UFBA

UNIVERSIDADE FEDERAL DA BAHIA
ESCOLA POLITÉCNICA
PROGRAMA DE PÓS GRADUAÇÃO EM
ENGENHARIA INDUSTRIAL - PEI

MESTRADO EM ENGENHARIA INDUSTRIAL

RODRIGO MARCEL ARAUJO OLIVEIRA

Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura

PEI
Programa de Pós-Graduação em Engenharia Industrial

SALVADOR
2023

UNIVERSIDADE FEDERAL DA BAHIA
ESCOLA POLITÉCNICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA INDUSTRIAL

RODRIGO MARCEL ARAUJO OLIVEIRA

Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura

Salvador
2023

RODRIGO MARCEL ARAUJO OLIVEIRA

Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura

Dissertação apresentada à Escola Politécnica da Universidade Federal da Bahia para obtenção do título de Mestre em Engenharia pelo Programa de Pós-graduação em Engenharia Industrial.

Área de concentração: Pesquisa Operacional

Orientador: Prof. Dr. Ângelo Márcio Oliveira Sant'Anna

Salvador

2023

Ficha catalográfica elaborada pelo Sistema Universitário de Bibliotecas (SIBI/UFBA),
com os dados fornecidos pelo(a) autor(a).

Marcel Araujo Oliveira, Rodrigo
Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura / Rodrigo Marcel Araujo Oliveira. -- Salvador, 2023.
120 f.

Orientador: Ângelo Sant'Anna.
Dissertação (Mestrado - Programa de Pós-Graduação em Engenharia Industrial) -- Universidade Federal da Bahia, Escola Politécnica, 2023.

1. Aprendizado de Máquina. 2. Classificação Multiclasse. 3. Classificação Binária. 4. Detecção de Anomalias. 5. Processos de Manufatura. I. Sant'Anna, Ângelo. II. Título.

Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura

RODRIGO MARCEL ARAUJO OLIVEIRA

Dissertação submetida ao corpo docente do Programa de Pós-graduação em Engenharia Industrial da Universidade Federal da Bahia como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Industrial.

Examinada por:

Prof. Dr. Ângelo Márcio Oliveira Sant'Anna

Doutor em Engenharia de Produção, Universidade Federal do Rio Grande do Sul, Brasil, 2009.

Orientador - PEI/UFBA

Prof. Dr. Paulo Henrique Ferreira da Silva

Doutor em Estatística, Universidade Federal de São Carlos, Brasil, 2015.

Membro Interno - DEST/UFBA

Prof. Dr. Danilo Marcondes Filho

Doutor em Engenharia de Produção, Universidade Federal do Rio Grande do Sul, Brasil, 2009.

Membro Externo - PPGEST/UFRGS

Salvador, BA - Brasil
Dezembro/2023

Dedico à minha companheira, Bruna Toth Santos, por todo apoio, paciência, amor e carinho, sem ela nada disso seria possível.

Agradecimentos

Agradeço a Deus, por sempre ter me abençoado, me conduzindo para alcançar meus sonhos e objetivos, por colocar anjos no meu caminho, pelas oportunidades, por me dar sabedoria para fazer as escolhas corretas.

Agradeço à minha companheira, Bruna Toth Santos, pelo amor, carinho, incentivo e apoio nessa jornada, sem ela nada disso seria possível. Agradeço as minhas cachorras, Dorinha, Pérola e Pandora, que são anjos na terra que nos dão todo suporte e força para continuar, foram e são fundamentais para manter saudável minha saúde mental.

À minha mãe, Maria do Carmo, que sempre acreditou nos meus sonhos, fazendo de tudo que estava ao seu alcance para se tornar realidade.

Ao meu padrasto, Fábio, que me incentivou nos meus estudos e sempre apoio para que tudo desse certo.

Às minhas irmãs, Patricia e Sabrina, pelo carinho e incentivo.

Agradeço ao meu orientador, professor Ângelo, pela paciência, dedicação, simplicidade, humildade, pelas oportunidades, sem dúvidas foi um privilégio aprender com ele.

Agradeço, amigos e colegas pelo incentivo.

Agradeço ao PEI e à UFBA, por todo suporte técnico e administrativo. Sou eternamente grato, por tudo!

"Se a educação sozinha não transforma a sociedade,

sem ela tampouco a sociedade muda."

(Paulo Freire)

Resumo

OLIVEIRA, RODRIGO MARCEL ARAUJO. **Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura**. 2023. 118 p. Dissertação (Mestrado em Engenharia Industrial) – Escola Politécnica, Univerdade Federal da Bahia, Salvador, 2023.

Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura estão cada vez mais presentes no contexto da Indústria 4.0. Esse cenário permite que fábricas desenvolvam novas metodologias para o monitoramento e controle de qualidade de seus produtos, obtendo melhores indicadores de eficiência operacional e oferecendo produtos cada vez mais competitivos no mercado. A interpretabilidade dos modelos de aprendizado de máquina pode facilitar na compreensão de como os modelos tomam decisões e podem auxiliar no rastreamento para detecção de defeitos e anomalias. Este trabalho apresenta resultados de modelos de aprendizado de máquina supervisionados e não supervisionados para detecção de defeitos e de anomalias no processo de manufatura no contexto de uma indústria multinacional. O trabalho tem como objetivo desenvolver modelos de aprendizado de máquina para classificação de atributos multiclasses, binários e modelos de detecção de anomalias. Os dados são provenientes do processo de uniformidade de pneus. Os modelos *Random Forest*, *Gradient Boosting Decision Tree*, *Ligth Gradient Boosting Machine*, *Logistic Regression*, *Support Vector Machine*, *Multi-layer Percepton*, *K-Nearest Neighbor*, *Gaussian Processes Classification* foram considerados para classificação do desempenho de pneus em conformidade com os padrões de produção. Os métodos *Local Interpretable Model-Agnostic Explanations* e *SHapley Additive exPlanations* foram utilizadas para interpretação dos modelos. No contexto de classificação multiclasse, o modelo *Random Forest* apresentou resultados robustos e satisfatórios para classificação do desempenho dos pneus. O modelo também apresentou resultados adequados para detecção de defeitos no contexto de classificação binária, todavia o modelo *Logistic Regression* obteve resultados comparáveis segundo o teste estatístico *McNemar*. A *Logistic Regression* foi utilizada para determinar valores de referência para cada variável preditora selecionada pelo modelo para manufatura de pneus em conformidade com os padrões de qualidade. No contexto de detecção de anomalias foram considerados os modelos: *Isolation Forest*; *Local Outlier Factor*; *One Class Support Vector Machine*; *Elliptic Envelop*. O modelo *Isolation Forest* apresentou resultados satisfatórios para detecção de anomalias, com auxílio da técnica *SHapley Additive exPlanations* foi possível identificar quais foram as variáveis com maior influência no modelo. As diferentes abordagens são ferramentas relevantes e fornecem soluções robustas para garantir a eficiência da gestão do controle de qualidade em processos de manufatura.

Palavras-chaves: Aprendizado de Máquina, Classificação Multiclasse, Classificação Binária, Detecção de Anomalias, Processos de Manufatura.

Abstract

OLIVEIRA, RODRIGO MARCEL ARAUJO. **Abordagens de aprendizado de máquina para reconhecimento de padrões em processos de manufatura**. 2023. 118 p. Dissertation (Master's degree in Industrial Engineering) – Polytechnic School, Federal University of Bahia, Salvador, Bahia, 2023.

Machine learning approaches for pattern recognition in manufacturing processes are increasingly prevalent in the context of Industry 4.0. This scenario allows factories to develop new methodologies for monitoring and controlling the quality of their products, obtaining better indicators of operational efficiency, and offering increasingly competitive products in the market. The interpretability of machine learning models can facilitate understanding how models make decisions and can assist in tracking for defect detection and anomalies. This work presents results from supervised and unsupervised machine learning models for defect and anomaly detection in the manufacturing process within the context of a multinational industry. The goal of this work is to develop machine learning models for multi-class and binary attribute classification, as well as anomaly detection models. The data come from the tire uniformity process. The Random Forest, Gradient Boosting Decision Tree, Light Gradient Boosting Machine, Logistic Regression, Support Vector Machine, Multi-layer Perceptron, K-Nearest Neighbor, and Gaussian Processes Classification models were considered for classifying tire performance in compliance with production standards. The Local Interpretable Model-Agnostic Explanations and SHapley Additive exPlanations methodologies were used for model interpretation. In the context of multi-class classification, the Random Forest model yielded robust and satisfactory results for classifying tire performance. The model also showed adequate results for defect detection in binary classification; however, the Logistic Regression model achieved comparable results according to the McNemar's statistical test. Logistic Regression was used to determine reference values for each selected predictor variable for tire manufacturing in compliance with quality standards. For anomaly detection, the following models were considered: Isolation Forest, Local Outlier Factor, One Class Support Vector Machine, and Elliptic Envelope. The Isolation Forest model provided satisfactory results for anomaly detection. With the help of the SHapley Additive exPlanations technique, it was possible to identify the variables with the greatest influence on the model. These different approaches are relevant tools and provide robust solutions to ensure the efficiency of quality control management in manufacturing processes.

Keywords: Machine Learning, Multiclass Classification, Binary Classification, Anomaly Detection, Manufacturing Processes.

Sumário

[1] De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

1 Introdução

O desenvolvimento de fábricas inteligentes é caracterizado pela integração de diferentes tecnologias de conectividade e processamento de dados, esse efeito é consequência do processo industrial revolucionário denominado por Indústria 4.0. A introdução de tecnologias como *Internet of Things* (IoT), robótica e Inteligência Artificial (IA), são os pilares para Indústria 4.0, isso possibilita grandes redes de conexões e trânsito de informações em tempo real. A integração desses sistemas de produção gera uma grande quantidade de dados para análise, permitindo o monitoramento de processos para manutenção e detecção de falhas (CARVALHO et al., 2019).

No contexto da Indústria 4.0 exige-se que fábricas desenvolvam soluções cada vez mais complexas para lidar com resolução de problemas baseados em decisões multicritério, reconhecimento de padrões não lineares e previsão de resultados. Os sistemas baseados em IA tornam-se fundamentais para essa transformação, possibilitam que empresas desenvolvam soluções para mitigar riscos operacionais e antecipar possíveis falhas no processo produtivo e melhorar a qualidade dos produtos (ZHAN et al., 2021).

O aprendizado de máquina é uma sub-área da IA que trata de algoritmos poderosos que permitem detectar mudanças de padrões nos dados; esses modelos são capazes de fazer inferências sobre os dados e detectar anomalias. Isso possibilita que empresas façam a intervenção de qualquer divergência no processo de manufatura de forma instantânea, permitindo a manutenção preventiva, aumentando a produtividade e, consequentemente, reduzindo custos operacionais (ACOSTA et al., 2023). A detecção de anomalias pode ser realizada por diferentes algoritmos de aprendizado de máquina; anomalias são eventos que não estão em conformidade com o padrão esperado em um conjunto de dados. Esse procedimento vem sendo adotado por empresas para otimização de processos que são essenciais para a garantia dos padrões de qualidade de fabricação (KAMOONA et al., 2021).

O uso de diferentes algoritmos de aprendizado de máquina para problemas de regressão, envolvendo variáveis de tipo misto em vários conjuntos de dados relacionados a pneus, foi objeto de estudo em Gutierrez-Gomez, Petry e Khadraoui (2020a). Os autores Acosta et al. (2021) utilizaram os modelos *Random Forest*, *Artificial Neural Network*,

K-Nearest Neighbors e *Relevance Vector Machine* para prever níveis de concentração de fósforo em uma siderurgia.

Aplicações de *Convulational Neural Network* foram utilizadas em Rajeswari et al. (2022), com o objetivo de descobrir imperfeições em pneus por meio do exame de imagens de projetos de trilhas de pneus. A quilometragem de um veículo depende de diferentes componentes e parâmetros, como condição da rua, peso do pneu, etc. O artigo concentra-se no exame do projeto de pista de pneu para deformidades. O sistema proposto utiliza um conjunto de dados de diferentes fabricantes de pneus. O resultado do experimento demonstra que é concebível prever as deformidades e a durabilidade de um pneu a partir de seus projetos de pista por meio de uma rede neural convolucional baseada em qualidade ponderada.

Os algoritmos *Relevance Vector Machine* e *Artificial Neural Network* foram utilizados para prever a tensão triaxial de contato do pneu com a estrada em Li, Guo e Zhou (2021), trata-se de um fator importante para avaliação do desempenho do veículo e do material da superfície da estrada. Esse trabalho apresentou uma abordagem para melhorar a precisão da previsão em todas as direções, prevendo tensões uniaxiais e triaxiais a cada passo, quando comparado a outros modelos de regressão de múltiplas saídas. O uso de *Artificial Neural Network* também foi objeto de estudo dos autores Lee et al. (2021), o trabalho objetiva prever o nível de ruído associado ao padrão do pneu, processo fundamental no estágio inicial de projeto do padrão do pneu.

Os autores Zhu, Liu e Yan (2021) apresentam um método para previsão da vida útil de pneus baseado em processamento de imagens e aprendizado de máquina. Em Xu et al. (2022) apresentam uma abordagem para estimativa da força dos pneus usando técnicas de redes neurais. O estudo apresenta um sistema de pneu inteligente com um sensor de aceleração triaxial, que é instalado no revestimento interno do pneu, assim técnicas de rede neural são usadas para processamento em tempo real dos dados do sensor. O resultado mostra-se que o pneu inteligente desenvolvido em conjunto com o aprendizado de máquina é eficaz na previsão precisa das forças do pneu sob diferentes condições de condução.

O presente trabalho consiste no desenvolvimento de algumas abordagens de modelos de aprendizado de máquina supervisionado e não supervisionado para detecção de defeitos em processos de manufatura no contexto de uma fábrica multinacional de pneus. O aprendizado supervisionado requer que os dados sejam previamente rotulados para o treinamento, trata-se de algoritmos para classificação usados para prever a classe de uma

instância inédita de dados sem rótulo. Os algoritmos para modelos não supervisionados não utilizam rótulos e têm a finalidade de encontrar padrões e grupos (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Essas abordagens de modelos com estruturas distintas são capazes de fazer o reconhecimento de padrões não lineares e fornecer soluções para diversos tipos de problemas; existem diferentes algoritmos para isso, modelos baseados em árvores, grafos, distâncias, distribuições de probabilidade.

O desgaste de pneus causa um grande número de acidentes de trânsito, os pneus são componentes importantes dos automóveis, é um dos principais fatores que determinam a segurança do veículo (ZHU; HAN; WANG, 2021). Em decorrência do crescimento do número de veículos nas estradas, anualmente, cerca de 1000 milhões de pneus inservíveis são descartados em todo o mundo (WANG et al., 2019). Os pneus inservíveis quando descartados em locais a céu aberto estão sujeitos a queimadas, contaminação do solo, propagação de doenças, além de representar outros riscos socioambientais (PEDRAM et al., 2017). No Brasil, a resolução do Conselho Nacional do Meio Ambiente (CONAMA) nº 258 de 26/08/99 regulamentou normas referentes à disposição dos pneus descartados e responsabilizou produtores e importadores pela destinação final adequada dos pneus inservíveis. No ano de 2010, foi aprovada a Lei 12.305 que institui a Política Nacional de Resíduos Sólidos, que também regula o descarte dos pneus (LAGARINHOS; TENÓRIO, 2013). Adotar estratégias inteligentes de manufatura de pneus é essencial para detecção de falhas e melhorar a qualidade dos produtos, consequentemente, aumentar o tempo de vida útil desses, e portanto, diminuir a quantidade de pneus inservíveis no meio ambiente.

1.1 Justificativa

O contexto da Indústria 4.0 possibilitou um novo ambiente competitivo e de inovação para as empresas, favorecendo soluções cada vez mais pautadas na sustentabilidade de processos produtivos. Esse novo paradigma de produção permite que empresas desenvolvam soluções cada vez mais robustas e customizáveis, assim como o desenvolvimento de novos modelos de negócio e implementação de projetos que garantam a integridade de toda a rede de valor, portanto é um cenário propício para desenvolvimento de novas abordagens tecnológicas utilizando dados (MAO et al., 2019).

O reconhecimento de padrões não lineares com novos recursos técnicos de aprendizado de máquina tem sido utilizado em vários setores da indústria para detectar defeitos e otimizar processos (GUTIERREZ-GOMEZ; PETRY; KHADRAOUI, 2020b). A interpretação de modelos de aprendizado de máquina é um desafio para a comunidade acadêmica, estudos mostram que técnicas como *SHapley Additive exPlanations* é uma alternativa para explicabilidade de modelos como *Random Forest* e *Light Gradient Boosting Machine* (SANTOS et al., 2023). Isso pode auxiliar na definição de políticas de riscos operacionais para tomada de decisões na contenção de falhas em processos de manufatura.

A seleção dos melhores parâmetros de modelos de aprendizado de máquina é uma etapa muito importante e não trivial no processo de modelagem (ACOSTA et al., 2023). Comumente, algoritmos inspirados na evolução natural são utilizados para problemas de otimização e busca em geral (CARVALHO; FARIAS; FILHO, 2023). O uso desses algoritmos para otimização de parâmetros de modelos de aprendizado de máquina é promissor; estudos relatam contribuições significativas no poder discriminativo dos modelos (RUAN et al., 2023). Para avaliar a capacidade de generalização dos modelos, os métodos de validação cruzada são fundamentais (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Existem diversos tipos de metodologias, métodos baseados em amostras estratificadas, em lotes de amostra para treinamento e validação. A escolha correta desse método também pode influenciar nos resultados da otimização de parâmetros do modelo.

O presente trabalho contempla diversas abordagens de modelagem para detecção de padrões nos dados, que não estão em conformidade com os critérios de qualidade estabelecidos pela fábrica. O estudo é inédito por se tratar da classificação do desempenho de pneus em função de dados de variáveis explicativas oriundas do processo de uniformidade dos pneus. A classificação no contexto de duas ou mais variáveis respostas é discutida, também são apresentadas abordagens de modelos de detecção de anomalias e o uso de técnicas para otimização de parâmetros por algoritmos genéticos e de inferência Bayesiana dos modelos de aprendizado. Foram utilizadas técnicas de amostragem para contornar problemas com número pequeno de amostras para treinamento dos modelos. O estudo apresenta uma comparação estatística dos diferentes resultados dos modelos de aprendizado. Metodologias como *SHapley Additive exPlanations* e *Local Interpretable Model-Agnostic Explanations* foram utilizadas para interpretabilidade dos resultados dos modelos, com a finalidade de identificar as variáveis com maior contribuição para identificação de pneus não conformes com os padrões de fabricação.

1.2 Objetivos da pesquisa

Objetivo geral

O presente trabalho propõe abordagens de aprendizado de máquina supervisionado e não supervisionado para reconhecimento de padrões no processo de manufatura em uma indústria multinacional.

Objetivos específicos

A dissertação objetiva compreender diferentes abordagens de reconhecimento de padrões com modelos de aprendizado de máquina, os objetivos específicos estabelecidos foram:

1. Propor modelos de aprendizado de máquina supervisionado para classificação de atributos multiclasses;
2. Propor modelos de aprendizado de máquina supervisionado para classificação de atributos binários;
3. Propor modelos de aprendizado de máquina não supervisionado para detecção de anomalias.

1.3 Método de pesquisa

Os artigos propostos nesta dissertação baseiam-se na lógica hipotética-dedutiva. Essa metodologia é caracterizada pela construção de hipóteses oriundas de problemas teóricos e práticos, esse conceito pode ser classificado por uma pesquisa aplicada (PRODANOV; FREITAS, 2013). A dissertação é caracterizada por objetivos exploratórios e experimentais, os artigos se baseiam em metodologias empíricas e teóricas para validação das hipóteses da pesquisa. O estudo tem foco na modelagem estatística e computacional; trata-se de uma pesquisa quantitativa.

1.4 Delimitações da pesquisa

O escopo desta pesquisa consiste em analisar dados de um estudo realizado em uma indústria multinacional que desenvolve pneus para carros de passeio. Os dados coletados são provenientes do sistema inteligente de manufatura da fábrica. A variável resposta deste estudo é o desempenho do pneu que é um indicador de grande importância para a fábrica. O pneu é inspecionado quanto ao tamanho, massa e forças irregulares sob certas condições de pressão, carga e velocidade (CHO et al., 2012). Esse conjunto de processos é denominado teste de uniformidade e está condicionado aos padrões aceitos por fabricantes globais de pneus, como forma de identificar pneus com baixo desempenho para que não sejam vendidos no mercado (DU et al., 2020). O estudo inclui a variável resposta ordinal com três categorias de desempenho: A, B e C. O melhor desempenho do pneu é representado pela categoria A. Essa avaliação é realizada por montadoras de automóveis que submetem aos produtores de pneus diversos testes de qualidades, com a finalidade de identificar quais são os produtos em conformidade com o padrão esperado (LIU et al., 2023). As variáveis explicativas são: flutuação da força radial (RFV); flutuação de força lateral (LFV); excentricidade radial (RRO); efeito de conicidade (CONICITY); efeito angular (PLY); força radial 1° (H1RFV) e 2° harmônico (H2RFV); força lateral 1° harmônico (H1LFV); excentricidade radial 1° harmônico (H1RRO); emenda de tampa (CAPSPLICE). A unidade de medida das forças segue o padrão internacional, isto é, Newton.

1.5 Relação entre os artigos

A dissertação está fundamentada em três artigos; essa organização está sintetizada na Tabela 1. O artigo 1 (*Machine learning approaches for multi-class classification in smart manufacturing processes*) contempla análises para investigar a relação entre as características das variáveis explicativas e associação com desempenho do pneu. Essa pesquisa tem como objetivo avaliar diferentes modelos de aprendizado de máquina para classificação do desempenho do pneu com categorias A, B e C, sendo A o melhor desempenho do pneu. O estudo foi realizado com auxílio da linguagem de programação `Python`. Inicialmente, foi realizada uma análise descritiva e com auxílio da técnica de análise de componentes principais foi possível identificar três grupos de desempenho de pneus.

Para seleção de variáveis foi aplicada a metodologia informação mútua, de modo que as variáveis selecionadas foram aquelas com correlação baixa entre si e com maior associação com as variáveis respostas. Uma das limitações do estudo foi o número de amostras em cada categoria; para contornar esse problema foram adotadas técnicas estatísticas para balanceamentos dos dados, como *Synthetic Minority Oversampling Technique* (SMOTE) com intuito de aumentar o número de exemplos nos conjuntos de treinamentos, além disso, para validação cruzada, o método *Leave-one-out* foi adotado para aproveitar o maior número de informações. Diferentes modelos foram considerados na etapa de treinamento, são eles: *Random Forest*; *Gradient Boosting Decision Tree*; *Light Gradient Boosting Machine*; *Logistic Regression*; *Support Vector Machine*; *K-Nearest Neighbor*; *Gaussian Processes Classification*; *Multi-layer Perceptron*. A otimização de parâmetros foi realizada com auxílio do método Bayesiano *BayesSearchCV* que isso possibilitou encontrar soluções mais robustas para resultados dos modelos. O modelo *Random Forest* apresentou os melhores resultados, com base nestes resultados foi realizada uma análise para avaliar quais variáveis eram importantes para classificação do desempenho dos pneus. Este artigo foi submetido para *Journal of Intelligent Manufacturing* em setembro de 2023.

O artigo 2 (*Explainable machine learning models for defects detection in industrial processes*) propõe avaliar o desempenho de modelos de aprendizado de máquina supervisionados para detecção de pneus em conformidade e não conformes com os padrões esperados de fabricação. As linguagens de programação `Python` e `R` foram utilizadas para desenvolvimento da pesquisa. O estudo tem a finalidade de encontrar valores de referência das variáveis com maior impacto para classificação de pneus em conformidade; a partir do estudo anterior a variável resposta foi modificada para duas categorias de desempenho, isto é, A representando pneus em conformidade com padrão de fabricação, e os desempenhos B e C denotados por pneus não conformes. O estudo contempla um conjunto de variáveis menor, pois parte da seleção de variáveis proposta no artigo 1. A distribuição das variáveis preditoras foi comparada entre os grupos de pneus com auxílio do teste *Kruskal-Wallis*. Os modelos avaliados para treinamento foram: *Random Forest*; *Light Gradient Boosting Machine*; *Logistic Regression*; *Support Vector Machine*; *Multi-layer Perceptron*. A etapa de seleção de parâmetros foi realizada com *GASearchCV*, uma metodologia baseada em algoritmos genéticos, em conjunto com método de validação cruzada *Leave-one-out*. O teste de *McNemar* foi considerado para comparar os resultados dos modelos no conjunto de validação. Para avaliação do impacto de cada variável preditora para decisão de classi-

Tabela 1 – Estrutura dos artigos de acordo com seus objetivos.

Artigo 1	Artigo 2	Artigo 3
Objetivo específico: Propor modelos de aprendizado de máquina para classificação de atributos multiclasses.	**Objetivo específico:** Propor modelos de aprendizado de máquina para classificação de atributos binários.	**Objetivo específico:** Propor modelos de aprendizado de máquina para detecção de anomalias.
Proposição: Avaliar desempenho de diferentes algoritmos de aprendizado de máquina para classificação do desempenho dos pneus das categorias A, B e C; investigar a relação entre as características das variáveis explicativas com desempenho do pneu; identificar quais são as variáveis explicativas do processo de uniformidade dos pneus que têm maior impacto para detectar o desempenho do pneu.	**Proposição:** Avaliar e comparar o desempenho de modelos de aprendizado de máquina para classificação de pneus em conformidade com os padrões esperados, isto é, desempenho A e pneus com desempenhos B e C; importância das variáveis preditoras no resultado do modelo; propor um método para identificar quais são os limiares das variáveis explicativas com maior impacto para detectar o desempenho do pneu na fronteira de classificação do modelo.	**Proposição:** Avaliar e comparar diferentes algoritmos de aprendizado de máquina não supervisionados para detecção de anomalias e avaliar a concordância dos resultados com desempenho dos pneus; identificar quais são as variáveis explicativas com maior impacto para detecção de anomalias.
Questionamentos: É possível discriminar o desempenho de pneus em função das variáveis do processo de uniformidade dos pneus. Quais estruturas de modelos de aprendizado de máquina que são capazes de fazer reconhecimento de padrões não lineares para classificação do desempenho do pneu? Como determinar as variáveis com maior poder discriminativo para classificação?	**Questionamentos:** Quais estruturas de modelos de aprendizado de máquina são aderentes para detecção de pneus defeituosos? Quais variáveis contribuem para o resultado do modelo? Como determinar as variáveis com maior poder discriminativo para classificação?	**Questionamentos:** Modelos de detecção de anomalias são capazes de discriminar o desempenho de pneus em função das variáveis do processo de uniformidade dos pneus? Quais variáveis contribuem para detecção de anomalias?
Metodologias: 1- Análise descritiva; informação mútua; análise de componentes principais. 2 - Synthetic Minority Oversampling Technique; BayesSearchCV, Leave-one-out. 3- Random Forest; Gradient Boosting Decision Tree; Light Gradient Boosting Machine; Logistic Regression; Support Vector Machine; K-Nearest Neighbor; Gaussian Processes Classification; Multi-layer Perceptron. 4 - SHapley Additive exPlanations	**Metodologias:** 1- Análise descritiva; teste Kruskal-Wallis. 2 - GASearchCV, Leave-one-out. 3- Random Forest; Light Gradient Boosting Machine; Logistic Regression; Support Vector Machine; Multi-layer Perceptron. 4 - Teste McNemar; SHapley Additive exPlanations; Local Interpretable Model-Agnostic Explanations.	**Metodologias:** 1- t-Distributed Stochastic Neighbor Embedding 2 - Isolation Forest; Local Outlier Factor; One-Class Support Vector Machine; Elliptic Envelop. 3- Teste McNemar; SHapley Additive exPlanations.

Fonte: Elaborada pelo autor.

ficação do modelo, foram consideradas as técnicas *SHapley Additive exPlanations* (SHAP) e *Local Interpretable Model-Agnostic Explanations* (LIME), para avaliação global e local. Posteriormente, o modelo paramétrico *Logistic Regression* foi considerado para encontrar valores de referência para pneus em conformidade, a proposta considera os valores dos parâmetros encontrados e um conjunto de restrições adotado no método de otimização linear para busca das melhores soluções. Esse estudo possibilitou encontrar valores de referência para variáveis que compõem o processo de uniformidade dos pneus; a partir desse cenário é possível desempenhar políticas de fabricação que antecedam o processo que gera o defeito do produto. Essa pesquisa foi submetida para *Computers & Industrial Engineering* em outubro de 2023.

