Decision Threshold Selection and Low-Pass Filter Application for Anomaly Detection with Sparse Autoencoder

Maira Farias Andrade Lira[1], **Elias Amancio Siqueira-Filho**[2], **Ricardo Bastos Cavalcante Prudêncio**[1]

[1]Informatics Center – Federal University of Pernambuco (UFPE)
Recife – PE – Brazil

[2]Advanced Institute of Technology and Innovation (IATI)
Recife – PE – Brazil.

{mfal,rbcp}@cin.ufpe.br, elias.amancio@iati.org.br

Abstract. *Anomaly detection techniques in mechanical equipment are useful to anticipate failures and avoid stops or unplanned corrective maintenance. New detection techniques, based on deep learning methods and the availability of data monitored in real-time explain this success. This work investigates the use of Sparse Autoencoders (SAE) for online anomaly detection, applied to predictive maintenance in a public database referent to the Porto city metro. We evaluated different approaches to define anomaly score threshold levels and found these strongly affect detection metrics. We also evaluated low-pass filters usage to reduce false alarms and their impact on true anomaly detection. The use of low-pass filters after a preliminary classification of anomalies reduced the false positive rate up to 5.5 times compared to the SAE without the filter but also reduced the true positive detection. We also verified that the filter use was essential to detect anomaly sequences.*

Resumo. *Técnicas de detecção de anomalias em equipamentos mecânicos podem antecipar falhas e evitar paradas ou manutenções corretivas não programadas. Novas técnicas de detecção, baseadas em aprendizado profundo e na disponibilidade de dados de monitoramento de sinais em tempo real fundamentam esse sucesso. Neste trabalho, investigamos o uso de Sparse Autoencoders (SAE) para detecção de anomalias online para manutenção preditiva em uma base de dados pública referente ao metrô da cidade do Porto. Avaliamos abordagens para cálculo dos limiares de decisão sobre o score de anomalia retornado pelo SAE e verificamos que a abordagem influencia fortemente as métricas de detecção. Também investigamos o uso de filtros passa-baixa para reduzir falsos alarmes e o impacto destes na detecção de anomalias reais. A aplicação do filtro passa-baixa após uma classificação preliminar de anomalias reduziu em até 5,5 vezes a taxa de falsos positivos quando comparada ao SAE sem filtro, porém também levou a redução da detecção de anomalias reais. Verificamos ainda que a utilização do filtro foi essencial para detectar sequências de anomalias.*

1. Introdução

Anomalias são padrões que diferem significativamente do comportamento geral observado em um conjunto de dados. Métodos de detecção de anomalias (AD - do inglês

anomaly detection) têm sido usado em contextos diversos de aplicação. O nosso trabalho tem como foco a AD em equipamentos mecânicos, que pode ser bastante útil no contexto de manutenção preditiva (PdM - do inglês *predictive maintenance*) para evitar perdas financeiras, auxiliando a identificar falhas nos equipamentos de forma antecipada. De fato, a aplicação desses métodos para PdM ganhou destaque com o monitoramento contínuo de sinais em equipamentos e uso de sensores diversos para captura de dados, que podem ser então classificados como normais ou anômalos.

Existem diversas técnicas na literatura para AD, incluindo técnicas baseadas em aprendizado profundo, com bastante destaque nos últimos anos para a AD em equipamentos com modelos de Sparse Autoencoders (SAE) [Sun et al. 2016, Wen et al. 2019, Davari et al. 2021]. No contexto de AD, um *score* de anomalia é obtido considerando o erro de reconstrução do SAE para cada padrão de entrada. Se o erro de reconstrução ultrapassar um limiar de decisão, o padrão de entrada é classificado como anômalo.

Nesse trabalho, focamos em dois aspectos práticos do uso de modelos de AD, também compartilhados pelos SAEs. O primeiro aspecto diz respeito à escolha dos limiares de decisão para classificação de padrões de forma não-supervisionada. Valores muito baixos para os limiares podem resultar em aumento na taxa de falso postivo (FP - do inglês *False Positive*), enquanto valores muito altos podem diminuir a taxa de FP mas também reduzir a taxa de verdadeiros positivos (TP - do inglês *True Positive*). Essa decisão é crucial para a detecção de anomalias e não é comumente explorada com profundidade na literatura. O segundo aspecto é a redução dos FP que são padrões que apresentam valores de *score* de anomalia altos mas que na realidade são padrões não-anômalos. Para isso, foram aplicados filtros passa-baixa como sugerido por [Ribeiro et al. 2016].

Nós implementamos uma estrutura de AD *online* baseada na arquitetura SAE de forma não-supervisionada. Os dois aspectos apontados acima foram investigados a partir de experimentos com diferentes estratégias para definição dos limiares de decisão e também duas abordagens para aplicação do filtro passa-baixa. Os experimentos foram realizados com dados de uma aplicação de PdM no metrô da cidade do Porto, Portugal, disponibilizado em [Davari et al. 2021]. Foi observado nos experimentos que a forma de calcular o limiar de decisão impactou diretamente nas métricas de desempenho preditivo e a aplicação de filtros se mostrou promissora na redução de falsos positivos.

Este artigo é organizado como se segue. Na seção 2 são apresentados alguns conceitos fundamentais ligados ao trabalho. A seção 4 detalha a metodologia aplicada bem como os cenários de definição de limiares de decisão e aplicação de filtros avaliados neste trabalho. Já a seção 5 discute os resultados obtidos nos experimentos e a seção 6 apresenta as principais conclusões obtidas pelo trabalho.

2. Conceitos Fundamentais

Nessa seção, apresentamos os conceitos fundamentais de Autoencoders (seção 2.1) e filtros passa-baixa (seçao 2.2).

