



ConBRepro

XIV CONGRESSO BRASILEIRO DE ENGENHARIA DE PRODUÇÃO

I CIGELUBRA

Congresso Internacional de Gestão e Engenharia Luso-Brasileiro

04 a 06 de dezembro de 2024

Energias Limpas nas Engenharias

Monitoramento De Sistemas De Energia Renovável Através De Aprendizado De Máquina Multiobjetivo

Gabriel Afonso Leal

Engenharia de Computação - Pontifícia Universidade Católica do Paraná (PUCPR)

Víctor Henrique Alves Ribeiro

Programa de Pós-Graduação em Engenharia de Produção e Sistemas - PUCPR

Resumo: A área de monitoramento de saúde estrutural tem ganhado crescente atenção, especialmente com o aumento da disponibilidade de energia eólica, onde sistemas de monitoramento das estruturas são fundamentais. A manutenção de estruturas eólicas gera custos significativos, tornando essencial a previsão do estado dos componentes para maximizar seu aproveitamento. Este estudo utiliza aprendizado de máquina e otimização multiobjetivo para monitorar a saúde das pás de aerogeradores, testando modelos de aprendizado de máquina em conjuntos de dados com métricas sobre o estado das pás. Dessa forma, é possível identificar impactos no desempenho do modelo e prever tanto casos de overfitting quanto cenários ideais.

Palavras-chave: Aerogerador, Detecção de Falhas, Aprendizagem de Máquina, Otimização Multiobjetivo

Monitoring of Renewable Energy Systems Through Multi-Objective Machine Learning

Abstract: The area of structural health monitoring has been receiving increasing attention, especially with the growing availability of wind energy, where structural monitoring systems are essential. Maintaining wind energy structures generates significant costs, making it crucial to predict component conditions to maximize their utilization. This study leverages machine learning and multi-objective optimization to monitor the health of wind turbine blades, testing machine learning models on datasets with metrics regarding blade conditions. This approach enables the identification of impacts on model performance, allowing for the prediction of both overfitting cases and ideal scenarios.

Keywords: Wind Turbine, Fault Detection, Machine Learning, Multi-Objective Optimization

1. Introdução

Na engenharia elétrica, um tema que vem ganhando grande relevância é o das energias renováveis, entre as quais a energia eólica é considerada eficaz e ecologicamente limpa, dado seu baixo impacto no ambiente ao redor. Diante disso, o monitoramento da qualidade

dos componentes de um gerador eólico é essencial para garantir a eficiência na geração de energia. No monitoramento da qualidade de um componente, é necessário analisar e classificar grandes conjuntos de dados. Para isso, uma tecnologia disponível são as técnicas de Machine Learning (ML). Como exposto em Dong et al. (2020), o Ensemble Learning (EL) é uma abordagem dentro das técnicas de ML que se destaca na execução de tarefas de monitoramento e detecção de falhas, apresentando-se como uma solução potencial para esse problema.

EL é capaz de criar diversos conjuntos de classificadores; no entanto, um subconjunto pode apresentar resultados superiores ao conjunto maior. É importante destacar que existe uma troca entre a quantidade de classificadores e sua acurácia, sendo possível otimizar a quantidade e a precisão para extrair um desempenho superior do conjunto gerado por EL. Dada a natureza do problema, é evidente a necessidade de equilibrar objetivos conflitantes. Como evidenciado em Alves Ribeiro e Reynoso-Meza (2020), para isso são desenvolvidas técnicas de otimização multiobjetivo, visando encontrar o ponto de maior equilíbrio na configuração dos classificadores. O presente artigo explora o uso de técnicas de aprendizado de máquina multiobjetivo para monitoramento de pás de aerogeradores, com o objetivo de identificar características essenciais para o bom funcionamento.

