UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

HIAGO HENRIQUE GOMES DE ARAUJO

UMA ABORDAGEM DE REDES NEURAIS NÃO SUPERVISIONADAS PARA A DETECÇÃO DE ANOMALIA DE EQUIPAMENTOS EM CENÁRIOS DE MAU FUNCIONAMENTO DE SENSORES

Recife

2022

HIAGO HENRIQUE GOMES DE ARAUJO

UMA ABORDAGEM DE REDES NEURAIS NÃO SUPERVISIONADAS PARA A DETECÇÃO DE ANOMALIA DE EQUIPAMENTOS EM CENÁRIOS DE MAU FUNCIONAMENTO DE SENSORES

Dissertação apresentada à Universidade Federal de Pernambuco, como parte das exigências do Programa de Pós-Graduação em Engenharia de Produção, para obtenção do título de mestre em Engenharia de Produção.

Áreas de Concentração: Confiabilidade, Pesquisa Operacional.

Orientadora: Isis Didier Lins, DSc.

Recife

2022

Catalogação na Fonte
Bibliotecário Gabriel Luz CRB-4 / 2222

A663a Araújo, Hiago Henrique Gomes de.
Uma abordagem de redes neurais não supervisionadas para a detecção de anomalia de equipamentos em cenários de mau funcionamento de sensores / Hiago Henrique Gomes de Araújo. 2022.
78 f: figs., tabs.

Orientadora: Profa. Dra. Isis Didier Lins.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia de Produção, Recife, 2022.
Inclui referências.

1. Engenharia de Produção. 2. Gestão da saúde e prognóstico de ativos. 3. Manutenção baseada na condição. 4. Detecção de anomalia. 5. Análise de vibração. 6. Aprendizagem profunda. 7. Redes neurais. I. Lins, Isis Didier (Orientadora). II. Título.

UFPE

658.5 CDD (22. ed.) BCTG / 2022 - 411

HIAGO HENRIQUE GOMES DE ARAUJO

UMA ABORDAGEM DE REDES NEURAIS NÃO SUPERVISIONADAS PARA A DETECÇÃO DE ANOMALIA DE EQUIPAMENTOS EM CENÁRIOS DE MAU FUNCIONAMENTO DE SENSORES

Dissertação apresentada à Universidade Federal de Pernambuco, como parte das exigências do Programa de Pós-Graduação em Engenharia de Produção, para obtenção do título de mestre em Engenharia de Produção.

Aprovada em: 16/08/2022.

BANCA EXAMINADORA

_____

Prof/ª. Drª. Isis Didier Lins (Orientador)
Universidade Federal de Pernambuco

_____

Prof. Dr. Márcio José das Chagas Moura (Examinador Interno)
Universidade Federal de Pernambuco

_____

Prof. Dr. Erick Giovani Sperandio Nascimento(Examinador Externo)
Senai CIMATEC

A meus pais e a todas as pessoas
cujos mundos me permitiram conhecer

AGRADECIMENTOS

A entrega e defesa desse documento representa o fechamento de um grande ciclo em minha vida acadêmica e sumariza um processo de 8 anos de vínculo e história com a Universidade Federal de Pernambuco. Portanto, os meus primeiros agradecimentos são direcionados aos meus pais e meu irmão que sempre viabilizaram e icentivaram minha educação e nunca deixaram de acreditar em mim.

Agradeço igualmente: A todos outros famílias, especialmente Tia Cristiane, Tia Márcia e Vovó da paz por toda disponibilidade e por sempre estarem com os braços, colos, lares e escuta abertos para acolhimento e segurança não só nos momentos felizes, mas também nos mais difíceis.

À minhas primas Amanda, Anne, Camilla e Natália por todos momentos de alegrias e conversas carinhosas.

Às irmãs Thaís e Iris Lucas por serem uma família que o destino me presenteou e por estarem sempre presentes como pessoas incríveis que tive oportunidade de conhecer e me inspirar, além de todos momentos de risadas, paciência e apoio;

À Professora Ísis por ser inspiração profissional e pessoal e por todo apoio na execução desse trabalho e outros. Além de toda compreensão e escuta nos momentos de dificuldade.

Ao Professor Márcio por todo apoio e confiança em diversos trabalhos implementados ao longo de todos esses anos de UFPE.

A todos os colegas do CEERMA em especial a Mona, Thaís e Marcela, por toda paciência, amizade e parcerias na execução de trabalhos acadêmicos;

A todos grandes amigos que pude fazer nessa trilha dos quais listo alguns, entre muitos outros igualmente importantes, que me recordo: Aguiar, Andréia, Calife, Carol Leal, Carol Pereira, Chico, Gabi, Ícaro, Isadora, Letícia, Luisa, Rafael, Sandy, Tatá, Tay e Xu, por todo carinho, empatia e aprendizado;

A Arthur, Caio, Dudu, Felipe, Leo, Lorran, Vicente e Yuri por toda diversão, momentos de risadas, conversas de apoio e por me ensinar que a distância não impede a construção de grandes laços;

A todos professores do departamento de engenharia de produção por desempenharem a função de difundir conhecimento.

Carinhosamente,

Hiago Araujo

RESUMO

Os avanços em tecnologias de automação, e avanços nos maquinários de produção tornam os custos de manutenção cada vez mais relevantes em relação aos custos totais nos sistemas de manufatura. Nesse contexto, o desenvolvimento e a redução do custo de implementação de sensores permite a estruturação de sistemas de informação capazes de coletar, armazenar e monitorar evidências acerca da saúde de equipamentos. Em um contexto de Manutenção Baseada na Condição (CBM), o processamento e a análise dos dados provenientes do monitoramento podem direcionar as políticas de manutenção. Neste trabalho, aborda-se a atividade de detecção de anomalia, presente na CBM, em que se identifica se o equipamento está prestes a falhar ou não, sem adentrar nas classificações das causas e modos de falha que levam ao comportamento anômalo. A detecção de anomalia pode ser facilitada por meio de algoritmos de aprendizagem de máquina. Esses modelos podem ser categorizados entre supervisionados e não supervisionados. No primeiro caso, é necessário que os dados (e.g., vibração, temperatura, pressão) estejam associados a uma informação binária ou categórica do funcionamento do equipamento. No entanto, em muitos contextos de manutenção e de sistemas de monitoramento de equipamentos industriais, os dados não são disponibilizados com os valores desses rótulos, ou não estão inteiramente rotulados, ou há dados insuficientes do equipamento em estado falho ou defeituoso. Nesses casos, os algoritmos não supervisionados ou semi-supervisionados podem ser usados como alternativas, visto que não necessitam de uma ‘grande quantidade de dados previamente rotulados. O mau funcionamento de sensores é um problema recorrente em contextos reais de manutenção baseada na condição. Surge então a necessidade da construção de modelos de detecção robusta de anomalia, capazes de detectar falhas mesmo na presença de problemas nos dados provenientes do monitoramento, tais como dados faltantes e presença de ruídos. Propõe-se uma abordagem envolvendo redes adversariais e autoencoders variacionais para detecção robusta e semi-supervisionada de anomalia. Foram propostas quatro categorias distintas de estruturas de redes neurais: Autoencoder, Autoencoder Variacional, Autoencoder Adversarial e Redes Generativas Adversariais. Os modelos propostos são testados, validados e comparados a partir de três bases de dados com sinais de vibração comumente usada como benchmarking na literatura, os resultados mostram que os modelos de Autoencoder Variacional apresentam os melhores resultados para os modelos em que há uma menor presença de dados faltantes e lidam melhor com os casos de ruído inserido na base de dados.

Palavras-chave: gestão da saúde e prognóstico de ativos; manutenção baseada na condição; detecção de anomalia; análise de vibração; aprendizagem profunda; redes neurais.

ABSTRACT

Advances in automation technologies, and advances in production machinery make maintenance costs increasingly relevant in relation to total costs in manufacturing systems. In this context, the technological upgrade and cost reduction of sensors structure, enables the development of information systems capable of collecting, storing, and monitoring evidence about the health of equipment. In a Condition Based Maintenance (CBM) context, the processing and analysis of the monitoring data can guide maintenance policies. In this paper, we address the anomaly detection activity, present in CBM, in which we identify whether the equipment is about to fail or not, without going into the classification of causes and failure modes that lead to anomalous behavior. Anomaly detection can be facilitated by means of machine learning algorithms. These models can be categorized into supervised and unsupervised.In the first case, data (e.g., vibration, temperature, pressure) is required to be associated with binary or categorical information about the health state of the equipment. However, in maintenance and industrial equipment monitoring system contexts, its usual that the data is not available with the values of these labels, or is not fully labeled, or there is insufficient data from the equipment in a faulty state. In these cases, unsupervised or semi-supervised algorithms can be used as alternatives, since they do not require a large amount of pre-labeled data. Besides the problem related to the amount of data labeled as faulty, sensor malfunction is a recurring problem in real-life condition-based maintenance contexts. The need then arises for the construction of robust anomaly detection models capable of detecting faults even in the presence of problems in the data coming from monitoring, such as missing data and the presence of noise. In this work, an approach involving adversarial networks and variational autoencoders is proposed. Four distinct categories of neural network structures have been used: Autoencoder, Variational Autoencoder, Adversarial Autoencoder, and Adversarial Generative Networks. The proposed models are tested, validated and compared from three different datasets with vibration signals commonly used as benchmarking in the literature, the results show that the Variational Autoencoder models present the best results for the models in which there is a lower presence of missing data and handle better the cases of noise inserted in the database. For the cases with higher uncertainty regarding missing data, the Generative Adversarial Network and Variational Autoencoder present superior performance when compared to the other deep learning models tested.

Keywords: anomaly detection*;* prognostic and health management; condition based maintenance; vibration analysis; neural network.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SÍMBOLOS

$\boldsymbol{x}$	Valor de variável independente
$\boldsymbol{y}$	Valor de variável dependente
$\boldsymbol{\hat{y}}$	Previsão do valor da variável dependente
$\boldsymbol{w_b}$-	Peso b de rede neural
$\boldsymbol{p_i}$	Probabilidade do ponto i ser classificado como verdadeiro
$\boldsymbol{W^T}$	Matrizes de pesos de rede neural
$\boldsymbol{J}$	Função perda de rede neural
$\boldsymbol{s_{j,i}}$	Posição da coordenada j da partícula i
$\boldsymbol{\eta}$	Taxa de aprendizagem
$\boldsymbol{m_i}$	Momento na posição i
$\boldsymbol{\beta}$	Taxa de momento
$\boldsymbol{\nabla_\theta J}$	Gradiente da função perda na posição θ
$\boldsymbol{s_i}$	Posição da partícula i
$\boldsymbol{\epsilon}$	Fator de correção para evitar divisão por 0
$\boldsymbol{v_i}$	Velocidade da partícula i
$\boldsymbol{\alpha}$	Taxa de aprendizagem de ADAM
$\boldsymbol{\hat{X}_i}$	Previsão do modelo *autoencoder*
$\boldsymbol{p}$	Probabilidade de dado faltante
$\boldsymbol{\mu}$	Média do ruído inserido no dado

LISTA DE SIGLAS

CBM	Condition-Based Maintenance
PHM	Prognosis and Health Management
AM	Aprendizagem de Máquina
MLP	Multi-Layer Perceptron
GD	Gradient Descent
SGD	Stochastic Gradient Descent
MBGD	Mini-Batch Gradient Descent
MO	Momentum Optimization
NAG	Nesterov Adaptative Gradient
ADAM	Adaptative momentum estimation
CNN	Convoluiional Neural Network
AE	Autoencoder
AAE	Adversarial Autoencoder
GAN	Generative Adversarial Network
VAE	Variational Autoencoder
CWRU	Case Western Reserve University
DE	Drive End
FE	Fan End
d.p.	Desvio Padrão

SUMÁRIO

1 INTRODUÇÃO

Avanços nas tecnologias de automação levaram os sistemas de manutenção a um aumento considerável das atividades de inspeção e reparo de ativos industriais. Essas tarefas são executadas e guiadas por políticas de manutenção, as quais podem ser classificadas em uma abundante gama de possibilidades. Nesse contexto, a redução de custos de sensores permitiu o desenvolvimento de sistemas de informação com autonomia para coletar, tratar, armazenar e monitorar dados referentes a saúde dos equipamentos e seus componentes (por exemplo: aceleração de vibração, temperatura, emissão acústica, corrente, voltagem, entre outros).

A estratégia de manutenção baseada na condição (CBM – do inglês – *Condition Based Maintenance*) sugere o uso desses dados para guiar os processos de planejamento e controle da manutenção, reduzindo-se custos com tarefas desnecessárias de reparos, bem como custos e perdas provenientes das quebras evitadas. Para garantir que as decisões da área de manutenção sejam tomadas baseadas na saúde do equipamento, é necessário um conjunto de modelos e sistemas que garantem a coleta de dados, a detecção de anomalia, o diagnóstico, o prognóstico e o suporte ao decisor. Esse conjunto de tarefas é denominado *Prognostics and Health Management* (PHM) (PECHT; KANG, 2018).

Detecção de anomalia, diagnóstico e prognóstico são tarefas essenciais na implementação de CBM. Primeiramente, é verificado se o sistema analisado opera nas condições normais de saúde, posteriormente, classifica-se o modo e a causa de falha do equipamento e a falha é isolada caso necessário. Já o prognóstico consiste em estimativas acerca do tempo restante de funcionamento do equipamento. Neste trabalho, aborda-se uma proposta de detecção de anomalia.

Na abordagem denominada de *data-driven*, as atividades de PHM são apoiadas por modelos estatísticos de aprendizagem de máquina, no caso da detecção de anomalia, uma grande quantidade de dados correlacionados ao estado de saúde do equipamento é entrada para esses algoritmos, os quais reconhecem e reproduzem padrões entre as métricas avaliadas e a presença ou não da falha. Indicadores de saúde de equipamentos são comumente coletados em uma frequência constante, e comumente mais de uma métrica é coletada ao longo do tempo. Portanto, é importante que as abordagens levem em consideração o caráter temporal e multivariado dos problemas a serem resolvidos.

Modelos de aprendizagem de máquina podem ser categorizados como supervisionados ou não supervisionados. No primeiro caso, os modelos buscam reconhecer padrão na relação de uma variável dependente e um conjunto de variáveis independentes. Nesse contexto, se a variável dependente é um número real, o problema é uma regressão, caso seja binária ou categórica, trata-se de um modelo de

classificação. Além disso, modelos supervisionados, demandam o uso de uma parte do conjunto de dados para a avaliação de desempenho (conjunto de teste) (DUDA; HART; STORK, 2001)

A aplicação de algoritmos supervisionados para detecção de anomalia requer que o dado numérico esteja associado com uma informação binária ou categórica acerca do estado de saúde do equipamento. Para modelos de aprendizagem de máquina, essa informação é costumeiramente repassada como um rótulo, que indica a natureza de cada dado. No entanto, em muitos sistemas, os dados são limitados e não é possível obter uma grande quantidade deles previamente rotulados, ou se identificam poucos dados registrados em um momento de defeito do equipamento (CERRADA; SÁNCHEZ; CABRERA, 2018). Nesses casos, modelos semi-supervisionados ou não supervisionados são uma alternativa viável, dado que é necessário uma quantidade pequena, ou nenhuma, de dados previamente rotulados.

Além da pequena quantidade de dados, problemas com o sensoriamento são comuns no contexto de CBM (LIANSHENG; DATONG; YU, 2017). É necessária a construção de abordagens com robustez suficiente para lidar com esse problema na coleta, nesse documento modelos foram testados em um contexto de múltiplos sensores em que problemas no sensoreamento foram simulados a partir da remoção de dados, ou adição de ruído. Modelos de aprendizagem profunda semi-supervisionada foram testados e comparados utilizando quatro conjuntos de dados de vibração fornecidos abertamente na literatura.

1.1 JUSTIFICATIVA E RELEVÂNCIA

Os custos de manutenção consistem em uma parcela relevante dos custos dos sistemas de produção. A CBM surge como alternativa para tornar essas atividades mais efetivas e menos onerosas, dado que essa política, em contraste às manutenções preventivas programadas em intervalos de tempo constantes, orienta que a execução das atividades seja planejada eficientemente a partir de informações coletadas acerca da saúde dos ativos industriais (AHMAD; KAMARUDDIN, 2012; JARDINE; LIN, D.; BANJEVIC, 2006). Ayo-Imoru e Cilliers (AYO-IMORU; CILLIERS, 2018), a partir de uma pesquisa de revisão de literatura, listam algumas vantagens da implementação da CBM na indústria, dentre elas encontram-se: redução do custo de operação, redução de substituições desnecessárias de equipamentos, redução de atividades emergenciais, e reposição apenas de componentes danificados.

Nesse contexto, o barateamento de sistema de coleta e armazenamento de dados na indústria, bem como os avanços nas tecnologias de computação e processamento de dados, desencadeia uma tendência promissora da aplicação de inteligência artificial como uma ferramenta de PHM

(QUATRINI *et al.*, 2020). É possível encontrar aplicações desses modelos em diversos setores da indústria, como nos setores de: energia eólica (ELASHA *et al.*, 2019), energia nuclear (GOHEL *et al.*, 2020), indústria naval (BERGHOUT *et al.*, 2021; CORADDU *et al.*, 2016; TAN, Y. *et al.*, 2019), transporte ferroviário (CHENARIYAN NAKHAEE *et al.*, 2019; LI, H. *et al.*, 2014), indústria aeroespacial (EID *et al.*, 2021; ZAIDAN *et al.*, 2015), indústria de óleo e gás (ABBASI; LIM; YAM, 2019; MING; PHILIP; SAHLAN, 2019; ORRÙ *et al.*, 2020), indústria automotiva (FERNANDES *et al.*, 2021; THEISSLER *et al.*, 2021), entre outros.

No entanto, para a implementação desses modelos, é necessária a instalação de um sistema capaz de coletar e armazenar uma grande quantidade de dados sobre a saúde dos equipamentos. Devido à complexidade do sensoriamento e dos sistemas de produção, em contextos reais, não é raro observar alguns problemas nos conjuntos de dados. Observam-se, por exemplo, dados faltantes em séries temporais (OMRI *et al.*, 2021a; TSANG *et al.*, 2006). Tan, Sehgal e Sahri (TAN, Y. L.; SEHGAL; SHAHRI, 2005) reafirmam que, assim como dados faltantes, ruído também é uma questão recorrente em redes de sensores. O primeiro caso geralmente ocorre devido a problemas de conexão dos sensores, enquanto os ruídos ocorrem devido à acurácia do equipamento e às condições externas que, ocasionalmente, afetam a coleta.

Devido à essa problemática, não é incomum encontrar, na literatura, estudos envolvendo a proposição de modelos robustos para lidar com esse tipo de problema, em diversos contextos diferentes do PHM apresentado anteriormente. Por exemplo, Monasterio, Burgess e Clifford (2012) aplicam essa abordagem em um contexto da medicina, enquanto outros autores avaliam a robustez de modelos no contexto de detecção de colisões (PAN & MANOCHA, 2017), ou no contexto de visão computacional (BEGGEL, PFEIFFER & BISCHL, 2019).

Outro problema abordado na literatura é referente ao balanceamento dos dados. No contexto industrial, o estado mais comum de um ativo refere-se ao saudável. Ou seja, é recorrente observar uma quantidade muito maior de dados relacionados ao período saudável da máquina do que ao estado de degradação (LIU et al., 2021). É comum também a ausência de dados rotulados, nesses casos o uso de abordagens não supervisionadas é uma possibilidade para viabilizar a aplicação de PHM *Data-Driven* nessas circunstâncias (GOURIVEAU; RAMASSO; ZERHOUNI, Noureddine, 2013). Portanto, observa-se a importância e demanda por modelos de PHM com robustez para lidar com essas questões comuns em redes de sensoriamento, nesse trabalho a abordagem proposta envolve detecção de anomalia, a qual configura um dos principais módulos do PHM (PECHT; KANG, 2018). Essa atividade envolve o processo de avaliar os dados de saúde do equipamento e inferir se aquele dado apresenta um comportamento aquém do considerado saudável.

O desenvolvimento desses modelos está diretamente atrelado à redução dos impactos econômicos ocasionados por falhas e baixa disponibilidade de ativos nos sistemas de produção nos quais a abordagem é implementada. Além disso, alguns estudos relatam que a aplicação de manutenção preditiva está, assim como outras ferramentas avançadas para suporte à decisão na área, diretamente associada com a redução de perdas ambientais e sociais provenientes de práticas inapropriadas de manutenção, emissão de gás carbônico devido a políticas inadequadas e impactos de acidentes ocasionados por quebra de equipamento (KARUPPIAH; SANKARANARAYANAN; ALI, 2021; POLESE *et al.*, 2021).

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo desse trabalho é a proposição de um modelo de detecção de anomalia via *deep learning* com avaliação de robustez para cenários de dados temporais com ruídos e dados faltantes.

1.2.1 Objetivos Específicos

Para se atingir o objetivo geral, os seguintes objetivos específicos são definidos:

- Propor e implementar abordagem de pré-processamento de conjuntos de dados de séries temporais de vibração;
- Implementar as estruturas de Autoencoders, Autoencoders Adversariais, Autoencoders Variacionais e Redes Geradoras Adversárias;
- Implementar abordagens dessas redes neurais semi-supervisionadas para a tarefa de detecção de anomalia de equipamentos;
- Implementar uma abordagem para a inserção artificial de ruídos e de dados faltantes nos dados temporais utilizados para avaliação dos modelos;
- Avaliar o desempenho dos modelos de detecção de anomalia em um contexto não supervisionado com dados faltantes e ruídos.

1.3 ESTRUTURA DA DISSERTAÇÃO

Além desta Introdução, os próximos capítulos desse documento mostram respectivamente:

- A teoria aplicada para a obtenção dos resultados englobando conceitos sobre políticas de manutenção e manutenção baseada na condição, modelos de aprendizagem de máquina, detecção de anomalia, treinamento e arquiteturas de redes neurais, e detalhes sobre a arquitetura utilizada nesse trabalho;
- Revisão de literatura, mostrando panorama de modelos e abordagens propostas nos documentos recentes na área de estudo;
- Uma descrição da metodologia aplicada com detalhes sobre a base de dados utilizada, parâmetros da experimentação, medida de qualidade utilizada para o modelo, construção de cenários de problemas em sensores;
- Apresentação e discussão dos resultados obtidos com a experimentação proposta no capítulo de metodologia;
- Conclusões do estudo e sugestões para trabalhos futuros.