2.1. Autoencoder

Os autoencoders (AEs) são um dos modelos de aprendizagem profunda que ganharam muito destaque para aplicação em PdM. Os AEs são redes neurais não-supervisionadas treinadas para reconstruir os valores de entradas na camada de saída [Li et al. 2022]. AE

tem como objetivo aprender a função aproximada da identidade para a saída $\hat{x}_j$ ser o mais próximo da entrada x_j por meio da minimização do erro de reconstrução. Quando parte dos dados de entrada são correlacionados, o AE vai descobrir parte destas correlações [Ng et al. 2011].

Os AEs são redes neurais simétricas constituídos de um *encoder*, uma camada latente e um *decoder*. O *encoder* tem como função comprimir a entrada do AE, a camada latente possui o *encoded data*, e o *decoder* reconstrói a informação codificada. Para adaptar este modelo para a detecção de anomalias, o erro de reconstrução da entrada é usado para definir um *score* de anomalia. Para classificar as instâncias como normais ou anômalas é utilizado um valor limiar (*threshold*). Se o erro de reconstrução for abaixo do limiar, as instâncias são consideradas normais. Caso contrário, são consideradas anômalas.

Há quatro variações principais de AEs que são: Undercomplete Autoencoder, Variational Autoencoder, Denoising Autoencoder e Sparse Autoencoder (SAE). SAE são usados tipicamente para aprender características ligadas com classificação e por isso são muito usados para detectar anomalias em conjuntos com alta dimensionalidade [Goodfellow et al. 2016, Tun et al. 2020, Davari et al. 2021].

2.1.1. Sparse Autoencoder

SAE é uma derivação de AE em que um termo de penalização de esparsidade é adicionado ao AE original na camada latente. Com isso, o SAE possuí um número de neurônios na primeira camada escondida maior e apenas uma parcela de neurônios é ativada e treinada simultaneamente durante o codificação e decodificação [Wen et al. 2019]. A Figura 1 apresenta a arquitetura geral de um SAE.

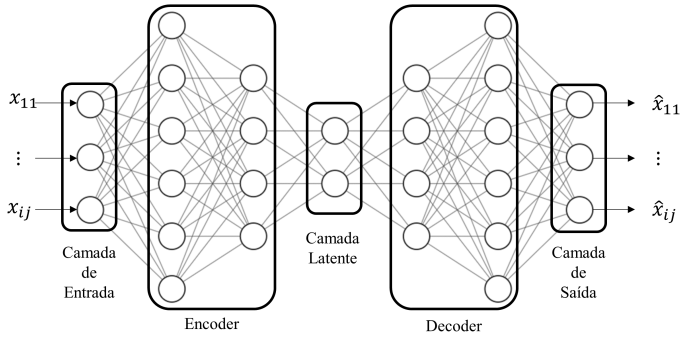

Figura 1. Arquitetura do Sparse Autoencoder.

A Equação 1 apresenta a função de custo proposta por [Ng et al. 2011]. A função de custo pode ser dividida em três partes: (1) o erro médio quadrático de reconstrução (MSE - do inglês *Mean Squarred Error*); (2) a regularização dos pesos L2; e (3) a regularização de esparsidade por meio da divergência de Kullback-Leibler (KL).

$$J_{SAE} = \frac{1}{m}\sum_{j=1}^{m}|\hat{x}_{ij} - x_{ij}|^2 + \lambda||W||_2^2 + \beta\sum_{h=1}^{s2}KL(\rho||\hat{\rho}_h) \quad (1)$$

onde no termo de MSE: m é o número de características, x_{ij} é a j-ésima característica do

ciclo i, $\hat{x}_{ij}$ é a reconstrução da entrada x_{ij}. No termo de regularização dos pesos: λ é o decaimento de peso usado para previnir o sobreajuste e $||W||$ são os pesos dos neurônios do SAE. No terceiro termo, de regularização de esparsidade, β é o peso de esparsidade, h é o índice das unidades escondidas e $s2$ é o número de neurônios na camada escondida, ρ e $\hat{\rho}_h$ são, respectivamente, o alvo de esparsidade e o parâmetro de ativação médio do neurônio h.

2.2. Filtro Passa-Baixa

Um grande desafio para AD não-supervisionado é a alta taxa de falsos positivos (FP). Ou seja, é comum que os algoritmos indiquem como anômalas instâncias normais. De acordo com [Ribeiro et al. 2016], um filtro passa-baixa (LPF - do inglês *low-pass filter*) suaviza mudanças abruptas de um sinal a ser filtrado, o que reduz sua amplitude e a taxa de FP. No caso de AD, o sinal a ser filtrado seria a sequência de *scores* de anomalia ordenada no tempo. O LPF, uma vez aplicado, poderia por exemplo atenuar picos no *score* de anomalia, dentro de uma sequência de dados bem comportados, o que possivelmente seriam casos de FP.

Há diversos tipos de LPF, sendo os mais simples baseados em médias móveis, tais como média móvel simples, média móvel exponencial e média móvel dos mínimos quadrados. A média móvel exponencial pode ser representada por $z_k = z_{k-1} + \alpha(y_k - z_{k-1})$, onde z, y são, respectivamente a saída e entrada do filtro, k representa o índice temporal e α é o parâmetro de suavização.

3. Trabalhos Relacionados

AD é crucial para o prognóstico de falhas de componentes críticos de um sistema e garantir a confiabilidade de operação de equipamentos. Em aplicações de PdM, a AD precisa alertar a falha em um estado inicial para os especialistas e operadores terem tempo hábil para realizar a manutenção antes da falha total [Fernandes et al. 2022, Gama et al. 2022]. Três abordagens de AD se destacam na literatura: detecção supervisionada, não-supervisionada e semi-supervisionada, com as abordagens não-supervisionadas ganhando destaque nos últimos anos pelo menor custo e serem adequadas para cenários com pouca ou nenhuma disponibilidade de dados rotulados [Chandola et al. 2009].