Por meio de testes e diferentes configurações, foi identificada a influência dos parâmetros iniciais sobre as métricas do sistema, destacando o impacto da quantidade de classificadores na complexidade do sistema e, conseqüentemente, em seu erro e precisão. O objetivo foi diminuir o número de classificadores para evitar o overfitting. Este artigo está organizado nas seguintes seções: *Deteção de falhas em pás de aerogeradores*, onde são apresentados os procedimentos de teste realizados para a criação da base de dados; *Materiais e métodos*, que aborda as tecnologias e métodos utilizados na análise dos dados; *Resultados*, onde são discutidos os achados dos experimentos; *Discussão*, com uma análise dos resultados e suas causas; e *Conclusão*, que resume o conteúdo do artigo e finaliza o raciocínio desenvolvido ao longo do texto.

2. Deteção de Falhas em Pás de Aerogeradores

A área de monitoramento de saúde estrutural (SHM) é de grande importância, pois concentra esforços para manter o desempenho dos componentes consistente e confiável. O SHM aplicado a estruturas de geração de energia eólica já está consolidado na indústria, como demonstrado pela presença de diversos estudos que visam garantir a eficiência das pás de aerogeradores durante seu funcionamento.

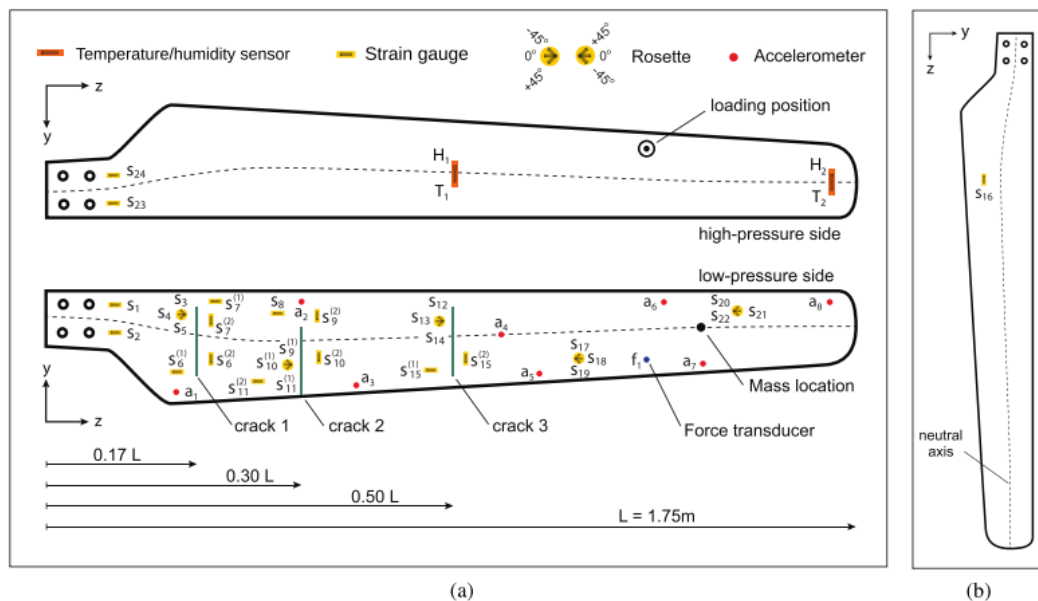
Com essa motivação, Ou et al. (2021) desenvolveram um conjunto de dados com as respostas de uma pá de aerogerador sob diversas condições. Os autores produziram duas réplicas das pás do modelo Windspot 3.5 kW da Sonkyo Energy, com 1,7 metros de comprimento e peso total de 5 kg. Para a realização dos testes, as duas pás foram fixadas, cada uma, a uma coluna metálica em um ambiente controlado, capaz de simular diferentes condições meteorológicas e tipos variados de vibração no objeto e na coluna de suporte. Uma pá foi utilizada como referência, enquanto a outra foi submetida a simulações de resposta à vibração com pesos em diferentes pontos da pá, variações de temperatura e presença de rachaduras na estrutura.

A Tabela 1 apresenta os diferentes conjuntos testados, em que cada estado foi submetido a variadas temperaturas no ambiente. Para a coleta dos dados, foram utilizados diversos tipos de sensores, incluindo pontes de strain gauge e acelerômetros, que foram posicionados conforme ilustrado na Figura 1.

