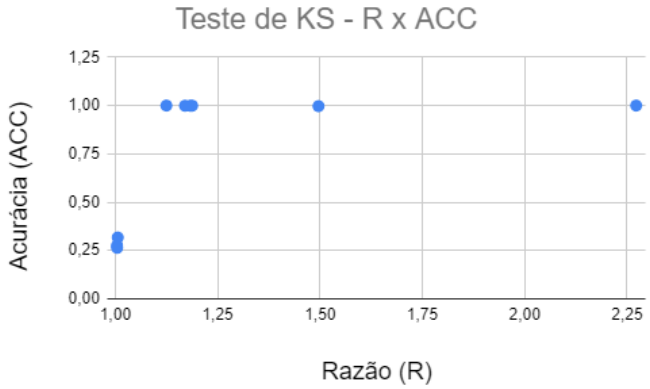
A Figuras 5 e 6 mostram os resultados obtidos por MLPs para as falhas combinadas em mancais e válvulas. Na Figura 5, os valores da razão R foram obtidos com a divergência

Figura 3 – Razão x Acurácia para o cenário com falhas em mancais, realizando o diagnóstico de falhas com MLPs. Os valores da razão R foram obtidos com a divergência de Kullback-Leibler.



Fonte: O autor (2020).

Figura 4 – Razão x Acurácia para o cenário com falhas em mancais, realizando o diagnóstico de falhas com MLPs. Os valores da razão R foram obtidos com o teste de Kolmogorov-Smirnov.



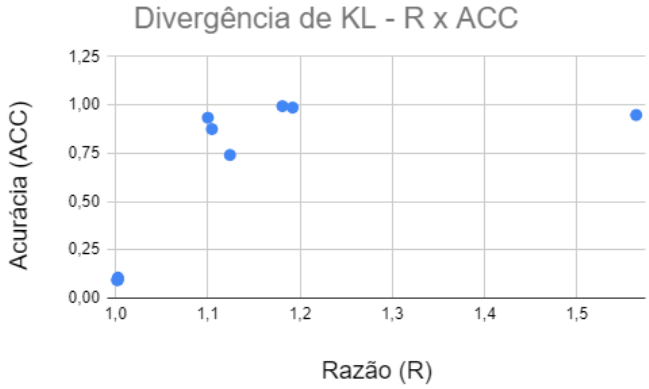
Fonte: O autor (2020).

de KL. Por outro lado, na Figura 6, os valores de R foram obtidos pelo teste de KS. As classes desses dados estão listadas na Tabela 2. Assim como na análise anterior, cada ponto se refere ao resultado médio obtido por 15 modelos treinados com os dados relacionados a um único sensor.

Nestas figuras, observa-se uma tendência similar à observada nas Figuras 3 e 4. Em outras palavras, a acurácia de modelos treinados com dados de sensores associados a maiores valores de R tendeu a ser maior. Da mesma forma, valores de R mais baixos tenderam a estar relacionados a modelos de pior desempenho.

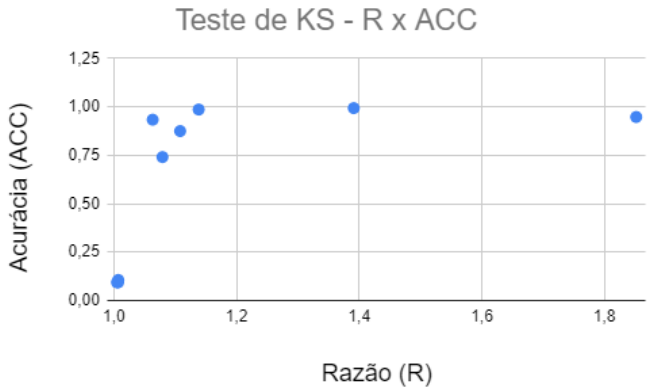
A principal diferença deste cenário em relação ao com falhas em mancais está na transição entre os valores de acurácia mais baixos e os mais elevados. No cenário com falhas em mancais apenas, a transição foi abrupta. Em outras palavras, os valores de acurácia passaram repentinamente de aproximadamente 30% para 100% quando os valores de R atingiram aproximadamente 1,1. Entretanto, conforme visto nas Figuras 5 e 6, a transição ocorreu de uma forma

Figura 5 – Razão x Acurácia para o cenário com falhas combinadas em mancais e válvulas, realizando o diagnóstico de falhas com MLPs. Os valores da razão R foram obtidos com a divergência de Kullback-Leibler.



Fonte: O autor (2020).

Figura 6 – Razão x Acurácia para o cenário com falhas combinadas em mancais e válvulas, realizando o diagnóstico de falhas com MLPs. Os valores da razão R foram obtidos com o teste de Kolmogorov-Smirnov.



Fonte: O autor (2020).

mais suave. Os modelos associados a dados com valores de R próximos a 1,1 não apresentaram o valor máximo de acurácia. Possivelmente este comportamento se deve às diferentes quantidades de classes nos conjuntos de dados. No cenário com falhas combinadas, a quantidade de classes é treze, contra quatro no cenário com apenas falhas em mancais. Este fator pode estar dificultando o processo de aprendizado dos classificadores, resultando numa pior separabilidade das características, requerendo uma maior quantidade de dados para treino ou mesmo maiores valores da razão entre as dissimilaridades interclasse e intraclasse.

Um aspecto digno de nota é que não parece ser viável estabelecer uma relação global entre a razão R e a acurácia média. Em outras palavras, observou-se nos experimentos com dados diferentes que, as relações entre R e a acurácia não foram iguais. Isto significa que não seria possível estabelecer um limiar de R como valor mínimo adequado para a escolha de um determinado sensor. Este limiar seria algo dependente do problema, *e.g.*, dados, em questão.

As Tabelas 6 e 7 listam os valores dos resultados apresentados nas Figuras 3, 4, 5 e 6. Os valores entre parênteses representam os desvios padrões.

Tabela 6 – Valores das razões obtidas com a divergência de KL e das acurácias médias das MLPs com 1024 neurônios na camada oculta, considerando as bases com falhas em mancais (BF) e com falhas combinadas (MF).

Sensor	**Razão - BF**	**Acc - BF**	**Razão - MF**	**Acc - MF**
MC1	1,1604	1,0000 (0,00)	1,1002	0,9335 (0,03)
MC2	1,1866	0,9993 (0,00)	1,1810	0,9933 (0,01)
CVC1	1,0021	0,3172 (0,32)	1,0027	0,0938 (0,09)
CVC2	1,0027	0,2757 (0,31)	1,0028	0,1060 (0,15)
CVC3	1,0059	0,2637 (0,32)	1,0017	0,0935 (0,11)
A1	1,4575	0,9968 (0,00)	1,1241	0,7405 (0,12)
A2	1,2450	0,9991 (0,00)	1,1921	0,9860 (0,01)
A3	1,9043	1,0000 (0,00)	1,5644	0,9474 (0,02)
A4	1,1161	0,9990 (0,00)	1,1046	0,8747 (0,06)

Fonte: O autor (2020).

Tabela 7 – Valores das razões obtidas com teste de KS e das acurácias médias das MLPs com 1024 neurônios na camada oculta, considerando as bases com falhas em mancais (BF) e com falhas combinadas (MF).

Sensor	**Razão - BF**	**Acc - BF**	**Razão - MF**	**Acc - MF**
MC1	1,1249	1,000 (0,00)	1,632	0,9335 (0,03)
MC2	1,1879	0,9993 (0,00)	1,3912	0,9933 (0,01)
CVC1	1,0063	0,3172 (0,32)	1,0069	0,0938 (0,09)
CVC2	1,0043	0,2757 (0,31)	1,0071	0,1060 (0,15)
CVC3	1,0045	0,2637 (0,32)	1,0050	0,0935 (0,11)
A1	1,4964	0,9968 (0,00)	1,0792	0,7405 (0,12)
A2	1,1704	0,9991 (0,00)	1,1383	0,9860 (0,01)
A3	2,2723	1,0000 (0,00)	1,8519	0,9474 (0,02)
A4	1,1830	0,9990 (0,00)	1,1046	0,8747 (0,06)

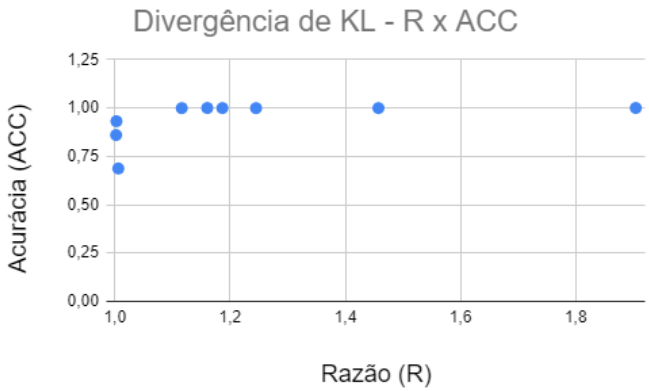
Fonte: O autor (2020).

3.2.1.2 Resultados para Redes Neurais Convolucionais

As redes neurais convolucionais também foram utilizadas nos experimentos deste capítulo. CNNs são algoritmos de classificação capazes de aprender padrões mais complexos nas distribuições de dados, quando comparadas às redes do tipo MLP. Os resultados apresentados nas Figuras 7, 8, 9 e 10 foram obtidos com a configuração de CNN que apresentou melhor desempenho dentre as configurações testadas, *i.e.*, 3 pares de camadas convolucionais e de *max-pooling*, além de 256 neurônios na camada densa. A Figura 7 mostra a relação dos valores de R obtidos por meio da divergência de Kullback-Leibler com as acurácias médias dos classificadores. A Figura 8 também traz essa relação, porém os valores de R foram obtidos por meio do teste de Kolmogorov-Smirnov. De modo semelhante aos experimentos com MLPs, cada ponto nas

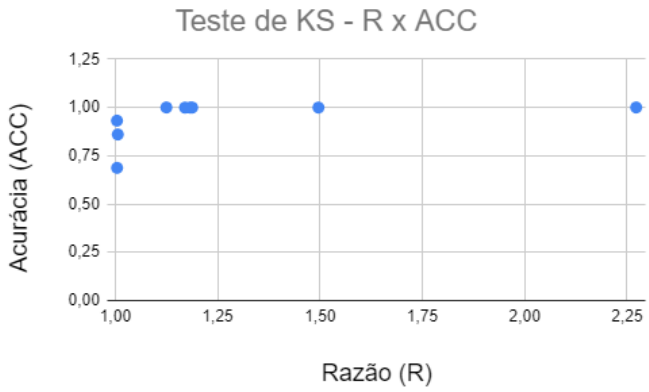
Figuras 7 e 8 traz o resultado médio de acurácia obtido por 15 modelos treinados com dados relacionados a um único sensor.

Figura 7 – Razão x Acurácia para o cenário com falhas em mancais, realizando o diagnóstico de falhas com CNNs. Os valores da razão R foram obtidos com a divergência de Kullback-Leibler.



Fonte: O autor (2020).

Figura 8 – Razão x Acurácia para o cenário com falhas em mancais, realizando o diagnóstico de falhas com CNNs. Os valores da razão R foram obtidos com o teste de Kolmogorov-Smirnov.



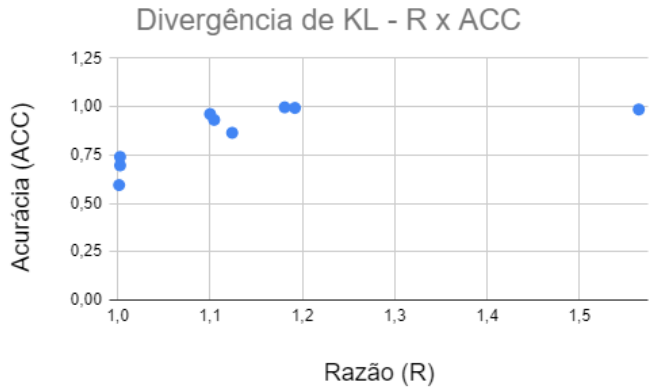
Fonte: O autor (2020).

Observa-se nas Figuras 7 e 8 um comportamento semelhante ao visto nas Figuras 3 e 4, isto é, os dados dos sensores relacionados aos menores valores da razão R foram aqueles que resultaram nos classificadores de menor acurácia. Em contrapartida, os dados atrelados a razões superiores a 1,1 apresentaram resultados próximos a 1. A maior diferença está nos resultados obtidos com os sensores associados a razões próximas a 1, que foram superiores àqueles obtidos com MLPs. Isto era esperado, pois, conforme mencionado anteriormente, as redes neurais convolucionais são capazes de aprender padrões mais complexos que as MLPs.

Esta mesma observação é válida para os resultados apresentados nas Figuras 9 e 10, se comparados àqueles vistos nas Figuras 5 e 6. Isto inclui a transição mais suave entre os valores

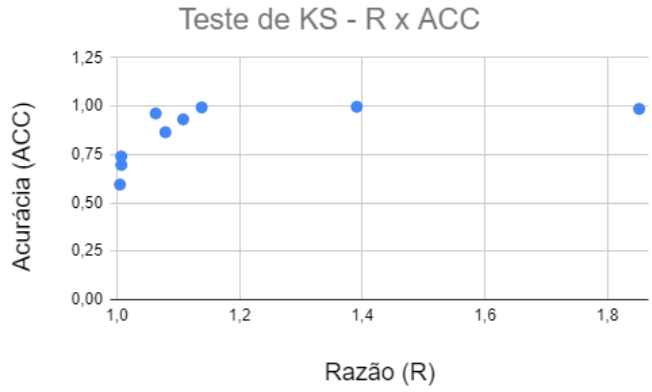
médios de acurácia associados a razões próximas a 1 e àqueles associados às razões superiores a 1,5. Os resultados apresentados nas Figuras 7, 8, 9 e 10 estão listados nas Tabelas 8 e 9. Os valores entre parênteses representam os desvios padrões.

Figura 9 – Razão x Acurácia para o cenário com falhas combinadas em mancais e válvulas, realizando o diagnóstico de falhas com CNNs. Os valores da razão R foram obtidos com a divergência de Kullback-Leibler.



Fonte: O autor (2020).

Figura 10 – Razão x Acurácia para o cenário com falhas combinadas em mancais e válvulas, realizando o diagnóstico de falhas com CNNs. Os valores da razão R foram obtidos com o teste de Kolmogorov-Smirnov.



Fonte: O autor (2020).

Os resultados obtidos pela MLP e pela CNN sugerem que as razões entre as medidas de dissimilaridade interclasse e intraclasse nos dados de um determinado sensor estão relacionadas a acurácia dos modelos treinados com esses mesmos dados. Considerando a MLP e a CNN, a tendência foi que os valores de razão próximos a 1 estavam associados aos classificadores de pior desempenho, que ia melhorando à medida que o valor dessa razão aumentava. Isto pode estar intimamente ligado à capacidade dos modelos de identificar padrões que discriminem as classes de forma eficiente, dado que quanto maior é o valor da razão, maior a dissimilaridade entre dados de classes diferentes em relação a dados de uma mesma classe. Por outro lado, observa-se que essas relações não ocorrem exatamente da mesma forma para os classificadores

Tabela 8 – Valores das razões obtidas com a divergência de KL e das acurácias médias das CNNs com 3 pares de camadas convolucionais e de *max-pooling*, com 256 neurônios na camada densa, considerando as bases com falhas em mancais (BF) e com falhas combinadas (MF).

Sensor	Razão - BF	Acc - BF	Razão - MF	Acc - MF
MC1	1,1604	1,0000 (0,00)	1,1002	0,9618 (0,02)
MC2	1,1866	0,9995 (0,00)	1,1810	0,9963 (0,01)
CVC1	1,0021	0,8601 (0,05)	1,0027	0,7397 (0,12)
CVC2	1,0027	0,9315 (0,03)	1,0028	0,6960 (0,15)
CVC3	1,0059	0,6870 (0,14)	1,0017	0,5944 (0,20)
A1	1,4575	0,9993 (0,00)	1,1241	0,8646 (0,06)
A2	1,2450	0,9992 (0,00)	1,1921	0,9933 (0,00)
A3	1,9043	1,0000 (0,00)	1,5644	0,9553 (0,02)
A4	1,1161	0,9993 (0,00)	1,1046	0,9314 (0,04)

Fonte: O autor (2020).

Tabela 9 – Valores das razões obtidas com o teste de KS e das acurácias médias das CNNs com 3 pares de camadas convolucionais e de *max-pooling*, com 256 neurônios na camada densa, considerando as bases com falhas em mancais (BF) e com falhas combinadas (MF).

Sensor	Razão - BF	Acc - BF	Razão - MF	Acc - MF
MC1	1,1249	1,000 (0,00)	1,0632	0,9618 (0,02)
MC2	1,1879	0,9995 (0,00)	1,3912	0,9963 (0,01)
CVC1	1,0063	0,8601 (0,05)	1,0069	0,7397 (0,12)
CVC2	1,0043	0,9315 (0,03)	1,0071	0,6960 (0,15)
CVC3	1,0045	0,6870 (0,14)	1,0050	0,5944 (0,20)
A1	1,4964	0,9993 (0,00)	1,0792	0,8646 (0,06)
A2	1,1704	0,9992 (0,00)	1,1383	0,9933 (0,00)
A3	1,2723	1,000 (0,00)	1,8519	0,9553 (0,02)
A4	1,1830	0,9993 (0,00)	1,1080	0,9314 (0,04)

Fonte: O autor (2020).

de complexidades diferentes, *e.g.*, as acurácias médias das CNNs treinadas com dados coletados pelos sensores CVC1, CVC2 e CVC3 foram próximas ou superiores 0,60, considerando os dois cenários de falhas. Considerando os resultados das MLPs para os mesmos dados, os valores médios de acurácia foram próximos ou inferiores a 0,30.

3.2.2 Cenário 2: Comparando diferentes formatos de dados

Nesta subseção, são avaliados os desempenhos dos modelos de classificação treinados com dados organizados em diferentes formatos.

3.2.2.1 Espectros de frequência

A acurácia dos modelos treinados para classificar falhas em mancais foi o primeiro ponto analisado. A Tabela 10 lista a acurácia média dos classificadores treinados com espectros

de frequência. Os resultados estão organizados de acordo com as dimensões dos vetores de espectros, e referem-se apenas às amostras de teste. Cada valor médio de acurácia apresentado foi obtido a partir de 15 modelos. Os valores entre parênteses representam os desvios padrões.

Tabela 10 – Acurácia média de modelos treinados com espectros de frequência de diferentes comprimentos para diagnosticar falhas em mancais.

Comprimento	**Acurácia**
8	0,4613 (0,14)
16	0,6009 (0,13)
32	0,7142 (0,12)
64	0,8204 (0,08)
128	0,8987 (0,06)
256	0,9542 (0,02)
512	0,9852 (0,01)
1.024	0,9983 (0,01)

Fonte: O autor (2020).

Como pode ser observado na Tabela 10, a acurácia dos classificadores cresceu com o aumento do comprimento dos espectros, *e.g.*, a acurácia cresceu aproximadamente 116,41% entre os comprimentos 8 e 1.024. Uma possível razão para este comportamento é a maior quantidade de informação disponível para os classificadores. Uma segunda razão poderia ser a melhoria na qualidade da informação, pois, com mais elementos, é possível obter espectros mais detalhados de cada condição de operação do dispositivo. Em outras palavras, a representação dos sinais no domínio das frequências se torna mais detalhada quando o espectro apresenta uma resolução mais alta. Desta forma, o espectro pode carregar mais informações discriminativas.

Conforme discutido em 3.1.1, duas medidas foram utilizadas para avaliar a qualidade da informação relacionada aos diferentes tamanhos de espectros, mas também para auxiliar a entender o desempenho dos classificadores. As medidas foram a divergência do Kullback-Leibler e o teste de Kolmogorov-Smirnov. Para essa avaliação, foram calculados os valores da razão (R) entre os valores médios das medidas interclasse e intraclasse da divergência de KL e do teste de KS, considerando cada comprimento de espectro.

As Tabelas 11 e 12 listam os valores médios da divergência de KL e do teste de KS entre pares de classes, respectivamente. Tais resultados estão relacionados aos espectros de frequência com 1.024 elementos. Para obter o raio R, inicialmente se calcula os valores médios interclasse e intraclasse das medidas de dissimilaridade entre todos os pares de espectros pertencentes às duas classes consideradas. Os elementos na diagonal principal da tabela fazem referência às medidas entre dados pertencentes a classes iguais, *e.g.*, P1 e P1, P2 e P2, etc. Por outro lado, os demais elementos se referem a classes diferentes, *e.g.*, P1 e P2, P2 e P4, etc. Desta forma, a razão R é obtida dividindo-se a média dos valores fora da diagonal principal, *i.e.*, média interclasse, pela média dos valores na diagonal principal, *i.e.*, média intraclasse. Como exemplo, o valor de R referente à Tabela 11 é 1,4575, pois os valores médios interclasse e intraclasse para a divergência

de KL são 0,0223 e 0,0153, respectivamente. Seguindo um processo semelhante para o teste de KS, utilizando-se os resultados da Tabela 12, obtém-se uma razão igual a 1,4964.

Tabela 11 – Divergência de KL média para cada par de classes, considerando o comprimento de espectro 1.024.

Classes	**P1**	**P2**	**P3**	**P4**
P1	0,0156	0,0248	0,0171	0,0195
P2	0,0241	0,0178	0,0262	0,0298
P3	0,0174	0,0271	0,0141	0,0158
P4	0,0197	0,0308	0,0157	0,0137

Fonte: O autor (2020).

Tabela 12 – Teste de KS médio para cada par de classes, considerando o comprimento de espectro 1.024.

Classes	**P1**	**P2**	**P3**	**P4**
P1	0,1205	0,1958	0,1382	0,1299
P2	0,1958	0,1032	0,2064	0,2094
P3	0,1382	0,2064	0,1078	0,1115
P4	0,1299	0,2094	0,1115	0,1100

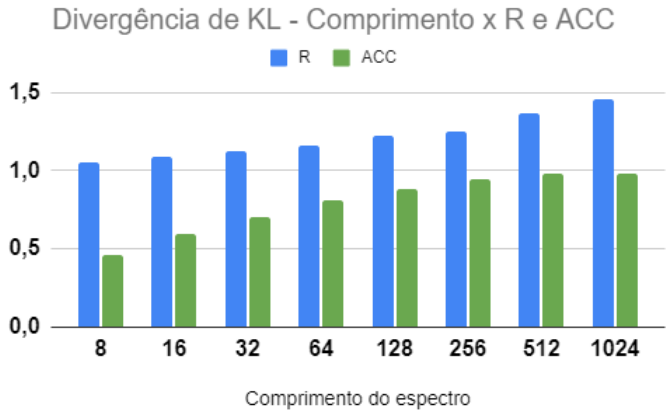
Fonte: O autor (2020).

Repetindo-se o processo para todos os comprimentos, obtém-se os resultados apresentados nas Figuras 11 e 12. As barras verdes representam os valores de acurácia listados na Tabela 10, enquanto as barras azuis representam os valores da razão R para a divergência de KL (Figura 11) e para o teste de KS (Figura 12). Observando essas figuras, nota-se que o valor de acurácia tende a aumentar à medida que a razão R aumenta. Isto ocorre para a razão R calculada tanto por meio da divergência de KL, quanto para a calculada pelo teste de KS, sugerindo que as razões entre as médias dos valores interclasse e intraclasse dessas medidas estão relacionadas à dificuldade ou facilidade de se treinar modelos precisos.

Este mesmo processo de análise pode ser realizado com o conjunto de dados que combina falhas de mancais e válvulas em compressores. Este cenário é mais complexo que o anterior, pois o conjunto de dados apresenta 13 classes de condições de operação do dispositivo, conforme visto na Tabela 2. As entradas dos modelos treinados nesta etapa também são espectros de frequência e seus comprimentos variam de 8 a 1.024. A Tabela 13 lista os valores médios de acurácia dos classificadores treinados. Os resultados estão organizados de acordo com os tamanhos de entrada e remetem apenas ao conjunto de teste. Os valores entre parênteses representam os desvios padrões.

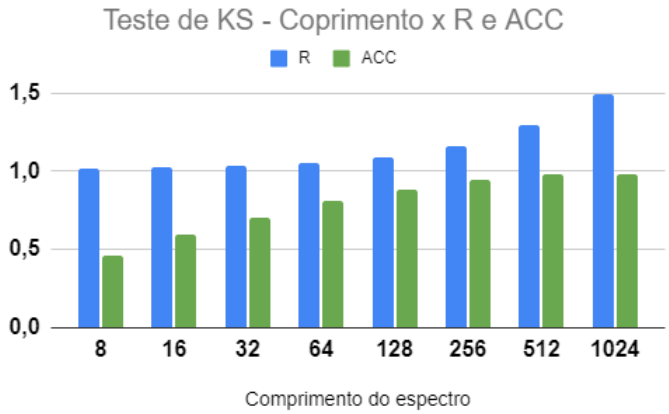
A Tabela 13 mostra uma tendência similar à apresentada na Tabela 10. Em outras palavras, os classificadores tornaram-se mais precisos à medida que o comprimento dos espectros aumentou. A acurácia aumentou 475%, de 0,1236 a 0,7109, com a variação do espectro de frequências de 8 a 1.024 elementos. Por outro lado, observa-se que o maior valor médio de

Figura 11 – Razões obtidas com divergência de KL e acurácias médias de classificadores para diferentes comprimentos de espectros.



Fonte: O autor (2020).

Figura 12 – Razões obtidas com o teste de KS e acurácias médias de classificadores para diferentes comprimentos de espectros.



Fonte: O autor (2020).

Tabela 13 – Acurácia média de modelos treinados com espectros de frequência de diferentes comprimentos para diagnosticar falhas combinadas em mancais e válvulas.

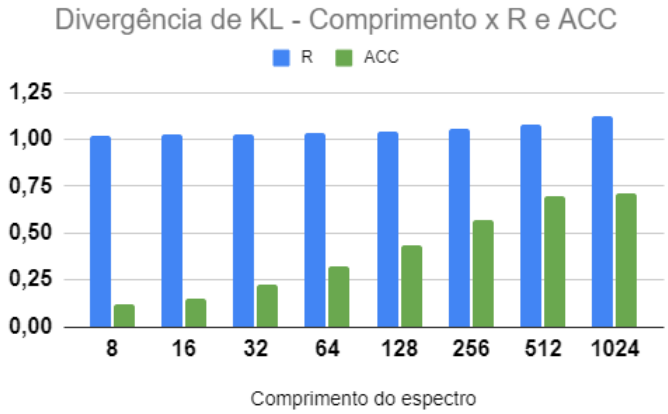
Input Size	**Accuracy**
8	0,1236 (0,02)
16	0,1537 (0,05)
32	0,2237 (0,11)
64	0,3234 (0,23)
128	0,4373 (0,15)
256	0,5724 (0,12)
512	0,6948 (0,11)
1.024	0,7109 (0,10)

Fonte: O autor (2020).

acurácia obtido ainda foi inferior ao maior valor dos modelos treinados com dados relacionados a falhas em mancais.

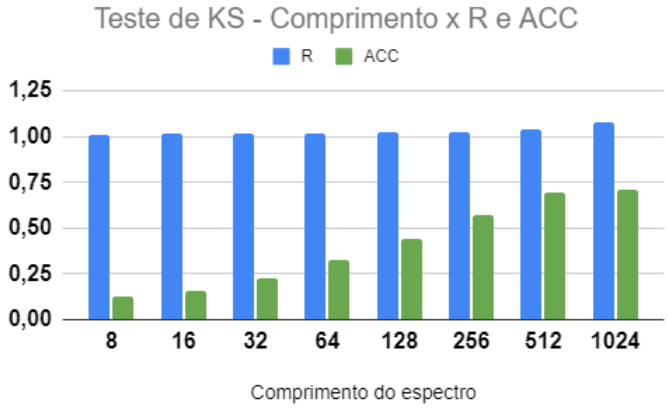
A análise realizada com as falhas em mancais, isto é, aquela relacionando a acurácia média dos modelos treinados com a razão entre as medidas de dissimilaridade interclasse e intraclasse, também foi realizada para este conjunto de dados. Os resultados são apresentados nas Figuras 13 e 14. As barras verdes representam os valores médios de acurácia listados na Tabela 13, enquanto as barras azuis representam os valores da razão R para a divergência de KL (Figura 13) e para o teste de KS (Figura 14).

Figura 13 – Razões obtidas com a divergência de KL e acurácias médias de classificadores para diferentes comprimentos de espectros.



Fonte: O autor (2020).

Figura 14 – Razões obtidas com o teste de KS e acurácias médias de classificadores para diferentes comprimentos de espectros.



Fonte: O autor (2020).

Observa-se nestas duas imagens que a acurácia aumentou à medida que a razão R aumentou. Isto ocorreu tanto para a divergência de KL quanto para o teste de KS, reforçando a hipótese de que as razões entre as medidas de dissimilaridades médias interclasse e intraclasse possam estar associadas à facilidade ou dificuldade de se treinar modelos precisos a partir de um determinado conjunto de dados.

Além disso, as razões R neste cenário são inferiores às obtidas no cenário com falhas em mancais, o que pode estar relacionado à menor acurácia dos modelos. Isto faz sentido, uma vez que as razões próximas a 1 indicam que o nível de dissimilaridade entre espectros pertencentes à mesma classe é semelhante ao da dissimilaridade entre espectros pertencentes à classes diferentes, dificultando o aprendizado de padrões discriminativos.

3.2.2.2 Espectrogramas

A primeira análise desta etapa trata das falhas em mancais. Desta vez, os dados de entrada dos modelos são espectrogramas. Diferentemente dos espectros de frequência, a informação está organizada na forma de matrizes bidimensionais. Conforme mencionado anteriormente, eles mostram como os componentes de frequência de um sinal variam ao longo do tempo. A Tabela 14 lista as acurácias médias dos classificadores treinados com espectrogramas relacionados a falhas em mancais. Os resultados foram ordenados de acordo com a resolução em frequência e estão relacionados apenas a amostras de teste. Os valores entre parênteses representam os desvios padrões.

Tabela 14 – Acurácia média de modelos treinados com espectrogramas de diferentes resoluções em frequência, *i.e.*, dimensões, para diagnosticar falhas em mancais.

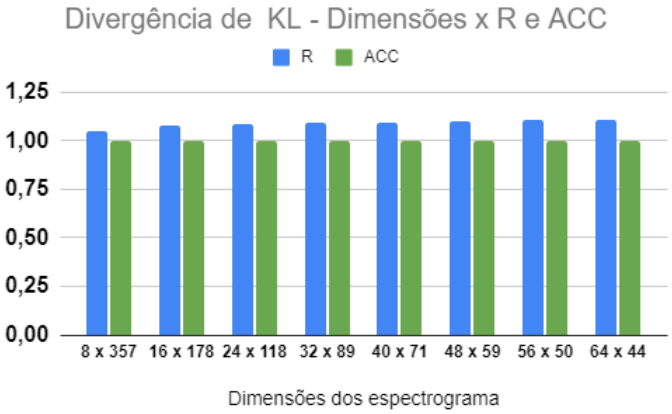
Dimensões	**Acurácia**
8 x 357	0,9980 (0,01)
16 x 178	0,9994 (0,00)
24 x 118	1,0000 (0,00)
32 x 89	1,0000 (0,00)
40 x 71	1,0000 (0,00)
48 x 59	1,0000 (0,00)
56 x 50	1,0000 (0,00)
64 x 44	1,0000 (0,00)

Fonte: O autor (2020).

A tendência observada na Tabela 14 é semelhante às vistas nas Tabelas 10 e 13. Em outras palavras, a acurácia melhorou à medida que a quantidade de elementos na frequência aumentou. No entanto, na Tabela 14 os classificadores atingiram uma acurácia de 100% com resoluções em frequência menores, *e.g.*, 24 elementos. Isto provavelmente ocorreu por conta da informação temporal complementar fornecida pelo janelamento dos sinais na construção dos espectrogramas.

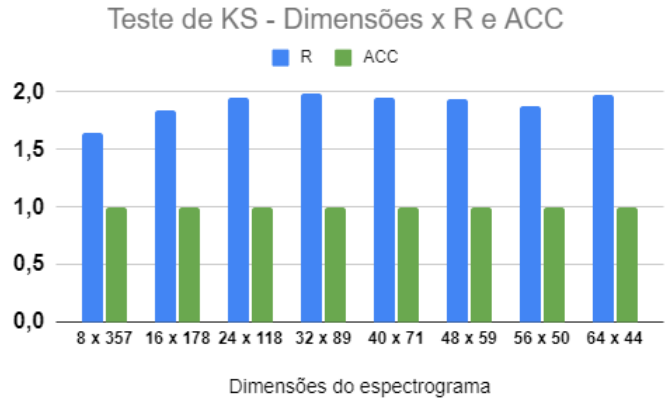
A análise envolvendo a razão R e as acurácias médias, realizada no Cenário 1, também foi desempenhada com os espectrogramas. Os resultados desta análise podem ser observados nas Figuras 15 e 16, para a divergência de KL e para o teste de KS, respectivamente. As barras verdes e azuis, assim como no caso anterior, representam a acurácia média do modelo e a razão R, respectivamente.

Figura 15 – Razões obtidas com a divergência de KL e acurácias médias de classificadores para diferentes dimensões de espectrogramas.



Fonte: O autor (2020).

Figura 16 – Razões obtidas com o teste de KS e acurácias médias de classificadores para diferentes dimensões de espectrogramas.



Fonte: O autor (2020).

A Tabela 14 e as Figuras 15 e 16 mostram que as acurácias dos modelos atingiram seu valor máximo, *i.e.*, 100%, para a maior parte das dimensões de espectrogramas. As exceções foram os formatos 8 x 357 e 16 x 178, que representaram valores de acurácia iguais a 0,9980 e 0,9994, respectivamente. Ainda assim, mesmo próximo do maior valor possível de acurácia, estas dimensões de espectrogramas estiveram relacionadas aos menores valores de R, que foram 1,05 e 1,07 para a divergência de KL e 1,64 e 1,84 para o teste de KS, respectivamente. Maiores aumentos nos valores da razão R não resultaram em mudanças significativas, dado que a acurácia já estava próxima ao seu valor máximo.

Esta análise também foi realizada para a base de dados com falhas combinadas em mancais e válvulas, *i.e.*, a com 13 classes. As entradas também são espectrogramas com diferentes dimensões, variando de 8 x 357 a 64 x 44. A Tabela 15 lista a acurácia média dos classificadores treinados. Os resultados foram ordenados de acordo com a resolução em frequência e estão relacionados apenas a amostras de teste. Os valores entre parênteses representam os desvios

padrões.

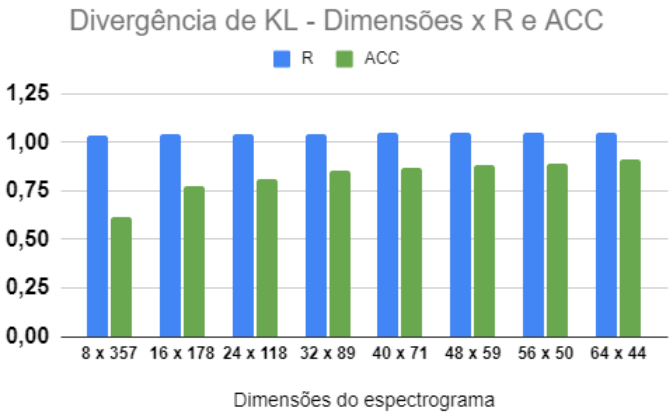
Tabela 15 – Acurácia média de modelos treinados com espectrogramas de diferentes resoluções em frequência (dimensões) para diagnosticar falhas combinadas em mancais e válvulas.