Com o avanço de metodologias não-supervisionadas ou semi-supervisionadas que utilizam aprendizado de máquina e aprendizado de máquina profundo (DL - do inglês *deep learning*), a performance de métodos de detecção de anomalias e predição de falhas melhorou drasticamente [Kim et al. 2021]. Os modelos de DL podem extrair automaticamente características dos dados originais com alta dimensionalidade e identificar com acurácia a saúde de equipamentos e ganharam muito destaque nos últimos anos, em especial os Sparses Autoencoders (SAE) [Sun et al. 2016, Wen et al. 2019, Davari et al. 2021].

No contexto de AD, os SAE detectam uma anomalia a partir de um *score* de anomalia baseado no erro de reconstrução para cada padrão de entrada. Se o erro de reconstrução ultrapassar um limiar de decisão (*threshold*), o padrão de entrada é classificado como anômalo. Por isso, a definição do valor do limiar de decisão é crítica, uma vez que valores baixos aumentam a incidência de FP e valores altos podem reduzir o TP.

[Borghesi et al. 2019] aplicaram o conceito de n-percentil do erro de reconstrução para o cálculo de *threshold* em um Autoencoder semi-supervisionado testando os percentis 95, 97 e 99 e não encontrou uma relação positiva ao simplesmente aumentar o limiar de decisão devido a redução de TP. [Tun et al. 2020] também adotaram o conceito de n-percentil em um SAE, com uma variação de n entre 1 e 100, mas determinaram o melhor *threshold* pelo conceito de *recall*, o que torna a abordagem semi-supervisionada. Apesar de ambos trabalhos explorarem o conceito de percentil, não foram exploradas abordagens com diferentes estratégias para a definição do limiar em modelo completamente não supervisionado. Neste trabalho, apesar de testarmos apenas testarmos um percentil, comparamos com outras estratégias de cálculo de *threshold* de forma completamente não-supervisionada.

[Li et al. 2022] desenvolveram uma abordagem de AD não-supervisionada baseada em autoencoder e redes neurais convolucionais e utilizaram o conceito de erro máximo de reconstrução para definir o limiar de decisão do detector de forma adaptativa. Apesar do trabalho comparar o modelo proposto com diferentes detectores da literatura, não testou diferentes abordagens de cálculo de *threshold*. Em nosso trabalho, nós comparamos o conceito de erro máximo com duas abordagens distintas de cálculo de *threshold* não-supervisionado e avaliamos a influência da estratégia adotada em métricas comumente usadas em AD.

[Perini et al. 2023] estudaram 21 abordagens para determinar um fator de contaminação de forma não-supervisionada para definição posterior do limiar de decisão a ser utilizado pelos detectores com 22 bases de dados e 10 detectores de anomalia com princípios distintos, com a abordagem de *threshold* baseada em quartis atingiu o quarto menor erro médio absoluto e a sétima maior deterioração de F1-Score. Em nosso trabalho foi utilizado como base uma abordagem fundamentada no conceito de quartis apresentada neste artigo. Nós tivemos como objetivo avaliar a influência da seleção do *threshold* em FP, TP, *recall*, precisão e F1-Score.

É importante ressaltar que a abordagem não-supervisionada em geral pode gerar uma maior taxa de falsos positivos e tende a ser menos robusta do que abordagens supervisionadas [Umer et al. 2022]. [Ribeiro et al. 2016] propuseram a aplicação de filtros passa-baixa (LPF, do inglês - *Low Pass-Filter*) para reduzir a taxa de FP pela atenuação da amplitude de picos gerados na detecção. [Davari et al. 2021], por exemplo, avaliaram a otimização de FP com a aplicação de LPF baseado em média móvel exponencial após a classificação de anomalias por um SAE, mas não exploraram outras técnicas de determinação de *threshold* além da abordagem de quartis, nem foram avaliadas outras abordagens de utilização do LPF ou o efeito do filtro no TP. Por isso, nesse trabalho, além de testarmos outras estratégias de *threshold*, nós também adotamos duas abordagens distintas de aplicação de filtro passa-baixa, sendo uma aplicação semelhante a proposta por [Davari et al. 2021] e outra mais simples e totalmente não-supervisionada. Além disso, discutimos o efeito das aplicações de LPF no FP e no TP, e realizamos uma maximização de F1-Score para obter um melhor equilíbrio de métrica invés de apenas uma minimização de FP.

Baseado na relevância da determinação do *threshold* e na redução de falsos alarmes em aplicações reais, este trabalho buscou investigar com maior profundidade dois objetivos principais, sendo eles: (1) investigar a influência da seleção dos limiares de de-

cisão para a classificação de padrões de forma não-supervisionada e utilizando diferentes princípios; e (2) observar o impacto da aplicação de um filtro passa-baixa na detecção de anomalias utilizando duas abordagens distintas.

4. Trabalho Desenvolvido

O presente trabalho avaliou a influência da seleção do limiar de decisão para modelos de AE para AD não-supervisionada. A Figura 2 ilustra a metodologia adotada. Inicialmente, um modelo de AE é aprendido de forma não-supervisionada, utilizando uma janela de dados sem anomalias. Diferentes abordagens para definição do limiar de decisão foram investigadas também de forma não-supervisionada e sem intervenção. A partir da inicialização, um procedimento de aprendizado *online* com janelas deslizantes foi adotado, em que tanto o modelo AE como o limiar de decisão são adaptados a cada iteração. Nesse procedimento, uma janela de treinamento é usada para treinar o modelo AE e para definir os limiares de decisão. Uma janela de dados de validação, por sua vez, é utilizada para definir um critério de parada do treinamento do AE e para remover dados potencialmente anômalos para o treinamento do AE na iteração seguinte do aprendizado *online*. Nos experimentos realizados, uma janela separada de teste é utilizada para avaliação do modelo de AD adaptado a cada iteração. A adaptação e teste do modelo de AD é repetido até que nenhuma janela de tempo esteja disponível.