Tabela 1 - Lista dos casos de experimentos e número de testes conduzidos para cada caso			
Casos	Descrição		
	Número de experimentos		
R	Estado saudável		
A	Massa adicionada 1 x 44g		
B	Massa adicionada 2 x 44g		
C	Massa adicionada 3 x 44g		
D	Rachadura 1: $l_1 = 5cm$		
E	Rachadura 1: $l_1 = 5cm$	Rachadura 2: $l_2 = 5cm$	6 por configuração de temperatura
F	Rachadura 1: $l_1 = 5cm$	Rachadura 2: $l_2 = 5cm$	Rachadura 3: $l_3 = 5cm$ 6 por configuração de temperatura
G	Rachadura 1: $l_1 = 10cm$	Rachadura 2: $l_2 = 5cm$	Rachadura 3: $l_3 = 5cm$ 6 por configuração de temperatura
H	Rachadura 1: $l_1 = 10cm$	Rachadura 2: $l_2 = 10cm$	Rachadura 3: $l_3 = 5cm$ 6 por configuração de temperatura
I	Rachadura 1: $l_1 = 10cm$	Rachadura 2: $l_2 = 10cm$	Rachadura 3: $l_3 = 10cm$ 6 por configuração de temperatura
J	Rachadura 1: $l_1 = 15cm$	Rachadura 2: $l_2 = 10cm$	Rachadura 3: $l_3 = 10cm$ 6 por configuração de temperatura
K	Rachadura 1: $l_1 = 15cm$	Rachadura 2: $l_2 = 15cm$	Rachadura 3: $l_3 = 10cm$ 6 por configuração de temperatura
L	Rachadura 1: $l_1 = 15cm$	Rachadura 2: $l_2 = 15cm$	Rachadura 3: $l_3 = 15cm$ 6 por configuração de temperatura

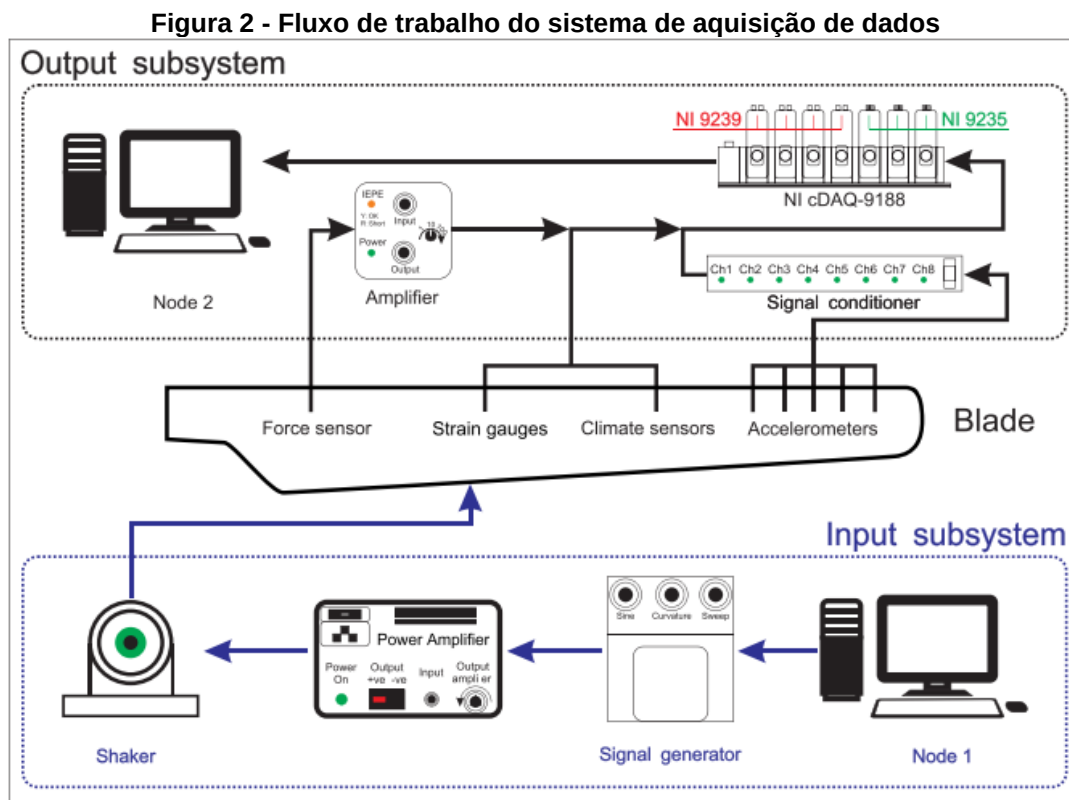
Fonte: OU et al., (2021)

