

Previsão de falhas em sistemas de compressores do metrô de Portugal

João Victor Prado de Souza^{1*}; Henrique Raymundo Gioia²

¹ Instituto de Pesquisas Eldorado. Engenheiro de Controle e Automação. Rua Edgar, nº 548 – Vila Leonor; 02077-020 São Paulo, SP, Brasil

² ESALQ/USP. Mestre em Economia Aplicada (Departamento de Economia, Administração e Sociologia). Av. Pádua Dias, nº 11 – São Dimas; 13418-900 Piracicaba, SP, Brasil

*autor correspondente: jvictorpsouza@gmail.com

Previsão de falhas em sistema de compressores do metrô de Portugal

Resumo

A manutenção preditiva tem se mostrado essencial para a continuidade operacional de sistemas críticos em ambientes industriais, como os compressores das Unidades Auxiliares de Potência (APU) do metrô de Portugal. Este trabalho teve como objetivo desenvolver uma abordagem baseada em aprendizado de máquina para a previsão de falhas nesses sistemas, utilizando o banco de dados MetroPT-3, composto por séries temporais multivariadas obtidas por sensores digitais e analógicos. O estudo empregou tanto modelos supervisionados (como “Random Forest” e “Support Vector Machines”), quanto modelos não supervisionados de detecção de anomalias, especialmente “autoencoders” (SAE e VAE). As etapas metodológicas incluíram o pré-processamento dos dados, seleção de variáveis, normalização e redução de dimensionalidade. Os modelos foram treinados e avaliados por meio de métricas como acurácia, precisão, “recall”, “F1-score” e erros de reconstrução. O melhor desempenho foi observado no modelo “Sparse Autoencoder” com dados digitais, que apresentou 100% de desempenho em diferentes conjuntos de teste, além de baixo tempo de processamento. Conclui-se que abordagens baseadas em detecção de anomalias são altamente eficazes para identificar falhas com antecedência, proporcionando maior confiabilidade ao sistema de manutenção.

Palavras-chave: manutenção preditiva; detecção de anomalias; aprendizado de máquina; Random Forest; Autoencoder.

Failure prediction in compressor system of the Portugal metro

Abstract

Predictive maintenance has proven to be essential for the operational continuity of critical systems in industrial environments, such as the compressors of the Auxiliary Power Units (APU) in the Portugal metro system. This study aimed to develop a machine learning-based approach for failure prediction in these systems, using the MetroPT-3 dataset, which consists of multivariate time series collected from digital and analog sensors. The study employed both supervised models (such as Random Forest and Support Vector Machines) and unsupervised anomaly detection models, especially autoencoders (SAE and VAE). The methodological steps included data preprocessing, variable selection, normalization, and dimensionality reduction. The models were trained and evaluated using metrics such as accuracy, precision, recall, F1-score, and reconstruction error. The best performance was observed in the Sparse Autoencoder model using digital sensor data, which achieved 100% precision and recall across different test sets, in addition to low processing time. It is concluded that anomaly detection-based approaches are highly effective in identifying failures in advance, providing greater reliability to the maintenance system.

Keywords: predictive maintenance; anomaly detection; machine learning; Random Forest; Autoencoder.

Introdução

A manutenção preditiva tem se tornado cada vez mais relevante em ambientes industriais, principalmente em setores onde a interrupção não planejada de sistemas críticos pode gerar altos custos e comprometer a segurança operacional. Um dos principais desafios enfrentados por empresas de transporte público, como as operadoras de metrô, é a necessidade de manter o desempenho eficiente de sistemas complexos, como os compressores das Unidades Auxiliares de Potência [APU]. Esses compressores são responsáveis por fornecer energia para funções auxiliares dos trens e, portanto, desempenham um papel crucial na operação geral (Jardine et al., 2006). Neste contexto, a aplicação de técnicas avançadas de detecção de anomalias e previsão de falhas em sistemas de compressão tem sido objeto de crescente interesse.

Este trabalho utilizou o banco de dados MetroPT-3, composto por dados temporais multivariados obtidos de sensores digitais e analógicos instalados em um sistema de compressão APU do metrô de Portugal. A análise de dados provenientes de diferentes tipos de sensores oferece uma visão abrangente do comportamento operacional, registrando variáveis como pressão, temperatura, corrente elétrica e estado de funcionamento de componentes críticos (Bengio, 2009). A diversidade de sinais coletados torna a análise complexa, exigindo a utilização de modelos robustos que sejam capazes de lidar com variáveis multivariadas e a detecção de padrões anômalos que possam indicar falhas iminentes.

A literatura destaca a importância da previsão de falhas como uma ferramenta para otimizar a manutenção em sistemas industriais. De acordo com Wang et al. (2008), a manutenção preditiva pode reduzir significativamente o tempo de inatividade dos sistemas e os custos operacionais, evitando paradas inesperadas e prolongadas. A implementação de técnicas de aprendizado de máquina, como “Random Forest” [RF], Máquinas de Vetores de Suporte [SVM] e redes neurais, tem se mostrado promissora para a análise de séries temporais e a detecção de anomalias em sistemas multivariados (Breiman, 2001; Cortes & Vapnik, 1995).

A “Random Forest” foi amplamente utilizada neste trabalho devido à sua capacidade de manejar grandes volumes de dados e lidar com variáveis de diferentes naturezas. Este método, que se baseia em um conjunto de árvores de decisão, provou ser eficaz na classificação e previsão de dados complexos, como os que compõem o banco MetroPT-3 (Pal, 2005). Além disso, a robustez do RF contra o overfitting torna-a uma ferramenta adequada para lidar com a variabilidade dos dados temporais.

As Máquinas de Vetores de Suporte são técnicas de aprendizado supervisionado usadas para classificação e regressão, e têm sido aplicadas com sucesso na detecção de anomalias em séries temporais multivariadas. A SVM visa maximizar a margem entre as diferentes classes de dados, separando as observações normais das anômalas (Cortes & Vapnik, 1995). Este trabalho utilizou a SVM para identificar padrões atípicos no funcionamento do compressor, o que é crucial para prevenir falhas antes que elas causem interrupções no sistema.

Outra abordagem explorada foi o uso de Redes Neurais Artificiais [RNA], particularmente redes neurais profundas (Deep Neural Networks [DNN]), que são adequadas para lidar com dados não lineares e de alta dimensionalidade. A aplicação de DNN permitiu capturar relações complexas entre as variáveis, que modelos lineares não conseguem identificar. As redes neurais têm se mostrado promissoras na detecção de anomalias em séries temporais, especialmente quando aplicadas a dados multivariados, como aqueles presentes no banco MetroPT-3 (Goodfellow, Bengio, & Courville, 2016).