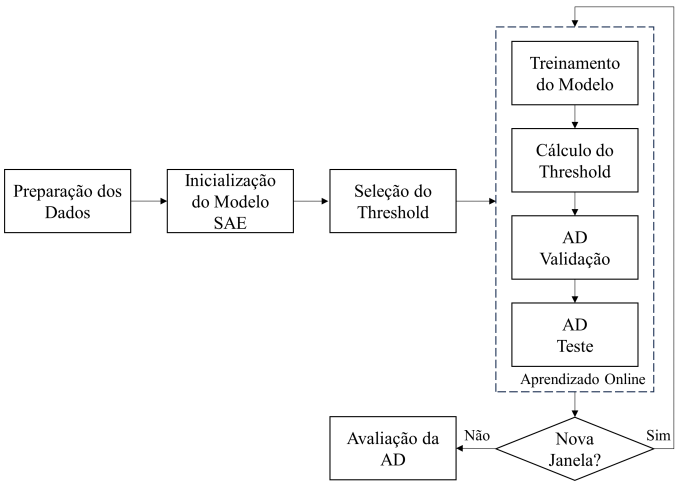

Figura 2. Diagrama da metodologia adotada.

Outro fator que também foi avaliado neste trabalho foi a aplicação LPF que almejam reduzir a taxa de FP detectados [Ribeiro et al. 2016]. Para este fim, foi avaliada a utilização do filtro em duas condições: (1) aplicação do filtro para o erro de reconstrução e posterior detecção de anomalias de forma totalmente não-supervisionada; (2) detecção inicial a partir do erro de reconstrução não filtrado e aplicação do filtro diretamente sobre as classificações geradas pelo detector que foi utilizado por [Davari et al. 2021]. Esses experimentos foram realizados para investigar a influência do filtro em FP e TP, e comparar as duas abordagens.

4.1. Preparação dos Dados

Foi selecionado um conjunto de dados público para detecção de falhas [Veloso et al. 2022], disponibilizado no repositório UCI por [Davari et al. 2021]. A

Tabela 1. Parâmetros da arquitetura SAE definidos por [Davari et al. 2021].

Parâmetros	SAE Analógico	SAE Digital
#Neurônios na Camada de Entrada	16	9
#Neurônios na 1ª Camada Escondida	128	36
#Neurônios na 2ª Camada Escondida	64	18
#Neurônios na 3ª Camada Escondida	32	-
#Neurônios na Camada Latente	12	6
β	5	6
λ	1e-5	2e-5
ρ	0,01	0,05

base de dados apresenta sinais analógicos e digitais relacionados a unidade de produção de ar (APU - do inglês *Air Production Unit*) de um trem do metrô da cidade do Porto, Portugal. Os dados foram coletados entre fevereiro e agosto de 2020 e disponibilizados de forma tabular com uma frequência de amostragem de 0,1 Hz. Para realizar a detecção de anomalias no APU, foram utilizadas as seguintes variáveis: (1) analógicas TP3 e Motor_current; (2) digitais COMP, DV Electric, TOWERS, MPG, LPS, Pressure_switch e Caudal_impulses.

Embora o conjunto de dados originais não tenha valores numéricos indefinidos, não é uma série temporal contínua. Por isso, definimos que um ciclo só é considerado válido se não houve ausência de informação por mais de um minuto. A repetição de dados analógicos por um longo período também não foi considerada confiável, devido a possíveis problemas de sensor, e, portanto, foi descartada.

Uma etapa de identificação do ciclo foi realizada conforme proposto por [Ribeiro et al. 2016] e [Davari et al. 2021] para reduzir a dimensionalidade dos dados. Cada ciclo foi então segmentado em sinais digitais e analógicos, resultando em conjuntos de atributos analógicos com dezesseis, e digitais com nove entradas, seguido por padronização pelo z-score.

4.2. Inicialização do modelo SAE

Foram desenvolvidos 2 modelos de aprendizagem de máquina profunda, um para os atributos analógicos e outro para os atributos digitais, utilizando a biblioteca *keras* em Python. Neste trabalho, foram adotadas arquiteturas semelhantes as utilizadas por [Davari et al. 2021] conforme apresentado na Tabela 1.

Além desses parâmetros, ambas as redes utilizaram o otimizador Adam com uma taxa de aprendizado de 0,0001, um tamanho de lote de 32 e um total de 1500 épocas. Nós optamos por utilizar a parada antecipada com paciência de 200 épocas para evitar o sobreajuste dos SAEs.

4.3. Seleção do Threshold

Foram realizadas três abordagens distintas de definição de *threshold*. A primeira abordagem estudada foi mencionada por [Davari et al. 2021] e é baseada na análise de *boxplot*. Inicialmente é realizado o treinamento do SAE utilizando apenas uma janela de ciclos sem anomalias com duração de três semanas. A partir do treinamento, são computados os erros de reconstrução de cada ciclo da janela de treinamento utilizando MSE.

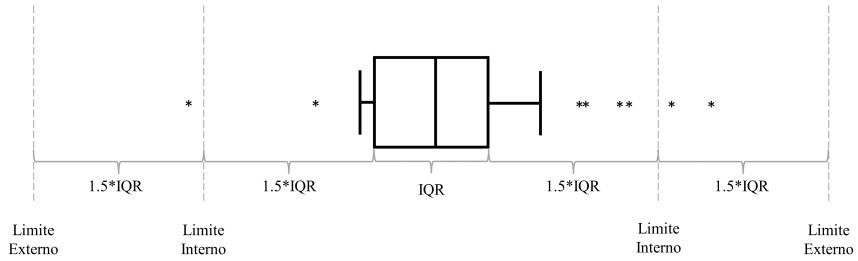

Figura 3. Parâmetros do *boxplot*.

Sendo assim, se o erro de reconstrução do conjunto de validação ou teste de uma janela w for superior ao limite superior do erro de reconstrução da janela de treinamento w, apresentado na Figura 3 como limite externo, o ciclo é considerado uma anomalia.