O artigo 3 (Modelos de aprendizado de máquina para detecção de anomalias no processo de manufatura de pneus) aborda o problema sob outra ótica; consiste no treinamento de modelos não supervisionados para detecção de anomalias, com intuito de avaliar a concordância entre as categorias de pneus em conformidade e não conformes com amostras classificadas como *outliers*. Essa metodologia permite que fabricantes identifiquem possíveis interferências no processo de manufatura de pneus para fabricação de pneus com defeitos, e portanto, fabricantes podem anteceder o processo de validação de qualidade aplicado por montadoras automobilísticas. O estudo foi realizado com auxílio da linguagem de programação `Python`. O estudo compara diferentes metodologias, modelos baseados em *kernel*, árvores, distância e distribuição de probabilidade. Os modelos avaliados foram: *Isolation Forest*; *Local Outlier Factor*; *One-Class Support Vector Machine*; *Elliptic Envelop*. O teste de *McNemar* foi adotado para comparar os resultados dos modelos e o algoritmo *Isolation Forest* apresentou o resultado mais aderente ao problema. Com auxílio do algoritmo *t-Distributed Stochastic Neighbor Embedding*, o resultado do modelo foi representado em três dimensões em função dos pontos de amostras e a classificação do pneu. Para análise global e local do impacto das variáveis para detecção de anomalia foi utilizada a técnica *SHapley Additive exPlanations*. Esse trabalho foi aceito pelo congresso *ENEGEP 2023 (Encontro Nacional de Engenharia de Produção)* em junho de 2023.

1.6 Produção científica durante a Dissertação

Produção em congressos

1. **Oliveira, R. M. A.**; Figueiredo, C. R.; Silva, L. A.; Farias, P. C. M. A.; Sant'anna, A. M. O. (2023). Modelo híbrido de aprendizado de máquina para extração de características em imagens de radiografia de tórax. In: **XVI Brazilian Congress on Computational Intelligence**, Salvador. CBIC23. São Paulo: SBIC, 2023. v. 1. p. 1-8.
2. **Oliveira, R. M. A.** ; Sant'anna, A. M. O. (2023). Modelos de aprendizado de máquina para detecção de anomalias no processo de manufatura de pneus. In: **ENEGEP 2023 Encontro Nacional de Engenharia de Produção**, Fortaleza. XLIII ENEGEP. São Paulo: ABEPRO, 2023. v. 1. p. 1-14.

Produção em periódicos

1. Acosta, S. M.; **Oliveira, R. M. A.**; Sant'Anna, A. M. O. (2023). Machine learning algorithms applied to intelligent tyre manufacturing, **International Journal of Computer Integrated Manufacturing**, ISSN: 0951-192X, IF: 4.10, Qualis A2, publicado online em *https://doi.org/10.1080/0951192X.2023.2177734*
2. **Oliveira, R. M. A.** ; Sant'Anna, A. M. O. (2023). Machine learning approaches for multi-class classification in intelligent manufacturing processes, **Journal of Intelligent Manufacturing**, ISSN: 0956-5515, IF: 8.30, Qualis A1, submetido em Setembro de 2023, JIM-D-23-01612.
3. **Oliveira, R. M. A.** ; Sant'Anna, A. M. O. (2023). Explainable machine learning models for defects detection in industrial processes, **Computers & Industrial Engineering**, ISSN: 0360-8352, IF: 7.90, Qualis A1, submetido em Outubro de 2023, CAIE-D-23-04348.

1.7 Estrutura do documento

O trabalho está estruturado em cinco capítulos: o Capítulo 1, que corresponde à Introdução, apresenta a contextualização da pesquisa, justificativa, objetivos, método de

pesquisa adotado, delimitações da pesquisa, relação entre os artigos propostos e produção científica. No Capítulo 2 é apresentado o artigo 1 (*Machine learning approaches for multi-class classification in intelligent manufacturing processes*). No Capítulo 3 é apresentado o artigo 2 (*Explainable machine learning models for defects detection in industrial processes*) e no Capítulo 4 o artigo 3 (Modelos de aprendizado de máquina para detecção de anomalias no processo de manufatura de pneus). No Capítulo 5 é apresentada a conclusão, considerações finais e sugestões para trabalhos futuros. Os anexos contêm os demais artigos desenvolvidos durante o período da dissertação.

2 Artigo 1 - Machine learning approaches for multi-class classification in intelligent manufacturing processes

Machine learning approaches for multi-class classification in intelligent manufacturing processes

Rodrigo Marcel Araujo Oliveira[1] and Ângelo Márcio Oliveira Sant'Anna[1]

[1]Polytechnic School, Federal University of Bahia, Salvador, Bahia, Brazil.

Contributing authors: rodrigomarcel@ufba.br; angelo.santanna@ufba.br;

Abstract

Machine learning models are increasingly used for recognizing non-linear patterns in manufacturing processes, optimizing manufacturing resources, and improving industrial process efficiency. This paper proposes developing machine learning algorithms for multi-class classification of tire uniformity processes in the intelligent manufacturing industry. The research evaluated the performance of eight machine learning algorithms: Random Forest, Support Vector Machine, K-Nearest Neighbor, Gaussian Processes, Logistic Regression, Multi-layer Perceptron, Gradient Boosting, and Light Gradient Boosting Machine. The estimator parameters were found through cross-validation using Bayesian optimization. The Synthetic Minority Oversampling Technique technique was used to address the issue of imbalanced classes. Random Forest and Gradient Boosting Decision Tree models demonstrated the highest predictive power and better performance than the other algorithms. The SHapley Additive exPlanations methodology was employed to interpret the model results, allowing the identification of predictor variables with the greatest impact on tire performance. These findings are crucial for guiding and proposing new tire manufacturing policies applied in the quality control process.

Keywords: Machine Learning, Bayesian Optimization, Intelligent manufacturing, SHAP, Tire Manufacturing

1 Introduction

The intelligence embedded in machines represents the current stage of industrial evolution, referred to as Industry 4.0. This evolution is characterized by the integration of various technologies such as Artificial Intelligence (AI), robotics, Internet of Things (IoT), and cloud computing, encompassing connectivity and data processing (Lu et al, 2020). The implementation of extensive networks connecting production units enables real-time information flow, generating data on process monitoring, maintenance, defect detection, etc. This ensures integrity across organizations within each production chain, fostering an environment of innovation and competitiveness and playing a fundamental role in decision-making processes (Usuga Cadavid et al, 2020).

In the context of Industry 4.0, the manufacturing process demands high complexity in problem-solving (Greis et al, 2023). Challenges are rooted in multi-criteria decisions, recognition of non-linear patterns in production systems, predictions, and process optimizations (Usuga Cadavid et al, 2020). AI has significantly influenced various sectors of the economy, wherein robotics, natural language processing, time series analysis, and image processing are some of the studied fields. Machine learning, a subset of AI, as stated by Tercan and Meisen (2022), enables enhanced efficiency in manufacturing processes. These techniques aid in cost reduction, optimization of operational resources, and improved production indicators, among other benefits (Rosati et al, 2023).

This paper proposes developing machine learning algorithms for multi-class classification of tire uniformity processes in the intelligent manufacturing industry. The main contribution of this research is to discover non-linear patterns for fault detection in tire manufacturing systems using machine learning models, within the context of a multinational tire industry. Data analysis and tire performance classification through machine learning techniques highlight pertinent issues in the research literature. The research encompasses addressing sampling techniques for imbalanced data, pinpointing classifiers that offer high accuracy to be used in selecting high-quality tires, and identifying predictor variables with significant impact on tire performance classification. In Section 2, it provides a comprehensive review of relevant research. Section 3 presents the primary theoretical classification methods used in the study, parameter optimization techniques, and methodologies for model evaluation. This section also discusses some sampling methods and SHapley Additive exPlanations (SHAP) analysis to assess variable impact on the model. In Section 4 we presents the proposed methodology to achieve the objective of this work. Section 5 describes the study dataset information. In Section 6, we present analyze the data using descriptive statistics, comparative study results and examine classification accuracy in all cases. In Section 6.3, aided by the SHAP technique, we present variables with the most impact on the response variable. Section 7 outlines our conclusions and future research directions.

2 Literature review

The classification of tire performance is an integral part of strategic planning in the tire manufacturing process. In this context, defect detection models for quality assurance of these products are crucial. Recently, machine learning models have gained prominence in academia for investigating non-linear patterns for automatic defect detection, a problem that has become essential in the industry (Wang et al, 2021).

Pattern recognition of non-linear patterns using different machine learning algorithms involving a mix of numerical and categorical variables, i.e., mixed-type variables, for tire performance prediction, is discussed in Gutierrez-Gomez et al (2020). In this work, several methods were employed, including linear methods such as linear regression; kernel-based techniques like Support Vector Regression (SVR) with linear and Gaussian kernels; non-parametric methods like K-Nearest Neighbor; gradient boosting techniques, and Random Forest. The interest lies in evaluating different encoding methods for handling categorical variables combined with the aforementioned models. Friedman's statistics (Demšar, 2006) and Nemenyi post-hoc tests were used to test the significance of performance differences between techniques and encoding strategies. The article provides recommendations for efficiently handling mixed-type variables while achieving high performance in regression tasks across various datasets.

Penumuru et al (2020) used the Random Forest, Support Vector Machine, k-Nearest Neighbor and Logistic Regression models for the classification of surfaces of metallic materials such as aluminum, copper, and steel. The models were trained based on image features, using RGB colors. The authors report that the proposed methodology can contribute to the handling of materials performed by devices or robots in the industry 4.0. In Acosta et al (2021), applications involving machine learning models such as Random Forest, K-Nearest Neighbor, Relevance Vector Machines (RVM), and Artificial Neural Network (ANN) demonstrate their suitability for solving non-linear problems. The article demonstrated that these algorithms have predictive modeling power to predict phosphorus concentration levels in a steel plant. Zhou et al (2022) proposed different machine learning model pipelines for quality monitoring in resistance spot welding.

Applications of convolutional neural networks were used in Rajeswari et al (2022) to discover tire imperfections through the examination of tire tread design images. Vehicle mileage depends on different components and parameters, such as road condition, tire weight, etc. The article focuses on examining tire tread design for deformities. The proposed system uses a dataset from different tire manufacturers. The experiment results show that it is conceivable to predict tire deformities and durability based on tire tread designs through a convolutional neural network based on weighted quality. Bustillo et al (2022) proposed a machining process modeling using machine learning models. The authors employed various algorithms such as ANN and Random Forest for predicting the tool life in the front turning operation of steel discs. Support

Vector Machine algorithm yielded the best results for in-situ identification of material batches (Lutz et al, 2021).

The SVR and ANN algorithms were used to predict triaxial tire-road contact stress in Li et al (2021), a significant factor in vehicle and road surface material performance evaluation. The article demonstrated the possibility of improving prediction accuracy in all directions, predicting step-by-step uniaxial and triaxial stresses when compared to other multiple-output regression models. This study provides a reference for tire-road contact stress measurement design and statistical scheme. The Random Forest model was employed in Acosta et al (2023) for anomaly detection and predicting the weight of tires in a factory. In this study, they utilized a differential evolution algorithm to optimize machine learning model parameters, leading to superior results compared to other methods.

The ANN were used by Lee et al (2021) to predict tire pattern-associated noise, a fundamental process in the early stage of tire pattern design. The trained convolutional neural network (CNN) can be used for pattern noise prediction for a tire to be designed in the early design stage using a single tire drawing image, while ANN can be used for pattern noise prediction for a real tire in the development stage. Zhou et al (2022) introduced strategic approaches for statistical arbitrage in stock markets, employing various machine learning models based on random forests and neural networks to predict stock prices.

Zhu et al (2021) presented a tire lifespan prediction method based on image processing and machine learning. A new method for extracting and expressing tire image features using a gray gradient co-occurrence matrix (GGCM) and a Gauss-Markov random field (GMRF) was proposed. The extracted features were classified using K-Nearest Neighbor. The experiment indicates the effectiveness and accuracy of the proposed method for tire lifespan prediction. The authors expect the obtained results to be used for real-time prediction of tire lifespan, thus reducing tire-related traffic accidents.

Tire force estimation is feasible with neural network techniques according to Xu et al (2021). The study introduces a smart tire system with a triaxial acceleration sensor installed in the inner lining of the tire. Neural network techniques are used for real-time processing of sensor data. The result shows that the developed smart tire, combined with machine learning models, is effective in accurately predicting tire forces under different driving conditions.

The interpretability of machine learning models using the SHAP approach was proposed by Jabeur et al (2021). In this work, the authors discuss strategies for gold price prediction. The eXtreme Gradient Boosting (XGBoost) model yielded the best results. SHAP values were used to assess the impact of each predictor variable on gold price prediction.

In the literature, there are no studies discussing tire performance with predictor variables from the tire uniformity processes. This paper aims to contribute to academia and industry by proposing machine learning algorithms that overcome current challenges. The developed machine learning

models for tire quality performance were: K-Nearest Neighbor (KNN), Logistic Regression (LR), Gaussian Processes Classification (GPC), Multi-layer Perceptron (MLP), Support Vector Machine (SVM), Random Forest (RF), Gradient Boosting Decision Tree (GBDT), and Light Gradient Boosting Machine (LGBM). The study's differentiator is proposing a classification model with a Bayesian approach for parameter optimization and using the SHAP technique to assess the impact of predictor variables on tire performance.

3 Background

In this section, concepts related to the theoretical frameworks of the techniques used in the development of this article will be discussed.

3.1 Machine learning models

3.1.1 Random Forest

The RF algorithm was proposed by Breiman (2001) as a combination of simple decision trees. The construction of a classification tree involves recursive binary splitting aimed at predicting or classifying a response variable Y from a set of variables $X_1, ..., X_r$. The algorithm essentially involves determining the regions in which the predictor variable space is partitioned. The Classification And Regression Trees (CART) methodology considers a partition of the predictor variable space into K regions, $R_1, ..., R_k$. For each observation in set R_j, the categories of Y in the discrete case will be the mode or the percentage of Y values corresponding to $X_1, ..., X_r$ in R_j (Breiman, 1996). The regions are constructed to minimize some classification errors. The Gini and entropy metrics, as per equations (1) and (2), are used as criteria for choosing the variable that forms the root node of each tree, where $\hat{p}_i$ represents the probability of class i, and k is the number of classes. Entropy is a measure of information that indicates the system's disorder, and Gini is a measure of impurity; both have minimum values equal to zero, indicating that the tree node is pure.

$$Gini = \sum_{i=1}^{k} \hat{p}_i(1 - \hat{p}_i). \tag{1}$$

$$Entropy = \sum_{i=1}^{k} -\hat{p}_i log(\hat{p}_i). \tag{2}$$

The ensemble trees are constructed through a bootstrap sampling process (Scornet et al, 2015). Random Forests adopt the bagging technique proposed by Breiman (1996) with the goal of reducing variance by predicting multiple separate models using different subsets of the training data. For classification problems, the generalization error depends on the weighting assigned to individual trees and the output correlation among these trees. In the process of tree voting, the majority class prevails among the predicted classes. The model's

performance is associated with the number and depth of trees in the forest, etc (Probst and Boulesteix, 2018).

3.1.2 Gradient Boosting Decision Tree

The GBDT method proposed by Friedman (2001) is based on the idea that combining classifiers with high error rates, such as shallow decision trees, can create an ensemble that significantly outperforms any of its individual components.

The training process for the full ensemble begins with fitting the first tree. The second tree is then fitted to an adapted version of this dataset, where observations from the first tree that were misclassified are given higher weights than those it classified correctly, enabling the second tree to perform better on the cases the first tree struggled with. This continues for all trees in the ensemble, and at the end of the training process, those trees that performed the best overall will have higher weights in the voting phase (Friedman, 2002). Hastie et al (2009) asserted that, the GBDT tree can be represented by equation (3):

$$T(x, \beta) = \sum_{k=i}^{J} I(x \in R_i)\phi_i(x). \tag{3}$$

where $\beta = \phi_i, R_i$ represents the parameters, J is number of nodes in the tree, and the function ϕ_i is characterized by parameterizing the split points of nodes and tree variables. GBDT is the sum of $T(x, \beta)$ as shown in equation (4), M is number of tree:

$$g_M = \sum_{k=i}^{M} T(x, \beta_i). \tag{4}$$

The value of ϕ_i given R_i is based on estimating the mean of $y_i \in R_i$, whereas for classification loss, the estimate is given by the modal class. To find R_j, the models are adjusted by minimizing a loss function over the training data. For each iteration, (5) is computed (Hastie et al, 2009), L is the loss function, N number of observations, given the current model $g_{m-1}(x_i)$, that is, the sum of the trees.

$$\arg \min_{(\beta_k, \phi_k)_k^N} \sum_{k=i}^{N} L(y_i, g_{m-1}(x_i) + T(x_i, \beta_i)) \tag{5}$$

The iterative optimization of (5) is based on the Gradient Descent technique (Friedman et al, 2000), The process involves computing (6):

$$\psi_{im} = \left[\frac{\partial L(y_i, g(x_i))}{\partial g(x_i)}\right]_{g(x_i)=g_{m-1}(x_i)}. \tag{6}$$

Hastie et al (2009) defines the commonly used loss function is the multinomial deviance for classification problems.

3.1.3 Light Gradient Boosting Machine

The LGBM has applications in various industry segments (Mishra and Paliwal, 2022). Considered a more sophisticated version of GBDT, the algorithm is based on decision trees that grow leaves vertically, as illustrated in Fig. 1. This contributes to greater efficiency and scalability of the model (Fan et al, 2019). The model proposed by Ke et al (2017) utilizes two new methodologies, Gradient-based One Side Sampling (GOSS), and Exclusive Feature Bundling (EFB). EFB effectively reduces the number of features without compromising the accuracy of the split point. GOSS is responsible for selecting data instances with higher gradients to estimate information gain and maintain accuracy. Typically, for Boosting models, information gain is measured by the variation after the split. Ke et al (2017) cited that information gain can be expressed by equation (7), where $x_1, .., x_n$ represent instances from the training set, and for each iteration, the negative gradients of the loss function are denoted as $g_1, ..., g_n$.

$$\hat{V}_j(d) = \frac{1}{n}\left(\frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i)^2}{n_l^j} + \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i)^2}{n_r^j}\right), \tag{7}$$

where $A_l = x_i \in A : x_{ij} \leq d$, $A_r = x_i \in A : x_{ij} \geq d$, $B_l = x_i \in B : x_{ij} \leq d$, $B_l = x_i \in B : x_{ij} \geq d$, and the normalization coefficient $\frac{(1-a)}{b}$.

Training instances are classified based on the absolute values of their gradients in descending order, i.e., the top $a \times 100\% \in A$. For the complement set A^c, a random subset B of size $b \times A^c$ is sampled. The instances are split according to the estimated variance gain in vector $\hat{V}_j(d)$ over the subset $A \cup B$.

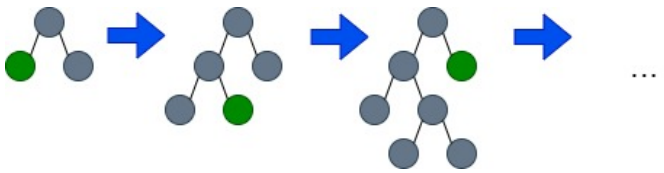


Fig. 1 Leaf wise tree growth

3.1.4 Logistic Regression

The LR, by providing probabilities, is commonly used in classification to predict the probability of any observation belonging to a certain class. LR models are fitted by maximum likelihood (Hastie et al, 2009). In the binary case, that is, $\boldsymbol{Y}_i = 0$ or $\boldsymbol{Y}_i = 1$, we model the probability of $\boldsymbol{Y}_i = 1$ with the sigmoid function:

$$l(\boldsymbol{\beta}) = \mathbb{P}(\boldsymbol{Y}_i = 1 \mid \boldsymbol{X} = \boldsymbol{x}_i) = \frac{e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}. \tag{8}$$

The vector $\boldsymbol{x}_i$ corresponds to the features of the observations, and $\boldsymbol{\beta}$ are the parameters of LR. The estimation of these values is based on the maximum likelihood method. The log-likelihood function can be written as follows:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{N} \boldsymbol{y}_i(\boldsymbol{\beta}^T \boldsymbol{x}_i) - \sum_{i=1}^{N} log(e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}). \tag{9}$$

The maximum likelihood estimators for $\boldsymbol{\beta}$ are those that maximize $l(\boldsymbol{\beta})$. Numerical optimization procedures, such as Newton-Raphson, estimate the values of $\boldsymbol{\beta}$ (Friedman et al, 2000). A term that helps reduce the variance of estimates and effects of collinearity added as regularization to the likelihood function is the Lasso penalty (Hastie et al, 2009), i.e., L1 regularization, according to equation (10).

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{N} \boldsymbol{y}_i(\boldsymbol{\beta}^T \boldsymbol{x}_i) - \sum_{i=1}^{N} log(e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}) + \lambda \sum_{j=1}^{N} |\beta_j|. \tag{10}$$

where λ is a scalar between zero and infinity, as this value increases, it amplifies bias and reduces the variance of the model. When dealing with highly correlated features, Lasso regularization acts as a variable selector. It chooses only one of the features and sets the coefficients of the others to zero.

3.1.5 Support Vector Machine

The SVM is a powerful and flexible class of algorithms commonly used for classification and regression, introduced by Cortes et al (1995). For the classification problem, the algorithm seeks the best-separating hyperplane between classes, maximizing the margin between the closest points of the two classes. These points on the boundaries of these classes are called support vectors. Points from one class on the other side of the separating hyperplane are weighted with low importance to reduce their influence. The SVM function is written according to equation (11).

$$f(\boldsymbol{X}) = \alpha + \beta^T \boldsymbol{X}, \tag{11}$$

where $f(\boldsymbol{X})$ denotes the predicted values, $\beta = (\beta_1, ..., \beta_p)^T$ corresponds to the coefficients, $\boldsymbol{X}$ is the matrix with p observations of variables $\boldsymbol{X}_1, ..., \boldsymbol{X}_p$. For each entry $\boldsymbol{X}_i$, that is, the i-th column of $\boldsymbol{X}$, it is associated with attributes $y_1, ..., y_p \in \{-1, 1\}$. The classification of X_i is determined by the sign of $f(\boldsymbol{X}_i)$, if the sign is positive, X_i is classified as $y = 1$, if negative, as $y = -1$, and thus $\forall i$ the relationship (12) is satisfied (Hastie et al, 2009).

$$y_i(\alpha + \beta^T \boldsymbol{X}_i) \geq 0. \tag{12}$$

The margin is the smallest distance from the hyperplane to the point x_i. The maximum margin is obtained by solving equation (13) using Lagrange multipliers.

$$\arg\max_{\alpha,\beta}\{\frac{1}{\|\beta\|}\min[y_i(\alpha+\beta^T\boldsymbol{X}_i)]\}. \tag{13}$$

In the context of soft margin, the objective is to minimize expression (14). Misclassified instances incur a penalty ϵ, which increases with the distance from the hyperplane (Kecman, 2005)

$$\min\frac{1}{2}\|\beta\|^2+C\sum_{j=1}^{n}\epsilon_j, \tag{14}$$

subject

$$y_i(\alpha+\beta^T\boldsymbol{X}_i)\geq 1-\epsilon_i, \tag{15}$$

for $\epsilon_i > 0\ \forall i$ where the parameter $C > 0$ controls the trade-off between the penalty of the slack variable and the size of the classification margin.

For non-linear problems, a kernel trick is used to map the input data to a higher-dimensional space such that in this space, the separating hyperplanes are constructed. The most commonly used kernels in practice are linear, polynomial, radial, and hyperbolic tangents (Hastie et al, 2009). Equation (16) denotes the SVM function associated with a kernel, where γ_i are functions of α and β, and Ω is the set of support vectors.

$$f(\boldsymbol{X})=\alpha+\sum_{j\in\Omega}\gamma_i K(\boldsymbol{x},\boldsymbol{x}_i)+\delta \tag{16}$$

3.1.6 K-Nearest Neighbor

The KNN is a non-parametric technique; the algorithm makes predictions based on the distance from the K nearest neighbors (James et al, 2013). The value assigned to a novel point is calculated from a simple majority vote of the nearest neighbors; this is a basic procedure that involves using uniform weights. However, it is possible to weigh the neighbors so that the closest points have a greater influence on the adjustment. The distance employed will depend on the nature of the problem; the most commonly used ones are the standard Euclidean (17), Mahalanobis, and Minkowski distances.

$$d_i=\|x_i-x^*\| \tag{17}$$

The algorithm involves setting a value for K, and for a novel point x^*, the probability that x^* belongs to class c is described by (18).

$$\mathbb{P}(Y=c\mid X=x^*)=\frac{1}{K}\sum_{i\in\Omega}I(y_i=c), \tag{18}$$

where Ω is the set of values for which the K training points are closest to x^*, based on a distance measure. The value of K can be obtained through cross-validation.

3.1.7 Gaussian Processes Classification

The Gaussian Process classification is a technique that involves estimating a function for arbitrary inputs, given a prior set of training data (Mackay, 1996). The Gaussian Process (GP) is composed of a family of random variables with a joint Gaussian distribution. According to Rasmussen (2004), this process is characterized by the expected value (19) such that $\mathcal{X} \to \mathbb{R}$, and its covariance function (20) with $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and is described by (21).

$$m(x) = \mathbb{E}[g(x)], \tag{19}$$

$$k(x, \acute{x}) = \mathbb{E}[(g(x) - m(x))(g(\acute{x}) - m(\acute{x}))], \tag{20}$$

$$g \sim \mathcal{G}(m(x), k(x, \acute{x})). \tag{21}$$

The function k is positive semidefinite symmetric, considering $X = (x_i)_{i=1}^m$ and $Y = (y_i)_{i=1}^n$ elements of $\mathcal{X}$, the covariance function can be written as (22).

$$K(X, Y) = \begin{bmatrix} k(x_1, y_1) & \cdots & k(x_1, y_n) \\ \vdots & \ddots & \vdots \\ k(x_m, y_1) & \cdots & k(x_m, y_n) \end{bmatrix}. \tag{22}$$

In GP regression, the relationship between input data X (independent variables) and output Y (dependent variables) can be written as (23), with $\epsilon_x \sim \mathcal{N}(0, \Sigma)$, and Σ being a positive semidefinite matrix.

$$Y = g(X) + \epsilon_x. \tag{23}$$

The estimation of the function in the Bayesian context for the relationship between Y and X can be described by (24).

$$Y \mid X \sim \mathcal{N}(m(X), K(X, X) + \Sigma). \tag{24}$$

In GPC, we are interested in estimating the probability of a point belonging to one of the $\mathcal{C}$ classes. For this, the problem can be characterized by a function $\phi_i(x^*) = \mathcal{P}(y^* = c \mid x^*)$, with x^* and y^* being points from Y and X. According to Rasmussen and Williams (2005), the posterior distribution of $\boldsymbol{g^*} \mid X, Y$ is given by (25), with $g(x^*) = \boldsymbol{g^*}$.

$$p(\boldsymbol{g^*} \mid X, Y, x^*) = \int p(\boldsymbol{g^*} \mid X, x^*, g(X)) p(g(X) \mid X, Y) dg(X). \tag{25}$$

The problem involves estimating $\phi_i(x)$ described by (26) using (25), where $p(k \mid \boldsymbol{g^*})$ is the probability of y^* given $\boldsymbol{g^*}$ for class c.

$$\phi_i(x^*) = \int p(k \mid \boldsymbol{g^*}) p(\boldsymbol{g^*} \mid X, Y, x^*) d\boldsymbol{g^*}. \tag{26}$$

3.1.8 Multi-layer Perceptron

The MLP is a neural network structure composed of an input layer, one or more hidden layers, and an output layer. The input layer receives input values, the hidden layers process the data, and the output layer has one unit for each classification feature. For each neuron in the hidden layer, the values from the previous layer are transformed using a weighted linear sum followed by some activation function $\phi(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$ that is differentiable, allowing the network to perform a mapping between an input space and an output space. Common activation functions in the classification context include sigmoid, ReLU, hyperbolic tangent (28), and softmax. The Fig. 2 shows the architecture of an MLP with two hidden layers. Haykin (2008) asserted that, the approximation function for a single-hidden-layer MLP can be written as (27), with m_0 input nodes, one hidden layer with m_1 neurons, w_{m_0} synaptic weights for neuron i, bias b_i, and ϕ_i synaptic weights for the output layer.

$$F(\boldsymbol{x}) = \sum_{j=1}^{m_1} \alpha_i \phi(\sum_{j=1}^{m_0} w_{ij} x_i + b_i), \tag{27}$$

$$\phi(z_i) = \frac{e^{z_i} - e^{-z_i}}{e^{z_i} + e^{-z_i}}. \tag{28}$$

The loss function for classification is the mean cross-entropy, with $y \in \{0, 1\}$, described by (29).

$$L(y, \hat{y}, \boldsymbol{W}) = \frac{1}{n} \sum_{k=0}^{n} y_i log(\hat{y_i}) + (1 - y_i) log(1 - \hat{y_i}) + \frac{\alpha}{2n} \|W\|_2^2. \tag{29}$$

The loss function is minimized by updating the weights using some optimization algorithm, with the most commonly used ones being Stochastic Gradient Descent (SGD) (Amari, 1993), Adam (Kingma and Lei Ba, 2015), and Limited-memory BFGS (Mokhtari and Ribeiro, 2015). In the context of the SGD algorithm, the weights are updated according to the gradient of the loss function given by equation (30).

$$W^{i+1} = W^i - \gamma \left(\frac{\partial R(w)}{\partial w} + \frac{\partial L}{\partial w} \right), \tag{30}$$

where γ is the learning rate, and $R(w)$ is a function adapted from the weights.