Além dessas técnicas de classificação, métodos de detecção de anomalias baseados em “autoencoders”, como o “Sparse Autoencoder” [SAE] e o “Variational Autoencoder” [VAE], também foram utilizados. O SAE é uma técnica de redução de dimensionalidade que extrai características latentes importantes dos dados, facilitando a detecção de anomalias (Ng, 2011). Já o VAE, uma variante probabilística do autoencoder, foi aplicado para modelar a distribuição dos dados normais e identificar desvios significativos, que indicam potenciais falhas (Kingma & Welling, 2014). Para tanto esses modelos retiram padrões dos dados e depois tenta reconstruir eles baseado nos aprendizados anteriores, então eles possuem uma fase de codificação (“encoder”) e outra de decodificação (“decoder”). E usando os dados de saída é observado se há um desvio dos valores esperados, indicando uma anomalia.

A importância desta pesquisa está em sua contribuição para a melhoria dos métodos de manutenção preditiva, utilizando técnicas de aprendizado de máquina e detecção de anomalias para aumentar a confiabilidade e segurança operacional dos sistemas de compressão do metrô. A capacidade de prever falhas com antecedência e identificar anomalias com precisão pode reduzir custos e aumentar a eficiência, proporcionando maior estabilidade nas operações. Este trabalho busca demonstrar a eficácia dessas abordagens em um cenário real de operação, com base no banco de dados MetroPT-3, e avaliar qual técnica apresenta melhor desempenho para a detecção precoce de falhas.

Metodologia

Este trabalho utilizou o banco de dados MetroPT-3 obtidos no artigo publicado por Veloso et al. (2022), que contém informações temporais multivariadas sobre um compressor do sistema de Unidade Auxiliar de Potência [APU] do metrô de Portugal. De acordo com Veloso et al. (2022) os dados foram coletados por meio de sensores digitais e analógicos instalados no sistema de compressão, registrando variáveis como pressão, temperatura e corrente elétrica. A metodologia adotada foi dividida em várias etapas: desde o pré-processamento dos dados até a modelagem e comparação de resultados obtidos pelos diferentes métodos de classificação e detecção de anomalias.

Pré-processamento dos dados para os modelos de classificação

A primeira etapa do processo consistiu em uma análise detalhada e preparação dos dados contidos no MetroPT-3. Este banco de dados multivariado, com variáveis coletadas ao longo do tempo por sensores de diferentes tipos, apresentou desafios típicos de séries temporais, como ruídos, valores ausentes e a presença de anomalias. Para garantir a qualidade dos dados utilizados nos modelos preditivos, o pré-processamento foi dividido nas seguintes atividades:

Limpeza e preenchimento de valores ausentes: Dados faltantes foram preenchidos utilizando métodos adequados à natureza dos sinais temporais, como interpolação linear e “forward fill” (Little & Rubin, 2019). Poucos dados estavam ausentes (menos de 1%) no banco de dados utilizado e sempre para apenas um dos sensores coletados por vez. Não houve caso em que dois ou mais sensores tiveram falta de leitura, de modo que apenas a técnica de “forward fill” foi suficiente e não apresentou um aumento de enviesamento dos dados.

A informação de falha do sistema não estava contida no banco de dados, ela foi fornecida em forma de tabela, no qual foi necessário a inserção desses dados manualmente em coluna representando as falhas de modo binário (presença ou ausência de falha).

Tabela 1: Falhas apontadas durante a aquisição de dados

Número	Início	Fim	Correção/ Manutenção
1	18/04/2020 00:00	18/04/2020 23:59	
2	29/05/2020 23:30	30/05/2020 06:00	30/05/2020 12:00
3	05/06/2020 10:00	07/06/2020 14:30	08/06/2020 16:00
4	15/07/2020 14:30	15/07/2020 19:00	16/07/2020 00:00

Fonte: Gama et al., 2022

Normalização e padronização dos dados: Como as variáveis possuem diferentes escalas, a normalização foi aplicada para garantir que modelos de aprendizado de máquina e algoritmos de detecção de anomalias não fossem influenciados por amplitudes discrepantes entre as variáveis (Han et al., 2011). Para normalização dos dados foi usada a técnica “MinMaxScaler” da biblioteca scikit-learn, como intuito de não enviesar nenhum dos modelos que seriam utilizados, visto que alguns deles podem sofrer enviesamento com dados não normalizados (Amorim et al., 2022).

Redução de dimensionalidade: Para otimizar o desempenho dos modelos, a técnica de Análise de Componentes Principais [PCA] foi aplicada, mantendo as variáveis mais significativas que explicam a maior parte da variância nos dados (Jolliffe, 2002). Esta técnica foi fundamental para reduzir a complexidade dos dados sem perder informações relevantes para a análise.

Seleção dos Modelos

A seleção dos modelos foi dividida em duas categorias principais: modelos de classificação e modelos de detecção de anomalias, com o objetivo de comparar o desempenho e adequação de cada abordagem ao problema de manutenção preditiva.

Modelos de classificação

Os dados foram divididos em três subconjuntos: treino, validação e teste. O conjunto de treinamento foi utilizado para ajustar os parâmetros dos modelos, enquanto o conjunto de validação serviu para a otimização dos hiperparâmetros, utilizando técnicas como a busca em grade (“grid search”). O conjunto de teste foi utilizado para avaliar a precisão final dos modelos na detecção de falhas e anomalias.

Para o treino de todos os modelos de classificação, foram utilizados os dados referentes a primeira e da segunda falha como base. Para não haver o enviesamento por desbalanceamento de dados em condições normais, foi selecionado metade do período da antes e depois de cada falha, conforme a ilustração abaixo.

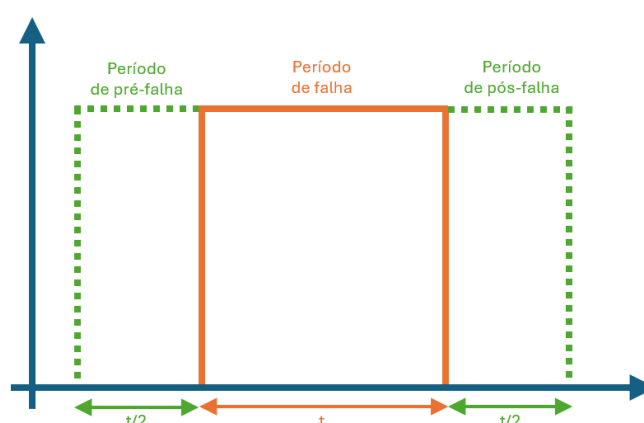


Figura 1: Dados selecionados para o treino dos modelos de classificação

Fonte: Resultados originais da pesquisa

Métricas de avaliação: As métricas de desempenho utilizadas incluem acurácia, precisão e “F1-score” para os modelos de classificação. Para os modelos de detecção de anomalias, métricas como o erro quadrático médio [MSE] e o erro absoluto médio [MAE] foram empregadas para avaliar a eficácia na reconstrução dos dados normais e na detecção de eventos anômalos (Han et al., 2011).