Então, a cada janela de treinamento w, o threshold é computado de acordo com a equação 2. Nessa equação, $Q1_w$ e $Q3_w$ são, respectivamente, o primeiro e terceiro quartil do erro de reconstrução do treinamento da janela w, RE^w. Nós apelidamos essa abordagem de **Box** para facilitar a compreensão e esta foi adotada como base para o desenvolvimento deste trabalho.

$$thres_w = Q3_w + 3 * (Q3_w - Q1_w) \tag{2}$$

A segunda abordagem foi baseada em [Li et al. 2022], em que o *threshold* é computado a cada ciclo do aprendizado online de um AE e considera como limite de anomalia o maior erro de reconstrução encontrado na janela de treinamento RE^w. Essa abordagem de *threshold* tende a gerar uma taxa de FP inferior a primeira abordagem, mas pode reduzir a taxa de TP. A abordagem **Max** foi selecionada com intuito de observar a relação entre FP e TP com limiares extremos naturalmente mais elevados.

A última abordagem avaliada foi discutida em [Borghesi et al. 2019] e em [Tun et al. 2020] e seleciona de forma semi-supervisionada *thresholds* distintos que equivalem a percentis específicos de 1 a 100 do conjunto de erro de reconstrução, tendo sido selecionado o percentil 95. Ou seja, com essa abordagem, o *threshold* calculado durante a janela de treinamento w vai definir que um erro de reconstrução superior ao valor que representa o percentil 95 é uma anomalia. Esta abordagem foi apelidada de **Perc** e foi considerada devido a sua baixa complexidade matemática.

4.4. Aprendizado *Online*

Desenvolvemos uma arquitetura de aprendizagem *online*, para aproximar de uma aplicação de manutenção preditiva convencional e permitir períodos de aprendizagem mais curtos. Foi adotada a técnica de janelas deslizantes com conjunto de aprendizado (três semanas de treinamento e uma semana de validação) e teste (uma semana).

Na etapa de aprendizado o SAE, com arquitetura pré-definida, foi treinado de forma não-supervisionada por meio da minimização do erro de reconstrução. A etapa de aprendizado utilizou dois conjuntos de dados: um conjunto de dados sem anomalias de treinamento com duração de três semanas e um conjunto de validação com uma semana de informações. O método de treinamento utilizou validação cruzada para evitar o

enviesamento da rede com a técnica de parada antecipada e a detecção de anomalias no conjunto de validação após a definição do *threshold*. O treinamento foi repetido até todo período analisado ser coberto.

A partir do erro da reconstrução obtido no conjunto de treinamento sem presença de anomalias, foi possível calcular o *threshold* cada uma das abordagens apresentadas na seção 4.3 em todas as janelas de forma adaptativa. Essa abordagem foi adotada já que o comportamento de cada uma das janelas pode variar ao longo do tempo.

Após o cálculo do *threshold* de cada janela w, foi realizada a detecção de anomalia no conjunto de validação e de teste por duas abordagens distintas. As abordagens apresentadas na Figura 4 indicam como foi aplicado o LPF nos experimentos.

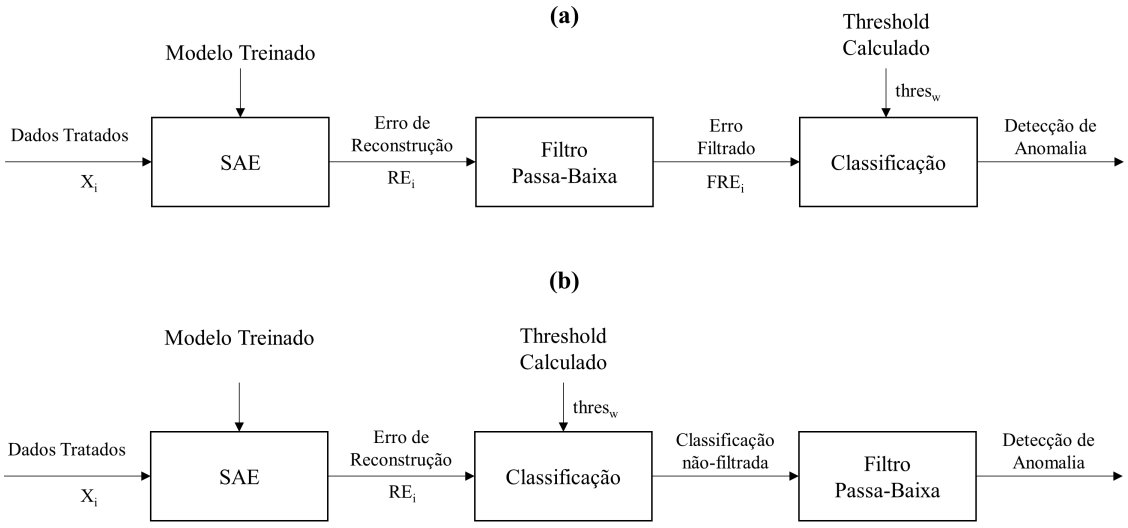

Figura 4. (a) Abordagem Pré: Filtro sob o erro de reconstrução, (b) Abordagem Pós: Filtro sob a classificação preliminar.

A abordagem **Pré** (Figura 4.(a)) foi utilizada para suavizar o erro de reconstrução RE_i para reduzir o número de FP e possibilitar a detecção de sequências de falhas de forma totalmente não-supervisionada, já que uma das características do conjunto de dados usados é a presença de ciclos subsequentes com anomalias. Para isso, o sinal de erro de reconstrução RE_i é aplicado no LPF apresentado na seção 2.2 e a classificação de cada ciclo avaliado é realizada. Se o sinal filtrado FRE_i for menor do que o *threshold*, o ciclo é classificado como normal (0). Caso contrário, anômalo (1).