3.2 Synthetic Minority Oversampling Technique

Imbalanced classification can impact the performance of models in finding decision boundaries. The Synthetic Minority Oversampling Technique (SMOTE) technique is an oversampling approach that allows the generating of synthetic samples for minority categories (Chawla et al, 2011). The idea is based on the KNN algorithm, involving selecting nearby instances in the feature space. That is, a random instance from the minority class is chosen, and then K nearest

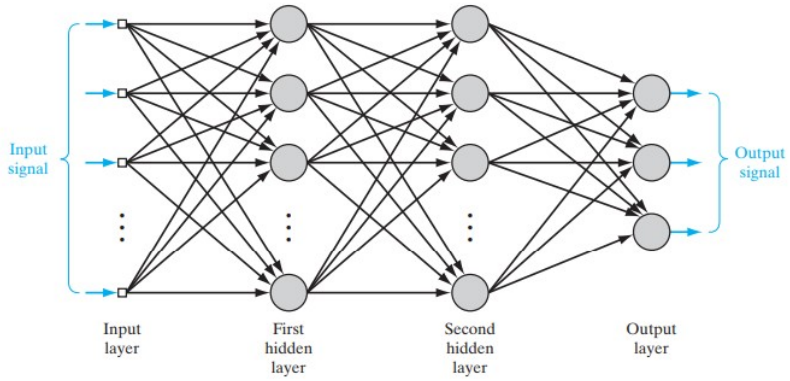


Fig. 2 Architecture of the MLP with one input layer featuring multiple units, two hidden layers, and one output layer with 3 units, source: Haykin (2008).

neighbors for this instance are found. For each randomly selected neighbor, a synthetic instance is created randomly between the two instances in the feature space. This approach is effective, enabling algorithms to learn broader regions to improve the classifier's generalization (Penumuru et al, 2020).

3.3 Hyperparameter tuning

The BayesSearchCV method is an alternative for estimating parameters that maximize certain model evaluation metrics (Young et al, 2019). It is also an optimized and efficient approach compared to methods like GridSearchCV, which perform an exhaustive search with various combinations of specified parameter values (Ahmad et al, 2022).

BayesSearchCV is based on Bayesian optimization framework that aims to minimize a black-box function globally. The technique is characterized by a probabilistic model with a prior distribution that minimizes an unknown objective function and an optimized acquisition function. The optimization algorithm aims to minimize the objective function as shown in equation (31), with $g : X \to \mathbb{R}$, where g is a black-box function with no known closed-form.

$$x^* = \arg\min_x g(x). \tag{31}$$

The probabilistic model is referred to as a surrogate model. It fits all the points of the objective function, and for each evaluation of the objective function, it is updated through the Bayesian process (Young et al, 2019). The BayesSearchCV method can incorporate different surrogate models such as GP and RF. Additionally, other acquisition functions can be defined, such as expected improvement, lower confidence bound, and probability of improvement (Masum et al, 2022). This paper applied GP as the surrogate model and used expected improvement as the acquisition function, as shown in equation (32), where x_t^+ is the observed optimum point:

$$-I(x) = E[g(x) - g(x_t^+)]. \tag{32}$$

3.4 SHapley Additive exPlanations

The interpretation of machine learning algorithms results from challenges to the academic community, and the SHAP method is based on cooperative game theory (Lundberg and Lee, 2017). The idea is to compute the contributions of features to individual predictions of complex models. The Shapley value represents a way to distribute the total gains among players; in other words, the value of a feature is determined by its contribution to the payout, as shown in equation (33).

$$\hat{\phi}_i = \sum_{S \subseteq \{1,\ldots,N\} \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M} |g_x(S \cup \{i\}) - g_x(S)|. \tag{33}$$

where S is a subset of features used in the model, x is the vector of feature values to be explained, M is its cardinality, and finally, the function g_x corresponds to the predicted feature values in the subset S.

The Shapley value can be estimated using Monte Carlo approximation (Štrumbelj and Kononenko, 2014). The equation (34) describes the function $\hat{f}(x^i_{+j})$ represents the prediction for x with a random subset of feature values. The value x^i is drawn from a random sample.

$$\hat{\phi}_j = \frac{1}{N} \sum_{i=1}^{N} (\hat{f}(x^i_{+j}) - \hat{f}(x^i_{-j})). \tag{34}$$

The SHAP approach proposed by Lundberg and Lee (2017) allows interpreting machine learning model predictions by calculating the impact of each feature on the prediction for a specific input. SHAP represents the Shapley value as an additive feature attribution method, a linear model of binary variables, as shown in equation (35).

$$g_j(z^*) = \phi_0 + \sum_{i=j}^{N} \phi_j z^*_j. \tag{35}$$

with g being the explanation model, z^* is the coalition vector, N is the maximum coalition size, and ϕ_j is the Shapley value assignment for feature j.

The Shapley value satisfies the properties of efficiency, symmetry, dummy, and additivity (Santos et al, 2023). The feature importance in SHAP considers the absolute values of the Shapley values, as shown in equation (36).

$$I_j = \frac{1}{n} \sum_{i=1}^{n} |\phi^i_j|. \tag{36}$$

3.5 Performance evaluation

Cross-validation is a commonly used statistical method for model evaluation in the machine learning context (Wong, 2015). The goal of this method is to assess the model's generalization ability, that is, to estimate parameters with data

from one sample and predict data in other sample sets (Wu et al, 2010). Some frequently employed methodologies include Hold-out, K-fold, and Leave-one-out (Arlot and Celisse, 2010). Hold-out involves splitting the data into training and validation sets based on a predefined proportion of the data (Raschka, 2018). In K-fold, the dataset is divided into K parts, and in each iteration, one of these parts is considered the test set, and the others serve as the training set (Meijer and Goeman, 2013). Parameter estimates and performance metrics are collected for each iteration, and their means are calculated at the end of the process, serving as final estimates and metrics. Leave-one-out is known for not wasting data, as only one sample is removed from the training set, allowing for different training and test sets (Shao and Er, 2016).

To evaluate the performance of the fitted models and subsequently select the best ones, we will primarily rely on metrics derived from the confusion matrix. The confusion matrix has two dimensions, one indexed by the actual class and the other by the classifier's predicted classes (Deng et al, 2016). Fig. 3 illustrates the structure of a confusion matrix in the context of multiclass classification, with classes denoted as C_1 to C_n. The matrix elements, N_{ij}, represent the number of samples belonging to class C_i but were classified as C_j.

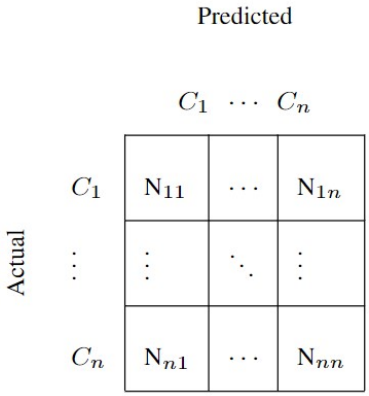


Fig. 3 Confusion matrix for multiclass classification

In the above matrix, we can define some metrics commonly used to evaluate classification models. Accuracy is a measure that corresponds to the proportion of correct predictions the model makes. A higher accuracy value indicates a higher correctness rate of the model, as shown in equation (37). Precision, defined in equation (38), represents the number of true positives divided by the sum of true positives and false positives. Recall, as shown in equation (39), is defined as the number of true positives divided by the sum of true positives and false negatives. The F_1-score, in equation (40), is the harmonic mean of precision and recall.

$$AC = \frac{\sum_{i=1}^{N} N_{ii}}{\sum_{i=1}^{N}\sum_{j=1}^{N} N_{ij}}, \ for\ i = 1, ..., N \tag{37}$$

$$PREC_i = \frac{N_{ii}}{\sum_{j=1}^{N} N_{ji}}, \ for\ i = 1, ..., N \tag{38}$$

$$REC_i = \frac{N_{ii}}{\sum_{j=1}^{N} N_{ij}}, \ for\ i = 1, ..., N \tag{39}$$

$$F_1 score_i = \frac{2 \cdot PREC_i \cdot REC_i}{PREC_i + REC_i}, \ for\ i = 1, ..., N \tag{40}$$

4 Proposed methodology

This section presents the proposed methodology to achieve the objective of this work. The machine learning models developed were RF, GBDT, LGBM, LR, GPC, MLP, KNN, and SVM. The procedure conducted in this study is summarized in Fig. 4 and consists of the following steps: data preparation, exploratory data analysis (EDA), data splitting for development and validation, descriptive analysis, variable selection, data normalization, training simulations, performance analysis with parameter tuning for each model through cross-validation, evaluation of results on the validation dataset, and interpretation analysis of the models using the SHAP method. To assess the performance and generalization ability of the developed models, several metrics based on the confusion matrix were used. For classification model evaluation, commonly known metrics in academia were used, namely accuracy, precision, recall, and F-score.

Data normalization using the Min-Max procedure was adopted due to the differences in the ranges of predictor variables, as per equation (41), where the method scales the variables within the range [0, 1]. This methodology was also used to construct the Principal Component Analysis (PCA) plot in Section 3.1.

$$Z = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{41}$$

The study was conducted using the Python programming language version 3.9, and free software tools such as Jupyter Notebook and Visual Studio Code were utilized. All models and methods were coded using a machine with an Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz processor, 8 GB of RAM, and an NVIDIA GEFORCE 2 GB graphics card. The scikit-learn library was used for developing the machine learning models, and for hyperparameter optimization and result interpretation, scikit-optimize and shap were respectively adopted.

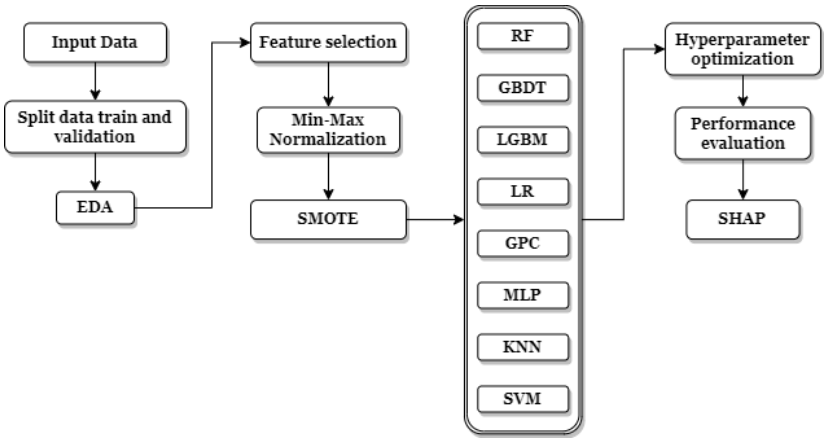


Fig. 4 Flowchart of the proposed methodology

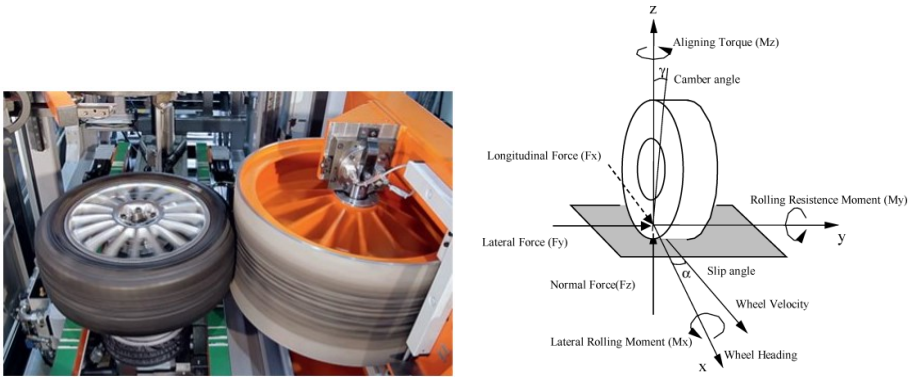
5 Real case study

The present case study was conducted in a multinational tire industry that has been in the market since the 19th century and is in several countries. This factory produces tires for passenger cars. Tire performance is a crucial indicator for the factory as it directly impacts sales. The collected data are sourced from the factory's intelligent manufacturing system.

The response variable in this study is tire performance, inspected for irregular size, mass, and force under specific pressure, load, and speed conditions (Cho et al, 2012). This set of processes is known as uniformity testing and adheres to the standards accepted by global tire manufacturers (Weyssenhoff et al, 2019). It aims to identify tires with subpar performance to prevent them from being sold. The study includes the ten variables described in Table 1. The database comprises a sample of 107 observations. Tire uniformity measurement involves rotating the tire while applying a load on a rotating drum in contact with the tire (Liu et al, 2023), along with force sensors mounted beside the tire's axis (Du et al, 2020). Fig. 5(a) displays the equipment with sensors measuring tire characteristics, and in Fig. 5(b), the forces acting on the tire are schematically represented, source: Gonçalves and Ambrósio (2005).

6 Results and Discussion

The input data was stratified and split into two subsets using the Hold-out technique based on the response variable. One sample was reserved for development and another for validation with a proportion of 75%, meaning 80 samples, and 25%, i.e., 27 samples for validation. For the development sample, the distribution of the number of tires for each performance category is: 34 (A); 35 (B); 11 (C). As for the validation dataset, the distribution is: 12 (A); 11 (B); 4 (C). Category A has the best performance.



(a) (b)

Fig. 5 Equipment with sensors (a) and Forces acting on the tire (b) to tire uniformity process.

Table 1 Description of variables from tire uniformity process

Variables	Description
GR	Tire performance [1]
RFV	Radial Force Variation
LFV	Lateral Force Variation
RRO	Radial Run-Out
PLY	Angular Effect
H1RFV	Radial Force 1° Harmonic
H2RFV	Radial Force 2° Harmonic
H1LFV	Lateral Force 1° Harmonic
H1RRO	Radial Run-Out 1° Harmonic
CAPSPLICE	Tread Splice
CONICITY	Conicity Effect

[1]Tire performance with categories: A; B; C.

6.1 Exploratory Data Analysis

Table 2 contains the main descriptive statistics of the predictor variables, and it is noteworthy that the means are close to the medians. The Fig. 7 corresponds to boxplot charts for each predictor variable based on tire performance. Additionally, it can be observed that the numerical magnitudes of the predictor variables differ. In Fig. 6, the Pearson correlation coefficient between the predictor variables is represented. It is expected that radial and lateral forces are correlated with their harmonic components; however, it was identified that the variables RFV and RRO are positively correlated.

Principal Component Analysis (PCA) is a multivariate analysis technique that can identify relationships between variables and explain these characteristics using components from the dimensionality reduction process (James et al, 2013). The goal of PCA is to transform the original data into another set of dimensions, equal to or fewer in number, known as principal components.

Table 2 Statistical measures of the variables from tire uniformity process

	Descriptive Statistics						
Variables	mean	std	min	25%	50%	75%	max
RFV	57.33	16.94	32.00	44.75	55.00	66.25	117.00
H1RFV	32.86	15.91	9.00	21.00	30.00	44.25	81.00
H2RFV	18.25	9.06	2.00	11.75	18.00	23.00	46.00
CONICITY	8.96	1.97	4.50	7.45	8.95	10.42	13.00
H1LFV	8.29	3.40	2.20	6.13	7.75	9.90	18.10
PLY	50.50	2.81	40.40	49.30	50.55	52.23	56.80
LFV	12.76	3.32	7.00	10.30	12.40	14.80	21.90
RRO	0.84	0.24	0.42	0.65	0.82	1.00	1.49
H1RRO	0.42	0.22	0.01	0.23	0.40	0.55	1.07
CAPSPLICE	0.30	0.12	0.15	0.22	0.27	0.33	0.75

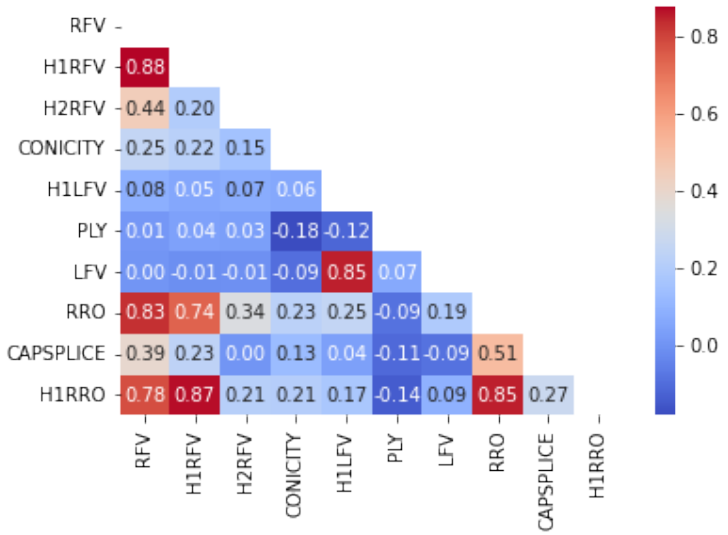


Fig. 6 Pearson correlations.

These components are linear combinations of the original variables, representing linearly uncorrelated variables, aiming to retain as much variation in the data as possible and approximate the covariance structure of the original variables. The technique involves finding linear combinations sequentially, where the first corresponds to the largest portion of variability, the next with the second largest portion, and so on.

This method allows us to analyze the entire set of variables in just two dimensions. The first principal component can be expressed as in equation (42), maximizing the variance in equation (43), subject to $\boldsymbol{\beta_1}^T\boldsymbol{\beta_1} = 1$. Here, $\mathbf{X}$ corresponds to the predictor variables, $\boldsymbol{\Sigma}$ is the covariance matrix, and $\boldsymbol{\beta}$ represents the coefficients. Similarly, the second principal component is obtained as $PC_2 = \boldsymbol{\beta_2}^T\boldsymbol{X}$, maximizing $\boldsymbol{\beta_2}^T\boldsymbol{\Sigma}\boldsymbol{\beta_2}$, with the constraints $\boldsymbol{\beta_2}^T\boldsymbol{\beta_2} = 1$ and $\boldsymbol{\beta_1}^T\boldsymbol{\beta_2} = 0$ to ensure orthogonality. Thus, the sum of the variances of the principal components equals the total variance of the system, as shown in equation

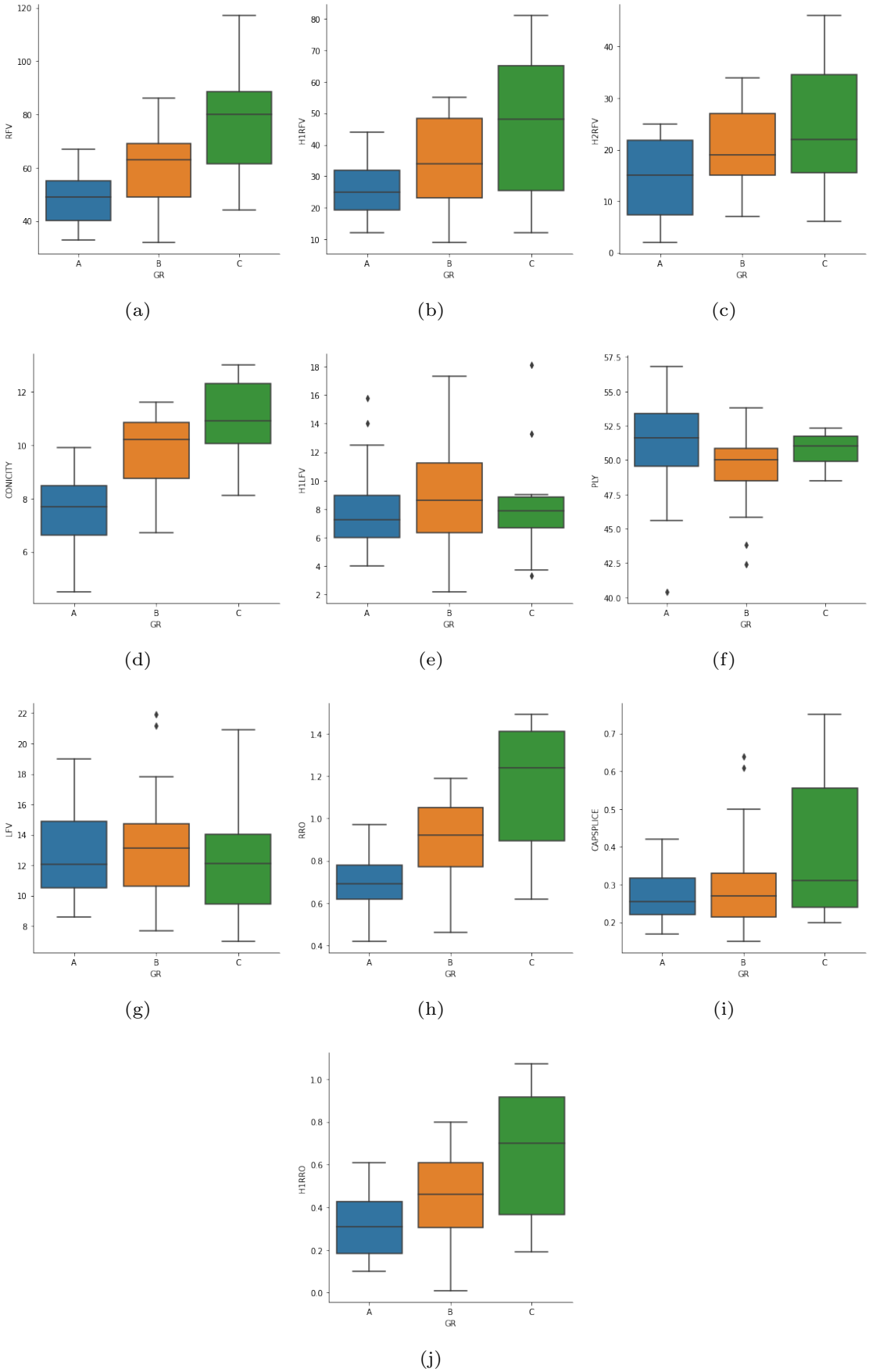


Fig. 7 Boxplot for each predictor variable based on tire performance.

(44), with estimated coefficients $\hat{\beta}_i$ and their variances $\hat{\lambda}_i$, corresponding, respectively, to the eigenvectors and eigenvalues of matrix $\mathbf{M}$.

$$PC_1 = \boldsymbol{\beta_1}^T \boldsymbol{X} = \beta_{1,1}X_1 + \cdots + \beta_{1,10}X_{10}. \tag{42}$$

$$Var(PC_1) = \boldsymbol{\beta_1}^T \boldsymbol{\Sigma} \boldsymbol{\beta_1}. \tag{43}$$

$$tr(\boldsymbol{M}) = tr(\sum_{i=1}^{2} \hat{\lambda}_i \hat{\beta}_i \hat{\beta}_i^T) = \sum_{i=1}^{2} \hat{\lambda}_i. \tag{44}$$

The Fig. 8 contains information about the two principal components. On the x-axis, the first principal component accounts for a variance of 40.4%, while the second component on the y-axis holds 19.7% of the variability. In this graph, colors represent the tire performance. It is noticeable that in this 2-dimensional projected space, the orange points corresponding to tire performance B are located between the points of classes A and C. This supports the idea that it is possible to map statistical patterns to identify class separation boundaries. The equations (45) and (46) correspond to the expressions of the PC_1 and PC_2 components, respectively. The variables Z_i, $i \in [1;\ 10]$, represent, respectively, the normalized variables: RFV; H1RFV; H2RFV; CONICITY; H1LFV; PLY; LFV; RRO; CAPSPLICE; H1RRO. The variables with the greatest contributions to PC_1 are: RFV, H1RFV, RRO.

$$\begin{aligned} PC_1 = {} & 0.43Z_1 + 0.45Z_2 + 0.19Z_3 + 0.19Z_4 + 0.13Z_5 \\ & -0.04Z_6 + 0.08Z_7 + 0.52Z_8 + 0.22Z_9 + 0.44Z_{10}. \end{aligned} \tag{45}$$

$$\begin{aligned} PC_2 = {} & -1 \cdot 10^{-1}Z_1 - 1 \cdot 10^{-1}Z_2 - 3 \cdot 10^{-1}Z_3 - 1 \cdot 10^{-1}Z_4 \\ & +7 \cdot 10^{-1}Z_5 - 10 \cdot 10^{-1}Z_6 + 7 \cdot 10^{-1}Z_7 \\ & +4 \cdot 10^{-1}Z_8 - 10 \cdot 10^{-1}Z_9 - 2 \cdot 10^{-1}Z_{10}. \end{aligned} \tag{46}$$

6.2 Feature Selection and Performance Models

Feature selection is the process of reducing the number of input variables for developing a predictive model. (Boratto et al, 2022) asserted that reducing the number of input variables is essential to decrease computational costs in modeling, avoid redundant information, and simplify models. The idea involves assessing the relationship between each predictor variable and the response variable using statistics to select those with the strongest relationship. The feature selection method utilized in this study combines the Pearson correlation coefficient with mutual information (Cover and Thomas, 2006).

Mutual information calculates a score for each feature based on entropy value. Therefore, a higher entropy value for a specific feature indicates a stronger association with the response variable. According to Vergara and Estévez (2014), mutual information is defined as shown in equation (47), where $P_{(\boldsymbol{X},\boldsymbol{Y})}$ is the joint probability density function of random variables X and Y, and $P_{(\boldsymbol{X})}$ and $P_{(\boldsymbol{Y})}$ are the marginal probability density functions of $\boldsymbol{X}$ and

Fig. 8 PCA to tire performance (GR).

$\boldsymbol{Y}$. The mutual information have a value of zero when the random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent.

$$I(\boldsymbol{X},\boldsymbol{Y}) = \int_{y\in\mathcal{Y}}\int_{x\in\mathcal{X}} P_{(\boldsymbol{X},\boldsymbol{Y})}(x,y) log\left(\frac{P_{(\boldsymbol{X},\boldsymbol{Y})}(x,y)}{P_{\boldsymbol{X}}(x)P_{\boldsymbol{Y}}(y)}\right) dxdy. \tag{47}$$

The proposed method involves selecting variables with Pearson correlation coefficients lower than 0.8. For values above this threshold, variables with the highest mutual information values were selected. These values were calculated and are listed in Table 3. Thus, the variables excluded from the study were: H1LFV; H1RFV; H1RRO.

Table 3 Information Gain of Predictor Variables.

Variables	Information Gain
CONICITY	0.38
RRO	0.31
RFV	0.19
H1RRO	0.18
H1RFV	0.17
H2RFV	0.11
PLY	0.05
LFV	0.02
CAPSPLICE	0.01
H1LFV	0.00

The SMOTE technique was employed to address the issue of data imbalance. This allowed the sample size of the training dataset to increase from 80

to 105 samples, with 35 elements in each class. The parameter for the number of nearest neighbors was set to 4.

For Bayesian optimization using BayesSearchCV, 64 iterations were considered for configurations of the number of sampled parameters. The Gaussian process was chosen as the optimizer. Cross-validation for model validation employed the Leave-one-out technique due to the limited amount of data. The optimized metric for each process was accuracy.

The search space for SVM parameters included $\gamma \in [0.01;\ 10]$, $C \in [0.1, 10]$, and the possible kernel scenarios: rbf, poly, and sigmoid. The KNN model used the Minkowski metric with the power parameter ranging from $[1, 3]$, and $[5, 8]$ for the number of neighbors, with leaf size between $[30, 50]$ for nearest neighbor calculations, and weight functions 'uniform' and 'distance'. For the GPC classifier, the covariance kernel function used was "RBF(1)", with the optimizer set as "fmin_l_bfgs_b" and "one_vs_one" as the classification strategy. The search interval for the maximum number of iterathe tions in the Newton method used to approximate the posterior during prediction was $[100, 130]$, and the number of optimizer restarts to find kernel parameters that maximize the log-marginal probability was $[2, 5]$. LR parameters had a search space of $C \in [10^{-5}, 100]$ with L1 and L2 penalties (Ridge and Lasso). For MLP, activation functions 'identity', 'logistic', 'tanh', and 'relu' were tested. The hidden layer neuron configuration was (100,). The learning rate ranged from $[10^{-4}, 0.6]$, alpha values were in the interval $[10^{-4}, 0.5]$, and weight updates were performed using the methods: 'constant', 'invscaling', and 'adaptive'.