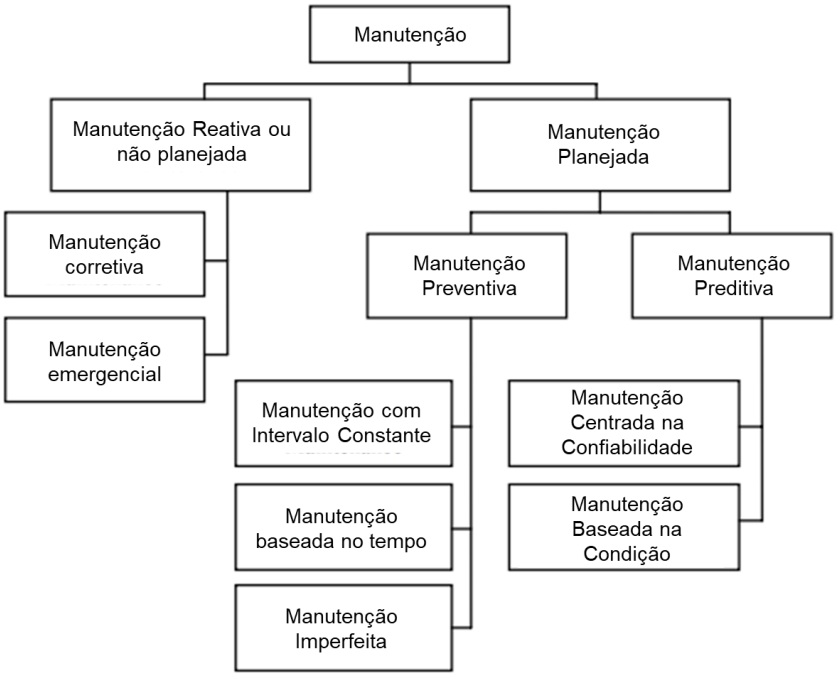
2 FUNDAMENTAÇÃO TEÓRICA

2.1 MANUTENÇÃO BASEADA NA CONDIÇÃO

Em gestão de manutenção, o planejamento da manutenção é orientado pela política de manutenção associada a cada equipamento. A Figura 1 apresenta um diagrama com a classificação das políticas de manutenção, as quais se dividem em manutenção não planejada e manutenção planejada. Desta forma, quando as ações de manutenção só são executadas sobre um determinado equipamento a partir do momento da quebra, considera-se a Manutenção Reativa. (BEN-DAYA *et al.*, 2009).

A diferença entre as duas categorias classificadas como manutenção não planejada é que a modalidade corretiva associa-se à restauração de um equipamento após a ocorrência de alguma falha, enquanto que para a manutenção emergencial as ações são tomadas no momento em que a falha ocasiona, sendo, portanto uma abordagem ainda menos proativa que a corretiva uma emergência referente à segurança ou desempenho do sistema (VATHOOPAN *et al.*, 2018).

Figura 1 - Políticas de Manutenção



Fonte: Adaptado de Ben Daya et al. (2009)

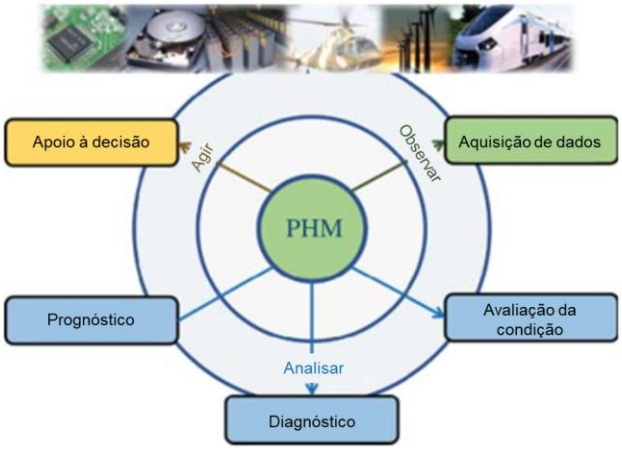
Quando o sistema possui uma intensidade de falha crescente, o custo de manutenção e de decorrência das quebras torna-se excessivamente custoso para o sistema. Portanto, faz-se necessário a execução de algumas atividades para preveni-la. A manutenção preventiva está relacionada com a execução de algumas atividades rotineiras como: lubrificação, reposição de componentes, inspeção,

entre outras estratégias voltadas para melhoria da confiabilidade do equipamento. (MARTIN, 1994)(MISRA, 2008). Para o caso da manutenção preventiva, as atividades são agendadas e executadas sem o aporte de indicadores da saúde dos ativos industriais. Dessa forma, à medida que a demanda por confiabilidade aumenta, as atividades preventivas aumentam drasticamente, tornando o custo dessa política de manutenção demasiadamente oneroso (AHMAD; KAMARUDDIN, 2012).

Com o aumento da necessidade por confiabilidade de sistemas, bem como desenvolvimento de tecnologias de sensoriamento, torna-se viável a coleta constante e contínua por: corrente, emissão acústica, temperatura, potência, vibração qualidade de óleo, entre outros. Levanta-se a oportunidade dsses dados orientarem o processo de tomada de decisão. Consequentemente, uma redução de tarefas dispendiosas e desnecessárias no equipamento. Bem como, garantem um processo de gerencimanento da manutenção embasado por dados. Essa filosofia é denominada manutenção baseada na condição (CBM – do inglês, *Condition Based Maintenance*) (JARDINE; LIN, D.; BANJEVIC, 2006).

A aplicação de CBM demanda por um sistema que englobe: modelos, equipamentos e pessoas. Para gerencisar a saúde dos ativos, essa estrutura deve executar algumas etapas fundamentais como: coleta e armazenamento dos dados, detecção de defeito dos equipamentos, diagnóstico do defeito, prognóstico do equipamento e, por último, garantir que essas informações sejam adequadamente utilizadas para suporte ao decisor. Denomina-se essa estrutura por Gestão de saúde e prognóstico de ativos (PHM - do inglês, *Prognosis and Health Management*) (PECHT; KANG, 2018)

Figura 2 – Framework para a implementação de PHM



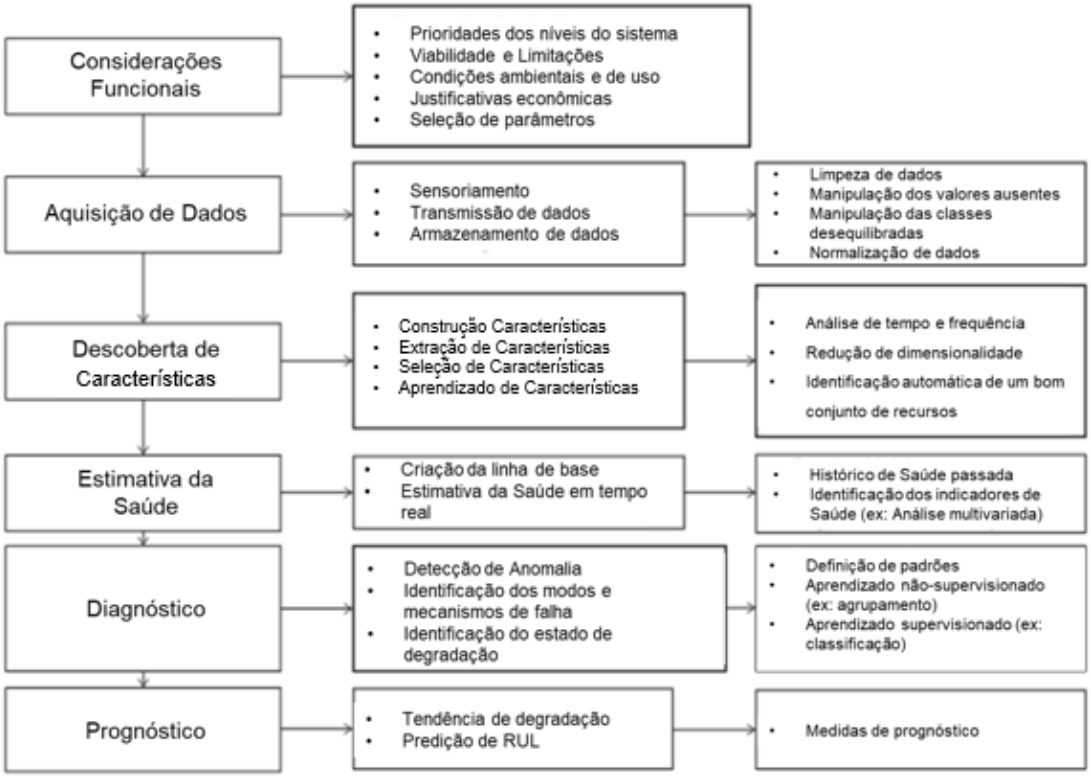
Fonte: Adaptado de Pecht & Kang, 2019

A Figura 2 apresenta um modelo de *framework* para a implementação e execução do PHM, estabelecendo que são competências de PHM a aquisição de dados de saúde do equipamento, as análises da saúde do equipamento, o diagnóstico e o prognóstico (DONG, M.; HE, D., 2007). No diagnóstico, o estado atual do equipamento é avaliado e categorizado. No prognóstico, são executadas previsões acerca das condições futuras do equipamento.

A análise para diagnóstico e prognóstico requerem o uso de modelos, os quais podem ser classificados como: abordagens baseadas na física da falha, abordagens baseadas em dados ou abordagem híbridas. Os modelos baseados em dados (*Data-driven*) são amplamente utilizados para essa função. Nesse caso, uma quantidade grande de dados é coletada e armazenada. Essas bases de dados são armazenadas e modelos de estatística ou aprendizagem de máquina são utilizados para reconhecer e reproduzir os padrões entre os fenômenos de falha do equipamento e os indicadores de saúde do equipamento.

A Figura 3 apresenta um esquema de etapas para a execução do PHM via abordagens *data-driven*.

Figura 3 – Esquema de Abordagem Data-Driven para PHM



Fonte: Adaptado de Pecht & Kang, 2019

2.3 APRENDIZAGEM DE MÁQUINA

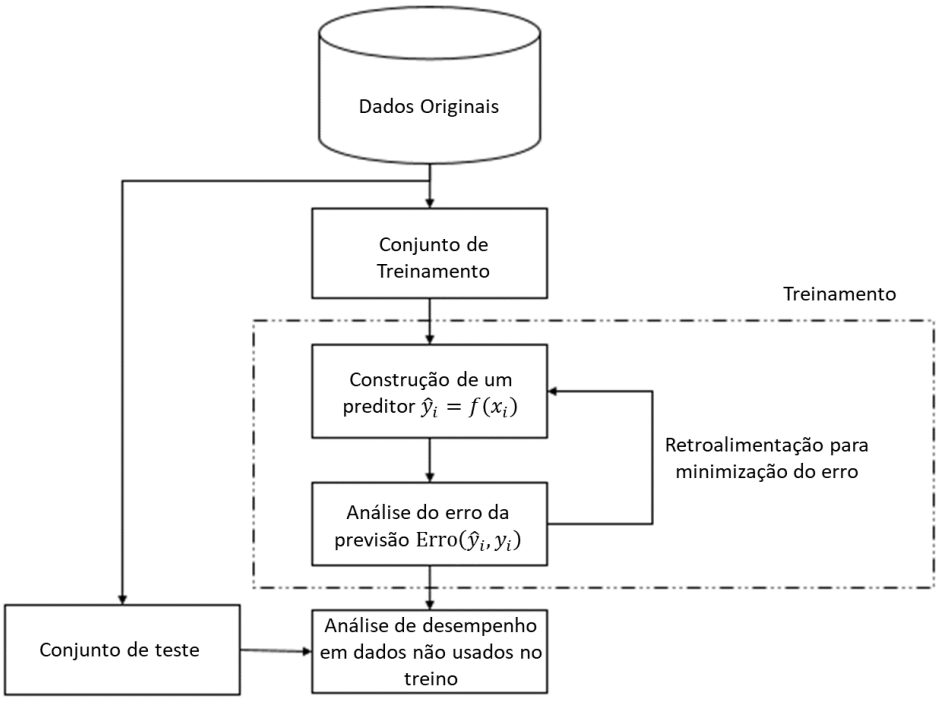
Para a área de estudo de ciência de dados, aprendizagem de máquina trata do desenvolvimento de modelos voltados para acesso de dados, reconhecimento e reprodução de padrões a partir da análise desse conjunto de dados (MITCHELL, 1997). O objetivo dos modelos de machine learning é aprender, de forma automatizada, padrões e relações de dependência em um conjunto de dados a partir de exemplos. (JANIESCH; ZSCHECH; HEINRICH, 2021).

Os modelos de aprendizagem de máquina podem ser categorizados entre: supervisionados ou não supervisionados. Para o caso não supervisionado, os dados não se encontram associados a um rótulo. Esse tipo de ferramenta é importante quando o banco de dados não possui esse recurso, o qual pode estar associado a um grande esforço humano para classificação prévia dos dados. (MAKHZANI *et al.*, 2015).

O aprendizado supervisionado é aplicado quando os dados e o contexto de aplicação se estruturam, de forma que: existe uma variável dependente y a ser prevista; o objetivo do algoritmo é reconhecer o padrão do valor de y a partir de valores de uma variável independente x. Dessa forma, os dados são previamente rotulados e, a ideia de aprendizagem supervisionada gira em torno de construir uma função preditiva $\hat{y} = f(x)$.

A natureza da variável dependente define o tipo de aprendizagem supervisionado. Essencialmente, caso pertença ao conjunto dos números reais trata-se de uma regressão e, para os casos de variáveis categóricas, trata-se de uma atividade de classificação (DUDA; HART; STORK, 2001). A Figura 4 apresenta um esquema para a aprendizagem supervisionada. Observa-se que o rótulo, ou variável dependente, é utilizado para cálculo de uma métrica de erro do preditor construído inicialmente. Esse erro é iterativamente minimizado pelo algoritmo.

Figura 4 – Aprendizagem Supervisionada



Fonte: Adaptado de Nguyen et al., 2016

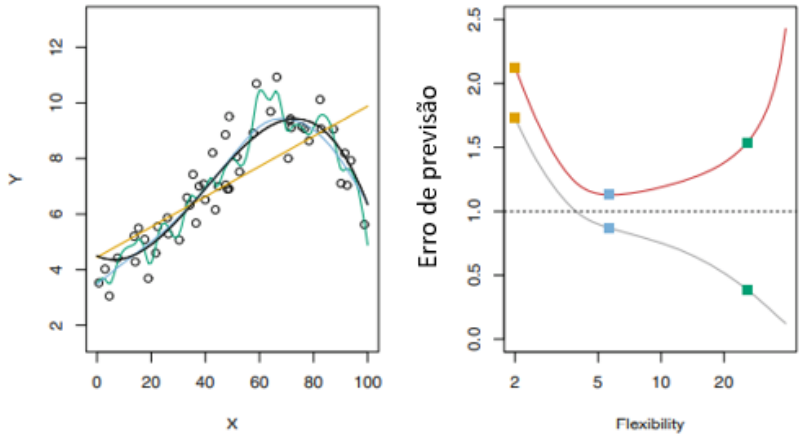
Além dessas duas abordagens, é possível encontrar trabalhos que categorizam alguns aprendizados como semi-supervisionados, nesses casos, geralmente há uma pequena quantidade de dados rotulados previamente (PACHECO; CERRADA; SANCHEZ, 2017). Além disso, é possível usar algoritmos semi-supervisionados para problemas de detecção de anomalia em que há pouquíssimos dados anômalos, nesses casos os rótulos são usados no pré-processamento e não no treinamento do modelo em si (KRISSAANE *et al.*, 2019).

Os algoritmos de aprendizagem de máquina, comumente fazem uma otimização em que se busca por um modelo, o qual deve se ajustar de forma mais adequada aos valores dos dados de treinamento. Por exemplo, em algoritmos supervisionados, busca-se por um preditor f que realiza a previsão da variável dependente. Essa busca requer um processo de avaliação da qualidade do ajuste, o qual envolve uma métrica que deve ser adaptada ao contexto de aplicação.(JAMES; WITTEN; HASTIE, 2019).

No entanto, a medida de desempenho deve ser avaliada não somente pelos resultados na base de dados utilizada no treinamento. Deve-se avaliar essencialmente em uma base de dados não utilizada na busca pela função preditora, a qual denomina-se dados de teste (Figura 4). Não é incomum que um algoritmo de aprendizado de máquina resulte em um preditor com baixo erro de treinamento e alto erro de teste. Descreve-se um fenômeno quando o modelo tem excesso de flexibilidade e poder de adaptação aos dados de treinamento. Nesse caso, o preditor encontrado se ajusta exageradamente aos dados de treinamento, mas não performa de maneira similar quando é necessário generalizar a solução para os dados do conjunto de teste. Esse processo é denominado sobreajuste (do inglês, *overfitting*). Um algoritmo com pouca flexibilidade para o contexto de aplicação constrói um preditor incapaz de observar, adequadamente, um padrão nos dados de treino. Esse fenômeno, oposto ao supracitado, é denominado sobajuste (do inglês, *underfitting*).(JAMES; WITTEN; HASTIE, 2019).

A Figura 5 apresenta, no diagrama à esquerda, exemplos de três modelos que fazem uma busca por um preditor $f(x_i) = \hat{y}_i$, nos quais os pontos apresentados são o conjunto de treinamento. A curva em preto, representa a função original e as curvas azul, verde e dourada representam o ajuste de três modelos distintos. O ajuste amarelo é uma regressão linear e não é flexível para se aproximar da função original, portanto exemplifica *underfitting.*

Figura 5 – Exemplos de *Overfitting* e *Underfitting*



Fonte: Adaptado de James et al, 2019

O preditor representado em verde possui o menor dos desvios para o conjunto de treinamento, com grande flexibilidade o algoritmo é capaz de construir um preditor que se aproxime mais dos pontos de treimanto. No entanto, o ajuste excessivo a esse conjunto de dados leva o algoritmo a uma resposta distante da função original, caracterizando um overfitting. Em contrapartida, diferentemente dos outros modelos, o preditor azul é capaz de reconhecer bem a função principal.

O gráfico da direita da Figura 5 apresenta como se comporta o erro de previsão para o conjunto de teste (curva vermelha) e de treinamento (curva cinza). Os pontos dourado, azul e verde representam a posição de cada um dos modelos apresentados anteriormente. Nota-se, o modelo representado pela cor dourada com alto erro em ambos os conjuntos de dados. Além disso, o modelo representado em verde com o menor erro para o conjunto de treino, mas com erro elevado no conjunto de teste. É necessário buscar um ponto de equilíbrio da complexidade do algoritmo para o contexto de forma a não gerar um overfitting.

2.2.1 Detecção de Anomalia

Chandola et al. (2009) definem detecção de anomalia como “O problema matemático de identificar padrões em dados que não se comportam como o esperado”. Em outras palavras, a detecção de anomalia trata do reconhecimento de pontos em um conjunto de dados que se distinguem de um padrão mapeado anteriormente pelo modelo. Esses registros podem ser denominados como anomalia, *outliers*, observações discordantes, exceções, entre outros, a depender do contexto em que é aplicado.

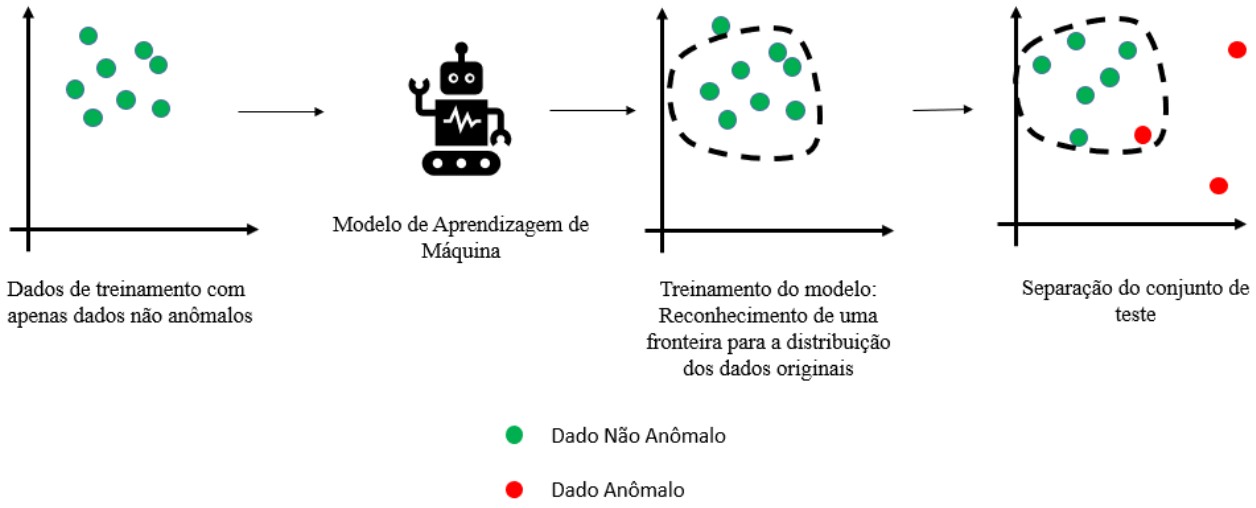
Assim como para os modelos de aprendizagem de máquina, o subconjunto de abordagens de detecção de anomalia pode ser particionado como supervisionado, não supervisionado e semi-

supervisionado. A seguir, descreve-se brevemente o funcionamento de cada abordagem (ALLA; ADARI, 2019).

- Supervisionado: Os dados possuem um rótulo que diferencia os dados normais das observações discordantes e esse rótulo é utilizado durante o treinamento do modelo;
- Semi-supervisionado: Treinam-se os modelos somente com dados de um comportamento específico (exemplo: dados de máquina com comportamento saudável) e, no conjunto de teste, busca-se identificar os pontos que divergem da distribuição original;
- Não supervisionado: Não é possível separar os dados originais em normais e anômalos, portanto, não há rótulos previamente estabelecidos e os modelos devem buscar quais pontos se afastam da distribuição majoritária dos dados.