Dimensões	Acurácia
8 x 357	0,6171 (0,12)
16 x 178	0,7744 (0,09)
24 x 118	0,8095 (0,08)
32 x 89	0,8567 (0,06)
40 x 71	0,8660 (0,06)
48 x 59	0,8837 (0,05)
56 x 50	0,8916 (0,06)
64 x 44	0,9127 (0,05)

Fonte: O autor (2020).

Neste cenário, os valores de acurácia aumentaram à medida que a resolução em frequência aumentou, assim como nos experimentos anteriores. As Figuras 17 e 18 mostram como a razão entre os valores médios das medidas de dissimilaridade interclasse e intraclasse variam com as dimensões dos espectrogramas. Também observa-se que a razão R e a acurácia dos classificadores aumentou de 1,03 a 1,05 (divergência de KL) e de 1,50 a 1,53 (teste de KS) à medida que os formatos dos espectrogramas variaram de 8 x 357 a 64 x 44. No entanto, o valor da razão cresceu a uma taxa muito menor que a acurácia. Isso pode ter ocorrido por conta do elevado número de classes que o problema apresenta, que aumenta as chances de haver grupos com maior semelhança entre si.

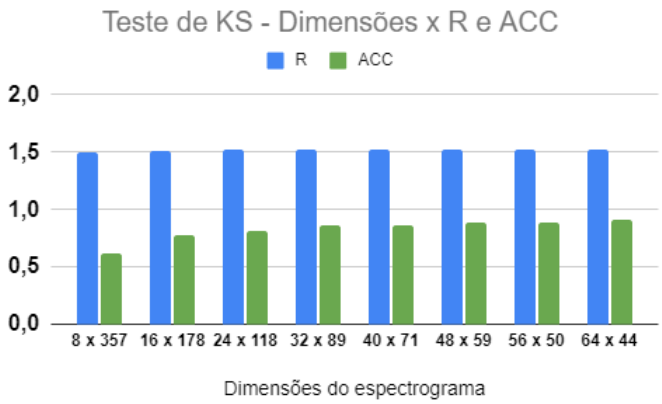
Figura 17 – Razões obtidas com a divergência de KL e acurácias médias de classificadores para diferentes dimensões de espectrogramas.



Fonte: O autor (2020).

Os resultados obtidos com estes experimentos levam a algumas considerações. A primeira remete ao desempenho dos classificadores em cada conjunto de dados. Os classificadores treinados para realizar o diagnóstico de falhas em mancais foram os que apresentaram melhor

Figura 18 – Razões obtidas com o teste de KS e acurácias médias de classificadores para diferentes dimensões de espectrogramas.



Fonte: O autor (2020).

desempenho em termos de acurácia. Este resultado está de acordo com a hipótese levantada na seção de metodologia, pois a base de dados com falhas em mancais esteve atrelada aos maiores valores da razão entre as medidas de dissimilaridade interclasse e intraclasse. Isto mostra que as classes podem ser discriminadas com maior facilidade.

A segunda consideração remete ao valor da razão e ao desempenho dos classificadores de diferentes tamanhos. Houve uma tendência global de melhoria de acurácia com o aumento dos valores da razão R, embora essa melhoria tenha se dado em graus diferentes para espectrogramas e espectros de frequência. Esta tendência foi observada tanto para a divergência de KL, quanto para o teste de KS. Esta informação pode ser útil na escolha do formato de entrada mais adequado para um dado problema de classificação, sem a necessidade de treinar modelos com todos estes formatos. Além disso, o processo de análise pode ser acelerado escolhendo-se subconjuntos de dados pertencentes à cada classe, em vez de utilizar todo o conjunto.

Um exemplo pode ser visto nas Figuras 11 e 12. A razão obtida pelo comprimento 8 do dado de entrada esteve muito próximo a 1. Isto significa que as dissimilaridades intraclasse estão muito próximas às dissimilaridades interclasse, sugerindo que os classificadores tenham uma maior dificuldade de aprender padrões discriminativos suficientes para identificar as classes. Desta forma, os níveis de acurácia dos modelos treinados tendem a ser menos satisfatórios. Por outro lado, o comprimento 1.024 esteve associado a uma razão próxima a 1,5, indicando uma maior facilidade para identificar as classes corretamente. De fato, os classificadores apresentaram maiores valores de acurácia para este comprimento do espectro de frequências.

3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

O capítulo tratou do uso de uma medida de dissimilaridade e de um teste estatístico (também utilizado como medida de dissimilaridade) na escolha de dados adequados para o diagnóstico de falhas. "Adequado", neste contexto, significa que existe informação discriminativa

entre as classes, facilitando o processo de classificação. Os dados em questão foram espectros de frequência e espectrogramas de diferentes tamanhos, obtidos a partir de sinais coletados por diferentes sensores. A medida de dissimilaridade e o teste estatístico foram a divergência de Kullback-Leibler e o teste e Kolmogorov-Smirnov, respectivamente.

A divergência de KL e o teste de KS foram utilizados no cálculo de uma razão R, obtida por meio da divisão da dissimilaridade média interclasse pela dissimilaridade média intraclasse. Neste estudo, verificou-se a relação entre a razão R e a acurácia dos classificadores treinados para realizar o diagnóstico de falhas. Esta relação foi avaliada tanto para os diferentes formatos de dados de entrada, quanto para os dados obtidos por diferentes sensores.

Viu-se que os classificadores mais precisos foram treinados com dados associados aos maiores valores de R, enquanto aqueles treinados com dados associados a valores de R próximos a 1 apresentaram o pior desempenho. Este resultado sugere que a razão R pode ser utilizada de forma comparativa na escolha do formato dos dados de entrada, ou mesmo dos dados obtidos por um determinado sensor, para uso em problemas de classificação no domínio abordado neste trabalho. Este processo eliminaria a necessidade de se treinar classificadores com todas as configurações possíveis de dados de entrada, a fim de se escolher a mais adequada. No entanto, mais estudos são necessários para avaliar o uso desta razão em outros tipos de problemas.

A escolha de dados adequados para treinar modelos de classificação é importante quando existe uma variedade de configurações possíveis para estes dados. Dois exemplos foram as duas bases de dados utilizadas neste capítulo, contendo sinais relativos a sensores diferentes, representados de formas também diferentes. Esta forma de análise foi utilizada na escolha de dados utilizados nos estudos realizados em outros capítulos desta tese.

4 DIAGNOSE DE FALHAS DE FORMA SUPERVISIONADA

Conforme visto na Seção 2, classificadores treinados de forma supervisionada também podem ser utilizados para identificar anomalias. Este processo é também conhecido como diagnose de falhas. Para tal, é necessário que a quantidade de amostras de todas as classes, *i.e.*, normais e anômalas, seja suficiente. Isto significa que, além de conhecer o cenário normal, todas as formas de anomalias de interesse devem ser conhecidas e conter informações suficientes para caracterizá-las. Este é o caso da base de dados utilizada no estudo desenvolvido neste capítulo, descrita na seção seguinte.

Apesar da detecção de anomalias envolvendo classificadores treinados de forma supervisionada fugir ao padrão tradicional da detecção de anomalias, *i.e.*, espera-se que as instâncias anômalas sejam raras ou inexistentes, o uso deste tipo de abordagem como ponto de partida pode melhorar o entendimento deste tipo de problema. Como exemplo, é possível utilizar os resultados do problema de classificação tradicional como referência de desempenho para o sistema de detecção de anomalias em desenvolvimento, bem como para auxiliar na identificação das limitações deste último em relação ao primeiro.

Desta forma, o estudo desenvolvido neste capítulo teve como objetivo realizar a detecção de anomalias em caixas de engrenagens por meio do uso de CNNs treinadas de forma supervisionada. Paralelamente, foi feita uma análise acerca da influência de diferentes parâmetros no desempenho do modelo, *e.g.* profundidade do extrator de características.

4.1 METODOLOGIA

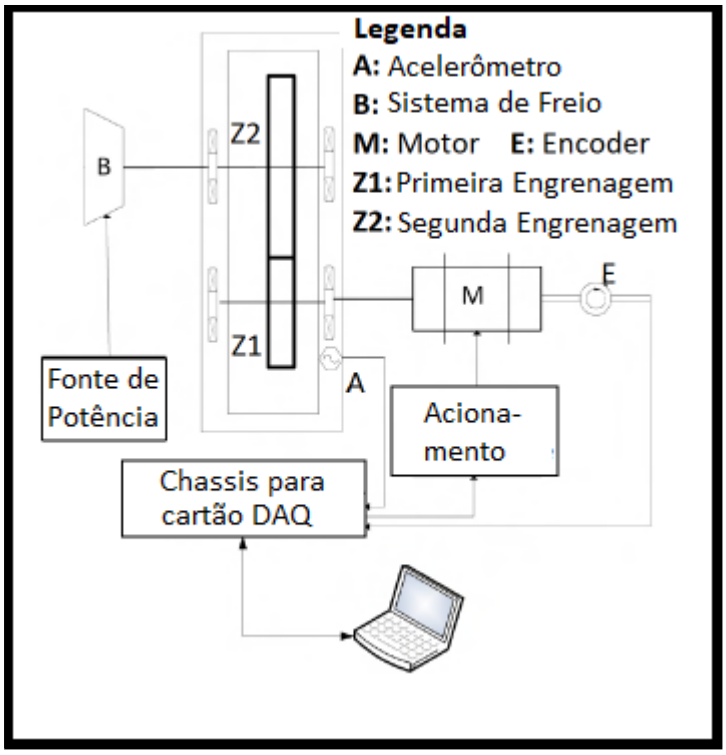
Nesta seção, apresenta-se detalhes sobre os experimentos realizados ao longo estudo desenvolvido no capítulo.

4.1.1 Base de dados

A base de dados utilizada nos experimentos deste capítulo trata da diagnose de falhas em caixas de engrenagens. Ela é constituída de sinais de vibração coletados por um acelerômetro posicionado sobre uma caixa de engrenagens. A Figura 19 ilustra o arranjo experimental utilizado na coleta dos sinais. Um motor elétrico (M) aciona a caixa de engrenagens por meio de um acoplamento. A caixa contém duas engrenagens (Z1 e Z2), que estão montadas em diferentes eixos. O eixo de saída da caixa está acoplado a um freio magnético (B). A Tabela 16 lista as informações relevantes acerca destes componentes.

A velocidade do motor elétrico é controlada por um acionador Danfoss VLT 1:5 kW, sendo o freio magnético acionado por uma fonte de tensão variável TDK Lambda GEN 150-10, 0V-150V, 10A. Os sinais de vibração são coletados por um acelerômetro unidirecional (A) IMI

Figura 19 – Arranjo experimental.



Fonte: O autor (2018).

Tabela 16 – Descrição dos componentes do arranjo experimental.

Componente	Descrição
M	Motor Siemens 1LA7 090-4YA60, 1:49 kW, 4 polos, 28:33 Hz
Z1	Pinhão, 76 mm, 30 dentes, Ângulo de pressão = 20 º, Ângulo de hélice = 20 º
Z2	Engrenagem, 112 mm, 45 dentes, Ângulo de pressão = 20 º, Ângulo de hélice = 20 º
B	Freio magnético, Proporcional à tensão de entrada, Acoplado por correia

Fonte: O autor (2018).

Sensor 603C01, 100 mV/g. Este foi posicionado verticalmente na caixa de engrenagens, próximo ao eixo de entrada, conforme ilustrado na Figura 19. A digitalização dos sinais analógicos foi realizada com o uso de cartões de aquisição de dados NI9234, utilizado para sensores piezoelétricos. Ele tem uma resolução de 24 bits e taxa de amostragem de 50 KHz, permitindo a detecção de frequências de até 25 kHz.

O principal objetivo do experimento com esta base de dados é, por meio de sinais de vibração, avaliar o grau das falhas relacionadas à quebra de dentes em engrenagens helicoidais. Diferentes níveis de danos foram produzidos num dos dentes da engrenagem helicoidal Z1, enquanto a engrenagem Z2 permaneceu sem modificações. Nove níveis de danos foram definidos, resultando num total de 10 cenários ou classes, *i.e.*, um cenário no qual a engrenagem Z1 não

apresenta danos, e nove nos quais apresenta danos em diferentes níveis. A caracterização das dez classes está listada na Tabela 17 e na Figura 20.

Figura 20 – As dez classes de danos no dente da engrenagem Z1.

Fonte: O autor (2018).

Tabela 17 – Níveis de dano no dente da engrenagem Z1.

Código	**Descrição**	**Dano (mm)**	**Dano (%)**
P1	Nível 1 ou sem danos	0,00	0,00
P2	Nível 2	2,37	11,58
P3	Nível 3	4,00	19,58
P4	Nível 4	5,73	28,06
P5	Nível 5	7,60	37,19
P6	Nível 6	10,57	51,71
P7	Nível 7	12,37	60,52
P8	Nível 8	14,33	70,15
P9	Nível 9	17,15	85,64
P10	Nível 10	20,43	100,00

Fonte: O autor (2018).

Os experimentos foram realizados de acordo com 15 cenários de operação, que correspondem à combinação de cinco frequências de rotação do motor elétrico e três níveis de

tensão de entrada do freio magnético. Das frequências de rotação, três são constantes e duas são variáveis. Elas estão listadas na Tabela 18. Em relação aos níveis de tensão aplicados ao freio magnético, todos são constantes e estão listados na Tabela 19. Desta forma, a combinação de dez cenários de falha com quinze cenários de operação resultou em 150 sinais. Sendo cada experimento realizado três vezes, chegou-se a um total de 450 sinais. Cada sinal foi normalizado para a faixa de valores entre 0 e 1, e divido em 40 sinais menores com duração de 0,25 segundos. Desta forma, o número total de sinais na base de dados tornou-se 18.000.

Tabela 18 – Cenários de frequências.

Código	**Tipo**	**Frequência de rotação (Hz)**	**Perfil**
F1	Constante	8	-
F2	Constante	12	-
F3	Constante	15	-
F4	Variável	8-15	Senoidal, Período de 2 s
F5	Variável	8-15	Quadrado, Período de 2 s

Fonte: O autor (2018).

Tabela 19 – Cenários de cargas.

Código	**Tensão**
L1	0 V (Sem carga)
L2	10 V
L3	30 V

Fonte: O autor (2018).

4.1.2 Organização da base de dados

Os sinais de vibração foram representados por meio de espectrogramas obtidos via transformada de Fourier de tempo curto. A STFT realiza o janelamento do sinal, permitindo visualizar como o espectro de frequências deste evolui ao longo do tempo, *i.e.* trata-se de uma representação em três dimensões: tempo, frequência e magnitude. Tais espectrogramas podem ser representados por matrizes bidimensionais, nas quais cada coordenada traz um valor numérico de magnitude relativa a uma dada frequência, num dado instante de tempo. Dentre as características da STFT que justificaram a sua adoção, estão o baixo custo computacional e a facilidade de interpretação quando comparada a outras técnicas (XIA et al., 2017), *e.g.*, transformadas *wavelets* (CHUI, 2016). O custo computacional torna-se uma característica particularmente importante para cenários industriais, dada a necessidade de aplicações em tempo real e a limitação de *hardware* em muitas situações.

Como parâmetros da STFT, foram adotadas a janela de Hamming de tamanho 128 e uma sobreposição de 50%. A janela de Hamming foi adotada devido a sua propriedade seletiva, enquanto os 50% de sobreposição trazem equilíbrio entre custo computacional baixo e transições suaves no espectro de frequências (OPPENHEIM; SCHAFER; BACK, 2015). A informação

dos espectrogramas foi condensada em imagens do tipo RGB, *i.e.*, com 3 canais, com 175 x 175 pixels, também normalizadas entre 0 e 1. A escolha do sistema de três canais se deu como uma forma de fornecer uma quantidade maior de informações para o classificador e melhorar seu desempenho. O problema associado a esta escolha é aumentar o custo computacional do processo de treinamento do modelo, bem como o de sua execução.

4.1.3 Modelo de classificação

O classificador utilizado foi uma rede neural convolucional. A CNN é uma ferramenta de uso frequente em trabalhos da literatura, adequada para problemas de classificação (RAWAT; WANG, 2017). Isto inclui a diagnose de falhas em caixas de engrenagens, como mostrado por Jing *et al*. (JING et al., 2017), Xia *et al*. (XIA et al., 2017), Zeng *et al*. (ZENG; LIAO; LI, 2016) e Liao *et al*. (LIAO; ZENG; LI, 2017).

4.1.4 Arranjo Experimental

Nos experimentos foram avaliadas três configurações de classificadores. Cada modelo foi composto por uma base convolucional e uma etapa de classificação. A base convolucional realiza a extração de características de forma implícita. Os modelos testados apresentaram bases convolucionais de profundidades diferentes, sendo o objetivo analisar a influência da profundidade destas no desempenho da rede. A organização das bases convolucionais está listada nas Tabelas 20 e 21. O estágio de classificação foi o mesmo para os três modelos: uma rede neural artificial do tipo MLP com 512 neurônios na camada oculta. O estágio de classificação interpreta as características extraídas pela base convolucional e identifica o nível de quebra do dente. A função de ativação ReLU (do inglês *Rectified Linear Unit*) foi utilizada pelos neurônios da base convolucional e do estágio de classificação, excetuando-se a camada de saída, que apresentou a função de ativação *Softmax*. O modelo completo foi treinado por 50 épocas com o algoritmo *backpropagation* (BISHOP et al., 1995), juntamente com o otimizador Adam (KINGMA; BA, 2014) e uma taxa de aprendizado inicial de 0,001.

Tabela 20 – Camadas utilizadas na construção das bases convolucionais.

Código da camada	Tipo	Configuração
C1	Convolucional	32 filtros 3x3
C2	*Max-pooling*	32 filtros 2x2
C3	Convolucional	64 filtros 3x3
C4	*Max-pooling*	64 filtros 2x2
C5	Convolucional	128 filtros 3x3
C6	*Max-pooling*	128 filtros 2x2
C7	Convolucional	256 filtros 3x3
C8	*Max-pooling*	256 filtros 2x2

Fonte: O autor (2018).

Tabela 21 – Camadas utilizadas nas bases convolucionais de cada um dos modelos.

Código do modelo	**Camadas**
Q1	C1, C2, C3, C4, C5, C6, C7, C8
Q2	C1, C2, C3, C4, C5, C6
Q3	C1, C2, C3, C4

Fonte: O autor (2018).

A influência do número de condições de operação no desempenho do classificador também foi analisada. Muitos trabalhos da literatura, a exemplo de (ZENG; LIAO; LI, 2016) e (LIAO; ZENG; LI, 2017), realizam a coleta de sinais e os experimentos sob condições controladas, *e.g.* carga e velocidade de rotação constantes. Em contrapartida, aplicações reais podem estar sujeitas a condições de operação variadas, *e.g.* caixas de engrenagens em uma linha de montagem, cuja velocidade de operação está sujeita ao ritmo de produção. Este, por sua vez, pode variar de acordo com a demanda. Tal fato torna necessário o uso de algoritmos capazes de lidar com as diferentes condições de operação do dispositivo monitorado, *e.g.*, diferentes velocidades e cargas, de modo a causar o mínimo de interferência possível no processo no qual a caixa de engrenagens está inserida.

Três cenários foram considerados nestes experimentos. Cada cenário corresponde a um número de condições de operação envolvidas na coleta dos sinais, como listado na Tabela 22. Em cada cenário foi considerada a mesma quantidade amostras de treino, independente da quantidade de condições de operação envolvidas. O significado dos códigos utilizados na Tabela 22, *e.g.* F1 e L2, encontra-se nas Tabelas 18 e 19. A escolha das frequências e cargas utilizadas nesta etapa foi feita de modo aleatório, uma vez que o foco é apenas avaliar a influência da quantidade de condições de operações envolvidas no processo de treinamento do classificador, e não avaliar a influência de condições específicas.

Tabela 22 – Condições de operação.

Quantidade de condições de operação	**Condições de operação**			
1	F1, L1	-	-	-
2	F1, L1	F5, L3	-	-
4	F1, L1	F5, L3	F2, L2	F4, L2

Fonte: O autor (2018).

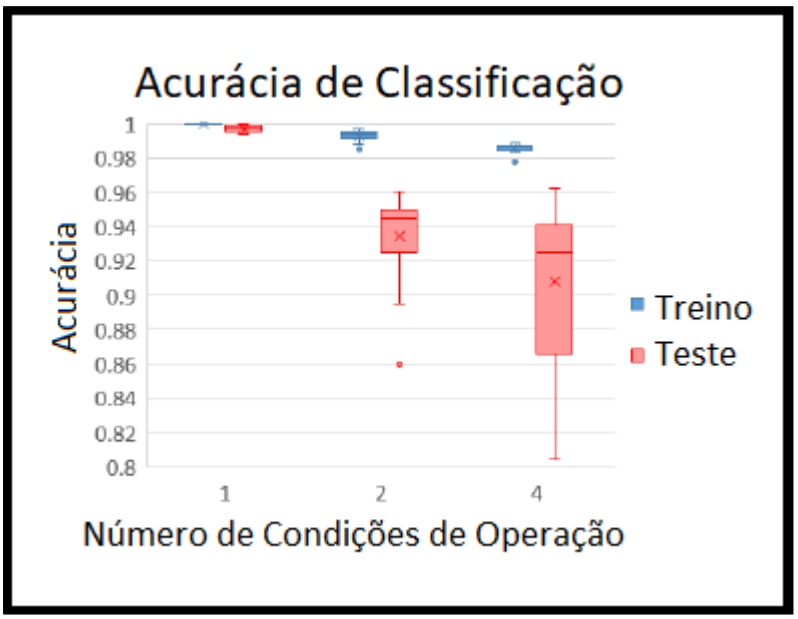
O processo de treinamento dos modelos em cada grupo de experimentos foi realizado considerando-se bases convolucionais e estágios de classificação como uma única estrutura. 50% dos espectrogramas foram utilizados para treinar os modelos, enquanto 25% foram utilizados para validação e 25% para treinamento. O processo foi realizado num computador com as seguintes configurações: OS Ubuntu 16.04 LTS, 64 bits, Memória 15.6 GB, Processador *Intel Xeon* (R) CPU E 5-2609 v3 @ 1,90 GHz x 12, *Graphics Gallium* 0.4 on NV117.

4.2 RESULTADOS E DISCUSSÕES

A influência do número de condições de operação no desempenho do classificador foi o primeiro aspecto analisado. Um total de 30 redes neurais convolucionais estruturadas de acordo com a configuração Q3 foram treinadas em cada um dos três cenários descritos na Seção 4.1.4, visando-se atingir resultados estatisticamente relevantes. A configuração Q3 foi escolhida nesta etapa de forma aleatória, dado que o objetivo do experimento é avaliar a influência da quantidade de condições de operação envolvidas no processo de treinamento do classificador, independente do classificador. Em cada cenário foram utilizados 400 espectrogramas para realizar o treinamento da rede, divididos entre as 10 classes de falhas listadas na Tabela 17.

O desempenho dos sistemas de diagnóstico de falhas foi avaliado de acordo com a acurácia. Este valor leva em consideração 200 imagens de teste pertencentes a cada uma das 10 classes de falhas. A Figura 21 e a Tabela 23 mostram o desempenho dos sistemas para o número de condições de operações variando de 1 a 4.

Figura 21 – Acurácia de 30 classificadores para os conjuntos de treinamento e de teste de acordo com diferentes condições de operação.



Fonte: O autor (2018).

Tabela 23 – Acurácia média e desvio padrão de 30 classificadores para os conjuntos de treinamento e de teste, de acordo com o número de condições de operação.

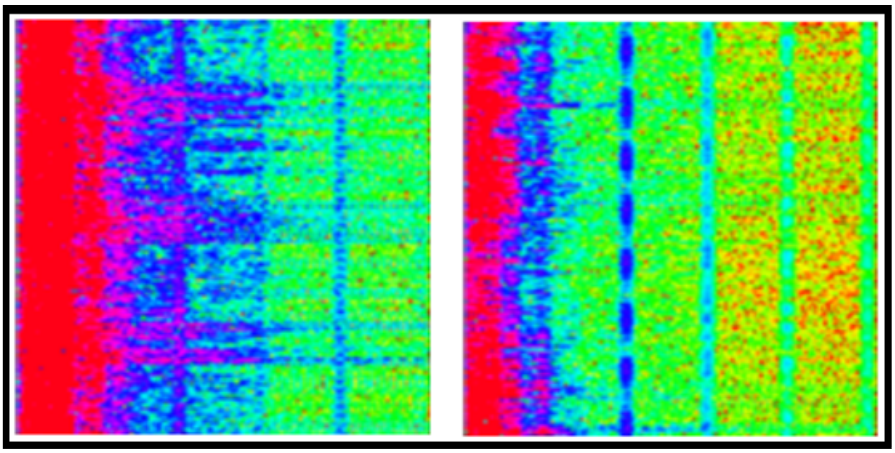
Condições de operação	1	2	4
Treino (Acurácia Média)	1	0,9930	0,9857
Treino (Desvio Padrão)	0	0,0031	0,0065
Teste (Acurácia Média)	0,9971	0,9343	0,9082
Teste (Desvio Padrão)	0,0019	0,0262	0,0466

Fonte: O autor (2018).

Vê-se na Figura 21 e na Tabela 23 que ocorreu uma pequena redução no desempenho do classificador, no que diz respeito aos resultados obtidos durante o processo de treinamento. A queda foi inferior a 1,5%, considerando os cenários com uma e quatro condições de operação. Por outro lado, levando-se em consideração os resultados obtidos com os conjuntos de teste, a queda de desempenho do cenário com uma condição de operação para o cenário com quatro foi de quase 10%. Isto sugere que o aumento no número de condições de operação reduz a capacidade de generalização do modelo. Tal comportamento era esperado e está relacionado à maior variedade de sinais pertencentes a cada classe de falha. Isto significa que o modelo deve aprender uma quantidade maior de padrões, exigindo um processo de treinamento mais longo ou com um maior número de amostras de treino.

Outro problema pode decorrer da variedade de padrões inerentes a cada classe. Os espectrogramas pertencentes a uma das classes podem se assemelhar mais aos espectrogramas de outras classes, que aos espectrogramas da própria classe, conforme mostrado nas Figuras 22 e 23. Isto pode dificultar o processo de treinamento do classificador.

Figura 22 – Dois espectrogramas pertencentes à classe P5.

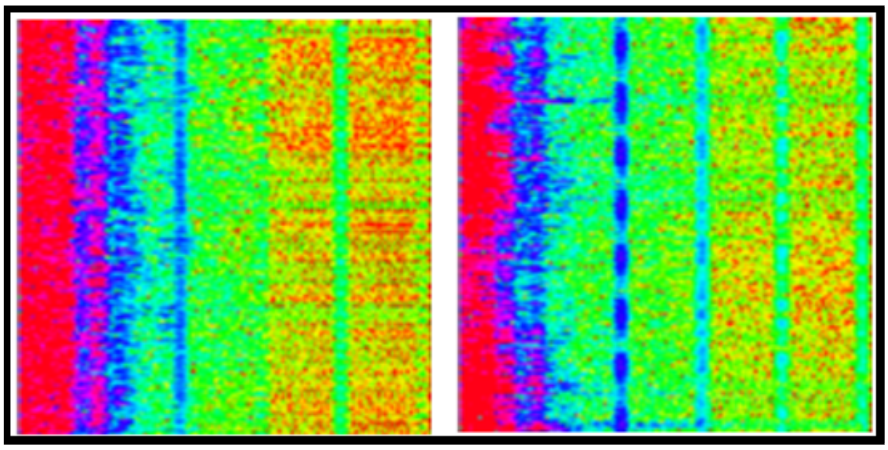


Fonte: O autor (2018).

Ciente deste comportamento, é necessário assegurar a capacidade de generalização da rede neural. Esta necessidade é particularmente importante, principalmente pela quantidade de condições de operação consideradas neste estudo, *i.e.*, 15. Uma boa capacidade de generalização pode ser atingida de diversas formas, como aumentar a quantidade de imagens para treinar o modelo ou o uso de classificadores mais robustos. Como a disponibilidade de sinais neste estudo foi suficientemente grande, *i.e.*, um total de 18.000, a capacidade de generalização dos modelos não se mostrou um problema, mesmo com as 15 condições de operação consideradas.

O segundo aspecto discutido neste estudo trata da influência da profundidade da base convolucional no desempenho do modelo. Como mencionado em 4.1.4, três configurações de CNN foram avaliadas. Os novos experimentos consideraram as 15 condições de operação presentes na base de espectrogramas. Para treinar o modelo foram utilizadas 50% das 18.000 imagens da base. Para cada configuração de modelo, *i.e.* Q1, Q2 e Q3, foram treinadas 30 redes

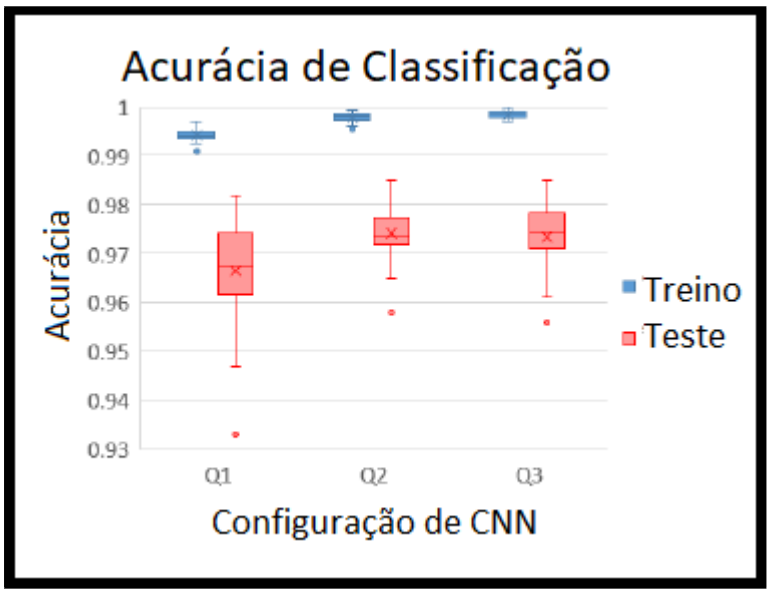
Figura 23 – Um espectrograma pertencente à classe P2 (esquerda) e outro pertencente à classe P5 (direita).



Fonte: O autor (2018).

neurais. O desempenho do sistema de classificação foi avaliado de acordo com 3 aspectos: a acurácia média do classificador, o tempo de treinamento e o tempo de classificação, dado uma quantidade disponibilizada de amostras de teste. A Figura 24 e a Tabela 24 mostram a acurácia média e o desvio padrão dos processos de treino e de teste para as CNNs de cada configuração. O tempo requerido para treinar os modelos e para realizar uma classificação estão ilustrados nas Figuras 25 e 26 respectivamente.

Figura 24 – Acurácia de treino e de teste para as três configurações de CNN.



Fonte: O autor (2018).

A Figura 24 mostra que a acurácia do classificador aumentou com a redução do número de camadas na base convolucional, variando de uma taxa média de 96,6% em Q1 para 97,4% em Q2, considerando as instâncias de teste. Uma possível justificativa é a quantidade de parâmetros treináveis, que aumentou de 11 milhões em Q1 para quase 58 milhões em Q3. O maior número

Tabela 24 – Acurácia média e desvio padrão de 30 classificadores para os conjuntos de treino e de teste, de acordo com a configuração da CNN.

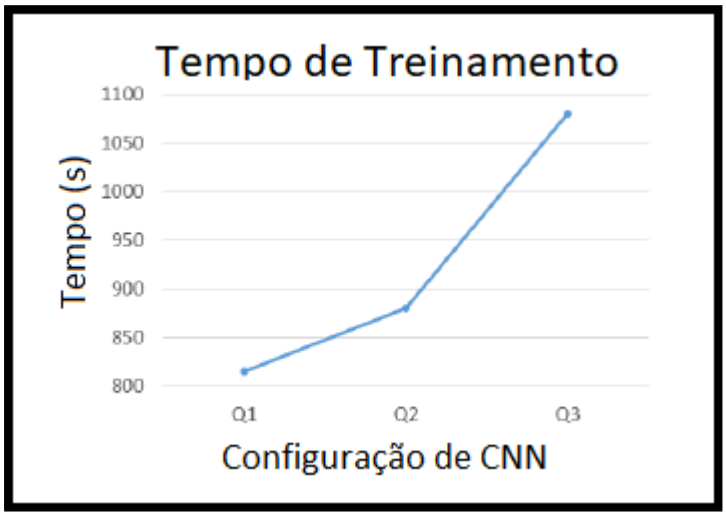
Configurações	Q1	Q2	Q3
Treino (Acurácia Média)	0,9943	0,9978	0,9984
Treino (Desvio Padrão)	0,0013	0,0010	0,007
Teste (Acurácia Média)	0,9664	0,9741	0,9743
Teste (Desvio Padrão)	0,0103	0,0056	0,0069

Fonte: O autor (2018).

de parâmetros treináveis permite que o sistema obtenha um mapeamento mais preciso entre os espectrogramas de entrada e os resultados das classificações. Em contrapartida, o custo computacional do modelo e o tempo de treinamento também aumentam.

Os tempos de treinamento para cada modelo de configurações Q1, Q2 e Q3 foram aproximadamente 815, 880 e 1080 segundos, respectivamente, como observado na Figura 25. Vê-se que o tempo de treinamento subiu aproximadamente 8% de Q1 a Q2, e 32,5% de Q1 a Q3. No que se refere ao tempo necessário para classificar um espectrograma, os valores médios obtidos por cada uma das três configurações foi inferior a 0,055 segundos, conforme ilustrado na Figura 26. Isto sugere que as três configurações podem ser suficientemente rápidas para a utilização em aplicações de tempo real. Das três configurações, a Q3 foi a que apresentou melhor desempenho, com aproximadamente 30 milissegundos para realizar uma classificação. Este fato deve-se ao número menor de camadas convolucionais apresentado pela configuração Q3. Em outras palavras, todos os cálculos inerentes aos filtros das camadas convolucionais C5 e C7 não são realizados nesta configuração.

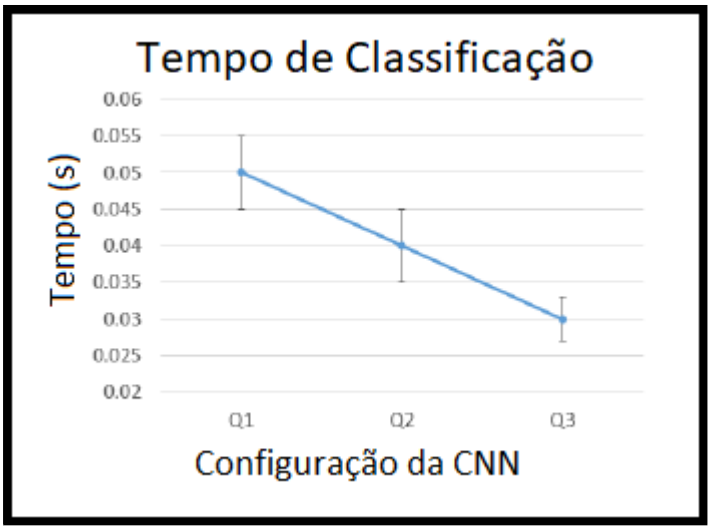
Figura 25 – Tempo de treinamento para as três configurações de CNN.