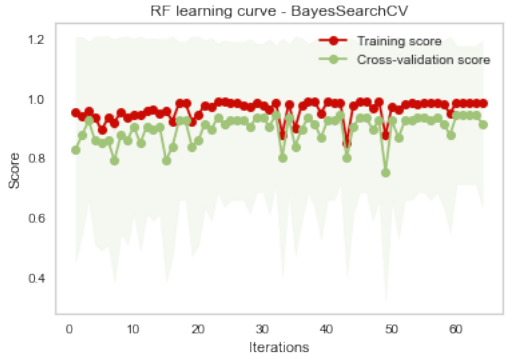
For RF, the search space for the number of trees in the forest ranged from $[5, 30]$, the maximum tree depth was set to $[4, 8]$, and the splitting quality was measured using Gini and entropy criteria. The number of features in the sample was set using 'sqrt' and 'log2', which were also used in GBDT. The minimum number of samples required to split an internal node and be in a leaf node were, respectively, $[8, 30]$ and $[2, 10]$. Finally, for boosting models GBDT and LGBM, the search space for the learning rate was $[10^{-3}, 0.5]$, and the number of estimators was in the interval $[20, 50]$. For GBDT, the search space for maximum tree depth was $[4, 7]$, the minimum number of samples required to split a node was $[5, 20]$, and for being a leaf node it was $[2, 10]$. The loss function 'deviance' was considered, as well as 'friedman_mse' and 'squared_error' criteria to measure the quality of a split. For LGBM, the search space for the minimum number of samples in a leaf was $[2, 10]$, and $[5, 20]$ for the number of leaves in a tree. The maximum tree depth was in the range $[5, 7]$, and the regularization term L1 had a search interval of $[10^{-3}, 0.5]$. For all numerical search intervals, a uniform prior distribution was considered.

The Fig. 9 represents the number of fitted models, and for each iteration, it shows the mean accuracy metric values in the training and validation sets during the Leave-one-out cross-validation process. The margin above the lines in the graph corresponds to the standard deviation of each model. Table

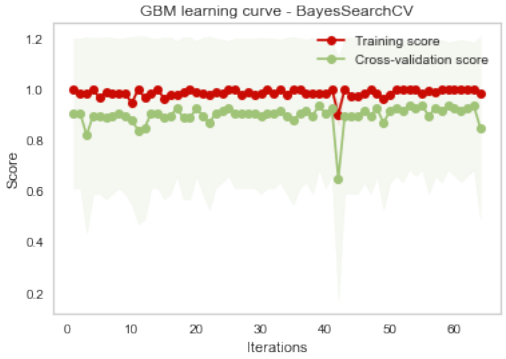
4 presents the accuracy values in the cross-validation process and the chosen parameters for each model. Notably, the models achieved an accuracy performance of around 90%

Table 4 Accuracy and Hyperparameters Cross-validation

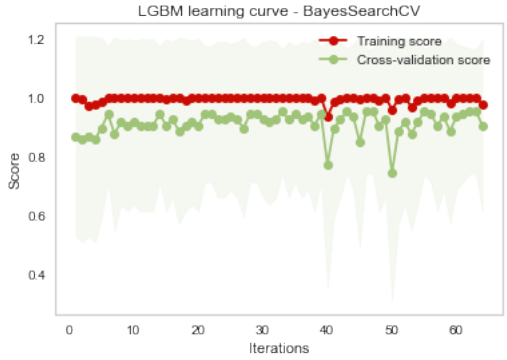
Models	Accuracy Cross-validation	Hyperparameters	Values
RF	0.94	criterion; max depth max features; min samples leaf; min samples split; min estimators	'entropy'; 7 'sqrt'; 2 8; 23
GBDT	0.93	criterion; learning rate; max depth max features; loss; n estimators min samples leaf; min samples split	'mse'; 0.03; 5 'sqrt'; 'deviance'; 50 10; 5
LGBM	0.95	max depth; learning rate min child samples; n estimators num iterations; num leaves objective; reg alpha	5; 0.05 7; 45 60; 5 'multiclass'; 0.23
LR	0.82	C; penalty solver; tol	24.8; 'l1' 'saga'; 10^{-3}
SVM	0.95	C; gamma; kernel	10.0; 3.9; 'rbf'
KNN	0.90	leaf size; n neighbors; p weights; metric	50; 6; 1 'minkowski'; 'distance'
GPC	0.67	max iter predict; optimizer n restarts optimizer; multi class	125; 'fmin l bfgs b' 2; 'one vs rest'
MLP	**0.96**	alpha; hidden layer sizes learning rate; learning rate init activation; solver	10^{-4}; (100,) 'invscaling'; 0.1 'relu'; 'adam'



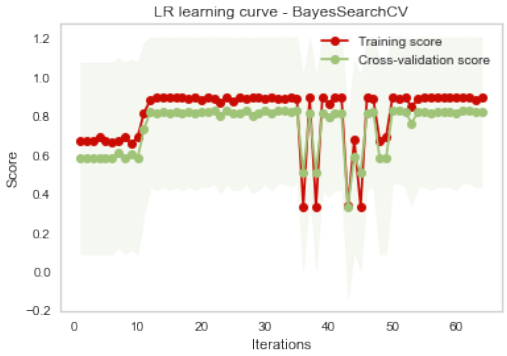
(a) RF



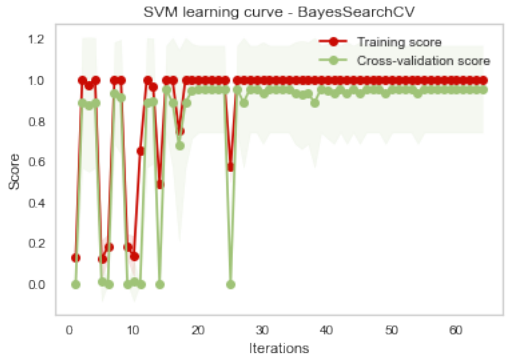
(b) GBDT



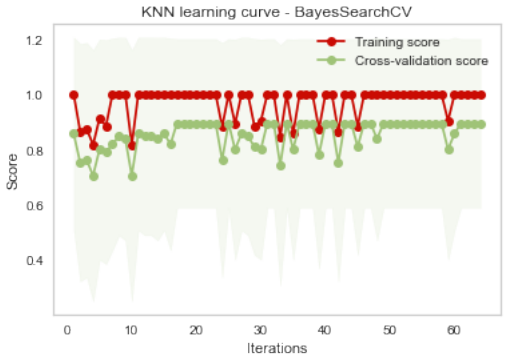
(c) LGBM



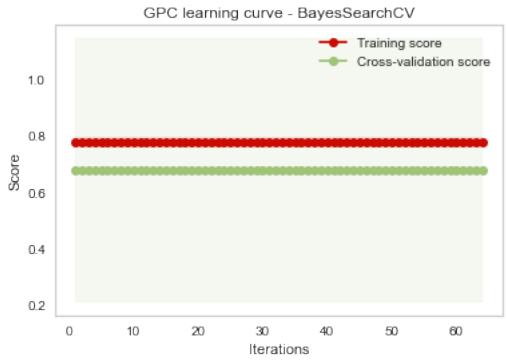
(d) LR



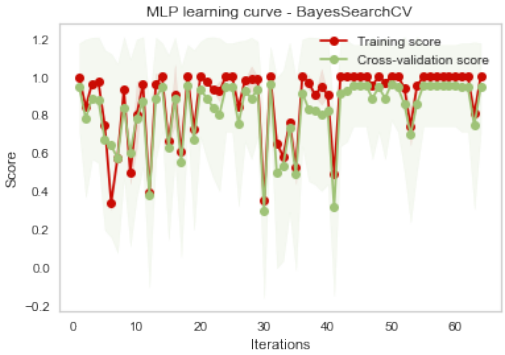
(e) SVM



(f) KNN



(g) GPC



(h) MLP

Fig. 9 Learning curves of the models based on Bayesian optimization iterations.

The performance metric was analyzed based on the number of training samples to assess the model's generalization ability. This serves as a mechanism to diagnose the bias and variance problem of the model. Fig. 10 shows the learning curves of the models. As the number of samples increases, the accuracy values of the cross-validation set converge toward the inference values in the training set. This methodology was also utilized by (Rezaei et al, 2022).

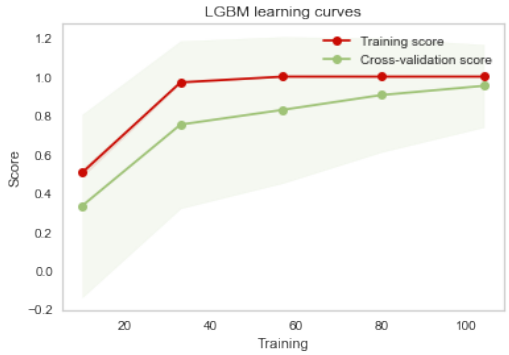
Table 5 presents the evaluation metrics based on the confusion matrix for the validation data. The SMOTE technique was not applied to this validation dataset. For each class of the response variable, the precision, recall, and F_1 score values were calculated for the corresponding models. The RF and GBDT models achieved the highest accuracy values. Notably, the MLP, LGBM, SVM, and KNN models failed to generalize the input patterns.

Table 5 Metrics on the validation dataset.

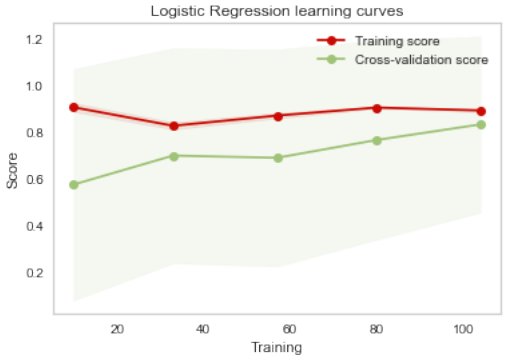
		Metrics[1]			
Models	GR	Accuracy	Precision	Recall	F_1 *score*
RF	A	**0.89**	1.00	1.00	1.00
	B		0.90	0.82	0.86
	C		0.60	0.75	0.67
GBDT	A	0.85	1.00	1.00	1.00
	B		0.82	0.82	0.82
	C		0.50	0.50	0.50
LGBM	A	0.81	0.92	1.00	0.96
	B		0.80	0.73	0.76
	C		0.50	0.50	0.50
LR	A	0.81	0.92	0.92	0.92
	B		0.75	0.82	0.78
	C		0.67	0.50	0.57
SVM	A	0.78	1.00	0.92	0.96
	B		0.73	0.73	0.73
	C		0.40	0.50	0.44
KNN	A	0.67	0.73	0.92	0.81
	B		0.62	0.45	0.53
	C		0.50	0.50	0.50
GPC	A	0.74	0.75	1.00	0.86
	B		0.83	0.45	0.59
	C		0.60	0.75	0.67
MLP	A	0.81	0.85	0.92	0.88
	B		0.75	0.82	0.78
	C		1.00	0.50	0.57

(a) RF

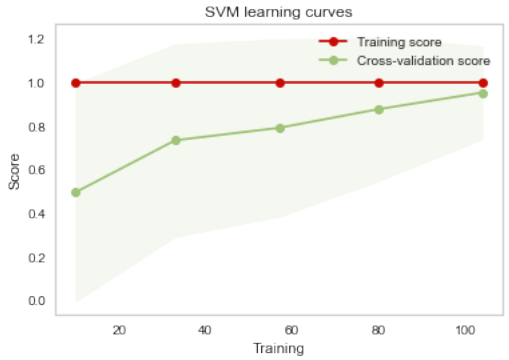
(b) GBDT



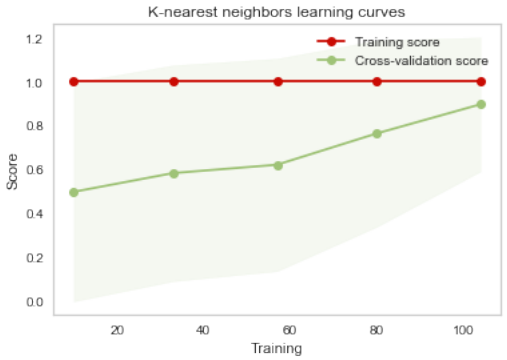
(c) LGBM



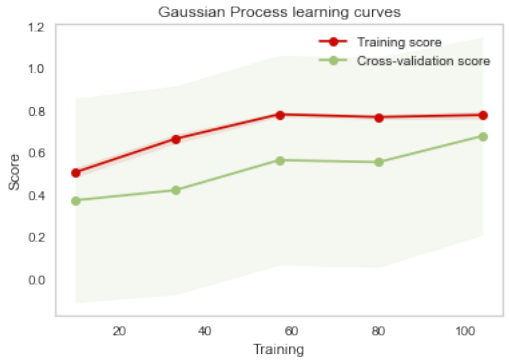
(d) LR



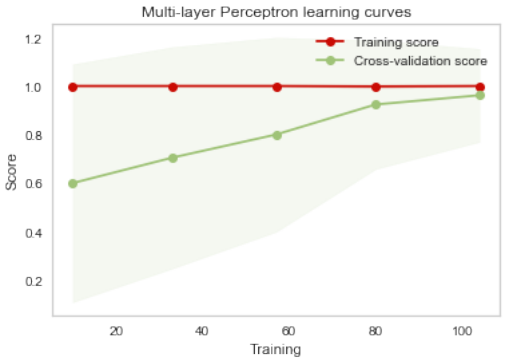
(e) SVM



(f) KNN



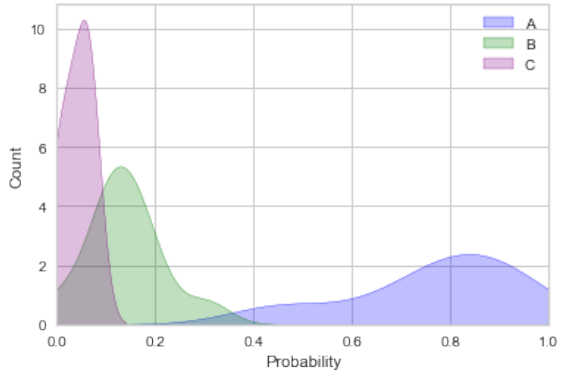
(g) GPC



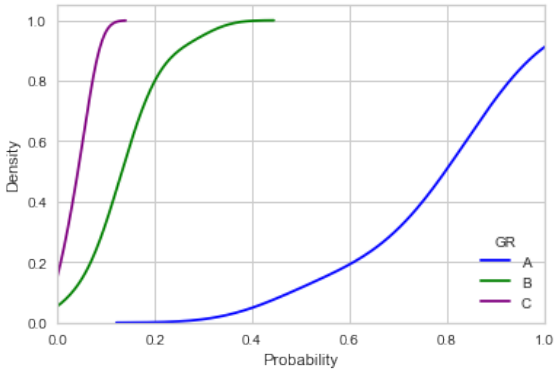
(h) MLP

Fig. 10 Learning curves of the models.

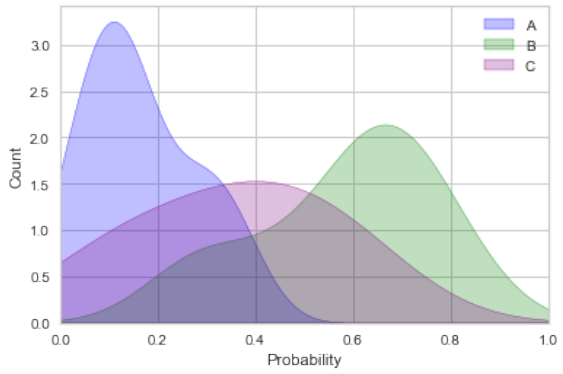
The RF model correctly predicted all samples from class A, 9 elements from class B, and 3 from class C. However, only 2 elements were misclassified in class C and 1 in class B, all from classes B and C. This result demonstrates that RF can be used for tire performance prediction. Figures 11(a), A11(c), and A11(e) represent the probabilities of the RF model in relation to the quantity of each class, while Figures A11(b), A11(d), and A11(f) represent the probabilities in relation to the density of each class.



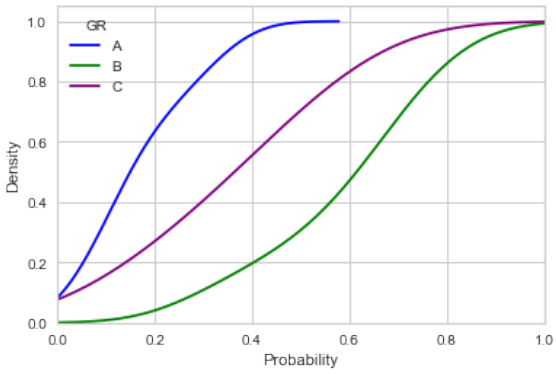
(a) Probability distribution of being in class A.



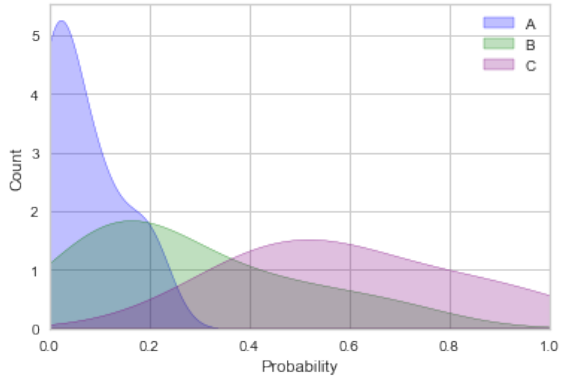
(b) Cumulative distribution of class being A.



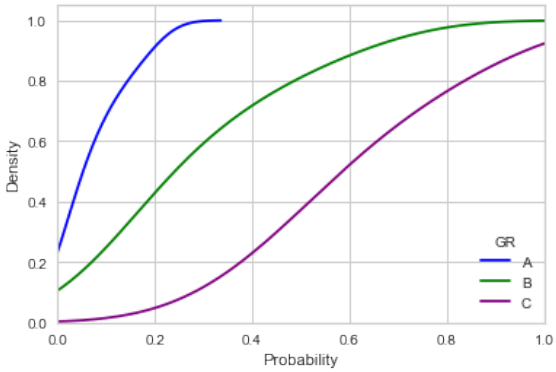
(c) Probability distribution of being in class B.



(d) Cumulative distribution of class being B.



(e) Probability distribution of being in class C.



(f) Cumulative distribution of class being C.

Fig. 11 RF model distributions

6.3 Model Interpretability

The SHAP methodology allows calculating the importance of variables and interpreting the effect of predictor variables on the response variable. With the RF model, it was possible to identify the explanatory variables with the highest importance for tire performance. RF was chosen due to its superior performance on the validation dataset. The Fig. 12 displays the SHAP values of the predictor variables for classes A, B, and C. The variables CONICITY, RFV, and H2RFV have a greater impact on class A, while the variable RRO has a higher contribution to class C. For class B, the variable PLY has a greater impact, and both variables LFV and CAPSPLICE contribute equally to classes B and C.

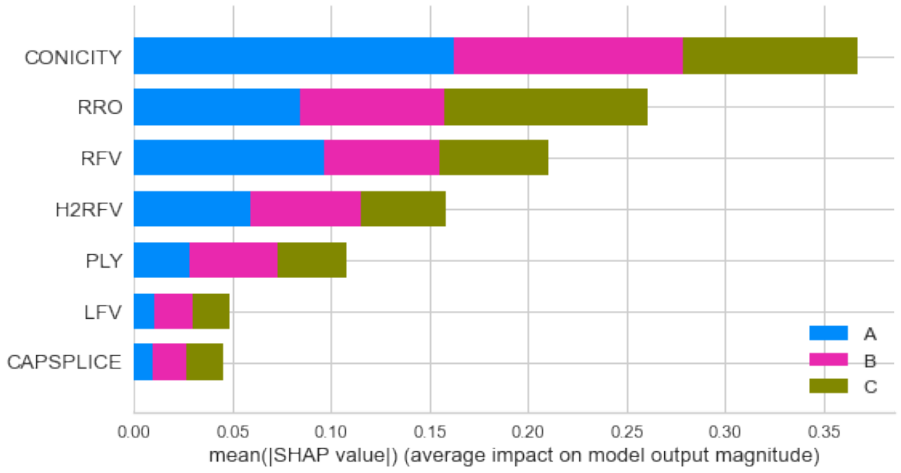


Fig. 12 SHAP importance

Figures 13, 14, and 15 correspond to the impacts of each sample element on tire performance classification. The variables are sorted in descending order based on their importance level in the model; the colors represent the impact of the explanatory variable, and the x-axis contains the SHAP values. The Fig. 13 indicates that low values of CONICITY, RFV, RRO, and H2RFV increase the probability of the class being A. For class B classification, it is noteworthy in Fig. 14 that low values of PLY and high values of LFV have a positive impact. In Fig. 15, high values of RRO, CONICITY, and RFV, along with low values of LFV, are important features for classifying as C.

SHAP dependence plots show the effects of interactions between predictor variables and the impact a variable has on a model's prediction. In Figures Appendix A1, A2, A3, you can identify interactions between predictor variables and the SHAP value. The relationship between RRO and RFV variables in Figure Appendix 1(a) for values in the range [0.5, 0.7] and [40, 60], respectively, shows a low impact for class B. However, values in the range [0.8, 1.0] for RRO and [60, 80] for RFV reveal an increase in the probability of class B.

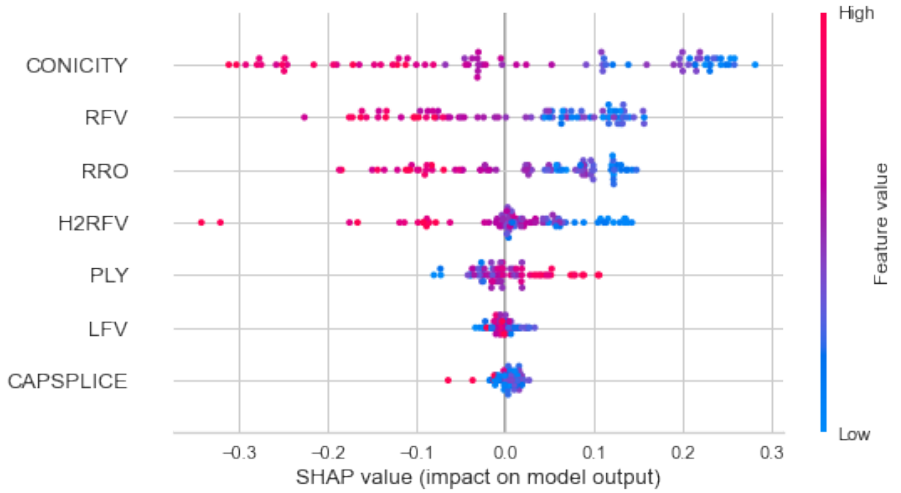


Fig. 13 SHAP Values classe A

Fig. 14 SHAP Values classe B

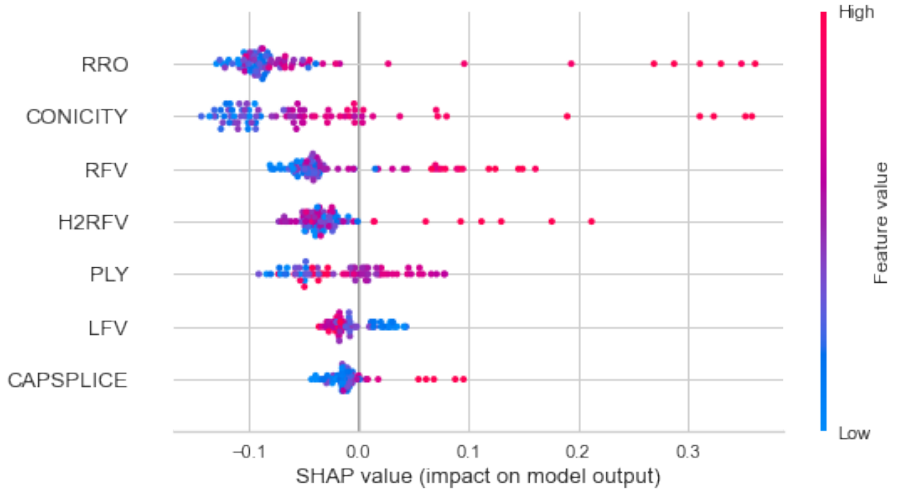


Fig. 15 SHAP Values classe C

50

7 Conclusion

Machine learning models can analyze the tire's performance in industrial processes in the context of classification. This article demonstrated that these models with distinct structures can efficiently predict tire performance classification based on predictor variables from the tire uniformity process. Exploratory analysis allowed us to identify that it is possible to find statistical patterns to separate classes with the help of PCA. Highly correlated variables were later removed in the variable selection using the highest mutual information scores.

A challenge of this study was dealing with small sample size, and the SMOTE technique allowed us to overcome this problem by creating artificial data and consequently increasing the sample size. For problem modeling, eight machine learning techniques were used: RF, GBDT, LGBM, LR, SVM, KNN, GPC, and MLP. To make the most of the training sample resources, the Leave-one-out technique was used. The RF model performed the best on the validation set, with higher accuracy and F_1-score values. Using the RF model with the aid of SHAP values analysis, it was possible to identify the importance of predictor variables for the response variable. The CONICITY, RRO, and RFV variables have the highest global impacts on tire performance, the same variables found by the mutual information method.

The works' results are of great importance for tire manufacturing processes and offer significant implications for the contribution of operational management in factories for fault detection and, consequently, cost reduction. These factories can benefit from the model's result and use the probability score for decision-making. SHAP value graphs regarding tire performance allow the identification of variables with the greatest local impact. Our results indicate that small SHAP values for CONICITY, RRO, RFV, and H2RFV are highly important for class A. This is crucial for factories to develop high-quality tires, ensuring a longer lifespan and reducing environmental impact. The limitations are inherent to the sample size. Several statistical techniques were used to overcome this problem, although the proposed model's result can achieve accuracy.

Future research should study decision boundaries of input variables, decision trees, or parametric models with possible confidence intervals for these variables. This can enable the implementation of policies for manufacturing tires with better performance. An unsupervised approach to identify groups and analyze their concordance with tire performances can provide relevant information. This may lead to robust approaches and allow a deeper understanding of the relationships between variables. The proposed approaches have great potential to be applied for fault detection or anomaly detection of tires, bearings, and gears, having good prospects in intelligent industrial manufacturing.

Acknowledgments

We would like to thank the Industrial Engineering Program at the Polytechnic School of the Federal University of Bahia and CAPES for the financial and administrative support provided for the development of this research.

Disclosure of potential conflicts of interest

The authors declare that no known competing financial interests or personal relationships could have appeared to influence the work reported in this paper.

Funding

The work was supported by the National Council of Scientific and Technological Development [309812/2021-6].