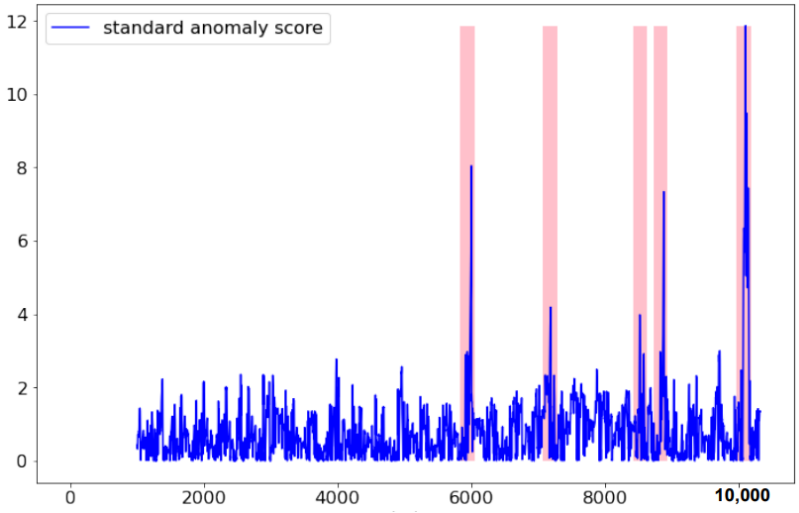
A Figura 6 apresenta um exemplo de detecção de anomalia não supervisionada em um conjunto de dados distribuídos em um espaço bidimensional. Note que o algoritmo busca reconhecer quais variáveis do conjunto de teste fogem da distribuição dos dados de treinamento. Para esse tipo de modelo de detecção de anomalia, é recorrente que os algoritmos sejam utilizados para cálculo de uma pontuação de anomalia, a qual é um indicador da diferença do valor de um dado de teste e da distribuição dos dados utilizados no treinamento (BEGGEL; PFEIFFER; BISCHL, 2020; SCHLEGL *et al.*, 2017). A Figura 7 apresenta um exemplo de gráfico de detecção de anomalia a partir de pontuação de anomalia, observa-se que os picos de valores de pontuação são categorizados como anômalos, nesse caso é necessário estabelecer um critério numérico para separar as duas categorias de dados.

Figura 6 - Aprendizagem Semi-supervisionada



Fonte: O Autor, 2022

Figura 7 – Método de detecção de anomalia por pontuação de anoamalia

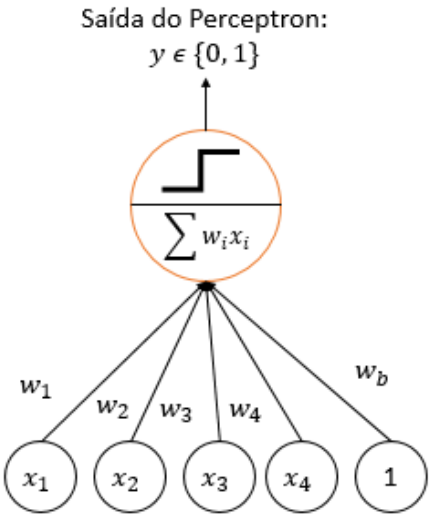


Fonte: Adaptado de Jiang et al., 2021

2.3 REDES NEURAIS

As redes neurais foram propostas pela primeira vez pelo neurocientista Warren MacCulloh (1943). No estudo, apresentou-se um algoritmo que executava uma representação simplificada do processamento de dados em neurônios biológicos. Entende-se, portanto, a inspiração biológica do algoritmo. O perceptron é a estrutura mais simples de rede neural proposta. Ela consiste por operador de função degrau, o qual se conecta a um somatório ponderado dos dados de entrada. O treinamento do perceptron (Figura 8) é representado pela busca dos pesos w, os quais minimizam uma função erro de saída. A saída do perceptron é definida por uma combinação linear dos valores das entradas.

Figura 8 - Perceptron



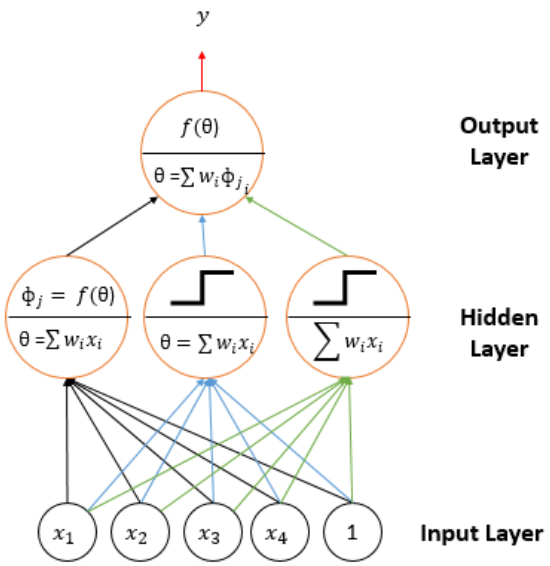
Fonte: O Autor, 2022

No entanto, por sua simplicidade, esse modelo não possui grande aplicabilidade para modelos reais. Posto que o algoritmo resolve, exclusivamente, problemas lineares, os quais estão fora da realidade da maioria dos problemas reais de classificação e regressão (GERON, 2011). No entanto, as redes neurais, como usualmente aplicadas, podem ser descritas como uma combinação de perceptrons e são capazes de resolver problemas reais em diversos contextos distintos.

Portanto, uma rede de múltiplas camadas de perceptron (MLP – do inglês – *Multi-Layer Perceptron*) é um tipo de arquitetura comumente utilizada. A primeira camada consiste nos dados de entrada. Para as camadas intermediárias, cada perceptron, faz operações matemáticas similares à apresentada pela Figura 8. A saída das operações matemáticas em cada camada é enviada como sinal aos perceptrons das camadas posteriores até a camada de saída (Figura 9). Quando uma rede neural possui uma quantidade relativamente grande de camadas, entende-se que ela faz parte da família de algoritmos de aprendizagem profunda (GÉRON, 2017).

A função de ativação da última camada define o formato da variável de saída da rede neural. (GÉRON, 2017). O treinamento de uma rede MLP consiste em uma busca estruturada dos valores dos pesos a partir de uma minimização de medidas de erros da classificação ou regressão. As possibilidades desses algoritmos serão melhor apresentadas no capítulo posterior.

Figura 9 – *Multi-Layer Perceptron*



Fonte: O Autor, 2022

2.3.1 Treinamento de Redes Neurais

De forma tradicional, redes neurais MLP possuem treinamento supervisionado, portanto o ajuste do conjunto de pesos W^T é calculado de acordo com uma função perda (*loss*) que represente a

diferença entre a previsão executada pelo modelo $\hat{Y} = (\hat{y}_1, \hat{y}_2, \hat{y}_3, \dots, \hat{y}_n)$ e o valor real $Y = (y_1, y_2, \dots, y_n)$ e um valor, e y_i pode ser uma variável binária, categórica ou pertencente a $\mathbb{R}^m$, a depender do objetivo da rede neural. Dois exemplos de funções *loss* amplamente utilizadas para, respectivamente, regressão e classificação binárias são o erro quadrático médio e entropia cruzada binária (AGGARWAL, 2018; GERON; GÉRON, 2017).

$$Erro\ quadrático\ médio = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(y_{ij} - \hat{y}_{ij}\right)^2 * \frac{1}{n * m}$$

$$Erro\ quadrático\ médio = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(y_{ij} - \hat{y}_{ij}\right)^2 * \frac{1}{n * m}$$

$$Entropia\ cruzada\ binária = -\frac{1}{n} * \left[\sum_{i=1}^{n} y_i \log(p_i) + (1 - y_i)\log(1 - p_i)\right]$$

Em que p_i é a previsão da probabilidade de $y_i = 1$ feita pela rede neural, e usualmente:

$$\hat{y}_i = \begin{cases} 0, & p_i \leq 0{,}5 \\ 1, & c.c. \end{cases}$$

Portanto, o processo de treinamento de um modelo de rede neural pode ser resumidamente expressado como a solução de um problema de programação matemática:

$$\min_{W^T} J(W) = Loss\ Function$$

Para solucionar esse problema, é comum a utilização de métodos iterativos de otimização, através dos quais um procedimento de busca e cálculo da função objetivo é feito de forma cíclica, e a estrutura de cálculo do algoritmo é construída para que a cada iteração, a solução candidata esteja cada vez mais aproximada da solução ótima para o problema. Esses modelos são utilizados quando a otimização exata não é viável devido à complexidade da função objetivo e das restrições. (DAVID *et al.*, 2015).

Problemas de otimização não convexa, envolvendo grande dimensionalidade do conjunto de variáveis de decisão são o caso mais comum para redes neurais. As abordagens iterativas popularmente utilizadas para treinamento de redes neurais, e implementadas em bibliotecas *open-source* de redes neurais (Por exemplo: Keras, Tensorflow, Pytorch) são embasadas no método estocástico de descida por gradiente (GD – do inglês, *Gradient discent*). Nesse caso, a posição da

solução candidata é atualizada a cada iteração na direção contrária do vetor gradiente. (DOGO *et al.*, 2018).

$$s_{j,i+1} = s_{j,i} - \eta \frac{\partial J}{\partial s_j}(s_i)$$

Em que $s_i = (s_{1,i}, s_{2,i}, \dots, s_{l,i})$. é o vetor da posição da solução candidata na iteração i do modelo e η é um hiperparâmetro do algoritmo denominado taxa de aprendizagem. Para o caso de redes neurais, usualmente há um grande volume de dados de treinamento o que pode tornar o cálculo da função perda por iteração muito custoso computacionalmente, o algoritmo de GD estocástico (SGD – do inglês, *Stochastic Gradient Discent*) e de GD por mini-lotes (MBGD – do inglês, *Mini-Batch Gradient Discent*) são propostos como solução a esse problema ao, respectivamente, utilizar apenas um dado de treinamento ou uma pequena quantidade. (KETKAR, 2017).

Ao longo dos anos, algumas melhorias foram propostas ao SGD, a otimização por momento (MO – do inglês, Momentum Optimization) adiciona, a cada iteração, o valor do gradiente a um vetor de momento m multiplicado pela taxa de aprendizagem, e a atualização da posição da solução candidata é feita subtraindo a posição anterior do vetor de momento. Em analogia à física, o gradiente no modelo GD é usado como velocidade enquanto por MO é usado como aceleração. (DOGO *et al.*, 2018; GÉRON, 2011; WU, W. *et al.*, 2008).

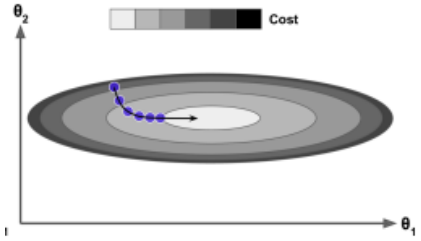
$$m_{i+1} = \beta m_i + \eta \nabla_\theta J(s_i)$$

$$s_{i+1} = s_i - m$$

Com esse ajuste é possível impedir um dos problemas advindo da forma de “tigela alongada” (Figura 10) que as funções perda de rede neurais tomam, o comportamento do SGD faz com que a convergência da solução fique muito lenta quando se aproxima do ótimo global (GÉRON, 2011). O hiperparâmetro β simula um coeficiente de atrito, o qual garante que o momento não tome dimensões exacerbadas. Uma pequena variante desse algoritmo é a otimização por gradiente acelerado de Nesterov (NAG – do inglês, *Nesterov accelerated gradient*), a ideia é medir o gradiente da função custo, não na posição s_i mas em $s_i + \beta m_i$, no caso de NAG o cálculo do momento segue a seguinte equação:

$$m_{i+1} = \beta m_i + \eta \nabla_\theta J(s_i + \beta m_i)$$

Figura 10 – Problema da Tigela Alongada



Fonte: Géron, 2011

O algoritmo RMSProp busca resolver outro problema ocasionado pelo formato das funções objetivos, na Figura 10, é possível observar que o algoritmo SGD, além de ficar lento quando se aproxima do ótimo global, inicialmente não se locomove para a direção do ótimo da função objetivo (GÉRON, 2011). Portanto, apresenta-se um ajuste que leva o algoritmo de forma mais direta para a solução:

$$m_{i+1} = \beta m_i + (1-\beta)\nabla_\theta J^2$$
$$s_{i+1} = s_i - \frac{\eta}{\sqrt[2]{m_i + \epsilon}} * \nabla_\theta J^2$$

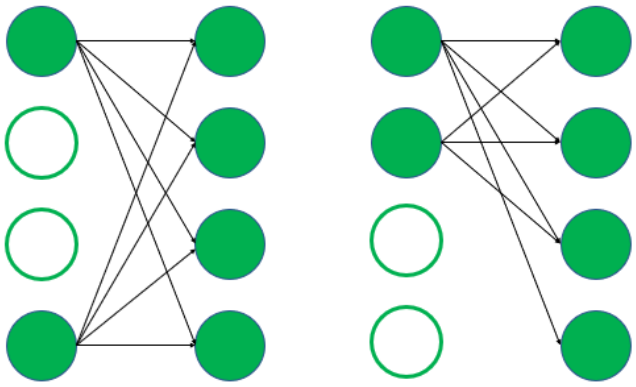
Em contrapartida, o algoritmo de estimativa de momento adaptativa (ADAM – do inglês, *Adaptative Momentum estimation*) foi proposto em 2014 (KINGMA; BA, 2014) como uma solução que combina as ideias dos modelos MO e RMSProp, portanto resolve tanto a questão de perda de velocidade de convergência da solução quando próximo do ótimo global, quanto direciona a solução a cada iteração, de forma mais eficiente em direção à solução. O uso de *mini-batchs* assim, como para o algoritmo MBGD é usual, mesmo com o uso dos outros algoritmos.

$$v_{i+1} = \beta v_i + (1-\beta)\nabla_\theta J^1$$
$$m_{i+1} = \alpha m_i + (1-\alpha)\nabla_\theta J^2$$
$$s_{i+1} = s_i - \frac{\eta v_i}{\sqrt[2]{m_i + \epsilon}} * \nabla_\theta J^2$$

Para aplicação desses modelos é necessário calcular o gradiente da função perda em relação aos pesos W^T. Desta forma, os pesos são ajustados a cada iteração, e o gradiente deve ser recalculado a cada nova posição calculada. O cálculo dos gradientes é feito a partir de um procedimento denominado *backpropagation*, uma aplicação de regra da cadeia. Com essa abordagem, calcula-se o gradiente da camada de saída e, utilizando os resultados, calcula-se os gradientes das camadas até o cálculo dos pesos para as camadas de entrada.

O mesmo problema de sobreajuste apresentado na seção 2.1 é presente também no caso de redes neurais. É necessário que a arquitetura de rede neural em associação com um modelo de busca apropriado. A regularização trata de procedimentos para solucionar esse problema, o uso de camadas de *dropout* é uma das principais abordagens nesse aspecto. O processo de regularização por *dropout* se dá por adicionar um fator aleatório à arquitetura da rede neural, é uma característica atribuída à uma ou mais camadas escondidas em uma rede MLP. De modo simplificado, quando uma camada é afetada por esse método, a cada iteração do processo de otimização, sorteia-se para cada um dos neurônios se ele fará, ou não, parte do processamento dos dados e predição. A Figura 11 apresenta um exemplo de uma rede neural, em que a camada à esquerda possui um dropout. A cada iteração, por um processo de amostragem, sorteia-se os neurônios eliminados da rede para aquela iteração. Nas iterações seguintes realiza-se um novo processo de amostragem.

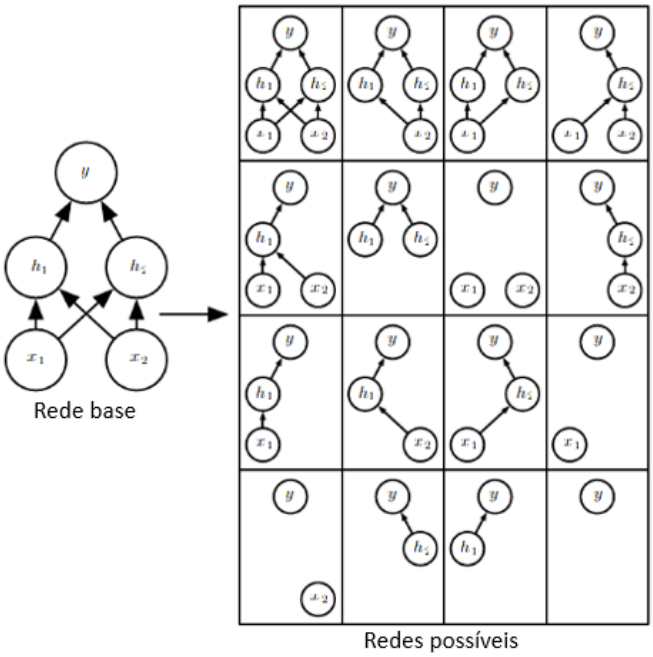
Figura 11 – Exemplo de Dropout



Fonte: O Autor, 2022

De forma mais específica, o dropout torna o algoritmo de aprendizado de máquina em uma agregação de todas possíveis redes neurais dada as combinações de neurônios eliminados, ou não. A Figura 12 apresenta um exemplo de uma rede com duas camadas, e todos possíveis modelos que são agregados pelo uso de dropout. A inserção de variabilidade no processo reduz a capacidade do modelo de se ajustar excessivamente aos dados de treinamento e mitiga a possibilidade de overfitting. (GOODFELLOW, I.; BENGIO; COURVILLE, 2016)

Figura 12 – Redes possíveis em um dropout



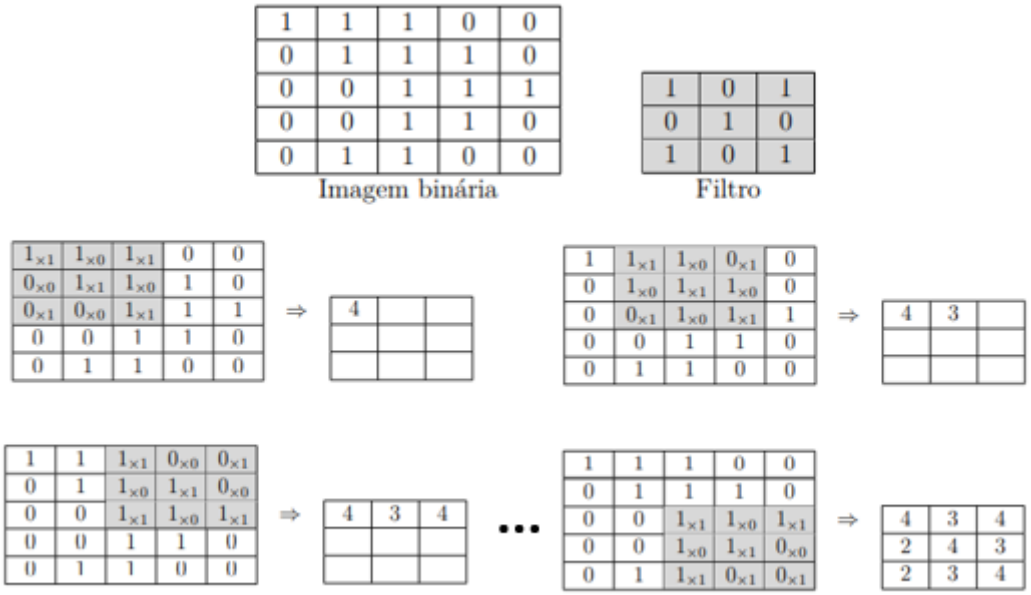
Fonte: Adaptado de Goodfellow et al. (2016)

2.3.1 Camada Convolucional

Uma operação de convolução em rede neural é um procedimento que tomou grande importância na área de estudo com a popularização das redes neurais convolucionais (CNN – do inglês, *Convolutional Neural Networks*) (CHOLLET, 2018). Apesar da popularidade para solucionar problemas de processamento de imagem a operação de convolução é usada em diversos contextos.

Em redes neurais, uma camada convolucional consiste em um conjunto de filtros que são aplicados a um dado de entrada. Durante o treinamento, os pesos dos filtros permitem uma busca por características em posições do banco de dados. A Figura 12 apresenta um esquema do funcionamento de filtro aplicado a um dado de imagem binário, a saída da combinação do filtro com a imagem resulta na matriz 3x3 apresentada.

Figura 12 – Filtro de Camada Convolucional aplicada a uma imagem binária



Fonte: Ferreira, 2020

O processo de aplicação das convoluções matemáticas a partir desse procedimento permite que o algoritmo de aprendizado profundo automatize o processo de extração de características do banco de dados. Essa propriedade da camada convolucional reduz a demanda por conhecimento especializado para executar a atividade de engenharia de características (em inglês, *featuring engineering*).

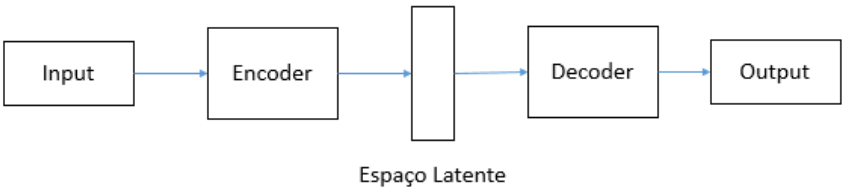
O procedimento de camadas convolucionais pode ser adaptado para dados que não estejam dispostos em duas dimensões. A camada convolucional 1D é comumente utilizada para dados temporais e sinais 1D. Além disso, essas abordagens apresentam grandes vantagens em termos de tempo de processamento computacional (KIRANYAZ et al., 2021).

2.3.2 Autoencoders

Os *autoencoders* são métodos, ao contrário de MLP, não supervisionados e foram propostos em 1986 como uma rede neural que é treinada exclusivamente para replicar o seu input. Os *autoencoders* são responsáveis por reduzir a dimensionalidade das entradas e, posteriormente, reconstruir os dados reduzidos de forma a se aproximarem o máximo possível da entrada original. Essas estruturas de redes neurais são compostas por uma rede que reduz a quantidade de dados em um espaço latente (*encoders*) de tamanho menor que o original e uma rede que, a partir desses dados,

replica o dado de entrada (*decoders*) (Figura 13). (BANK; KOENIGSTEIN; GIRYES, 2020; RUMELHART; HINTON; WILLIAMS, 1986).