A abordagem **Pós** Figura 4.(b)) foi baseada no trabalho de [Davari et al. 2021] e o filtro passa-baixa apresentado na seção 2.2 é aplicado apenas após a classificação inicial. Nesse caso, o filtro passa-baixa é aplicado sob a sequência binária gerada na classificação inicial baseada apenas no erro de reconstrução RE_i. Para realizar a detecção final, foi utilizado um segundo *threshold* constante pré-definido experimentalmente a partir da maximização do F1-Score ao longo de todas as janelas de teste. Então, valores de classificação filtrada acima do novo *threshold* são etiquetadas como anomalia e abaixo são considerados ciclos normais.

As anomalias detectadas no conjunto de validação de cada janela deslizante foram removidas do conjunto de treinamento da janela seguinte, o que possibilitou um trata-

mento não-supervisionado de dados julgados como anômalos. A ausência de anomalias no treinamento é essencial para otimizar a detecção de anomalias por AEs e calcular corretamente as diferentes abordagens de *threshold*.

Após a remoção de anomalias verificadas no conjunto de validação, foi realizada a detecção de anomalias no conjunto de teste. Após o aprendizado e teste do AD na janela, todas as janelas foram deslizadas em uma semana e o processo foi repetido até que não houvesse nenhuma entrada a ser testada.

4.5. Avaliação da Detecção de Anomalias

As anomalias que foram detectadas em cada janela de teste foram avaliadas e comparadas com as anomalias conhecidas e rotuladas em [Davari et al. 2021]. Para isto, foram computados algumas métricas tradicionais de estudos de detecção de anomalia e classificação, como *Recall*, Precisão e F1-Score, além da taxa de verdadeiros positivos (TPR) e falsos positivos (FPR). Nessa avaliação foi considerado como positivo uma anomalia e negativo um ciclo normal.

Além destas métricas, também foi avaliada a capacidade dos modelos de aprendizagem profunda de predizer com uma certa antecedência a presença da anomalia. Por fim, foi avaliada a influência do filtro em ambas abordagens apresentadas nesta seção e da seleção do *threshold* utilizado no conjunto de dados analógicos e digitais tratados.

5. Experimentos e Resultados

Nos experimentos deste trabalho, utilizamos uma base de dados pública para detecção de falhas disponibilizado por [Davari et al. 2021]. Para os experimentos foram considerados os dados entre os meses de março e julho de 2020. Para calcular as métricas de desempenho preditivos, foi utilizada as informações dos dados rotulados como anomalia disponibilizadas pelos autores da base. Toda os resultados foram apresentados considerando os ciclos como instâncias individuais e não sequências de ciclos como os autores da base utilizaram.

Para avaliar o efeito da aplicação do LPF na detecção de anomalias utilizando o SAE, foram utilizadas duas abordagens que nomeamos de **Pré**, quando o LPF foi aplicado diretamente sob o erro de reconstrução de forma não-supervisionada, e **Pós**, quando o LPF foi aplicado sobre a classificação gerada após a avaliação do erro de reconstrução de forma semi-supervisionada. Também foram testadas três abordagens de cálculo de *threshold* adaptativos **Box**, **Max** e **Perc** mencionadas na seção 4.3 para avaliar a influência da seleção do *threshold* em abordagem não-supervisionada de AD.

Para cada combinação de método de escolha de limiar e aplicação de filtro, foram testados dez valores de α distintos entre $[0,01; 0,1]$, além dos dois conjuntos de dados (analógicos e digitais). Assim para cada combinação de métodos, foram avaliados 20 modelos. A partir dessa avaliação, foram selcionados os melhores modelos experimentados para cada combinação de seleção de *threshold* e abordagem de filtro pela maximização do F1-Score. A Tabela 2 apresenta as métricas de avaliação utilizadas para o estudo da influência da seleção do *threshold* e da abordagem de aplicação de filtro. Para ressaltar que a partir da variação do α não identificamos nenhuma relação direta entre o parâmetro de suavização com a redução de FP. Os modelos Box Melhor Pré, Box Melhor Pós, Max

Tabela 2. Métricas analisadas para os melhores modelos experimentados de acordo com a maximização de F1-Score.

Métrica	Box			Max			Perc		
	Sem Filtro	Melhor Pré	Melhor Pós	Sem Filtro	Melhor Pré	Melhor Pós	Sem Filtro	Melhor Pré	Melhor Pós
TPR (%)	33,33	70,14	15,97	13,89	15,28	7,64	26,86	52,78	24,31
FPR (%)	10,39	26,65	1,87	1,49	3,74	0,77	8,50	21,25	4,32
Recall (%)	33,33	70,14	15,97	13,88	15,28	7,64	29,86	52,78	24,31
Precisão (%)	8,03	6,68	18,85	20,20	10,78	21,15	8,72	6,33	13,26
F1-Score (%)	12,94	12,20	17,29	16,46	12,09	11,22	13,50	11,30	17,16

Melhor Pré, Max Melhor Pós, Perc Melhor Pré e Perc Melhor Pós usaram um α de respectivamente: 0,04; 0,05; 0,08; 0,1; 0,1; 0,02. Além disso, dentre os melhores modelos selecionados apenas Max Sem Filtro utilizou dados analógicos.

5.1. Influência da Seleção de *Threshold*

Nos resultados, verificamos que a seleção do *threshold* modificou diretamente as métricas de avaliação para todas as abordagens de filtro analisadas. Dentre os métodos de seleção avaliados, o Box apresentou as melhores taxas de detecção real. De fato, o TPR referente a aplicação do modelo Box sem filtro foi 2,4 vezes superior ao modelo Max sem filtro. Isso se deve ao fato do *threshold* computado por Max ser muito influenciado por picos de erro durante o treinamento, o que pode levar a desprezar anomalias de *score* menos extremo. A abordagem Perc apresentou resultados próximos ao da abordagem Box ao se avaliar o TPR.