Fonte: O autor (2018).

Dado que a solução proposta está relacionada a uma aplicação em tempo real, alguns aspectos devem ser destacados. O primeiro refere-se ao tempo de treinamento. Um dos principais requerimentos de sistemas que operam em tempo real é que eles sejam capazes de propiciar

Figura 26 – Tempo de execução para as três configurações de CNN..



Fonte: O autor (2018).

respostas rápidas e precisas. Desta forma, um tempo de treinamento baixo, embora desejável, não é algo obrigatório. O segundo aspecto refere-se ao número de parâmetros treináveis. Quanto maior a quantidade de parâmetros treináveis, maior é o espaço de memória ocupado pelo classificador. Como exemplo, os tamanhos de cada modelo que apresentam estruturas Q1 e Q3 são 84 MB e 441 MB, respectivamente. Modelos muito grandes podem causar problemas relacionados ao armazenamento, dependendo da infraestrutura disponível. Desta forma, a escolha da configuração mais adequada para um problema depende do nível de desempenho necessário e dos recursos disponíveis.

Por fim, os resultados obtidos pela rede neural convolucional foram comparados aos de outra técnica baseada em aprendizado de máquina tradicional, consolidada na literatura. Trata-se da máquina de vetores de suporte. O objetivo é ter um parâmetro de comparação para avaliar quão competitivo o resultado da solução proposta pode ser. Os dados de entrada utilizados foram os mesmos espectrogramas. A extração de características foi feita com o auxílio da técnica SURF (do inglês *Speeded Up Robust Features*) (BAY et al., 2008). 60% das características mais robustas foram utilizadas para criar a bolsa de características, de modo a reduzir o custo computacional deste processo. Porcentagens mais baixas resultaram em perda de acurácia do SVM. As características foram organizadas em bolsas de 500 características por meio do algoritmo de clusteriação k-médias. Foram testados cenários cujas quantidades de características variaram de 300 a 700, sendo os melhores resultados obtidos com 500. 30 SVMs foram treinadas com este conjunto de características. Os resultados estão listados na Tabela 25.

Pode-se ver que o resultado obtido pela CNN mostrou-se competitivo, tanto em termos de acurácia, quanto em termos de tempo de execução. A acurácia média da CNN foi 5,3% maior que a da SVM, com um tempo de classificação aproximadamente 78,6% menor. É provável que o resultado esteja relacionado à maior qualidade das características extraídas pela base

Tabela 25 – Acurácia e tempo de classificação médios para as soluções baseadas em CNN e SVM.

Classificador	**Acurácia média (%)**	**Tempo de classificação médio (s)**
CNN Q3	0,9743	0,03
SVM + Bolsa de características	0,9254	0,14

Fonte: O autor (2018).

convolucional da CNN, dado que o treinamento das estruturas que realizam a extração de características e a classificação é feito de forma conjunta. Isto direciona o aprendizado de características para melhorar o desempenho do problema em questão. Vale ressaltar que tais resultados foram obtidos com a implementação de unidades gráficas de processamento (GPUs, do inglês *graphical processing units*), com bibliotecas otimizadas para acelerar a realização dos cálculos.

4.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Os experimentos desenvolvidos neste capítulo demonstraram a viabilidade do diagnóstico de falhas em caixas de engrenagens com redes neurais convolucionais e espectrogramas obtidos via STFT. Os valores de acurácia atingidos foram superiores a 97%, com tempos de classificação variando entre 0,055 e 0,027 segundos.

Também verificou-se que a variabilidade nas condições de operação da caixa de engrenagens influenciou significativamente a capacidade de generalização do modelo. Isto limita a quantidade de cenários nos quais um classificador pode atuar. Cenários reais, como uma linha de manufatura, podem não apresentar condições controladas o suficiente para a aplicação de um modelo treinado sob condições bastante específicas, como normalmente visto em trabalhos da literatura. Uma possível solução para este problema, adotada neste estudo, foi o uso de uma elevada quantidade de dados pertencentes a cada uma das condições de operação para treinar os modelos.

Outro aspecto analisado foi a relação da profundidade da base convolucional do modelo com o seu desempenho, tanto em termos de acurácia quanto em termos de tempos de treinamento e execução. As variações do tamanho, em MB, e da quantidade de parâmetros treináveis com o aumento da profundidade do modelo também foram investigadas neste capítulo.

Concluindo, o estudo desenvolvido permitiu uma melhor compreensão acerca do funcionamento de redes neurais convolucionais aplicadas à detecção de falhas. Desta modo, torna-se possível pensar em formas de melhorar o desempenho de classificadores neste tipo de problema, sendo o assunto do próximo capítulo.

5 USO DE UM ESTÁGIO DE DECISÃO NA DIAGNOSE SUPERVISIONADA DE FALHAS

O trabalho desenvolvido no capítulo 4 foi um ponto de partida para o estudo da detecção de anomalias. Este capítulo traz uma continuação do trabalho, ainda focando em modelos treinados de forma supervisionada, *i.e.*, na diagnose de falhas.

Sabe-se que um problema comum dos modelos de aprendizado profundo é o custo computacional (RAWAT; WANG, 2017), *e.g.* o treinamento de tais modelos costuma ser longo e demandar o processamento de grande quantidade de informação. Este problema normalmente é superado com o uso de GPUs (ZENG; LIAO; LI, 2016). No entanto, este tipo de *hardware* nem sempre se encontra à disposição, seja por conta do preço mais elevado ou da inviabilidade de implementação. Desta forma, torna-se necessário encontrar meios de se reduzir o custo computacional da solução baseada em aprendizado profundo, mas sem comprometer o seu desempenho em termos de acurácia.

Neste capítulo, avalia-se o uso de um estágio de decisão na saída do sistema de diagnose de falhas, que se trata de um classificador treinado de forma supervisionada. As saídas deste último normalmente representam as probabilidades de que uma dada entrada pertença às categorias de falhas existentes. O modo de falha associado ao maior valor de probabilidade é normalmente escolhido. Embora este tipo de abordagem tenha se mostrado adequado a um grande número de aplicações, a informação fornecida pelas demais saídas do sistema é perdida. Acredita-se que o uso desta informação seja capaz de melhorar o desempenho do classificador.

5.1 METODOLOGIA

Nesta seção, apresenta-se detalhes sobre os experimentos realizados ao longo estudo desenvolvido no capítulo.

5.1.1 Base de dados

A base de dados utilizada foi a apresentada na seção 4.1.1.

5.1.2 Organização da base de dados

A base de dados foi organizada conforme apresentado na seção 4.1.2, isto é, a representação dos sinais se deu por meio de espectrogramas obtidos via STFT. A configuração paramétrica da STFT também foi a mesma: janela de Hamming de tamanho 128 e sobreposição de 50%. A diferença está nos espectrogramas obtidos. No capítulo anterior, a informação dos espectrogramas foi condensada em imagens do tipo RGB com 175 x 175 *pixels*, normalizadas entre 0 e 1.

Neste capítulo, além da forma de representação em RGB, também foi analisado um cenário no qual foram utilizados espectrogramas definidos em escala de cinza.

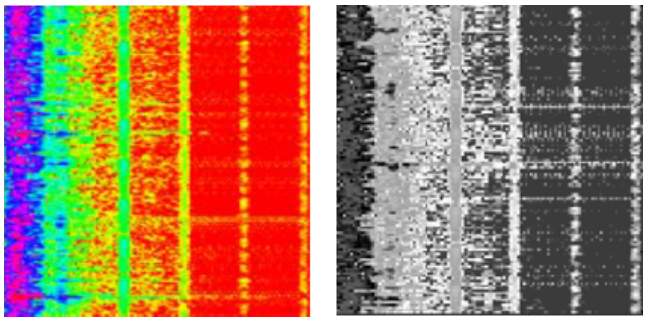
5.1.3 Modelo

O classificador utilizado nos experimentos deste capítulo também foi uma rede neural convolucional. Uma SVM e uma rede neural do tipo MLP foram utilizadas como estágio de decisão, em cenários distintos. Estas visam analisar as saídas da CNN e melhorar o seu desempenho. A SVM, por exemplo, já foi utilizada em aplicações semelhantes a esta, *e.g.* Li *et al*. (LI et al., 2015) propuseram o uso de SVM para combinar os resultados de classificadores que atuam em conjuntos de dados multimodais. Neste caso, as informações a serem combinadas pertencem à mesma modalidade, *i.e.*, tais informações são as saídas de um mesmo modelo.

5.1.4 Arranjo Experimental

Dois cenários experimentais foram desenvolvidos para este estudo. O primeiro deles utiliza imagens RGB, enquanto o segundo utiliza imagens em escala de cinza. As imagens RGB permitem que uma quantidade maior de informação seja utilizada pelo sistema de classificação, dado que artifícios como mapas de calor podem ser utilizados para aumentar a quantidade de informação das entradas do sistema. Por outro lado, o custo computacional cresce em relação ao das imagens em escala de cinza, cuja informação disponível se resume à magnitude da transformada de Fourier. A Figura 27 ilustra espectrogramas pertencentes a estes dois cenários.

Figura 27 – Exemplo de espectrogramas em formato RGB (esquerda) e em escala de cinza (direita).

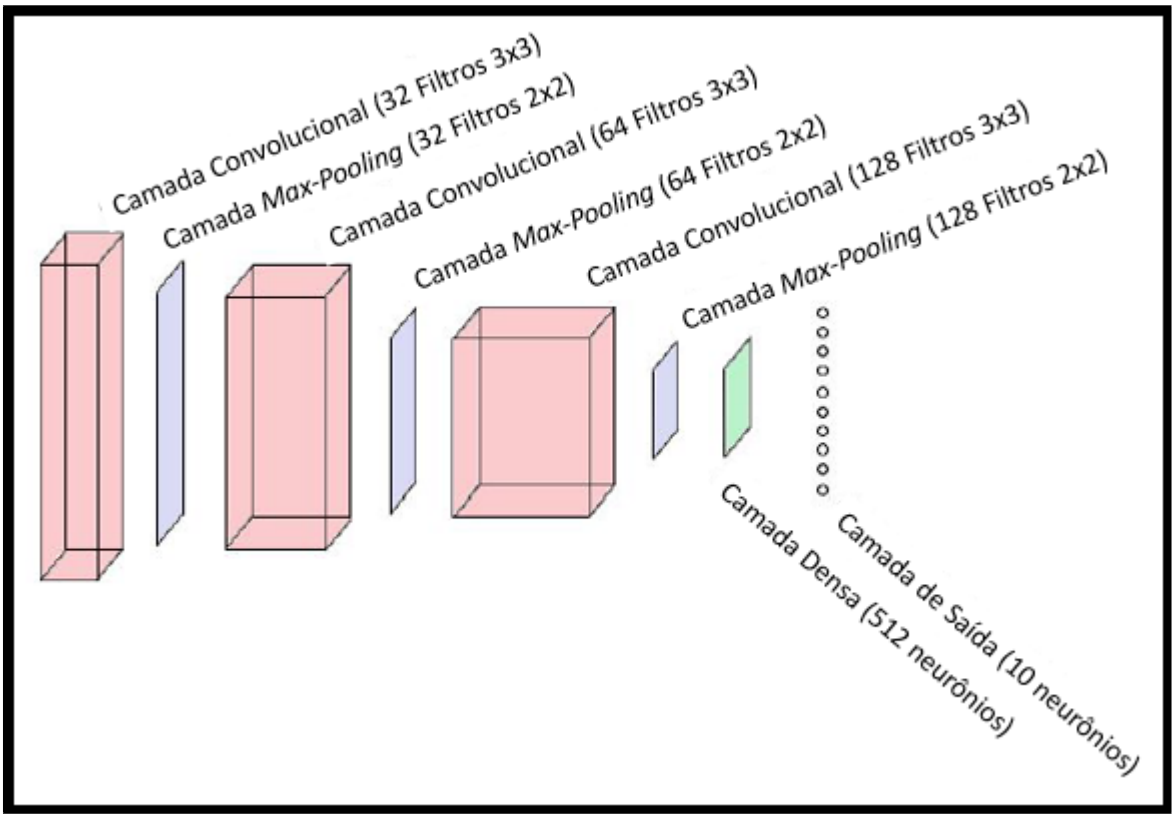


Fonte: O autor (2019).

Nos experimentos foram avaliadas configurações de redes neurais convolucionais compostas por uma base convolucional e uma etapa de classificação. A base convolucional realiza a extração de características de forma implícita. A estrutura da base convolucional utilizada está listada nas Tabelas 5 e 6, indicada pelo código Q2, sendo ela composta por três camadas convolucionais e três de subamostragem dispostas de modo alternado. O estágio de classificação é composto por duas camadas: uma camada oculta com 512 neurônios e a camada de saída com dez neurônios. O estágio de classificação interpreta as características extraídas pela base convolucional e identifica o nível de quebra do dente. Dado que as saídas deste modelo representam

valores de probabilidade que variam de 0 a 1, a função de ativação utilizada na camada de saída foi a *softmax*. Nas demais camadas foi utilizada a função de ativação ReLU. Esta arquitetura foi adotada devido aos resultados satisfatórios reportados por Monteiro *et al.* (MONTEIRO et al., 2018) no problema de detecção de anomalias em caixas de engrenagens. Tal arquitetura está ilustrada na Figura 28.

Figura 28 – Arquitetura da rede neural convolucional.



Fonte: O autor (2019).

Um estágio de decisão foi adicionado ao topo da rede neural convolucional, de modo a analisar a sua saída e melhorar o seu desempenho. Dois cenários foram montados em relação ao estágio de decisão. No primeiro cenário o estágio é uma SVM e no segundo é uma MLP com 21 neurônios na camada oculta. O processo de treinamento dos modelos (CNN e estágio de decisão) foi realizado separadamente. Inicialmente foi treinada a CNN. O estágio de decisão foi treinado tendo as saídas da CNN como dados de entrada. 50% dos espectrogramas foram utilizados para treinar os modelos, enquanto 25% foram utilizados para validação e 25% para teste. O treinamento ocorreu em cenários com 10 e 50 épocas. A configuração do computador utilizado para treinar o modelo foi SO Windows 10 Home, 64 bits, Memória (RAM) 15,9 GB, Processador Intel® Corte™ i7-6500 CPU @ 2,50 GHz x 2, AMD Radeon™ T5 M330 (Sem suporte CUDA).

5.2 RESULTADOS E DISCUSSÕES

O primeiro aspecto discutido é o tempo de treinamento para um sistema de diagnóstico de falhas baseado em redes neurais convolucionais profundas. Sabe-se que os computadores com GPUs lidam melhor com a demanda computacional de soluções baseadas em aprendizado profundo que aqueles que apresentam apenas CPUs. Em contrapartida, os computadores com GPUs são mais caros, o que pode limitar a sua disponibilidade.

A Tabela 26 ilustra o problema do custo computacional. Ela mostra os tempos médios de treinamento de 30 CNNs, para cada cenário, com a arquitetura ilustrada na Figura 11. Cada cenário refere-se a uma configuração de computador utilizada. Todas as imagens utilizadas no treinamento dos modelos, que foi realizado por 50 épocas com o algoritmo *backpropagation*, foram do tipo RGB. Esta quantidade de modelos foi utilizada para garantir a relevância estatítica dos resultados. A configuração do computador com GPU, utilizada por Monteiro et al. (MONTEIRO et al., 2018), foi a seguinte: OS Ubuntu 16.04 LTS, 64 bits, Memória 15,6 GB, Processador Intel Xeon (R) CPU E 5-2609 v3 @ 1,90 GHz x 12, Graphics Galium 0.4 on NV117. A do computador com CPU foi apresentada na seção 5.1.4.

Tabela 26 – Tempo de treinamento médio de redes neurais convolucionais com diferentes configurações de computadores.

Configuração	**Tempo médio de treinamento (s)**
Computador com CPU	10.601,02
Computador com GPU	815,13

Fonte: O autor (2019).

Pode-se observar na Tabela 26 que o processo de treinamento sem GPU foi quase 13 vezes mais longo em relação processo que utilizou a GPU. Por outro lado, em algumas situações, dependendo da quantidade de dados ou do tempo disponível, o uso de computadores sem GPU pode ser inviável.

Algumas ações podem ser tomadas para resolver este problema, como reduzir o número de épocas de treinamento ou mesmo simplificar a estrutura do modelo. Levando-se em consideração a redução no número de épocas de treinamento, como pode ser observado na Tabela 27, a redução do número de épocas de 50 para 10 épocas promoveu uma queda no tempo médio de treinamento de aproximadamente 78,7%. Em contrapartida, tal redução tem um custo para o desempenho. A acurácia média reduziu em torno de 1,8%. Este comportamento já era esperado, dado que os modelos tiveram menos iterações para aprender os padrões encontrados nos dados de treino. Os resultados listados na Tabela 27 foram obtidos com o treinamento de 30 CNNs em cada cenário.

Antes de propor modificações ao sistema de diagnóstico de falhas, é necessário identificar as maiores dificuldades do modelo. O modelo treinado por 10 épocas foi tomado como referência.

Tabela 27 – Acurácia e tempo de treinamento médios de redes neurais convolucionais treinadas por 50 e 10 épocas.

Número de épocas de treinamento	**Acurácia média (%)**	**Tempo médio de treinamento (s)**
50	96,64	10.601,02
10	94,84	2.263,97

Fonte: O autor (2019).

A Tabela 29 lista os valores médios e os desvios padrões de acurácia para 30 modelos. Pode-se observar que, para alguma algumas classes, *e.g.* P1 e P2, o sistema apresentou valores de acurácia mais elevados, *e.g.* próximos a 100%. Por outro lado, os modelos apresentaram um desempenho pior para classes como P6 e P7.

Tabela 28 – Acurácia média e desvio padrão de 30 modelos treinados durante 10 épocas.

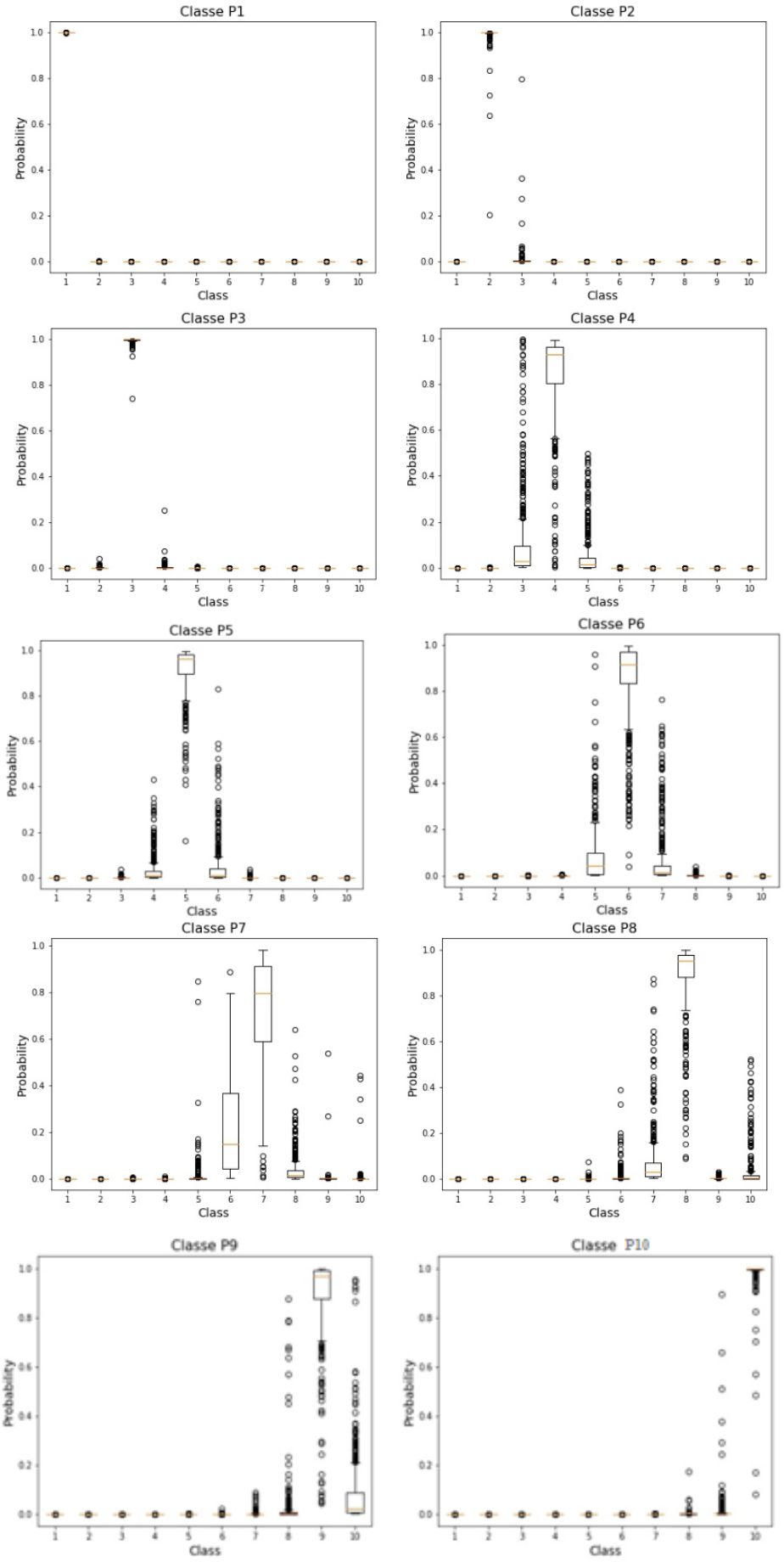
Classe	**Acurácia média** (%)	**Desvio padrão**
P1	100	0
P2	99,79	0,32
P3	98,30	6,20
P4	95,92	4,47
P5	94,32	7,66
P6	89,89	12,30
P7	83,79	13,03
P8	94,10	7,86
P9	96,50	4,60
P10	95,87	7,26

Fonte: O autor (2019).

Esta análise pode ser aprofundada pela observação das saídas do classificador. A Figura 29 mostra a distribuição das probabilidades das saídas dos modelos, de acordo com a classe do espectrograma de entrada. Pode-se observar que as probabilidades das saídas referentes às entradas da classe P1 foram próximas a 1 para a classe P1 e próximas a 0 para as demais classes. Isto ajuda a entender o porquê dos modelos treinados apresentarem uma acurácia média de 100% para esta classe. Por outro lado, os perfis de distribuição pertencentes às classes P6 e P7 mostram que as saídas das redes não apresentaram resultados tão precisos quanto o da classe P1. Desta forma, determinar a classe da imagem de entrada apenas escolhendo o maior valor de probabilidade pode levar a classificações incorretas devido à presença significativa de *outliers*.

Para superar esta questão, foi proposta uma solução baseada no uso das probabilidades de saída referentes às 10 classes para definição da classe correta. A solução foi implementada por meio da adição de um estágio de decisão baseado em aprendizado de máquina tradicional, *e.g.*, MLP e SVM, ao sistema de classificação original. Tais técnicas identificam a classe da falha por meio da informação contida nas probabilidades de saída da CNN. Por conseguinte, a resposta final do sistema é obtida por meio da análise de uma distribuição de probabilidades, e não de um único valor.

Figura 29 – Probabilidade de saída de modelos treinados por 10 épocas, para todas as 10 classes.



Fonte: O autor (2019).

A técnica de aprendizado de máquina adotada nesta etapa da pesquisa como estágio de decisão foi a máquina de vetores de suporte. Ela foi treinada de forma supervisionada, tendo como informação de entrada as saídas da rede neural convolucional referentes à base de treino. Os resultados da modificação proposta estão listados nas Tabelas 29 e 30. A Tabela 30 mostra os resultados para cada classe, enquanto a Tabela 29 mostra os resultados médios relacionados a todas as classes e modelos. 30 modelos foram treinados para cada cenário, com ou sem estágio de decisão.

Tabela 29 – Acurácia média e desvio padrão de 30 modelos treinados durante 10 épocas, com e sem estágio de decisão.

	Sem estágio de decisão		**Com estágio de decisão**	
Classe	**Acurácia média (%)**	**Desvio padrão**	**Acurácia média (%)**	**Desvio padrão**
P1	100	0	100	0
P2	99,79	0,32	99,79	0,32
P3	98,30	6,20	99,42	0,79
P4	95,92	4,47	97,74	1,71
P5	94,32	7,66	97,56	1,89
P6	89,89	12,30	94,36	4,29
P7	83,79	13,03	92,70	3,36
P8	94,10	7,86	96,71	1,84
P9	96,50	4,60	97,78	1,66
P10	95,87	7,26	97,96	1,51

Fonte: O autor (2019).

Tabela 30 – Acurácia média e tempo de treinamento médio de modelos que apresentam ou não um classificador adicional (o estágio de decisão).

Cenário	**Acurácia média (%)**	**Tempo médio de treinamento (s)**
Sem classificador	94,84	2.263
Com classificador	97,4	2.264

Fonte: O autor (2019).

Pode-se inferir a partir da Tabela 29 que a inclusão de um estágio de decisão propiciou melhorias no desempenho do sistema de diagnóstico de falhas para as 10 classes, tanto em termos de acurácia média quanto de desvio padrão. Ainda, observa-se na Tabela 30 que a acurácia média cresceu por volta de 2,56%, ao custo de um aumento de menos de 1 segundo no tempo médio de treinamento. Tais resultados tornam-se ainda mais significativos quando comparados aos obtidos com o treinamento de 50 épocas, sem estágio de decisão, vistos na Tabela 27. A acurácia média da configuração proposta foi apenas 0,76% superior em relação ao outro modelo, que foi treinado em computadores sem GPU, porém o tempo de treinamento do modelo com estágio de decisão foi 78,64% menor. Isto tornou o processo de treinamento significativamente mais rápido, sem comprometer a acurácia do sistema.

Além disso, duas outras métricas foram incluídas para atestar a confiabilidade dos resultados obtidos. Tais métricas foram o f-score e a AUC. Seus valores médios estão listados na Tabela 31, e ambos confirmaram a tendência de crescimento observada na Tabela 30.

Tabela 31 – F-score e AUC médios de modelos que apresentam ou não um classificador adicional (o estágio de decisão).

Cenário	F-score média	AUC média
Sem classificador	0,95	0,97
Com classificador	0,97	0,99

Fonte: O autor (2019).

Levando-se em consideração o tempo médio para realizar a classificação de uma única entrada, a adição de um estágio de decisão não promoveu mudanças significativas. O tempo de classificação médio, que foi por volta de 30 ms sem o estágio de decisão, aumentou menos que 1 ms.

Com o intuito de avaliar quão significativas foram as melhorias de desempenho propiciadas pela solução proposta, pode-se aplicar o teste estatístico de Wilcoxon às saídas do sistema de diagnóstico de falhas com e sem o estágio de decisão. Os resultados estão listados na Tabela 32. O teste de Wilcoxon é um teste de hipótese não paramétrico que pode ser utilizado para avaliar se duas determinadas distribuições são ou não equivalentes (BIJMA; JONKER; VAART, 2017). Se elas não forem equivalentes, temos que a melhoria atingida foi de fato significativa (representada por "Δ"). Caso contrário, a melhoria de desempenho não foi significativa (representada por - "). A Tabela 32 mostra que melhorias significativas foram atingidas para classes P4, P5, P6, P7 e P9. Apesar das classes remanescentes também terem demonstrado melhores resultados em relação às do modelo de comparação, as melhorias delas não foram estatisticamente relevantes.

Tabela 32 – Resultados do teste de Wilcoxon, comparando-se os modelos com e sem estágio de decisão.

Classe	**Resultado**
P1	-
P2	-
P3	-
P4	Δ
P5	Δ
P6	Δ
P7	Δ
P8	-
P9	Δ
P10	-

Fonte: O autor (2019).

O uso de uma rede neural do tipo MLP como estágio de decisão também foi avaliado nesta etapa do trabalho, para comparação. A entrada da rede neural teve dimensão igual a 10,

i.e., a CNN apresenta 10 saídas, a camada oculta contou com 21 neurônios. A sigmóide logística foi utilizada como função de ativação. O número de épocas de treinamento foi igual a 200. Os dados utilizados no treinamento, teste e validação dos modelos foram os mesmos do cenário com a SVM. 30 MLPs foram treinadas para garantir a relevância estatística dos resultados. Os resultados estão listados nas Tabelas 33 e 34. A Tabela 33 lista os valores médios do f-score, AUC e acurácia. A Tabela 34 lista dos valores médios dos tempos de treino e classificação.

Tabela 33 – F-score, AUC e acurácia médios de modelos que apresentam estágios de decisão baseados em MLPs e SVMs.

Cenário	**F-score média**	**AUC média**	**Acurácia Média** (%)
SVM	0,97	0,99	97,4
MLP	0,97	0,99	97,25

Fonte: O autor (2019).

Tabela 34 – Tempos de treinamento e de execução médios de modelos que apresentam estágios de decisão baseados em MLPs e SVMs.

Cenário	**Tempo de treinamento médio (s)**	**Tempo de execução médio**
SVM	0,33	0,00022
MLP	2,35	0,00001

Fonte: O autor (2019).

Com relação às métricas, apesar da pequena vantagem apresentada pela SVM em termos de acurácia, o teste de Wilcoxon sugeriu que os resultados obtidos por ambas as abordagens de estágios de decisão foram os mesmos. Em contrapartida, o tempo de treinamento da abordagem com a MLP foi aproximadamente 7,1 vezes mais longo que para o cenário com a SVM, enquanto o tempo de execução foi aproximadamente 20 vezes mais curto.

As análises realizadas com o estágio de decisão foram ampliadas ao cenário com as imagens em escala de cinza. O objetivo é analisar como o sistema de diagnóstico de falhas se comporta quando há uma redução na quantidade de informações disponíveis, *i.e.*, um único canal em vez de três, para a classificação. A Tabela 35 mostra a acurácia e o tempo de treinamento médios para 30 modelos pertencentes a cada cenário, *i.e.*, RGB e escala de cinza, durante 10 épocas. Observa-se que os modelos treinados com imagens RGB tiveram uma acurácia média aproximadamente 10% maior em relação aos modelos treinados com espectrogramas em escala de cinza. Em contrapartida, estes últimos precisaram de menos tempo para serem treinados, o que é provável que tenha acontecido por conta da quantidade menor de parâmetros treináveis dos modelos.

A Tabela 36 e a Figura 30 ajudam a entender melhor a diferença de desempenho entre os cenários analisados, considerando as 10 classes individualmente. Na Tabela 36 é possível observar que, mesmo nos casos nos quais a acurácia média permanece alta, *e.g.*, classe P1, existe uma queda de desempenho para os modelos treinados em escala de cinza, inclusive em relação ao desvio padrão. A Figura 30 mostra a distribuição das probabilidades de saída da CNN de

Tabela 35 – Acurácia média e tempo de treinamento médio de modelos treinados com espectrogramas RGB e em escala de cinza.

Cenário	Acurácia média (%)	Tempo médio de treinamento (s)
Escala de cinza	84,98	1.967
RGB	94,84	2.263

Fonte: O autor (2019).

acordo com as classes. O número de *outliers* para o cenário com espectrogramas em escala de cinza é significativamente superior ao cenário com imagens RGB. Tal comportamento ocorreu para todas as classes.

Tabela 36 – Acurácia média e desvio padrão de 30 modelos treinados durante 10 épocas, com imagens RGB e em escala de cinza.

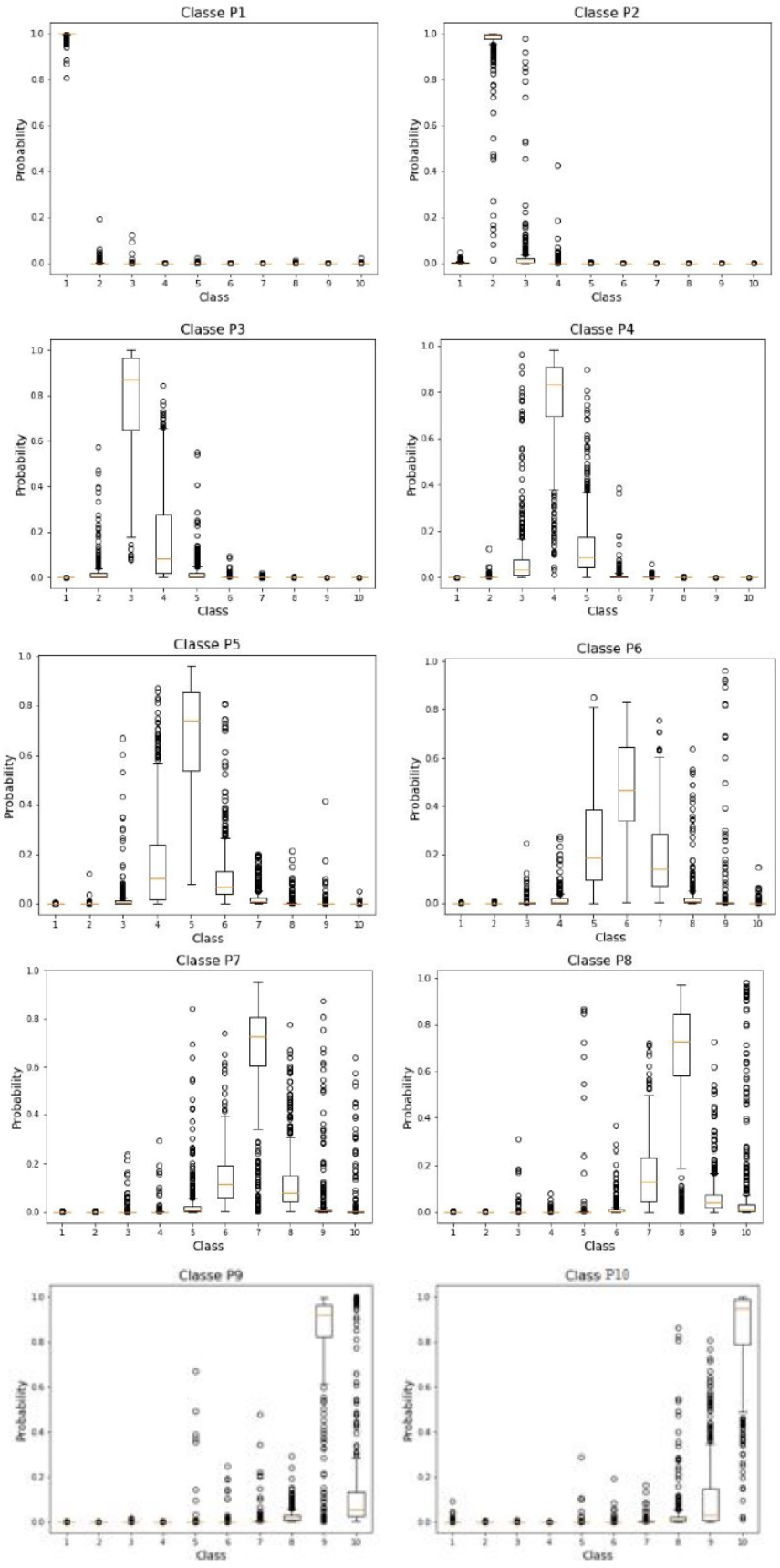
	RGB		Escala de cinza	
Classe	Acurácia média (%)	Desvio padrão	Acurácia média (%)	Desvio padrão
P1	100	0	99,98	0,07
P2	99,79	0,32	98,35	1,33
P3	98,30	6,20	93,04	6,69
P4	95,92	4,47	81,58	12,42
P5	94,32	7,66	75,50	15,05
P6	89,89	12,30	69,13	19,51
P7	83,79	13,03	74,30	15,03
P8	94,10	7,86	82,70	7,16
P9	96,50	4,60	85,47	8,51
P10	95,87	7,26	89,81	9,57

Fonte: O autor (2019).