Foram utilizados modelos supervisionados para a classificação dos dados entre estados normais e anômalos. Os modelos empregados foram:

“Random Forest” [RF]: O “Random Forest” é um algoritmo de aprendizado de máquina supervisionado que utiliza múltiplas árvores de decisão para realizar a classificação dos dados. Este método foi escolhido por sua capacidade de lidar com grandes volumes de dados e sua eficácia em detectar padrões complexos em dados multivariados (Breiman, 2001). Além disso, a robustez contra “overfitting” é uma vantagem significativa, especialmente em cenários com alta dimensionalidade, como o conjunto de dados MetroPT-3. Para o Random Forest, os hiperparâmetros ajustados foram: “max_depth”, que indica a profundidade máxima das árvores; filtro temporal que somente indica um defeito, caso ele fique presente por pelo menos a quantidade de minutos ajustada (usado para diminuir a incidência de defeitos falso-positivos); e por fim as colunas utilizadas (Breiman, 2001).

Máquinas de Vetores de Suporte [SVM]: A SVM foi usada para a classificação dos dados com base nas variáveis temporais do compressor. Esse método é eficaz para encontrar a melhor margem de separação entre os estados normais e anômalos, permitindo a detecção de possíveis falhas antes que elas ocorram (Cortes & Vapnik, 1995). A SVM foi ajustada para desempenhar tanto a classificação quanto a regressão, dependendo da natureza do problema de falhas do sistema. Para o SVM, os hiperparâmetros ajustados foram: “kernel”, que indica a

natureza da relação entre os elementos (linear, polinomial, circular, ...), basicamente e indica o formato da separação entre os grupos de elementos; “gamma”, que indica a flexibilidade na separação dos grupos, ou seja, quão importante os pontos são considerando a distância deles em relação com a fronteira (parâmetro crucial para evitar um sobreajuste ou subajuste do modelo); filtro temporal que somente indica um defeito, caso ele fique presente por pelo menos a quantidade de minutos ajustada (usado para diminuir a incidência de defeitos falso-positivos); e por fim as colunas utilizadas (Cristianini & Shawe-Taylor, 2000).

Modelos de detecção de anomalias

O primeiro passo do pré-processamento dos dados para os modelos de detecção de anomalias foi criar a coluna com a informação de quais dados são anomalias e quais não são, de forma semelhante da qual foi feita para os modelos de classificação, a diferença aqui foi que os dados do dia anterior da falha também foram considerados anomalias, para que o modelo pudesse buscar padrões da falha antes mesmo delas acontecerem.

Os dados de treino utilizados foram todos referente a períodos não anômalos, pois nesse tipo de modelo, ele deve aprender a identificar quando os dados fogem do padrão esperado aprendido. Além disso foram criados períodos de aprendizado e teste, variando conforme o tempo passa separados de mês em mês, assim foram obtidos três conjuntos de dados, simulando um sistema que continua aprendendo em tempo real enquanto funciona (Box et al. 2015).

Os dados de treino foram utilizados os dados a partir de fevereiro e adicionados os dados dos meses subsequentes. Os dados de teste foram os meses subsequentes ao período usado para o treino até julho.

- 1º conjunto:
 - Treino: fevereiro até abril;
 - Teste: maio até julho;
- 2º conjunto:
 - Treino: fevereiro até maio;
 - Teste: junho até julho;
- 3º conjunto:
 - Treino: fevereiro até junho;
 - Teste: julho;

Por fim para os dados analógicos também foi aplicada uma técnica de normalização, nesse caso o método “Standard Scaler” da biblioteca scikit-learn. Esse método faz com que os dados possuam uma média igual a 0 e um desvio padrão igual a 1. Esse método foi aplicado pelo mesmo motivo dos modelos de classificação, para que os modelos não fossem enviesados devido uma tendência numérica dos diferentes parâmetros utilizados (Amorim et al., 2022).

Após a normalização os dados foram divididos em analógicos e digitais, para que fosse possível uma comparação entre eles de acordo com o resultado dos treinamentos.

A detecção de anomalias foi tratada utilizando métodos não supervisionados, voltados para a identificação de padrões fora da norma, que podem indicar potenciais falhas no sistema de compressão. Os modelos usados nesta seção foram:

“Sparse Autoencoder” [SAE]: Foi utilizado para realizar a redução de dimensionalidade e aprender uma representação latente compacta dos dados normais do sistema (Ng, 2011). Este modelo foi treinado para reconstruir os dados de entrada, sendo as anomalias identificadas pelos erros de reconstrução, que indicam comportamentos atípicos, no caso demonstram as falhas.

“Variational Autoencoder” [VAE]: O VAE, uma técnica baseada em aprendizado profundo e modelos probabilísticos, foi empregado para modelar a distribuição dos dados normais do compressor. Com o VAE, anomalias foram detectadas ao identificar dados que não se ajustavam à distribuição normal aprendida, indicando eventos de falha ou comportamento anormal (Kingma & Welling, 2014).

A estrutura dos modelos SAE e VAE são bem semelhantes, compostas basicamente por: entrada, codificador, espaço latente, decodificador e saída. As camadas do codificador eram compostas por 2 camadas convolucionais de uma dimensão com 32 e 16 filtros respectivamente. A camada de espaço latente, conhecida como “bottleneck” foi diferente para cada modelo de acordo com o tipo de modelo e tipo de dados. O decodificado é composto por 2 camadas convolucionais de uma dimensão transpostas com 16 e 32 filtros, respectivamente. E por fim uma camada de saída que foi diferente para cada modelo e conjunto de dados. Nota-se que o decodificador é o oposto simétrico ao codificador, pois o objetivo é simplificar os dados, e reconstruí-los em seguida. A função de ativação usadas nas camadas ocultas foi a função “relu”, com exceção da camada de “bottleneck” dos modelos VAE, em que foi usada a função linear.

No treino é feita uma comparação entre os dados de entrada e os dados de saída, espera-se que eles sejam próximos, indicando que o modelo foi capaz de reconstruir os dados com sucesso ou não.

Para a avaliação do modelo, foi aplicado o “mean square error” [MSE], depois um filtro passa-baixa para suavização (regulado pelo parâmetro “alpha”) e depois um último filtro de limítrofe (“threshold”) para indicar se é falha (1) ou ausência de falha (0).

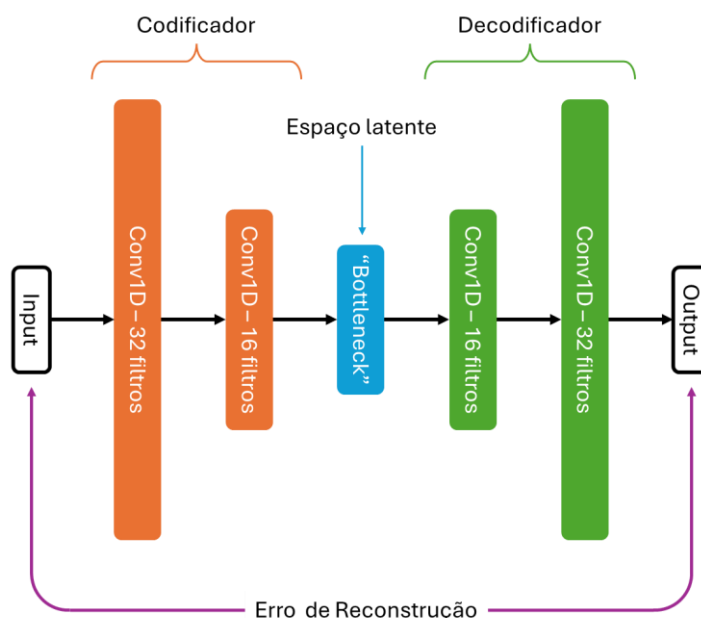


Figura 2: Estrutura de modelos com autoencoders

Fonte: Resultados originais da pesquisa

Então foram treinados 4 modelos (SAE com dados analógicos, VAE com dados digitais, SAE com dados digitais e VAE com dados analógicos) para o primeiro conjunto de dados com 50 épocas e nessa primeira análise foram usados alguns parâmetros que foram definidos inicialmente baseado na literatura e posteriormente ajustados manualmente para valores adequados, mas ainda não otimizados:

Tabela 2: Parâmetros dos filtros para cada modelo

Modelos	“alpha”	“threshold”
VAE com dados digitais	0.05	0.1
SAE com dados digitais	0.05	$7.5 * 10^{-65}$
VAE com dados analógicos	0.1	0.004
SAE com dados analógicos	0.1	0.00188

Fonte: Resultados originais da pesquisa

Depois da análise inicial e comparação dos quatro modelos, foi selecionado o melhor modelo entre eles (SAE com dados digitais), foi iniciada a etapa de otimização e para tanto, foram aplicadas as seguintes variações de valores para “alpha”: 0.03 e 0.07, 0.12, 0.82, 0.9 e 1.

Achando o melhor valor para “alpha” (igual a 0.82), o “threshold” foi variado para os seguintes valores: $15 * 10^{-65}$, $3.75 * 10^{-65}$ e $1.5 * 10^{-65}$.

Para a segunda análise de ajuste fino do “alpha” e “threshold”, foi usado o primeiro conjunto de dados para treinar e testar os modelos, possibilitando uma comparação direta variando somente um dos parâmetros por vez.

Com esses valores, o modelo treinado com o primeiro conjunto de dados já estava suficientemente adequado, não sendo necessárias mais otimizações, que demandariam mais tempo e não trariam resultados muito melhores. Então foi considerado que o modelo já estava adequadamente definido e treinado

Por fim o modelo com os melhores parâmetros foi treinado e testado com os outros dois conjuntos de dados para verificação do seu desempenho.

Comparação dos resultados dos modelos

Após o treinamento e a validação, os resultados obtidos por cada modelo foram comparados. A comparação se deu com base nas métricas de avaliação previamente mencionadas, levando em conta a capacidade de cada abordagem para prever falhas com antecedência e identificar anomalias que pudessem sugerir a necessidade de manutenção preditiva. Os modelos baseados em aprendizado de máquina foram comparados aos métodos de detecção de anomalias baseados em “autoencoders”, permitindo uma análise abrangente da eficácia de cada abordagem no contexto de dados multivariados.

Os códigos-fonte utilizados para a implementação e execução das etapas descritas estão disponíveis no repositório público do GitHub, cujo link se encontra no Apêndice A.

Resultados e Discussão

Pré-processamento dos dados para modelos de classificação

Da etapa inicial de pré-processamento, foi possível analisar a matriz de correlação entre as variáveis restantes e dessa forma ver quais variáveis tinham mais correlação com a variável de saída “y”, e ao mesmo tempo, baixa correlação entre si. Para tal foram selecionadas as seguintes variáveis: “DV_pressure”, “Oil_temperature”, “TP2”, “Motor_current”, “H1”, “MPG”, “DV_eletric”, “COMP”.

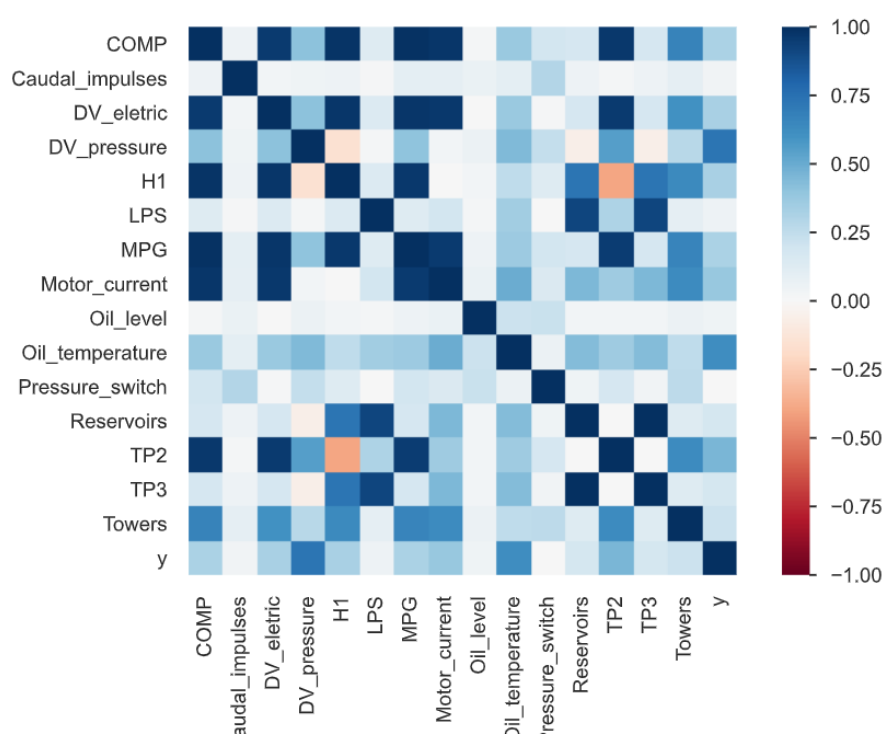


Figura 3: Tabela de correlação das variáveis do banco de dados.

Fonte: Resultados originais da pesquisa

Modelo de classificação

Random Forest: Para o primeiro treino foram utilizadas todas as colunas pré-selecionadas. Após o treinamento, foi inserido um filtro temporal que só indicaria falha, caso ela permanecesse por mais de 15 minutos, pois foi denotado que algumas leituras anômalas são esperadas por conta de se tratar de um sistema real, então para diminuir a sensibilidade do sistema de identificação foi definido esse filtro. O valor do filtro escolhido foi baseado no texto de estudo dos dados dos próprios autores da coleta dos dados e nas observações obtidas.

Para o primeiro modelo treinado, foi identificado que as 3 primeiras falhas foram corretamente identificadas, mas a terceira não pôde ser corretamente identificada. Mesmo usando o filtro, houve falso-positivos, indicando falhas, mesmo que elas não estavam presentes.

Para o segundo modelo treinado foram utilizadas menos variáveis, somente as com mais alta correlação com a falha (“DV_pressure”, “Oil_temperature”, “TP2”), e o filtro foi aumentado para 30 minutos. Desse modo as 4 falhas foram identificadas pelo modelo, mas houve uma maior incidência de falso-positivos.

Máquinas de Vetores de Suporte: Para esses modelos foram utilizadas somente as 3 variáveis com maior correlação, assim como no segundo modelo de “Random Forest” (“DV_pressure”, “Oil_temperature”, “TP2”) e o filtro foi mantido em 50 minutos.