Figura 1 - Configuração dos sensores no objeto de teste (a) e no objeto para compensação de temperatura (b)



Fonte: OU et al., (2021)

Todo o sistema de medição foi estruturado em dois subsistemas. O subsistema de entrada, responsável por definir as excitações aplicadas à amostra, apresentava dois tipos de excitação: um ruído branco com largura de banda de frequência efetiva entre 0 e 400 Hz e uma varredura senoidal com frequência variando de 1 a 300 Hz. O subsistema de saída era responsável por processar os dados enviados pelos sensores de força, pontes de *strain gauge*, sensores de clima e acelerômetros. O sistema de aquisição de dados está ilustrado na Figura 2.



Fonte: OU et al., (2021)

3. Materiais e Métodos

Este trecho apresenta as tecnologias utilizadas neste trabalho e é composto pelas seguintes subseções: Pré-processamento, que expõe o método utilizado para tratamento dos dados; e Machine Learning, que discute o aprendizado de máquina e o método relevante para o presente estudo.

3.1. Pré-Processamento

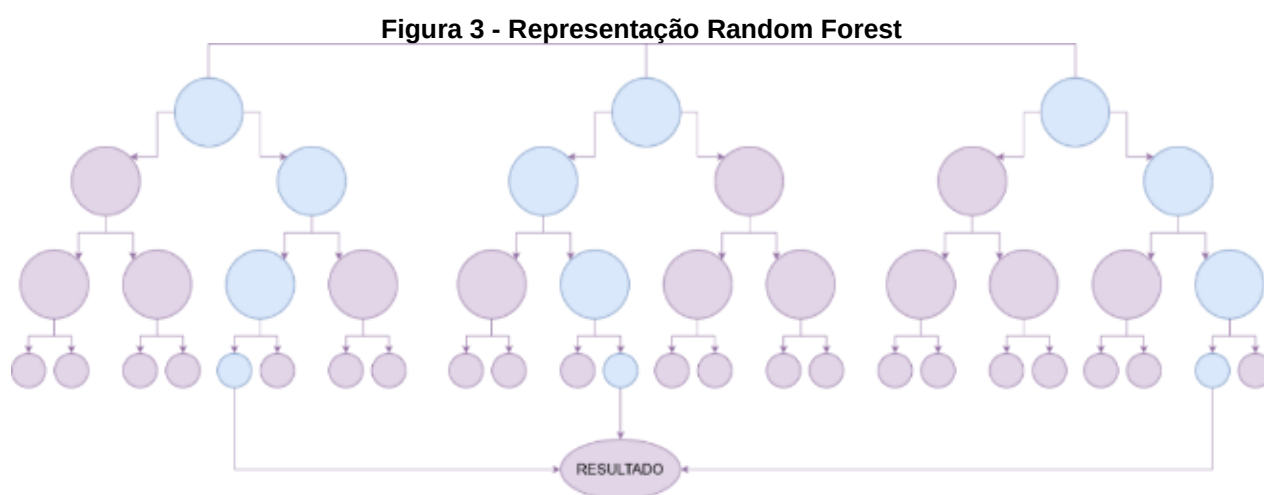
Para inserir os dados em um modelo de ML, é necessário primeiro processá-los, eliminando dados redundantes para evitar problemas como overfitting ou oversampling. Assim, este trabalho utiliza o pré-processamento proposto por Calderano et al. (2022), empregando dados dos acelerômetros para treinamento e teste. Esses dados foram selecionados a partir dos conjuntos A, B, C e R da Tabela 1 e passaram por uma modelagem autorregressiva com 10 componentes, seguida de uma análise de componentes principais (PCA). A PCA reduz a dimensionalidade do conjunto de dados, preservando a máxima variabilidade dos dados restantes e transformando um conjunto maior de variáveis correlacionadas em um conjunto menor de variáveis não correlacionadas.