Também aplicando um estágio de decisão definido pela máquina de vetores de suporte no cenário com imagens em escala de cinza, a solução proposta foi capaz de promover melhorias em todas as 10 classes. Os resultados estão exibidos nas Tabelas 37 e 38. A Tabela 37 mostra os resultados médios de acurácia e os desvios padrões de 30 modelos treinados, considerando as classes separadamente. Tais resultados referem-se aos cenários com e sem estágio de decisão. A Tabela 38 mostra os mesmos resultados médios de acurácia, mas tendo referência os modelos como um todo.

É possível inferir a partir da Tabela 37 que a inclusão de um estágio de decisão melhorou o desempenho do sistema de diagnose de falhas, considerando as 10 classes individualmente. Além disso, na Tabela 38 pode-se ver que a acurácia média aumentou aproximadamente 4,2%, ao custo de um aumento de menos de 1 segundo no tempo de treinamento. Esta melhoria relativa foi superior à atingida no cenário com os espectrogramas em RGB, também considerando os cenários com e sem estágio de decisão. Em contrapartida, mesmo após as melhorias promovidas pela inserção do estágio de decisão, o desempenho do sistema treinado com imagens em escala de cinza foi inferior ao do sistema treinado com espectrogramas em RGB, sem estágio de decisão, como visto na Tabela 35.

Figura 30 – Probabilidade de saída de modelos treinados por 10 épocas, para todas as 10 classes, com espectrogramas em escala de cinza.



Fonte: O autor (2019).

Tabela 37 – Acurácia média e desvio padrão de 30 modelos treinados durante 10 épocas, com imagens em escala de cinza, com e sem estágio de decisão.

	Sem classificador adicional		**Com classificador adicional**	
Classe	**Acurácia média (%)**	**Desvio Padrão**	**Acurácia média (%)**	**Desvio Padrão**
P1	99,98	0,07	99,99	0,06
P2	98,35	1,33	99,59	0,65
P3	93,04	6,96	94,73	1,78
P4	81,58	12,42	86,64	3,82
P5	75,50	15,05	81,26	3,54
P6	69,13	19,51	79,60	4,14
P7	74,30	15,03	81,61	2,47
P8	82,70	7,16	86,16	1,83
P9	85,47	8,51	88,46	1,41
P10	89,81	9,57	94,70	2,89

Fonte: O autor (2019).

Tabela 38 – Acurácia média e tempo de treinamento médio de modelos treinados com espectrogramas em escala de cinza, com e sem estágio de decisão.

Cenário	**Acurácia média (%)**	**Tempo médio de treinamento (s)**
Sem classificador adicional	84,98	1.967
Com classificador adicional	89,16	1.968

Fonte: O autor (2019).

A Tabela 39 mostra o f-score e a AUC médios para os sistemas com e sem estágio de decisão. Tais métricas reforçam a tendência de melhoria causada pela adição do estágio de decisão.

Tabela 39 – F-score e AUC de modelos que apresentam ou não classificadores adicionados como estágios de decisão.

Cenário	**F-score média**	**AUC média**
Com classificador	0,89	0,94
Sem classificador	0,85	0,92

Fonte: O autor (2019).

No que concerne o tempo de classificação médio para este cenário, a adição de um estágio de decisão não resultou em piora de desempenho significativa. Ele aumentou de 0,022 a 0,023 segundos.

Para avaliar se o aumento de desempenho proporcionado pelo estágio de decisão foi ou não significativo para o cenário com os espectrogramas em escala de cinza, em termos de acurácia, o teste de Wilcoxon foi mais uma vez utilizado. Os resultados estão listados na Tabela 40, a qual mostra que as melhorias de acurácia foram significativas para as classes P4, P5, P6, P7, P8 e P10. Apesar das demais classes também terem apresentado melhorias, estas não foram significativas.

Tabela 40 – Resultados do teste de Wilcoxon, comparando-se os modelos com e sem estágio de decisão, para o cenário com imagens em escala de cinza.

Classe	**Resultado**
P1	-
P2	-
P3	-
P4	Δ
P5	Δ
P6	Δ
P7	Δ
P8	Δ
P9	-
P10	Δ

Fonte: O autor (2019).

5.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo, foi analisado o uso de um estágio de decisão para interpretar as saídas de um sistema de diagnose de anomalias baseado em redes neurais convolucionais. Tais saídas correspondem à probabilidade de que uma entrada pertença a cada uma das classes de um dado conjunto possível de classes. Desta forma, em vez de simplesmente escolher a classe com maior valor de probabilidade como resposta do sistema, analisa-se a distribuição das saídas da CNN para que o processo de diagnóstico seja mais confiável.

Os resultados mostraram que é possível melhorar a acurácia do sistema de classificação e reduzir em quase 80% o tempo de treinamento sem comprometer o tempo de execução, o qual aumentou aproximadamente 0,001 segundos. Esta melhoria é especialmente significativa para casos nos quais um *hardware* poderoso, *e.g.*, GPUs, não está disponível. Desta forma, um sistema de diagnóstico com um dado nível de acurácia pode ser obtido apenas utilizando uma fração do tempo que seria necessário para realizar o treinamento completo. Os resultados foram obtidos com o uso de uma SVM como estágio de decisão, o qual utiliza as saídas da CNN como informação de entrada. Resultados similares foram obtidos com o uso de uma MLP como estágio de decisão. Isto sugere que a solução proposta pode ser implementada com outros algoritmos, além da SVM.

O uso de espectrogramas em escala de cinza e RGB também foi analisado neste capítulo. Foi visto que, mesmo que a adição de um estágio de decisão tenha causado melhoria em ambos os cenários, tais melhorias ocorreram em diferentes magnitudes. Apesar dos sistemas treinados com imagens em escala de cinza terem apresentado o maior aumento de acurácia, o melhor desempenho absoluto foi atingido pelos modelos treinados com imagens RGB. Esta diferença pode ser explicada pela quantidade de informação disponível em cada tipo de imagem, como previamente discutido. Além disso, foi visto que a diferença nos tempos de classificação não foi significativa, em termos de valores absolutos. Isto sugere que o uso de imagens RGB não

comprometeria a operação do sistema em aplicações em tempo-real.

Como em capítulos anteriores, o problema de detecção de anomalias torna-se mais complexo quando não se tem informações disponíveis sobre as classes anômalas. Neste e em capítulos anteriores, foram feitas análises voltadas para técnicas de detecção de anomalias baseadas classificadores com múltiplas classes. Desta forma, foi analisada a influência de diferentes parâmetros, *e.g.* tipo de imagem e profundidade da arquitetura, no nível de desempenho dos modelos treinados. Dado o conhecimento adquirido por meio destes estudos, os próximos capítulos tratarão do problema de detecção de anomalias utilizando apenas uma das classes para treinar o modelo.

6 USO DE REDES NEURAIS CONVOLUCIONAIS NA EXTRAÇÃO DE CARACTERÍSTICAS PARA A DETECÇÃO DE ANOMALIAS

Este capítulo dá prosseguimento aos estudos desenvolvidos nos capítulos 4 e 5. Sabe-se que a carência de informações sobre casos anômalos é algo comum na detecção de anomalias. Isto pode ocorrer por diversos motivos, *e.g.*, não se conhece todos os módulos de falhas possíveis ou a obtenção destes é inviável, como no caso de uma máquina nova. Sabe-se que, apesar das melhorias proporcionadas pelo uso de aprendizado profundo em diversas áreas, como a classificação de múltiplas classes e a predição de séries temporais, seu uso é menos difundido entre aplicações relacionadas à detecção de anomalias. Dentre os principais obstáculos, está a dificuldade de se aprender padrões discriminativos quando a única informação disponível pertence a uma das classes, *i.e.*, a classe normal, ou quando as classes de dados são altamente desbalanceadas (CHALAPATHY; CHAWLA, 2019).

Diante deste cenário, propõe-se no capítulo uma nova abordagem de detecção de anomalias baseada em aprendizado profundo. Ela consiste de um sistema híbrido que combina a extração implícita de características, realizada por um modelo de aprendizado profundo, com um classificador de uma classe, baseado em aprendizado de máquina tradicional. Este tipo de combinação é bastante presente na literatura (TIAN; LI; LIU, 2019; SINGH; MOHAN, 2018; VARTOUNI; TESHNEHLAB; KASHI, 2019; PEREIRA; SILVEIRA, 2019).

A contribuição do modelo proposto neste capítulo está no extrator de características, que é uma rede neural convolucional treinada como regressor para aprender a reproduzir uma distribuição predeterminada e escolhida de forma aleatória. O treinamento do extrator é feito de forma supervisionada, de modo que ele aprenda o mapeamento entre os espectrogramas pertencentes à classe normal (entrada) e a distribuição predeterminada (saída). Desta forma, espera-se que os espectrogramas pertencentes à classe normal estejam associados a características extraídas com padrões semelhantes aos das distribuições utilizadas para treinar a rede neural. Por outro lado, entradas anômalas devem resultar em saídas que sejam diferentes daquelas esperadas. A detecção de anomalias propriamente dita é realizada neste estudo pela OC-SVM. Esta é uma técnica tradicional de aprendizado de máquina dedicada à problemas de classificação de uma classe. A detecção é feita por meio da análise das características extraídas pela CNN, *i.e.*, o extrator de características.

6.1 METODOLOGIA

Nesta seção, apresenta-se detalhes sobre os experimentos realizados ao longo estudo desenvolvido no capítulo.

6.1.1 Base de dados

A base de dados utilizada foi a apresentada na seção 4.1.1

6.1.2 Organização da base de dados

A base de dados foi tratada de forma semelhante à apresentada nos Capítulos 4 e 5, sendo ela é composta por espectrogramas representados em escala de cinza. No entanto, a sua organização para o problema de detecção de anomalias se deu de forma diferente. Ela foi dividida em duas classes: a normal e a anômala. A classe normal é composta apenas por dados pertencentes à classe P1 da divisão original da base de dados, descrita na Tabela 16 e na Figura 19. Os sinais pertencentes a esta classe estão relacionados ao cenário onde as engrenagens não apresentam danos. A classe anômala, em contrapartida, é composta por dados pertencentes às demais classes da divisão original, isto é, da classe P2 à P10.

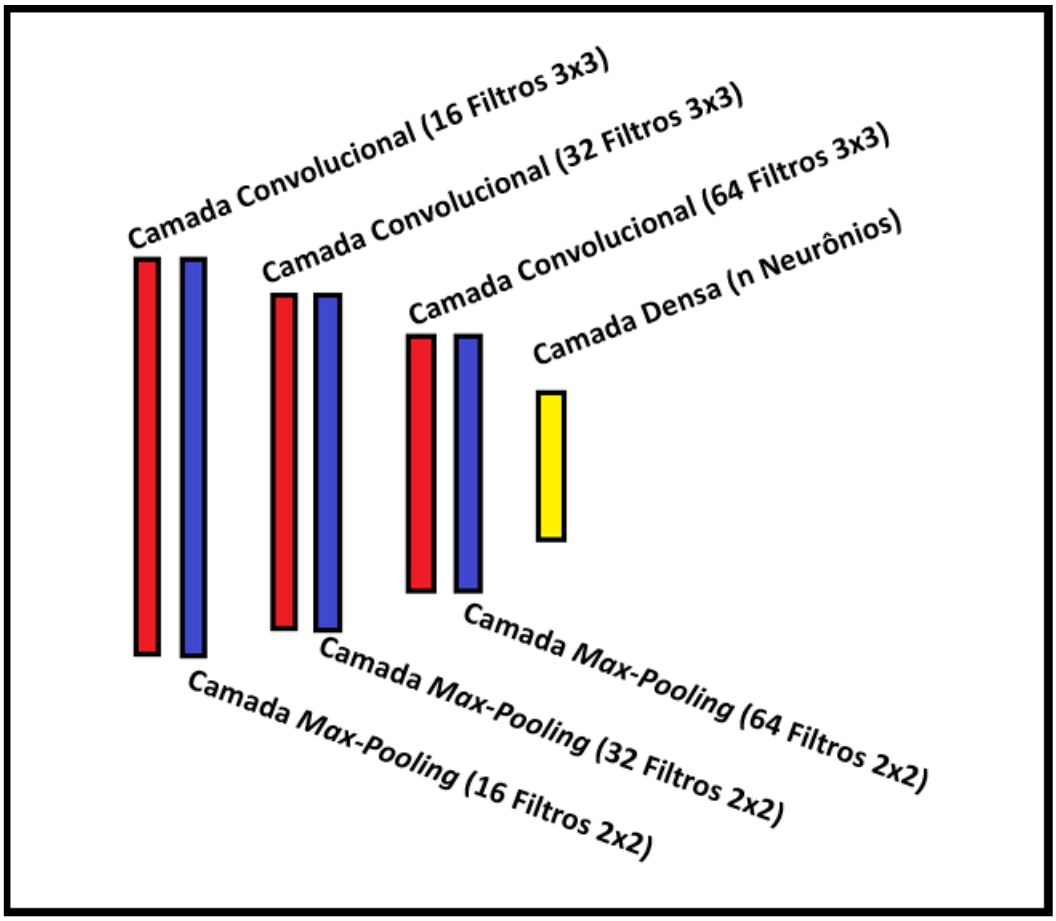
6.1.3 Modelo

A técnica de extração de características proposta é modelada como uma rede neural convolucional. Ela consiste de três pares de camadas convolucionais e de *max-pooling*, distribuídas de forma alternada, seguidas por uma camada de saída densamente conectada. Foi adotada uma configuração bidimensional para as camadas convolucionais e de *max-pooling*, dado o tipo de informação de entrada que foi adotada, *e.g.* "imagens"ou matrizes de 128 x 128 *pixels*. Os passos destas camadas foram 1 e 2, respectivamente, e a ReLU foi função de ativação adotada para os neurônios. A arquitetura do modelo foi inspirada em (MONTEIRO et al., 2018; MONTEIRO; BASTOS-FILHO, 2019b), ilustrada na Figura 31.

Espera-se que o modelo trabalhe como um regressor, *i.e.*, ele é treinado para se adaptar a um conjunto de valores contínuos, e não a classes. Estes valores contínuos a serem aprendidos foram gerados de forma aleatória, segundo uma distribuição uniforme de probabilidades. O objetivo deste estudo não foi encontrar um conjunto ótimo destes "valores contínuos"para o problema, mas mostrar que o treinamento de redes neurais, *i.e.*, o extrator de características, com o intuito de aprender um padrão predefinido pode melhorar o processo de detecção de anomalias. Por isto que estes valores foram gerados de forma aleatória. Os valores-alvo as são características utilizadas pela OC-SVM para realizar a detecção de anomalias. Os algoritmos utilizados para treinar o modelo, de forma supervisionada, foram o *backpropagation* e o otimizador Adam. O modelo foi treinado por 50 épocas, e o tamanho do *batch* foi 144.

As funções de ativação empregadas nas camadas convolucionais são ReLUs, dado que elas têm apresentado bons resultados em diversas aplicações envolvendo redes neurais convolucionais (RAWAT; WANG, 2017). A ReLU também foi utilizada como função de ativação da camada de saída, desde que a solução proposta não requer que a saída do modelo tenha um limite superior, ou mesmo que represente algum valor de probabilidade. A função de perda adotada foi o erro quadrático médio, desde que este é um problema de regressão.

Figura 31 – Configuração do modelo de extração de características proposto.



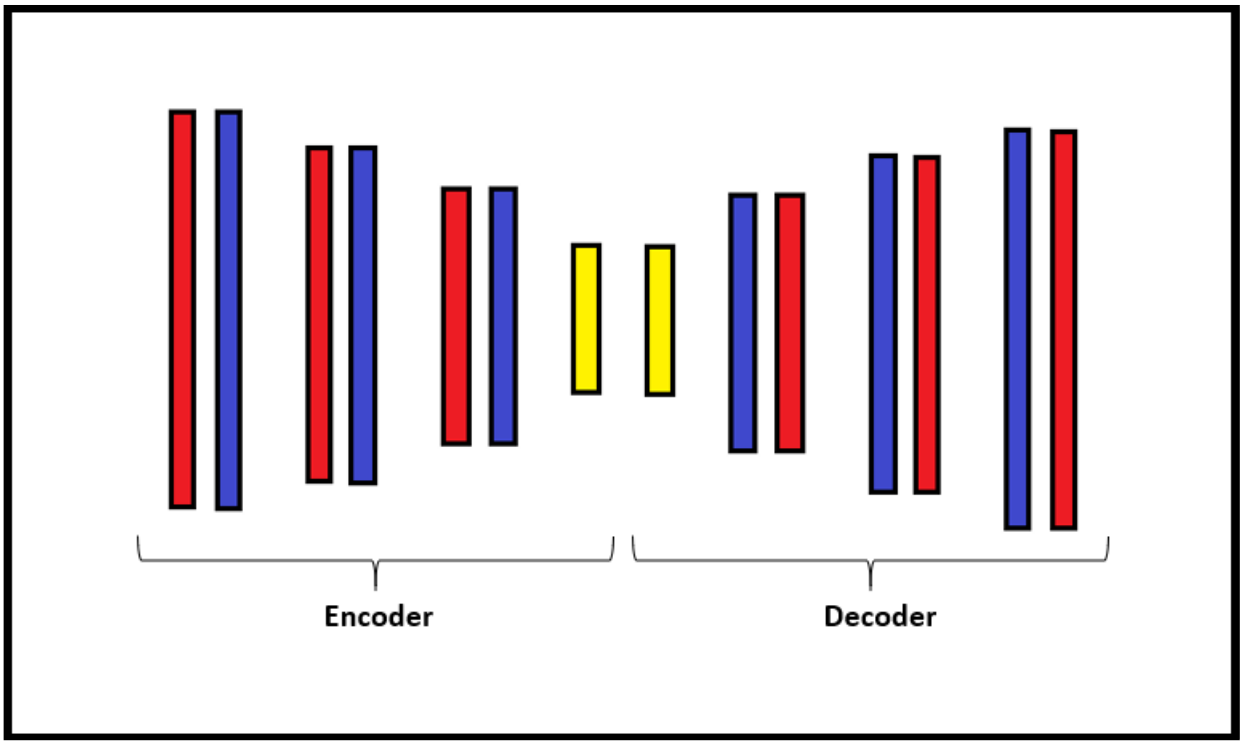
Fonte: O autor (2019).

6.1.4 Arranjo Experimental

A solução proposta foi comparada a duas técnicas de detecção de anomalias comumente utilizadas em trabalhos do estado da arte. A primeira técnica usa *autoencoders* convolucionais para reconstruir os dados de entrada, de modo que seja possível definir um limite baseado no erro de reconstrução e utilizá-lo na detecção das anomalias. Como os dados de entrada são espectrogramas, os filtros das camadas convolucionais destes *autoencoders* apresentaram dimensões 3 x 3, enquanto aqueles das camadas de *max-pooling* apresentaram dimensões 2 x 2. A arquitetura dos autoencoders é apresentada na Figura 32. O *enconder* apresenta a mesma estrutura ilustrada na Figura 31, enquanto o *decoder* é simétrico ao *encoder* em relação à camada densa.

A segunda técnica empregou os *encoders* dos *autoencoders* convolucionais como extratores de características. Eles trabalharam em conjunto com uma técnica tradicional de aprendizado de máquina, a OC-SVM, que realiza a detecção de anomalias com as características extraídas. Não foi considerada neste capítulo a comparação com uma abordagem de detecção de anomalias baseada em classificadores de múltiplas classes, como em (MONTEIRO et al., 2018), desde que ela supõe que se tenha informações prévias sobre as classes anormais. Neste estudo, pretende-se

Figura 32 – Configuração do *autoencoder*.



Fonte: O autor (2019).

utilizar apenas as informações fornecidas pela classe normal para treinar os modelos de detecção de anomalias. Os espectrogramas de classes anômalas são usados apenas para avaliar o modelo.

O conjunto de dados usado para treinar esses modelos consiste em 1440 espectrogramas pertencentes à classe normal. O de teste, por outro lado, contém 2160 espectrogramas, nos quais 360 pertencem à classe normal e os demais estão divididos entre as classes relacionadas às falhas, *i.e.*, 200 espectrogramas por classe (de P2 a P10). Esses espectrogramas de falhas foram selecionados aleatoriamente do conjunto contendo todos os espectrogramas (são 16.200 no total). Foi utilizado este número de espectrogramas anômalos para não causar distorções nas métricas de avaliação, uma vez que o número de instâncias anômalas é muito maior que o número de amostras normais. 15 modelos foram treinados por 50 épocas em cada cenário. O processo de treinamento foi realizado no *Google Colaboratory* (CARNEIRO et al., 2018). Todos os *scripts* foram escritos em linguagem de programação *Python* (ROSSUM; DRAKE, 2011).

6.2 RESULTADOS E DISCUSSÕES

A primeira análise compreende a detecção de anomalias baseada no erro de reconstrução. Foi utilizada a configuração de modelo descrita na seção 6.1, Figura 32, para a construção dos *autoencoders*. Quatro tamanhos diferentes de camadas densas (n) foram considerados: 32, 64, 96 e 128 neurônios. O limiar adotado para a distinção de amostras normais e anômalas foi o 95º percentil da distribuição dos erros de reconstrução, que foi obtida a partir dos dados de treino. Cada modelo treinado tem o seu próprio limiar. O valores médios das métricas de

classificação obtidos a partir de 60 modelos treinados, *i.e.* 15 para cada tamanho de camada densa, estão listados na Tabela 41. Cinco métricas foram consideradas: acurácia, precisão, recall, f-score e AUC. As quatro primeiras são definidas pelas equações (6.1), (6.2), (6.3) e (6.4), respectivamente. A AUC, por outro lado, é a área sob a curva ROC, obtida pela relação entre as taxas de positivos verdadeiros e positivos falsos, para vários valores do limiar de classificação. Nesta tabela, n representa o número de neurônios na camada densa. Tais resultados referem-se apenas às amostras de teste.

$$acurácia = \frac{verdadeiros\ positivos + verdadeiro\ negativos}{total} \tag{6.1}$$

$$precisão = \frac{verdadeiros\ positivos}{verdadeiros\ positivos + falsos\ positivos} \tag{6.2}$$

$$recall = \frac{verdadeiros\ positivos}{verdadeiros\ positivos + falsos\ negativos} \tag{6.3}$$

$$f - score = \frac{2 * precião * recall}{precião + recall} \tag{6.4}$$

Tabela 41 – Valores médios das métricas de classificação para a detecção de anomalias baseadas no erro de reconstrução.

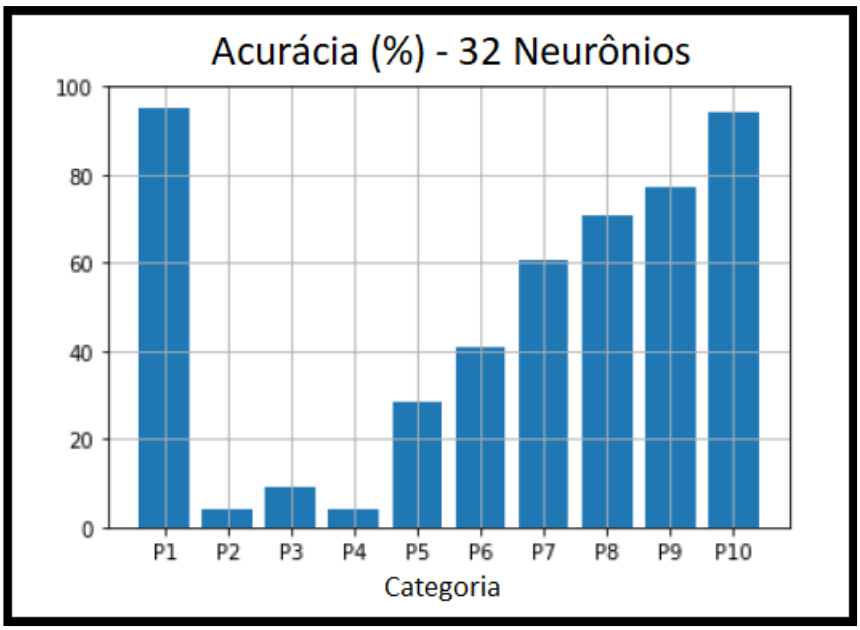
n	Acurácia	Precisão	*Recall*	F-Score	AUC
32	0,518	0,251	0,950	0,614	0,614
64	0,488	0,239	0,947	0,607	0,607
96	0,477	0,236	0,945	0,604	0,604
128	0,451	0,227	0,941	0,597	0,597

Fonte: O autor (2019).

Foi observado que o melhor desempenho foi atingido por *autoencoders* com 32 neurônios na camada densa. No entanto, este não foi um bom desempenho, *i.e.*, pode-se ver que os modelo classificaram corretamente pouco mais da metade das amostras, de acordo com a acurácia. Os valores de precisão e *recall* podem ajudar no entendimento deste cenário. O alto valor de *recall* mostra que o modelo apresenta bons resultados quando se trata de classificar corretamente as amostras pertencentes à classe normal. Por outro lado, a precisão baixa reflete a alta ocorrência de falsos positivos em relação ao número de positivos verdadeiros, sugerindo que muitas amostras anômalas foram erroneamente classificadas como normais pelo modelo.

A Figura 33 mostra a acurácia média de *autoencoders* com 32 neurônios na camada densa, de acordo com cada classe individualmente. Apesar da detecção de anomalias visar preferencialmente identificar amostras como normais ou anômalas, os níveis de dano são considerados nesta análise apenas para entender melhor as características de cada abordagem.

Figura 33 – Acurácia média de detectores de anomalias baseados no erro de reconstrução, considerando 32 neurônios na camada densa, a classe normal (P1) e todas as classes com danos (P2 a P10).



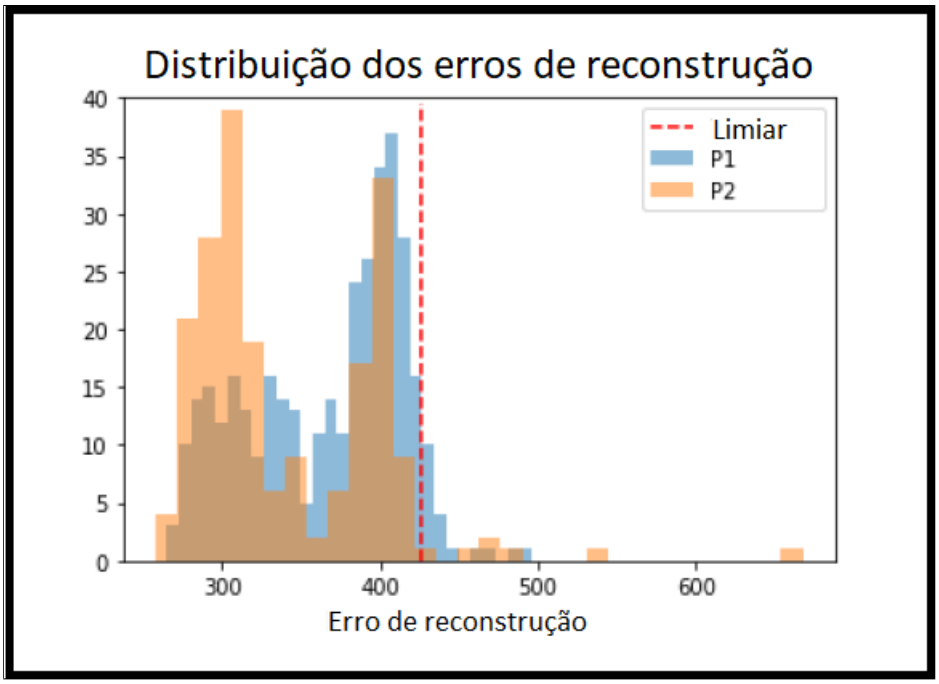
Fonte: O autor (2019).

De acordo com a Figura 33, a acurácia tende a aumentar à medida que o nível de dano aumenta, considerando as classes anômalas. Isto sugere que danos menores nas engrenagens não resultam em mudanças significativas nos padrões da reconstrução dos espectrogramas anômalos. Desta forma, a detecção de anomalias baseada no erro de reconstrução apresentou o pior resultado nos cenários com danos baixos. Por outro lado, danos mais severos resultaram em maiores taxas de acerto. As Figuras 34 e 35 ajudaram no entendimento deste problema. Elas mostram os histogramas dos erros de reconstrução considerando a classe normal (P1) , e as classes anômalas P2 e P10, *i.e.* os menores e maiores níveis de danos, respectivamente. É possível observar distribuições que se sobrepõem na Figura 34, tornando difícil definir dados como normais ou anormais apenas pela definição de um limiar. A Figura 35 mostra que este problema é menor para a classe P10.

A segunda análise levou em consideração o uso de *encoders* como extratores de características. Estas características são utilizadas pelo OC-SVM para realizar a detecção de anomalias propriamente dita. Nesta etapa foram utilizados os *encoders* dos *autoencoders* empregados na análise anterior. Os resultados estão listados na Tabela 42. O OC-SVM foi treinado para classificar 95% dos dados de treinamento como normais e 5% como anômalos.

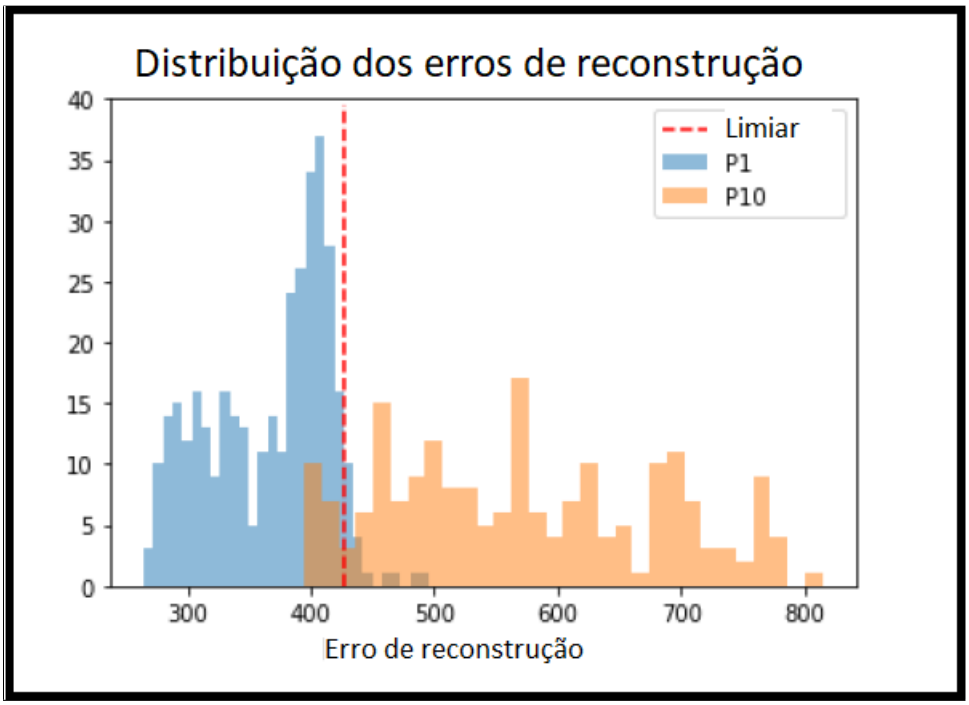
A combinação analisada nesta etapa apresentou resultados significativamente melhores em relação à baseada no erro de reconstrução. Como exemplo, a acurácia média do detector de anomalias híbrido para n, *i.e.*, número de neurônios na camada densa, igual a 32 foi 63% maior em relação a do detector baseado em *autoencoders*. Por outro lado, os melhores resultados da abordagem híbrida foram obtidos por *encoders* com 128 neurônios nas camadas densas. Isto

Figura 34 – Distribuição dos erros de reconstrução para as classes P1 (normal) e P2 (anomalia).



Fonte: O autor (2019).

Figura 35 – Distribuição dos erros de reconstrução para as classes P1 (normal) e P10 (anomalia).



Fonte: O autor (2019).

sugere que uma maior quantidade de informações, *e.g.* 128 elementos, melhoraram a capacidade do OC-SVM de definir uma fronteira de decisões adequada. Também vale ressaltar que o cenário com 96 neurônios na camada densa apresentou o melhor valor médio de *recall*, porém as demais métricas foram inferiores às referentes ao modelo com n igual a 128. Isto significa que os

Tabela 42 – Valores médios das métricas de classificação para a detecção de anomalias baseadas em *encoders* + OC-SVM.

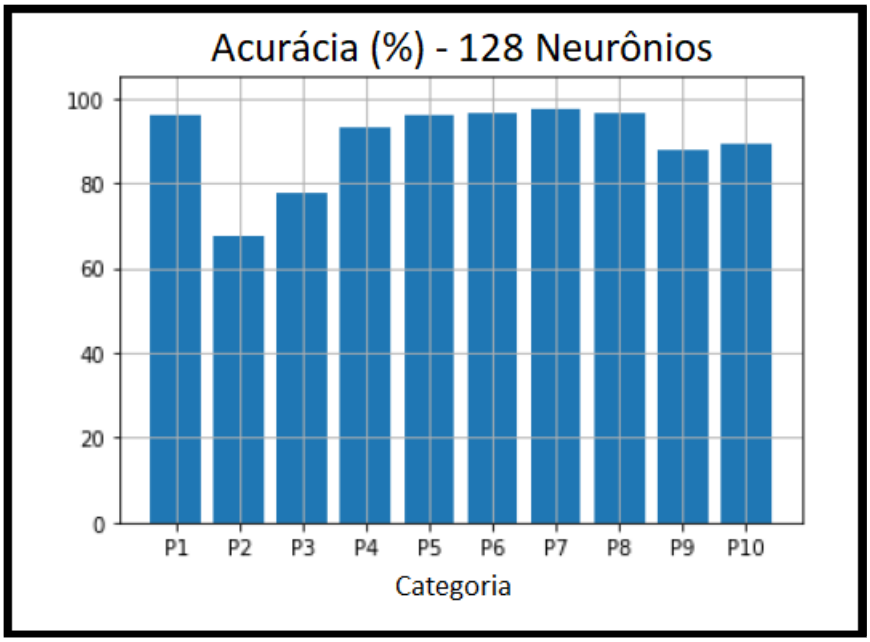
n	Acurácia	Precisão	*Recall*	F-Score	AUC
32	0,844	0,558	0,949	0,691	0,773
64	0,865	0,580	0,954	0,715	0,785
96	0,851	0,598	0,961	0,723	0,794
128	0,902	0,656	0,959	0,773	0,823

Fonte: O autor (2019).

sistemas com n igual a 96 apresentam menos falsos negativos, *i.e.*, menores taxas de alarmes falsos, mas a sua capacidade de detectar anomalias não é tão boa quanto a dos sistemas com 128 neurônios na camada densa.