Figura 13 - Autoencoders

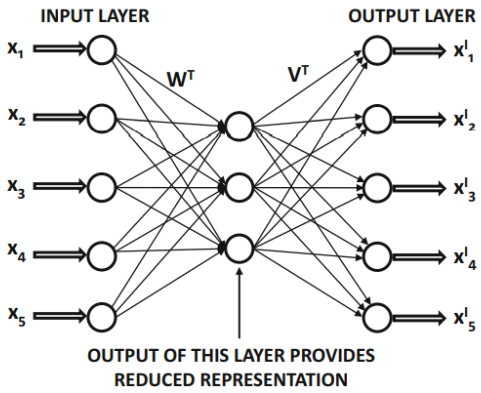


Fonte: O Autor, 2022

Através da análise do *encoder* e do *decoder*, é possível assumir que os *encoders* devem aprender a identificar as informações importantes e combiná-las adequadamente, de forma a reduzir a dimensionalidade com menor perda de informação possível, e os *decoders* são treinados para traduzir a informação presente no espaço latente (FOSTER, 2019).

O *autoencoder* mais básico possível consiste em uma rede de apenas três camadas: uma camada dos dados de entrada, uma camada com o espaço latente e uma camada de saída da arquitetura (Figura 14).

Figura 14 – *Autoencoder* com uma camada escondida



Fonte: Aggarwal, 2018

A otimização dessa rede neural, consiste na busca pelos pesos do *encoder* (W^T) e do *decoder* (V^T) de forma a reduzir uma função perda, por se tratar de um modelo de regressão. Uma função objetivo de uso comum é o erro quadrático médio (MSE – do inglês – *Mean Squared Error*) (GÉRON, 2011). Portanto, é possível afirmar que, para um conjunto de dados de treinamento $X = (X_1, X_2, X_3, .., X_n)$, em que $X_i \in \mathbb{R}^m$ e representa um vetor de entrada da rede neural, o treinamento da rede neural consiste em resolver o seguinte modelo de otimização:

$$\min_{W,V} Erro\ de\ reconstrução = MSE = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(x_{ij} - \hat{x}_{ij}\right)^2 * \frac{1}{n*m}$$

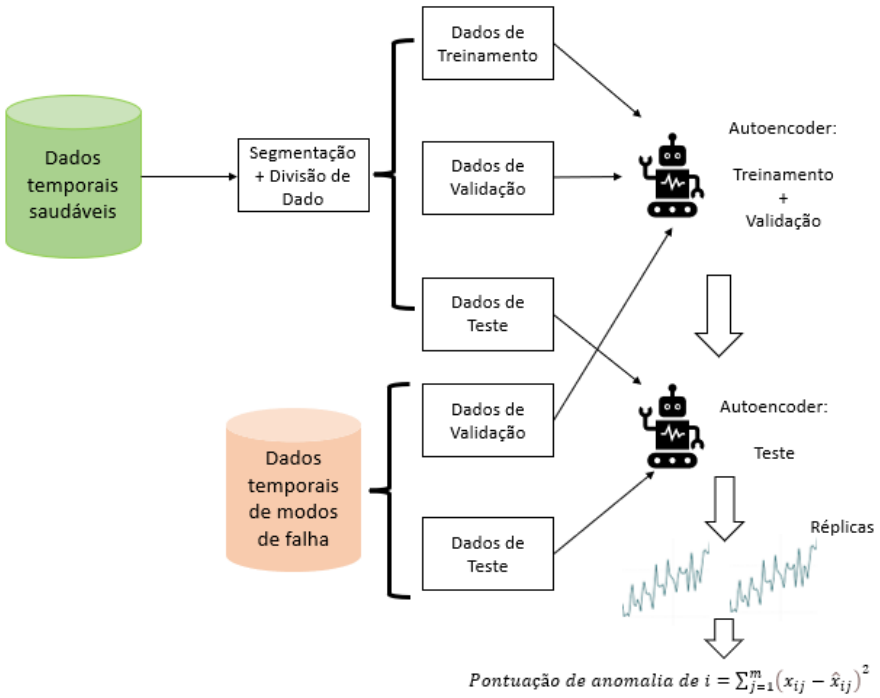
em que $X_i = (x_{i1}, x_{i2}, \ldots, x_{im})$

e $\hat{X}_i = (\hat{x}_{i1}, \hat{x}_{i2}, \ldots, \hat{x}_{im})$ é o resultado da saída do modelo.

O algoritmo para resolver o problema de programação matemática é implementado a partir de cálculos de gradiente e *backpropagation* como apresentado na seção 2.3.2.

O uso de *Autoencoders* para detecção de anomalia segue o esquema apresentado na Figura 15, a rede neural é treinada para replicar os dados de treinamento, os quais devem se comportar como os dados não anômalos. Após a otimização dos pesos da rede neural, espera-se gerar uma função capaz de codificar e replicar os dados que se comportem de maneira similar aos de treinamento. No entanto, o *autoencoder* não conseguiria executar o mesmo processo para dados muito distintos desse comportamento. Portanto, é possível utilizar o erro de replicação como pontuação de anomalia.

Figura 15 – Esquema para detecção de anomalia via *AutoEncoder*



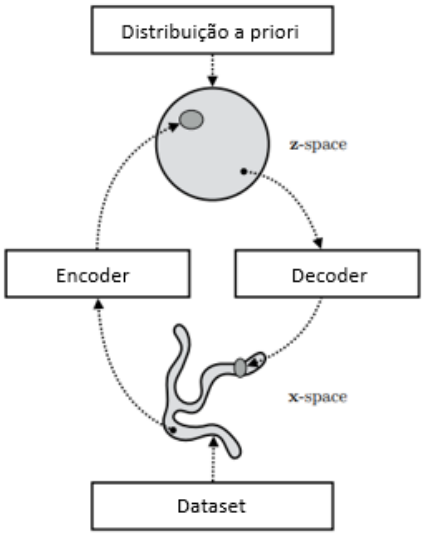
Fonte: O Autor, 2022

2.3.2 Autoencoder Variacional

A construção do *Autoencoder* Variacional (VAE – do inglês *Variational Autoencoder*) foi proposta por Kingma e Welling (2014) como uma solução para a necessidade de modelos capazes de ajustar dados com grande dimensionalidade a uma distribuição probabilística multivariada a priori. Portanto, um modelo *Autoencoder* Variacional busca codificar os dados de treinamento, não em um espaço vetorial de dimensão reduzida, mas em uma distribuição de probabilidade cuja função de densidade de probabilidade é definida a priori e, durante o processo de otimização, buscam-se os melhores parâmetros para essa distribuição. A Figura 16 apresenta o funcionamento de um VAE o *encoder* executa um mapeamento estocástico do espaço dos dados de entradas x, cuja distribuição é tipicamente complexa. O espaço da distribuição a priori é definido por equações as quais se conhecem, o *encoder* converte os dados em parâmetros da distribuição.

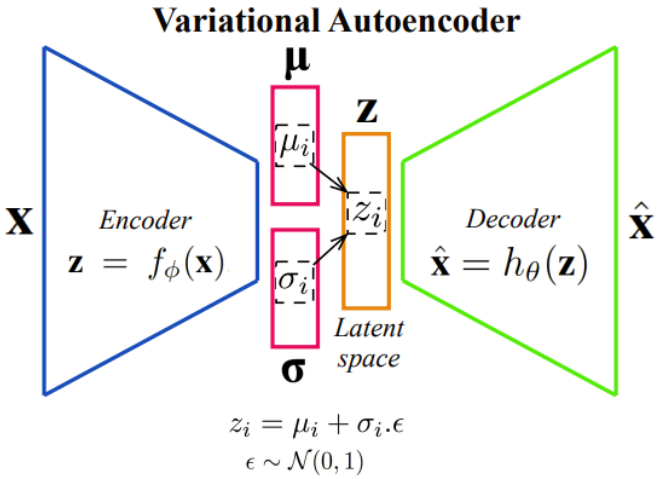
A Figura 17 apresenta um esquema da arquitetura de *Autoencoder* Variacional. A arquitetura contém um processo de amostragem de uma normal multivariada por simulação e o processo da rede converte os dados de entrada em parâmetros dessa distribuição. Para o uso em modelagem de detecção de anomalia, o procedimento se dá de maneira semelhante ao apresentado na Figura 15. Em outras palavras, usa-se o erro de replicação como pontuação de anomalia.

Figura 16 – Processo de representação *autoencoder* variacional



Fonte: Adaptado de Kingma; Welling, 2019.

Figura 17 – Arquitetura de rede neural VAE



Fonte: Zemouri, 2020

O processo de treinamento do VAE baseia-se em uma função perda que contêm tanto o erro de replicação (similar ao AE), quanto a divergência de Kullback-Leibler. Esse indicador representa a diferença entre duas distribuições de probabilidades, e é utilizado para avaliar o quanto a distribuição dos dados do espaço latente se diferenciam da distribuição a priori definida.

Para um conjunto de dados de treinamento $X = (X_1, X_2, X_3, .., X_n)$, em que $X_i \in \mathbb{R}^m$ e é a observação i do conjunto de dados de treinamento. Assumindo que o espaço latente tem dimensão l. $\mu_{i,j}$ e $\sigma_{i,j}$ são os valores de saída do encoder para, respectivamente a média e o desvio padrão da posição j do espaço latente para a observação i do conjunto de dados . As equações abaixo apresentam o cálculo da função perda de treinamento do VAE. Para o caso em que a distribuição a priori é uma variável normal ($\mu = 0, \sigma = 1$)

$$\text{Loss} = Erro\ de\ reconstrução + Divergência\ de\ Kullback$$

$$Loss = MSE + KL = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(x_{ij} - \hat{x}_{ij}\right)^2 * \frac{1}{n*m} - 0{,}5 * \sum_{i=1}^{n}\sum_{j=1}^{l} 1 + \log\left({\sigma_{i,j}}^2\right) - \mu_{i,j}^2 - {\sigma_{i,j}}^2$$

em que $X_i = (x_{i1}, x_{i2}, \ldots, x_{im})$

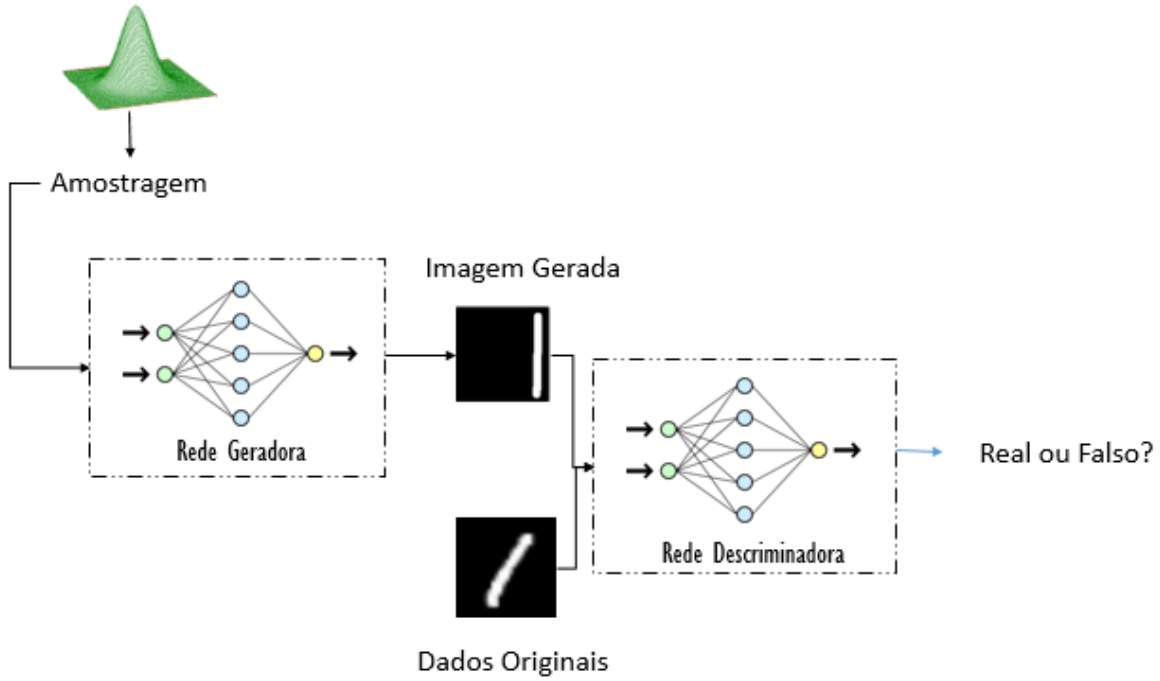
$\hat{X}_i = (\hat{x}_{i1}, \hat{x}_{i2}, \ldots, \hat{x}_{im})$ é o resultado da saída do modelo.

2.3.3 Rede Generativa Adversarial

As GANs correspondem a uma estrutura de rede proposta por Goodfellow et al. (2014), com o objetivo de implementar um framework de rede neural profunda capaz de gerar dados com a mesma natureza de um conjunto a partir de dados amostrados de uma distribuição de probabilidades.

Para executar essa tarefa o treinamento envolve dois modelos, um voltado para a geração de dados e outro para a discriminação entre dados reais e replicados. A Figura 18 apresenta um esquema da estrutura de um modelo adversarial. Portanto, a rede geradora tem o papel de construir dados a partir de uma distribuição a priori, enquanto a rede discriminadora tem o papel de distinguir os dados reais dos replicados (GOODFELLOW, I.; BENGIO; COURVILLE, 2016).

Figura 18 – *Generative Adversarial Networks*



Fonte: O Autor, 2022

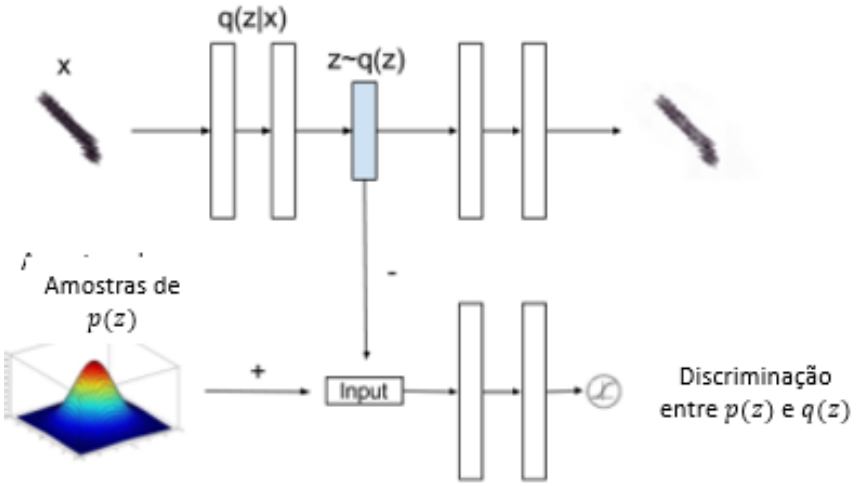
O treinamento de rede geradora adversária consiste em um jogo de Min-Max, na qual os pesos da rede discriminadora são determinados para que a separação dos dados falsos e verdadeiros seja feita corretamente. No entanto, a rede geradora é treinada para enganar a função discriminadora minimizando a sua taxa de acerto (MAKHZANI *et al.*, 2015).

Após o treinamento do modelo generativo adversarial, duas redes neurais estão com os pesos definidos: uma arquitetura capaz de transformar dados amostrados de uma distribuição a priori, usualmente a normal, em dados com comportamento similar ao da base utilizada, e outra rede capaz de separar dados gerados dos dados reais. Para detecção de anomalia via GAN, é possível utilizar a saída da rede discriminadora como pontuação de anomalia, dado que ela faz uma classificação binária baseada em uma saída real entre zero e um. A qual, quanto mais próxima de um, mais é possível afirmar que aquele dado não se assemelha à distribuição dos dados utilizados no treinamento (SHIN; BU; CHO, 2020).

2.3.4 Autoencoders Adversariais

O autoencoder adversarial foi proposto por Makhizani et al. (2015) como um modelo que une elementos das GANs com o autoencoder, construindo-se um modelo replicador e gerador de dados; a Figura 19 apresenta a arquitetura dessas redes. Note que, assim como nos autoencoders, a estrutura contém o encoder e o decoder, sendo adicionada uma rede discriminadora, a qual é treinada para diferenciar dados entre os resultados do espaço latente e dados amostrados de uma distribuição a priori.

Figura 19 – Autoencoders Adversariais



Fonte: Adaptado de Makhizani et al., 2015

Makhizani et al. (2015) apresentam diversas aplicações desse modelo, sendo de grande versatilidade na execução de muitas atividades de aprendizagem não supervisionada, especialmente, redução de dimensionalidade e clustering. Além disso, os autores compararam os resultados com outras arquiteturas de autoencoders e GANs e obtiveram resultados favoráveis para o uso dos autoencoders adversariais.

3 REVISÃO DE LITERATURA

Martin (1994) cita em uma pesquisa de estado da arte sobre diagnóstico de falhas e manutenção baseada em condição, que, antes do surgimento de máquinas de comando numérico computadorizado, a manutenção nas indústrias se dava após a quebra dos equipamentos. Essa estratégia é evidentemente insuficiente, pois ocasiona paralisações na linha de produção e, portanto, causa grandes volumes de estoque de material em processamento e exige disponibilidade de equipes de manutenção.

Com o passar dos anos, os avanços em coletas de dados e na ciência da computação favoreceram a chegada dos algoritmos de aprendizagem de máquina no contexto da gestão da manutenção. Em uma revisão de literatura mais recente, Zhang et al. (2019) apresentam uma gama de exemplos de como esses algoritmos vêm mudando a realidade de prognóstico e gestão da saúde de equipamentos industriais. Nesse contexto, a detecção de anomalia é uma das atividades que está sendo utilizada nessa conjuntura e os autoencoders correspondem a um dos modelos matemáticos não supervisionados mais usados com esse objetivo, especialmente para dados de séries temporais, por exemplo, de sinais de vibração (ZHANG, L. *et al.*, 2019).

No contexto de modelos de redes neurais e detecção de anomalia Zhao et al. (2018) desenvolvem um modelo de detecção de anomalia via autoencoders. A rede neural, por ser semi-supervisionada, é treinada apenas com dados não anômalos. Espera-se, maior erro de replicação para os dados anômalos. Para o caso de múltiplos sensores, assim como abordado neste trabalho, Renddy et al. (2016) propõem uma abordagem de autoencoders multi-modais, os quais são capazes de realizar replicação de dados temporais multivariados. A abordagem é similar ao do artigo supracitado, os autores treinam os autoencoders com os dados normais e usam o erro de replicação para definir os dados anômalos.

Para o caso de dados de series temporais, é necessário levar em consideração as características temporais e dificuldade de previsão durante o processo de análise, Araujo et al. (2019) propõem um modelo de previsão de RUL para rolamentos baseado em previsão de séries temporais e replicação de bootstrap de máxima entropia para construção de intervalo de confiança e análise da robustez da previsão.

Além do uso de autoencoders é possível encontrar estudos na literatura que abordam detecção semi-supervisionada de anomalia a partir de redes generativas. Schlegel et al. (SCHLEGL *et al.*, 2017) propuseram um framework para o uso delas em problemas de detecção de anomalia semi-supervisionados, no entanto, o modelo proposto foi aplicado para diagnósticos oftalmológicos por imagem.

No contexto de gestão da manutenção, Verstraete, Droguett e Modarres (2020) propõem um modelo de GAN para prognóstico semi-supervisionado. Nesse caso, o modelo foi proposto para realizar a previsão do tempo de vida útil remanescente dos equipamentos, a abordagem é capaz de lidar com regressões para os casos em que nem todos os dados estão rotulados. Paralelamente, Jiang et al. (2019) usaram uma GAN para detecção de anomalia em dados temporais de equipamentos industriais. Os dados usados na validação da abordagem são de vibração de rolamentos e formam uma série univariada. No trabalho, os autores apontam a necessidade de tratar dados multivariados como uma perspectiva para trabalhos futuros.

A proposição por modelos robustos de detecção de anomalia vem ganhando destaque na literatura, as abordagens de redes neurais adversariais generativa e redes de autoencoder são aplicadas em diversos desses estudos que abordam os mais variados contextos. Liang et al. (2021) propõem uma abordagem GAN para detecção de anomalia para dados de contexto industrial e leva em consideração dificuldades de balanceamento de base de dados e de poucos dados de padrão defeituoso.

Xiong e Zuo (2021) apresentam um estudo de detecção de anomalia com autoencoder e avaliam a robustez desse modelo aplicado à um contexto da geologia no qual o conjunto de dados apresenta grande volume de dados faltantes e presença de ruídos nas observações. Os autores definem que a robustez do modelo de autoencoder é satisfatória para o contexto de análise geoquímica.

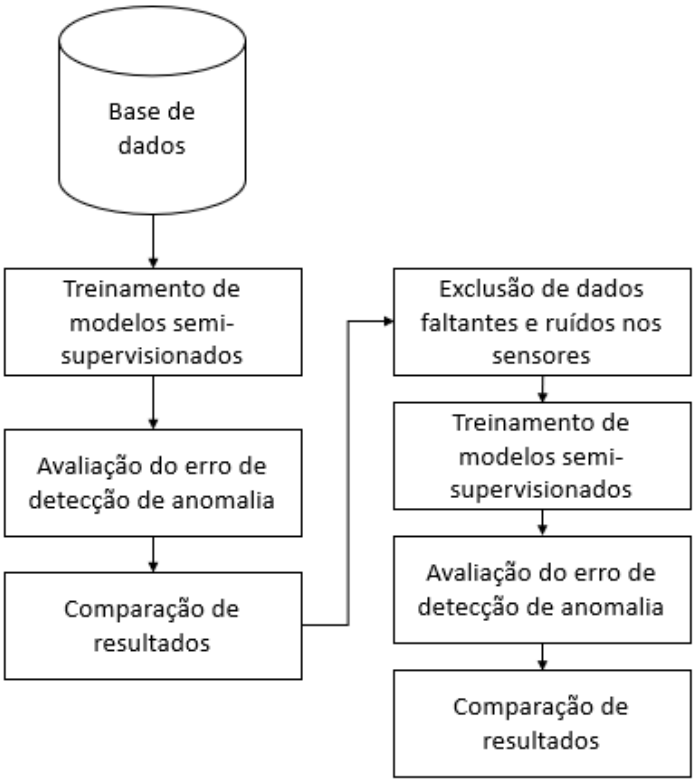
Uma abordagem similar é aplicada por Shin, Bu e Cho (2020) a partir de redes GAN para detecção de anomalia robusta em casos de visão computacional e processamento de dados de vídeos de câmeras de vigilância. Outro estudo da mesma aplicação que propõe uma abordagem de rede neural que une conceitos de autoencoder e redes neurais adversariais, é proposto por Vu et al. (2019). Em ambos os estudos os modelos apresentam resultados favoráveis para o uso dessa ferramenta.