O primeiro modelo treinado, teve um resultado bem similar ao primeiro modelo de Random Forest, as 3 primeiras falhas foram identificadas e foram indicados uma quantidade alta de falso-positivos.

Já no segundo modelo a janela temporal foi estendida e foi selecionado um período igual ao da falha antes de depois delas, nesse caso todas as falhas foram identificadas, mas houve uma incidência muito grande de falso-positivos.

Para análise de eficiência dos modelos, foram utilizadas 3 métricas: acurácia, precisão e F1. Observando os resultados, nota-se que o melhor resultado foi o primeiro modelo de “Random Forest” que apresentou resultados razoáveis. Os resultados falso-positivo na prática indicariam uma análise manual do sistema em busca de uma falha, idealmente busca-se diminuir a incidência disso, mas ainda assim não apresentaria um problema tão grande, pois o maior ponto seria identificar risco de falhas graves que poderiam afetar o sistema a ponto de pará-lo por um tempo grande.

Tabela 3: Resultados dos testes dos modelos

Modelo	Variáveis	Filtro (min)	Acurácia	Precisão	“F1-score”
Random Forest 1	8	15	99,28%	75,71%	83,68%
Random Forest 2	3	30	99,07%	68,31%	80,71%
SVM 1	3	50	98,69%	60,81%	73,85%
SVM2	3	50	98,30%	53,85%	68,76%

Fonte: Resultados originais da pesquisa

Analisando as métricas considerando as indicações falso-negativas mais relevantes, pois indicam uma falha não identificada, tempos que o segundo modelo de “Random Forest” foi o melhor. Ele foi capaz de identificar todas as falhas e apresentou boas métricas.

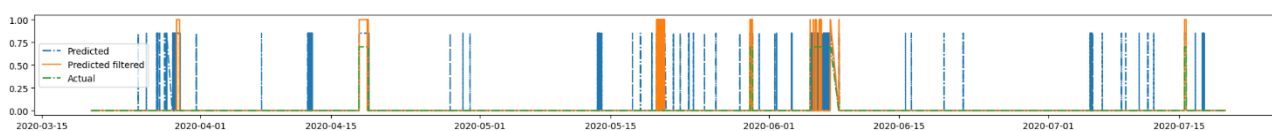


Figura 4: Resultado do modelo 2 de Random Forest

Fonte: Resultados originais da pesquisa

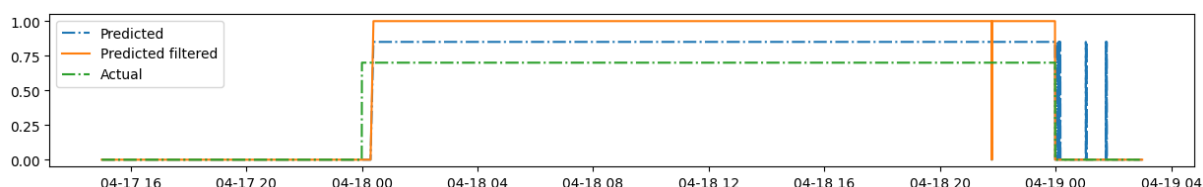


Figura 5: Resultado do modelo 2 de "Random Forest" para primeira falha

Fonte: Resultados originais da pesquisa

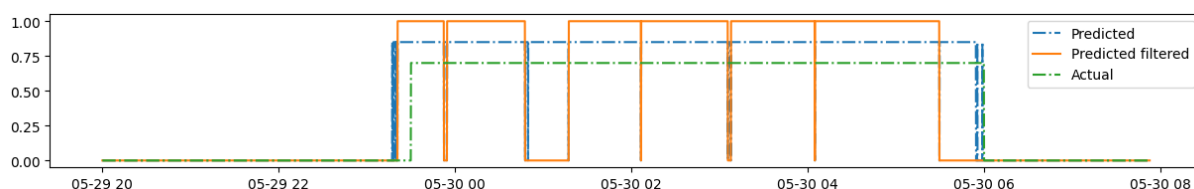


Figura 6: Resultado do modelo 2 de "Random Forest" para segunda falha

Fonte: Resultados originais da pesquisa

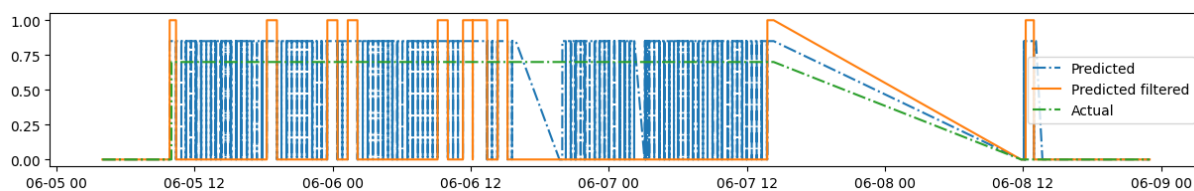


Figura 7: Resultado do modelo 2 de "Random Forest" para terceira falha

Fonte: Resultados originais da pesquisa

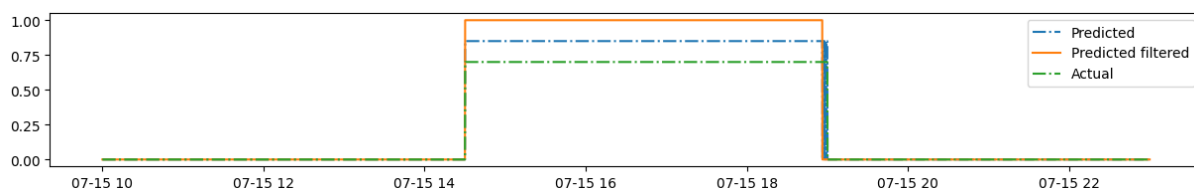


Figura 8: Resultado do modelo 2 de "Random Forest" para quarta falha

Fonte: Resultados originais da pesquisa

De modo geral as técnicas de classificação não são rápidas o suficiente para indicar as falhas com um período de antecedência grande para que a correção seja feita de forma ideal. São boas para indicar a falha, mas não tão boas para prevê-las, mas com certa manipulação é possível usá-las para esse fim (Nascimento et al., 2022). Para tanto, foram aplicadas outras técnicas mais sofisticadas para a previsão com maior tempo hábil antes da falha.

Modelos de detecção de anomalias

Usando o primeiro conjunto de dados, os 4 tipos de modelo foram treinados e avaliados, utilizando uma matriz de confusão que mostra as previsões corretas, os falso-positivos e os falso-negativos. Além da matriz foram usadas 3 métricas para dados normais e anômalos: precisão, “recall” e “F1-score”. Por fim, foi usada também a acurácia geral do modelo que foi o critério mais relevante para a comparação.

A seguir está apresentada a tabela com as métricas e a matriz de confusão do teste para cada modelo com o primeiro conjunto de dados.