Appendix A Graphics

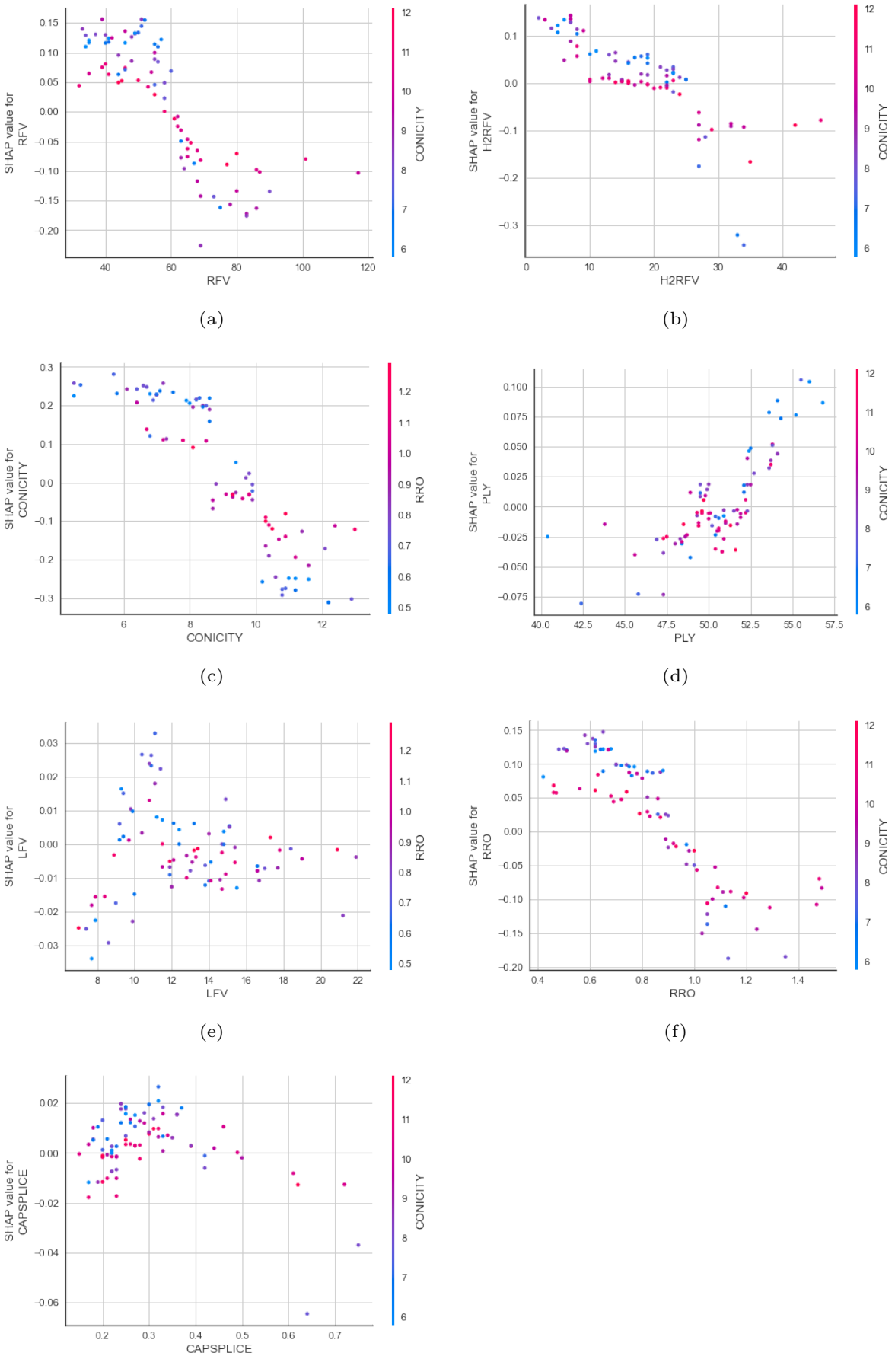


Fig. A1 SHAP dependence plot - class A

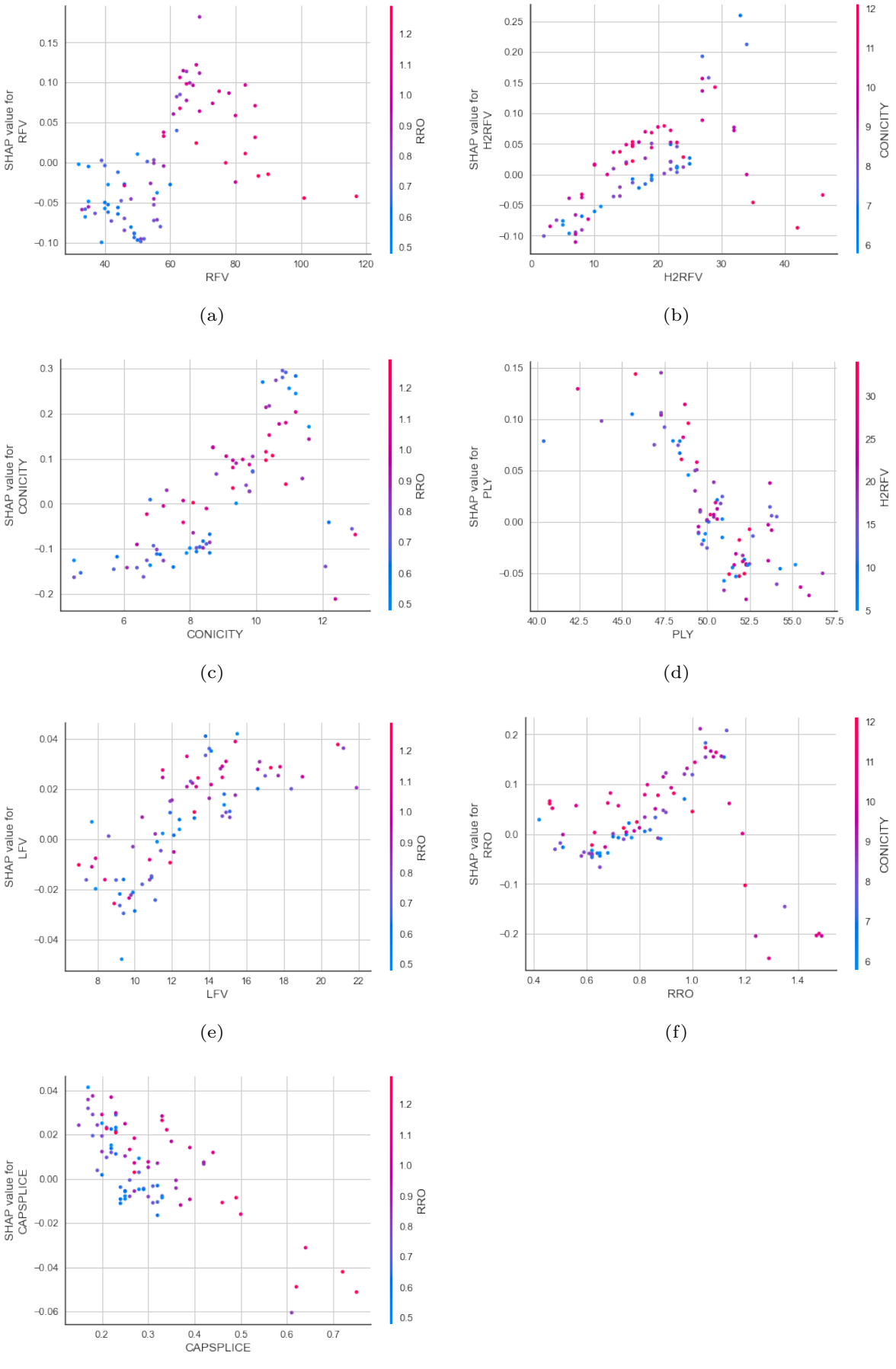


Fig. A2 SHAP dependence plot - class B

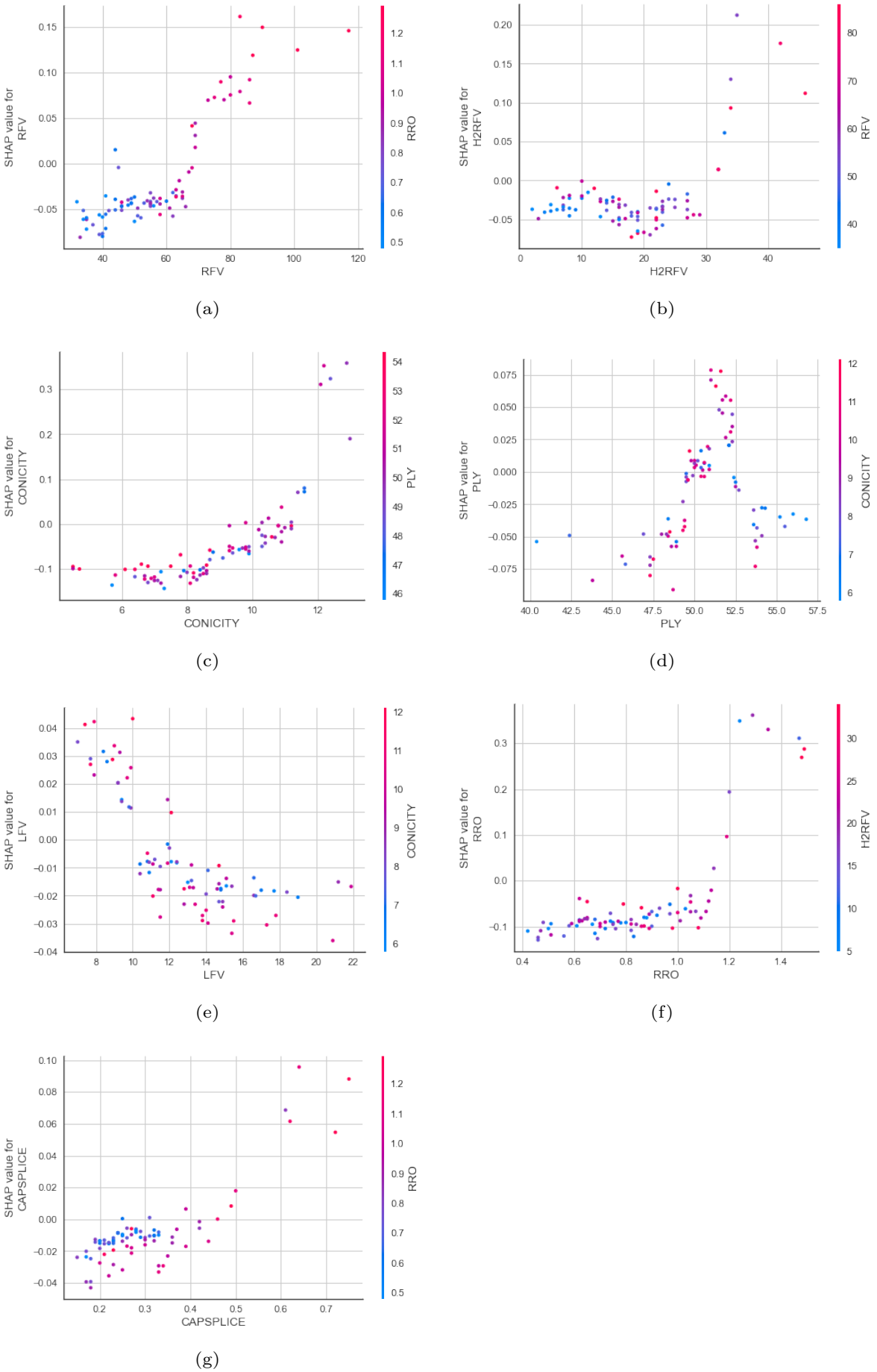


Fig. A3 SHAP dependence plot - class C

55

References

Acosta SM, Amoroso AL, Sant Anna AMO, et al (2021) Relevance vector machine with tuning based on self-adaptive differential evolution approach for predictive modelling of a chemical process. Applied Mathematical Modelling 95:125–142. https://doi.org/10.1016/J.APM.2021.01.057

Acosta SM, Oliveira RMA, Sant'Anna O (2023) Machine learning algorithms applied to intelligent tyre manufacturing. International Journal of Computer Integrated Manufacturing pp 1–11. https://doi.org/10.1080/0951192X.2023.2177734, URL https://www.tandfonline.com/doi/abs/10.1080/0951192X.2023.2177734

Ahmad GN, Fatima H, Shafiullah, et al (2022) Efficient Medical Diagnosis of Human Heart Diseases using Machine Learning Techniques with and without GridSearchCV. IEEE Access pp 1–1. https://doi.org/10.1109/ACCESS.2022.3165792, URL https://ieeexplore.ieee.org/document/9751602/

Amari Si (1993) Backpropagation and stochastic gradient descent method. Neurocomputing 5(4-5):185–196. https://doi.org/10.1016/0925-2312(93)90006-O

Arlot S, Celisse A (2010) A survey of cross-validation procedures for model selection. https://doiorg/101214/09-SS054 4(none):40–79. https://doi.org/10.1214/09-SS054, URL https://projecteuclid.org/journals/statistics-surveys/volume-4/issue-none/A-survey-of-cross-validation-procedures-for-model-selection/10.1214/09-SS054.fullhttps://projecteuclid.org/journals/statistics-surveys/volume-4/issue-none/A-survey-of-cross-validat

Boratto TH, Saporetti CM, Basilio SC, et al (2022) Data-driven cymbal bronze alloy identification via evolutionary machine learning with automatic feature selection. Journal of Intelligent Manufacturing pp 1–17. https://doi.org/10.1007/S10845-022-02047-3/METRICS, URL https://link.springer.com/article/10.1007/s10845-022-02047-3

Breiman L (1996) Heuristics of instability and stabilization in model selection. https://doiorg/101214/aos/1032181158 24(6):2350–2383. https://doi.org/10.1214/AOS/1032181158, URL https://projecteuclid.org/journals/annals-of-statistics/volume-24/issue-6/Heuristics-of-instability-and-stabilization-in-model-selection/10.1214/aos/1032181158.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-24/issue-6/Heuristics-of-in

Breiman L (2001) Random Forests 45:5–32

Bustillo A, Reis R, Machado AR, et al (2022) Improving the accuracy of machine-learning models with data from machine test repetitions. Journal of Intelligent Manufacturing 33(1):203–221. https://doi.org/10.1007/S10845-020-01661-3/FIGURES/9, URL https://link.springer.com/article/10.1007/s10845-020-01661-3

Chawla NV, Bowyer KW, Hall LO, et al (2011) SMOTE: Synthetic Minority Over-sampling Technique. Journal Of Artificial Intelligence Research 16:321–357. https://doi.org/10.1613/jair.953, URL http://arxiv.org/abs/1106.1813http://dx.doi.org/10.1613/jair.953

Cho JR, Lee JH, Jeong KM, et al (2012) Optimum design of run-flat tire insert rubber by genetic algorithm. Finite Elements in Analysis and Design 52:60–70. https://doi.org/10.1016/J.FINEL.2011.12.006

Cortes C, Vapnik V, Saitta L (1995) Support-vector networks. Machine Learning 1995 20:3 20(3):273–297. https://doi.org/10.1007/BF00994018, URL https://link.springer.com/article/10.1007/BF00994018

Cover TM, Thomas JA (2006) Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing) p 776. URL http://www.amazon.com/Elements-Information-Edition-Telecommunications-Processing/dp/0471241954

Demšar J (2006) Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research 7:1–30

Deng X, Liu Q, Deng Y, et al (2016) An improved method to construct basic probability assignment based on the confusion matrix for classification problem. Information Sciences 340-341:250–261. https://doi.org/10.1016/J.INS.2016.01.033

Du M, Sun P, Zhou S, et al (2020) A Study on the Influence of Tire Speed and Pressure on Measurement Parameters Obtained from High-Speed Tire Uniformity Testing. Vehicles 2020, Vol 2, Pages 559-573 2(3):559–573. https://doi.org/10.3390/VEHICLES2030031, URL https://www.mdpi.com/2624-8921/2/3/31/htmhttps://www.mdpi.com/2624-8921/2/3/31

Fan J, Ma X, Wu L, et al (2019) Light Gradient Boosting Machine: An efficient soft computing model for estimating daily reference evapotranspiration with local and external meteorological data. Agricultural Water Management 225. https://doi.org/10.1016/J.AGWAT.2019.105758

Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). https://doiorg/101214/aos/1016218223

28(2):337–407. https://doi.org/10.1214/AOS/1016218223, URL https://projecteuclid.org/journals/annals-of-statistics/volume-28/issue-2/Additive-logistic-regression--a-statistical-view-of-boosting-With/10.1214/aos/1016218223.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-28/issue-2/Additive-logi

Friedman JH (2001) Greedy function approximation: A gradient boosting machine. https://doiorg/101214/aos/1013203451 29(5):1189–1232. https://doi.org/10.1214/AOS/1013203451, URL https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-appro

Friedman JH (2002) Stochastic gradient boosting. Computational Statistics and Data Analysis 38(4):367–378. https://doi.org/10.1016/S0167-9473(01)00065-2

Gonçalves JP, Ambrósio JA (2005) Road vehicle modeling requirements for optimization of ride and handling. Multibody System Dynamics 13(1):3–23. https://doi.org/10.1007/S11044-005-2528-5

Greis NP, Nogueira ML, Bhattacharya S, et al (2023) Stability modeling for chatter avoidance in self-aware machining: an application of physics-guided machine learning. Journal of Intelligent Manufacturing 34(1):387–413. https://doi.org/10.1007/S10845-022-01999-W/FIGURES/18, URL https://link.springer.com/article/10.1007/s10845-022-01999-w

Gutierrez-Gomez L, Petry F, Khadraoui D (2020) A Comparison Framework of Machine Learning Algorithms for Mixed-Type Variables Datasets: A Case Study on Tire-Performances Prediction. IEEE Access 8:214902–214914. https://doi.org/10.1109/ACCESS.2020.3041367

Hastie T, Tibshirani R, Friedman J (2009) The Elements of Statistical Learning https://doi.org/10.1007/978-0-387-84858-7, URL http://link.springer.com/10.1007/978-0-387-84858-7

Haykin S (2008) Neural Networks and Learning Machines. Pearson Prentice Hall New Jersey USA 936 pLinks 3:906. https://doi.org/978-0131471399

Jabeur SB, Mefteh-Wali S, Viviani JL (2021) Forecasting gold price with the XGBoost algorithm and SHAP interaction values. Annals of Operations Research pp 1–21. https://doi.org/10.1007/S10479-021-04187-W/FIGURES/4, URL https://link.springer.com/article/10.1007/s10479-021-04187-w

James G, Witten D, Hastie T, et al (2013) Statistical Learning pp 15–57. https://doi.org/10.1007/978-1-4614-7138-7{_}2

Ke G, Meng Q, Finley T, et al (2017) LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems 30. URL https://github.com/Microsoft/LightGBM.

Kecman V (2005) Support Vector Machines – An Introduction pp 1–47. https://doi.org/10.1007/10984697{_}1

Kingma DP, Lei Ba J (2015) Adam: A Method for Stochastic Optimization. undefined

Lee SK, Lee H, Back J, et al (2021) Prediction of tire pattern noise in early design stage based on convolutional neural network. Applied Acoustics 172. https://doi.org/10.1016/J.APACOUST.2020.107617

Li X, Guo M, Zhou X (2021) A multivariate multiple regression analysis of tire-road contact peak triaxial stress by using machine learning methods. Mechanics of Advanced Materials and Structures https://doi.org/10.1080/15376494.2021.2008067

Liu Z, Wang F, Cai Z, et al (2023) Stochastic analysis for in-plane dynamic responses of low-speed uniformity of tires due to geometric defects. Mechanical Systems and Signal Processing 197:110377. https://doi.org/10.1016/J.YMSSP.2023.110377

Lu Y, Liu C, Wang KI, et al (2020) Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. Robotics and Computer-Integrated Manufacturing 61. https://doi.org/10.1016/J.RCIM.2019.101837

Lundberg SM, Lee SI (2017) A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems 2017-Decem:4766–4775. https://doi.org/10.48550/arxiv.1705.07874, URL https://arxiv.org/abs/1705.07874v2

Lutz B, Kisskalt D, Mayr A, et al (2021) In-situ identification of material batches using machine learning for machining operations. Journal of Intelligent Manufacturing 32(5):1485–1495. https://doi.org/10.1007/S10845-020-01718-3/FIGURES/8, URL https://link.springer.com/article/10.1007/s10845-020-01718-3

Mackay DJC (1996) INTRODUCTION TO GAUSSIAN PROCESSES URL http://www.cs.toronto.edu/~radford/.

Masum M, Shahriar H, Haddad H, et al (2022) Bayesian Hyperparameter Optimization for Deep Neural Network-Based Network Intrusion Detection. Proceedings - 2021 IEEE International Conference on Big Data, Big Data 2021 pp 5413–5419. https://doi.org/10.1109/BigData52589.2021.9671576, URL http://arxiv.org/abs/2207.09902http://dx.doi.org/10.1109/BigData52589.2021.9671576

Meijer RJ, Goeman JJ (2013) Efficient approximate k-fold and leave-one-out cross-validation for ridge regression. Biometrical Journal 55(2):141–155. https://doi.org/10.1002/BIMJ.201200088

Mishra AK, Paliwal S (2022) Mitigating cyber threats through integration of feature selection and stacking ensemble learning: the LGBM and random forest intrusion detection perspective. Cluster Computing pp 1–12. https://doi.org/10.1007/S10586-022-03735-8/FIGURES/3, URL https://link.springer.com/article/10.1007/s10586-022-03735-8

Mokhtari A, Ribeiro A (2015) Global Convergence of Online Limited Memory BFGS. Journal of Machine Learning Research 16:3151–3181

Penumuru DP, Muthuswamy S, Karumbu P (2020) Identification and classification of materials using machine vision and machine learning in the context of industry 4.0. Journal of Intelligent Manufacturing 31(5):1229–1241. https://doi.org/10.1007/S10845-019-01508-6/TABLES/7, URL https://link.springer.com/article/10.1007/s10845-019-01508-6

Probst P, Boulesteix AL (2018) To Tune or Not to Tune the Number of Trees in Random Forest. Journal of Machine Learning Research 18:1–18. URL http://jmlr.org/papers/v18/17-269.html.

Rajeswari M, Julie EG, Robinson YH, et al (2022) Detection of tyre defectsusingweighted quality-based convolutional neural network. Soft Computing 26(9):4261–4273. https://doi.org/10.1007/S00500-022-06878-3

Raschka S (2018) Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning https://doi.org/10.48550/arxiv.1811.12808, URL https://arxiv.org/abs/1811.12808v3

Rasmussen CE (2004) Gaussian Processes in Machine Learning. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 3176:63–71. https://doi.org/10.1007/978-3-540-28650-9{_}4, URL https://link.springer.com/chapter/10.1007/978-3-540-28650-9_4

Rasmussen CE, Williams CKI (2005) Gaussian Processes for Machine Learning. Gaussian Processes for Machine Learning https://doi.org/10.7551/MITPRESS/3206.001.0001, URL https://direct.mit.edu/books/

book/2320/Gaussian-Processes-for-Machine-Learning

Rezaei HB, Amjadian A, Mohammad et al (2022) An ensemble method of the machine learning to prognosticate the gastric cancer. Annals of Operations Research 2022 pp 1–42. https://doi.org/10.1007/S10479-022-04964-1, URL https://link.springer.com/article/10.1007/s10479-022-04964-1

Rosati R, Romeo L, Cecchini G, et al (2023) From knowledge-based to big data analytic model: a novel IoT and machine learning based decision support system for predictive maintenance in Industry 4.0. Journal of Intelligent Manufacturing 34(1):107–121. https://doi.org/10.1007/S10845-022-01960-X/FIGURES/8, URL https://link.springer.com/article/10.1007/s10845-022-01960-x

Santos JI, Pereda M, Ahedo V, et al (2023) Explainable machine learning for project management control. Computers & Industrial Engineering 180:109261. https://doi.org/10.1016/J.CIE.2023.109261

Scornet E, Biau G, Vert JP (2015) CONSISTENCY OF RANDOM FORESTS 1. The Annals of Statistics 43(4):1716–1741. https://doi.org/10.1214/15-AOS1321, URL http://www.kaggle.com/c/dsg-hackathon

Shao Z, Er MJ (2016) Efficient Leave-One-Out Cross-Validation-based Regularized Extreme Learning Machine. Neurocomputing 194:260–270. https://doi.org/10.1016/J.NEUCOM.2016.02.058

Štrumbelj E, Kononenko I (2014) Explaining prediction models and individual predictions with feature contributions. Knowledge and Information Systems 41(3):647–665. https://doi.org/10.1007/S10115-013-0679-X/TABLES/4, URL https://link.springer.com/article/10.1007/s10115-013-0679-x

Tercan H, Meisen T (2022) Machine learning and deep learning based predictive quality in manufacturing: a systematic review. Journal of Intelligent Manufacturing 2022 33:7 33(7):1879–1905. https://doi.org/10.1007/S10845-022-01963-8, URL https://link.springer.com/article/10.1007/s10845-022-01963-8

Usuga Cadavid JP, Lamouri S, Grabot B, et al (2020) Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0. Journal of Intelligent Manufacturing 31(6):1531–1558. https://doi.org/10.1007/S10845-019-01531-7/FIGURES/23, URL https://link.springer.com/article/10.1007/s10845-019-01531-7

Vergara JR, Estévez PA (2014) A review of feature selection methods based on mutual information. Neural Computing and Applications 24(1):175–186. https://doi.org/10.1007/S00521-013-1368-0/TABLES/4, URL https://link.springer.com/article/10.1007/s00521-013-1368-0

Wang Y, Zhang Y, Zheng L, et al (2021) Unsupervised Learning with Generative Adversarial Network for Automatic Tire Defect Detection from X-ray Images. Sensors 2021, Vol 21, Page 6773 21(20):6773. https://doi.org/10.3390/S21206773, URL https://www.mdpi.com/1424-8220/21/20/6773/htmhttps://www.mdpi.com/1424-8220/21/20/6773

Weyssenhoff A, Opala M, Koziak S, et al (2019) Characteristics and investigation of selected manufacturing defects of passenger car tires. Transportation Research Procedia 40:119–126. https://doi.org/10.1016/J.TRPRO.2019.07.020

Wong TT (2015) Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. Pattern Recognition 48(9):2839–2846. https://doi.org/10.1016/J.PATCOG.2015.03.009

Wu J, Mei J, Wen S, et al (2010) A self-adaptive genetic algorithm-artificial neural network algorithm with leave-one-out cross validation for descriptor selection in QSAR study. Journal of Computational Chemistry 31(10):1956–1968. https://doi.org/10.1002/JCC.21471, URL https://onlinelibrary-wiley.ez10.periodicos.capes.gov.br/doi/full/10.1002/jcc.21471https://onlinelibrary-wiley.ez10.periodicos.capes.gov.br/doi/abs/10.1002/jcc.21471https://onlinelibrary-wiley.ez10.periodicos.capes.gov.br/doi/10.1002/jcc.21471

Xu Q, Fu R, Wu F, et al (2021) Roadside estimation of a vehicle’s center of gravity height based on an improved single-stage detection algorithm and regression prediction technology. IEEE Sensors Journal https://doi.org/10.1109/JSEN.2021.3114703

Young MT, Hinkle J, Ramanathan A, et al (2019) HyperSpace: Distributed Bayesian Hyperparameter Optimization. Proceedings - 2018 30th International Symposium on Computer Architecture and High Performance Computing, SBAC-PAD 2018 pp 339–347. https://doi.org/10.1109/CAHPC.2018.8645954

Zhou B, Pychynski T, Reischl M, et al (2022) Machine learning with domain knowledge for predictive quality monitoring in resistance spot welding. Journal of Intelligent Manufacturing 33(4):1139–1163. https://doi.org/10.1007/S10845-021-01892-Y/TABLES/8, URL https://link.springer.com/article/10.1007/s10845-021-01892-y

Zhu J, Han K, Wang S (2021) Automobile tire life prediction based on image processing and machine learning technology:. https://doi-orgez10periodicoscapesgovbr/101177/16878140211002727 13(3):1–13. https://doi.org/10.1177/16878140211002727, URL https://journals-sagepub-com.ez10.periodicos.capes.gov.br/doi/10.1177/16878140211002727

3 Artigo 2 - Explainable machine learning models for defects detection in industrial processes

Explainable machine learning models for defects detection in industrial processes

Rodrigo Marcel Araujo Oliveira[a], Ângelo Márcio Oliveira Sant'Anna[a]

[a]*Polytechnic School, Federal University of Bahia, Prof. Aristídes Novis street 02, Salvador 40.210-630, Bahia, Brazil*

ARTICLE INFO

Keywords:
Explainable Artificial Intelligence
Machine Learning
Defect Detection
Tire Manufacturing
Optimization

ABSTRACT

Machine learning algorithms in non-linear pattern recognition for defect detection in manufacturing processes are increasingly prevalent in the context of Industry 4.0. This scenario enables factories to develop methodologies for monitoring and controlling the quality of their products, obtaining better operational efficiency indicators, and offering increasingly competitive products in the market. Explainable Artificial Intelligence can facilitate understanding how the models make decisions and assist in tracking anomaly points. This paper proposes an explainability framework for machine learning models in defect detection. This work applied the approach in the tire manufacturing process from the multinational industry. Models such as random forest, gradient boosting decision tree, light gradient boosting machine, logistic regression, support vector machine, and multi-layer perceptron were considered to evaluate the performance of tires in compliance with production standards. The Local Interpretable Model-Agnostic Explanations and SHapley Additive exPlanations methods were used to identify the relevant process variables. The proposed framework determines reference values for each selected process variable for tire manufacturing in compliance with quality standards. This approach provides a relevant tool for quality control management in tire manufacturing industries.

1. Introduction

Industry 4.0 is a revolutionary process in product distribution and manufacturing characterized by integrating technology systems across various sectors of the production chain. This new technological landscape allows for automation, enabling robots to perform increasingly complex and efficient tasks. Artificial Intelligence (AI) plays a fundamental role in this transformative process, enabling the analysis of system behaviors, interpretation of events, identification of anomalies, and assisting in decision-making (Acosta et al., 2023). The use of AI technologies in manufacturing processes is of utmost importance for optimizing operational resources, reducing costs, and providing competitive advantages for companies in the market. This enables companies to offer better solutions to society, ranging from product quality and price accessibility to the availability of products with reduced environmental impact.

Anomaly refers to an unexpected deviation that various processes can generate. Anomaly detection serves as a pattern change alert, detecting equipment failures, defects, fraud in financial systems, genetic mutations, and more. Machine learning (ML) methods have been employed for fault detection in factory production line equipment (Carvalho et al., 2019). Algorithms like random forest (RF), gradient boosting decision tree (GBDT), light gradient boosting machine (LGBM), logistic regression (LR), support vector machine (SVM), and multi-layer perceptron (MLP) allow learning from data and attribute classification. Optimization techniques such as Genetic Algorithms (GA) are commonly used in conjunction with these models for parameter selection. Explainable Artificial Intelligence (XAI) is crucial for risk mitigation and understanding events, possibly guiding decision-making (Ali et al., 2023). Methodologies like Local Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP) are widely recognized techniques for ML model explainability. These techniques enable an understanding of how models make decisions by identifying the contribution of each predictor variable to the final classification (Santos et al., 2023).