A Figura 36 mostra o mesmo tipo de informação apresentado na Figura 33, *i.e.*, a acurácia média considerando cada classe individualmente. Como pode ser observado, a principal melhoria alcançada pelo uso da combinação de *encoders* e OC-SVM é o aumento da acurácia referente às classes relacionadas aos menores danos. Isto significa que uma maior quantidade de informação disponível, ou seja, 128 características ao invés de apenas o erro de reconstrução, permite ao sistema identificar diferenças mais relevantes entre classes mais próximas como P1 e P2, por exemplo. Esta informação ajuda a entender as vantagens da solução proposta neste capítulo em relação à baseada no erro de reconstrução.

Figura 36 – Acurácia média de detectores de anomalias híbridos baseados em *encoders* e OC-SVMs, considerando 128 neurônios na camada de saída do *encoder*, para a classe normal (P1) e todas as dez classes anômalas (P2 a P10).



Fonte: O autor (2019).

A última análise diz respeito à solução proposta neste capítulo, isto é, um extrator de características treinado para aprender um padrão predefinido e aleatoriamente escolhido.

A configuração utilizada foi a ilustrada na Figura 31, com quatro tamanhos diferentes de camadas de saída para o extrator de características: 32, 64, 96 e 128 neurônios. O extrator de características, baseado em CNN, foi treinado para aprender características com valores aleatórios uniformemente distribuídos no intervalo entre 0 e 10. Este valor de limite superior foi escolhido por ter apresentado o melhor resultado dentre os valores testados, *e.g.* 1; 2,5; 5 e 10. Não foram realizados testes com valores maiores de limite superior. Em contrapartida, por mostrar uma tendência de crescimento com o limite superior variando de 1 a 10, há razões para acreditar que o desempenho do sistema possa ser melhorado, porém isto será visto em estudos futuros. A grande faixa de valores de saída esperados para o extrator de características, *e.g.* até 10, é o que justifica a adoção da função de ativação *ReLU* também para a camada de saída do mesmo.

Os resultados médios de 60 modelos treinados (15 em cada cenário de tamanho da camada densa) estão listados na Tabela 43, na qual n significa o número de neurônios na camada densa. A OC-SVM também foi treinada para classificar 95% das amostras de treino como normal, assim como na análise anterior, e os resultados apresentados dizem respeito apenas às amostras de teste.

Tabela 43 – Valores médios das métricas de classificação para a detecção de anomalias baseadas na combinação CNN + OCSVM.

n	**Acurácia**	**Precisão**	***Recall***	**F-Score**	**AUC**
32	0,940	0,764	0,937	0,841	0,875
64	0,944	0,774	0,941	0,849	0,881
96	0,948	0,789	0,946	0,860	0,890
128	0,946	0,780	0,947	0,855	0,884

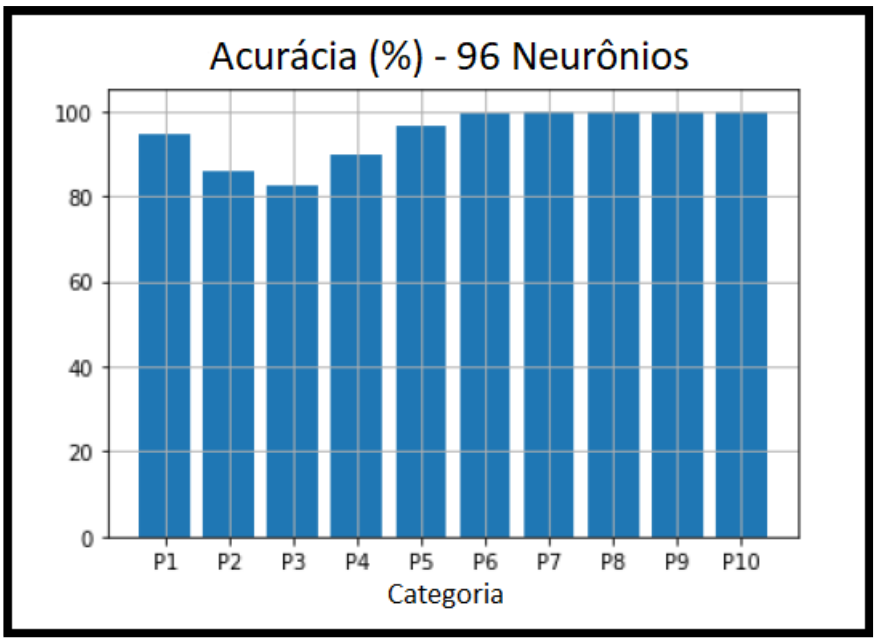
Fonte: O autor (2019).

A Tabela 43 mostra que o melhor desempenho foi atingido pelos modelos cujos extratores de características apresentaram saídas com 96 neurônios. No entanto, o cenário com 128 neurônios foi o que apresentou o melhor valor de *recall*. Isto sugere que modelos neste cenário, *i.e.*, 128 neurônios, apresentaram menores taxas de alarmes falsos. Por outro lado, a sua chance de classificar erroneamente exemplos anômalos é maior. Também foi observado que todos os cenários referentes aos diferentes valores de n apresentaram resultados bastante próximos.

A Figura 37 mostra a acurácia média de modelos com 96 neurônios na saída dos extratores de características, considerando cada classe da base de dados individualmente. Como nas abordagens vistas anteriormente, os melhores resultados são observados em classes relacionadas a danos maiores na engrenagem. Por outro lado, os resultados médios obtidos por cada classe foram superiores neste último cenário.

Além disso, os resultados obtidos pela combinação do extrator de características baseado em CNN com OC-SVM foram superiores àqueles obtidos com as demais soluções analisadas neste capítulo. Como exemplo, o pior resultado obtido com a solução proposta ainda foi 4,2%

Figura 37 – Acurácia média para detectores de anomalias baseados em CNNs e OC-SVM, considerando 96 neurônios na saída do extrator de características, para a classe normal (P1) e todas as classes anômalas (P2 a P10).



Fonte: O autor (2019).

superior à melhor acurácia obtida pela abordagem que compreende o uso do *encoder* com a OC-SVM. Comparando com a abordagem baseada no erro de reconstrução, essa porcentagem foi superior a 81%. Considerando o melhor resultado de acurácia obtido pela solução proposta, tais melhorias foram de 5,1% e 83%, respectivamente. A Tabela 44 mostra os melhores resultados obtidos pelas técnicas analisadas neste capítulo. Observa-se que a solução proposta apresentou o melhor desempenho em 4 das 5 métricas. A exceção foi o *recall*, sugerindo que o sistema proposto de detecção de anomalias apresenta melhor desempenho na identificação de anomalias, ao custo de uma maior taxa de alarmes falsos.

Tabela 44 – Valores médios das métricas de classificação para as técnicas analisadas neste capítulo.

Abordagem	**Acurácia**	**Precisão**	**Recall**	**F-Score**	**AUC**
Erro de reconstrução	0,518	0,251	0,950	0,397	0,614
Encoder + OCSVM	0,902	0,656	0,959	0,773	0,823
CNN + OCSVM	0,948	0,789	0,946	0,860	0,890

Fonte: O autor (2019).

Tais resultados mostram que o treinamento do extrator de características para aprender uma distribuição específica pode ser melhor que treinar modelos de forma não supervisionada, ou utilizar simplesmente o erro de reconstrução como parâmetro para identificar anomalias Treinando de forma não supervisionada, o modelo aprende as características mais gerais do conjunto de dados normais. Para a detecção de anomalias, por outro lado, tais características

podem ou não também estar presentes nas classes anômalas, prejudicando o desempenho do sistema.

Desta forma, treinando o modelo para aprender uma distribuição específica, pode-se criar relações mais específicas entre os dados de entrada e os de saída. Tais relações são construídas de tal forma que, apresentando-se ao modelo alguma entrada diferente dos dados utilizados durante o treinamento, as características obtidas na saída sejam diferentes daquelas esperadas, sendo possível que algoritmos como a OC-SVM use-as para identificar comportamentos normais ou anômalos.

6.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo, foi vista uma solução baseada em sistemas híbridos para a detecção de anomalias. A solução proposta combina um modelo de extração de características com um classificador de uma classe. A contribuição desta proposta está na extração de características, desempenhada por uma rede neural convolucional treinada como um regressor, cujo objetivo é aprender uma distribuição predefinida. As saídas do extrator alimentam uma máquina de vetores de suporte de uma classe, que realiza a detecção de anomalias propriamente dita.

O desempenho do sistema proposto foi avaliado com o uso de dados relacionados ao processo de diagnóstico de uma caixa de engrenagens. Os resultados da abordagem proposta foram comparados ao de técnicas comumente encontradas em trabalhos da literatura. Após algumas análises, foi observado que a técnica de detecção de anomalias proposta teve resultados melhores em relação às outras soluções analisadas. Acredita-se que a forma de treinamento do extrator de características permitiu que o mesmo criasse relações mais específicas entre os dados de entrada e de saída, em vez de apenas aprender atributos gerais do conjunto de treino, como treinamentos não supervisionados normalmente fazem.

Por outro lado, o treinamento do detector de anomalias e do extrator de características permaneceu sendo realizado sem relação direta com objetivos da detecção de anomalias. Diante disto, viu-se a necessidade de estabelecer uma rotina de treinamento que vise uma melhor separabilidade das representações de dados normais e anômalos durante o processo de aprendizado, o que será visto no capítulo seguinte.

7 USO DE REDES NEURAIS CONVOLUCIONAIS TREINADAS COM SELETORES DE PROTÓTIPOS NA EXTRAÇÃO DE CARACTERÍSTICAS PARA A DETECÇÃO DE ANOMALIAS

Este capítulo traz uma continuação do estudo desenvolvido no capítulo 6, que tratou do uso de redes neurais convolucionais para extrair características do conjunto de dados. No capítulo 6, as redes neurais foram treinadas para mapear espectrogramas a distribuições de valores predefinidas, porém geradas aleatoriamente. Tais espectrogramas eram relacionados à condição normal de operação do dispositivo, sendo as características extraídas utilizadas por uma abordagem tradicional de classificador de uma classe, o OC-SVM, para detectar anomalias.

Desta vez, propõe-se um método para otimizar o processo de extração de características pela rede neural convolucional. O objetivo é agrupar características extraídas de instâncias pertencentes à mesma classe que foi utilizada para treinar o extrator de características, *i.e.*, a classe normal. Enquanto isso, espera-se que instâncias oriundas de classes diferentes, *i.e.*, classes ditas anômalas, sejam mapeadas para regiões diferentes do espaço de características.

Tal método consiste em utilizar um algoritmo de seleção de protótipos para aprimorar o aprendizado de um extrator de características inicializado aleatoriamente. O processo é iterativo e utiliza dados pertencentes a uma única classe, que é a classe normal. O extrator de características é alimentado com lotes de espectrogramas, obtendo-se suas respectivas saídas, *i.e.*, representações no espaço características. Os dados de entrada, juntamente com suas representações no espaço de características, são utilizados para treinar a rede neural para mapear as entradas fornecidas em uma região mais restrita do espaço de características. Este processo é repetido por um dado número de iterações.

Desde que instâncias anômalas não são utilizadas durante o processo de treinamento do extrator de características, elas tendem a ser mapeadas para outras regiões do espaço de características, tendo-se como referência as amostras normais. Isto possibilita o uso de classificadores de uma classe baseados em aprendizado de máquina, *e.g.*, OC-SVM, para realizar a classificação de instâncias em normais ou anômalas, de acordo com a região do espaço de características em que as mesmas se encontram.

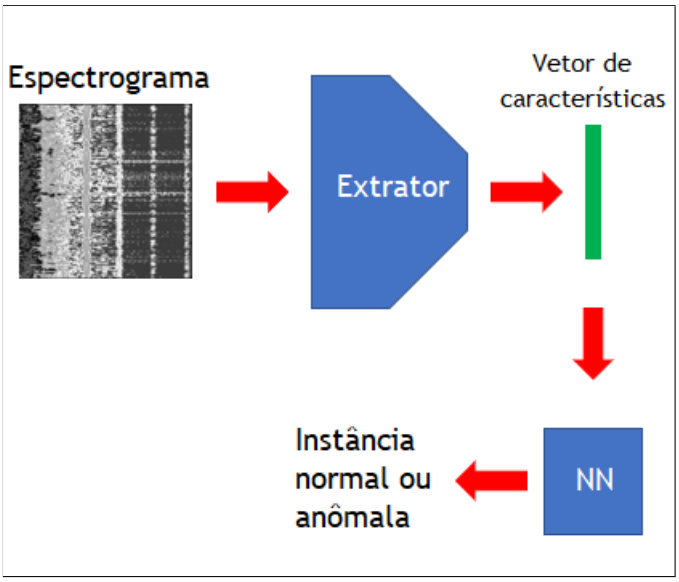
7.1 METODOLOGIA

Nesta seção, apresenta-se detalhes sobre os experimentos realizados ao longo estudo desenvolvido no capítulo.

7.1.1 O detector de anomalias

Conforme mencionado no texto introdutório deste capítulo, propõe-se um novo método para obtenção de um extrator de características baseado em aprendizado profundo. O extrator é parte de um sistema híbrido de detecção de anomalias em equipamentos industriais. Este sistema, por usa vez, é composto por dois estágios. O primeiro consiste no extrator de características, que é responsável por condensar em vetores de características a informação fornecida ao sistema. As entradas do sistema são espectrogramas obtidos a partir de sinais de vibração coletados por acelerômetros posicionados sobre o equipamento em questão. O segundo estágio é um classificador de uma classe baseado no algoritmo dos vizinhos mais próximos (NN, do inglês *nearest neighbors*). Ele classifica um vetor de características como pertencente a uma instância normal ou anômala de acordo com a distância entre este vetor e um conjunto de vetores pertencentes às instâncias de treino, *i.e.*, pertencentes à classe normal. Para uma dada instância de teste, se a distância média aos vetores de treino for superior a um determinado limiar, ela é rotulada como anômala. Caso contrário, ela é classificada como normal. A Figura 38 ilustra o detector de anomalias completo. A Tabela 45 detalha as camadas do extrator de características.

Figura 38 – Detector de anomalias completo.



Fonte: O autor (2021).

Nessa estrutura, as camadas convolucionais e de *max-pooling* são responsáveis por identificar padrões específicos na informação de entrada, bem como reduzir a sua dimensionalidade e destacar as ativações importantes na saída. As camadas de *batch normalization* aumentam a estabilidade e a velocidade do processo de treinamento. A camada de linearização reorganiza a informação na forma de um vetor. A camada de *dropout* realiza a regularização, melhorando a distribuição da informação ao longo dos neurônios da camada densa, que fornece os vetores de características em sua saída.

Através de experimentos preliminares, foi observado que os detectores de anomalias de

Tabela 45 – Camadas do Extrator de Características

Índice	**Camada**	**Detallhes**
1	Convolucional	8 Filtros 7x7
2	*Batch Normalization*	Momento 0.9
3	*Max Pooling*	Janela 2x2
4	Convolucional	16 Filtros 3x3
5	*Batch Normalization*	Momento 0.9
6	*Max Pooling*	Janela 2x2
7	Convolucional	32 Filtros 3x3
8	*Batch Normalization*	Momento 0.9
9	*Max Pooling*	Janela 2x2
10	Convolucional	64 Filtros 3x3
11	*Batch Normalization*	Momento 0.9
12	*Max Pooling*	Janela 2x2
13	Linearização	-
14	*Dropout*	Taxa de *Dropout* 0.2
15	Densa	128 neurônios

Fonte: O autor (2021).

melhor desempenho estavam associados a vetores com uma maior dispersão de características. O uso da função de ativação linear para as camadas convolucionais e densas contribuiu positivamente para este comportamento, dado que ela não apresenta regiões de saturação típicas de funções como a sigmóide, tangente hiperbólica ou ReLU. Estas regiões limitam a magnitude da saída das camadas acima e/ou abaixo de um determinado valor, reduzindo a faixa de valores possíveis para a saída de cada neurônio. Por conta da saturação, valores diferentes recebidos pela função de ativação podem ser mapeados para uma mesma região do espaço de características, *e.g.*, todos as entradas negativas da função ReLU são mapeadas para 0. Este comportamento pode ser um problema para classificadores treinados com dados pertencentes a uma única classe, *i.e.*, a classe normal, dado que não existem instâncias anômalas para ajustar o mapeamento das características de modo a facilitar a discriminação das classes.

A função de perda é um erro quadrático médio (MSE, do inglês mean squared error) (KAY, 1993) adaptado. Esta função é apresentada na Equação (7.1) e consiste em dividir o MSE original pela média da soma dos quadrados dos elementos dos vetores de características. Com esta adaptação, pretende-se evitar que as todas saídas do extrator convirjam para zero durante o processo de treinamento, o que seria inútil à aplicação de detecção de anomalias.

$$Loss = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \widehat{Y}_i\right)^2}{\frac{1}{n}\sum_{i=1}^{n}\widehat{Y}_i^{\,2}} \tag{7.1}$$

na qual n é a quantidade de elementos do vetor de características, Y_i é o valor real de uma característica de índice i, e $\widehat{Y}_i$ é o valor da predição do modelo para esta característica.

7.1.2 Treinamento do detector de anomalias

A primeira etapa do processo de treinamento consiste em refinar iterativamente o mapeamento entre a entrada e a saída de um extrator de características inicializado aleatoriamente com o algoritmo *Glorot* (GLOROT; BENGIO, 2010). Esta forma de inicialização resulta numa maior uniformidade das ativações ao longo das camadas da rede neural, *i.e.*, pois os valores possíveis para os pesos da rede variam de acordo com o número de entradas e saídas dos tensores de pesos. Isto permite uma maior penetração dos sinais de entrada em camadas mais profundas da rede, ajudando a evitar problemas como o aumento ou a redução excessiva do gradiente durante o processo de treinamento. Os pesos são inicializados aleatoriamente de acordo com uma distribuição uniforme de limites $[Limite, -Limite]$, sendo o valor $Limite$ definido pela Equação 7.2. A atualização dos pesos foi feita com o otimizador *Adam*, com uma taxa de aprendizado inicial de 0.001. O algoritmo de treinamento foi o *backpropagation*.

$$Limite = \sqrt{\frac{6}{fan_{in} + fan_{out}}} \tag{7.2}$$

na qual fan_{in} e fan_{out} são os números de unidades de entrada e saída, respectivamente.

O algoritmo de vizinhos mais próximos também é utilizado nesta etapa como um seletor de protótipos. A sua escolha se deve à natureza do algoritmo que realiza a detecção de anomalias, que também é o de vizinhos mais próximos. A cada iteração, seleciona-se um lote de $2S$ espectrogramas de treino. Então, alimenta-se o extrator de características com estes espectrogramas e coleta-se as suas respectivas saídas, *i.e.*, $2S$ vetores de características. O passo seguinte consiste em utilizar o NN para selecionar os S vetores de características com a menor distância média aos n vetores vizinhos. Então, escolhe-se aleatoriamente S espectrogramas do lote contendo os $2S$ anteriormente selecionados, e treina-se o extrator para mapear os S espectrogramas aos S vetores de características. Este processo é repetido por $maxIter$ iterações. Ao final do processo, alimenta-se o extrator de características com todos os espectrogramas de treino e utiliza-se o NN para calcular a distância média entre os vetores de características resultantes e os seus n vizinhos mais próximos. O limiar da detecção de anomalias é determinado pela seleção de um percentil destas distâncias, *e.g.*, 99.

O Algoritmo 1 resume o processo descrito. As Figuras 39 e 40 ilustram a seleção das instâncias e o treinamento do extrator de características, respectivamente.

7.1.3 Base de dados

Neste capítulo, foram utilizadas 3 bases de dados na realização dos experimentos. A primeira delas trata de falhas em caixas de engrenagens e já foi apresentada em 4.1.1. As outras duas referem-se a falhas em compressores. Uma destas trata de falhas em mancais, enquanto a

Algorithm 1 Treinando o detector de anomalias

- Inicializar o extrator de características com pesos aleatórios;
- Definir o número máximo de iterações ($maxIter$);
- Definir o tamanho do lote de treino (s);
- Definir o número de vizinhos do algoritmo NN (n);
- $iter \leftarrow 0$;

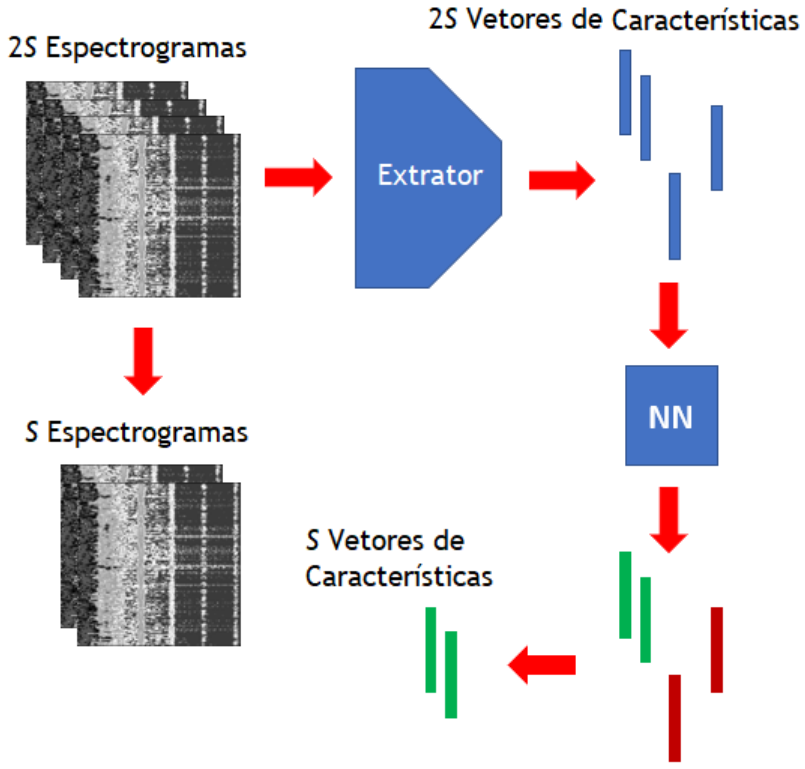
while $iter \leq maxIter$ **do**

- Selecionar de forma aleatória um lote com $2S$ espectrogramas do conjunto de treino;
- Alimentar o extrator de características com esses espectrogramas e coletar os respectivos vetores de características;
- Considerando os $2S$ vetores de características coletados, utilizar o NN para selecionar os S vetores mais próximos uns dos outros;
- Selecionar aleatoriamente S espectrogramas do lote com $2S$ espectrogramas;
- Treinar o extrator de características com os S espectrogramas selecionados (entradas) e os S vetores de características (saídas esperadas);
- $iter \leftarrow iter + 1$;

end while

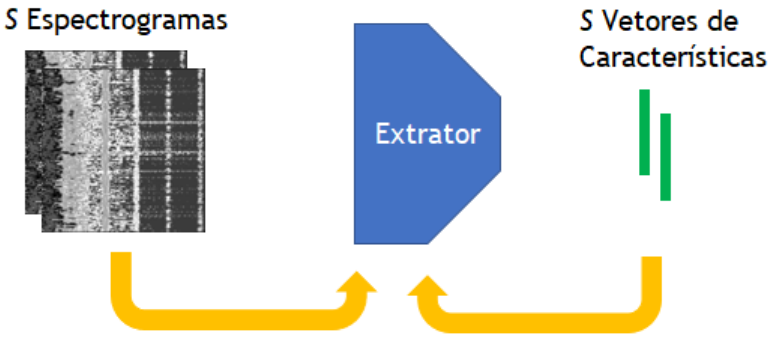
- Alimentar o extrator de características com todos os espectrogramas de treino e coletar os vetores de características correspondentes;
- Usar o NN para calcular a média das distâncias entre cada vetor de características e os seus n vizinhos mais próximos;
- Usar as distâncias calculadas para definir o limiar de anomalia.

Figura 39 – Selecionando instâncias para treinar o extrator de características.



Fonte: O autor (2021).

Figura 40 – Treinando o extrator de características.



Fonte: O autor (2021).

outra trata de falhas relacionadas a mancais e válvulas de forma combinada. Estas bases foram apresentadas em 3.1.2.1 e 3.1.2.2, respectivamente.

7.1.4 Arranjo experimental

Para cada conjunto de dados, as classes foram divididas em dois grupos: normal (classe P1) e anômalo (demais classes). Dado que apenas instâncias do grupo normal são necessárias para treinar o extrator de características, estas foram divididas em instâncias de treino (80% dos espectrogramas) e de teste (20%). Os dados do grupo anômalo foram utilizados apenas na avaliação do detector de anomalias. As quantidades de espectrogramas alocados em cada grupo estão listadas na Tabela 49. Os modelos foram treinados por 10.000 iterações ($maxIter =$ 10.000, no Algoritmo 1).

Tabela 46 – Número de espectrogramas alocados por grupo.

Conjunto de dados	Normal (Treino)	Normal (Teste)	Anômalo (Teste)
Caixa de engrenagens	1.440	360	16.200
Falhas em mancais	800	200	3.000
Falhas em mancais e válvulas	800	200	12.000

Fonte: O autor (2021).

Também foi analisada a influência do número de vizinhos do seletor de protótipos e do classificador no desempenho do detector de anomalias. A métrica de desempenho utilizada neste estudo foi a acurácia. Variou-se o número de vizinhos em dois momentos: no treinamento do extrator de características (NN como seletor de protótipos) e na detecção de anomalias (NN como classificador de uma classe). Em ambos os momentos, o número de vizinhos variou de 1 a 6.

Para cada cenário de treino considerado, *i.e.*, para cada número de vizinhos do seletor de protótipos, foram treinados 15 modelos a fim de se avaliar a repetibilidade dos resultados. Os modelos foram treinados na plataforma Colab com suporte de GPU.

7.2 RESULTADOS E DISCUSSÕES

Conforme mencionado em 7.1.3, três conjuntos de dados foram utilizados para avaliar o detector de anomalias obtido por meio do processo descrito em 7.1.2. Os conjuntos de dados foram: falhas em caixas de engrenagens, falhas em mancais de compressores e falhas em mancais e válvulas de compressores. Também foi investigado como o desempenho do detector de anomalias é influenciado pela quantidade de vizinhos do algoritmo de vizinhos mais próximos. Neste estudo, a quantidade de vizinhos variou de 1 a 6, conforme já mencionado, tanto para o seletor de protótipos, quanto para o classificador de uma classe.

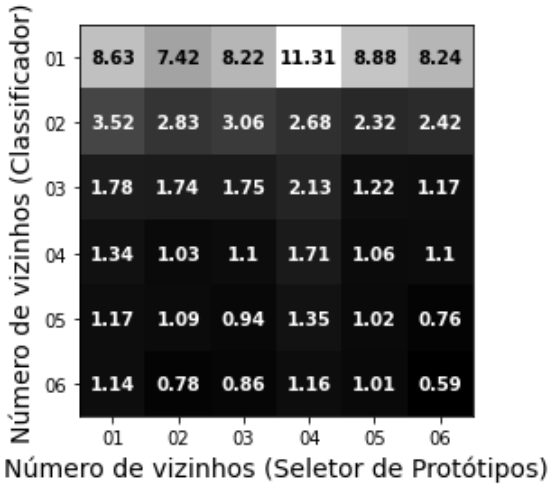
Os primeiros experimentos utilizaram o conjunto de dados de falhas em caixas de engrenagens. As Figuras 41 e 43 mostram as acurácias médias (em %) dos detectores de anomalias para diferentes quantidades de vizinhos mais próximos, considerando as classes normal e anômala, respectivamente. Enquanto isso, as Figuras 42 e 44 mostram os desvios padrões relativos aos valores de acurácia dos modelos, também para as classes normal e anômala, respectivamente. Tais resultados compreendem 36 combinações de quantidades de vizinhos mais próximos considerados, *i.e.*, o NN na seleção de protótipos pode apresentar de 1 a 6 vizinhos, bem como o NN na classificação. Os valores nestas figuras correspondem apenas às instâncias de teste, ou seja, que não foram utilizadas no processo de treinamento dos detectores de anomalias. Quinze modelos foram treinados para cada combinação.

Figura 41 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em caixas de engrenagens.

Fonte: O autor (2021).

As Figuras 41 e 42 apresentam resultados relativos à classe normal. Pela Figura 41, percebe-se que os detectores de anomalias se tornaram mais precisos à medida que a quantidade de vizinhos no classificador aumentou. Por outro lado, aumentando-se a quantidade de vizinhos na seleção de protótipos, não foi possível notar variação de desempenho significativa em termos de acurácia.

Figura 42 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em caixas de engrenagens.



Fonte: O autor (2021).

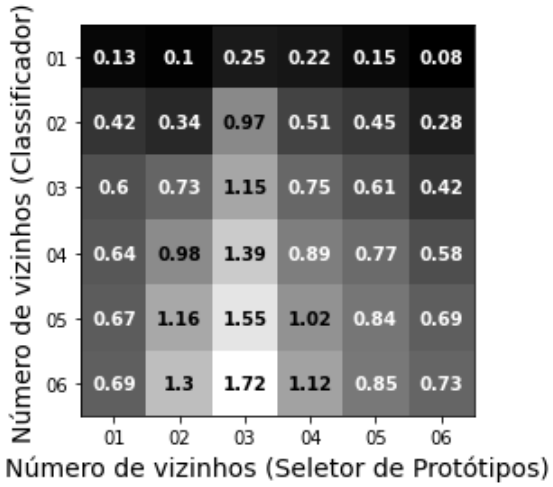
Figura 43 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em caixas de engrenagens.

Fonte: O autor (2021).

Os classificadores com mais vizinhos também apresentaram a menor variação de desempenho. Em média, o desvio padrão das acurácias de tais classificadores diminuiu de 8,78 para 0,92. Estes dois valores se referem ao desvio padrão médio considerando os cenários em que os números de vizinhos dos classificadores são 1 e 6, respectivamente. Em contrapartida, o desvio padrão não mudou significativamente quando a quantidade de vizinhos do seletor de protótipos variou.

É possível observar na Figura 43 que a acurácia média dos dados anômalos variou menos que a dos dados normais, tanto em relação à quantidade de vizinhos do classificador, quanto a do seletor de protótipos. Além disso, conforme visto na Figura 44, os valores de desvio padrão em

Figura 44 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em caixas de engrenagens.

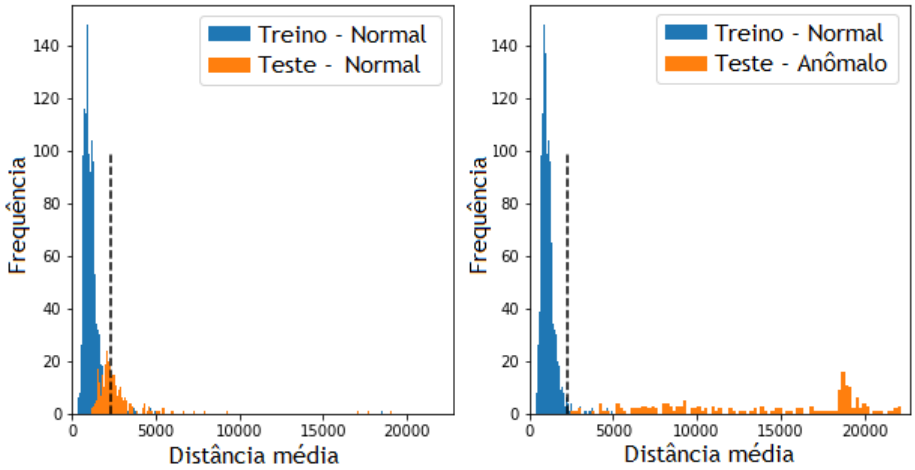


Fonte: O autor (2021).

todos os cenários ficaram abaixo de dois.

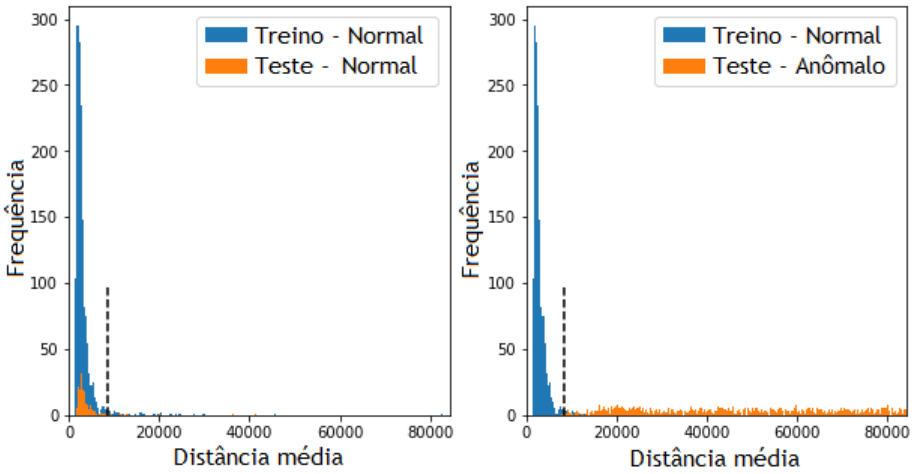
Nota-se uma outra diferença de comportamento relacionado às classes normal e anômala. Quando se aumenta a quantidade de vizinhos no classificador, o desempenho do detector de anomalias tende a aumentar com os dados normais, mas diminui com os dados anômalos. Este comportamento pode ser explicado pela análise das distâncias médias entre as instâncias testadas e as instâncias de treino. As Figuras 45 e 46 comparam as distribuições dessas distâncias para dados de treino e de teste, normais e anômalos. A Figura 45 ilustra o cenário no qual os classificadores consideram a distância da amostra de teste a apenas 1 vizinho, enquanto a Figura 46 ilustra o cenário no qual 6 vizinhos são considerados.

Figura 45 – Histogramas das distâncias médias entre amostras de treino e de teste, considerando classificadores com apenas 1 vizinho.



Fonte: O autor (2021).

Figura 46 – Histogramas das distâncias médias entre amostras de treino e de teste, considerando classificadores com 6 vizinhos.



Fonte: O autor (2021).

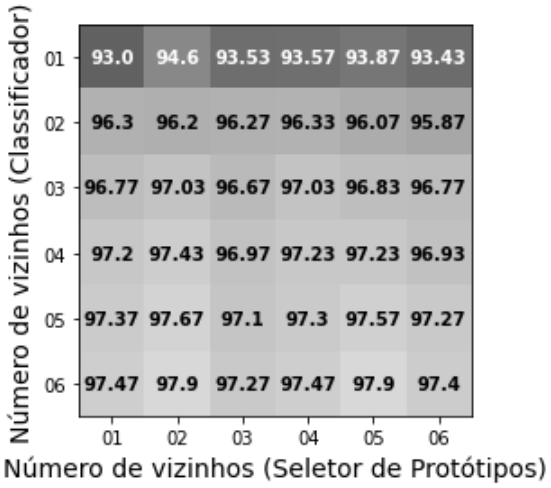
Observa-se um ajuste mais preciso entre as distribuições *Treino - Normal* e *Teste - Normal* no cenário com 6 vizinhos, mostrado na Figura 46. Conforme explicado em 7.1.2, apenas dados pertencentes à classe normal foram utilizados para definir o limiar de anomalia. Desta forma, o ajuste mais preciso melhorou os resultados relacionados à classe normal, uma vez que as instâncias de teste pertencentes a esta classe reproduziram o comportamento observado das instâncias de treinamento, também pertencentes à classe normal. Por outro lado, uma vez que as distâncias médias das instâncias anômalas de teste já estavam localizadas inicialmente além do limiar de anomalia, o deslocamento desse não resultou em mudanças significativas, mesmo que tenha causado uma leve diminuição na acurácia para a classe anômala.