Tabela 4: Métricas para cada modelo

Modelos	Precisão	“Recall”	“F1-Score”
VAE com dados digitais	88%	100%	94%
SAE com dados digitais	100%	100%	100%
VAE com dados analógicos	49%	37%	42%
SAE com dados analógicos	37%	44%	40%

Fonte: Resultados originais da pesquisa

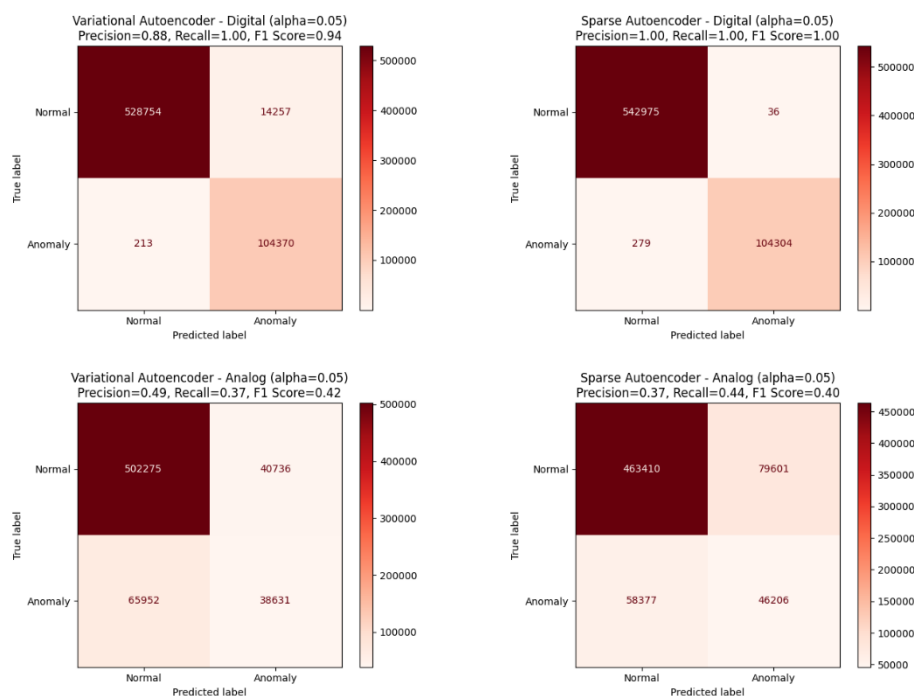


Figura 9: Matriz de confusão para cada modelo na ordem VAE com dados digitais, SAE com dados digitais, VAE com dados analógicos e SAE com dados analógicos

Fonte: Resultados originais da pesquisa

Como é fácil de observar em todos os conjuntos de teste o modelo que teve melhor desempenho foi o SAE usando dados dos sensores digitais como variáveis. Além disso ele foi o mais rápido no treino e teste, apresentando um tempo de 6 minutos 15 segundos, ou seja, 2,85 vezes menor em relação ao modelo mais demorado e computacionalmente custoso, o VAE com dados analógicos que levou 17 minutos e 49 segundos.

A seguir está apresentada as matrizes de confusão do teste para as variações de “alpha” com o primeiro conjunto de dados, na imagem as matrizes são referentes aos valores de “alpha” em ordem crescente (0.03, 0.05, 0.07, 0.12, 0.82, 0.9 e 1):

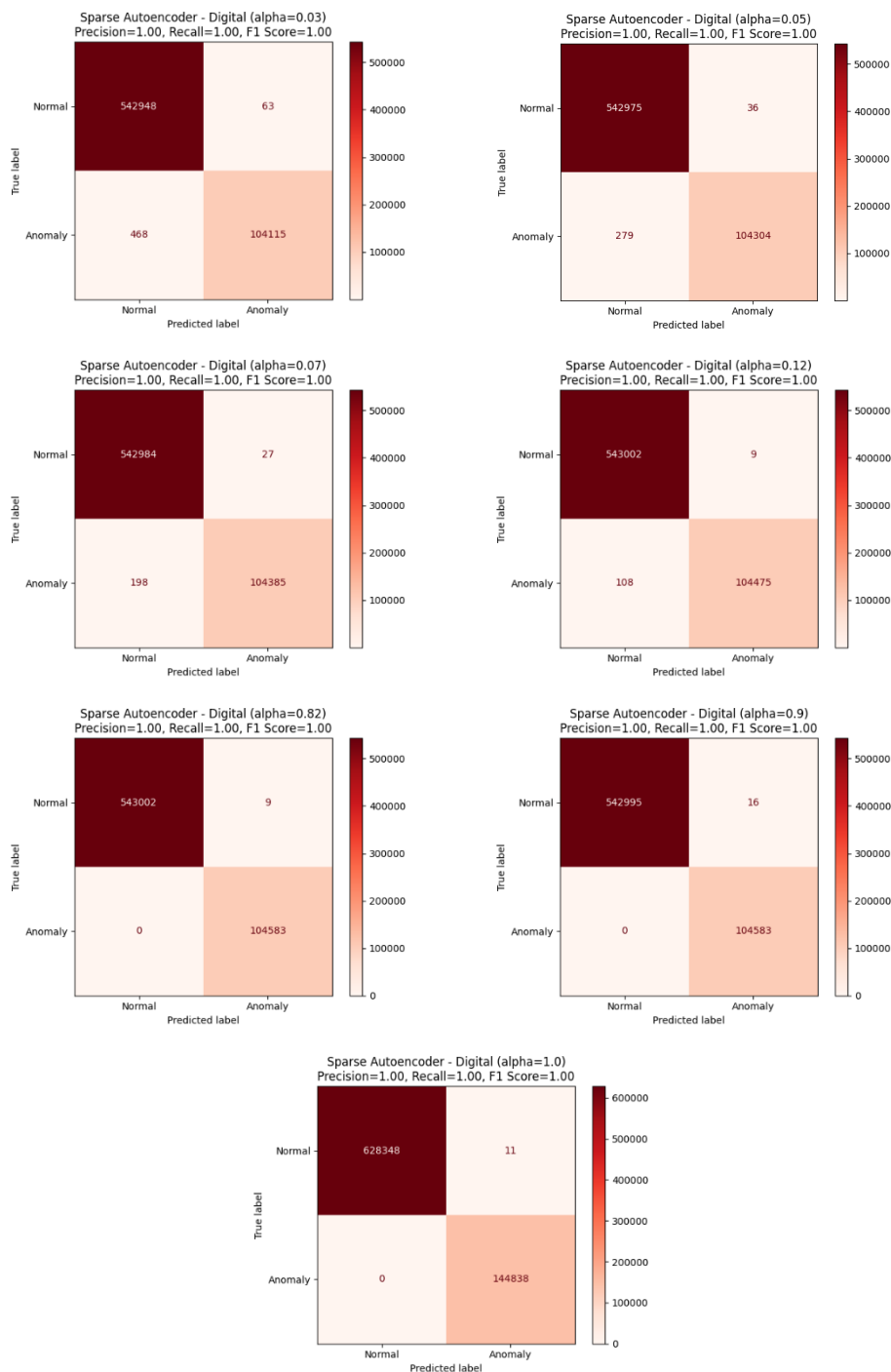


Figura 10: Matrizes de confusão para variações do "alpha" igual a 0.03, 0.05, 0.07, 0.12, 0.82, 0.9 e 1

Fonte: Resultados originais da pesquisa

Para a variação do α , todas as métricas (precisão, “recall” e “F1-score”) ficaram iguais a 100%. Nas matrizes de confusão, a variação não foi tão grande considerando a grande quantidade de pontos avaliados. Mesmo assim, pode-se considerar que o filtro regulado pelo parâmetro “alpha” apresentou um melhor desempenho quando “alpha” era igual a 0.82.