Although there have been several works on the applications of machine learning algorithms for different knowledge areas. This paper proposes an explainability framework for machine learning algorithms in defect detection. This work applied the proposed framework in the tire manufacturing process from the multinational industry. The main contribution of this research is developing an efficient explainability scheme for identifying reference values for process variables with the greatest influence on defect detection to industrial processes in real-time data in large manufacturing processes, focusing on failures and loss detection.

The research involves the development of various ML models for tire performance classification. Section 2 provides a literature review of different AI system approaches applied in industrial contexts. Theoretical concepts of RF, GBDT, LGBM, LR, SVM, and MLP algorithms used for modeling are discussed in Section 3, along with the parameter optimization methodology using GA, and the LIME and SHAP approaches for model explainability. Section 4 introduces the proposed methodology for model evaluation, sampling techniques, and attribute selection. Additionally, this section presents an approach to compare model results using McNemar's statistical test and a method to identify classification boundary points for each predictor variable of the model to define optimal points for compliant tire manufacturing. The case study explanation is presented in

rodrigomarcel@ufba.br (R.M.A. Oliveira)
ORCID(s): 0000-0002-2646-2882 (R.M.A. Oliveira); 0000-0001-8332-8877 (M.O. Sant'Anna)

Section 5. Section 6 presents the results, including descriptive data analysis, inference analysis, model performance evaluation, explainability of results, and optimization of optimal classification points. Section 7 summarizes the results obtained, contributions, and limitations of the study and suggests directions for future work.

2. Related works

Defect detection is an integral part of the strategic planning in the manufacturing process of various industrial products, ensuring the quality assurance of manufactured products and resulting in cost savings. Recent studies have demonstrated the efficiency of machine learning models in detecting non-linear patterns in manufacturing processes. Different machine learning models have been used to predict phosphorus concentration levels in a steel mill (Acosta et al., 2021). Anomaly detection using machine learning models has been studied to assess dam safety (Salazar et al., 2021). The study compared classifiers and concluded that these models can distinguish various anomalies. The results indicated that SVM was the classifier with the highest predictive power (Carvalho et al., 2019). The authors conducted a literature review on machine learning methods applied to Predictive Maintenance (PdM), a crucial factor in the operational efficiency of industries. Techniques such as SVM, RF, Artificial Neural Networks, and k-means were discussed for applications in fault detection of equipment (Zhan et al., 2021).

A novel method for detecting anomalies in time series data, named HR-AD, was proposed in Zhan et al. (2021) based on a hierarchical representation of time series. The Industrial Internet of Things (IIoT) aims to connect equipment and applications in advanced manufacturing processes, providing a large volume of sensor data for real-time monitoring. Authors developed the HR-AD method, which proved effective in anomaly detection (Cakir et al., 2021). A solution based on IIoT and machine learning models was developed for bearing damage classification, where the decision tree model showed quick results and acceptable performance metrics compared to SVM, Linear Discriminant Analysis (LDA), RF, and KNN models (Cakir et al., 2021).

Demšar (2006) reviewed the proposals for the Wilcoxon signed-rank test for comparing two classifiers and the Friedman test with corresponding post-hoc tests for comparisons with multiple classifiers across various datasets. Relevance vector machines (RVM) for tire weight prediction and anomaly detection were proposed in Acosta et al. (2023). A differential evolution algorithm was employed to optimize RVM model parameters, showing superior performance compared to other machine learning algorithms. This approach allowed for automated fault detection and proposed reference values for tire weight, enhancing manufacturing process performance (Gutierrez-Gomez et al., 2020). Predicting tire performance using different machine learning algorithms involving a mix of numerical and categorical variables was discussed by Gutierrez-Gomez et al. (2020), utilizing methods such as Support Vector Regression (SVR), k-Nearest Neighbors (KNN), Gradient Boosting (GBoost), and RF. Different encoding methods for handling categorical variables combined with machine learning methods were evaluated.

Deep learning models presented in Ko et al. (2021) were used for tire defect detection based on depth images. The proposed model utilized deep learning segmentation to detect atypical defect data, aiming to identify ventilation splashes occurring during tire manufacturing. Challenges were reported due to limited information in grayscale depth images, as well as the varying sizes and shapes of the defects for different tires. Predicting the remaining lifespan of sensor-equipped equipment is a crucial step in prognostic and integrity management systems used in machines to reduce economic losses. Authors proposed a methodology using Long Short-Term Memory (LSTM) for predicting the transition point from the health stage to the degradation stage of these machines, obtaining a confidence interval as a result (Yan et al., 2022). Predicting solder paste stencil hole fill during printing using Artificial Neural Networks (ANN) was studied in Martinek and Krammer (2019). The authors applied models based on neuro-fuzzy adaptive inference systems and gradient-boosting decision trees to optimize parameters, improving the quality and reliability of assembled products using pin-in-paste technology.

Machine learning models were utilized in Qi and Tang (2018) to estimate slope stability, employing different datasets and techniques like logistic regression, decision trees, RF, GBM, SVM, and MLP. SVM exhibited the best performance in terms of AUC and precision, with the cohesion variable showing the most significant influence on slope stability. The SVM was also used by Li et al. (2021) to classify and cluster cities in China for analyzing energy efficiency associated with low-carbon technologies. Customer retention is challenging for many companies, with LR, SVM, RF, and DT used for churn prediction in Al-Mashraie et al. (2020). Logistic Regression demonstrated the highest prediction accuracy. Chou et al. (2020) used the SVM algorithm for monitoring water condensation temperature profiles in high-pressure hose curing processes. Santos et al. (2023) presented a novel approach for project control analysis, applying machine learning models to evaluate the added value associated with the deviations analysis , times, and predicted costs. The SHAP methodology allowed for identifying key relationships between different tasks and desired outcomes.

3. Background information

3.1. Random forest

The RF algorithm was proposed by Breiman (2001), is composed of a combination of simple decision trees designed to predict or classify a response variable Y from a set of predictor variables $\boldsymbol{X} = (X_1, ..., X_r)^T$. Decision trees produce results with high variance. Breiman (1996) introduced the bagging technique to overcome this problem by utilizing the bootstrap sampling process. The algorithm

determines regions in which the predictor variable space is partitioned. The regions R_k are constructed to minimize some classification errors. The methodology involves selecting a variable X_k and a cutoff point t, such that $R_u(i,t) = \boldsymbol{X} : X_k < t$ and $R_p(i,t) = \boldsymbol{X} : X_k \leq t$, i.e., partitioning the predictor variable space associated with the pair (i,t) that minimizes the classification error. This approach involves generating multiple training sets so that the final classifier considers the voting of the trees, with the majority class winning among the predicted classes. This process is repeated until a stopping criterion is met. The model's performance is associated with the number and depth of trees in the forest, etc.

Cross-entropy and Gini index are commonly used criteria for choosing the variable that forms the root node of each tree (Probst and Boulesteix, 2018). Shannon entropy is a measure of uncertainty that calculates a score to gauge the purity of a dataset, as shown in equation (1), where $P(X = x)$ is the probability of x belonging to a certain set (Batina et al., 2011). Joint entropy between two variables X and Y is represented by equation (2). A higher entropy value for a particular variable indicates a stronger association with the response variable (Wang et al., 2020).

$$H(X) = -\int_{x \in X} P(X = x) \cdot \log(P(X = x)), \tag{1}$$

$$H(X,Y) = -\int_{x \in X, y \in Y} P(X = x, Y = y) \cdot \log(P(X = x, Y = y)) \tag{2}$$

3.2. Gradient boosting decision tree

The concept of boosting is associated with the goal of reducing bias and variance in machine learning models. The GBDT, proposed by Friedman (2001), combines weak classifiers, i.e., those with high error rates, such as shallow decision trees, to obtain a strong classifier with low error rates.

The trees are generated sequentially from a single training dataset with different observation probability weights. Observations misclassified in one tree receive higher weights for selection in the subsequent tree, leading the second tree to perform better in cases where the first tree struggled. The purpose of this process is to focus attention on cases where classification is more challenging. At the end of the training process, classifiers are weighted based on error rates, and the final classifier is obtained through majority voting (Friedman, 2002).

Hastie et al. (2009) defined that the models are adjusted by minimizing a loss function over the training data at each iteration, as shown in equation (3).

$$\arg\min_{(\beta_k, \phi_k)_k^N} \sum_{k=i}^{N} H(y_i, g_{m-1}(x_i) + T(x_i, \beta_i)), \tag{3}$$

where $T(x, \beta)$ is the tree in GBDT, $g_m(x)$ is the sum of the trees, the value of ϕ_k in the context of classification is based on the estimate given by the modal class, $\beta = \{\phi_k, R_k\}$ are the tree parameters. The iterative optimization in equation (3) is based on the Gradient Descent technique (Friedman et al., 2000).

3.3. Light gradient boosting machine

LGBM is considered a more sophisticated version of GBDT. The model proposed by Ke et al. (2017) incorporates two new methodologies: Exclusive Feature Bundling (EFB) and Gradient-based One Side Sampling (GOSS). EFB effectively reduces the number of features without compromising the split point accuracy. GOSS selects data instances with higher gradients to estimate information gain and maintain accuracy.

Typically, for Boosting models, information gain is measured by variation after the split. Training instances are ranked based on the absolute values of their gradients in descending order. Ke et al. (2017) asserted that information gain can be expressed as shown in equation (4).

LGBM uses the Leaf-wise tree growth algorithm, denoted as Leaf-wise. This method allows trees to converge more quickly but increases the risk of overfitting. One of the most important parameters in LGBM is num-leaves, which controls the model's complexity. It defines the number of leaves a weak learner can have and is closely related to the tree depth.

$$\hat{V}_j(d) = \frac{1}{n}\left(\frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b} \sum_{x_i \in B_l} g_i)^2}{n_l^j} + \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b} \sum_{x_i \in B_r} g_i)^2}{n_r^j} \right), \tag{4}$$

where $A_l = x_i \in A : x_{ij} \leq d$, $A_r = x_i \in A : x_{ij} \geq d$, $B_l = x_i \in B : x_{ij} \leq d$, $B_l = x_i \in B : x_{ij} \geq d$, and normalization coefficient $\frac{(1-a)}{b}$.

3.4. Logistic regression

The LR is a statistical technique that allows us to estimate the probability associated with the occurrence of a specific event and is commonly used in the context of classification (Qi and Tang, 2018). The model can be written as shown in equation (5), in the binary case, modeling the probability of $p = \mathbb{P}(\boldsymbol{Y}_i = 1 \mid \boldsymbol{X} = \boldsymbol{x_i})$.

$$\mathbb{P}(\boldsymbol{Y}_i = 1 \mid \boldsymbol{X} = \boldsymbol{x_i}) = \frac{e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}. \tag{5}$$

The vector $\boldsymbol{x}_i$ corresponds to the features of the observations, and $\boldsymbol{\beta}$ are the coefficients of LR that are adjusted by maximum likelihood, as shown in equation (6) (Hastie et al., 2009).

$$l(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N} \boldsymbol{y}_i \frac{e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}} - (1 - \boldsymbol{y}_i) \log(1 - \frac{e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^T \boldsymbol{x}_i}}). \tag{6}$$

The maximum likelihood estimator for $\boldsymbol{\beta}$ is the one that maximizes the log-likelihood function $l(\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y})$. Typically, the Newton-Raphson algorithm estimates the values of $\boldsymbol{\beta}$ (Friedman et al., 2000).

3.5. Support vector machine

The SVM is a powerful and flexible class of algorithms commonly used for classification and regression, introduced by Cortes et al. (1995). The SVM is an algorithm that can be used in classification and regression. For the classification problem, the algorithm seeks the best-separating hyperplane between classes. Points on the boundaries of these classes are called support vectors. Points of one course on the other side of the separating hyperplane are weighted with low weight to reduce their influence.

The margin is the smallest distance between the hyperplane and the point x_i. The maximum margin is obtained by solving equation (7) using Lagrange multipliers.

$$\arg\max_{\alpha,\boldsymbol{\beta}} \{ \frac{1}{\|\boldsymbol{\beta}\|} \min[y_i(\alpha + \boldsymbol{\beta}^T \boldsymbol{X}_i)] \}, \tag{7}$$

where $\alpha + \boldsymbol{\beta}^T \boldsymbol{X}_i$ denotes the prediction values, $\boldsymbol{\beta} = (\beta_1, ..., \beta_k)^T$ corresponds to the coefficients, and $\boldsymbol{X} = \boldsymbol{X}_1, ..., \boldsymbol{X}_k$ is the matrix with n observations of variables associated with attributes $y_1, ..., y_k \in \{-1, 1\}$. $\boldsymbol{X}_i$ is classified based on the sign of $\alpha + \boldsymbol{\beta}^T \boldsymbol{X}_i$, if the sign is positive, $\boldsymbol{X}_i$ is classified as $y = 1$, if negative, as $y = -1$ (Hastie et al., 2009). Misclassified instances are penalized as the distance from the hyperplane increases (Kecman, 2005).

For nonlinear problems, a kernel trick is used to map input data into a higher-dimensional space so that separating hyperplanes can be constructed in that space region. The most commonly used kernels in practice include linear, polynomial, radial, and hyperbolic tangent kernels (Hastie et al., 2009).

3.6. Multi-layer perceptron

The MLP is a model that belongs to a class of neural networks known as feed-forward networks, which connect neurons in one layer directly to the next layer. The input layer receives the data, the hidden layers are responsible for processing, and the output layer classifies the features. MLP can handle both linearly separable and non-linearly separable data. Neurons in the hidden layer are transformed in each iterative process using some differentiable function $\phi(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$, called the activation function. Commonly used functions in the context of classification are sigmoid, ReLU, hyperbolic tangent, and softmax.

The approximation function for an MLP with one hidden layer can be written as in equation (8), with n_0 input nodes, one hidden layer with n_1 neurons, w_{n_0} synaptic weights for neuron i, bias b_i, and ϕ_i synaptic weights for the output layer (Haykin, 2008).

$$G(\boldsymbol{x}) = \sum_{j=1}^{n_1} \alpha_i \phi(\sum_{j=1}^{n_0} w_{ij} x_i + b_i). \tag{8}$$

The commonly used loss function for classification is cross-entropy, which is minimized by updating some optimization algorithms, with the most commonly used ones being Stochastic Gradient Descent (SGD) (Amari, 1993), Adam (Kingma and Lei Ba, 2015), and Limited-memory BFGS (Mokhtari and Ribeiro, 2015).

3.7. Local Interpretable Model-Agnostic Explanations

The scientific community is increasingly interested in the explainability of machine learning algorithm results. A common problem is identifying the contribution of each variable to the model's response value. The LIME Slack et al. (2020) is a technique that allows the interpretation of individual predictions from black-box machine learning models. The algorithm involves training an interpretable model based on a dataset with samples and predictions from the black-box model. The trained model approximates the model's predictions locally (Alvarez-Melis and Jaakkola, 2018).

In this work, the decision tree and logistic regression can represent the interpretable scheme. Equation (9) represents the local surrogate models; it's an explanation model. For each observation x, the model is represented by f that minimizes a loss function L, which measures how close the explanation is to the prediction of the original model g. The set F corresponds to the family of interpretable models, and Ω is the model complexity, while π is a measure of proximity to the neighborhood around the instance x (Ribeiro et al., 2016).

$$\phi(x) = \arg\min_{f \in F} L(g, f, \pi_x) + \Omega(f)\ . \tag{9}$$

3.8. SHapley Additive exPlanations

The interpretation of complex machine learning models with SHAP values is notably one of the most widely used methods. This technique is based on game theory; essentially, the SHAP method involves distributing the total gains to the players, and the value of the feature is given by its contribution to the payment (Štrumbelj and Kononenko, 2014). A coalition game can be defined as a set N of players, where cooperation between players consists of coalitions represented by the subset S of N. To compute these feature contributions, a player's Shapley value can be defined as per equation (10).

$$\hat{\phi}_i(h_x) = \sum_{S \subseteq \{1,...,N\} \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M} \cdot |h_x(S \cup \{i\}) - h_x(S)|, \tag{10}$$

where S is a subset of features used in the model, M is the total number of possible feature input orders, x is the feature values vector, and the function h_x maps different coalitions among players to predict the payment values of the set S.

The SHAP method proposed by (Lundberg and Lee, 2017) incorporates interpretation methods for model predictions based on Shapley values. The Shapley value through an additive feature attribution method, which calculates the impact of each feature on the prediction, as shown in equation (11).

$$\rho_j(\acute{z}) = \phi_0 + \sum_{i=j}^{N} \phi_j \acute{z_j}, \tag{11}$$

where ρ is the explanation model, $\acute{z}$ is the coalition vector, N is the coalition size, and ϕ_j are Shapley values for a feature j. According to Santos et al. (2023), the SHAP method satisfies the properties of efficiency, symmetry, null play, and additivity. It provides both local and global interpretations, including feature importance, feature dependence, interactions, etc.

3.9. Genetic algorithm

Evolutionary algorithms are inspired by Darwinian principle of species evolution. They are probabilistic algorithms that provide solutions in each interactive process, eliminating weaker solutions while considering stronger ones in subsequent evaluations. The GA is an evolutionary algorithm's type used in optimization and general search problems (Katoch et al., 2021). The concepts of generation, chromosomes, fitness function, crossover, mutation, and elitism are fundamental elements of the GA. A generation is characterized by an interaction containing a population of chromosomes, representing solutions encoded as a bit array. A set of variables characterizes the genes of these chromosomes. The selection of chromosomes with the best characteristics is associated with the fitness function (Carvalho et al., 2023). Crossover involves selecting two chromosomes and generating a new chromosome with a different configuration. The mutation concept diversifies individuals in the population so that the chromosomes generated in the crossover process are not stuck in local minima. This operator randomly changes one chromosome's genes with a low probability (Katoch et al., 2021). Elitism involves selecting the best individuals from each generation, and the selection of parent chromosomes for the crossover process is associated with a probability distribution with individuals having higher fitness in the generation.

The GASearchCV method was implemented in Python in the sklearn-genetic-op package by Ruan et al. (2023), as an alternative for estimating parameters that maximize various model evaluation metrics such as accuracy, recall, precision, etc. Figure (1) represents the flowchart of this process. This methodology involves: selecting random sets of hyperparameters based on the defined search space and a fixed population size; fitting a model for each set of hyperparameters and calculating the fitness function value for each cross-validation fold; evaluating candidates according to the fitness function; creating new generations, i.e., new sets of hyperparameters by combining the last generation with different approaches; these processes are repeated until the number of generations is reached or another stopping criterion is achieved (Oyedele et al., 2023). The best hyperparameters are selected based on the set with the best cross-validation score.

4. Proposed framework

The work aims to assess the impact of each process variable on the performance of machine learning models for predicting tire performance. The study also proposes a method to determine the optimal values of each process variable to ensure the manufacturing of tires conforming to manufacturing quality standards. The models studied included MLP, SVM, RF, GBM, LGBM, and LR. The GASearchCV optimization method was used to estimate the parameters of each model, except for the LR model. The Hold-out methodwas used to split the data into development and validation sets based on a predefined data ratio (Raschka, 2018).

The evaluation of the performance of the fitted models was based on metrics derived from the confusion matrix, where the real class indexes one dimension, and the other is indexed by the predicted class of the classifier (Deng et al., 2016). The accuracy corresponds to the proportion of correct predictions made by the model based on (12). Precision, expressed by equation (13), is defined as the number of true positives (TP) divided by the number of true positives plus the number of false positives (FP). The metric recall, also known as sensitivity, according to equation (14), is defined as the number of true positives divided by the number of true positives plus the number of false negatives (FN). Specificity, expressed by equation (15), corresponds to the number of true negatives (TN) divided by the number of true negatives plus the number of false positives. The F_1-score, according to equation (16), is the harmonic mean of precision and recall. The area under the receiver operating characteristic curve (AUC-ROC), defined by the x-axis containing false positive rate information and the y-axis containing true positive rate information, was used as the evaluation criterion.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{12}$$

$$Precision = \frac{TP}{TP + FP}, \tag{13}$$

$$Recall = \frac{TP}{TP + FN}, \tag{14}$$

$$Specificity = \frac{TN}{TN + FP}, \tag{15}$$

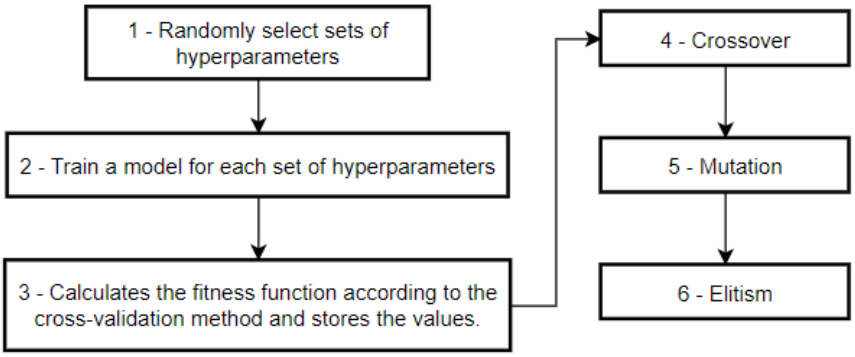


Figure 1: Genetic Algorithm SearchCV optimization

$$F_1 score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \tag{16}$$

To assess the models' generalization ability, the Leave-One-Out Cross-Validation (LOOCV) method was employed. The model is trained on a subset of data points and tested with the remaining data. This process is repeated n times, where n corresponds to the number of cross folds. LOOCV involves a sample of size $n - 1$ observations for model training and tests the observation left out to the training, making it suitable for small datasets (Wong, 2015).

For the LR model specifically, the Bayesian Information Criterion (BIC) was adopted. Also known as Schwarz Information Criterion, BIC is a model selection methodology that allows the model to involve the minimum number of estimated predictor variables for inferring the response variable without losing information (Hastie et al., 2009). The method considers a set $\mathcal{M}_r$ of candidate models, where r denotes the number of models, each associated with a prior probability $p(\mathcal{M}_l)$ with $\mathcal{M}_l \in \mathcal{M}r$. The BIC criterion is based on the Bayes factor, as shown in equation (17), where $\mathbf{x}$ is a set of n observations. When $\mathcal{B}ji(\mathbf{x}) > 1$, it indicates that $\mathcal{M}_j$ is the candidate with the most significant contribution to inferring the data.

$$\mathcal{B}_{ji}(\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M}_j)}{p(\mathbf{x}|\mathcal{M}_i)}. \tag{17}$$

The method selects the model where the posterior probability is maximum. The problem is to maximize equation (18), where $f(\boldsymbol{x}|\boldsymbol{\theta_l})$ is the likelihood function, and $\pi_l(\boldsymbol{\theta_l})$ is the prior distribution of parameters $\theta_l \in \Omega$ under $\mathcal{M}_l$.

$$p(\mathbf{x}|\mathcal{M}_l) = \int_{\Omega_l} f(\boldsymbol{x}|\boldsymbol{\theta_l})\pi_l(\boldsymbol{\theta_l})d\boldsymbol{\theta_l}. \tag{18}$$

The method selects the model where the posterior probability is maximum. The problem is to maximize equation (18), where $f(\boldsymbol{x}|\boldsymbol{\theta_l})$ is the likelihood function, and $\pi_l(\boldsymbol{\theta_l})$ is the prior distribution of parameters $\theta_l \in \Omega$ under $\mathcal{M}_l$.

$$BIC = -2\log f(\boldsymbol{x}|\hat{\boldsymbol{\theta}}) + k\log n. \tag{19}$$

The performance results of the RF, GBM, LGBM, SVM, MLP, and LR models were compared using the statistical hypothesis test McNemar's test (Nadeau and Bengio, 2003). The test was employed to analyze the statistical significance of differences in classifier performance on the validation data. The method is based on the Chi-square test applied to a 2x2 contingency table to assess the goodness of fit. The test statistic can be written as in equation (20), where n_{ji} indicates the number of tires misclassified by method j but correctly classified by method i, and n_{ij} indicates the number of tires misclassified by method i but not by method j. This ratio follows a Chi-square distribution with one degree of freedom. The null hypothesis of the test corresponds to similarity in the error proportions of the models, while the alternative hypothesis suggests that the models' performances are different (Wu, 2022).

$$T = \frac{(|n_{ji} - n_{ij}| - 1)^2}{n_{ji} + n_{ij}} \sim \chi^2(1). \tag{20}$$

In the first stage of this study, data preparation and splitting into development and validation sets were performed. In the second stage, descriptive and inferential data analysis was conducted using the Kruskal-Wallis test to compare variable behavior based on tire performance (Kruskal and Wallis, 1952). In the third stage, the study assessed the performance of each model based on parameter tuning through cross-validation, analyzed the models' generalization capabilities, and evaluated the results on the validation set. The fourth stage focused on analyzing the global and local explainability of the models using SHAP and LIME methods. In the fifth stage, optimal values on the tire performance classification boundary were analyzed, considering the LR model. In this stage, the HiGHS method was adopted for optimizing optimal values; it is an algorithm for determining the optimal solution of a linear programming model (Huangfu and Hall, 2018). Figure 2 summarizes the development process of the study.

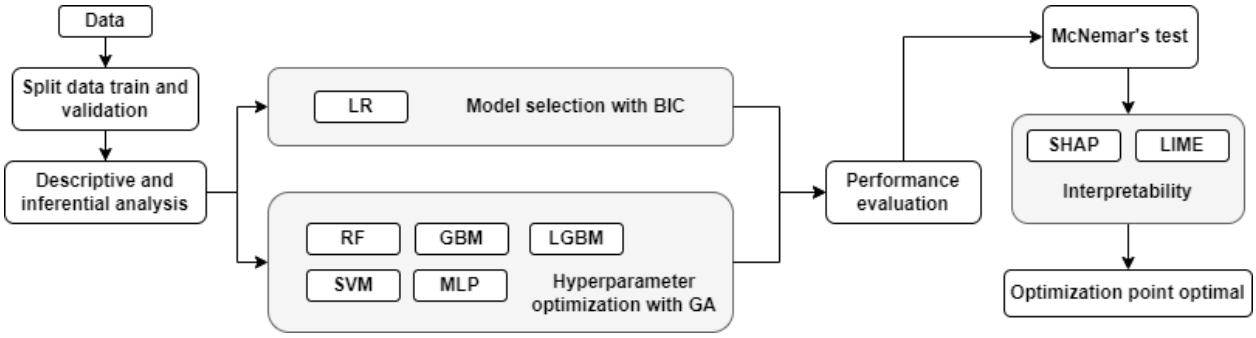


Figure 2: Proposed Interpretability framework

The study was conducted using the programming languages Python v3.9 and R v4.0, utilizing open-source tools such as Jupyter Notebook, Visual Studio Code, and R Studio. The development work utilized a machine with an Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz, Intel Core i7 7th Gen processor, 8 GB of RAM, and an NVIDIA GEFORCE 2 GB graphics card. The MLP, SVM, RF, GBM, and LGBM models were trained using the *scikit-learn* and *sklearn-genetic-op* libraries for hyperparameter optimization. For explainability analysis, the *shap* and *lime* libraries were used. The LR model was developed in the R language using the *glm* method. The *scipy.optimize* library in Python was used to solve programming problems.

5. Case study

The development of this work involves analyzing data from a study conducted in a multinational tire industry that manufactures tires for passenger cars and trucks. Tire performance is a key indicator for the factory and has a direct impact on the company's financial balance. The collected data come from the tire uniformity system, which consists of a set of dynamic mechanical properties subject to standards and measurement conditions of the factory's intelligent manufacturing process (Liu et al., 2023). Currently, tire and vehicle manufacturers are adopting new approaches in tire uniformity processes to identify low-performing tires so they are not sold in the market.

Tires are inspected for irregular size, mass, and forces under specific pressure, load, and speed conditions. The forces acting on the tire are divided into three axes: radial, lateral, and tangential. The radial axis supports the vehicle's weight, i.e., from the tire's center towards the tread (Du et al., 2020). The lateral axis is parallel to the vehicle tire's mounting axis. The axis the tire moves on refers to the tangential axis. The study includes variables such as radial force variation (RFV); radial force 2nd harmonic (H2RFV); lateral force variation (LFV); radial runout (RRO); plysteer (PLY); cap splice (CAPSPLICE); CONICITY. The response variable in this study is tire performance, i.e., tires conforming (C) and non-conforming (NC) to the quality process. The dataset contains a sample of 107 observations.

The change in radial force describes the RFV as the tire rotates under load. LFV represents the variation in lateral force that acts to the left or right along the tire's axis. PLY describes the lateral force the tire produces associated with carcass asymmetry. CONICITY is characterized by the tire's tendency to roll like a cone, consequently affecting the vehicle's steering performance. RRO describes the tire's eccentricity. Force variation measurements can be analyzed according to their harmonic components by applying a Fourier transform. H2RFV represents the 2nd harmonic of RFV, describing the magnitude of the force variation that pulses the vehicle per revolution, in this case, exerting two pulses per revolution. CAPSPLICE is the cap splice. Figure 3 illustrates the tire uniformity measurement process, which involves rotating the tire while applying a load on a rotating drum together with force sensors mounted alongside the tire's axis (Cho et al., 2012).