No contexto de monitoramento de condição de ativos físicos industriais e PHM, é possível encontrar estudos que apresentam a presença de dados faltantes e de ruídos em conjunto de dados como um problema recorrente. Portanto, justifica-se a necessidade de desenvolver estudos que abordem o funcionamento de modelos *data-driven* de PHM nesses contextos. (LEE et al., 2014; OMRI et al., 2021b; YUCESAN; DOURADO; VIANA, 2021).

4 METODOLOGIA

A ideia deste trabalho consiste em testar como modelos de aprendizagem profunda e redes neurais se comportariam em ambientes com dados em falta ou ruído causado por falhas de sensores. Foram utilizadas três bases de dados distintas para avaliar a qualidade dos resultados dos modelos de detecção de anomalia. Primeiramente, os dados temporais são transformados em vetores de características e rótulos. Para isso é necessário separar anomalias de dados saudáveis e separar os dados em vetores. Posteriormente os dados são divididos em treino, validação e teste e os algoritmos são executados e avaliados. A Figura 20 mostra um esboço da metodologia do trabalho, cujos passos serão explicados detalhadamente posteriormente.

Figura 20 – Metodologia Proposta



Fonte: O Autor, 2022

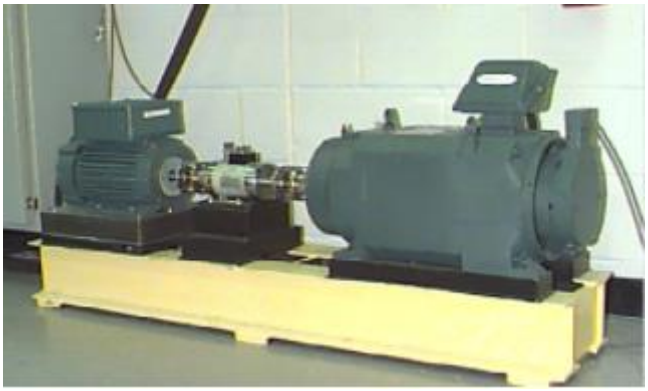
Os testes foram executados em três bases de dados de vibração para detecção de anomalia distintas.

4.1 BANCOS DE DADOS DE TESTE

Primeiramente, a base de dados CWRU consiste em dados de vibração coletados de um rolamento de um motor em laboratório. As falhas são implantadas nos motores a partir de máquinas de descarga elétrica com diâmetros de testes são executados com a carga do motor partindo de 0 a 3 HP (CASE WESTERN RESERVE UNIVERSITY, 2020).

A Tabela 1 mostra os diferentes modos de falhas e diâmetros utilizados. Os dados de vibração são coletados a partir de acelerômetros conectados ao equipamento (Figura 21), a vibração é coletada em dois pontos, na turbina superior e inferior do aparato, os dados são coletados a uma taxa de 12000 amostras por segundo.

Figura 21 – Bancada de Testes de Rolamento CWRU



Fonte: Case Western Reserve University, 2020b

Essa bancada de teste foi construída com o objetivo de construir banco de dados a partir de inserção dos principais modos de falhas presentes em rolamentos, que são em geral, componentes críticos de equipamentos rotativos (motores, rotores, geradores, compressores, bombas, entre outros) presentes em grande parte das indústrias (HENG *et al.*, 2009; LIU, Ruonan *et al.*, 2018).

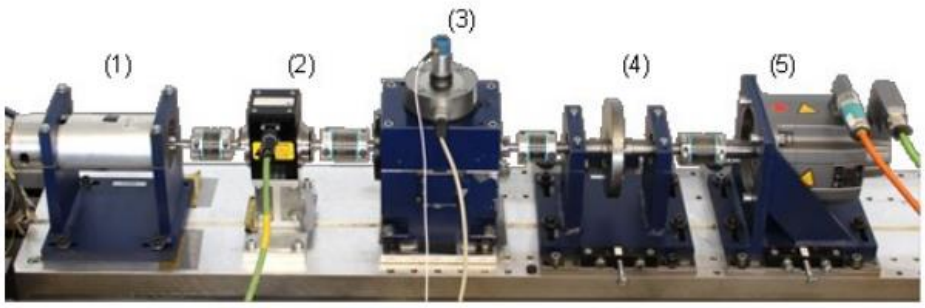
Tabela 1 – Simulação de Falhas na bancada de teste CWRU

Modo de Falha	**Diâmetro da Falha inserida**
Dados Saudáveis	0
Falha de pista interna de rolamento	0.007; 0.014; 0.021; 0.028
Falha de pista externa de rolamento	0.007; 0.014; 0.021
Falha em Esfera de rolamento	0.007; 0.014; 0.021; 0.028

Outra base de dados de diagnóstico de rolamentos utilizada nesse trabalho é fornecida pela sociedade de prevenção de falhas (MPFT – do inglês, Machinery Failure Prevention Technology). O conjunto de dados consiste em dados de amostragens distintas: três para rolamentos sob condição normal de operação, dez para falhas na pista externa do rolamento e sete para falhas na pista interna do rolamento (BECHHOEFER, 2018).

A terceira base de dados em que foi feita a aplicação dos modelos foram os dados de rolamento fornecidos pelo centro de dados de rolamento da Pareborn University (PU), os dados consitem 32 conjuntos de sinais de corrente e de vibração, os quais se dividem entre operação do rolamento sob condições saudáveis (6 conjuntos) e operações com falhas nas pistas internas e externas em localizações distintas inseridas tanto artificialmente nos rolamentos (12 conjuntos de dados), quanto por testes acelerados de vida (14 conjuntos de dados) (LESSMEIER *et al.*, 2016). A Figura 22 apresenta o aparato de testes e coletas de dados de falha dos rolamentos:

Figura 22 – Bacada de testes de rolamentos PU



Fonte: LESSMEIER et al., 2016

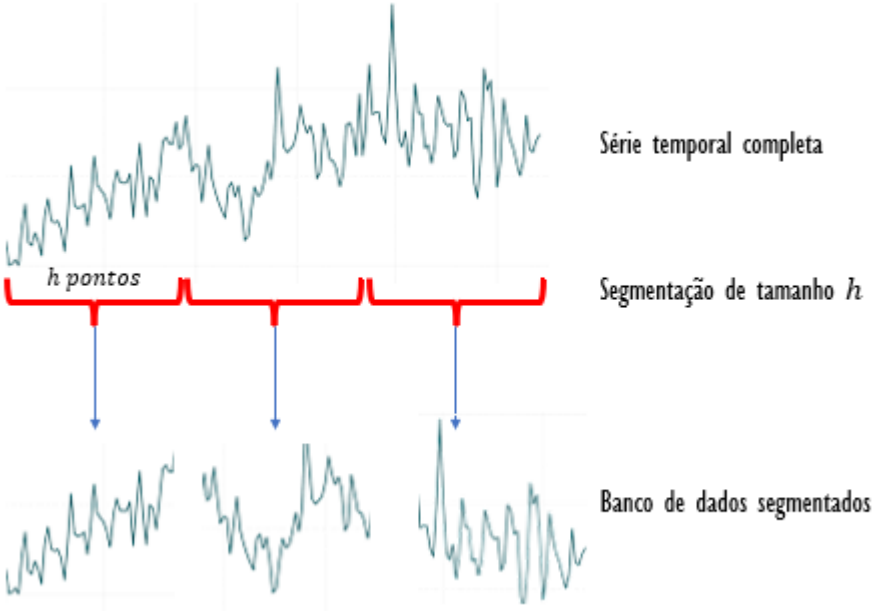
Os dados coletados nessas plataformas são temporais. Para executar os modelos de detecção de anomalia é necessário separar o dado temporal em vários vetores de um tamanho fixo.

4.2 PRÉ-PROCESSAMENTO

Como durante a coleta não há sobreposição de dados, esse processo é chamado de segmentação (GERON; GÉRON, 2017; SHUMWAY; STOFFER; STOFFER, 2000). A

Figura 23 ilustra o funcionamento da segmentação, para o qual foi aplicada uma janela de tamanho $h = 1024$, os dados extraídos pelo janelamento da série temporal foram, por fim, processados pelo algoritmo de transformada rápida de Fourier (FFT – do inglês – Fast Fourier Transform). Esse procedimento, consiste em uma transformação dos dados do domínio do tempo para o domínio das frequências e é amplamente utilizado na literatura referente ao diagnóstico por sinais de vibração de equipamentos rotativos (LIN, H. C.; YE, 2019).

Figura 23 – Processo de segmentação para treinamento de rede neural



Fonte: O Autor, 2022

Após o processo de segmentação, os dados passam por uma etapa de escalonamento, importante para evitar falhas por saturação de gradiente no processo de treinamento das redes neurais, para esse trabalho utilizou-se o seguinte procedimento:

$$x_t = \frac{x_t - \min(X)}{\max(X) - \min(X)}$$

em que $X \in x_t$ e representa o vetor com todos dados temporais dentro de uma segmentação $X = (x_1, x_2, \dots, x_h)$.

Para fazer a detecção de anomalia, é necessário separar os dados saudáveis dos anômalos. Portanto, os dados com inserção de falha no laboratório foram considerados anômalos, enquanto os dados sem inserção foram considerados saudáveis (Tabela 2). Além disso, é preciso dividir os dados para treinamento, validação e teste. Os dados de treinamento são utilizados para o treinamento dos modelos de aprendizagem de máquina, os dados de validação são utilizados para o estabelecimento do limiar do score de anomalia e os dados de teste são utilizados para o cálculo do F1-Score, o qual é uma medida de desempenho voltada para classificação binária de bases de dados pouco balanceadas.

A divisão dos dados é apresentada na Tabela 2 para todos os conjuntos de dados testados, os dados anômalos não fazem parte do treinamento do modelo de detecção de anomalia, portanto foram divididos em 50% para validação e 50% para teste. Paralelamente, os dados saudáveis foram divididos em 80% para treinamento 10% para teste e 10% para validação.

Tabela 2 - Divisão dos Dados CWRU, MPFT e PU

		Dados Anômalos	**Dados Saudáveis**		
		Teste	**Treino**	**Teste**	**Validação**
Número de observações no banco de dados	**Proporção**	**100%**	**80%**	**10%**	**10%**
	CWRU	7290	2550	852	852
	MPFT	4758	3513	439	439
	PU	2500	2400	300	300

4.3 ARQUITETURAS DE REDE NEURAL

As arquiteturas das redes neurais implementadas são definidas desde a Tabela 3 à Tabela 8. O Autoencoder foi definido a partir de uma arquitetura de MLP com camadas densas, além de camadas de dropout para garantir a regularização do modelo (Tabela 3). As camadas 1-5 representam a rede de *encoder*. As camadas 7-11 representam a arquitetura do decoder, observa-se a simetria entre as arquiteturas. Desta forma o encoder faz o processo de resumir os 800 pontos referentes ao janelamento das duas séries temporais em um espaço latente (camada 6) de dimensão 32.

Tabela 3 - Arquitetura de AE implementada

Camada	Neurônios	Tipo de camada	Função Ativação
1	512	Entrada	-
2	256	Densa	Selu
3	-	Dropout ($p = 0{,}3$)	-
4	128	Densa	Selu
5	128	Densa	Selu
6	32	Densa (Espaço Latente)	Selu
7	128	0,2	Selu
8	128	0,5	Selu
9	-	Dropout ($p = 0{,}3$)	-
10	256	Densa	Selu
11	512	Saída	-

A Tabela 4 apresenta a arquitetura da rede VAE implementada, foi adicionada uma camada convolucional. Essa camada permite uma avaliação posicional dos dados da rede temporal no processo de resumo dos dados em um espaço latente. No caso do VAE o espaço latente (camada 6) consiste em uma distribuição normal multivariada independente com 32 dimensões. As médias e desvios padrão são definidos pelos resultados das camadas anteriores. A saída da camada 6 para a camada 7 consiste em uma amostragem obtida dessas distribuições normais. As camadas 7 a 11 fazem um processo de decodificação e são simétricas ao encoder (camadas 1 - 5).

Tabela 4 - Arquitetura de VAE implementada

Camada	Neurônios/ Filtros	Tipo de camada	Função Ativação
1	512	Entrada	
2	4 filtros (32x1)	Conv1D	Relu
3	256	Densa	Selu
3	-	Dropout ($p = 0{,}3$)	-
4	128	Densa	Selu
5	128	Densa	Selu
6	32x32	Densa + Amostragem (Espaço Latente)	Selu
7	128	Densa	Selu
8	128	Densa	Selu
9	-	Dropout ($p = 0{,}3$)	-
10	256	Densa	Selu
11	512	Saída	-

A Tabela 5 e Tabela 6 apresentam as arquiteturas implementadas em conjunto para implementação da rede GAN. Uma rede GAN consiste na concatenação dessas duas arquiteturas: a rede geradora responsável pela replicação de dados; a rede discriminante responsável pela diferenciação entre dados gerados artificialmente. Observa-se que a sáida da rede geradora (Tabela 5) é de mesma dimensão da entrada da rede discriminante (Tabela 7).

Tabela 5 - Arquitetura de GAN (Gerador)

Camada	Neurônios/ Filtros	Tipo de camada	Função Ativação
0	-	Amostragem Normal ($\mu = 0, \sigma = 1$)	
1	32	Entrada	-
2	5 filtros (64x1)	Conv 1D	Relu
3	5 filtros (32x1)	Conv 1D	Relu
4	512	Saída	-

Tabela 6 - Arquitetura de GAN (Discriminante)

Camada	Neurônios/ Filtros	Tipo de camada	Função Ativação
1	512	Entrada	-
2	4 filtros (32x1)	Conv 1D	Relu
3	4 filtros (32x1)	Conv 1D	Relu
4	-	Dropout ($p = 0{,}3$)	-
5	100	Dense	-
6	1	Saída	Sigmoide

A Tabela 7 e Tabela 8 apresentam as arquiteturas implementadas para o AAE implementado neste trabalho. A rede neural é subdividida em um AutoEncoder e um discriminante, observa-se que a dimensão de entrada dessa rede é a mesma dimensão definida para o espaço latente do Autoencoder. A rede discriminante diferencia os dados originais representados no espaço latente de amostragens de uma distribuição fornecida a priori. Nesse caso, trata-se de uma distribuição normal multivariada independente de dimensão 32 com $\mu = 0$ e $\sigma = 1$.

Tabela 7 - Arquitetura de AAE (AutoEncoder)

Camada	Neurônios	Tipo de camada	Função Ativação
1	512	Entrada	-
2	256	Densa	Selu
3	-	Dropout ($p = 0{,}3$)	-
4	128	Densa	Selu
5	128	Densa	Selu
6	32	Densa (Espaço Latente)	Selu
7	128	0,2	Selu
8	128	0,5	Selu
9	-	Dropout ($p = 0{,}3$)	-
10	256	Densa	Selu
11	512\	Saída	-

Tabela 8 - Arquitetura de AAE (Discriminante)

Camada	Neurônios/ Filtros	Tipo de camada	Função Ativação
0	32	Amostragem	
1	32	Entrada	-
2	32	Dense	Relu
3	32	Dense	Relu
5	-	Dropout ($p = 0{,}3$)	-
6	32	Dense	Relu
7	1	Saída	Relu

Os parâmetros utilizados para o treinamento de cada um dos modelos são apresentados na Tabela 9, as funções perdas são definidas de acordo com cada modelo, e o número de épocas foi definido pela necessidade para cada modelo alcançar resultados favoráveis e de acordo com a convergência de cada modelo.

Tabela 9 – Parâmetros de treinamento

Modelo	Épocas	Algoritmo de otimização	Função *Loss*	Tamanho de *Batch*
AE	50	ADAM	MSE	256
VAE	50	ADAM	Divergência de Kullback	256
GAN	1000	ADAM	Entropia cruzada binária	256
AAE	500	ADAM	MSE	256

Os modelos foram implementados utilizando as bibliotecas Keras e TensorFlow, do Python, os quais foram executados em uma estação de trabalho com processador Intel(R) Core ™ i9-9900K, GPU Nvidia GeForce RTX 2080 Ti ™. No capítulo a seguir, são apresentados e discutidos os resultados obtidos nessa experimentação.

4.4 DETECÇÃO DE ANOMALIA E AVALIAÇÃO DE MODELO

Os modelos de redes neurais são treinados com dados de comportamento saudável e, posteriormente, são utilizados para a avaliação da pontuação de anomalia dos conjuntos de validação e testes, os dados de validação são utilizados para a construção de um limiar de anomalia, para isso

foi utilizado o quantil 95% da pontuação de anomalia entre os dados de validação, como apresenta a equação abaixo.

$$T\ (s) = [A(s, validação)]_{95\%}$$

Em que,

$T\ (s)$ – limiar de anomalia para o modelo s

$A(s, validação)$ – População de pontuações de anomalias para o modelo s no conjunto de $validação$

Após o treinamento dos modelos, calcula-se o F1-Score dos resultados obtidos no conjunto de teste de acordo com a fórmula a seguir:

$$F1 - Score\ = 2 * \frac{precision * recall}{precision\ +\ recall}, \text{em que}$$

$$Precision\ = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive}$$

$$Recall\ = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative}$$

4.5 SIMULAÇÃO DE MAU FUNCIONAMENTO EM SENSORES

Após os testes na base de dados original, é necessário avaliar os modelos para os casos com mau funcionamento de sensores. Portanto, avaliaram-se dois casos: o primeiro, em que há dados faltantes na série temporal, e o segundo, quando há presença de ruídos externos ao equipamento. Nesse último caso, considerou-se a presença de um dado normal. O dado do cenário simulado pode ser representado matematicamente pela fórmula:

$$s_{i,t}\ = x_t * m\ +\ \varepsilon$$

Em que s_t é o dado indexado no tempo t no cenário i simulado, m é uma variável binária que assume valor 0 com probabilidade p_i, valor que representa a probabilidade de o dado estar ausente naquela coleta e ε é um valor amostrado de uma normal com média 0 e desvio padrão σ_i. Foram testados nove cenários distintos com os valores apresentados na Tabela 10.

Tabela 10 – Cenários de dados faltantes e adição de ruído branco a sensores

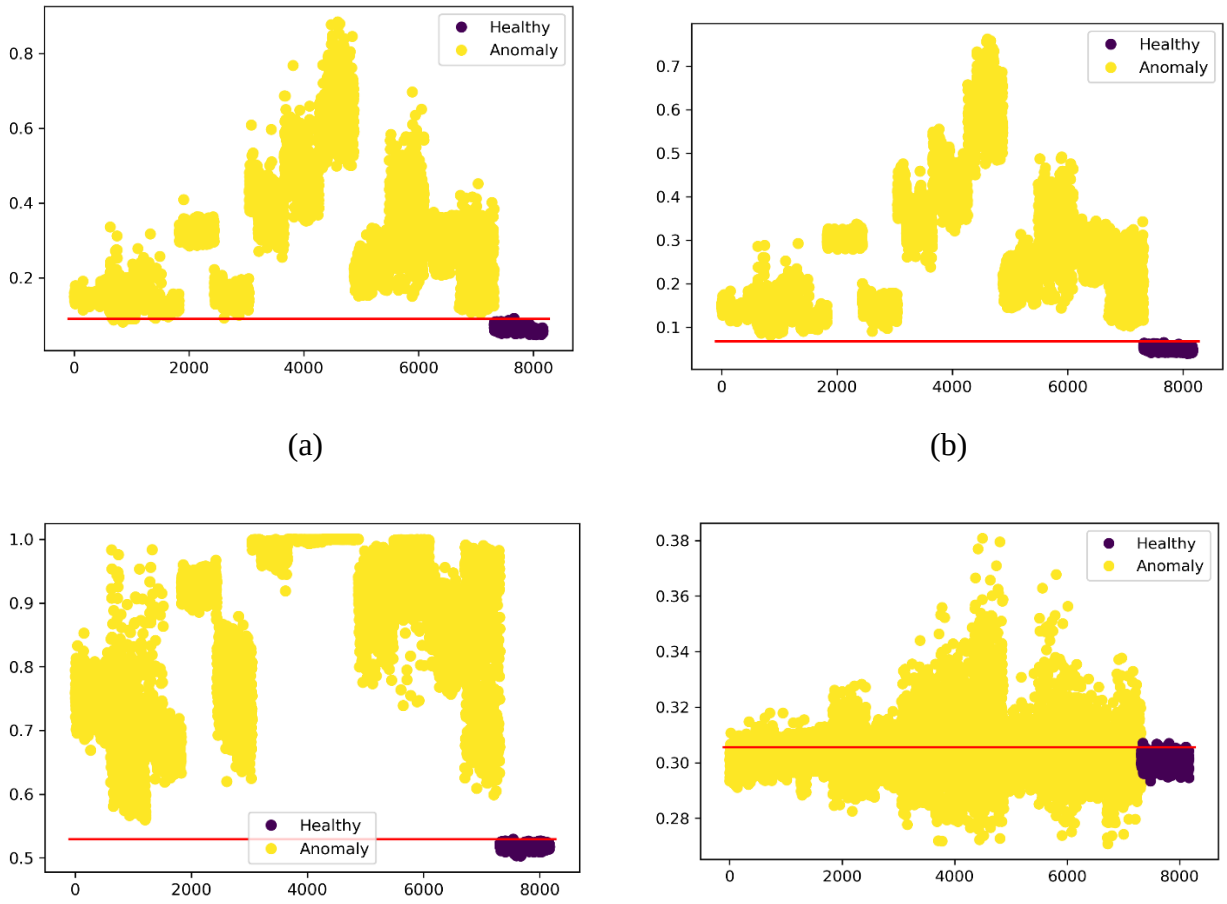
Cenário	***p***	**σ**
1	0,3	0
2	0,5	0
3	0,7	0
4	0	0,2
5	0	0,3
6	0	0,5
7	0,3	0,2
8	0,3	0,5
9	0,5	0,2

Os três primeiros cenários representam situações com dados faltantes. Os cenários 4-6 estão relacionados aos casos em que há ruido nos dados coletados. E, por último, os cenários 7-9 associam-se a sinais em que há tanto falta de dados como presença de ruido. Os modelos foram executados 30 vezes para cada um dos cenários e para a base de dados original. Dessa forma, buscou-se dar validade estatística aos resultados e investigar como se comporta a variabilidade da métrica de desempenho em cada um dos cenários.