Esses dados são então organizados em um estudo principal, no qual a temperatura dos casos é considerada. O conjunto de dados destinado ao desenvolvimento do modelo de ML inclui dados onde a temperatura é maior ou igual a zero, enquanto o conjunto de teste consiste em dados com temperaturas menores que zero. O objetivo do disto é medir a capacidade de generalização do modelo em diferentes condições.

3.2. Machine Learning

Como discutido em Hastie, Tibshirani e Friedman (2001), o aprendizado de máquina (ML) é amplamente utilizado em diversos tipos de problemas, sendo uma de suas principais aplicações a área de classificação. Nessa área, algoritmos de ML lidam com grandes volumes de dados e são capazes de classificá-los em diferentes classes. Nesse contexto, a aprendizagem supervisionada atua como um orientador do algoritmo, fornecendo dados previamente classificados para ajudar na identificação das classes presentes no conjunto de dados, de modo que, ao encontrar novos dados não classificados, o algoritmo possa categorizá-los nos grupos pré-estabelecidos. Assim, problemas de classificação incluem várias aplicações, como detecção de falhas e monitoramento.

O Random Forest é um algoritmo amplamente estabelecido nas técnicas de aprendizado de máquina, especialmente eficaz para tarefas de classificação, como demonstrado em Breiman (2001). Seu funcionamento é baseado na técnica de árvores de decisão, onde um conjunto de dados é classificado por meio de regras do tipo “se x for maior que y , então a , caso contrário, b ”. O Random Forest combina o resultado de várias árvores de decisão, onde, em problemas de classificação, a classe com maior número de votos entre as árvores é escolhida. Esse processo é ilustrado na Figura 3. Essa técnica se destaca pela robustez e precisão, pois, ao combinar os resultados de várias árvores, reduz o risco de overfitting, como explicado em detalhe por Fernández-Delgado et al. (2014). Além disso, utiliza uma técnica chamada bagging, na qual várias amostras são geradas a partir do conjunto original, de modo que para cada árvore o algoritmo seleciona uma amostra aleatória dos dados de treinamento, aumentando a eficiência e precisão do modelo.



Fonte: os autores

3.3. Avaliação

Para avaliar os modelos gerados no estudo, utiliza-se a acurácia, que indica o quão bem o modelo de ML classifica dados inéditos com base apenas no aprendizado adquirido na etapa de treinamento. O cálculo da acurácia é apresentado na Equação 1.

$$Acuracia = \frac{Tp + Tn}{Tp + Tn + Fp + Fn} \quad (1)$$

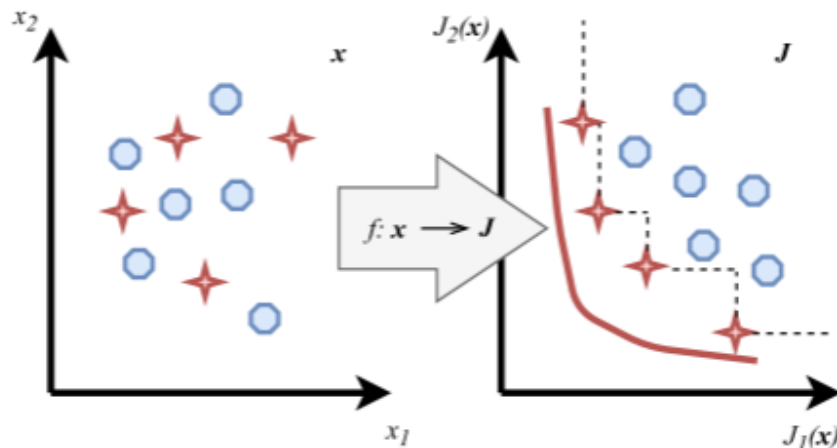
onde os *true positives* (verdadeiros positivos) são representados por Tp , os *true negatives* (verdadeiros negativos) por Tn , e Fp e Fn referem-se, respectivamente, a *false positives* (falsos positivos) e *false negatives* (falsos negativos).