O FPR também foi fortemente influenciado pela seleção do *threshold*. Como esperado, a abordagem Max gerou o menor FPR, sendo cerca de 85,65% inferior quando comparado a abordagem Max sem filtro com a abordagem Box sem filtro. Ao avaliar os modelos sem filtro, verificou uma oscilação no F1-Score de até 27,2% entre a abordagem Max e Box. Porém a diferença entre a abordagem Box e Perc sem filtro foi de apenas 4,33%. É interessante futuramente avaliar outros valores para a abordagem Perc além de 95 para otimizar o cálculo do *threshold*.

Vale salientar que a priorização das taxas TPR ou FPR dependem fortemente do contexto de aplicação. É priorizada a minimização de FPR quando os falsos alarmes possuem alto custo e a melhor técnica para este caso foi Max. Já se o custo de falso negativo for elevado, é priorizada uma maximização de TPR, que nos experimentos foi obtido por Box. Então, essa avaliação ressalta que a técnica de determinação do *threshold* de forma não-supervisionada possui alta influência para a detecção de anomalias com uso de SAE.

5.2. Avaliação da Abordagem de Filtro Passa-Baixa

Após avaliarmos a influência da seleção do *threshold*, verificamos os resultados obtidos pelos melhores modelos por cada combinação de *threshold* e aplicação de LPF Pré e Pós. Para ilustrar a aplicação dos filtros, Figura 5 apresenta a evolução dos *scores* de anomalia para um período de teste, considerando: (a) a abordagem sem filtro; (b) a abordagem Pré, com os respectivos sinais de entrada e saída do filtro; e (c) a abordagem Pós, com os respectivos sinais de entrada e saída do filtro.

Ao compararmos o FPR dentre os melhores modelos analisados, observamos que a abordagem Pós de filtro reduziu em até 5,5 vezes do FPR quando comparado a aplicação do modelo sem filtro com o threshold Box. Porém a abordagem Pré aumentou em até 2,57

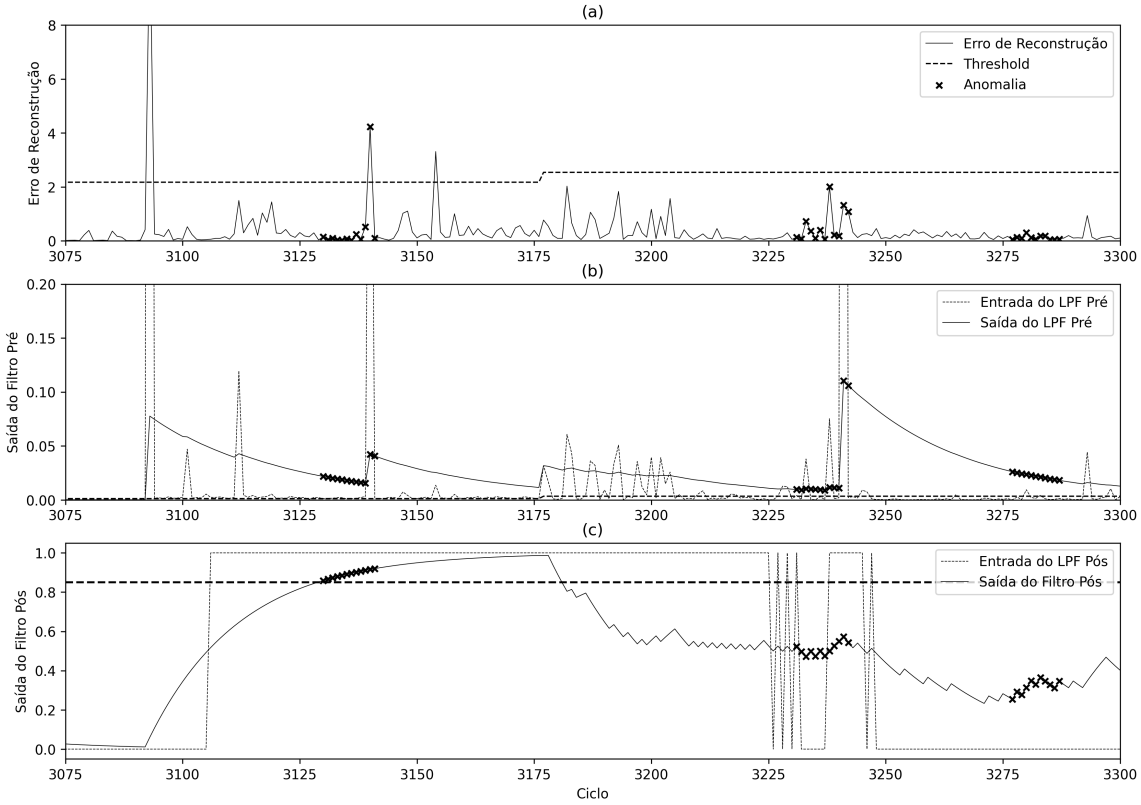

Figura 5. (a) Melhor modelo sem filtro, (b) Melhor modelo abordagem de filtro Pré, (c) Melhor modelo abordagem de filtro Pós.

vezes o número de ciclos normais detectados como anomalia. Esse fenômeno pode ser observado na Figura 5(b) quando é comparado a entrada do filtro (erro de reconstrução) e sua respectiva saída (erro filtrado). O comportamento do erro de reconstrução de forma isolada apresenta picos elevados, ilustrado tanto na Figura 5 (a) e (b), e a aplicação do filtro com os valores testados de α suavizou o sinal de uma forma muito lenta, o que levou a uma elevada taxa de FP na abordagem Pré. Já na abordagem Pós, a aplicação do filtro sob a classificação ajudou a desprezar algumas anomalias pontuais geradas por eventuais picos do erro de reconstrução.

Já em relação ao TPR, a abordagem Pré foi capaz de detectar 2,1 vezes mais anomalias corretamente quando comparada a aplicação sem filtro. A abordagem Pós reduziu o TPR em até 2,08 vezes quando comparada a abordagem sem filtro. O menor nível de TP da abordagem Pós está diretamente relacionado ao atraso que o modelo gera com a aplicação do filtro. Esse atraso pode ser visualizado na Figura 5(c). Apesar disso, é interessante salientar que a aplicação sem filtro não foi capaz de detectar sequência de ciclos anômalos nem antecipar uma falha, enquanto ambas as abordagens com filtro foram.