5 RESULTADOS

Inicialmente, os modelos foram executados para a base de dados original, A Figura 24 apresenta exemplo de resultados para essa primeira etapa para o conjunto de dados CWRU. No gráfico de detecção de anomalia, os dados saudáveis estão representados em roxo e os dados referentes aos modos de falha apresentados na Tabela 11 estão em amarelo. Nota-se que os resultados para o *Autoencoder* Adversarial (Figura 24 - d) indicam incapacidade do modelo de generalizar a solução, caracterizando um subajuste para esse modelo. Como essa base de dados é considerada de baixo grau de dificuldade (ZHOU et al., 2021) e os cenários de problemas de sensoriamento causam redução na qualidade dos modelos, o teste para outras bases de dados e para os cenários de dados faltantes foi mantido apenas para os outros três modelos. Para os modelos de autoencoder e a rede GAN, os resultados foram satisfatórios pois há uma separabilidade dos dados pela linha vermelha.

Figura 24 – Gráficos de detecção de anomalia para conjunto de dados CWRU original segundo modelo (a) AutoEncoder (b) VAE (c) GAN (d) AAE

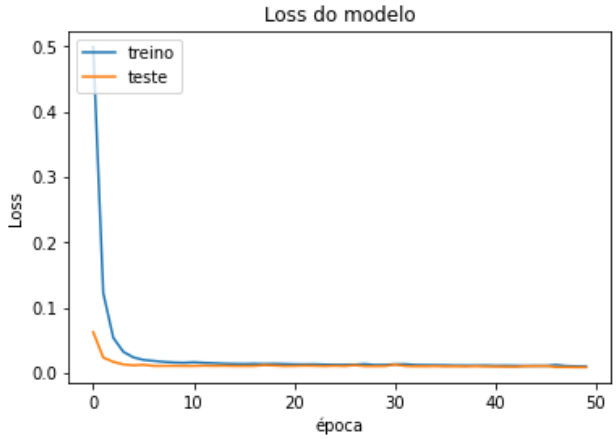


Fonte: O Autor, 2022

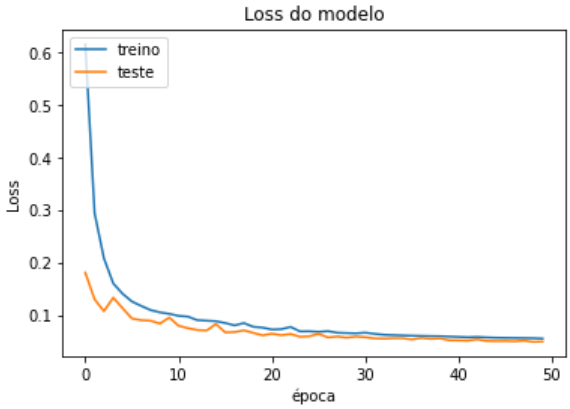
A Figura 25 apresenta o gráfico da convergência para os três modelos implementados para o conjunto de dados CWRU. Para os modelos AE e VAE propostos observa-se uma redução da função perda tanto para os conjuntos de treinamento, quanto para o de teste. Para o modelo GAN as perdas plotadas em gráfico dizem respeito à rede geradora e a rede discriminadora. Nas primeiras 200 iterações observa-se um grande aumento do erro da rede geradora em conjunto com queda do erro da rede discriminante.

Posteriormente, a rede geradora se adapta ao treinamento da rede discriminante, e o erro da rede geradora decresce. Os erros das redes discriminante e geradora passam a oscilar até estabilizar em uma faixa de 1000 épocas. A Figura 26 apresenta a convergência da replicação ao longo das iterações e um dado para o conjunto que original. Ao longo das iterações (Figura 26, a - e) o resultado de replicação se aproxima do formato dos dados originais (Figura 27, f). As imagens apresentam a replicação do modelo GAN para ambas séries temporais coletadas durante a coleta dos dados de vibração. Um dos pontos de coletas é referente a turbina (FE – do inglês – *Fan Ending*) e o outro ponto de coleta refere-se ao eixo de transmissão (DE – do inglês – *Drive Ending*).

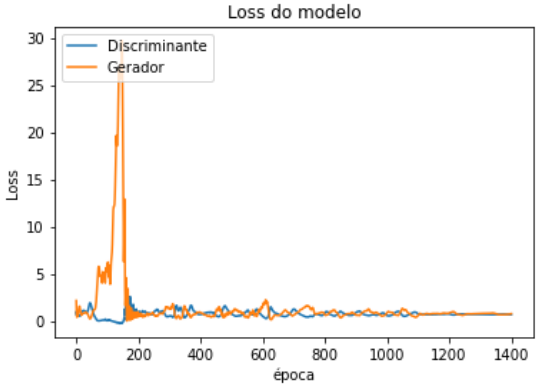
Figura 25 – Convergência da Função Loss do modelo para os conjuntos de conjunto de dados original segundo modelo (a) AutoEncoder (b) VAE (c) GAN



(a)



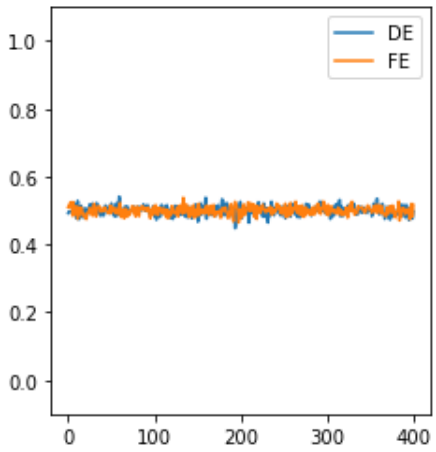
(b)



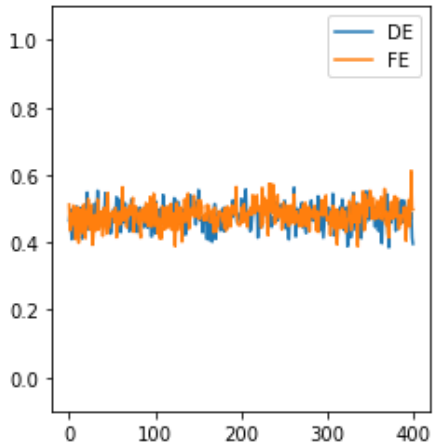
(c)

Fonte: O Autor, 2022

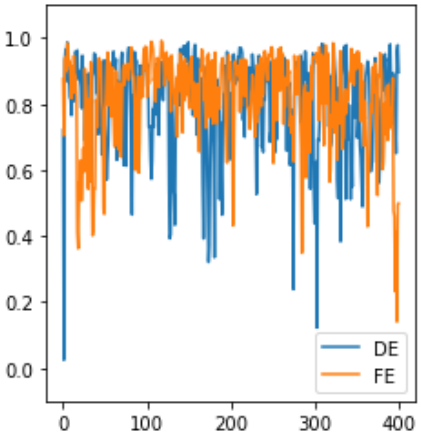
Figura 26 - Exemplos de replicação do dado de série temporal por número de épocas: (a) 1 época (b) 400 épocas (c) 800 épocas (d) 1000 épocas (e) 1500 épocas (f) Dados Originais



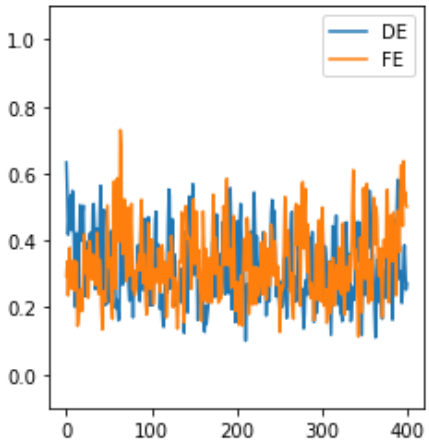
(a)



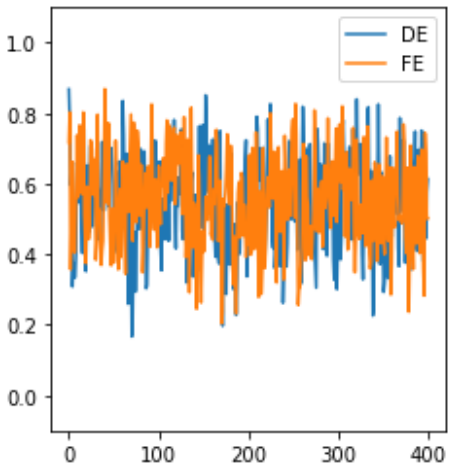
(b)



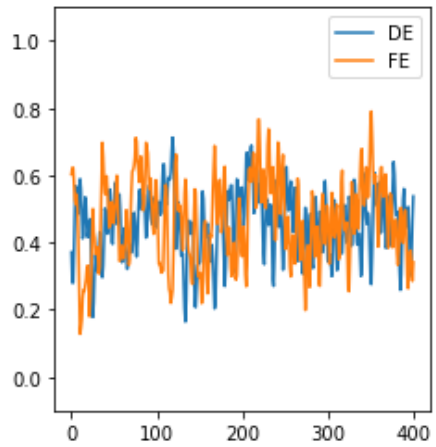
(c)



(d)



(e)

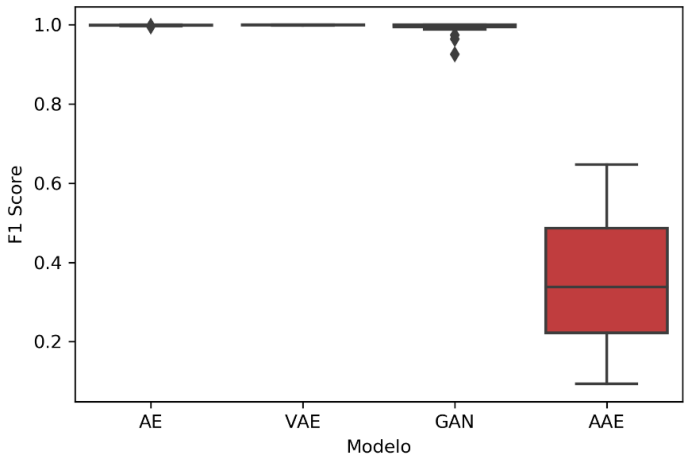


(f)

Fonte: O Autor, 2022

Após executar os modelos 30 vezes, é possível observar melhor os resultados obtidos em cada um deles. A Figura 27 apresenta o diagrama *boxplot* para os resultados na base original, o modelo de AAE é substancialmente pior que os outros por apresentar uma distribuição de F1-Score no conjunto de teste completamente abaixo das distribuições dos outros três modelos.

Figura 27 – BoxPlot de resultados sem alterações nos dados



Fonte: O Autor, 2022

A Tabela 11 apresenta as médias e os desvios padrões da pontuação F1-Score para cada um dos modelos, o modelo VAE apresentou resultados de média maior com menor desvio padrão, portanto, representa um modelo mais substancial para aplicação na base de dados avaliada.

Tabela 11 – Resultados de F1–Score para base de dados original para 30 replicações

	Autoencoder		**Autoencoder Adversarial**		**Autoencoder Variacional**		**Rede Generativa Adversarial**	
	média	**d.p.**	**média**	**d.p.**	**média**	**d.p.**	**média**	**d.p.**
F1-score validação	0,998824	0,0007	0,354342	0,15469	**0,99935**	0,0006	0,993273	0,014828
F1-score teste	0,998572	0,0007	0.354743	0,152556	**0,99935**	0,0006	0,993273	0,014828

Posteriormente, foram executados os testes nos cenários de dados faltantes e inserção de ruído branco (Tabela 10), os resultados de médias e desvio padrão para todos os cenários, bem como os *boxplots* dos F1-Score de teste, são apresentados na Tabela 12 e Figura 28, Figura 29, Figura 30.

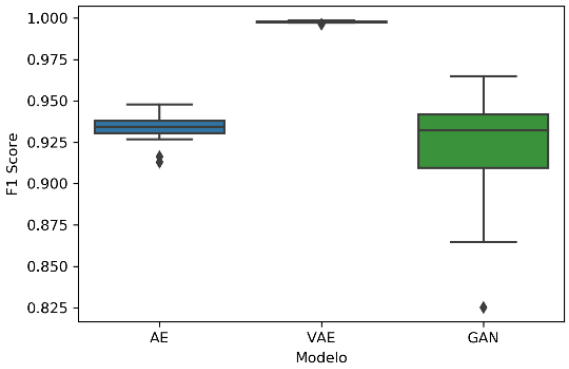
Tabela 12 – Resultados de F1–Score para cenários de problemas em coleta e armazenamento de dados para 30 replicações

			Autoencoder		**Autoencoder Variacional**		**Rede Generativa Adversarial**	
CENÁRIO	**p**	**σ**	**Média**	**d.p.**	**média**	**d.p.**	**média**	**d.p.**
7	0,3	0,2	0,901691	0,009143	**0,954606**	0,009089	0,871094	0,039739
8	0,3	0,5	0,779652	0,009255	**0,801578**	0,008767	0,685409	0,124198
9	0,5	0,2	0,820367	0,011661	0,814996	0,007129	**0,824304**	0,029245

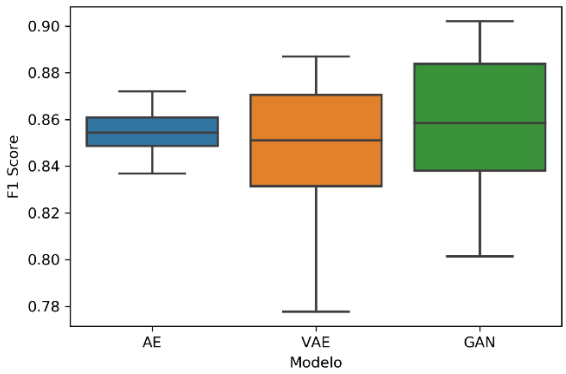
Os primeiros três cenários representam dados faltantes no sensoriamento, com um aumento gradativo da probabilidade de cada ponto de coleta ser ausente. Nota-se um comportamento distinto de cada um dos modelos diante dessa adversidade. No primeiro dos cenários o modelo VAE ainda apresenta a maior média de F1-Score com o menor desvios padrão.

No entanto, ao longo que o parâmetro é ampliado, verifica-se grande queda na avaliação de qualidade do modelo de Variational Autoencoder, estando substancialmente abaixo dos outros dois modelos. Desta forma, para o cenário 4 o desempenho do VAE é muito baixo e alcança uma média de F1-Score de 0,42172. A Figura 28 apresenta os diagramas *boxplot* dos cenários 1, 2, 3. É possível notar a evolução dos boxplots em relação ao valor de p. A Figura 29 (c) representa o cenário 9, em que há uma alta probabilidade de dado faltante com uma presença menor de ruído.

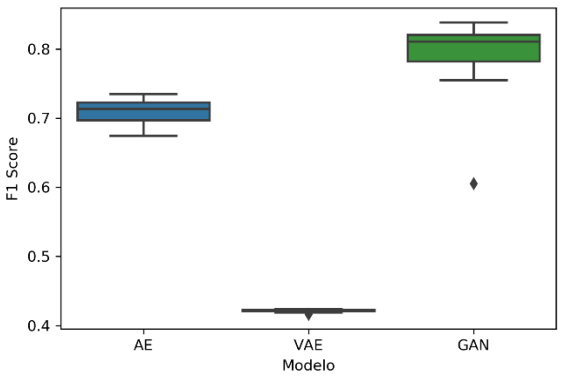
Figura 28 – BoxPlot para os cenários 1 (a), 2 (b) e 3 (c)



(a)



(b)



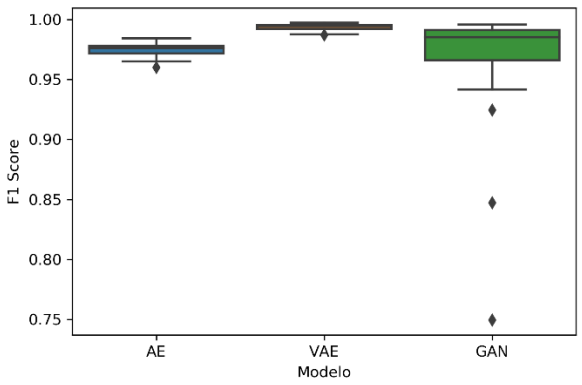
(c)

Fonte: O Autor, 2022

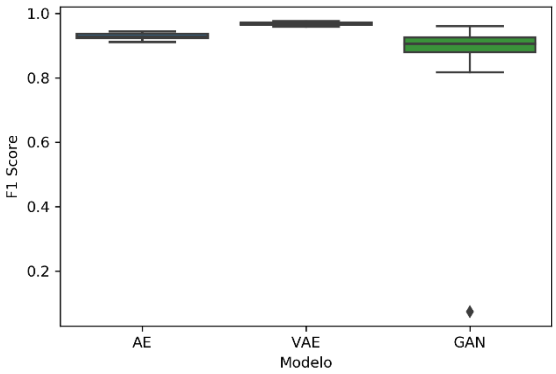
Os próximos 3 cenários abordam a inserção de ruído nos dados. As quedas de precisão dos modelos são maiores para os resultados do Autoencoder e das GANs, é possível observar essa evolução na Figura 29, indicando que no caso desse problema de sensoriamento, o modelo VAE testado apresenta maior robustez. A robustez do Variational Autoencoder em relação ao ruído normal pode ser associada ao algoritmo já levar em consideração a inserção de diversidade e amostragem aleatórias durante o processo de replicação dos dados originais.

Além disso, é possível encontrar estudos que associam a robustez do VAE ao processo de cálculo das covariâncias do espaço latente e a possibilidade de redução de dimensões desnecessárias no espaço latente (DAI *et al.*, 2018). É possível encontrar estudos, que adaptam o Variational AutoEncoder para torná-lo mais robusto a dados poluídos por ruídos. (AKRAMI *et al.*, 2022; GAO *et al.*, 2020; HSU; ZHANG, Y.; GLASS, 2017)

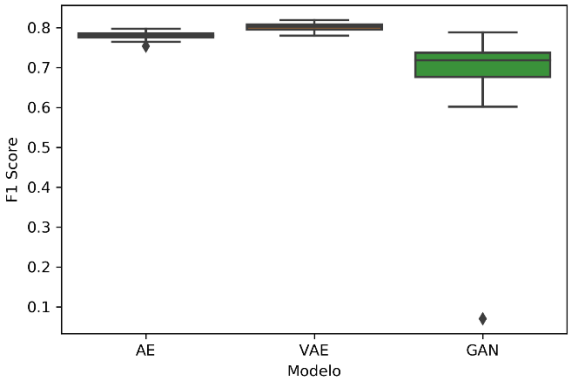
Figura 29 – BoxPlot para os cenários 4(a), 5 (b) e 6 (c)



(a)



(b)

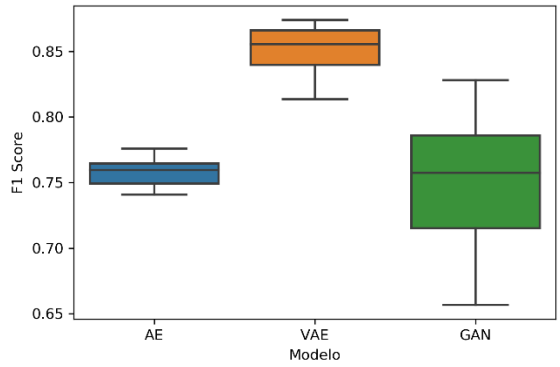


(c)

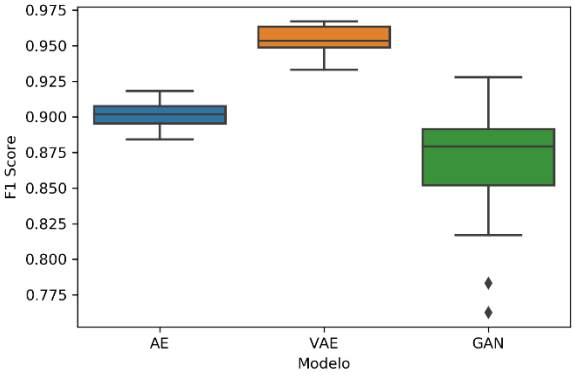
Fonte: O Autor, 2022

A Figura 30 apresenta os resultados para os cenários em que ambos os problemas são inseridos aos dados, para 3 dos 4 cenários o modelo *Variational Autoencoder* apresenta resultados mais satisfatórios que a GAN e o *autoencoder*. Além disso, em todos os diagramas de *boxplot* é possível observar a medida de desempenho da detecção de anomalia via GAN apresenta uma dispersão maior que os demais modelos. Para o cenário 9, no qual a GAN apresenta uma média maior que VAE, porém há uma diferença muito pequena entre as médias, e a dispersão dos resultados para VAE são substancialmente menores.