3.4. Seleção de Classificadores

Como visto na Seção 3.2, o Random Forest utiliza árvores de decisão para classificação; entretanto, cada árvore possui suas próprias características, como precisão e erro. Dito isso, alguns classificadores no algoritmo podem ter uma influência mais negativa do que positiva no resultado final, tornando necessária a poda dessas árvores para que os classificadores restantes desempenhem suas funções de forma mais eficaz e precisa. Isso resulta em um modelo mais preciso e robusto.

Ao reduzir o número de classificadores, podem ser observadas algumas variações nas respostas do conjunto restante, incluindo o erro, que pode ter grande influência no resultado. Para lidar com isso, são utilizados algoritmos de otimização multiobjetivo, com o papel de balancear o ganho em performance com o erro obtido, buscando a melhor solução. O conjunto de classificadores que oferecem o melhor custo-benefício é chamado de frente de Pareto, onde se encontram os modelos mais robustos e precisos dentro dos classificadores do Random Forest. Esse processo é ilustrado na Figura 4.

Figura 4 - O exemplo do processo de otimização multiobjetivo (MOO), onde as variáveis de decisão (esquerda) gera os objetivos (direita). Os círculos representam as soluções denominadas pelas estrelas, onde são transformadas na aproximação da fronteira de Pareto (curva).



Fonte: Alves Ribeiro (2021)

3.5. Otimização Multiobjetivo

A otimização multiobjetivo é um tipo de otimização em que dois ou três objetivos são definidos, e o algoritmo busca a solução com o melhor custo-benefício entre esses objetivos. Esses objetivos podem ser definidos como métricas, como erro e complexidade, que são utilizadas para otimizar os modelos abordados neste artigo.

Para realizar a otimização multiobjetivo, é necessário primeiro definir o problema, incluindo suas limitações e requisitos. Nessa etapa, são selecionados os objetivos para otimização e as variáveis de decisão a serem manipuladas.

Essa relação entre problema e otimização exige uma seleção criteriosa dos classificadores, para a qual a técnica de *multi-objective ensemble member selection* (MOEMS), introduzida por Alves Ribeiro e Reynoso-Meza (2020), é particularmente valiosa. Essa técnica visa excluir modelos que possam impactar negativamente a performance de predição. A seleção é realizada por meio de um vetor binário, que indica quais classificadores são incluídos ou excluídos do conjunto. Assim, define-se o problema como:

$$\min_x J(x) = [J_{pred}(x), J_{comp}(x)] \quad (2)$$

sujeito a

$$x_i \in \{0,1\}, i \in [1, \dots, n] \quad (3)$$

$$\sum_{i=1}^n x_i \geq 1 \quad (4)$$

onde os objetivos são constituídos de: um conjunto de métricas de predição ($J_{pred}(x)$), para classificação ou regressão e as medidas de complexidade ($J_{comp}(x)$), como a proporção de modelos base selecionados. No vetor de decisão x , x_i , é a seleção binária de cada modelo base com n modelos diferentes, onde pelo menos um é selecionado

Nos experimentos realizados, utilizou-se o NSGA-3 como algoritmo de otimização. Trata-se de um algoritmo genético multiobjetivo, empregado para encontrar as melhores soluções dentro da fronteira de Pareto. Como discutido em Deb e Jain (2014) e Jain e Deb (2014), o NSGA-3 compartilha várias características com o NSGA-2. No entanto, uma melhoria importante no NSGA-3 é o uso de pontos de referência, que são distribuídos uniformemente e ajudam a guiar o algoritmo a encontrar soluções em diferentes regiões da fronteira de Pareto, associando cada solução ao ponto de referência mais próximo.