A seguir está apresentada a matriz de confusão do teste para as variações de “threshold” com o primeiro conjunto de dados, na imagem as matrizes são referentes aos valores de “threshold” em ordem crescente (1.5×10^{-65} , 3.75×10^{-65} , 7.5×10^{-65} e 15×10^{-65}):

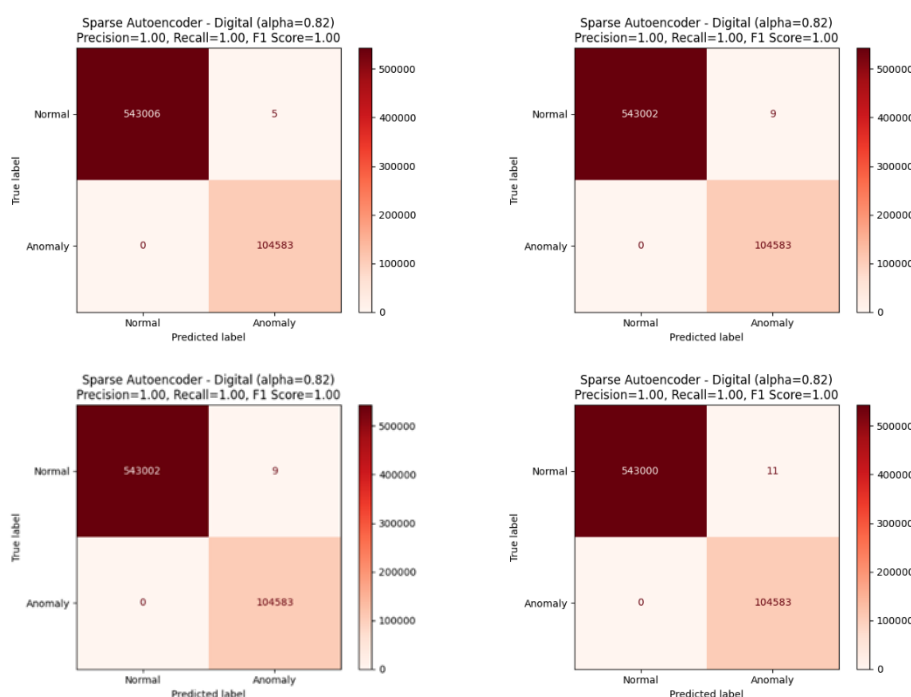


Figura 11: Matrizes de confusão para variações do “threshold” igual a 1.5×10^{-65} , 3.75×10^{-65} , 7.5×10^{-65} e 15×10^{-65}

Fonte: Resultados originais da pesquisa

E por fim para a variação do valor de “threshold”, todas as métricas (precisão, “recall” e “F1-score”) ficaram iguais a 100%. O “threshold” igual a 1.5×10^{-65} , apresentou o melhor resultado quando analisada a matriz de confusão, mas a variação foi tão pequena que é praticamente desprezível e pode ser ligada a erros inerentes de modelos de machine learning.

Essa etapa de ajuste fino poderia até mesmo ser desprezada, pois o modelo obtido na primeira fase já tinha um ótimo desempenho e nessa etapa a variação só é percebida analisando a matriz de confusão, porque analisando todas as métricas usadas (acurácia, “recall” e “F1-score”), em todas as variações de “alpha” e “threshold”, elas apresentaram valores iguais a 100%.

Assim depois de achar os parâmetros ótimos do melhor modelo, ele foi submetido a um treino contínuo, o mesmo que aconteceria em um cenário real, então o mesmo modelo foi treinado e testado com os outros conjuntos de dados mencionados na metodologia. Assim agrupando os resultados para os três conjuntos, é obtida uma tabela com as métricas e as matrizes de confusão para cada conjunto:

Tabela 5: Métricas para cada conjunto de dados

Conjunto de dados	Precisão	“Recall”	“F1-Score”
1º conjunto	100%	100%	100%
2º conjunto	100%	100%	100%
3º conjunto	98%	100%	99%

Fonte: Resultados originais da pesquisa

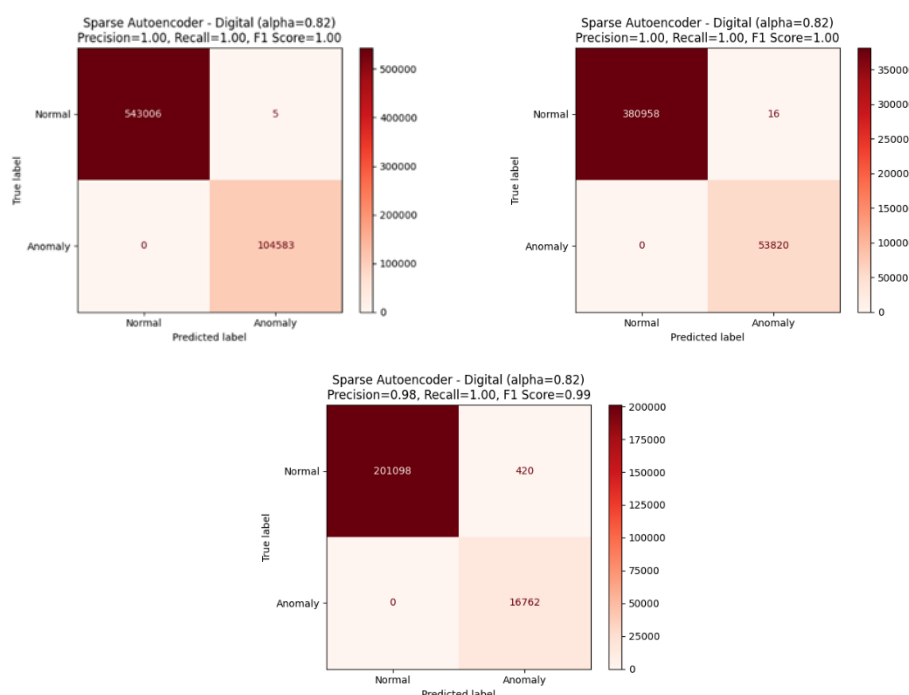


Figura 12: Matriz de confusão para os conjuntos de dados em sequência

Fonte: Resultados originais da pesquisa

Foi possível ver que houve uma ligeira piora no segundo e no terceiro conjunto de dados. Isso acontece, por se tratar de um sistema real mecânico existem flutuações comuns dos dados e isso pode ocasionar um leve desvio do padrão ideal dos dados, por isso é importante sempre acompanhar a eficácia do modelo. Pode ser que depois de algum tempo, seja necessário refazer o treino desconsiderando dados muito antigos, pois com o tempo o

sistema tende a mudar seu comportamento por desgastes ou variações inerentes de processos reais do porte de um sistema metroviário, visto que esse modelo aprende a reconhecer padrões de funcionamento com os dados, os mais antigos podem vir a se tornar obsoletos com o tempo.