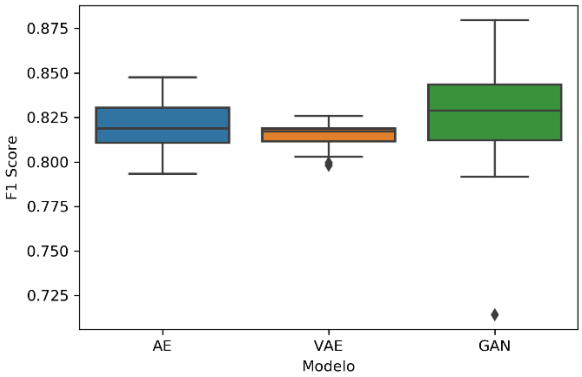
Figura 30 - BoxPlot para os cenários 7 (a), 8 (b), e 9 (c)



(a)



(b)

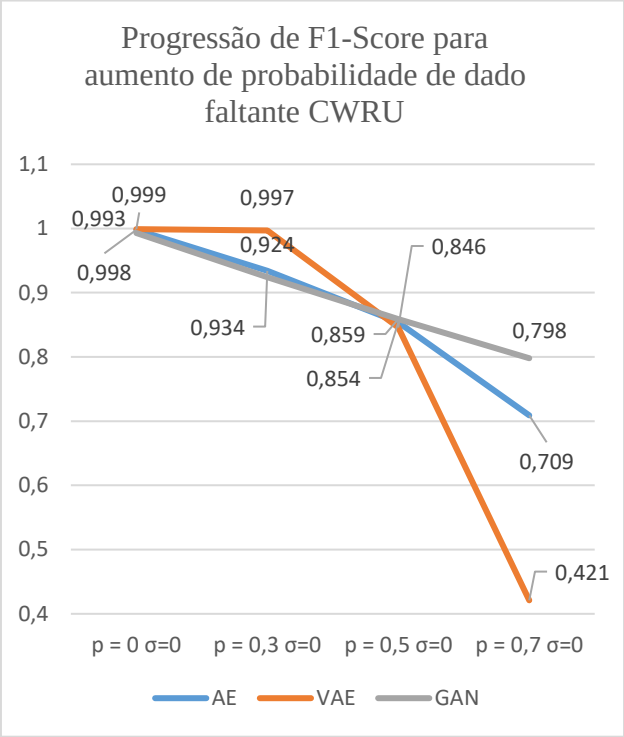


(c)

Fonte: O Autor, 2022

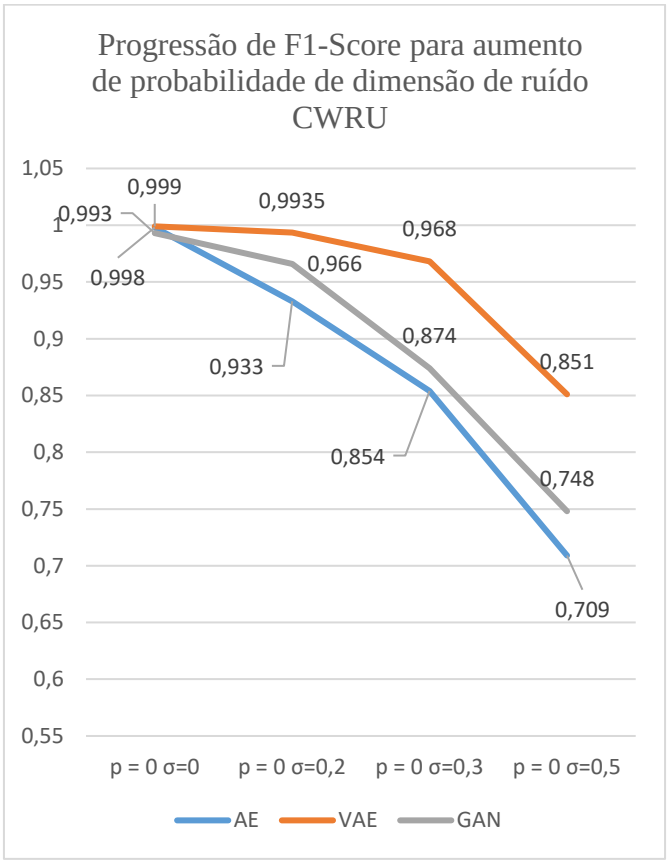
As relações das medidas de qualidade dos modelos com a variação dos parâmetros de dados faltantes e ruídos são sumarizadas pelos gráficos apresentados na Figura 31 e Figura 32. É possível observar a queda abrupta do F1-Score médio do modelo VAE à medida que há um aumento da proporção de dados faltantes, enquanto o modelo GAN possui uma queda visualmente linear para o valor médio. Além disso, é possível observar que o modelo VAE é consistentemente melhor que os outros dois propostos para lidar com inserção de ruídos no conjunto CWRU.

Figura 31 - Evolução de Valor Médio de F1-Score para dados CWRU com aumento de dados faltantes



Fonte: O Autor, 2022

Figura 32 - Evolução de Valor Médio de F1-Score para dados CWRU com aumento de ruídos de sensoriamento

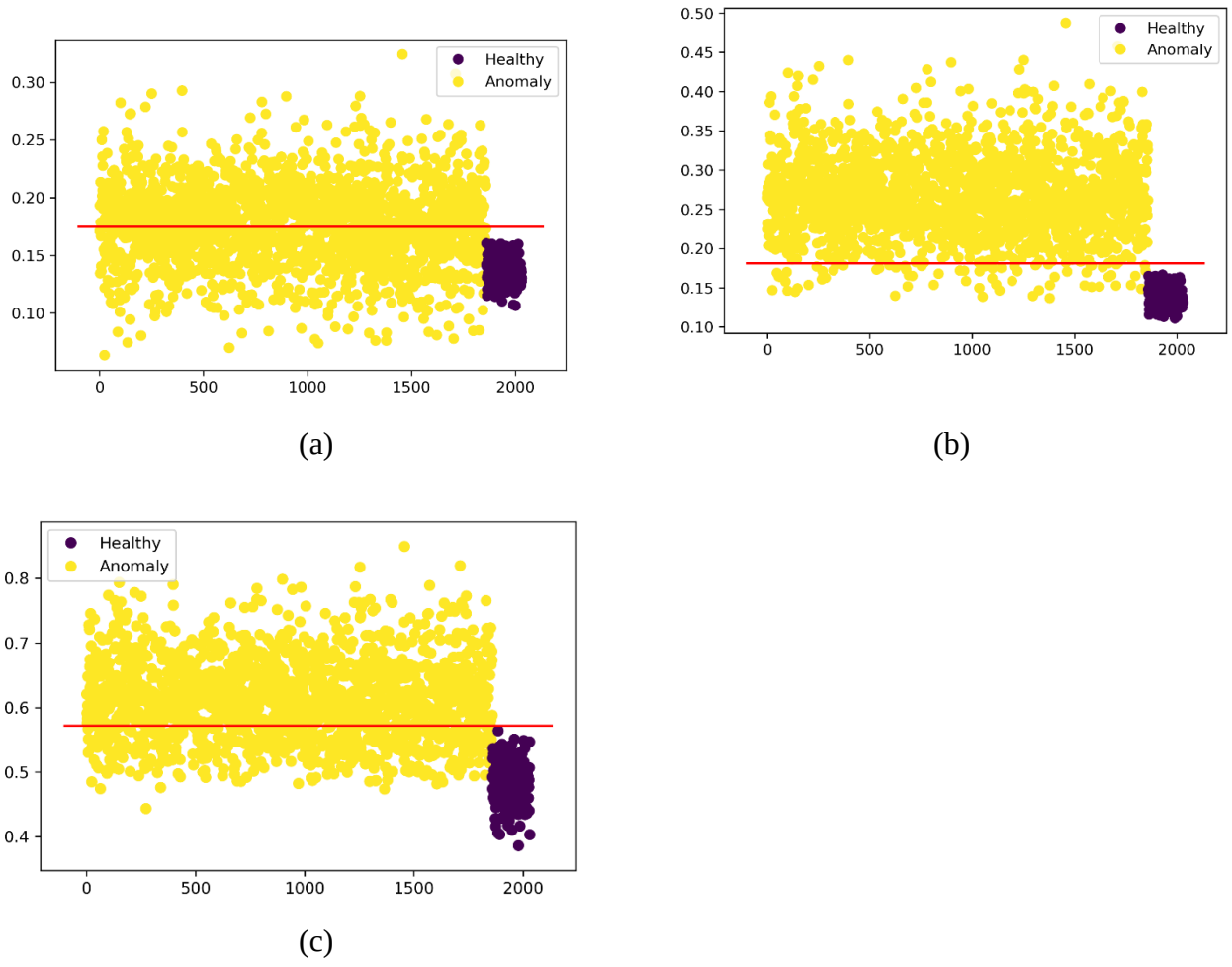


Fonte: O Autor, 2022

Posteriormente foram avaliados os mesmo modelos para conjuntos de variáveis distintos e com um maior grau de dificuldade de classificação. Devido à maior dificuldade dos dados foi necessário aplicação de pré-processamento via FFT para obtenção de resultados satisfatórios para os modelos.

Ao analisar os resultados para o segundo conjunto de dados, o MFPT, é possível perceber que os modelos tiveram mais dificuldade para lidar com esse conjunto. Isso é visível na Figura 33, onde é visto que dados com anomalia ficaram abaixo da linha vermelha para os três modelos testados. Assim, a detecção de anomalia obteve menos sucesso do que para o conjunto CWRU.

Figura 33 – Gráficos de detecção de anomalia para conjunto de dados MFPT original segundo modelo (a) AutoEncoder (b) VAE (c) GAN



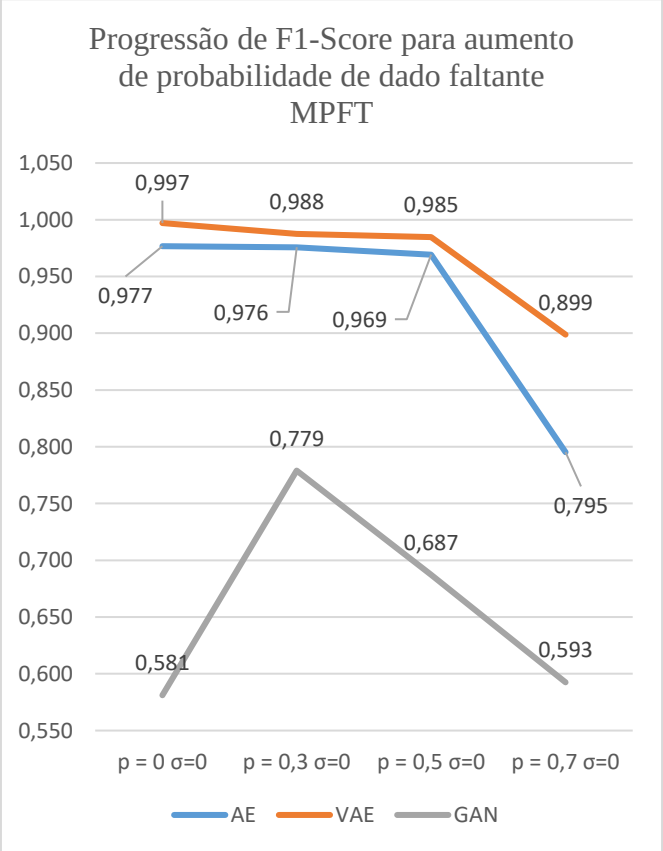
Fonte: O Autor, 2022

A Tabela 13 apresenta os resultados para a metodologia proposta quando analisada base de dados MPFT, é possível observar um decréscimo do desempenho a medida que a probabilidade de dado faltante e dimensão do ruído de sensoriamento o modelo de autoencoder supera os outros

modelos no cenário sem adição de ruído ou dados faltantes, no entanto o modelo de autoencoder variacional possui melhores resultados que os outros dois modelos à medida que aumenta.

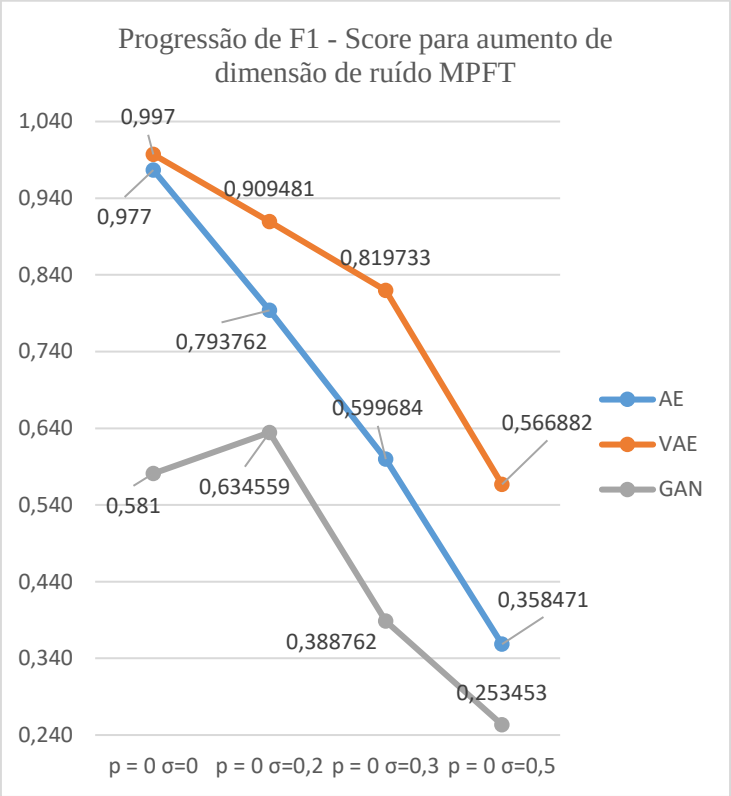
Para esse conjunto de dados, o modelo de GAN não tem desempenho melhor em nenhum dos cenários, indicando uma perda grande da qualidade dessa abordagem para bases de dados mais complexas. Além das médias de valor de F1-Score abaixo do valor dos outros modelos, é possível observar um desvio padrão substancialmente maior para esse modelo que para os outros dois concorrentes. O desvio padrão do F1-Score do modelo GAN varia entre 0,27 e 0,45 para uma métrica cujos valores são limitados entre 0 e 1. Dessa forma, é possível concluir que esse modelo é o mais instável entre os considerados. Observa-se uma necessidade de mais épocas de treinamento para que se obtenha maior consistência entre os resultados. A Figura 34 e Figura 35 apresentam a evolução das medidas de F1-Score de acordo com a evolução dos cenários. É possível observar uma robustez maior dos modelos em relação a dados faltantes que para o aumento da dimensão do ruído simulado em sensoriamento.

Figura 34 – Evolução de Valor Médio de F1-Score para dados MFPT com aumento de dados faltantes



Fonte: O Autor, 2022

Figura 35 - Evolução de Valor Médio de F1-Score para dados MFPT com aumento de ruídos de sensoriamento



Fonte: O Autor, 2022

Observa-se, para essa base de dados, maior robustez dos modelos em relação ao aumento do volume de dados faltantes que para a adição de ruído branco. A diferença entre as médias para o modelo VAE do cenário 0 e do cenário 3 é de 0,098. Em contrapartida a diferença para os cenários 0 e 6 é de 0,43.

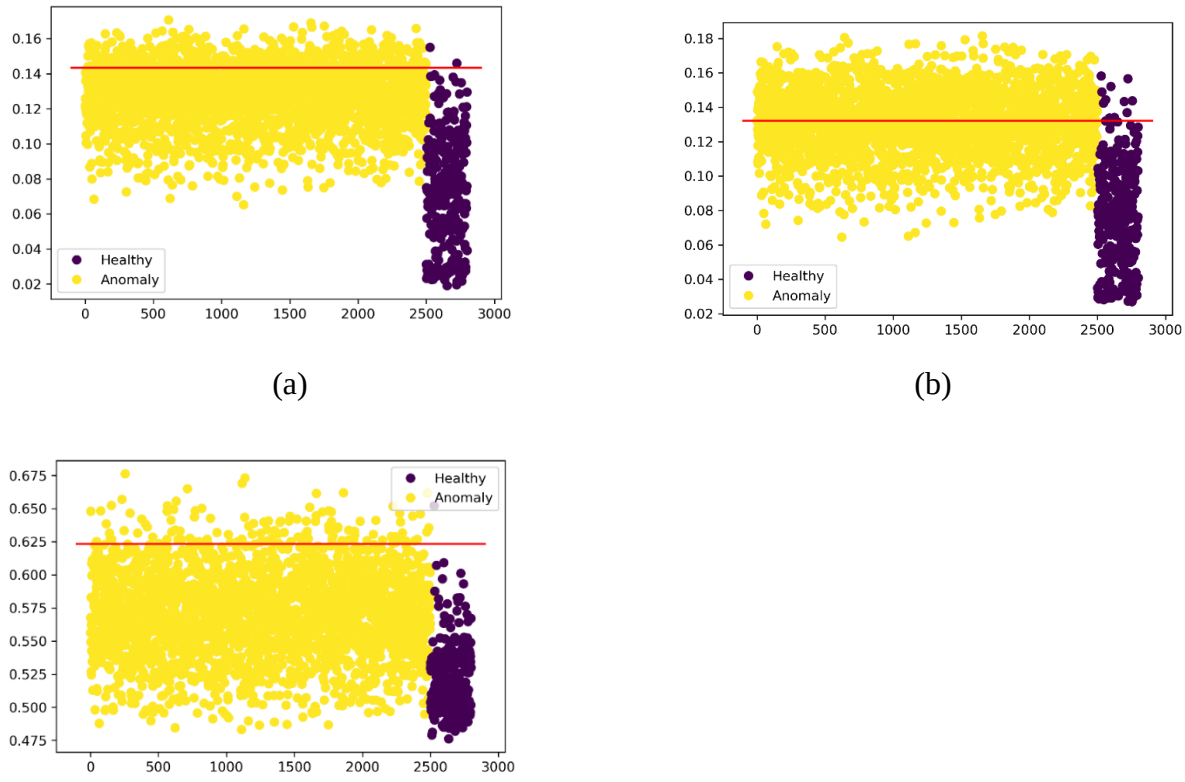
Tabela 13 – Resultados de F1–Score para cenários de problemas em coleta e armazenamento de dados para 30 replicações

			AUTOENCODER		**VARIATIONAL AUTOENCODER**		**GENERATIVE ADVERSARIAL NETWORK**	
CENÁRIO	**p**	**σ**	**Média**	**d.padrão**	**média**	**d.padrão**	**média**	**d.padrão**
7	0,3	0,2	0,995731	0,001951	**0,993466**	0,004834	0,508046	0,442785
8	0,3	0,5	0,533754	0,043908	**0,662418**	0,02929	0,362813	0,308288
9	0,5	0,2	0,979811	0,011532	**0,989209**	0,005869	0,573374	0,430381

Para o outro conjunto de dados mais complexo, o PU, o resultado da separação é o menos assertivo (Figura 36), pois muitos dados anômalos acabaram se misturando com os saudáveis na zona

abaixo da linha. Além disso, para esse conjunto de dados, foram detectados resultados de falso positivo, visto que há dados saudáveis localizados acima da linha vermelha, o que significa que eles foram erroneamente classificados como anômalos. Esse comportamento não foi identificado nos demais conjuntos analisados anteriormente.

Figura 36 – Gráficos de detecção de anomalia para conjunto de dados PU original segundo modelo (a) AutoEncoder (b) VAE (c) GAN



Fonte: O Autor, 2022

A Tabela 14 apresenta os resultados para o conjunto de dados PU, e nela pode-se observar que os resultados foram baixos para todos modelos em todos cenários representados, e que os modelos de autoencoder variacional apresentaram resultados melhores em cada um dos cenários. No entanto, os resultados indicam dificuldade dos modelos de alcançar convergência para os dados PU. Portanto, conclui-se que os modelos propostos não foram capazes de generalizar os resultados para essa base de dados.

Tabela 14 – Resultados de F1–Score para cenários de problemas em coleta e armazenamento de dados para 30 replicações

			AUTOENCODER		**VARIATIONAL AUTOENCODER**		**GENERATIVE ADVERSARIAL NETWORK**	
CENÁRIO	**p**	**σ**	**média**	**d.padrão**	**média**	**d.padrão**	**média**	**d.padrão**
0	0	0	0,317888	0,086435	**0,363812**	0,073427	0,198292	0,256206
1	0,3	0	0,225177	0,040373	**0,281605**	0,036524	0,152853	0,23357
2	0,5	0	0,072842	0,027505	**0,131691**	0,036004	0,101861	0,164555
3	0,7	0	0,149243	0,032313	**0,198892**	0,027947	0,123053	0,175051
4	0	0,2	0,054458	0,029662	**0,189351**	0,059931	0,166155	0,226726
5	0	0,3	0,051579	0,012829	**0,119652**	0,029635	0,075507	0,079716
6	0	0,5	0,023287	0,008282	**0,103429**	0,034822	0,104505	0,20477
7	0,3	0,2	0,094497	0,022228	**0,248799**	0,040661	0,188483	0,214086
8	0,3	0,5	0,047029	0,013771	**0,144893**	0,018965	0,113319	0,194431
9	0,5	0,2	0,040781	0,008978	**0,155306**	0,034094	0,254765	0,288402

6 CONCLUSÃO

A manutenção preditiva e baseada na condição é amplamente discutida na literatura e no contexto da indústria em nível mundial, com o barateamento e acesso a equipamentos e mão-de-obra especializada para implementação de instrumentos industriais e sistemas de informação. Experimenta-se, na atualidade, uma grande disponibilidade de dados de operação e condição de ativos. Portanto, amplia-se a discussão da aplicação de conceitos de *Big Data* e *Analytics* para o contexto de gestão da manutenção, entre eles é o uso de algoritmos de inteligência artificial e aprendizagem de máquina para PHM.

Nesse contexto, a qualidade do sensoriamento e armazenamento de dados é um grande desafio encontrado em cenários industriais, sendo, muito comum em dados temporais a presença de dados faltantes ou interferência de ruídos externos. Esse trabalho faz um estudo de quatro modelos distintos de detecção de anomalia (AE, VAE, GAN e AAE) diante desses dois desafios encontrados em bancos de dados de contexto real.

Foram utilizados dados *open source* de bancadas de rolamento, um equipamento costumeiramente crítico em equipamentos rotativos (ELASHA *et al.*, 2019; SOUALHI; MEDJAHER; ZERHOUNI, Noureddine, 2015; TABRIZI *et al.*, 2015). Verifica-se, portanto, a necessidade de testar a aplicabilidade dos modelos propostos e a relação das dificuldades de sensoriamento com a qualidade de resultados e robustez de modelos.