4. Resultados

Nesta etapa, discutem-se e analisam-se os resultados obtidos a partir da execução dos testes. Como mencionado na Seção 3.1, a separação dos dados é feita com base na temperatura, onde o grupo de teste inclui temperaturas abaixo de zero e o grupo de desenvolvimento, temperaturas acima de zero. O conjunto de desenvolvimento é separado aleatoriamente em 80% para treino da Random Forest e 20% para análise de erro durante a otimização. Este conjunto de dados foi submetido a diferentes configurações iniciais de classificador, número de gerações e tamanho da população, como apresentado na Tabela 2. A coluna de classificadores refere-se ao número de árvores presentes no algoritmo Random Forest, enquanto o número de gerações e o tamanho da população são configurações do algoritmo de otimização. O primeiro indica o máximo de gerações executadas em cada otimização e o segundo o número de testes realizados em cada geração.

A Tabela 3 apresenta os dados resultantes dos testes realizados com separação de temperatura. Os dados disponíveis incluem: a complexidade medida durante a otimização de cada conjunto de classificadores; o erro de teste; o erro de otimização; e o tempo necessário para a execução do treinamento. Os conjuntos complexidade e erro de otimização formam os pontos que correspondem à fronteira de Pareto.

O primeiro detalhe a ser observado nesta tabela é a variação no número de componentes na fronteira de Pareto entre os diferentes casos, destacando-se especialmente os casos 4 e 15, que apresenta quatro soluções distintas. Também vale mencionar a presença de diversos erros de otimização nulos, o que pode indicar overfitting durante a otimização.

Tabela 2 - Valores para execução dos testes

	Classificadores	Número de gerações	Tamanho de população
case 1	200	30	20
case 2	150	20	20
case 3	100	10	20
case 4	50	30	20
case 5	25	20	20
case 6	200	10	15
case 7	150	30	15
case 8	100	20	15
case 9	50	10	15
case 10	25	30	15
case 11	200	20	10
case 12	150	10	10
case 13	100	30	10
case 14	50	20	10
case 15	25	10	10

Fonte: os autores

Tabela 3 Tabela de resultados

	Complexidade	Erro de otimização	Erro de teste	Tempo de treinamento
caso 1	0,16	0	0,054908	82,059222
caso 2	0,233333	0	0,051581	42,262168
caso 3	0,24	0	0,048253	22,050148
caso 4	0,02	0,041551	0,136439	69,608983
	0,04	0,030471	0,232945	
	0,1	0	0,183028	
	0,06	0,00277	0,148087	
caso 5	0,12	0	0,209651	48,731366
	0,08	0,01385	0,144759	
	0,04	0,036011	0,141431	
caso 6	0,34	0	0,049917	17,958268
caso 7	0,193333	0	0,054908	46,379346
caso 8	0,15	0,00277	0,058236	32,122282
	0,17	0	0,048253	
caso 9	0,18	0	0,091514	16,212489
caso 10	0,12	0,00277	0,044925	55,799686
	0,16	0	0,049917	
	0,08	0,022161	0,046589	
caso 11	0,29	0	0,056572	21,617784
caso 12	0,346667	0	0,044925	13,418955
caso 13	0,22	0	0,071547	31,016017
caso 14	0,18	0	0,114809	22,505231
	0,16	0,00831	0,183028	
caso 15	0,24	0	0,156406	12,092394
	0,2	0,00277	0,178037	
	0,16	0,00554	0,236273	
	0,12	0,00831	0,246256	

Fonte: os autores

Entre todos os casos, destacam-se os testes 4, 5, 14 e 15, nos quais é possível observar o impacto evidente do baixo número de classificadores iniciais. Esse impacto se reflete na menor complexidade e em um erro de teste mais elevado. De forma contrária, os melhores resultados foram obtidos nos experimentos com maior disponibilidade inicial de

classificadores, o que permitiu ao modelo Random Forest desempenhar melhor sua função de generalização e evitar o overfitting.

5. Discussão

Ao se tratar de gerar modelos precisos e consistentes é importante considerar a complexidade e o erro de teste em conjunto, pois neste se concentra características importantes para elevar o acerto em validação e em execuções posteriores.