Mesmo assim, se as métricas forem observadas, nunca houve um resultado muito ruim, indicando que o modelo reagiu de forma muito satisfatória.

O mais importante é que em todos os casos, os falso-negativos foram nulos, indicando que todas as anomalias foram de fato identificadas, que é o ponto mais relevante, pois em nenhum momento o modelo deixou de indicar uma possível falha com antecedência. O desvio nas métricas se dá por falso-positivos, onde o modelo indica um funcionamento normal como anômalo, isso pode levar a uma verificação manual dos operadores, que é esperada por se tratar de uma manutenção preditiva, de fato eleva o custo da operação, pois leva a verificações que seriam desnecessárias, mas ainda assim fica muito abaixo do esperado, ainda mais quando comparados os resultados obtidos pelos modelos de classificação.

Além disso, a abordagem proposta demonstrou ser robusta e adaptável a sistemas que operam continuamente, oferecendo uma solução promissora para a redução de custos operacionais e para o aumento da segurança e da confiabilidade dos sistemas.

Como trabalhos futuros, recomenda-se:

- A implementação de técnicas de otimização automática de hiperparâmetros, como “Bayesian Optimization” ou “Optuna”, visando maior eficiência no ajuste dos modelos;
- A integração de dados externos, como condições ambientais ou operacionais, que possam influenciar o comportamento do sistema;
- A exploração de outras arquiteturas baseadas em “Transformers” para séries temporais, como o “Informer” ou “Temporal Fusion Transformer”, que têm mostrado resultados promissores em previsões multivariadas;
- A aplicação do modelo em tempo real em ambiente industrial, com validação prática dos alertas gerados;
- E, por fim, o estudo de abordagens “online learning”, para adaptar os modelos continuamente conforme novas amostras são coletadas.

Essas extensões poderão ampliar ainda mais a aplicabilidade e a precisão dos sistemas preditivos, promovendo soluções mais inteligentes e proativas na manutenção industrial.

Considerações Finais

Este trabalho demonstrou a viabilidade da aplicação de técnicas de machine learning para prever falhas em sistemas de compressores do metrô de Portugal, contribuindo significativamente para a eficiência da manutenção preditiva. A partir da análise comparativa entre modelos de classificação e modelos de detecção de anomalias, verificou-se que os métodos baseados em “autoencoders” apresentaram melhor desempenho, especialmente o modelo SAE com dados digitais, que foi capaz de identificar todas as falhas com alta precisão e antecedência. Esse resultado reforça a importância de se utilizar dados multivariados e técnicas avançadas de modelagem em contextos industriais reais.

Referências

- Amorim, L.B.V.; Cavalcanti, G.D.C.; Cruz, R.M.O. 2022. The choice of scaling technique matters for classification performance. *Applied Soft Computing* 133(1). Elsevier, Amsterdã, Países Baixos.
- Bengio, Y. 2009. Learning deep architectures for AI. *Foundations and Trends in Machine Learning* 2(1): 1-127. Now Publishers, Hanover, MA, EUA.
- Box, G.E.P.; Jenkins, G.M.; Reinsel, G.C. 2015. *Time Series Analysis: Forecasting and Control*. 5ed. John Wiley & Sons, Hoboken, NJ, EUA.
- Breiman, L. 2001. Random Forests. *Machine Learning*, (45): 5-32. Kluwer Academic Publishers, Dordrecht, Países Baixos.
- Chandola, V.; Banerjee, A.; Kumar, V. 2009. Anomaly detection: A survey. *ACM Computing Surveys* 41(3): 15-58. Association for Computing Machinery (ACM), Nova Iorque, NY, EUA.
- Cortes, C.; Vapnik, V. 1995. Support-vector networks. *Machine Learning* 20(3): 273-297. Kluwer Academic Publishers, Dordrecht, Países Baixos.
- Cristianini, N.; Shawe-Taylor, J. 2000. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, Cambridge, Reino Unido.
- Veloso, B.; Ribeiro, R.; Gama, J.; Pereira, P. 2022. MetroPT-3: A dataset for predictive maintenance in metro systems. *Scientific Data* 9(1): 764. Nature Publishing Group, Londres, Reino Unido.
- Goodfellow, I.; Bengio, Y.; Courville, A. 2016. *Deep Learning*. 1ed. MIT Press, Cambridge, MA, EUA.
- Han, J.; Kamber, M.; Pei, J. 2011. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, Burlington, MA, EUA.
- Hyndman, R.J.; Athanasopoulos, G. 2021. *Forecasting: Principles and Practice*. 3ed. OTexts, Melbourne, VIC, Austrália.

James, G.; Witten, D.; Hastie, T.; Tibshirani, R. 2013. An Introduction to Statistical Learning: With Applications in R. 2ed. Springer, New York, NY, EUA.

Jardine, A.K.S.; Lin, D.; Banjevic, D. 2006. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing 20(7), 1483-1510. Academic Press, San Diego, CA, EUA.

Jolliffe, I.T. 2002. Principal Component Analysis. Springer Series in Statistics, New York, NY, EUA.

Kingma, D.P.; Welling, M. 2014. Auto-Encoding Variational Bayes. Proceedings of the 2nd International Conference on Learning Representations (ICLR): 1-14. Banff, AB, Canadá.

Kohavi, R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95) (2): 1137-1143. Morgan Kaufmann Publishers, San Francisco, CA, EUA.

Little, R.J.; Rubin, D.B. 2019. Statistical Analysis with Missing Data. 3ed. Wiley, Hoboken, NJ, EUA.

Mobley, R.K. 2002. An Introduction to Predictive Maintenance. 2ed. Butterworth-Heinemann, Oxford, Oxfordshire, Inglaterra.

Murphy, K.P. 2012. Machine Learning: A Probabilistic Perspective. 1ed. MIT Press, Cambridge, MA, EUA.

Nascimento, E.G.S.; Liang, J.S.; Figueiredo, I.S.; Guarieiro, L.L.N. 2022. T4PdM: a Deep Neural Network based on the Transformer Architecture for Fault Diagnosis of Rotating Machinery. IEEE, Piscataway, NJ, EUA.

Ng, A. 2011. Sparse Autoencoder. CS294A Lecture Notes: 1-19. Universidade de Stanford, Stanford, CA, EUA.

Pal, M. 2005. Random forest classifier for remote sensing classification. International Journal of Remote Sensing 26(1): 217-222. Taylor & Francis, Abingdon, Reino Unido.

Apêndice A

O repositório contendo todos os códigos-fonte desenvolvidos para este trabalho pode ser acessado em: <https://github.com/joaovprado/TCC-MBA-Previsao-de-falhas-em-sistemas-de-compressores-do-metro-de-Portugal>.