Os mesmos experimentos foram realizados com os dados de falhas em mancais de compressores. As Figuras 47 e 49 mostram a acurácia média (%) do detector de anomalias construído com o método proposto, considerando as classes normal e anômala, respectivamente. Além disso, as Figuras 48 e 50 mostram os desvios padrões relativos aos valores de acurácia dos modelos, também para as classes normal e anômala, respectivamente. Elas contêm resultados para 36 combinações de quantidades de vizinhos mais próximos considerados, *i.e.*, 6 para o classificador e 6 para o seletor de protótipos. Os resultados correspondem aos valores médios de acurácia obtidos por 15 modelos em cada cenário, considerando apenas dados de teste.

As Figuras 47 e 48 apresentam os resultados obtidos para a classe normal. Pela Figura 47, assim como ocorreu no cenário em que foram analisadas as falhas em caixas de engrenagens, percebe-se que os detectores de anomalias tornaram-se mais precisos à medida que a quantidade de vizinhos no classificador aumentou. Por outro lado, aumentando-se a quantidade de vizinhos na seleção de protótipos, não houve variação de desempenho significativa em termos de acurácia.

Os classificadores com mais vizinhos também apresentaram a menor variação de desempenho. Em outras palavras, o desvio padrão das acurácias de tais classificadores diminuiu

Figura 47 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em mancais de compressores.



Fonte: O autor (2021).

Figura 48 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em mancais de compressores.

Fonte: O autor (2021).

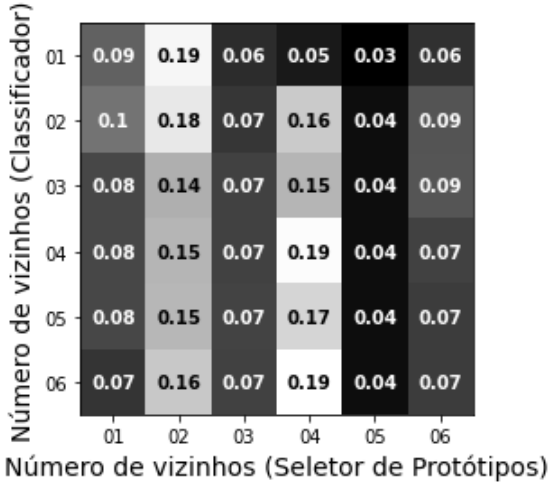
de 2,81 para 1,18. Estes valores se referem ao desvio padrão médio considerando os cenários em que os números de vizinhos dos classificadores são 1 e 6, respectivamente. Entretanto, o desvio padrão não mudou significativamente quando a quantidade de vizinhos que variou foi a do seletor de protótipos. Este comportamento também foi observado no cenário das falhas em caixas de engrenagens.

É possível observar na Figura 49 que a acurácia média dos dados anômalos variou menos que a dos dados normais, tanto em relação à quantidade de vizinhos do classificador quanto a do seletor de protótipos. Além disso, conforme visto na Figura 50, os valores de desvio padrão em todos os cenários ficaram abaixo de 0,2.

Figura 49 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em mancais de compressores.

Fonte: O autor (2021).

Figura 50 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em mancais de compressores.



Fonte: O autor (2021).

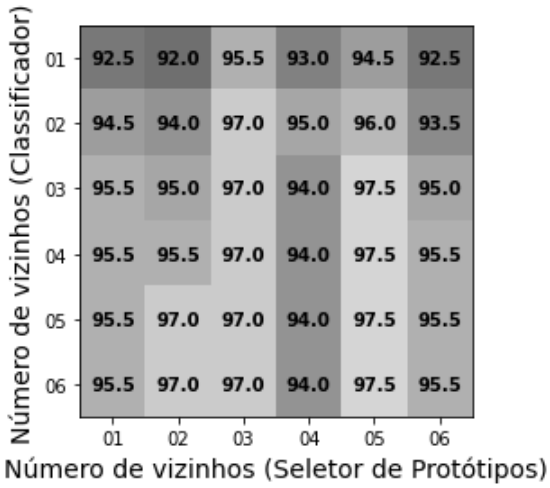
Neste cenário, também se observa uma diferença de comportamento relacionada às classes normal e anômala. Quando aumenta-se a quantidade de vizinhos no classificador, o desempenho do detector de anomalias tende a aumentar com os dados normais, mas permanece aproximadamente constante com os dados anômalos. Este comportamento pode ser explicado com a mesma análise desenvolvida no cenário com falhas em caixas de engrenagens, ilustrada pelas Figuras 45 e 46.

No geral, obteve-se maiores valores de acurácia com os dados referentes a falhas em mancais de compressores. Isto possivelmente está relacionado à natureza dos módulos de falhas.

Os dados referentes às caixas de engrenagens apresentam uma única forma de falha, mas com graus diferentes de severidade. Considerando os graus de severidade próximos, as características associadas aos mesmos tendem a ser próximas, dificultando a identificação precisa de cada um, *e.g.*, classe normal e classe com danos muito pequenos. Por outro lado, os dados de falhas em compressores contêm diferentes modalidades de defeitos, resultando em características mais diversificadas. Isto facilita o processo de detecção das anomalias. Outro fator que influencia a variação de desempenho para as bases de dados diferentes são as condições de coleta dos sinais. Na coleta dos sinais de vibração com as caixas de engrenagens, as condições de trabalho eram variadas, *i.e.* frequências de rotação e cargas diferentes. Este fato resulta numa variedade maior de sinais e padrões pertencentes a uma mesma classe. Em contrapartida, a condição de trabalho envolvida na coleta de sinais para o cenário do compressor foi constante, tornando os sinais pertencentes a cada classe mais homogêneos, facilitando o aprendizado as características.

Os experimentos também foram realizados com dados de falhas em mancais e válvulas de compressores. As Figuras 51 e 53 mostram a acurácia média (%) dos detectores de anomalias construídos com o método proposto, considerando as classes normal e anômala, respectivamente. Além disso, as Figuras 52 e 54 mostram os desvios padrões relativos aos valores de acurácia dos modelos, também para as classes normal e anômala, respectivamente. Elas contêm resultados de diferentes combinações de quantidades de vizinhos mais próximos considerados, *i.e.*, de 1 a 6 para o classificador e de 1 a 6 para o seletor de protótipos. Os resultados correspondem aos valores médios de acurácia obtidos por 15 modelos em cada cenário, considerando apenas dados de teste.

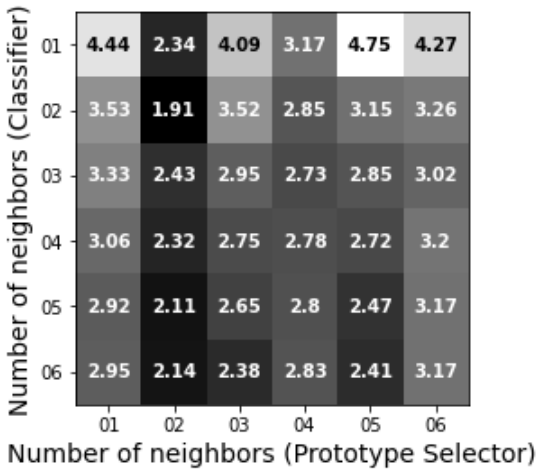
Figura 51 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em mancais e válvulas de compressores.



Fonte: O autor (2021).

As Figuras 51 e 52 apresentam os valores médios de acurácia e também os desvios padrões, respectivamente, para a classe normal. As figuras trazem a mesma tendência observada

Figura 52 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe normal, para o conjunto de dados referente a falhas em mancais e válvulas de compressores.



Fonte: O autor (2021).

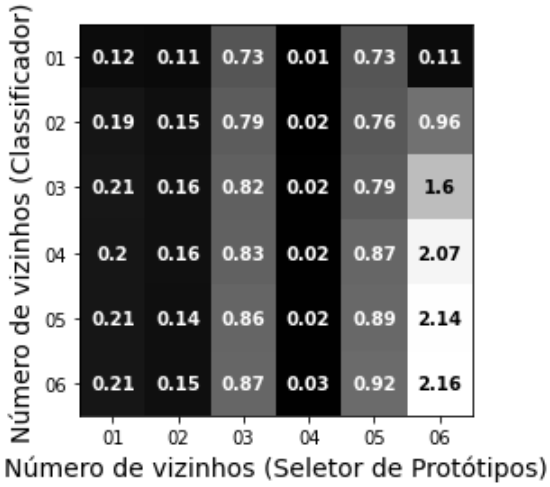
Figura 53 – Acurácia média (%) dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em mancais e válvulas de compressores.

Fonte: O autor (2021).

nos resultados obtidos com as outras duas bases de dados. Em outras palavras, percebe-se que os detectores de anomalias se tornaram mais precisos à medida que a quantidade de vizinhos no classificador aumentou. Por outro lado, aumentando-se a quantidade de vizinhos na seleção de protótipos, não houve variação de desempenho significativa em termos de acurácia. Além disso, os classificadores com mais vizinhos apresentaram a menor variação de desempenho, ou seja, menores desvios padrões. Entretanto, o desvio padrão não mudou significativamente quando a quantidade de vizinhos do seletor de protótipos variou.

De acordo com a Figura 53, a acurácia média dos dados anômalos variou menos que a dos dados normais, tanto em relação à quantidade de vizinhos do classificador quanto a do

Figura 54 – Desvios padrões referentes às acurácias dos detectores de anomalias construídos através do método proposto, considerando apenas a classe anômala, para o conjunto de dados referente a falhas em mancais e válvulas de compressores.



Fonte: O autor (2021).

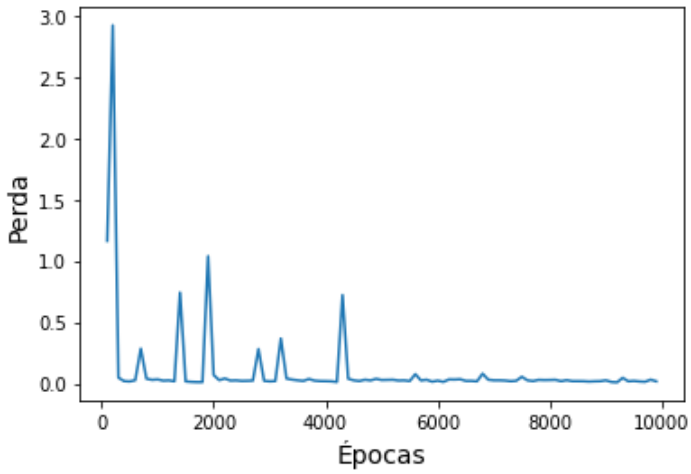
seletor de protótipos. Além disso, conforme visto na Figura 54, a média dos valores de desvio padrão considerando todos os cenários ficou abaixo de 1.

Também foi observada a diferença de comportamento relacionada às classes normal e anômala. Quando aumenta-se a quantidade de vizinhos no classificador, o desempenho do detector de anomalias tende a aumentar com os dados normais, mas apresenta uma leve diminuição com os dados anômalos. A mesma análise desenvolvida no cenário com falhas em caixas de engrenagens, ilustrada pelas Figuras 45 e 46, pode explicar esse comportamento.

As Figuras 55, 56 e 57 ajudam a visualizar como o processo de treinamento dos extratores de características converge, considerando as bases falhas em caixas de engrenagens, falhas em mancais de compressores e falhas em mancais e válvulas de compressores, respectivamente. Tais imagens se referem à evolução do valor da perda em função do número de épocas de treinamento. Nas três imagens, é possível observar que existe uma tendência de redução no valor da perda à medida que o número de épocas de treinamento aumenta. Isto mostra que o mapeamento das características de instâncias normais está convergindo para uma região restrita e comum do espaço de características, conforme esperado. Observa-se também que, eventualmente, há um aumento repentino e momentâneo no valor da perda. Este aumento se deve à forma como os lotes de treinamento são formados, *i.e.*, os espectrogramas são escolhidos aleatoriamente. O aumento resulta da seleção de espectrogramas cujos padrões ainda não foram aprendidos pelo extrator de características, levando ao mapeamento destas instâncias para regiões diferentes do espaço de características.

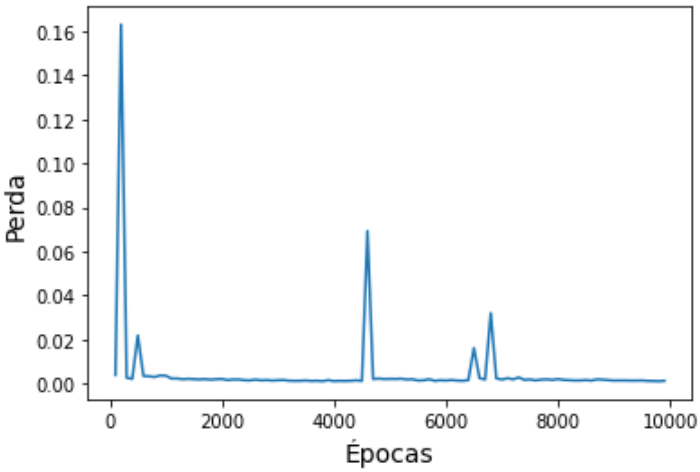
Conforme mencionado em 7.1.4, o treinamento dos modelos foi realizado na plataforma Colab. Esta plataforma na nuvem disponibiliza gratuitamente diferentes opções de *hardware*, de acordo com a demanda a seus serviços. Quando a demanda pelos serviços é alta, por exemplo,

Figura 55 – Evolução do valor da perda em função do número de épocas de treinamento, considerando a base falhas em caixas de engrenagens.



Fonte: O autor (2021).

Figura 56 – Evolução do valor da perda em função do número de épocas de treinamento, considerando a base Falhas em mancais de compressores.

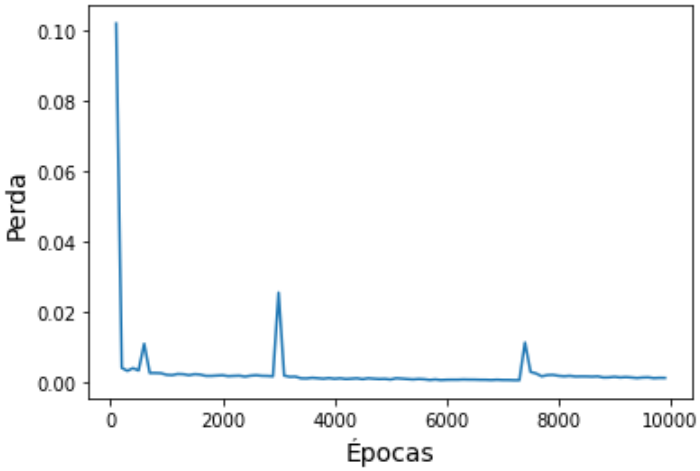


Fonte: O autor (2021).

o tempo de uso gratuito de computadores com GPU é menor. Isto significa que o processo de treinamento de todos os modelos envolveu o uso de diferentes computadores, uma vez que, em dados momentos, apenas computadores com CPU estavam disponíveis. Este fato tornou a comparação do custo computacional por meio de métricas como o tempo de treinamento injusta.

Ainda assim, é possível tecer alguns comentários acerca da complexidade do processo de treinamento. No que diz respeito ao extrator de características, a quantidade de parâmetros treináveis, como esperado, variou de acordo com as dimensões dos espectrogramas de entrada. Tomando como exemplo os extratores treinados com a base de falhas em caixas de engrenagens, cujos espectrogramas apresentavam dimensões 128 x 128 *pixels*, a quantidade de parâmetros treináveis foi 549.360,00. Enquanto isso, a base de falhas em mancais, com espectrogramas 64 x 64 *pixels*, resultou em extratores com 161.254,00 parâmetros treináveis. Consequentemente,

Figura 57 – Evolução do valor da perda em função do número de épocas de treinamento, considerando a base Falhas em mancais e válvulas de compressores.



Fonte: O autor (2021).

os extratores associados aos espectrogramas com dimensões 128 x 128 tenderam a apresentar treinamentos mais longos que aqueles associados aos espectrogramas de dimensões 64 x 64. Vale ressaltar que os modelos apresentaram a mesma profundidade, *i.e.*, quantidade de camadas.

O número de vizinhos do classificador e do seletor de protótipos também interfere no custo computacional da detecção de anomalias, seja na etapa do treinamento ou da classificação. Porém, a quantidade de protótipos do classificador e a quantidade de instâncias por lote para treinar o extrator de características contribuem de forma mais significativa para o aumento dos tempos de classificação e treinamento, respectivamente. Isto ocorre porque, quanto maior o número de protótipos do algoritmo de vizinhos mais próximos, maior a quantidade de distâncias a serem computadas todas as vezes que o algoritmo é acionado. Isto sugere que a técnica proposta neste capítulo apresenta um custo computacional mais elevado em relação à técnica proposta no capítulo 6, por exemplo. Esta última apresentou um processo de treinamento típico de uma rede neural para o extrator de características, sem o uso de seleção de protótipos ou mesmo outros processos em conjunto. Por outro lado, o aumento do custo computacional acompanhou uma melhoria na acurácia do detector de anomalias, tomando os resultados obtidos pelas duas propostas com a base de falhas em caixas de engrenagens.

7.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo, foi proposto um método para treinar um extrator de características baseado em aprendizado profundo para problemas de detecção de anomalias. O método, que utiliza a seleção de protótipos durante o treinamento do extrator, buscou otimizar a extração de características para agrupar instâncias pertencentes a uma mesma distribuição, *i.e.*, a classe normal.

O método foi testado em três bases de dados relacionadas a falhas em caixas de en-

grenagens e compressores, obtendo resultados promissores. O desempenho dos detectores de anomalias contendo o extrator, em termos de acurácia, foi superior a 95% para a classe normal, e próximos a 100% para a classe anômala.

Analisou-se também a influência das quantidades de vizinhos mais próximos do seletor de protótipos e do classificador no desempenho do detector de anomalias. Foi visto que o aumento da quantidade de vizinhos do classificador resultou numa melhoria significativa de acurácia, sugerindo que o maior número de vizinhos contribuiu para a separabilidade de dados normais e anômalos. Isto ocorreu porque o uso de mais vizinhos permitiu uma melhor avaliação da densidade de instâncias de treinamento, *i.e.*, normais, em torno das instâncias testadas. Em contrapartida, o aumento do número de vizinhos do seletor de protótipos modificou o desempenho significativamente.

Os experimentos realizados com a abordagem proposta são aprofundados no capítulo seguinte. Tais experimentos envolvem novas bases de dados e objetivam identificar possíveis limitações.

8 USO DE REDES NEURAIS CONVOLUCIONAIS TREINADAS COM SELETORES DE PROTÓTIPOS NA EXTRAÇÃO DE CARACTERÍSTICAS PARA A DETECÇÃO DE ANOMALIAS: ESTUDO COMPARATIVO

Neste capítulo, compara-se os resultados obtidos no Capítulo 7, *i.e.*, por meio da solução proposta no mesmo, com resultados de trabalhos da literatura que utilizam as mesmas bases de dados, ou, na falta destes, com resultados obtidos por técnicas de detecção de anomalias consolidadas na literatura.

8.1 METODOLOGIA

Nesta seção, apresenta-se detalhes sobre os experimentos realizados ao longo estudo desenvolvido no capítulo.

8.1.1 Base de dados

Nesta etapa do estudo, foram utilizadas 7 bases de dados na realização dos experimentos. A primeira base trata de falhas em caixas de engrenagens e foi apresentada em 4.1.1. Outras duas bases tratam de falhas em compressores, sendo uma destas relativa a defeitos em mancais e outra relativa a falhas combinadas em mancais e válvulas. Estas bases foram apresentadas em 3.1.2.1 e 3.1.2.2, respectivamente. Estas três bases de dados não são públicas e foram disponibilizadas pela *Universidad Politécnica Salesiana*, através do grupo de pesquisas GIDTEC (do espanhol *Grupo de Investigación y Desarrollo en Tecnologías Industriales*), conforme já mencionado.

As quatro bases restantes pertencem ao conjunto de dados MIMII (do inglês *Malfunctioning Industrial Machine Investigation and Inspection*) (PUROHIT et al., 2019), composto por sinais sonoros utilizados em inspeções de máquinas industriais. Tais sinais foram coletados por microfones de oito canais e pertencem a quatro tipos ou categorias de dispositivos, que operam sob condições normais e anômalas em ambientes fabris reais. Os tipos de dispositivos em questão são: bombas de água, válvulas, ventiladores industriais e sistemas de deslizamento linear. Cada tipo conta com sete dispositivos, *e.g.*, existem sete bombas, sete ventiladores, etc. Os sete dispositivos de cada tipo podem ou não ser do mesmo modelo.

Os sons produzidos por estes dispositivos podem ser dinâmicos ou estacionários, além de apresentar diferentes características e graus de dificuldade na identificação do mal funcionamento. As diferentes características podem estar associadas a diferentes fatores, *e.g.*, o modelo do dispositivo, a função desempenhada por este e o tipo de falha. As dificuldades, por outro lado, podem estar associadas a aspectos como o nível de ruído no ambiente e ao tipo de falha. Os tipos de falhas são variados, incluindo vazamentos, desbalanceamentos, contaminações, etc. As diferentes funções desempenhadas pelos dispositivos e alguns exemplos de defeitos estão

listados na Tabela 47. Três níveis de ruído ambiente foram considerados na formação das bases de dados, cada um relacionado a uma razão sinal-ruído: -6 dB, 0 dB e 6 dB.

Tabela 47 – Lista de funções desempenhadas e exemplos de mal funcionamento.

Tipo de dispositivo	Operação	Exemplos de defeitos
Válvula	Diferentes frequências de abertura e fechamento	Mais de dois tipos de contaminação
Bomba	Sucção/Descarga em reservatórios de água	Vazamento, contaminação, entupimento, etc.
Ventilador	Operação normal	Desbalanceamento, mudança de tensão, entupimento, etc.
Sistema de deslizamento linear	Deslizamento em diferentes velocidades	Trilho danificado, correia frouxa, falta de graxa, etc.

Fonte: O autor (2021).

Este conjunto de dados está disponível para download gratuito em (MIMII. . . ,). Ele contém 26.092 arquivos de sons relativos às condições normais de operação, considerando os quatro tipos de dispositivos citados no parágrafo anterior. Ele também apresenta 6.065 sons coletados sob condições anômalas de operação.

Para os estudos desenvolvidos neste capítulo com o conjunto de dados MIMII, foram escolhidos os sinais sonoros de acordo com os critérios listados na Tabela 48. O objetivo desta escolha foi reduzir a quantidade de informação utilizada no estudo, devido a questões como o armazenamento dos espectrogramas, custo computacional e tempo do processo de treinamento dos extratores de características, etc.

Tabela 48 – Características dos sinais utilizados nos estudos deste capítulo, referentes ao conjunto de dados MIMII.

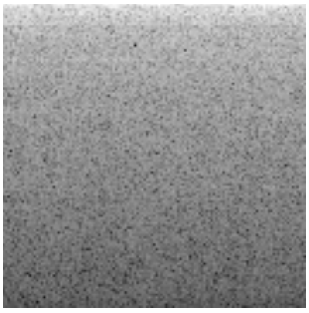
Descrição	Característica
Canal do microfone	1
Razão sinal-ruído	0 dB
Dispositivos	Bomba, ventilador, válvula e sistema de deslizamento linear com código de identificação 00

Fonte: O autor (2021).

8.1.2 Organização da bases de dados

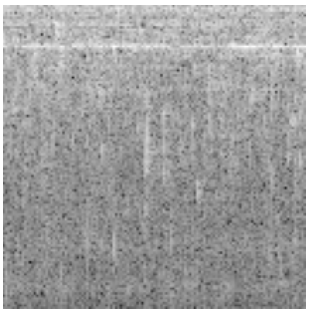
As bases relacionadas à caixa de engrenagem e aos compressores são as mesmas utilizadas nos experimentos do Capítulo 7. De modo semelhante, os sinais sonoros das bases MIMII foram representados por meio de espectrogramas obtidos via STFT, cuja parametrização foi a mesma utilizada nos experimentos dos capítulos 5. Exemplos de espectrogramas relacionados aos dispositivos podem ser observados da Figura 58 à 61.

Figura 58 – Exemplo de espectrograma obtido dos sinais coletados dos ventiladores.



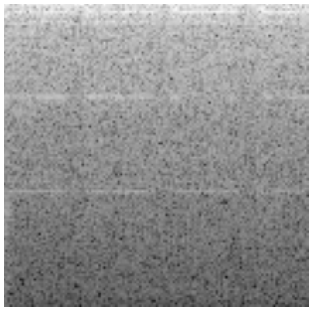
Fonte: O autor (2021).

Figura 59 – Exemplo de espectrograma obtido dos sinais coletados das bombas.



Fonte: O autor (2021).

Figura 60 – Exemplo de espectrograma obtido dos sinais coletados dos sistemas de deslizamento linear.



Fonte: O autor (2021).

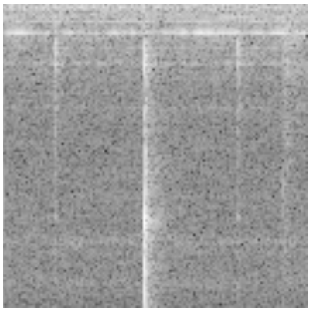


Figura 61 – Exemplo de espectrograma obtido dos sinais coletados das válvulas.

Fonte: O autor (2021).

8.1.3 Arranjo experimental

Para cada conjunto de dados, as classes foram divididas em dois grupos: normal e anômalo. Dado que apenas instâncias do grupo normal são necessárias para treinar o extrator de características, estas foram divididas em instâncias de treino (80% dos espectrogramas) e de teste (20%). Os dados do grupo anômalo foram utilizados apenas na avaliação (teste) do detector de anomalias. As quantidades de espectrogramas alocados em cada grupo estão listadas na Tabela 49. Os modelos foram treinados por 10.000 iterações.

Tabela 49 – Número de espectrogramas alocados por grupo.

Conjunto de dados	Normal (Treino)	Normal (Teste)	Anômalo (Teste)
Caixa de engrenagens	1.440	360	16.200
Falhas em mancais	800	200	3.000
Falhas em mancais e válvulas	800	200	12.000
Válvulas	800	200	200
Bombas	800	200	200
Ventiladores	800	200	200
Sistemas de deslizamento linear	800	200	200

Fonte: O autor (2021).

Para cada cenário de treino considerado, *i.e.*, cada detector de anomalias e cada conjunto de dados, foram treinados 15 modelos objetivando-se avaliar a repetibilidade dos resultados. Os modelos foram treinados na plataforma Google Colab, com suporte de GPU.

8.2 RESULTADOS E DISCUSSÕES

Nesta seção, apresenta-se e discute-se os resultados obtidos ao longo do estudo desenvolvido neste capítulo.

8.2.1 Resultados de referência

O artigo que introduz a base MIMII (PUROHIT et al., 2019) apresenta uma série de resultados de *benchmark*, ou de referência, para a detecção de anomalias com este conjunto de dados. Os experimentos objetivam detectar defeitos por meio de modelos treinados apenas com sinais sonoros referentes às condições normais de operação dos dispositivos.

A técnica utilizada na obtenção dos resultados de *benchmark* foi o *autoencoder* convolucional, que é uma técnica relativamente comum e bem sucedida em problemas de detecção de anomalias, conforme visto em 2.2. Os sinais sonoros utilizados na obtenção dos resultados de *benchmark* foram coletados pelo Canal 1 dos microfones. Além disso, a representação utilizada para os sinais foi o espectrograma de log-Mel (KOIZUMI et al., 2020). Os *autoencoders* foram treinados para aprender a reconstruir apenas os dados referentes às condições normais de operação dos dispositivos, de modo que o erro de reconstrução referente às condições anômalas fosse superior ao de condições normais. Mais detalhes acerca das transformações dos sinais e

da arquitetura do *autoencoder* podem ser vistos em (PUROHIT et al., 2019). Os resultados de referência considerados neste trabalho estão listados na Tabela 50, e estão relacionados àqueles modelos treinados com sinais cujas características estão listadas na Tabela 48. Os desempenhos dos detectores de anomalia são avaliados através da área sob a curva ROC (AUC, do inglês *area under curve*).

Tabela 50 – Resultados de referência para os dados da base MIMII.

Dispositvo	**Código**	**Razão sinal-ruído**	**AUC**
Válvula	00	0 dB	0,55
Bomba	00	0 dB	0,65
Ventilador	00	0 dB	0,63
Sistema de deslizamento linear	00	0 dB	0,99

Fonte: O autor (2021).

Além dos resultados de referência, são apresentados resultados obtidos por algumas técnicas de uso comum em problemas de detecção de anomalias. Também são apresentados os resultados obtidos em outros trabalhos da literatura que utilizam a base de dados MIMII. Por outro lado, as bases de falhas em caixas de engrenagens e compressores não apresentam resultados de referência. A análise comparativa é feita com os resultados de técnicas da literatura utilizadas na detecção de anomalias.

8.2.2 Resultados obtidos com aprendizado de máquina tradicional

Nesta etapa, busca-se confirmar a necessidade do uso de aprendizado profundo em problemas com as características apresentadas, *e.g.*, quantidade de dados, tipos de dado de entrada, etc. Sendo os resultados obtidos apenas com o aprendizado de máquina tradicional satisfatórios, *e.g.*, taxas de acerto próximas a 100%, não haveria por que buscar técnicas de maior complexidade e custo computacional para a resolução dos problemas abordados.

As técnicas de aprendizado de máquina tradicional escolhidas foram: máquina de vetores de suporte de uma classe, ou OC-SVM, fator *outlier* local, ou LOF, e floresta de isolamento, ou IF. Estas foram escolhidas por serem técnicas consolidadas na literatura no que se refere à detecção de anomalias. A implementação das técnicas foi feita através do módulo de Python Scikit-Learn (PEDREGOSA et al., 2011).

A Tabela 51 lista os resultados obtidos com as três técnicas utilizadas, além dos resultados de referência. Os melhores resultados obtidos em cada base de dados estão destacados em vermelho. A obtenção dos resultados incluiu uma busca inicial da configuração paramétrica mais adequada para cada uma das técnicas, considerando cada base de dados. Então, o desempenho do LOF foi avaliado para uma quantidade de vizinhos que variou de 2 a 11. Da mesma forma, o IF foi avaliado para uma quantidade de estimadores que variou entre 10 e 100, de 10 em 10 vizinhos,

e o OC-SVM foi avaliado para valores de gama de 1×10^{-1} a 1×10^{-10}. O desempenho dos detectores de anomalias foi quantificado por meio da AUC média e do desvio padrão, obtidos a partir de 15 modelos treinados em cada cenário. O desvio padrão está representado pelo número entre parênteses.

Tabela 51 – Comparação dos resultados de referência com resultados obtidos por detectores de anomalias baseados em técnicas de aprendizado de máquina tradicional. Os melhores resultados obtidos em cada base de dados estão destacados em vermelho.

Dispositvo	**Referência**	**LOF**	**IF**	**OC-SVM**
Válvula	0,55	**0,91 (0,00)**	0,58 (0,01)	0,68 (0,00)
Bomba	**0,65**	0,48 (0,00)	0,51 (0,01)	0,60 (0,00)
Ventilador	**0,63**	0,49 (0,00)	0,49 (0,00)	0,55 (0,00)
Sistema de deslizamento linear	**0,99**	0,93 (0,00)	0,82 (0,05)	0,96 (0,00)
Caixa de engrenagens	X	0,71 (0,00)	0,92 (0,02)	**0,95 (0,00)**
Compressor (falhas combinadas)	X	**0,94 (0,00)**	0,55 (0,02)	0,61 (0,00)
Compressor (falhas em mancais)	X	**0,99 (0,00)**	0,57 (0,03)	0,84 (0,00)

Fonte: O autor (2021).

Observa-se que o desempenho dos algoritmos variou significativamente de acordo com a base de dados. Considerando apenas os dados relativos aos dispositivos da base MIMII, vê-se que o desempenho de referência foi superado apenas no cenário Válvula, no qual o LOF apresentou uma AUC média de 0,91. Nos demais cenários, os resultados de referência foram superiores. No entanto, os resultados obtidos nos cenários Sistema de deslizamento linear e Bomba foram próximos ao resultado de referência, *e.g.*, as diferenças do resultados obtidos pelo OC-SVM em relação aos de referência, para as duas bases, foram de aproximadamente 3% e 7,7%, respectivamente.

Com relação às 3 bases restantes, *i.e.*, caixa de engrenagens e compressor, as técnicas de aprendizado de máquina tradicional apresentaram um desempenho elevado, *i.e.*, os maiores valores de AUC em cada base foram iguais ou maiores que 0,94.

Os resultados também mostram um variação significativa de desempenho das técnicas utilizadas nas diferentes bases de dados. O LOF, por exemplo, apresentou o melhor desempenho em três bases, mas apresentou o pior desempenho em outras três, com valores de AUC variando entre 0,48 e 0,99. A OC-SVM, por outro lado, apresentou valores de AUC superiores a 0,9 em duas bases, mas o desempenho foi relativamente baixo, *e.g.*, abaixo de 0,7, em outras quatro bases. Algo semelhante ocorreu com a IF.

Outro aspecto digno de nota concerne a variação da configuração paramétrica adequada de cada técnica para cada cenário, *e.g.*, enquanto o melhor resultado do LOF foi obtido com três vizinhos na base de Válvulas, o melhor desempenho no cenário Sistema de deslizamento linear

ocorreu com 11 vizinhos. Estas questões são particularmente importantes, pois nem sempre existe a disponibilidade de dados anômalos em quantidade suficiente para se avaliar com segurança o desempenho dos modelos treinados, considerando as diferentes configurações paramétricas dos mesmos. Desta forma, torna-se necessária a busca por técnicas estáveis e capazes de gerar bons resultados em diferentes cenários, com uma configuração paramétrica aproximadamente padrão. É possível que as técnicas de aprendizado profundo possam atenuar, ou mesmo resolver, este problema. Isto será visto mais detalhadamente nas seções futuras deste capítulo.

8.2.3 Resultados obtidos em trabalhos da literatura com a base MIMII

Conforme visto no Capítulo 1, a detecção de anomalias não é algo recente na literatura, sendo o assunto discutido em milhares de trabalhos. Alguns destes trabalhos foram desenvolvidos com os dados da base MIMII. Estes trabalhos fazem uso de diversas técnicas, como *autoencoders*, aprendizado de máquina tradicional, etc. As abordagens desenvolvidas nos trabalhos são sucintamente descritas a seguir.

8.2.3.1 *Deep Context-Aware Novelty Detection* (CANDE) (RUSHE; NAMEE, 2020)

Esta (RUSHE; NAMEE, 2020) é uma abordagem semi-supervisionada que utiliza rótulos auxiliares para proporcionar informações contextuais. Estas permitem que um único modelo se adapte a uma variedade de contextos nos quais as definições de normal e novidade sejam diferentes. A arquitetura desta abordagem compreende dois componentes: um *autoencoder* profundo e uma função de codificação contextual.