Portanto, foi avaliado que utilizar o LPF imediatamente após a reconstrução (abordagem Pré) não auxiliou na redução de FPR já que pontos de muito elevados de erro foram propagados por muitos ciclos. Porém esta abordagem melhorou significativamente a taxa de *recall* obtida que costuma ser crítica na detecção de falhas. A abordagem Pós da aplicação do filtro conseguiu reduzir significativamente o número de FPR assim como

proposto por [Ribeiro et al. 2016], porém levou a uma degradação do TPR e, consequentemente, do *recall*. Considerando o F1-Score, houve uma melhoria quando se utilizou a abordagem Box e Perc juntamente com a abordagem Pós de filtro quando comparado ao modelo sem filtro. Além disso, para essa segunda abordagem, foram selecionados um segundo *threshold* de forma supervisionada. Essa seleção pode ser prejudicial para aplicações sem consulta de especialistas ou acesso aos relatórios de falha.

6. Conclusão

Esse artigo propôs investigar a influência da seleção da abordagem de cálculo de *threshold* adaptativo em um modelo *online* com arquitetura de Sparse Autoencoder. Além disso, também foi avaliada a aplicação de filtro passa-baixa de duas formas: (Pré) sob o erro de reconstrução da janela de teste, ou seja, imediatamente antes da classificação das instâncias; e (Pós) após a classificação inicial das instâncias. Para ter uma referência da abordagem sem aplicação de filtro, também foram consideradas 6 aplicações com filtro utilizando dados analógicos e digitais da base de dados e as mesmas abordagens de definição de *threshold* adaptativo.

Não houve diferença significativa entre as abordagens Box e Perc de cálculo de *threshold* adaptativo nas métricas, porém a abordagem Max apresentou um comportamento distinto de TPR e FPR. Enquanto a aplicação do filtro na reconstrução aumentou TPR e FPR, a aplicação do filtro na classificação reduziu o FPR em até 5,5 vezes quando comparado a abordagem sem filtro. Apesar disso, ambas as aplicações de filtro permitiram antecipar a falha e detectar sequências de anomalias.

Por isso, nós sugerimos que seja estudado posteriormente a aplicação de diferentes tipos de filtros passa-baixa com as abordagens Pré e Pós em cenários com anomalias sequenciais e mais experimentos explorando a abordagem Perc para o cálculo do *threshold* adaptativo. Além disso, também devem ser estudadas formas de determinar o *threshold* da abordagem Pós de forma não-supervisionada para promover mais autonomia ao modelo, bem como buscar uma otimização do TPR.

Por fim, também é interessante explorar a aplicação de teoria de resposta ao item com uma base de modelos não-supervisionados e explorar técnicas de explicabilidade. Ambas as técnicas devem aumentar a credibilidade da detecção para o usuário e também auxiliar na detecção da causa-raiz de possíveis falhas para aplicações de PdM.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) por meio do Programa de Excelência Acadêmica (PROEX).

Referências

Borghesi, A., Bartolini, A., Lombardi, M., Milano, M., and Benini, L. (2019). Anomaly detection using autoencoders in high performance computing systems. In *The Thirty-First AAAI Conference on Innovative Applications of Artificial Intelligence (IAAI-19)*, volume 33, pages 9428–9433. AAAI.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3).

Davari, N., Veloso, B., Ribeiro, R. P., Pereira, P. M., and Gama, J. (2021). Predictive maintenance based on anomaly detection using deep learning for air production unit in the railway industry. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10. IEEE.

Fernandes, M., Corchado, J. M., and Marreiros, G. (2022). Machine learning techniques applied to mechanical fault diagnosis and fault prognosis in the context of real industrial manufacturing use-cases: a systematic literature review. *Applied Intelligence*, 52:14246–14280.

Gama, J., Ribeiro, R. P., and Veloso, B. (2022). Data-driven predictive maintenance. *IEEE Intelligent Systems*, 37(4):27–29.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Kim, D., Lee, S., and Kim, D. (2021). An applicable predictive maintenance framework for the absence of run-to-failure data. *Applied Sciences*, 11(11).

Li, Z., Sun, Y., Yang, L., Zhao, Z., and Chen, X. (2022). Unsupervised machine anomaly detection using autoencoder and temporal convolutional network. *IEEE Transactions on Instrumentation and Measurement*, 71.

Ng, A. et al. (2011). Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19.

Perini, L., Bürkner, P.-C., and Klami, A. (2023). Estimating the contamination factor's distribution in unsupervised anomaly detection. PMLR 202, pages 27668–27679, Honolulu, United States of America. Proceedings of Machine Learning Research.

Ribeiro, R. P., Pereira, P., and Gama, J. (2016). Sequential anomalies: a study in the railway industry. *Machine Learning*, 105:127–153.

Sun, W., Shao, S., Zhao, R., Yan, R., Zhang, X., and Chen, X. (2016). A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement*, 89:171–178.

Tun, M. T., Nyaung, D. E., and Phyu, M. P. (2020). Network anomaly detection using threshold-based sparse. In *IAIT2020: Proceedings of the 11th International Conference on Advances in Information Technology*, IAIT2020, New York, NY, USA. Association for Computing Machinery.

Umer, M. A., Junejo, K. N., Jilani, M. T., and Mathur, A. P. (2022). Machine learning for intrusion detection in industrial control systems: Applications, challenges, and recommendations. *International Journal of Critical Infrastructure Protection*, 38:100516.

Veloso, B., Ribeiro, R. P., Gama, J., and Pereira, P. M. (2022). The MetroPT dataset for predictive maintenance. *Scientific Data*, 9:764.

Wen, L., Gao, L., and Li, X. (2019). A new deep transfer learning based on sparse auto-encoder for fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(1):136–144.