A generalização é uma forte ferramenta disponível ao se tratar de EL juntamente com MOO, no entanto, este último desempenha o papel de equilibrar a disponibilidade de classificadores através da poda de classificadores, para que se possa se afastar de algoritmos com overfitting, contudo salienta-se a observação da quantidade de podas feitas pois esta pode gerar um modelo mal aproveitado e acomodado com os dados de teste.

O Random Forest é um modelo de EL que possui a vantagem de uma melhor generalização com um maior número de classificadores. No entanto, quando o número inicial de classificadores é baixo, como observado nos testes, o processo de otimização multi-objetivo causa overfitting, visto que os poucos classificadores selecionados resultam em baixa regularização. Portanto, destaca-se a importância de remover classificadores com cautela, pois uma quantidade muito baixa de classificadores finais pode comprometer a acurácia e aumentar o risco de overfitting.

6. Conclusões

O extenso uso de ML tem impulsionado inovações na área, e SHM se beneficia amplamente disso, pois a utilização de técnicas de EL com MOO permite a análise de dados de pás de aerogeradores, visando a detecção de anomalias na estrutura. Através de testes com diferentes parâmetros iniciais, foi possível observar o comportamento de um sistema Random Forest com otimização multiobjetivo integrada, selecionando componentes por meio de MOEMS.

Observou-se que a quantidade inicial de classificadores disponibilizada ao algoritmo Random Forest pode limitar seu desempenho durante a otimização, seja pela poda excessiva de classificadores ou pelo aproveitamento inadequado da estrutura, o que resulta em baixa complexidade e em erros de teste maiores do que os de validação, levando ao overfitting nos dados de treino.

Diante disso, o objetivo deste estudo é aprofundar o conhecimento sobre a aplicação de aprendizado de máquina multiobjetivo, além de desenvolver métodos portáteis e leves para a detecção de anomalias em pás, visando aumentar a eficiência e consistência desses equipamentos.

Os experimentos realizados enfrentaram limitações de poder de processamento e de quantidade de dados analisados, restringindo os resultados a uma fração dos classificadores e características dos dados. Para trabalhos futuros, é relevante explorar o impacto da inclusão de diferentes medições no conjunto de dados, como as pontes de gauge.

Referências

ALVES RIBEIRO, V. H.; REYNOSO-MEZA, G. Ensemble learning by means of a multi-objective optimization design approach for dealing with imbalanced data sets. **Expert Systems with Applications**, v. 147, p. 113232, 2020

ALVES RIBEIRO, V. H. Multi-objective Ensemble Learning: A Case Study with Data-driven Soft Sensors. Curitiba, 213 p., 2021. Tese (Doutorado) – Pontifícia Universidade Católica do Paraná.

BREIMAN, Leo. Random forests. **Machine learning**, v. 45, p. 5-32, 2001.

CALDERANO, Pedro; DE MARINS, Daniel B.; AYALA, Helon. A comparison of Feature Extraction Methods for Crack and Ice Monitoring in Wind Turbine Blades: System Identification and Matrix Decomposition. In: 2022 30th Mediterranean Conference on Control and Automation (MED). IEEE, 2022. p. 779-784.

DEB, Kalyanmoy et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, v. 6, n. 2, p. 182-197, 2002.

DEB, Kalyanmoy; JAIN, Himanshu. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. **IEEE transactions on evolutionary computation**, v. 18, n. 4, p. 577-601, 2013.

DONG, Xibin et al. A survey on ensemble learning. **Frontiers of Computer Science**, v. 14, p. 241-258, 2020.

FERNÁNDEZ-DELGADO, Manuel et al. Do we need hundreds of classifiers to solve real world classification problems?. **The journal of machine learning research**, v. 15, n. 1, p. 3133-3181, 2014.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. The elements of statistical learning: data mining, inference, and prediction. 2017.

JAIN, Himanshu; DEB, Kalyanmoy. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach. **IEEE Transactions on evolutionary computation**, v. 18, n. 4, p. 602-622, 2013.

OU, Yaowen et al. Vibration-based monitoring of a small-scale wind turbine blade under varying climate conditions. Part I: An experimental benchmark. **Structural Control and Health Monitoring**, v. 28, n. 6, p. e2660, 2021.