O rótulo denota a partição do conjunto de dados que indica o contexto da informação retirada, *e.g.*, o dia ou o ambiente no qual o dado foi coletado, ou mesmo a sua classe, se conhecida. O condicionamento do *autoencoder* profundo é feito com modulação linear baseada em características. Em outras palavras, a saída de uma certa camada do *autoencoder* passa por uma transformação que envolve mudança de escala e translação. O grau desta transformação depende de um vetor de contexto, que pode ser obtido por técnicas simples e diretas, como o *one-hot encoding*.

A detecção da anomalias se dá pela análise do erro de reconstrução do *autoencoder*. Sabendo-se que este é treinado apenas com dados pertencentes à classe normal, espera-se que amostras de teste pertencentes à classe normal apresentem um erro de reconstrução abaixo de um determinado limiar. Em contrapartida, espera-se que amostras de teste pertencentes a classes anômalas apresentem erros acima deste limiar. A função de custo utilizada no treinamento dos *autoencoders* foi o MSE. Os dados utilizados foram espectrogramas na escala log-Mel.

Os resultados obtidos para as máquinas de ID 00 estão listados na Tabela 52.

8.2.3.2 *Outlier-Exposed Classifier* (OEC)

Esta abordagem (PRIMUS, 2020) visa transformar o problema de detecção de anomalias não supervisionado num problema supervisionado. Tomando-se como referência a máquina com um determinado ID, *e.g.*, Bomba 00, os dados normais associados a esta máquina compõem uma das classes, enquanto os dados normais associados a máquinas do mesmo tipo, mas com IDs diferentes, e de máquinas de outros tipos, compõem uma segunda classe. Desta forma, é possível treinar o detector de anomalias de forma supervisionada.

O modelo utilizado foi uma ResNet de arquitetura proposta por Koutini *et al.* (KOUTINI; EGHBAL-ZADEH; WIDMER, 2019) Os dados utilizados, assim como na abordagem anterior, foram espectrogramas na escala log-Mel.

Os resultados obtidos para as máquinas de ID 00 estão listados na Tabela 52.

8.2.3.3 Autoencoders com unidades LSTM

A abordagem proposta por Jalali *et al.* (JALALI; SCHINDLER; HASLHOFER,) consiste de um *autoencoder* profundo cujas camadas são unidades LSTM. A escolha deste tipo de camada se deve ao bom desempenho no aprendizado de características temporais. A detecção de anomalias é realizada através da análise do erro de reconstrução da informação de entrada, *i.e.*, espectrogramas de Mel. A ideia do processo de detecção é a mesma tradicionalmente utilizada em abordagens com *autoencoders*, isto é, dado que o treinamento do modelo se dá apenas com dados normais, espera-se que o erro de reconstrução de dados normais seja inferior ao de dados anômalos.

Os resultados obtidos para as máquinas de ID 00 estão listados na Tabela 52.

8.2.3.4 Neuron-Net

Esta abordagem (DURKOTA et al., 2020) usa redes neurais siamesas (KOCH; ZEMEL; SALAKHUTDINOV, 2015) para extrair características da informação de entrada, *i.e.*, espectrogramas obtidos por meio da STFT. A detecção de anomalias é feita pelo KNN.

A rede siamesa padrão consiste de três estruturas: dois *encoders* e um agregador. Os *encoders* criam representações da informação de entrada num espaço latente de características, utilizando um mesmo conjunto de pesos. O agregador calcula a distância de duas instâncias codificadas e retorna um *score* de similaridade. A rede siamesa é treinada para informar se duas instâncias apresentadas pertencem ou não à mesma distribuição. Durante o processo de treinamento do extrator de características, pares de instâncias aleatoriamente selecionados no conjunto de dados alimentam a rede. Se ambas as instâncias vêm de uma mesma distribuição, *e.g.*, bomba ID 00 e bomba ID 00, a rede minimiza a distância das características extraídas. Caso contrário, *e.g.*, bomba ID 00 e ventilador ID 00, ou mesmo bomba ID 00 e bomba ID 01, esta distância é maximizada.

O detector de anomalias KNN é implementado por meio da biblioteca de Python PyOD (ZHAO; NASRULLAH; LI, 2019). Os resultados obtidos para as máquinas de ID 00 estão listados na Tabela 52.

8.2.3.5 Discussão dos resultados na base MIMII

Como pode ser visto na Tabela 52, a técnica que apresentou maior valor médio de AUC foi a OEC, *i.e.*, 0,96 considerando os quatro cenários. Em segundo lugar, aparece a Neuron-Net, com AUC média igual a 0,84. As demais técnicas, por outro lado, apresentaram desempenhos mais próximos aos valores de referência. O desempenho superior das técnicas em destaque pode estar relacionado à forma como os detectores de anomalias foram treinados. Conforme visto em 8.2.3.2 e 8.2.3.4, os modelos de detecção de anomalias foram treinados com dados pertencentes a dispositivos diferentes, *e.g.*, bombas e ventiladores de IDs diferentes. Isto permitiu a exposição dos modelos a uma maior variedade de comportamentos, o que provavelmente permitiu a obtenção de redes capazes de extrair características que tornassem processo de discriminação de instâncias normais e anômalas mais eficaz. Desta forma, esta estratégia mostrou-se promissora para o problema tratado.

Em contrapartida, a estratégia exige a disponibilidade de sons obtidos a partir outras fontes, além do dispositivo monitorado, *e.g.*, Se o dispositivo monitoriado é a Bomba 00, é necessário sinais relativos também a dispositivos como a Bomba 01, o Ventilador 04, etc. Dependendo do cenário tratado, tais dados podem nem mesmo existir. Também não fica claro se quaisquer sinais podem ser utilizados neste processo como sinais anômalos, *e.g.*, o som emitido por um animal, ou se o desempenho resulta da existência de sinais coletados de dispositivos semelhantes, *e.g.*, válvulas com diferentes IDs. Esta questão pode ser um impeditivo ao uso deste tipo de estratégia, pois, como foi visto no Capítulo 1, o cenário mais comum na detecção de anomalias é que os dados com maior disponibilidade sejam aqueles referentes à operação normal do dispositivo monitorado. Por conta do uso de outros sons além daqueles pertencentes à classe normal do equipamento monitorado (no treinamento dos modelos), os resultados da OEC e da Neuron-Net não serão utilizados como referência para o estudo desenvolvido nesta tese, apesar de promissores. A comparação focará nos modelos treinados com dados normais relativos a um único equipamento.

Outro ponto digno de nota é que os trabalhos não mencionam se tais resultados foram obtidos através da média de um número n de detectores treinados, ou se apenas fazem referência ao melhor resultado atingido por cada modelo.

8.2.4 Resultados obtidos com as bases Caixa de Engrenagens e Compressor, com técnicas de detecção de anomalias da literatura

Na ausência de trabalhos da literatura que tratam da detecção de anomalias com as bases Caixa de Engrenagens e Compressor, os experimentos referentes a estas bases foram

Tabela 52 – Comparação dos resultados de referência com resultados obtidos por detectores de anomalias baseados em técnicas apresentadas em trabalhos da literatura. Os melhores resultados obtidos em cada base de dados estão destacados em vermelho.

Dispositvo	**Referência**	**CANDE**	**OEC**	**AE + LSTM**	**NEURON-NET**
Válvula	0,55	0,48	**0,98**	0,71	0,97
Bomba	0,65	0,61	**0,93**	0,69	0,79
Ventilador	0,63	0,66	**0,94**	0,57	0,58
Sistema de deslizamento linear	0,99	0,97	0,98	0,98	**1,00**

Fonte: O autor (2021).

realizados com algumas técnicas apresentadas em 2.2. A realização destes experimentos é um passo importante, pois permite estabelecer uma referência de desempenho para a técnica proposta.

Uma das abordagens utilizadas foi o *autoencoder* convolucional. Este foi treinado apenas com dados pertencentes à classe normal, de modo que dados anômalos sejam detectados por meio da comparação dos seus erros de reconstrução com os erros dos dados normais.

O segundo sistema de detecção de anomalias foi uma abordagem híbrida constituída de dois estágios. O primeiro estágio foi um extrator de características baseado em aprendizado profundo, tratando-se do *encoder* de um *autoencoder* treinado para reconstruir os dados normais. O segundo estágio foi a OC-SVM, responsável por detectar as anomalias pela análise das caracteristicas extraídas pelo *encoder*. O treinamento dos dois estágios se deu de forma separada.

As arquiteturas do *autoencoder* e da abordagem híbrida são as mesmas apresentadas em 6.1.4.

Duas técnicas baseadas em redes generativas adversarias foram testadas nesta etapa: a AnoGAN (SCHLEGL et al., 2017) e a GANomaly (AKCAY; ATAPOUR-ABARGHOUEI; BRECKON, 2018). No entanto, não foi possível obter a convergência dos modelos com as configurações estruturais propostas pelos autores nos respectivos trabalhos. Por conta disso, os resultados inerentes a estas técnicas não foram incluídos na análise.

Os resultados obtidos pelo *autoencoder* e pela abordagem híbrida estão listados na Tabela 53, sendo apresentados em termos da AUC média e do desvio padrão (entre parênteses). Tais resultados foram obtidos a partir de 15 modelos treinados em cada cenário. Em vermelho, tem-se destacado os melhores resultados para cada base de dados.

Pode-se observar que o *autoencoder* apresentou o melhor resultado em duas bases: Compressor com falhas combinadas e Compressor com falhas em mancais. Por outro lado, a abordagem híbrida apresentou um desempenho superior na base Caixa de Engrenagens. Este comportamento pode estar relacionado à complexidade dos dados de cada base. A base Caixa de Engrenagens é uma base mais complexa, pois trata de uma única modalidade de falha

Tabela 53 – Resultados obtidos pelo *autoencoder* e pela abordagem híbrida *encoder* + OC-SVM, com as bases Caixa de Engrenagens e Compressor.

Dispositivo	Referência	Autoencoder	Encoder + OC-SVM
Caixa de Engrenagens	X	0,60 (0,30)	**0,87 (0,06)**
Compressor (falhas combinadas)	X	**0,98 (0,01)**	0,85 (0,10)
Compressor (falhas em mancais)	X	**0,99 (0,01)**	0,90 (0,05)

Fonte: O autor (2021).

com diferentes níveis de severidade e diferentes condições de operação, *e.g.*, frequência de rotação e carga. É provável que uma métrica relativamente simples como o erro de reconstrução não consiga representar as classes do problema de forma satisfatória, devendo-se recorrer às características extraídas de camadas intermediárias do *autoencoder*. Em relação às bases do Compressor, por apresentarem classes associadas a diferentes modalidades de falhas, é possível que a diferença entre as classes seja significativa para que o erro de reconstrução dos dados pertencentes a estas consiga separá-las de forma satisfatória.

8.2.5 Resultados obtidos com a técnica proposta

Nesta seção, apresenta-se os resultados obtidos com a técnica proposta no Capítulo 7, considerando as sete bases de dados utilizadas. O desempenho dos detectores de anomalias foi aferido por meio da AUC, de modo a facilitar a comparação com resultados da literatura. A Tabela 54 apresenta estes resultados juntamente com os resultados de referência. Quinze modelos foram treinados para cada base de dados, de modo que os valores de AUC apresentados são as médias destes quinze modelos. Os desvios padrões estão entre parênteses. Destacados em vermelho, tem-se o melhor resultado obtido em cada base de dados.

Tabela 54 – Resultados do processo de detecção de anomalias utilizando o extrator de características (FE) treinado conforme a proposta apresentada no Capítulo 7, com o classificador de uma classe baseado no algoritmo NN.

Dispositvo	Referência	FE + NN
Válvula	0,55	**0,73 (0,15)**
Bomba	0,65	**0,83 (0,03)**
Ventilador	**0,63**	0,58 (0,03)
Sistema de deslizamento linear	**0,99**	0,72 (0,08)
Caixa de Engrenagens	X	**0,96 (0,00)**
Compressor (falhas combinadas)	X	**0,97 (0,02)**
Compressor (falhas em mancais)	X	**1,00 (0,00)**

Fonte: O autor (2021).

Observa-se na Tabela 54 que o desempenho dos sistemas obtidos com a solução proposta variou de acordo com a base utilizada. Considerando as bases Caixa de Engrenagens e Compressor, com falhas em mancais e combinadas, observa-se valores médios de AUC superiores a 0,95. O maior valor foi obtido na base Compressor (com falhas em mancais), *i.e.*, AUC média igual a 0,996 e desvio padrão próximo a 0. Em relação as bases MIMII, o desempenho obtido apresentou uma variação maior. Considerando os dispositivos Bomba e Válvula, os valores médios de AUC foram inferiores a 0,9, porém ainda foram superiores aos valores de referência. Em contrapartida, considerando os dispositivos Ventilador e Sistema de deslizamento linear, o desempenho da solução proposta não conseguiu superar os resultados de referência.

Mais experimentos foram realizados com a solução proposta, de modo a investigar com maior profundidade as dificuldades desta abordagem com os dados da base MIMII, principalmente em relação aos dispositivos Ventilador e Sistema de deslizamento linear. O primeiro ponto investigado foi o classificador de uma classe no topo do sistema de detecção de anomalias. Além do classificador baseado em NN, foram utilizados o LOF, a IF e a OC-SVM. Os resultados são apresentados na Tabela 55. Considerando cada classificador, quinze modelos foram treinados para cada base de dados. Os resultados correspondem aos valores médios de AUC, juntamente com os desvios padrões (entre parênteses). Destacados em vermelho, tem-se o melhor resultado obtido em cada base de dados.

Tabela 55 – Resultados do processo de detecção de anomalias utilizando o extrator de características (FE) treinado conforme a proposta apresentada no Capítulo 7, com quatro algoritmos de classificação de uma classe diferentes.

Dispositvo	**Referência**	**FE + NN**	**FE + LOF**	**FE + IF**	**FE + OC-SVM**
Válvula	0,55	**0,73 (0,15)**	0,50 (0,01)	0,55 (0,03)	0,54 (0,03)
Bomba	0,65	**0,83 (0,03)**	0,61 (0,03)	0,65 (0,06)	0,66 (0,06)
Ventilador	**0,63**	0,58 (0,03)	0,52 (0,02)	0,59 (0,04)	0,58 (0,04)
Sistema de deslizamento linear	**0,99**	0,72 (0,08)	0,78 (0,05)	0,50 (0,08)	0,57 (0,07)
Caixa de Engrenagens	X	0,96 (0,00)	0,95 (0,01)	**0,97 (0,00)**	**0,97 (0,00)**
Compressor (falhas combinadas)	X	**0,97 (0,02)**	0,93 (0,02)	0,88 (0,02)	0,89 (0,02)
Compressor (falhas em mancais)	X	**1,00 (0,00)**	0,96 (0,01)	0,95 (0,01)	0,95 (0,01)

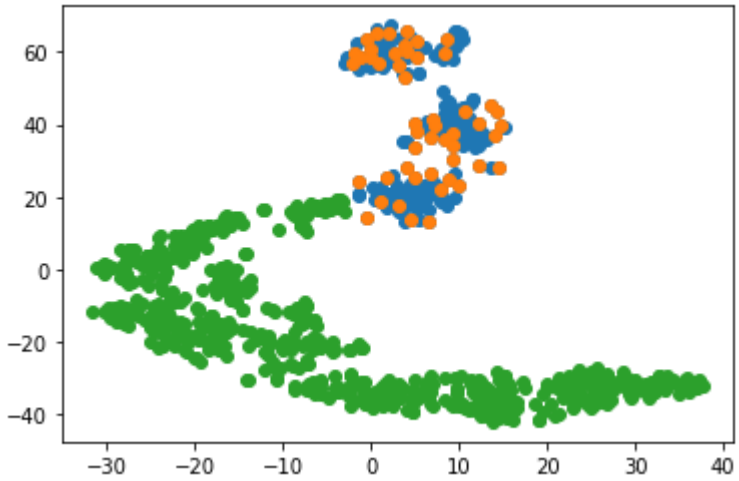
Fonte: O autor (2021).

Estes resultados sugerem que o problema não está relacionado ao classificador, mas às características extraídas. Em quatro bases de dados, o classificador baseado no algoritmo de vizinhos mais próximos esteve associado aos maiores valores médios de AUC. Nos casos em que não esteve, *e.g.*, Ventilador, Sistema de deslizamento linear e Caixa de Engrenagens, a modificação do classificador não resultou em melhoria significativa da AUC média. O processo de extração de características pode não estar promovendo uma separabilidade adequada das

características de instâncias normais e anômalas nas bases de dados mencionadas.

O segundo ponto investigado foi o conjunto de características extraídas pelos modelos inicializados aleatoriamente, *i.e.*, antes do processo de treinamento. Para avaliar esta questão, utilizou-se o t-SNE (do inglês *t-Distributed Stochastic Neighbor Embedding*). O t-SNE (MAATEN; HINTON, 2008) é um algoritmo de aprendizado de máquina para visualização de dados baseado na incorporação de vizinhança estocástica. É uma ferramenta de uso frequente na visualização de dados com alta dimensionalidade, *e.g.*, 128 dimensões. As Figuras 62, 63, 64, 65 e 66 mostram as representações em duas dimensões, obtidas por meio do t-SNE, das características extraídas por modelos inicializados aleatoriamente para as bases de dados Caixa de engrenagens, Falhas em mancais de compressor, Ventilador e Válvula, respectivamente.

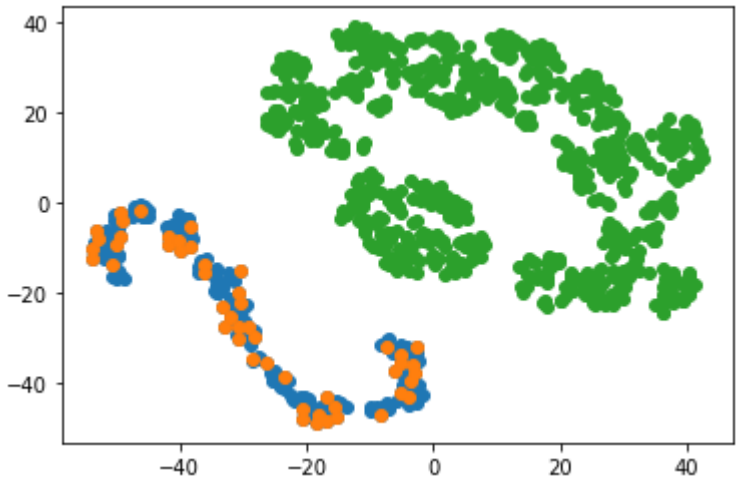
Figura 62 – Representação em duas dimensões das características extraídas dos dados da base Caixa de engrenagens, pelo modelo inicializado aleatoriamente. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

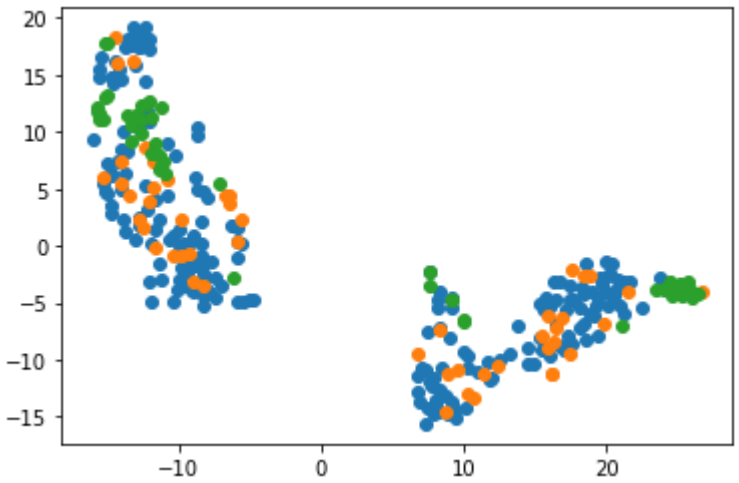
Nas representações relativas às bases Caixa de Engrenagens e Compressor (falhas em mancais), mostradas nas Figuras 62 e 63, observa-se uma maior separação entre dados normais, *i.e.*, representados em azul e laranja, e anômalos, *i.e.*, representados em verde, quando comparadas às representações relativas às bases Ventilador, Válvula e Sistema de deslizamento linear, mostradas nas Figuras 64, 65 e 66. Na realidade, as Figuras 64 , 65 e 66 mostram uma mistura significativa das características extraídas de instâncias normais e anômalas. Esta mistura se deve a dois aspectos: a inicialização aleatória do extrator de características e a existência de uma quantidade maior de padrões semelhantes em espectrogramas de instâncias normais e anômalas, *e.g.*, em relação aos casos apresentados nas Figuras 62 e 63. Como os pesos do extrator de características são inicializados aleatoriamente, *i.e.*, não estão otimizados para identificar os padrões que diferenciam as instâncias normais das anômalas, a maior quantidade de padrões semelhantes resulta num mapeamento das características extraídas para regiões próximas e/ou iguais do espaço de características.

Figura 63 – Representação em duas dimensões das características extraídas dos dados da base Compressor (falhas em mancais), pelo modelo inicializado aleatoriamente. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

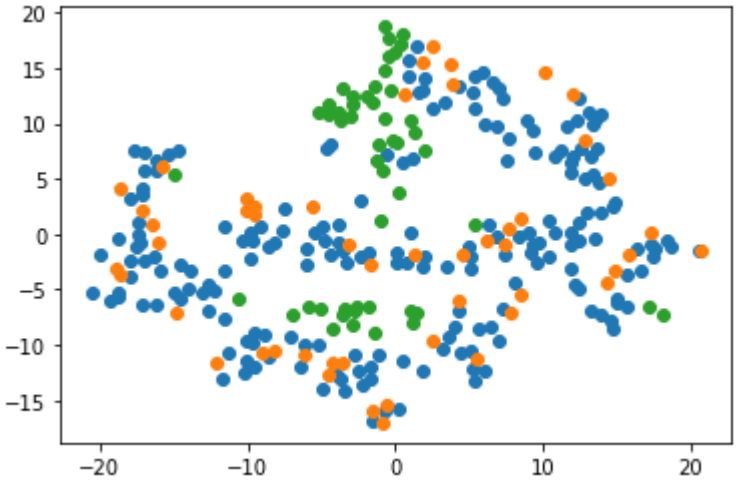
Figura 64 – Representação em duas dimensões das características extraídas dos dados da base Ventilador, pelo modelo inicializado aleatoriamente. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

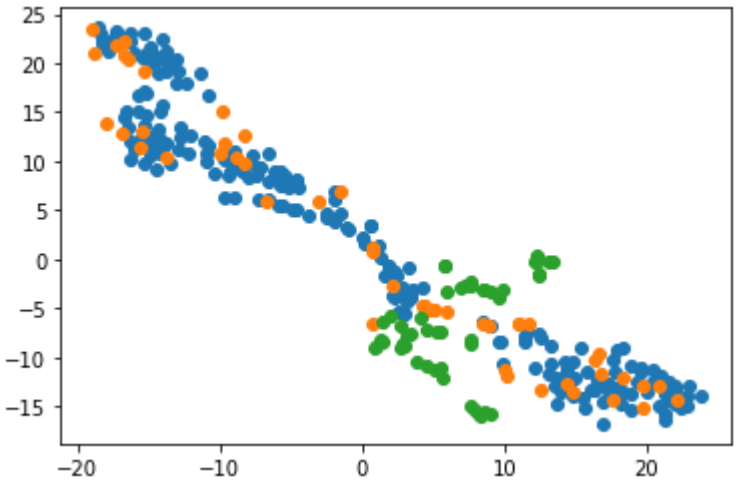
Este aspecto é responsável por impor dificuldades ao processo de treinamento do extrator de características, pois os lotes de dados de treinamento são selecionados de forma aleatória em cada iteração. Estando as características dos dados anômalos em meio à distribuição das características dos dados normais, a escolha aleatória das instâncias de treinamento direciona o processo de aprendizado para incluir também as características dos dados anômalos. O efeito disso pode ser observado na distribuição de características extraídas dos mesmos modelos associados às Figuras 62, 63, 64, 65 e 66, porém após o processo de treinamento dos extratores. Estas novas distribuições de características estão ilustradas nas Figuras 67, 68, 69, 70 e 71. Isto

Figura 65 – Representação em duas dimensões das características extraídas dos dados da base Válvula, pelo modelo inicializado aleatoriamente. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

Figura 66 – Representação em duas dimensões das características extraídas dos dados da base Sistema de deslizamento linear, pelo modelo inicializado aleatoriamente. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.

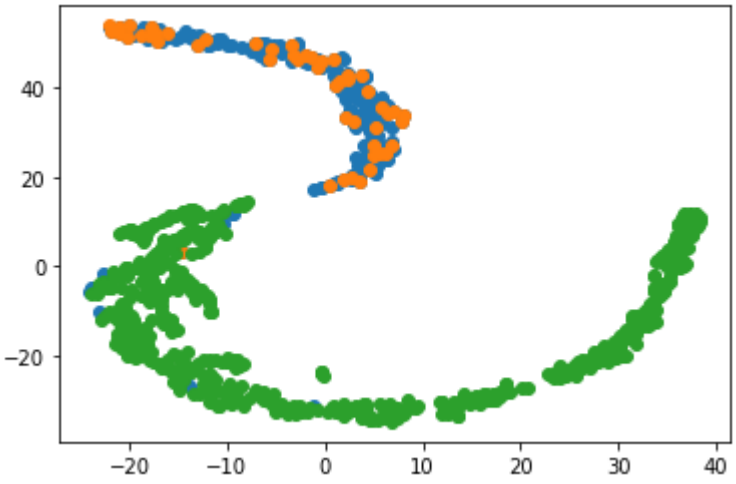


Fonte: O autor (2021).

sugere que, no estágio atual de desenvolvimento, a técnica proposta não é adequada para tratar de maneira eficaz todas as distribuições de dados vistas, tornando necessário a realização de algumas modificações.

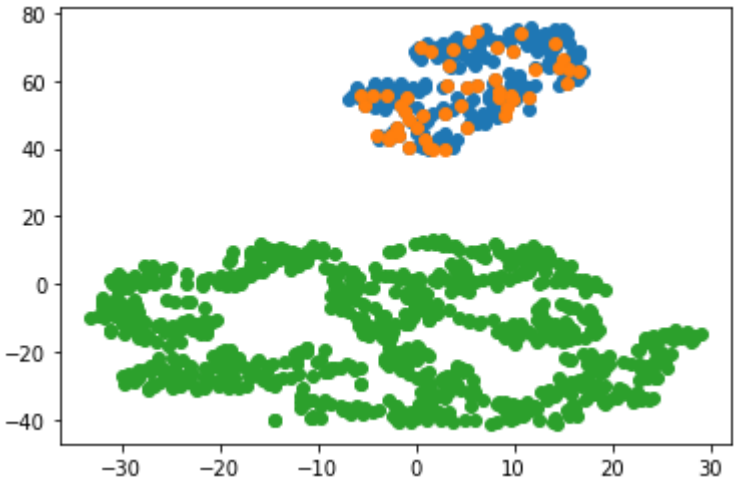
Pretende-se aprofundar tais modificações em trabalhos futuros, porém, uma pequena discussão acerca de uma delas será iniciada nesta tese. Uma possível solução para o problema seria inserir um processo de agrupamento das características extraídas dos dados normais antes de iniciar o processo de treinamento do extrator. Este processo pode identificar grupos homogêneos

Figura 67 – Representação em duas dimensões das características extraídas dos dados da base Caixa de engrenagens, pelo modelo treinado. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

Figura 68 – Representação em duas dimensões das características extraídas dos dados da base Compressor (falhas em mancais), pelo modelo treinado. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.

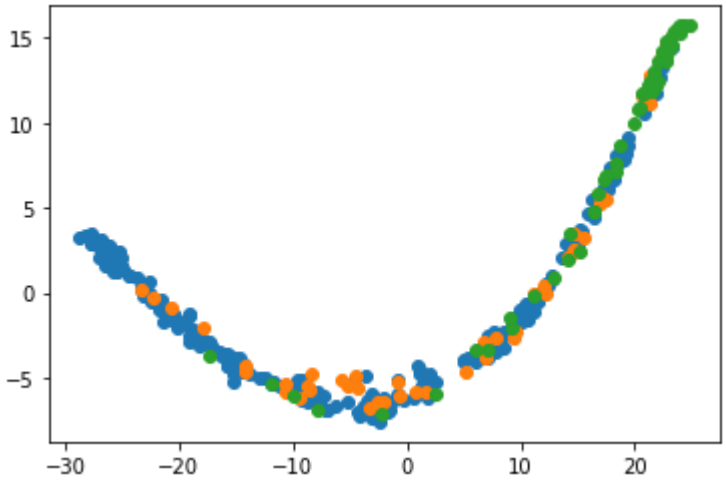


Fonte: O autor (2021).

de dados normais, com menor probabilidade de conter representações de dados anômalos em meio às representações de dados normais, *e.g.*, após o processo de agrupamento, é possível que as representações de dados anômalos se posicionem nas periferias dos agrupamentos. Isto permitiria que o processo de aprendizado com instâncias de treino escolhidas aleatoriamente pudesse ser realizado dentro de cada agrupamento identificado.

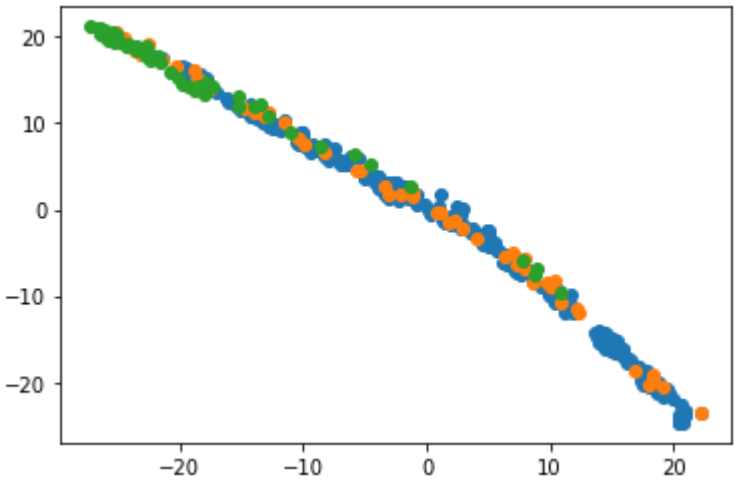
Outro aspecto digno de nota é a evolução das representações bidimensionais das características ao longo do processo de treinamento do extrator. Observando as Figuras 62 e 67, referentes às características extraídas antes e após o processo de treinamento com dados da

Figura 69 – Representação em duas dimensões das características extraídas dos dados da base Ventilador, pelo modelo treinado. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

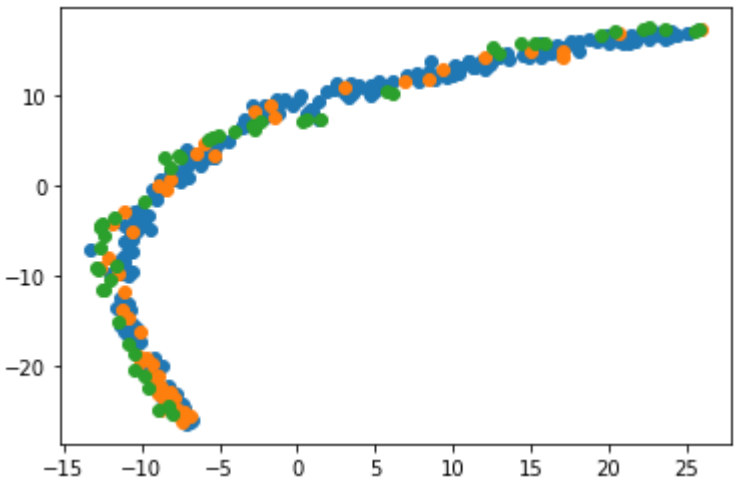
Figura 70 – Representação em duas dimensões das características extraídas dos dados da base Válvula, pelo modelo treinado. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

base Caixa de Engrenagens, aparenta que o extrator inicializado de forma aleatória é suficiente para realizar a extração de características no processo de detecção de anomalias, por conta da separação acentuada entre dados normais e anômalos. No entanto, ressalta-se que o t-SNE é uma técnica que auxilia na visualização dos dados, e as distâncias observadas na representação das características não são necessariamente proporcionais às distâncias observadas no espaço de características. Como exemplo, a AUC associada ao extrator inicializado aleatoriamente, cujas características estão representadas na Figura 62, é 0,91, enquanto a AUC do modelo treinado, cujas características estão representadas na Figura 67, é 0,97.

Figura 71 – Representação em duas dimensões das características extraídas dos dados da base Sistema de deslizamento linear, pelo modelo treinado. Nas cores azul, laranja e verde tem-se as representações dos dados de treino, teste-normal e teste-anômalo, respectivamente.



Fonte: O autor (2021).

8.2.6 Comparação dos resultados da técnica proposta com resultados de técnicas da literatura

Nesta subseção, compara-se os resultados da técnica proposta no capítulo 7 com técnicas encontradas na literatura.

8.2.6.1 Bases MIMII

A Tabela 56 concentra todos os resultados referentes as bases MIMII. Observa-se que os maiores valores de AUC obtidos em cada uma das quatro bases de dados estão atrelados a abordagens que utilizaram dados de dispositivos diferentes (OEC e NEURON-NET), pertencentes ou não ao mesmo tipo, no treinamento do detector de anomalias. Este tipo de abordagem mostra-se promissora, no entanto, depende da disponibilidade de dados destes dispositivos. Considerando abordagens que utilizaram apenas os dados referentes às classes normais, a abordagem proposta, *i.e.*, FE + KNN, apresentou o maior valor de AUC na base Bomba, o segundo maior valor na base Válvula e o terceiro maior na base Ventilador.

8.2.6.2 Bases Caixa de Engrenagens e Compressor

A Tabela 57 concentra todos os resultados referentes as bases Caixa de Engrenagens e Compressor. Observa-se que os maiores valores médios de AUC em duas das três bases foram obtidos pela técnica proposta. Na base Compressor (falhas combinadas), a técnica proposta ficou com o segundo maior valor médio de AUC, atrás do *autoencoder* por apenas 0,01. Ela também foi a única técnica a apresentar valores médios de AUC superiores a 0,95 nas três bases de dados.

Tabela 56 – Resultados do processo de detecção de anomalias utilizando todas as técnicas apresentadas neste capítulo, considerando as bases de dados MIMII.

Dispositvo	**Válvula**	**Bomba**	**Ventilador**	**Sistema de deslizamento linear**
Referência	0,55	0,65	0,63	0,99
LOF	0,91 (0,00)	0,48 (0,00)	0,49 (0,00)	0,93 (0,00)
IF	0,58 (0,01)	0,51 (0,01)	0,49 (0,00)	0,82 (0,05)
OC-SVM	0,68 (0,00)	0,60 (0,00)	0,55 (0,05)	0,96 (0,00)
CANDE	0,48	0,61	0,66	0,97
OEC	**0,98**	**0,93**	**0,94**	0,98
AE + LSTM	0,71	0,69	0,57	0,98
NEURON-NET	0,97	0,79	0,58	**1,00**
FE + KNN	0,73 (0,15)	0,83 (0,03)	0,58 (0,03)	0,72 (0,08)

Fonte: O autor (2021).

Tabela 57 – Resultados do processo de detecção de anomalias utilizando todas as técnicas apresentadas neste capítulo, considerando as bases de dados Caixa de Engrenagens e Compressor.