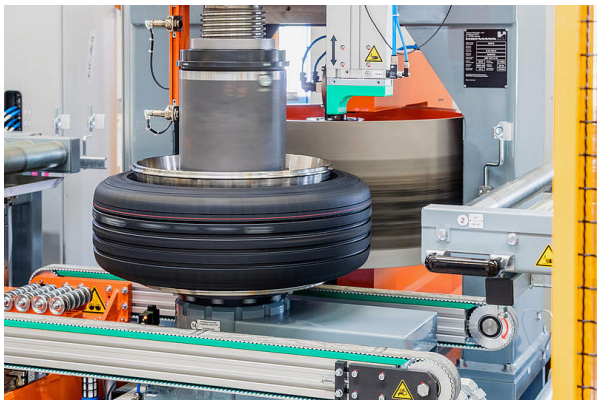


Figure 3: Equipment with sensors that measure the characteristics of the tire.

6. Results and Discussion

6.1. Descriptive and inferential statistics

The proportion for the development sample was 75%, with 81 samples, and 25% with 26 samples for validation. In the development set, there were 45 conforming (C) and 36 non-conforming (NC) tires, while in the validation set, there were 16 C and 10 NC tires.

Table 1
Descriptive statistics tire C

	Statistics						
Variables	mean	std	min	25%	50%	75%	max
RRO	0.68	0.14	0.42	0.58	0.67	0.78	0.97
CONICITY	7.53	1.43	4.50	6.78	7.60	8.33	9.90
RFV	47.42	10.15	31.00	36.75	49.00	55.00	67.00
H2RFV	15.00	7.10	2.00	7.75	16.00	21.25	25.00
CAPSPLICE	0.28	0.07	0.17	0.23	0.27	0.32	0.42
PLY	51.18	3.21	40.40	49.50	51.55	53.00	56.50
LFV	12.58	2.92	8.30	10.55	12.20	14.83	19.00

Table 2
Descriptive statistics tire NC

	Statistics						
Variables	mean	std	min	25%	50%	75%	max
RRO	0.96	0.25	0.46	0.79	0.99	1.11	1.48
CONICITY	9.94	1.77	6.50	8.80	10.30	11.00	13.20
RFV	65.51	16.28	32.00	55.00	66.00	78.00	101.00
H2RFV	23.11	9.28	4.00	16.00	22.00	30.00	46.00
CAPSPLICE	0.33	0.15	0.15	0.23	0.27	0.39	0.75
PLY	50.27	2.44	42.40	49.40	50.60	51.90	53.80
LFV	12.54	3.13	7.00	10.70	11.90	14.70	20.90

The main statistics of the development set's predictor variables for C tires are summarized in Table 1, and for NC tires, this information is represented in Table 2. It can be observed that the means and medians of the CAPSPLICE, PLY, and LFV variables for both C and NC tires are close, indicating that these variables might not have discriminative power for tire performance. The third quartile value of the RRO variable for C tires is lower than the second quartile value for NC tires, indicating an overlap in the distributions for tire performance classification. The same occurs for the CONICITY variable.

The Kruskal-Wallis test was used to compare the distribution of each predictor variable in the C and NC tire groups. This test is a non-parametric alternative to analysis of variance (ANOVA) and can be employed when there is no justification for assuming normality of the data Aghighi et al. (2021). The variables RRO, CONICITY, RFV, and H2RFV have p-values less than 0.1%, indicating significant evidence, at a 5% significance level, to reject the hypothesis of similarity in the distribution of these variables observed in the two groups. For the variables CAPSPLICE, PLY, and LFV, the p-values are 28.0%, 10.5%, and 98.9%, respectively. Therefore, at the 5% significance level, there is no statistically significant evidence to claim differences between the C and NC tire groups for the CAPSPLICE, PLY, and LFV variables.

6.2. Performance evaluation

To optimize the models' hyperparameters using GA, a population size of 10 with 5 generations and a crossover probability of 0.8 were considered. The mutation probability was set to 0.1, and the fitness function was defined by accuracy. LOOCV was employed for cross-validation due to the limited amount of data.

For the GBDT and LGBM boosting models, the search space for the learning rate was set between [0.001, 0.01]. The search spaces for the GBDT parameters were as follows: [4, 8] for the maximum tree depth; [5, 20] for the number of estimators; [5, 25] for the minimum samples required to split an internal node; [2, 8] for the minimum samples required to be a leaf node; 'deviance' loss function; 'friedman_mse' and 'squared_error' criteria to measure split quality; 'log2' and 'sqrt' were considered for selecting the number of features when searching for the best split.

For the LGBM model, the boosting type considered was 'goss', and the search space for the minimum samples in a leaf was [2, 10], and [5, 20] for the number of leaves in a tree. The maximum tree depth was in the range [5, 40]. The search space for L1 regularization term was [0.01, 0.5], and the maximum number of leaves for learning was [5, 20]. The number of estimators ranged from [5, 40]. The search spaces for SVM parameters were: $\gamma \in [0.01; 10]$; $C \in [0.1, 10]$; 'rbf', 'poly', and 'sigmoid' for possible kernel scenarios.

For MLP, the activation functions considered were: 'identity', 'logistic', 'tanh', and 'relu'. The tuple representing the number of neurons in the hidden layer was (100,). The

Table 3
Accuracy and hyperparameters cross-validation

Models	Accuracy Cross-validation	Hyperparameters	Values
RF	**0.98**	criterion; max depth max features; min samples leaf; min samples split; min estimators	'entropy'; 6 'sqrt'; 3 27; 20
GBDT	0.86	criterion; learning rate; max depth max features; loss; n estimators min samples leaf; min samples split	'mse'; 0.01; 8 'log2'; 'deviance'; 16 2; 7
LGBM	0.93	max depth; learning rate; boosting type min child samples; n estimators; num leaves objective; reg alpha	5; 0.01; 'goss' 8; 34; 17 'binary'; 0.28
SVM	0.88	C; gamma; kernel	2.2; 8.7; 'poly'
MLP	0.93	alpha; hidden layer sizes learning rate; learning rate init activation; solver	0.08; (100,) 'constant'; 0.02 'relu'; 'adam'

initial learning rate was in the range [0.001, 0.1], and the alpha value ranged from [0.001, 0.1]. For the 'solver' parameter, the methods 'adam' and 'sgd' were considered, and the weight updates were handled using 'constant', 'invscaling', and 'adaptive' methods.

For the RF model, the search intervals for the parameters were: [5, 20] for the number of trees in the forest; [4, 8] for the maximum tree depth; 'gini' and 'entropy' to measure split quality; 'sqrt' and 'log2' for selecting the number of features in the sample; [8, 30] for the minimum samples required to split an internal node; [2, 10] for the minimum samples required to be a leaf node. Table 3 contains the accuracy values in the cross-validation process and the final parameters for each model. It is noteworthy that the models achieved an accuracy performance of around 90%. The RF model resulted in the best performance in the cross-validation process.

The results of the LR model coefficients' metrics are summarized in Table 4. The values of $\beta_0, \beta_1, \beta_2$, and β_3 represent the intercept and the coefficients of the variables selected by the BIC method: RRO, CONICITY, and H2RFV, respectively. The parameter estimates are associated with a significance level of 5%. The coefficients associated with the variables RRO, CONICITY, and H2RFV are positive. Therefore, it can be concluded that as the values of these variables increase, the probability of a tire being classified as NC also increases. Figure 5 represents the AUC-ROC curve obtained by the model on the development set. As shown, the optimal cutoff probability for classification is 0.41, considering the lowest false positive rate. This value was used to calculate the model's performance metrics.

Table 5 contains the evaluation metrics for the development and validation data. The best performances were achieved by the RF and LR models in the validation set. It is noticeable that the SVM failed to generalize input patterns. The GBDT and LGBM models had similar results in the development set, while in the validation set, the GBDT and LR models achieved the best recall results. For precision metric, the LGBM, RF, and LR models had the best results. Regarding specificity metric, the RF and LGBM models achieved the best results. The MLP neural network showed good performance in both datasets, demonstrating the predictive power of the network. The LR model's performance metrics in the development and validation sets were closer compared to the metrics obtained by the RF model, indicating the method's robustness and generalization ability for data inference.

Table 4
LR model result

Parameter estimates				
Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
β_0	-47.08	15.01	-3.14	0.002
β_1	17.91	6.00	2.99	0.003
β_2	3.01	0.99	3.05	0.002
β_3	0.36	0.13	2.85	0.004

The figure 5 depicts the p-values of the McNemar statistical test for each comparison among the model inferences on the validation set. The test results at a significance level of 5% suggest the non-rejection of the null hypothesis for all comparisons, indicating no statistically significant evidence that the models exhibit different performances.

Table 5
Metrics on the validation set

Models	Dataset	Metrics					
		Accuracy	Precision	Recall	Specificity	$F_1 score$	AUC-ROC
RF	development	1.00	1.00	1.00	1.00	1.00	1.00
	validation	0.92	**1.00**	0.88	**1.00**	0.93	0.94
GBDT	development	0.99	0.98	1.00	0.97	0.99	0.99
	validation	0.92	0.94	**0.94**	0.90	0.94	0.92
LGBM	development	0.98	0.96	1.00	0.94	0.98	0.97
	validation	0.88	**1.00**	0.81	**1.00**	0.90	0.91
SVM	development	1.00	1.00	1.00	1.00	1.00	1.00
	validation	0.77	0.92	0.69	0.90	0.79	0.79
MLP	development	0.98	0.98	0.98	0.97	0.98	0.98
	validation	0.88	0.93	0.88	0.90	0.90	0.89
LR	development	0.96	0.96	0.98	1.00	0.97	0.99
	validation	**0.96**	**1.00**	**0.94**	0.94	**0.97**	**0.97**

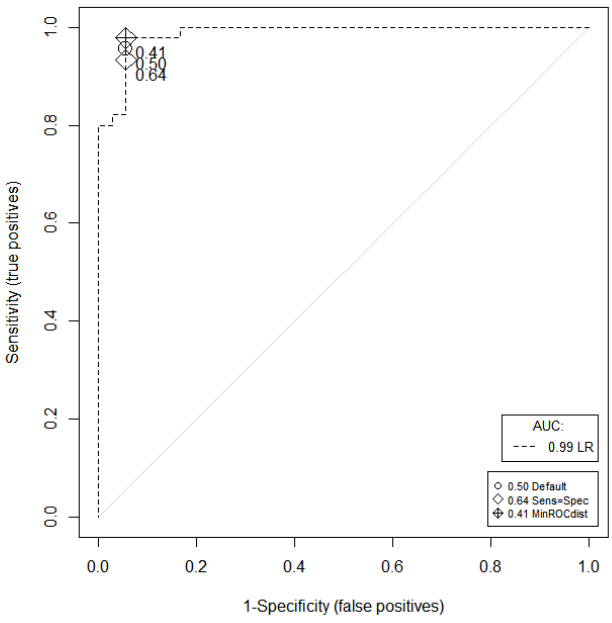


Figure 4: AUC-ROC model LR

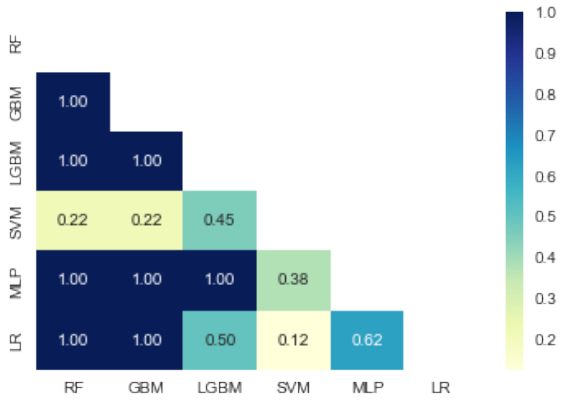


Figure 5: McNemar's test

6.3. Explainability scheme

The SHAP approach allows interpreting machine learning model predictions consistently and objectively. For each variable, it is possible to identify its impact in terms of SHAP values on the model's decision for anomaly detection; positive values indicate a positive impact on the model's prediction. Figure 6 provides a global interpretation of the results considering the RF model, where each point represents an example from the development dataset. The SHAP method is ideal for tree-based models, so the RF model was chosen due to its performance in the development and validation sets.

Variables RRO, CONICITY, RFV, and H2RFV have a higher impact on the model; high values represent a greater contribution to detecting NC tires, as observed by the color shade. For the variable PLY, it is noticed that for high values, the probability of the tire being classified as C increases.

Figure 7 represents a local interpretation by choosing two random points from the development dataset. In Figure 7(a), there is an example representing an NC tire, and in Figure 7(b), there is an example of a C tire. The gray axis corresponds to the impact of each variable on the RF model response. For the NC tire, the obtained value was 0.96, indicating a high probability of being an anomaly, and the

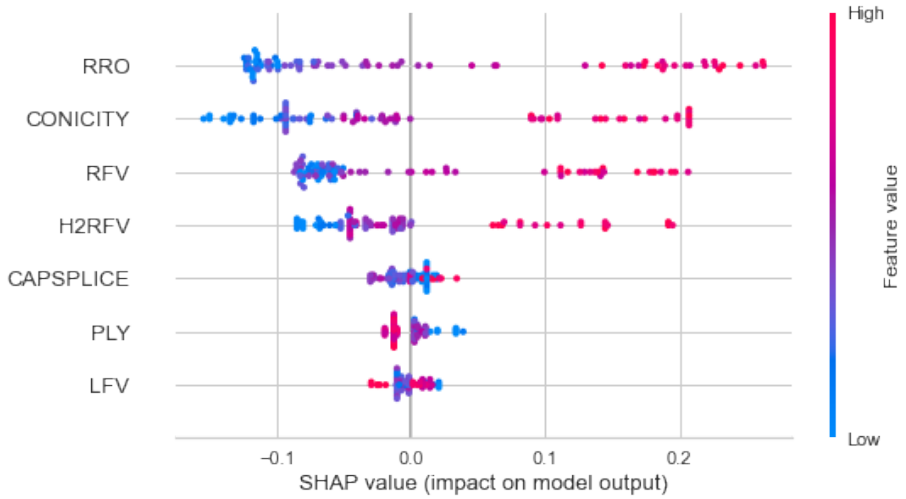
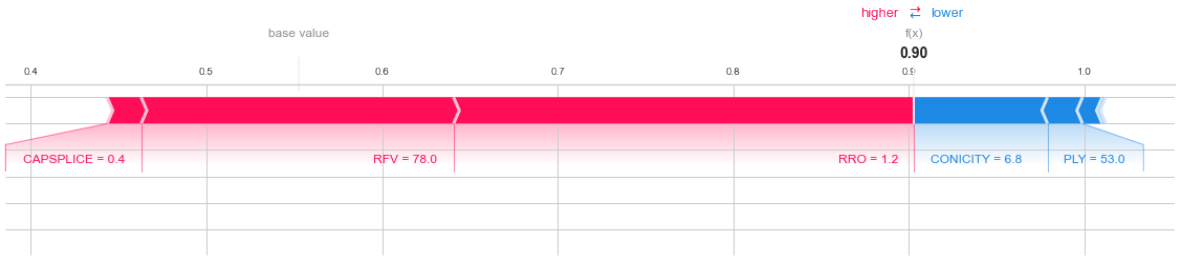
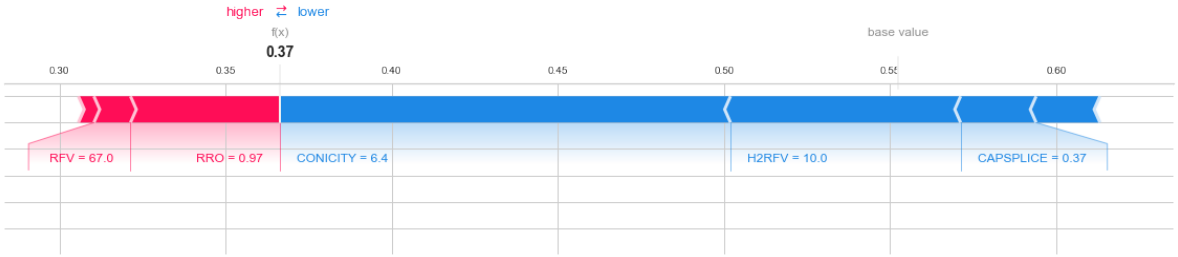


Figure 6: SHAP summary plots.



(a) Tire NC



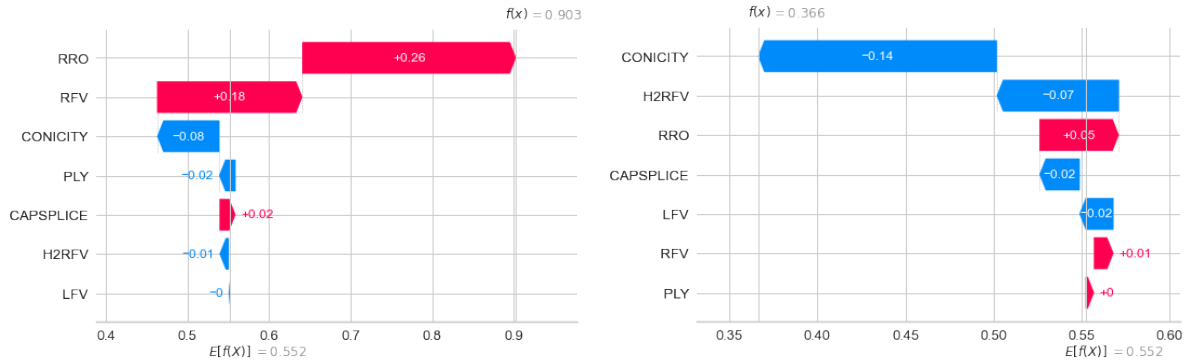
(b) Tire C

Figure 7: Force plot for tire nonconforming (a) and conforming (b).

values of the variables CAPSPLICE, RFV, and RRO that contributed to this result were 0.4, 78.0, and 1.2, respectively. For the C tire, the point's result was 0.37, indicating a low probability of being an anomaly, with low values of CONICITY, H2RFV, and CAPSPLICE contributing to this outcome.

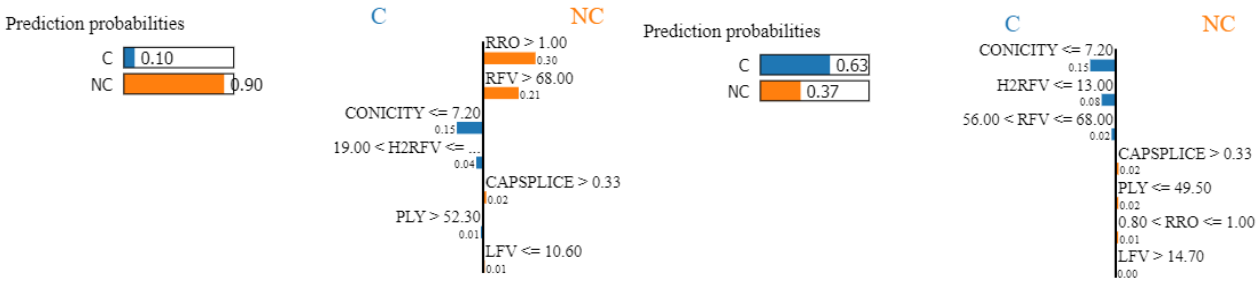
The model's result can also be interpreted with the help of the waterfall chart. Figure 8 represents the same points analyzed earlier. In Figure 8(a), the SHAP values are represented by the size of the bars. The x-axis represents the predicted values for tire classification, and the y-axis represents each predictor variable. The value of $f(x)$ is the model's predicted value, $E[f(x)]$ is the expected value of the model's output. The bottom of the chart starts with $E[f(x)]$,and then the contributions of each variable move the expected model output value, so that the sum of all SHAP values is $E[f(x)] - f(x)$. The variable RRO has the highest magnitude for predicting this observation. In Figure 8(b), it can be observed that the impact value of the CONICITY variable had the most significant contribution to the classification of the C tire.

In the SHAP dependency plots represented in Figure 10, it is possible to identify interactions between predictor variables and the SHAP value. The relationship between CONICITY and RFV variables in Figure 10(a) demonstrates that CONICITY and RRO values of 10.0 and 50.0, respectively, it contribute to the model's prediction, classifying the tire as C. In Figure 10(c), it can be observed that values



(a) Tire NC (b) Tire C

Figure 8: Waterfall plots for individual process variables's predictions.



(a) Tire NC (b) Tire C

Figure 9: Explanation with LIME method

greater than 10.0 and 1.0 for the CONICITY and RRO variables, respectively, indicate a higher contribution for the tire to be classified as NC.

The LIME method is an alternative approach for local interpretation of machine learning models. To compare it with the SHAP method, a local analysis of the same observations discussed earlier was performed. Figure 9(a) represents the local interpretation for the NC tire observation. It can be observed that the prediction value is 0.90, and the impact of each variable on the tire's performance prediction. Similar to the SHAP method, the values of the RRO and RFV variables contribute positively to increasing the probability of the tire being classified as NC. In Figure 9(b), the impact values for the C tire can be seen. Values smaller than 7.2 and 15.0 for the CONICITY and H2RFV variables, respectively, indicate a higher probability for the tire to be classified as C.

6.4. Optimization point optimal

The explainability of machine learning models is crucial for understanding the relationship between predictor variables and the model's predictions. However, diagnostics using SHAP and LIME methodologies do not allow for identifying variable values on the model's classification boundary for predicting tire performance. This is essential for anticipating the manufacturing process of tires with anomalies. In this context, a method is proposed to find optimal values for the variables that make up the model to determine the classification boundary values. For this purpose, the LR model was considered due to the results obtained in the development and validation sets and because it is a parametric model, which facilitates result interpretation.

The estimated LR model can be written according to equation (21), where X_1, X_2, X_3 represent the variables as RRO, CONICITY, and H2RFV, respectively. The values $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3)$ correspond to the estimated coefficient values, with $\hat{\beta}_0$ being the intercept.

$$\log(\frac{\hat{p}}{1-\hat{p}}) = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_3. \tag{21}$$

Equation (21) represents a plane in the R^3 space; thus, there are infinitely many solutions that satisfy the equality condition. To find a solution on the plane to maximize the values of points $x_1 \in X_1$, $x_2 \in X_2$, $x_3 \in X_3$, an optimization approach with constraints was adopted using the HiGHS method. The problem involves solving expression (22) with the constraint system (23).

$$\max_{x_1,x_2,x_3 \in \boldsymbol{x}} \hat{\boldsymbol{\beta}}^T \boldsymbol{x} \tag{22}$$

$$\left\{ \hat{\boldsymbol{\beta}}^T \boldsymbol{x} \leq \log(\frac{\hat{p}}{1-\hat{p}}) - \hat{\beta}_0 \right. . \tag{23}$$

The search bounds for $\boldsymbol{x} = (x_1, x_2, x_3)$ for each variable were set to values between the third quartile and the

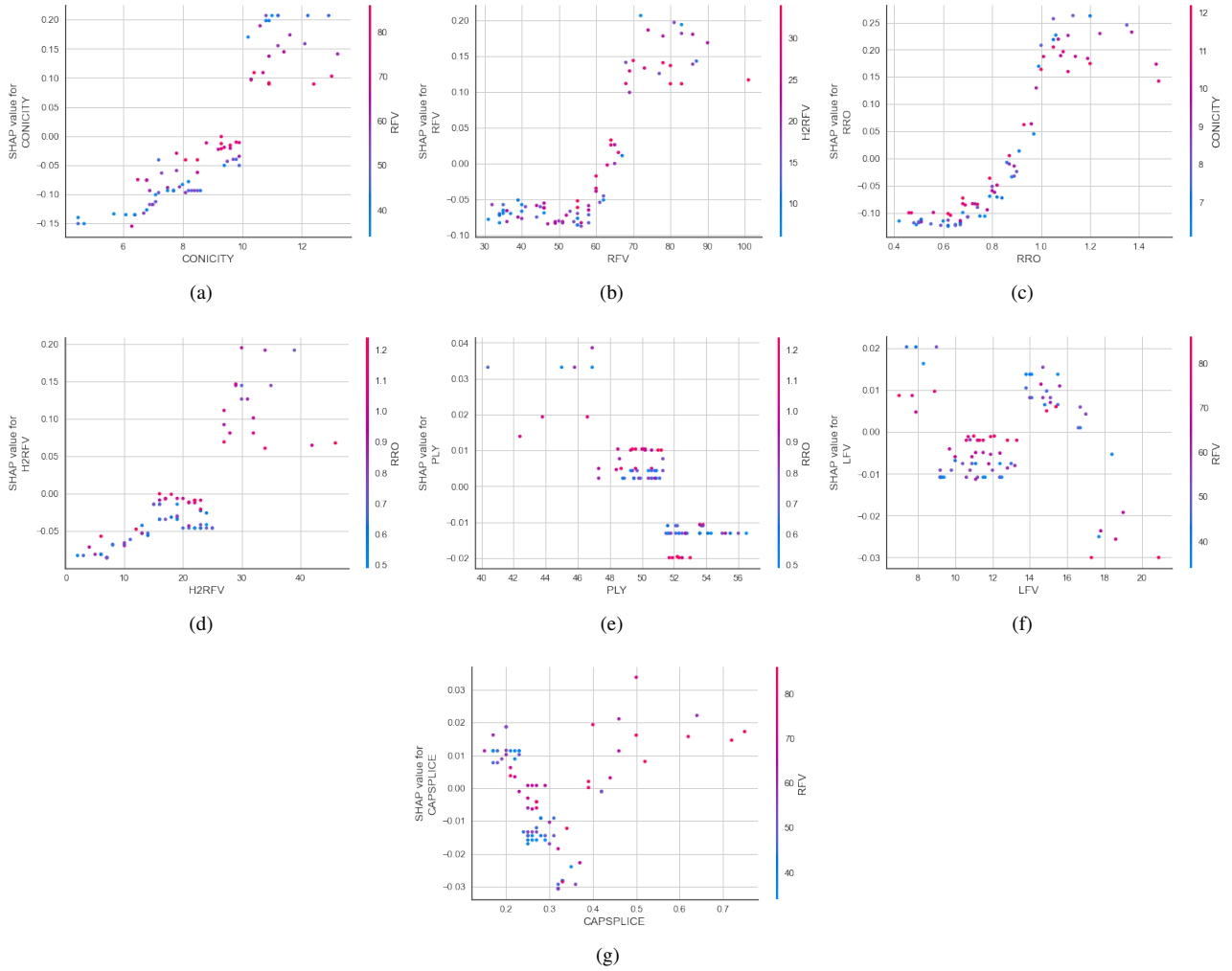


Figure 10: SHAP dependence plot

maximum value, considering the classification of tire C, as described in Table 1. In Figure 4, it was observed that the optimal probability cutoff to predict whether a tire is classified as NC or not is 0.41, and the estimated coefficients' values of the LR model are represented in Table 4.

The optimal values found to maximize RRO, CONICITY, and H2RFV were 0.78, 8.33, and 21.32 respectively. Surprisingly, for RRO and CONICITY variables, these values correspond to the third quartile of each variable. Clearly, the values found represent only one point in the plane of possible solutions associated with the LR model's outcome. The accuracy, precision, recall, specificity, F_1 score, and AUC-ROC metrics obtained based on these thresholds in the development set were, respectively: 0.83; 0.86; 0.82; 0.84; 0.83; 0.83. For the validation set, the metric values were: 0.77; 0.86; 0.75; 0.80; 0.80; and 0.78. Figure 11 represents anomaly points based on these optimal values, indicating that most tires classified as anomalies are indeed NC. This result allows the identification of potential reference values for RRO, CONICITY, and H2RFV variables in tire manufacturing in compliance with performance standards adopted by factories worldwide.

7. Conclusion

The study presented different supervised machine learning algorithms for defect detection in the context of a multinational tire factory. Algorithms with distinct structures could learn non-linear patterns in variables derived from the tire uniformity process, classifying high-performance tires and those not conforming to manufacturing standards. Predictive models for defect detection included RF, GBDT, LGBM, LR, SVM, and MLP.