Foram testados resultados dos modelos de rede neural: *Autoencoder, Variational Autoencoder, Generative Adversarial Network* e *Adversarial Autoencoder*. Foram construídos nove cenários que envolviam distintas proporções de dados removidos do banco de dados original e inserção de ruído via amostragem de variável com distribuição normal. Para o banco de dados avaliado, ainda sem nenhuma alteração implementada, foi possível observar que *o Variational Autoencoder* apresentou resultados melhores, com a maior média de F1-Score e menor dos desvios padrão. O modelo de *Adversarial Autoencoder* implementado não foi capaz de generalizar uma resposta apropriada de detecção de anomalia, portanto decidiu-se por não realizar os testes deste modelo nos dados com simulação de falhas de sensoriamento.

Zhao et al. (2021) em um estudo de benchmarking de conjunto de dados de vibração para classificação de saúde de equipamentos rotativos apresenta o conjunto de dados CWRU como nível 1 de dificuldade e os conjuntos MPFT e PU como nível 3 de dificuldade, no entanto os resultados foram satisfatórios apenas para os conjuntos MPFT e CWRU, o pipeline proposto não foi eficiente para detectar anomalia para os dados PU.

Entre os conjuntos de dados em que houve boa convergência dos modelos, observa-se maior robustez para os dados CWRU; esse resultado condiz com o esperado dada diferença de dificuldade

de classificação entre as duas bases. Para os cenários simulados no conjunto CWRU, é possível observar que quanto maior a probabilidade de dados ausentes maior é a qualidade da detecção de anomalia por GAN em relação aos outros dois modelos testados. Em contrapartida, a resposta do modelo VAE para ruídos no banco de dados é substancialmente melhor, o modelo de AE clássico não sobressaiu em relação aos outros em nenhum dos cenários avaliados. Exceto em casos com alta proporção de dados faltantes é recomendado, para essa base de dados e entre os modelos testados, o uso do modelo VAE por apresentar menor desvio padrão da qualidade e maiores desempenhos médios.

Os dados utilizados no procedimento implementado, são considerados uma base de fácil separação dentre as bases de dados comumente utilizadas na literatura (ZHAO, Z. *et al.*, 2020). Portanto, em trabalhos futuros, é possível avaliar a robustez de modelos de detecção de anomalia para bases de dados mais complexas e de outros contextos de PHM. Além do uso de bases de dados de *benchmarking*, para alcançar um produto tecnológico é necessário avaliar o desempenho desses modelos em contextos de bancos de dados reais, em especial com problemas de dados faltantes, ou inserção de ruídos à coleta de dados.

Além disso, é possível também avaliar a robustez de outros modelos de redes neurais já apresentados na literatura, como forma de solução para o problema de detecção semi-supervisionada de anomalia de PHM. Dentre eles: Autoencoders em conjunto com redes regressivas (MALHOTRA *et al.*, 2016), ou o uso de redes recorrentes estocásticas (SU *et al.*, 2019). Ainda, os modelos apresentados nesse trabalho foram relacionados a apenas uma das possíveis atividades de PHM, a detecção de anomalia, é necessário também avaliar a robustez de modelos de redes neurais para outras atividades de PHM, por exemplo, diagnóstico de falha, construção de indicadores de saúde e prognóstico de falhas.

REFERÊNCIAS

ABBASI, T.; LIM, K. H.; YAM, K. S. Predictive Maintenance of Oil and Gas Equipment using Recurrent Neural Network. [S.l.]: [s.n.], 2019. V. 495.

AGGARWAL, C. C. **Neural Networks and Deep Learning: A Textbook**. [S.l.]: [s.n.], 2018.

AHMAD, R.; KAMARUDDIN, S. An overview of time-based and condition-based maintenance in industrial application. **Computers and Industrial Engineering**, 2012. v. 63, n. 1, p. 135–149.

AKRAMI, H. *et al.* A robust variational autoencoder using beta divergence. **Knowledge-Based Systems**, 28 fev. 2022. v. 238, p. 107886. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/S0950705121010534>. Acesso em: 19 jan. 2022.

ALLA, S.; ADARI, S. K. **Beginning Anomaly Detection Using Python-Based Deep Learning**. USA: Apress, 2019.

AYO-IMORU, R. M.; CILLIERS, A. C. A survey of the state of condition-based maintenance (CBM) in the nuclear power industry. **Annals of Nuclear Energy**, 2018. v. 112, p. 177–188.

BANK, D.; KOENIGSTEIN, N.; GIRYES, R. Autoencoders. **arXiv preprint arXiv:2003.05991**, 2020.

BECHHOEFER, E. MFPT - Bearing Dataset. 2018. Disponível em: <https://www.mfpt.org/fault-data-sets/>. Acesso em: 27 fev. 2022.

BEGGEL, L.; PFEIFFER, M.; BISCHL, B. Robust Anomaly Detection in Images Using Adversarial Autoencoders. Cham: Springer, 2020. V. 11906 LNAI, p. 206–222.

BEN-DAYA, M. *et al.* **Handbook of maintenance management and engineering**. [S.l.]: Springer London, 2009.

BERGHOUT, T. *et al.* A deep supervised learning approach for condition-based maintenance of naval propulsion systems. **Ocean Engineering**, 2021. v. 221, p. 108525.

CASE WESTERN RESERVE UNIVERSITY. Apparatus & Procedures | Case School of Engineering | Case Western Reserve University. 2020. Disponível em: <https://engineering.case.edu/bearingdatacenter/apparatus-and-procedures>. Acesso em: 11 dez. 2021.

CERRADA, M.; SÁNCHEZ, R. V.; CABRERA, D. A semi-supervised approach based on evolving clusters for discovering unknown abnormal condition patterns in gearboxes. **Journal of Intelligent and Fuzzy Systems**, 2018. v. 34, n. 6, p. 3581–3593.

CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. **ACM computing surveys (CSUR)**, 2009. v. v. 41, n., n. July 2009, p. 1– 58,.

CHENARIYAN NAKHAEE, M. *et al.* The Recent Applications of Machine Learning in Rail Track Maintenance: A Survey. Cham: Springer, 2019. V. 11495 LNCS, p. 91–105.

CHOLLET, F. **Deep Learning with Phyton**. [S.l.]: [s.n.], 2018.

CORADDU, A. *et al.* Machine learning approaches for improving condition-based maintenance of naval propulsion plants. **Proceedings of the Institution of Mechanical Engineers Part M: Journal of Engineering for the Maritime Environment**, 2016. v. 230, n. 1, p. 136–153.

DAI, B. *et al.* Connections with Robust PCA and the Role of Emergent Sparsity in Variational Autoencoder Models. **Journal of Machine Learning Research**, 2018. v. 19, p. 1–42. Disponível em: <http://jmlr.org/papers/v19/17-704.html.>. Acesso em: 19 jan. 2022.

DAVID, M. *et al.* A Review and Taxonomy of Interactive Optimization Methods in Operations Research. **ACM Transactions on Interactive Intelligent Systems (TiiS)**, 23 set. 2015. v. 5, n. 3. Disponível em: <https://dl.acm.org/doi/abs/10.1145/2808234>. Acesso em: 11 dez. 2021.

DOGO, E. M. *et al.* A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks. **Proceedings of the International Conference on Computational Techniques, Electronics and Mechanical Systems, CTEMS 2018**, 1 dez. 2018. p. 92–99. . Acesso em: 11 dez. 2021.

DONG, M.; HE, D. A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology. **Mechanical Systems and Signal Processing**, 2007. v. 21, n. 5, p. 2248–2266.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern Classification**. 2. ed. New York: Wiley, 2001.

EID, A. *et al.* A novel deep clustering method and indicator for time series soft partitioning. **Energies**, 2021. v. 14, n. 17.

ELASHA, F. *et al.* Prognosis of a wind turbine gearbox bearing using supervised machine learning. **Sensors (Switzerland)**, 2019. v. 19, n. 14, p. 3092.

FERNANDES, J. *et al.* The role of industry 4.0 and bpmn in the arise of condition-based and predictive maintenance: a case study in the automotive industry. **Applied Sciences (Switzerland)**, 2021. v. 11, n. 8.

FERREIRA, W. **Detecção de Imagens Falsificadas Baseada em Descritores Locais de Textura e Rede Neural Convolucional**. Goiânia: Universidade Federal de Goiás, 2020.

FOSTER, D. **Generative deep learning : teaching machines to paint, write, compose, and play**. [S.l.]: [s.n.], 2019. V. 1.

GAO, Y. *et al.* RVAE-ABFA : Robust Anomaly Detection for HighDimensional Data Using Variational Autoencoder. **Proceedings - 2020 IEEE 44th Annual Computers, Software, and Applications Conference, COMPSAC 2020**, 1 jul. 2020. p. 334–339. . Acesso em: 19 jan. 2022.

GÉRON, A. **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow - Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd Edition (2019)**. Sebastopol, CA: O'Reilly Media. ISBN, 2011. V. 44.

GERON, A.; GÉRON, A. **Hands-On Machine Learing With Scikit-Learn & Tensor Flow**. [S.l.]: [s.n.], 2017.

GOHEL, H. A. *et al.* Predictive maintenance architecture development for nuclear infrastructure using machine learning. **Nuclear Engineering and Technology**, 2020. v. 52, n. 7, p. 1436–1442.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning An MIT Press Book**. [S.l.]: [s.n.], 2016. V. 29.

GOODFELLOW, I. J. *et al.* Generative adversarial nets. [S.l.]: [s.n.], 2014. V. 3.

GOURIVEAU, R.; RAMASSO, E.; ZERHOUNI, Noureddine. Strategies to face imbalanced and unlabelled data in PHM applications. **Chemical Engineering Transactions**, 2013. v. 33, p. 115–120.

HENG, A. *et al.* Rotating machinery prognostics: State of the art, challenges and opportunities. **Mechanical Systems and Signal Processing**, 1 abr. 2009. v. 23, n. 3, p. 724–739. . Acesso em: 11 dez. 2021.

HSU, W. N.; ZHANG, Y.; GLASS, J. Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. **2017 IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2017 - Proceedings**, 2 jul. 2017. v. 2018-January, p. 16–23. . Acesso em: 19 jan. 2022.

JAMES, G.; WITTEN, D.; HASTIE, T. **Introduction to Statistical Learning with Applications in R**. [S.l.]: [s.n.], 2019. V. 11.

JANIESCH, C.; ZSCHECH, P.; HEINRICH, K. Machine learning and deep learning. **Electronic Markets**, 1 set. 2021. v. 31, n. 3, p. 685–695. Disponível em: <https://link.springer.com/article/10.1007/s12525-021-00475-2>. Acesso em: 13 set. 2022.

JARDINE, A. K. S.; LIN, D.; BANJEVIC, D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. **Mechanical Systems and Signal Processing**, 2006. v. 20, n. 7, p. 1483–1510.

JIANG, J.-R. *et al.* Semi-Supervised Time Series Anomaly Detection Based on Statistics and Deep Learning. **Applied Sciences 2021, Vol. 11, Page 6698**, 21 jul. 2021. v. 11, n. 15, p. 6698. Disponível em: <https://www.mdpi.com/2076-3417/11/15/6698/htm>. Acesso em: 13 set. 2022.

KARUPPIAH, K.; SANKARANARAYANAN, B.; ALI, S. M. On sustainable predictive maintenance: Exploration of key barriers using an integrated approach. **Sustainable Production and Consumption**, 1 jul. 2021. v. 27, p. 1537–1553. . Acesso em: 11 dez. 2021.

KETKAR, N. Stochastic gradient descent. **Deep learning with Python**. [S.l.]: Springer, 2017, p. 113–132.

KINGMA, D. P.; BA, J. Adam: A Method for Stochastic Optimization. 22 dez. 2014. Disponível em: <https://arxiv.org/abs/1412.6980>. Acesso em: 11 dez. 2021.

_____; WELLING, M. Auto-encoding variational bayes. [S.l.]: [s.n.], 2014.

_____; _____. An Introduction to Variational Autoencoders. **Foundations and Trends in Machine Learning**, 6 jun. 2019. v. 12, n. 4, p. 307–392. Disponível em: <http://arxiv.org/abs/1906.02691>. Acesso em: 10 fev. 2022.

KIRANYAZ, S. *et al.* 1D convolutional neural networks and applications: A survey. **Mechanical Systems and Signal Processing**, 1 abr. 2021. v. 151, p. 107398. . Acesso em: 10 fev. 2022.

KRISSAANE, I. *et al.* Anomaly Detection Semi-Supervised Framework for Sepsis Treatment. **Computing in Cardiology**. [S.l.]: IEEE, 2019, V. 2019-Septe, p. 1.

LEE, J. *et al.* Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. **Mechanical Systems and Signal Processing**, 1 jan. 2014. v. 42, n. 1–2, p. 314–334. . Acesso em: 12 jan. 2022.

LESSMEIER, C. *et al.* Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification. **PHM Society European Conference**, 2016. v. 3, n. 1. Disponível em: <https://www.papers.phmsociety.org/index.php/phme/article/view/1577>. Acesso em: 27 mar. 2022.

LI, H. *et al.* Improving rail network velocity: A machine learning approach to predictive maintenance. **Transportation Research Part C: Emerging Technologies**, 2014. v. 45, p. 17–26.

LIANG, H. *et al.* Robust unsupervised anomaly detection via multi-time scale DCGANs with forgetting mechanism for industrial multivariate time series. **Neurocomputing**, 29 jan. 2021. v. 423, p. 444–462. . Acesso em: 12 jan. 2022.

LIANSHENG, L.; DATONG, L.; YU, P. Anomalous sensing data recovery with mutual information and Relevance Vector Machine. [S.l.]: IEEE, 2017. V. 2018-Janua, p. 247–252.

LIN, H. C.; YE, Y. C. Reviews of bearing vibration measurement using fast Fourier transform and enhanced fast Fourier transform algorithms: **https://doi.org/10.1177/1687814018816751**, 16 jan. 2019. v. 11, n. 1, p. 1–12. Disponível em: <https://journals.sagepub.com/doi/full/10.1177/1687814018816751>. Acesso em: 5 jun. 2022.

LIU, Ruonan *et al.* **Artificial intelligence for fault diagnosis of rotating machinery: A review**. **Mechanical Systems and Signal Processing**.

MAKHZANI, A. *et al.* Adversarial Autoencoders. **arXiv e-prints**, 2015. p. arXiv:1511.05644.

MALHOTRA, P. *et al.* LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection. 1 jul. 2016. Disponível em: <http://arxiv.org/abs/1607.00148>. Acesso em: 21 dez. 2021.

MARTIN, K. F. A review by discussion of condition monitoring and fault diagnosis in machine tools. **International Journal of Machine Tools and Manufacture**, 1 maio. 1994. v. 34, n. 4, p. 527–551. . Acesso em: 27 jun. 2021.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics 1943 5:4**, dez. 1943. v. 5, n. 4, p. 115–133. Disponível em: <https://link.springer.com/article/10.1007/BF02478259>. Acesso em: 19 out. 2021.

MING, N. K.; PHILIP, N.; SAHLAN, S. Proactive and predictive maintenance strategies and application for instrumentation & control in oil & gas industry. **International Journal of Integrated Engineering**, 2019. v. 11, n. 4.

MISRA, K. B. Maintenance Engineering and Maintainability: An Introduction. **Handbook of Performability Engineering**. [S.l.]: Springer London, 2008, p. 755–772.

MITCHELL, T. **Machine Learning. New York: McGraw Hill**. [S.l.]: [s.n.], 1997.

MONASTERIO, V.; BURGESS, F.; CLIFFORD, G. D. Robust classification of neonatal apnoea-related desaturations. **Physiological Measurement**, 2012. v. 33, n. 9, p. 1503–1516.

NGUYEN, D. *et al.* Joint Network Coding and Machine Learning for Error-prone Wireless Broadcast. 28 dez. 2016. Disponível em: <http://arxiv.org/abs/1612.08914>. Acesso em: 7 fev. 2022.

OMRI, N. *et al.* Towards an adapted PHM approach: Data quality requirements methodology for fault detection applications. **Computers in Industry**, 2021a. v. 127, p. 103414.

_____ *et al.* Towards an adapted PHM approach: Data quality requirements methodology for fault detection applications. **Computers in Industry**, 1 maio. 2021b. v. 127, p. 103414. . Acesso em: 12 jan. 2022.

ORRÙ, P. F. *et al.* Machine learning approach using MLP and SVM algorithms for the fault prediction of a centrifugal pump in the oil and gas industry. **Sustainability (Switzerland)**, 2020. v. 12, n. 11.

PACHECO, F.; CERRADA, M.; SANCHEZ, R. V. Framework for discovering unknown abnormal condition patterns in gearboxes using a semi-supervised approach. [S.l.]: IEEE, 2017. V. 2017-Decem, p. 63–68.

PECHT, M. G.; KANG, M. Introduction to PHM. **Prognostics and Health Management of Electronics**. [S.l.]: IEEE, 2018, p. 1–37.

POLESE, F. *et al.* Predictive Maintenance as a Driver for Corporate Sustainability: Evidence from a Public-Private Co-Financed R&D Project. **Sustainability 2021, Vol. 13, Page 5884**, 24 maio. 2021. v. 13, n. 11, p. 5884. Disponível em: <https://www.mdpi.com/2071-1050/13/11/5884/htm>. Acesso em: 11 dez. 2021.

QUATRINI, E. *et al.* Condition-based maintenance-An extensive literature review. **Machines**, 2020. v. 8, n. 2, p. 31.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning internal representations by error propagation. Parallel Distributed Processing: Exploration in the Microstructure of Cognition, vol. 1. **Foundations**, 1986. p. 318–362.

SCHLEGL, T. *et al.* Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. Cham: Springer, 2017. V. 10265 LNCS, p. 146–147.

SHIN, W.; BU, S.-J.; CHO, S.-B. 3D-Convolutional Neural Network with Generative Adversarial Network and Autoencoder for Robust Anomaly Detection in Video Surveillance. **International Journal of Neural Systems**, 2020. v. 10, n. 6, p. 2050034. Disponível em: <www.worldscientific.com>. Acesso em: 20 dez. 2021.

SHUMWAY, R. H.; STOFFER, D. S.; STOFFER, D. S. **Time series analysis and its applications**. [S.l.]: Springer, 2000. V. 3.

SOUALHI, A.; MEDJAHER, K.; ZERHOUNI, Noureddine. Bearing Health Monitoring Based on Hilbert – Huang Transform , Support Vector Machine , and Regression. 2015. v. 64, n. 1, p. 52–62.

SU, Y. *et al.* Robust anomaly detection for multivariate time series through stochastic recurrent neural network. [S.l.]: [s.n.], 2019. p. 2828–2837.

TABRIZI, A. *et al.* Early damage detection of roller bearings using wavelet packet decomposition, ensemble empirical mode decomposition and support vector machine. **Meccanica**, 2015. v. 50, n. 3, p. 865–874.

TAN, Y. *et al.* A one-class SVM based approach for condition-based maintenance of a naval propulsion plant with limited labeled data. **Ocean Engineering**, 2019. v. 193, p. 106592.

TAN, Y. L.; SEHGAL, V.; SHAHRI, H. H. **Sensoclean: Handling noisy and incomplete data in sensor networks using modeling**. **Main**.

THEISSLER, A. *et al.* Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry. **Reliability Engineering & System Safety**, 1 nov. 2021. v. 215, p. 107864. . Acesso em: 11 dez. 2021.

TSANG, A. H. C. *et al.* Data management for CBM optimization. **Journal of Quality in Maintenance Engineering**, 2006. v. 12, n. 1, p. 37–51.

VATHOOPAN, M. *et al.* Modular Fault Ascription and Corrective Maintenance Using a Digital Twin. **IFAC-PapersOnLine**, 2018. v. 51, n. 11, p. 1041–1046.

VERSTRAETE, D.; DROGUETT, E.; MODARRES, M. A deep adversarial approach based on multisensor fusion for remaining useful life prognostics. **Proceedings of the 29th European Safety and Reliability Conference, ESREL 2019**, 2020. v. v. 20, n., p. 1072–1077.

VU, H. *et al.* Robust Anomaly Detection in Videos Using Multilevel Representations. **Proceedings of the AAAI Conference on Artificial Intelligence**, 17 jul. 2019. v. 33, n. 01, p. 5216–5223. Disponível em: <https://ojs.aaai.org/index.php/AAAI/article/view/4456>. Acesso em: 12 jan. 2022.

WU, W. *et al.* Convergence of gradient method with momentum for back-propagation neural networks. **Journal of computational mathematics**, 2008. p. 613–623.

XIONG, Y.; ZUO, R. Robust Feature Extraction for Geochemical Anomaly Recognition Using a Stacked Convolutional Denoising Autoencoder. **Mathematical Geosciences 2021**, 25 fev. 2021. p. 1–22. Disponível em: <https://link.springer.com/article/10.1007/s11004-021-09935-z>. Acesso em: 12 jan. 2022.

YUCESAN, Y. A.; DOURADO, A.; VIANA, F. A. C. A survey of modeling for prognosis and health management of industrial equipment. **Advanced Engineering Informatics**, 1 out. 2021. v. 50, p. 101404. . Acesso em: 12 jan. 2022.

ZAIDAN, M. A. *et al.* Bayesian Hierarchical Models for aerospace gas turbine engine prognostics. **Expert Systems with Applications**, 2015. v. 42, n. 1.

ZEMOURI, R. **Semi-supervised Adversarial Variational Autoencoder**.

ZHANG, L. *et al.* A Review on Deep Learning Applications in Prognostics and Health Management. **IEEE Access**, 2019. v. 7, p. 162415–162438.

ZHAO, H. *et al.* Anomaly detection and fault analysis of wind turbine components based on deep learning network. **Renewable Energy**, 2018. v. 127, p. 825–834.

ZHAO, Z. *et al.* Deep learning algorithms for rotating machinery intelligent diagnosis: An open source benchmark study. **ISA Transactions**, 1 dez. 2020. v. 107, p. 224–255. . Acesso em: 21 dez. 2021.