Dispositvo	**Caixa de Engrenagens**	**Compressor (falhas combinadas)**	**Compressor (falhas em mancais)**
Referência	X	X	X
LOF	0,71 (0,00)	0,94 (0,00)	0,99 (0,00)
IF	0,92 (0,02)	0,55 (0,02)	0,57 (0,03)
OC-SVM	0,95 (0,00)	0,61 (0,00)	0,84 (0,00)
Autoencoder	0,60 (0,30)	**0,98 (0,01)**	0,99 (0,01)
Encoder + OC-SVM	0,87 (0,06)	0,85 (0,10)	0,90 (0,05)
FE + KNN	**0,96 (0,00)**	0,97 (0,02)	**1,00 (0,00)**

Fonte: O autor (2021).

Conforme observado em análises e comparações realizadas neste capítulo, a técnica proposta apresentou resultados promissores e competitivos em ambos os grupos de bases de dados. Os melhores resultados vieram com os dados de Caixa de Engrenagens e Compressor. Em relação às bases MIMII, foram identificadas limitações em alguns cenários. Por outro lado, existem alternativas a serem abordadas em trabalhos futuros com potencial para superar estas limitações.

9 CONCLUSÕES

A detecção de anomalias é um tema relevante na literatura, estando presente em diversos domínios de aplicação, *e.g.*, detecção de doenças, fraudes em cartões de crédito, invasões em redes de comunicação, falhas em máquinas rotativas, etc. Ela é uma etapa essencial em processos decisórios, como o planejamento da manutenção em fábricas ou o início do tratamento de doenças graves.

Na literatura, encontram-se diversos tipos de abordagens de detecção de anomalias. Estes podem ou não exigir a disponibilidade de dados rotulados. Um dos cenários mais frequentes em aplicações reais considera apenas a disponibilidade de dados pertencentes à classe normal. Outros cenários relativamente frequentes incluem a disponibilidade reduzida de dados anômalos rotulados ou mesmo a inexistência completa de rótulos. Estas questões originam uma série de desafios para a detecção de anomalias, como a definição de uma região no espaço amostral que contenha todos os comportamentos normais possíveis, sem compreender os anômalos. O desbalanceamento entre as classes, ou mesmo o conhecimento de uma única classe, torna inadequado o uso de abordagens supervisionadas tradicionais no estabelecimento das fronteiras desta região. Isto faz necessário o uso de técnicas capazes de lidar com essas limitações.

Neste trabalho, desenvolveu-se uma forma de treinamento de extratores de características baseados em aprendizado profundo para detectores de anomalias híbridos. O desenvolvimento deste processo foi realizado em etapas, de modo a permitir uma melhor compreensão das particularidades do problema. Isto torna possível a proposição de melhorias capazes de superar as dificuldades impostas pelo problema de maneira mais eficiente. O sistema foi testado em bases de dados contendo sinais sonoros e de vibração coletados por sensores, relacionados a falhas em dispositivos eletromecânicos.

A primeira etapa do estudo tratou da seleção de dados com potencial para discriminar as classes dos problemas abordados. Esta etapa foi necessária por conta da característica de algumas bases, que apresentavam um mesmo fenômeno sendo descrito por diferentes sensores. A solução encontrada foi o uso de medidas de dissimilaridade para escolher os sinais de sensores com maior potencial discriminativo. Os resultados do estudo mostraram que os dados associados ao classificadores mais precisos foram aqueles com as maiores razões entre as dissimilaridades interclasse e intraclasse, considerando a medida de dissimilaridade utilizada. Tais resultados fazem sentido, uma vez que as razões indicam a diferença entre os dados de classes diferentes em relação aos dados de uma mesma classe. Este tipo de análise permite a seleção de dados com maior chance de treinar classificadores precisos, permitindo também o desenvolvimento de modelos de classificação menos complexos e menos custosos computacionalmente. A menor complexidade resulta da prescindibilidade de se utilizar estruturas capazes de fundir a informação oriunda de diferentes modalidades de dados, enquanto o menor custo computacional vem do uso

de menos informações para classificar exemplos, permitindo a implementação dos sistemas de classificação em plataformas de *hardware* mais simples e baratas.

A segunda etapa, por sua vez, envolveu uma avaliação do uso de classificadores treinados de forma supervisionada, *e.g.* CNNs, para a diagnose de falhas. Foram utilizados os dados pertencentes a todas as classes disponíveis na base de dados, normais e anômalas. Nesta etapa, foram realizadas diversas simulações com diferentes configurações de redes neurais convolucionais, bem como diferentes formas de organização da base de dados. Os experimentos objetivaram promover um melhor entendimento sobre a base de dados e sobre a influência de alguns parâmetros, como a quantidade de camadas, no desempenho de modelos de aprendizado profundo.

O terceiro estudo foi sobre o uso de estágios de decisão, *e.g.* SVMs, na melhoria dos resultados do classificador treinado supervisionadamente. Viu-se que é bastante comum em abordagens da literatura o uso da saída de maior probabilidade para determinar a classe da instância testada. A informação presente nas demais saídas, porém, é ignorada. O uso do estágio de decisão visa utilizar a informação de todas as saídas do classificador para melhorar a acurácia do processo de classificação. Experimentos realizados com CNNs mostraram que o uso de um estágio de decisão foi capaz de melhorar o desempenho dos sistemas de classificação, mesmo com processos de treinamento 80% mais curtos. Modelos que utilizaram SVMs como estágio de decisão, treinados por 10 épocas, melhoraram a acurácia em aproximadamente 0,76%, se comparados ao mesmos modelos treinados por 50 épocas e sem estágio de decisão. Se a comparação for feita com modelos sem estágio de decisão e treinados por 10 épocas, o crescimento foi de aproximadamente 2,56%. O aumento dos tempos de treinamento e execução causado pela adição do estágio de decisão foi de 1 s e 0,001 s, respectivamente, sendo um custo considerado baixo para as melhorias proporcionadas pelo uso deste artifício. Outro algoritmo, *e.g.* MLP, também foi utilizado como estágio de decisão, obtendo resultados semelhantes aos da SVM.

Na quarta etapa, começou-se a abordar o treinamento dos detectores de anomalias com dados pertencentes apenas à classe normal. Ela focou num sistema híbrido para detectar anomalias, composto por um extrator de características baseado em aprendizado profundo, *e.g.* CNN, e por um algoritmo detector de anomalias propriamente dito, *e.g.* OC-SVM. Este último utiliza as características extraídas da informação de entrada pela CNN para realizar a detecção. Nesta abordagem, o treinamento de ambos os algoritmos foi realizado de forma independente. A CNN foi treinada como um regressor, apenas com dados pertencentes à classe normal, buscando aprender uma distribuição de valores aleatoriamente gerados e previamente definidos. O objetivo foi fazer com que as características relativas aos dados normais ficassem próximas àquelas da distribuição utilizada no treinamento do modelo. Enquanto isso, esperava-se que as características relativas aos dados anômalos fossem diferentes, facilitando o estabelecimento de uma fronteira de decisão pelo OC-SVM. Os resultados mostraram que o modelo proposto foi capaz de fornecer resultados médios de acurácia próximos a 95%, superando os resultados de técnicas da literatura

como o *autoencoder* e outros modelos híbridos.

O uso dos valores gerados de forma aleatória se baseou no seguinte raciocínio: como não há informações sobre instâncias anômalas, ou mesmo sobre onde estão localizadas no espaço de características, a região para onde as características extraídas são mapeadas não importaria. O importante seria que as características de instâncias normais apresentassem um comportamento em comum, resultante do processo de treinamento do extrator de características, e que as diferenciasse de instâncias anômalas.

O estudo seguinte objetivou a melhoria do sistema proposto na quarta etapa. Para isso, buscou-se uma forma de integrar o processo de treinamento do extrator de características com a seleção de protótipos, de modo a otimizar o aprendizado do extrator para a detecção de anomalias. O papel da seleção de protótipos neste processo foi concentrar as características extraídas de instâncias normais numa determinada região do espaço de características. O algoritmo que realizou a detecção de anomalias foi baseado na técnica de vizinhos mais próximos, dado que este também foi o algoritmo utilizado na seleção de protótipos. Esta foi uma forma de tornar o treinamento do extrator mais próximo do processo de detecção de anomalias, *i.e.*, orientar o processo de treinamento do extrator de acordo com as características do detector. Os resultados obtidos com esta nova abordagem foram superiores àqueles obtidos na etapa anterior. Três bases de dados foram utilizadas nesta análise, sendo os valores de acurácia obtidos nas três bases superiores a 95%, considerando os dados normais, e próximos a 100% para os dados anômalos.

A forma de treinamento desenvolvida neste estudo difere de técnicas da literatura por utilizar a inicialização aleatória dos pesos do extrator de características no posicionamento das características extraídas pelo mesmo. A partir da inicialização dos pesos, a movimentação das características de instâncias normais se dá com o objetivo de concentrá-las no espaço de características, com o auxílio de uma técnica de seleção de protótipos. Este processo funciona como um aprimoramento do mapeamento inicial do extrator, visando uma padronização do comportamento das instâncias utilizadas no treinamento do extrator. Outras técnicas, como o *autoencoder*, posicionam estas características objetivando reduzir erros de reconstrução, de predição, etc. Em outras palavras, são objetivos que não estão diretamente relacionados à detecção de anomalias, e não visam facilitar o processo de separação de instâncias normais e anômalas de um conjunto de dados.

O último estudo desenvolvido na tese buscou aprofundar a análise do método proposto no Capítulo 7. Esta análise envolveu o uso de outras bases de dados, bem como a comparação com resultados de técnicas da literatura. A partir do estudo, foi possível observar melhor o desempenho da técnica proposta em diferentes problemas de detecção de anomalias, de modo a identificar o seus pontos positivos, bem como as suas limitações.

Com o auxílio do t-SNE, notou-se que o desempenho da técnica foi superior quando as características de instâncias anômalas estavam fora ou na periferia da região que continha as características de instâncias normais. Quando elas estavam em meio às instâncias normais, a

técnica apresentou dificuldades para realizar a separação. Isto pode ser explicado pelo processo de seleção de protótipos utilizado, que foi baseado no algoritmo de vizinhos mais próximos, e que considerou toda a distribuição de dados normais. Quando as características de instâncias anômalas estavam em meio às de instâncias normais, elas eram indiretamente afetadas pelo processo envolvendo as características normais. Por outro lado, estando as instâncias anômalas na periferia ou fora da região povoada por instâncias normais, o processo de seleção de protótipos contribuiu para aumentar a separação. Isto explica os valores de AUC próximos a 1 em três bases de dados, *i.e.*, Caixa de Engrenagens, Compressor (falhas em mancais) e Compressor (Falhas combinadas). Ao mesmo tempo, explica o desempenho mais fraco obtido nas bases Ventilador, Válvulas e Sistema de deslizamento linear. Ainda assim, considerando as técnicas da literatura que utilizaram apenas os dados normais para treinar os detectores de anomalias, os resultados obtidos com o método proposto foram promissores, estando entre os melhores resultados alcançados em cada base.

9.1 TRABALHOS FUTUROS

Em trabalhos futuros, pretende-se aprimorar o processo de detecção de anomalias desenvolvido ao longo da tese. Como foi observado, existem algumas limitações que precisam ser corrigidas, de modo a adequar este processo a uma gama maior de problemas. Um exemplo de ponto a ser trabalhado é a inserção de uma etapa de agrupamento das características de instâncias normais, antes mesmo de iniciar o processo de treinamento do extrator. Segmentar a distribuição de características normais pode resultar em distribuições menores sem a mistura de instâncias normais e anômalas. Isto possibilitaria realizar o processo de seleção de protótipos em cada uma destas distribuições menores de instâncias normais, evitando a influência indireta sobre instâncias anômalas.

Outro aspecto a ser trabalhado é o uso de técnicas alternativas de seleção de protótipos. Neste trabalho, a seleção de protótipos baseou-se em vizinhos mais próximos, de modo a estabelecer uma relação mais estreita com o detector de anomalias, também baseado em vizinhos mais próximos. Em contrapartida, existem outras técnicas de seleção de protótipos, bem como outros detectores de anomalias, que poderiam ser investigados com o intuito de melhorar o desempenho do método proposto. O uso de *transfer learning* também é cogitado como trabalho futuro. Extratores de características treinados com outras bases de dados poderiam ser utilizados como ponto de partida para o processo de treinamento com o seletor de protótipos, em vez do extrator inicializado aleatoriamente. Isto poderia tornar o processo de treinamento mais rápido e eficaz.

9.2 ARTIGOS RELACIONADOS À TESE

Nesta seção, estão listados os artigos submetidos ao longo do desenvolvimento deste trabalho e relacionados ao assunto discutido.

Trabalhos aceitos em periódicos

- MONTEIRO, R. P.; CERRADA, M.; CABRERA, D. R.; SÁNCHEZ, R. V. BASTOS-FILHO, C. J. Using a Support Vector Machine-based Decision Stage to improve the Fault Diagnosis on Gearboxes. *Computational Intelligence and Neuroscience*, 2019.
- MONTEIRO, R. P.; BASTOS-FILHO, C. J. A. Using the Kullback-Leibler Divergence and Kolmogorov-Smirnov Test to Select Input Sizes to the Fault Diagnosis Problem Based on a CNN Model. *Learning and NonLinear Models.*, V. 18, N. 2, 2020.

Trabalhos submetidos em periódicos

- MONTEIRO, R. P.; CERRADA, M.; CABRERA, D. R.; SÁNCHEZ, R. V. BASTOS-FILHO, C. J. A. Hybrid Nearest Neighbors-Based Deep Learning Approach for Anomaly Detection in Industrial Machines. *IEEE Transactions on Industrial Informatics*, 2020.

Trabalhos aceitos em conferências

- MONTEIRO, R. P.; CERRADA, M.; CABRERA, D. R.; SÁNCHEZ, R. V. BASTOS-FILHO, C. J. A. Convolutional neural networks using fourier transform spectrogram to classify the severity of gear tooth breakage. *2018 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)*, August 15-17, 2018, Xi'an, China.
- MONTEIRO, R. P.; BASTOS-FILHO, C. J. A. Feature Extraction Using Convolutional Neural Networks for Anomaly Detection. *CBIC 2019: XIV Congresso Brasileiro de Inteligência Computacional*, Nov 3-6, 2019, Belém, Brazil.
- MONTEIRO, R. P.; BASTOS-FILHO, C. J. A. Detecting defects in sanitary wares using deep learning. In: *2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*. IEEE, 2019. p. 1-6.
- LAPENDA, L. V. N.; MONTEIRO, R. P.; BASTOS-FILHO, C. J. A. Autoencoder latent space: an empirical study. In: *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2020. p. 2453-2460.

Capítulo de livro

- MONTEIRO, R. P.; BASTOS-FILHO, C. J. A. Using Kullback-Leibler Divergence to Identify Prominent Sensor Data for Fault Diagnosis. In: *International Conference on Intelligent Data Engineering and Automated Learning.* Springer, Cham, 2020. p. 136-147.

9.3 ARTIGOS NÃO RELACIONADOS À TESE

Nesta seção, estão listados os artigos submetidos ao longo do desenvolvimento deste trabalho, mas não relacionados ao assunto discutido.

Trabalhos aceitos em conferências

- LIMA, G. A.; MONTEIRO, R. P.; ROCHA, P.; LINS, A. J.; Bastos-Filho, C. J. A. Mild Cognitive Impairment Diagnosis and Detecting Possible Labeling Errors in Alzheimer's Disease with an Unsupervised Learning-based Approach. In: *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2020. p. 2905-2911.
- VERÇOSA, L. F.; LIRA, R.; MONTEIRO, R., SILVA, K.; MAGALHÃES, J.; MACIEL, A.; LEITE, B.; BASTOS-FILHO, C. J. A. Impact of Unusual Features in Credit Scoring Problem. In: *Anais do VIII Symposium on Knowledge Discovery, Mining and Learning*. SBC, 2020. p. 81-88.
- MELO, R.; Monteiro, R. P.; OLIVEIRA, J. P. G.; JERONIMO, B.; Bastos-Filho, C. J. A.; ALBUQUERQUE, A. P.; KELNER, J. Guitar Tuner and Song Performance Evaluation Using a NAO robot. In: *2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE)*. IEEE, 2020. p. 1-6.

REFERÊNCIAS

AKCAY, S.; ATAPOUR-ABARGHOUEI, A.; BRECKON, T. P. Ganomaly: Semi-supervised anomaly detection via adversarial training. In: SPRINGER. **Asian conference on computer vision**. Perth, Australia, 2018. p. 622–637.

BACK, A. D.; TRAPPENBERG, T. P. Input variable selection using independent component analysis. In: IEEE. **IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339)**. Washington, DC, USA, 1999. v. 2, p. 989–992.

BAY, H. et al. Speeded-up robust features (surf). **Computer vision and image understanding**, Elsevier, v. 110, n. 3, p. 346–359, 2008.

BIJMA, F.; JONKER, M.; VAART, A. Van der. **An introduction to mathematical statistics**. Amsterdam, Países Baixos: Amsterdam University Press, 2017.

BISHOP, C. M. et al. **Neural networks for pattern recognition**. Oxford, UK: Oxford university press, 1995.

BORDA, M. **Fundamentals in information theory and coding**. Berlim, Alemanha: Springer Science & Business Media, 2011.

BREUNIG, M. M. et al. Lof: identifying density-based local outliers. In: ACM. **ACM sigmod record**. Nova York, NY, EUA, 2000. v. 29, n. 2, p. 93–104.

CABRERA, D. et al. Bayesian approach and time series dimensionality reduction to lstm-based model-building for fault diagnosis of a reciprocating compressor. **Neurocomputing**, Elsevier, v. 380, p. 51–66, 2020.

CABRERA, D. et al. Generative adversarial networks selection approach for extremely imbalanced fault diagnosis of reciprocating machinery. **IEEE Access**, IEEE, v. 7, p. 70643–70653, 2019.

CARNEIRO, T. et al. Performance analysis of google colaboratory as a tool for accelerating deep learning applications. **IEEE Access**, IEEE, v. 6, p. 61677–61685, 2018.

CHALAPATHY, R.; CHAWLA, S. Deep learning for anomaly detection: A survey. **arXiv preprint arXiv:1901.03407**, 2019.

CHALAPATHY, R.; MENON, A. K.; CHAWLA, S. Anomaly detection using one-class neural networks. **arXiv preprint arXiv:1802.06360**, 2018.

CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. **ACM computing surveys (CSUR)**, ACM, v. 41, n. 3, p. 15, 2009.

CHINNIAH, Y. Analysis and prevention of serious and fatal accidents related to moving parts of machinery. **Safety science**, Elsevier, v. 75, p. 163–173, 2015.

CHUI, C. K. **An introduction to wavelets**. Massachusets, EUA: Elsevier, 2016.

DEFREITAS, A. et al. Improved anomaly detection in experimental wind tunnel data using pca. In: **AIAA Scitech 2020 Forum**. Orlando, FL, EUA: AIAA, 2020. p. 1198.

DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. **2009 IEEE conference on computer vision and pattern recognition**. Miami, FL, EUA, 2009. p. 248–255.

DOKUZ, A. Ş.; ÇELIK, M.; ECEMIŞ, A. Anomaly detection in bitcoin prices using dbscan algorithm. p. 436–443, 2020.

DURKOTA, K. et al. **NEURONNET: Siamese Network for Anomaly Detection**. Tokyo, Japão, 2020.

ERIKSSON, D.; FRISK, E.; KRYSANDER, M. A method for quantitative fault diagnosability analysis of stochastic linear descriptor models. **Automatica**, Elsevier, v. 49, n. 6, p. 1591–1600, 2013.

ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: **Kdd**. Portland, Oregon, EUA: AAAI Press, 1996. v. 96, n. 34, p. 226–231.

FRIEDMAN, N.; GEIGER, D.; GOLDSZMIDT, M. Bayesian network classifiers. **Machine learning**, Springer, v. 29, n. 2-3, p. 131–163, 1997.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: JMLR WORKSHOP AND CONFERENCE PROCEEDINGS. **Proceedings of the thirteenth international conference on artificial intelligence and statistics**. Sardinia, Italy, 2010. p. 249–256.

GOLAN, I.; EL-YANIV, R. Deep anomaly detection using geometric transformations. In: **Advances in Neural Information Processing Systems**. Montréal, Canadá: NIPS, 2018. p. 9758–9769.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. Massachusets, EUA: MIT press, 2016.

GOODFELLOW, I. et al. Generative adversarial nets. In: **Advances in neural information processing systems**. Montreal, Quebec, Canada: NPIS, 2014. p. 2672–2680.

GU, P.; CHOW, M.; SHAO, S. Outlier detection based on sparse coding and neighbor entropy in high-dimensional space. In: **Proceedings of the 17th ACM International Conference on Computing Frontiers**. Catania, Sicília, Itália: ACM, 2020. p. 202–207.

HASHEMIAN, H. M. State-of-the-art predictive maintenance techniques. **IEEE Transactions on Instrumentation and measurement**, IEEE, v. 60, n. 1, p. 226–236, 2010.

HE, J. et al. Investigation of a multi-sensor data fusion technique for the fault diagnosis of gearboxes. **Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science**, SAGE Publications Sage UK: London, England, v. 233, n. 13, p. 4764–4775, 2019.

HE, K. et al. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. Las Vegas, NV, EUA: IEEE, 2016. p. 770–778.

HSU, C.-W. et al. A practical guide to support vector classification. Taipei, 2003.

IANDOLA, F. N. et al. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. **arXiv preprint arXiv:1602.07360**, 2016.

ISLAM, Q. N. **Mastering PyCharm**. Birmingham, UK: Packt Publishing Ltd, 2015.

JALALI, A.; SCHINDLER, A.; HASLHOFER, B. Dcase challenge 2020: Unsupervised anomalous sound detection of machinery with deep autoencoders.

JIANG, G. et al. Stacked multilevel-denoising autoencoders: A new representation learning approach for wind turbine gearbox fault diagnosis. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 66, n. 9, p. 2391–2402, 2017.

JIANG, Y.; WANG, W.; ZHAO, C. A machine vision-based realtime anomaly detection method for industrial products using deep learning. In: IEEE. **2019 Chinese Automation Congress (CAC)**. [S.l.], 2019. p. 4842–4847.

JING, L. et al. An adaptive multi-sensor data fusion method based on deep convolutional neural networks for fault diagnosis of planetary gearbox. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 17, n. 2, p. 414, 2017.

KAY, S. M. **Fundamentals of statistical signal processing**. Upper Saddle River, NJ, EUA: Prentice Hall PTR, 1993.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014.

KOCH, G.; ZEMEL, R.; SALAKHUTDINOV, R. Siamese neural networks for one-shot image recognition. In: LILLE. **ICML deep learning workshop**. Lille, France, 2015. v. 2.

KOIZUMI, Y. et al. Description and discussion on dcase2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring. **arXiv preprint arXiv:2006.05822**, 2020.

KOIZUMI, Y. et al. Unsupervised detection of anomalous sound based on deep learning and the neyman–pearson lemma. **IEEE/ACM Transactions on Audio, Speech, and Language Processing**, IEEE, v. 27, n. 1, p. 212–224, 2018.

KOUTINI, K.; EGHBAL-ZADEH, H.; WIDMER, G. Receptive-field-regularized cnn variants for acoustic scene classification. **arXiv preprint arXiv:1909.02859**, 2019.

KRIZHEVSKY, A.; NAIR, V.; HINTON, G. Cifar-10 (canadian institute for advanced research). Acesso: 2021-12-31. Disponível em: <http://www.cs.toronto.edu/~kriz/cifar.html>.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. Nova York, NY, Eua: ACM, 2012. p. 1097–1105.

KULLBACK, S. **Information theory and statistics**. Chelmsford, Massachusetts, EUA: Courier Corporation, 1997.

KUREMOTO, T. et al. Time series forecasting using a deep belief network with restricted boltzmann machines. **Neurocomputing**, Elsevier, v. 137, p. 47–56, 2014.

LECUN, Y.; CORTES, C. MNIST handwritten digit database. 2010. Acesso: 2021-01-31. Disponível em: <http://yann.lecun.com/exdb/mnist/>.

LI, C. et al. Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis. **Neurocomputing**, Elsevier, v. 168, p. 119–127, 2015.

LI, D. et al. Anomaly detection with generative adversarial networks for multivariate time series. **arXiv preprint arXiv:1809.04758**, 2018.

LIAO, Y.; ZENG, X.; LI, W. Wavelet transform based convolutional neural network for gearbox fault classification. In: IEEE. **2017 Prognostics and System Health Management Conference (PHM-Harbin)**. Harbin, China, 2017. p. 1–6.

LIM, S. K. et al. Doping: Generative data augmentation for unsupervised anomaly detection with gan. In: IEEE. **2018 IEEE International Conference on Data Mining (ICDM)**. Singapura, 2018. p. 1122–1127.

LIU, R. et al. Artificial intelligence for fault diagnosis of rotating machinery: A review. **Mechanical Systems and Signal Processing**, Elsevier, v. 108, p. 33–47, 2018.

LLOYD, S. Least squares quantization in pcm. **IEEE transactions on information theory**, IEEE, v. 28, n. 2, p. 129–137, 1982.

LUO, D.; LU, J.; GUO, G. Road anomaly detection through deep learning approaches. **IEEE Access**, IEEE, v. 8, p. 117390–117404, 2020.

MAATEN, L. Van der; HINTON, G. Visualizing data using t-sne. **Journal of machine learning research**, v. 9, n. 11, 2008.

MCCULLAGH, P. What is a statistical model? **Annals of statistics**, JSTOR, p. 1225–1267, 2002.

MIMII Dataset: Sound Dataset for Malfunctioning Industrial Machine Investigation and Inspection. Acesso: 2021-12-31. Disponível em: <https://zenodo.org/record/3384388>.

MONTEIRO, R. et al. Convolutional neural networks using fourier transform spectrogram to classify the severity of gear tooth breakage. In: IEEE. **2018 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)**. Xi'an, China, 2018. p. 490–496.

MONTEIRO, R. P.; BASTOS-FILHO, C. J. Detecting defects in sanitary wares using deep learning. In: IEEE. **2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI)**. Guayaquil, Ecuador, 2019. p. 1–6.

MONTEIRO, R. P.; BASTOS-FILHO, C. J. Feature extraction using convolutional neural networks for anomaly detection. In: BRAZILIAN SOCIETY OF COMPUTATIONAL INTELLIGENCE (SBIC). **CBIC 2019: XIV Congresso Brasileiro de Inteligência Computacional**. Belem, PA, Brasil, 2019. p. –.

MONTEIRO, R. P. et al. Using a support vector machine based decision stage to improve the fault diagnosis on gearboxes. **Computational intelligence and neuroscience**, Hindawi, v. 2019, 2019.

MÜLLER, R. et al. Acoustic anomaly detection for machine sounds based on image transfer learning. **arXiv preprint arXiv:2006.03429**, 2020.

MULONGO, J. et al. Anomaly detection in power generation plants using machine learning and neural networks. **Applied Artificial Intelligence**, Taylor & Francis, v. 34, n. 1, p. 64–79, 2020.

MUNIR, M. et al. Deepant: A deep learning approach for unsupervised anomaly detection in time series. **IEEE Access**, IEEE, v. 7, p. 1991–2005, 2018.

MUNOZ-ORGANERO, M. Outlier detection in wearable sensor data for human activity recognition (har) based on drnns. **IEEE Access**, IEEE, v. 7, p. 74422–74436, 2019.

NAKAZAWA, T.; KULKARNI, D. V. Anomaly detection and segmentation for wafer defect patterns using deep convolutional encoder–decoder neural network architectures in semiconductor manufacturing. **IEEE Transactions on Semiconductor Manufacturing**, IEEE, v. 32, n. 2, p. 250–256, 2019.

NEUHAUSER, M. **Nonparametric statistical tests: A computational approach**. Boca Raton, FL, EUA: CRC Press, 2011.

NGUYEN-AN, H. et al. Generating iot traffic: A case study on anomaly detection. In: IEEE. **2020 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN**. [S.l.], 2020. p. 1–6.

OORD, A. v. d. et al. Wavenet: A generative model for raw audio. **arXiv preprint arXiv:1609.03499**, 2016.

OPPENHEIM, A. V.; SCHAFER, R. W.; BACK, H. R. **"Discrete Time Signal Processing", Pearson Education, 2005.** [S.l.]: BHARATHIDASAN ENGINEERING COLLEGE, 2015.

OUARDINI, K. et al. Towards practical unsupervised anomaly detection on retinal images. In: **Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data**. [S.l.]: Springer, 2019. p. 225–234.

OZA, P.; PATEL, V. M. One-class convolutional neural network. **IEEE Signal Processing Letters**, IEEE, v. 26, n. 2, p. 277–281, 2018.

PARK, S. H.; PARK, H. J.; CHOI, Y.-J. Rnn-based prediction for network intrusion detection. In: IEEE. **2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)**. [S.l.], 2020. p. 572–574.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in python. **the Journal of machine Learning research**, JMLR. org, v. 12, p. 2825–2830, 2011.

PEREIRA, J.; SILVEIRA, M. Learning representations from healthcare time series data for unsupervised anomaly detection. In: IEEE. **2019 IEEE International Conference on Big Data and Smart Computing (BigComp)**. [S.l.], 2019. p. 1–7.

PRIMUS, P. **Reframing unsupervised machine condition monitoring as a supervised classification task with outlier-exposed classifiers**. [S.l.], 2020.

PUROHIT, H. et al. Mimii dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. **arXiv preprint arXiv:1909.09347**, 2019.

RAMASWAMY, S.; RASTOGI, R.; SHIM, K. Efficient algorithms for mining outliers from large data sets. In: ACM. **ACM Sigmod Record**. [S.l.], 2000. v. 29, n. 2, p. 427–438.

RAO, K. R.; KIM, D. N.; HWANG, J. J. **Fast Fourier transform-algorithms and applications**. [S.l.]: Springer Science & Business Media, 2011.

RAWAT, W.; WANG, Z. Deep convolutional neural networks for image classification: A comprehensive review. **Neural computation**, MIT Press, v. 29, n. 9, p. 2352–2449, 2017.

RAZA, M.; QAYYUM, U. Classical and deep learning classifiers for anomaly detection. In: IEEE. **2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)**. [S.l.], 2019. p. 614–618.

REDACAO, M. E. **QUAL É O LIMITE MÁXIMO E MÍNIMO PARA BATIMENTOS CARDÍACOS?** [S.l.]: Mundo Estranho, 2019. <https://super.abril.com.br/mundo-estranho/qual-e-o-limite-maximo-e-o-minimo-para-os-batimentos-cardiacos/>. Acesso: 2019-10-29.

ROSSUM, G. V.; DRAKE, F. L. **The python language reference manual**. [S.l.]: Network Theory Ltd., 2011.

RUSHE, E.; NAMEE, B. M. Anomaly detection in raw audio using deep autoregressive networks. In: IEEE. **ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2019. p. 3597–3601.

RUSHE, E.; NAMEE, B. M. Deep context-aware novelty detection. **arXiv preprint arXiv:2006.01168**, 2020.

SARMADI, H.; KARAMODIN, A. A novel anomaly detection method based on adaptive mahalanobis-squared distance and one-class knn rule for structural health monitoring under environmental effects. **Mechanical Systems and Signal Processing**, Elsevier, v. 140, p. 106495, 2020.

SCHLEGL, T. et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: SPRINGER. **International conference on information processing in medical imaging**. [S.l.], 2017. p. 146–157.

SHARKEY, A. J.; CHANDROTH, G. O.; SHARKEY, N. E. Acoustic emission, cylinder pressure and vibration: a multisensor approach to robust fault diagnosis. In: IEEE. **Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium**. [S.l.], 2000. v. 6, p. 223–228.

SHERIDAN, K. et al. An application of dbscan clustering for flight anomaly detection during the approach phase. In: **AIAA Scitech 2020 Forum**. [S.l.: s.n.], 2020. p. 1851.

SHYLENDRA, A. et al. Low power unsupervised anomaly detection by nonparametric modeling of sensor statistics. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems**, IEEE, 2020.

SINGH, D.; MOHAN, C. K. Deep spatio-temporal representation for detection of road accidents using stacked autoencoder. **IEEE Transactions on Intelligent Transportation Systems**, IEEE, v. 20, n. 3, p. 879–887, 2018.

SRIVASTAVA, A. et al. Deep learning for detecting diseases in gastrointestinal biopsy images. In: IEEE. **2019 Systems and Information Engineering Design Symposium (SIEDS)**. [S.l.], 2019. p. 1–4.

STALLKAMP, J. et al. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. **Neural Networks**, n. 0, p. –, 2012. ISSN 0893-6080. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0893608012000457>.

TIAN, Q.; LI, J.; LIU, H. A method for guaranteeing wireless communication based on a combination of deep and shallow learning. **IEEE Access**, IEEE, v. 7, p. 38688–38695, 2019.

TULUPTCEVA, N. et al. Anomaly detection with deep perceptual autoencoders. **arXiv preprint arXiv:2006.13265**, 2020.

VANROSSUM, G.; DRAKE, F. L. **The python language reference**. [S.l.]: Python Software Foundation Amsterdam, Netherlands, 2010.

VARTOUNI, A. M.; TESHNEHLAB, M.; KASHI, S. S. Leveraging deep neural networks for anomaly-based web application firewall. **IET Information Security**, IET, v. 13, n. 4, p. 352–361, 2019.

VIJAYAKUMAR, V. et al. Isolation forest and local outlier factor for credit card fraud detection system.

WANG, Z.; ZHOU, Y.; LI, G. Anomaly detection by using streaming k-means and batch k-means. In: IEEE. **2020 5th IEEE International Conference on Big Data Analytics (ICBDA)**. [S.l.], 2020. p. 11–17.

WOLD, S.; ESBENSEN, K.; GELADI, P. Principal component analysis. **Chemometrics and intelligent laboratory systems**, Elsevier, v. 2, n. 1-3, p. 37–52, 1987.

XIA, M. et al. Fault diagnosis for rotating machinery using multiple sensors and convolutional neural networks. **IEEE/ASME Transactions on Mechatronics**, IEEE, v. 23, n. 1, p. 101–110, 2017.

XIANG, B.; LIU, Z.; ZHANG, K. Flagging implausible inspection reports of distribution transformers via anomaly detection. **IEEE Access**, IEEE, v. 8, p. 75798–75808, 2020.

XU, S.; WU, H.; BIE, R. Cxnet-m1: Anomaly detection on chest x-rays with image-based deep learning. **IEEE Access**, IEEE, v. 7, p. 4466–4477, 2018.

YOUNG, T. et al. Recent trends in deep learning based natural language processing. **ieee Computational intelligenCe magazine**, IEEE, v. 13, n. 3, p. 55–75, 2018.

YU, S. et al. Hyperspectral anomaly detection based on low-rank representation using local outlier factor. **IEEE Geoscience and Remote Sensing Letters**, IEEE, 2020.

ZENATI, H. et al. Efficient gan-based anomaly detection. **arXiv preprint arXiv:1802.06222**, 2018.

ZENG, X.; LIAO, Y.; LI, W. Gearbox fault classification using s-transform and convolutional neural network. In: IEEE. **2016 10th International Conference on Sensing Technology (ICST)**. [S.l.], 2016. p. 1–5.

ZHAO, Y.; NASRULLAH, Z.; LI, Z. Pyod: A python toolbox for scalable outlier detection. **arXiv preprint arXiv:1901.01588**, 2019.