The results of the models showed similar performance, as indicated by the McNemar's statistical test, demonstrating no significant difference in error proportions among the models. However, RF and LR values exhibited the best accuracy and ROC-AUC results in the validation dataset. Considering that LIME and SHAP analyses are ideal for tree-based models, the RF model was chosen for analysis to identify the importance of predictor variables for defect

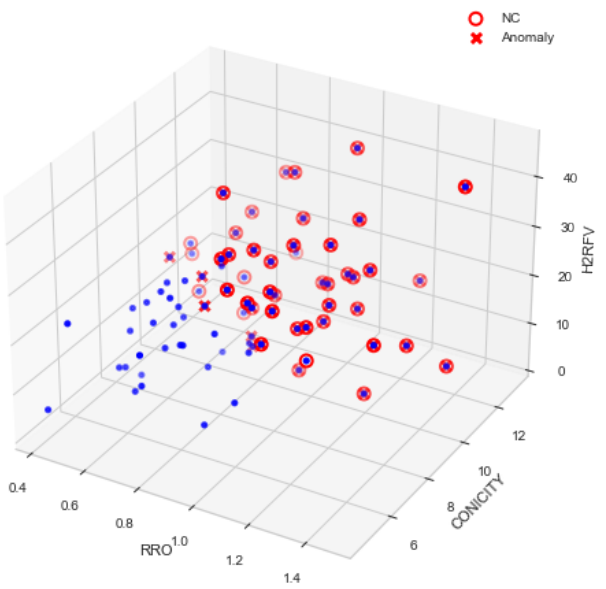


Figure 11: Anomaly points

detection. The global analysis from the SHAP method indicated that high values of RRO, CONICITY, RFV, and H2RFV positively impact the probability of a tire being classified as NC. In the local analysis using LIME and SHAP methods, the influence of each predictor variable on the final tire performance classification was identified, and both methods converged on the same conclusions. Clearly, the interpretations of the LR model results align with the SHAP and LIME analyses of the RF model.

A linear optimization approach based on the LR model's presented result coefficients was proposed to identify reference values for predictor variables in tire manufacturing compliance with manufacturing standards. The equation of the plane associated with the LR model's optimal classification point allowed the construction of a linear system with constraints. Based on this, the proposed approach maximized the predictor variable values associated with the model. This methodology allowed the determination of reference values for RRO, CONICITY, and H2RFV variables for classifying high-performance tires.

For future research, it is recommended to study reference values for other predictor variables in the tire uniformity process. Generalized linear models could be an alternative to finding associations between predictor variables. Manufacturing process monitoring is essential to ensure operational efficiency in factories. Anomaly detection is part of strategic management to guarantee product quality. In this scenario, AI systems can provide more accurate and efficient results. The results of this study are of great importance for tire manufacturing defect detection, enabling the implementation of new manufacturing policies to ensure products with better quality, longer lifespan, and reduced environmental impact as waste in nature.

Conflict of interest

The authors declare that no known competing financial interests or personal relationships could have appeared to influence the work reported in this paper.

References

Acosta, S.M., Amoroso, A.L., Sant Anna, A.M.O., Canciglieri Junior, O., 2021. Relevance vector machine with tuning based on self-adaptive differential evolution approach for predictive modelling of a chemical process. Applied Mathematical Modelling 95, 125–142. doi:10.1016/J.APM.2021.01.057.

Acosta, S.M., Marcel, R., Oliveira, A., Márcio, , Sant'anna, O., 2023. Machine learning algorithms applied to intelligent tyre manufacturing. International Journal of Computer Integrated Manufacturing 10, 1–11. URL: https://www.tandfonline.com/doi/abs/10.1080/0951192X.2023.2177734, doi:10.1080/0951192X.2023.2177734.

Aghighi, S., Niaki, S.T.A., Mehdizadeh, E., Najafi, A.A., 2021. Open-shop production scheduling with reverse flows. Computers & Industrial Engineering 153, 107077. doi:10.1016/J.CIE.2020.107077.

Al-Mashraie, M., Chung, S.H., Jeon, H.W., 2020. Customer switching behavior analysis in the telecommunication industry via push-pull-mooring framework: A machine learning approach. Computers & Industrial Engineering 144, 106476. doi:10.1016/J.CIE.2020.106476.

Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J.M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., Herrera, F., 2023. Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. Information Fusion 99, 101805. doi:10.1016/J.INFFUS.2023.101805.

Alvarez-Melis, D., Jaakkola, T.S., 2018. On the Robustness of Interpretability Methods, in: Workshop on Human Interpretability in Machine Learning. URL: https://arxiv.org/abs/1806.08049v1.

Amari, S.i., 1993. Backpropagation and stochastic gradient descent method. Neurocomputing 5, 185–196. doi:10.1016/0925-2312(93)90006-O.

Batina, L., Gierlichs, B., Prouff, E., Rivain, M., Standaert, F.X., Veyrat-Charvillon, N., 2011. Mutual information analysis: A comprehensive study. Journal of Cryptology 24, 269–291. URL: https://link.springer.com/article/10.1007/s00145-010-9084-8, doi:10.1007/S00145-010-9084-8/METRICS.

Breiman, L., 1996. Heuristics of instability and stabilization in model selection. https://doi.org/10.1214/aos/1032181158 24, 2350–2383. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-24/issue-6/Heuristics-of-instability-and-stabilization-in-model-selection/10.1214/aos/1032181158.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-24/issue-6/Heuristics-of-in, doi:10.1214/AOS/1032181158.

Breiman, L., 2001. Random Forests. Machine Learning 45, 5–32. doi:10.1023/A:1010933404324.

Cakir, M., Guvenc, M.A., Mistikoglu, S., 2021. The experimental application of popular machine learning algorithms on predictive maintenance and the design of IIoT based condition monitoring system. Computers & Industrial Engineering 151, 106948. doi:10.1016/J.CIE.2020.106948.

Carvalho, J.L., Farias, P.C., Simas Filho, E.F., 2023. Global Localization of Unmanned Ground Vehicles Using Swarm Intelligence and Evolutionary Algorithms. Journal of Intelligent and Robotic Systems: Theory and Applications 107, 1–21. URL: https://link.springer.com/article/10.1007/s10846-023-01813-6, doi:10.1007/S10846-023-01813-6/METRICS.

Carvalho, T.P., Soares, F.A., Vita, R., Francisco, R.d.P., Basto, J.P., Alcalá, S.G., 2019. A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering 137, 106024. doi:10.1016/J.CIE.2019.106024.

Cho, J.R., Lee, J.H., Jeong, K.M., Kim, K.W., 2012. Optimum design of run-flat tire insert rubber by genetic algorithm. Finite Elements in Analysis and Design 52, 60–70. doi:10.1016/J.FINEL.2011.12.006.

Chou, S.H., Chang, S., Tsai, T.R., Lin, D.K., Xia, Y., Lin, Y.S., 2020. Implementation of statistical process control framework with machine

learning on waveform profiles with no gold standard reference. Computers & Industrial Engineering 142, 106325. doi:10.1016/J.CIE.2020.106325.

Cortes, C., Vapnik, V., Saitta, L., 1995. Support-vector networks. Machine Learning 1995 20:3 20, 273–297. URL: https://link.springer.com/article/10.1007/BF00994018, doi:10.1007/BF00994018.

Demšar, J., 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research 7, 1–30.

Deng, X., Liu, Q., Deng, Y., Mahadevan, S., 2016. An improved method to construct basic probability assignment based on the confusion matrix for classification problem. Information Sciences 340-341, 250–261. doi:10.1016/J.INS.2016.01.033.

Du, M., Sun, P., Zhou, S., Huang, H., Zhu, J., 2020. A Study on the Influence of Tire Speed and Pressure on Measurement Parameters Obtained from High-Speed Tire Uniformity Testing. Vehicles 2020, Vol. 2, Pages 559-573 2, 559–573. URL: https://www.mdpi.com/2624-8921/2/3/31/htmhttps://www.mdpi.com/2624-8921/2/3/31, doi:10.3390/VEHICLES2030031.

Friedman, J., Hastie, T., Tibshirani, R., 2000. Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). https://doi.org/10.1214/aos/1016218223 28, 337–407. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-28/issue-2/Additive-logistic-regression--a-statistical-view-of-boosting-With/10.1214/aos/1016218223.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-28/issue-2/Additive-logi, doi:10.1214/AOS/1016218223.

Friedman, J.H., 2001. Greedy function approximation: A gradient boosting machine. https://doi.org/10.1214/aos/1013203451 29, 1189–1232. URL: https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.fullhttps://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-appro, doi:10.1214/AOS/1013203451.

Friedman, J.H., 2002. Stochastic gradient boosting. Computational Statistics and Data Analysis 38, 367–378. doi:10.1016/S0167-9473(01)00065-2.

Gutierrez-Gomez, L., Petry, F., Khadraoui, D., 2020. A Comparison Framework of Machine Learning Algorithms for Mixed-Type Variables Datasets: A Case Study on Tire-Performances Prediction. IEEE Access 8, 214902–214914. doi:10.1109/ACCESS.2020.3041367.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning URL: http://link.springer.com/10.1007/978-0-387-84858-7, doi:10.1007/978-0-387-84858-7.

Haykin, S., 2008. Neural Networks and Learning Machines. Pearson Prentice Hall New Jersey USA 936 pLinks 3, 906. doi:978-0131471399.

Huangfu, Q., Hall, J.A., 2018. Parallelizing the dual revised simplex method. Mathematical Programming Computation 10, 119–142. URL: https://link.springer.com/article/10.1007/s12532-017-0130-5, doi:10.1007/S12532-017-0130-5/FIGURES/5.

Katoch, S., Sumit, ., Chauhan, S., Kumar, V., 2021. A review on genetic algorithm: past, present, and future. Multimedia Tools and Applications 80, 8091–8126. URL: https://doi.org/10.1007/s11042-020-10139-6, doi:10.1007/s11042-020-10139-6.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y., 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems 30. URL: https://github.com/Microsoft/LightGBM.

Kecman, V., 2005. Support Vector Machines – An Introduction , 1–47doi:10.1007/10984697{_}1.

Kingma, D.P., Lei Ba, J., 2015. Adam: A Method for Stochastic Optimization. undefined .

Ko, D., Kang, S., Kim, H., Lee, W., Bae, Y., Park, J., 2021. Anomaly Segmentation Based on Depth Image for Quality Inspection Processes in Tire Manufacturing. Applied Sciences 2021, Vol. 11, Page 10376 11, 10376. URL: https://www.mdpi.com/2076-3417/11/21/10376/htmhttps://www.mdpi.com/2076-3417/11/21/10376, doi:10.3390/APP112110376.

Kruskal, W.H., Wallis, W.A., 1952. Use of Ranks in One-Criterion Variance Analysis. Journal of the American Statistical Association 47, 583–621. doi:10.1080/01621459.1952.10483441.

Li, W., Xu, J., Ostic, D., Yang, J., Guan, R., Zhu, L., 2021. Why low-carbon technological innovation hardly promote energy efficiency of China? – Based on spatial econometric method and machine learning. Computers & Industrial Engineering 160, 107566. doi:10.1016/J.CIE.2021.107566.

Liu, Z., Wang, F., Cai, Z., Wei, Y., Marburg, S., 2023. Stochastic analysis for in-plane dynamic responses of low-speed uniformity of tires due to geometric defects. Mechanical Systems and Signal Processing 197, 110377. doi:10.1016/J.YMSSP.2023.110377.

Lundberg, S.M., Lee, S.I., 2017. A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems 2017-Decem, 4766–4775. URL: https://arxiv.org/abs/1705.07874v2, doi:10.48550/arxiv.1705.07874.

Martinek, P., Krammer, O., 2019. Analysing machine learning techniques for predicting the hole-filling in pin-in-paste technology. Computers & Industrial Engineering 136, 187–194. doi:10.1016/J.CIE.2019.07.033.

Mokhtari, A., Ribeiro, A., 2015. Global Convergence of Online Limited Memory BFGS. Journal of Machine Learning Research 16, 3151–3181.

Nadeau, C., Bengio, Y., 2003. Inference for the generalization error. Machine Learning 52, 239–281. URL: https://link.springer.com/article/10.1023/a:1024068626366, doi:10.1023/A:1024068626366/METRICS.

Oyedele, A.A., Ajayi, A.O., Oyedele, L.O., Bello, S.A., Jimoh, K.O., 2023. Performance evaluation of deep learning and boosted trees for cryptocurrency closing price prediction. Expert Systems with Applications 213. doi:10.1016/J.ESWA.2022.119233.

Probst, P., Boulesteix, A.L., 2018. To Tune or Not to Tune the Number of Trees in Random Forest. Journal of Machine Learning Research 18, 1–18. URL: http://jmlr.org/papers/v18/17-269.html.

Qi, C., Tang, X., 2018. Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. Computers & Industrial Engineering 118, 112–122. doi:10.1016/J.CIE.2018.02.028.

Raschka, S., 2018. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning URL: https://arxiv.org/abs/1811.12808v3, doi:10.48550/arxiv.1811.12808.

Ribeiro, M.T., Singh, S., Guestrin, C., 2016. "Why should i trust you?" Explaining the predictions of any classifier. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 13-17-August-2016, 1135–1144. URL: https://dl.acm.org/doi/10.1145/2939672.2939778, doi:10.1145/2939672.2939778.

Ruan, G., Schmidhalter, U., Yuan, F., Cammarano, D., Liu, X., Tian, Y., Zhu, Y., Cao, W., Cao, Q., 2023. Exploring the transferability of wheat nitrogen status estimation with multisource data and Evolutionary Algorithm-Deep Learning (EA-DL) framework. European Journal of Agronomy 143. doi:10.1016/J.EJA.2022.126727.

Salazar, F., Conde, A., Irazábal, J., Vicente, D.J., 2021. Anomaly Detection in Dam Behaviour with Machine Learning Classification Models. Water 2021, Vol. 13, Page 2387 13, 2387. URL: https://www.mdpi.com/2073-4441/13/17/2387/htmhttps://www.mdpi.com/2073-4441/13/17/2387, doi:10.3390/W13172387.

Santos, J.I., Pereda, M., Ahedo, V., Galán, J.M., 2023. Explainable machine learning for project managemenSantos, J. I., Pereda, M., Ahedo, V., & Galán, J. M. (2023). Explainable machine learning for project management control. Computers & Industrial Engineering, 180, 109261. https://doi.org/10.1016/J.CIE.2023. Computers & Industrial Engineering 180, 109261. doi:10.1016/J.CIE.2023.109261.

Slack, D., Hilgard, S., Jia, E., Singh, S., Lakkaraju, H., 2020. Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. AIES 2020 - Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society , 180–186URL: https://dl.acm.org/doi/10.1145/3375627.3375830, doi:10.1145/3375627.3375830.

Štrumbelj, E., Kononenko, I., 2014. Explaining prediction models and individual predictions with feature contributions. Knowledge and Information Systems 41, 647–665. URL: https://link.springer.com/article/10.1007/s10115-013-0679-x, doi:10.1007/S10115-013-0679-X/TABLES/4.

Wang, J., Zheng, P., Zhang, J., 2020. Big data analytics for cycle time related feature selection in the semiconductor wafer fabrication system. Computers and Industrial Engineering 143. doi:10.1016/J.CIE.2020.106362.

Wong, T.T., 2015. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. Pattern Recognition 48, 2839–2846. doi:10.1016/J.PATCOG.2015.03.009.

Wu, Y., 2022. Weighted McNemar's test for the comparison of two screening tests in the presence of verification bias. Statistics in Medicine 41, 3149–3163. doi:10.1002/SIM.9409.

Yan, J., He, Z., He, S., 2022. A deep learning framework for sensor-equipped machine health indicator construction and remaining useful life prediction. Computers & Industrial Engineering 172, 108559. doi:10.1016/J.CIE.2022.108559.

Zhan, P., Wang, S., Wang, J., Qu, L., Wang, K., Hu, Y., Li, X., 2021. Temporal anomaly detection on IIoT-enabled manufacturing. Journal of Intelligent Manufacturing 32, 1669–1678. URL: https://link.springer.com/article/10.1007/s10845-021-01768-1, doi:10.1007/S10845-021-01768-1/FIGURES/5.

4 Artigo 3 - Modelos de aprendizado de máquina para detecção de anomalias no processo de manufatura de pneus

XLIII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO
"A contribuição da engenharia de produção para desenvolvimento sustentável das organizações: Cadeias Circulares, sustentabilidade e tecnologias"
Fortaleza, Ceará, Brasil, 16 a 19 de outubro de 2023.

MODELOS DE APRENDIZADO DE MÁQUINA PARA DETECÇÃO DE ANOMALIAS NO PROCESSO DE MANUFATURA DE PNEUS

Rodrigo Marcel Araujo Oliveira (UFBA)

rodrigomarcel@ufba.br

Ângelo Márcio Oliveira Sant'Anna (UFBA)

angelo.santanna@ufba.br

A detecção de anomalias é a identificação de eventos que não estão em conformidade com o padrão esperado. Este artigo propõe desenvolver algoritmos de aprendizado de máquina não supervisionados para detectar anomalias em pneus que não estão em conformidade com os padrões de fabricação. Os dados são de um estudo realizado em uma indústria multinacional fabricante de pneus. Foram desenvolvidos os algoritmos Isolation Forest, Local Outlier Factor, One Class Support Vector Machine e Elliptic Envelope para avaliar o desempenho na detecção de anomalias com pneus não conformes. O algoritmo t-Distributed Stochastic Neighbor Embedding foi utilizado para representação dos dados em dimensão menor. O teste estatístico de McNemar foi utilizado para comparar o desempenho dos modelos propostos. A metodologia SHapley Additive exPlanations permitiu identificar a magnitude das variáveis preditoras para contribuição na decisão do melhor modelo Isolation Forest para detecção de pneus não conformes. Discutimos esses resultados com uma visão global e local das abordagens propostas.

***Palavras-chave**: Aprendizado de máquina, Detecção de anomalias, Manufatura de pneus, SHapley Additive exPlanations.*

1. Introdução

A inteligência artificial (IA) é um dos pilares do processo revolucionário da manufatura tradicional para a Indústria 4.0. A integração de sistemas inteligentes é um desafio nesse novo contexto para resolução de problemas complexos no setor industrial. O aprendizado de máquina é um ramo da IA, trata-se de algoritmos capazes de aprender padrões em função dos dados. Recentemente, modelos de aprendizado de máquina vêm ganhando espaço na academia para detecção de anomalias. A detecção de *outliers* ou anomalias é a identificação de eventos que não estão em conformidade com o padrão esperado em um conjunto de dados. Segundo Kamoona *et al*., (2021), esse procedimento vem sendo adotado na otimização de processos para fabricação inteligente que são essenciais para a garantia dos padrões de qualidade.

Em Acosta *et al*., (2021) modelos de aprendizado de máquina foram utilizados para prever níveis de concentração de fósforo em uma siderurgia. O treinamento de modelos de *deep learning* para detecção de defeitos de pneus baseado em imagem de profundidade é apresentado em Ko *et al*., (2021). O estudo visa detectar respingos de ventilação que ocorrem durante a fabricação de pneus. Os autores relatam diversos desafios, visto que a imagem de profundidade expressa em tons de cinza tem informações limitadas, além disso, embora o defeito seja o mesmo para os pneus, estes apresentam diferentes tamanhos e formas. O modelo proposto utiliza segmentação de aprendizado profundo que pode detectar dados de defeitos atípicos.

Em Zhan *et al*., (2021) é proposto um novo método de detecção de anomalias de séries temporais. A internet industrial das coisas (IIoT) tem como objetivo conectar equipamentos e aplicações em processos de manufatura avançada, fornecendo um número grande de dados de sensores para monitoramento em tempo real. Nesse artigo, os autores desenvolveram um novo método denominado HR-AD, baseado na representação hierárquica de séries temporais, o resultado do experimento demonstrou ser eficaz para detecção de anomalias.

A abordagem utilizando máquina de vetores de relevância (RVM) para previsão do peso do pneu e detecção de anomalias é proposta por Acosta *et al*., (2023). Nesse estudo o RVM apresentou o melhor desempenho comparado com outros algoritmos de aprendizado de máquina. Essa nova abordagem possibilitou um processo automático para detecção de falhas, além de propor valores de referência para o peso do pneu, melhorando o desempenho do processo de fabricação.

A previsão de desempenho de pneus com uso de diferentes algoritmos de aprendizado de máquina envolvendo mistura de variáveis numéricas e categóricas é discutida em Gutierrez-Gomez; Petry; Khadraoui, (2020). O trabalho utiliza alguns métodos como: regressão por vetores de suporte (SVR); métodos não paramétricos como o K vizinhos mais próximos (KNN);

técnicas de aumento de gradiente (GBoost) e Florestas Aleatória (RF). O interesse é avaliar diferentes métodos de codificação para lidar com variáveis categóricas combinadas com os de aprendizado de máquina. Métodos não supervisionados baseados em Redes Adversárias Generativas (GAN) para inspeção de defeitos nos pneus são apresentados por Wang *et al*., (2021). O estudo possibilitou a automatização do processo de inspeção de defeitos nos pneus que utilizam sensores de imagem de raios X.

Algoritmos de classificação de uma classe surgiram nos últimos anos com intuito de detectar *outliers*. Alguns métodos são baseados em árvores de decisão (HARIRI; KIND; BRUNNER, 2021), funções kernel (LI *et al.,* 2003), e densidade (BREUNIG *et al.,* 2000), (ASHRAFUZZAMAN *et al.,* 2020), etc. O algoritmo *One Class Support Vector Machine* foi utilizado por Muñoz-Marí *et al*., (2010) para detecção de imagens de sensoriamento remoto, segundo os autores o modelo possibilitou um método de detecção de mudanças no monitoramento urbano multifonte, nuvem multiespectral, etc. Os modelos *Isolation Forest* e *Local Outlier Factor* foram utilizados como base do trabalho de Alsini *et al*., (2021) para produção eficiente de misturas de concreto. O algoritmo *Elliptic Envelope* foi utilizado para detectar anomalias geoquímicas (CHEN *et al.,* 2021). Almuhtaram; Zamyadi; Hofmann, (2021) utilizaram algoritmos de detecção de anomalias em dados de fluorescência de ciana.

Este artigo propõe desenvolver algoritmos de aprendizado de máquina não supervisionados para detectar anomalias em pneus que não estão em conformidade com os padrões de fabricação. O estudo de caso utiliza informações do processo de uniformidade dos pneus provenientes de uma indústria multinacional. Foram desenvolvidos os algoritmos *Isolation Forest, Local Outlier Factor, One Class Support Vector Machine* e *Elliptic Envelope* para avaliar o desempenho na detecção de anomalias com pneus não conformes em relação aos padrões de produção. O algoritmo *t-distributed stochastic neighbor embedding* foi utilizado como auxílio para visualização dos dados em dimensão menor. Para avaliação dos resultados foram utilizadas algumas métricas de desempenho: *accuracy*, *precision*, *recall*, *f-score*, *roc-auc*. Comparamos os resultados dos desempenhos dos modelos com auxílio do teste de hipótese estatístico de *McNemar*. No final, utilizamos a metodologia *SHapley Additive exPlanations* (SHAP) para interpretação do modelo *Isolation Forest*, foi possível explicar quais foram os recursos com maior impacto no resultado do modelo. Discutimos esses resultados com uma visão global e local.

2. Revisão da literatura

2.1. *Local outlier factor*

O algoritmo *Local Outlier Factor* (LOF) é um modelo não supervisionado usado para detecção de *outlier*, a metodologia proposta por Breunig *et al.,* (2000) consiste em calcular o desvio de densidade local de um determinado ponto em relação aos pontos vizinhos, o resultado é uma pontuação que representa os pontos de anomalias. A estimativa da densidade local é obtida calculando as distâncias entre os k-vizinhos mais próximos, a metodologia consiste em comparar as densidades semelhantes, isto é, verificar quais pontos têm densidade menor que seus vizinhos (ALGHUSHAIRY *et al.,* 2020). Essas distâncias são utilizadas para calcular a distância de alcance, conforme a Equação 1, esta é definida como o máximo da distância entre dois pontos e a k-distância em relação ao ponto de referência.

$$R_k(i,j) = max\{d_k(j), d(i,j)\} \tag{1}$$

A distância de alcance é calculada para estimativa da densidade de alcançabilidade local, conforme a Equação 2, ela é uma métrica que determina o quanto os pontos estão próximos.

$$LRD_k(i) = {1}\Big/{\left(\frac{\sum_{j \in N_k(i)} R_k(i,j)}{|N_k(i)|}\right)} \tag{2}$$

A Equação 3 representa o cálculo do LOF, ele é obtido pela razão entre a média das densidades de alcançabilidade local com número k de vizinhos e a densidade de alcance local do ponto. Conforme Alsini *et al.,* (2021) o dado é considerado uma anomalia se LOF for maior que 1.

$$LOF_k(i) = \frac{\sum_{j \in N_k(i)} \frac{LRD_k(j)}{LRD_k(i)}}{|N_k(i)|} \tag{3}$$

2.2. *Isolation forest*

As árvores de decisão são modelos de aprendizado de máquina que representam regras de decisão baseadas nos valores dos atributos (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). O *Isolation Forest* (*iForest*) proposto por Liu; Ting; Zhou, (2008) é um modelo para detecção de anomalias determinado por um conjunto de árvores de decisão binárias que pode ser utilizado como método supervisionado ou não.

O algoritmo consiste em fazer partições aleatórias dos dados em que cada árvore é denominada árvore de isolamento (CHENG; ZOU; DONG, 2019). Para um recurso numérico selecionado aleatoriamente a ramificação da árvore é realizada em função de um valor φ aleatório do intervalo. Os pontos menores que φ são enviados para o lado esquerdo da ramificação, caso contrário vão para o lado direito. Esse processo é executado recursivamente até que a profundidade máxima seja atingida. O processo é realizado novamente em função do número de árvores da floresta, em que é selecionada uma subamostra para construção da árvore aleatória. Segundo Hariri; Kind; Brunner, (2021) a pontuação de um ponto inédito é representada pela Equação 4, em que $E(h(x))$ é o valor médio das profundidades que um ponto inédito atinge em todas as árvores da floresta, $c(k)$ é um fator de normalização que representa a profundidade média mal sucedida em uma busca na árvore binária de pesquisa, conforme a Equação 5, em que *k* é no número de pontos usado na construção da árvore e *H(i)* é número harmônico estimado segundo a Equação 6.

$$S(x,k) = 2^{\frac{-E(h(x))}{c(k)}} \quad (4)$$

$$c(k) = 2H(k-1) - (\frac{2(k-1)}{k}) \quad (5)$$

$$H(i) = ln\, i \; + 0{,}5772156649 \quad (6)$$

A pontuação de anomalia é referenciada por -1 para anomalias e 1 a pontos normais em função do parâmetro de contaminação, este é definido pela proporção de outliers no conjunto de dados.

2.3. *One class support vector machine*

O *One Class Support Vector Machine* (OCSVM) é um modelo não supervisionado para detecção de anomalias ou outliers (LI *et al.,* 2003). O modelo não possui rótulos de destino para o processo de treinamento do modelo, este aprende o limite para os pontos de dados normais e identifica os dados fora da borda como anomalias.

O algoritmo OCSVM foi proposto por Schölkopf *et al.,* (2001), a estrutura do modelo é uma adaptação do *Suppot Vector Machine* (CORTES; VAPNIK; SAITTA, 1995) para o caso de uma classe. A metodologia consiste em maximizar a margem que separa os dados de treinamento da origem. A função objetivo para ser otimizada é representada pela Equação 7.

$$\min_{\omega,\xi_j,\rho} \frac{1}{2}\|\omega\|^2 + \frac{1}{\upsilon m}\sum_{j=1}^{m}\xi_j - \rho \quad (7)$$

sujeito,

$$\left(\omega \cdot \phi\left(x_j\right)\right) \geq p - \xi_j, \forall\, j = 1, \ldots, m \quad (8)$$

$$\xi_j \geq 0, \forall\, j = 1, \ldots, m \quad (9)$$

O parâmetro $\upsilon \in (0, 1)$ define o limite superior na fração de *outilers*, ω é o vetor normal, os ξ_j são variáveis de folga que são usadas para modelar os erros de separação, ρ representa o viés associado à distância do hiperplano à origem. Já a função ϕ é uma projeção não linear dos dados em um espaço de alta dimensão, essa está associada a função kernel considerada no problema, conforme a Equação 10. As funções de kernel comumente utilizadas são: linear, sigmoid, RBF e polinomial. Tax & Duin, (2004) adotam uma abordagem esférica, o algoritmo obtém um limite esférico, no espaço de recursos, e o volume dessa esfera é minimizado.

$$K\left(x_i, x_j\right) = \phi(x_i)\phi\left(x_j\right) \quad (10)$$

2.4. *Elliptic envelope*

O método *Elliptic Envelope* (EE) é um modelo estatístico com distribuição gaussiana multivariada que avalia a covariância entre as variáveis. O princípio desse modelo é delinear uma elipse de modo que dados que estão fora dela sejam considerados anomalias ou *outliers*. O método EE usa o determinante de covariância mínima para estimar a forma e o tamanho da elipse (ROUSSEEUW; VAN DRIESSEN, 1999). A métrica da distância de Mahalanobis conforme a Equação 11 é utilizada para calcular a distância entre um ponto e uma distribuição, onde μ e o parâmetro de localização e $\sum$ a matriz de covariância. Esses valores são ordenados e a partir disso é definido o limite para determinar os pontos de *outilers*.

$$d_{\mu,\sum}(x_i) = \left((x_i - \mu)^t \sum^{-1} (x_i - \mu)\right)^{1/2} \quad (11)$$

2.5. *Algoritmo t-distributed stochastic neighbor embedding*

O algoritmo *t-distributed stochastic neighbor embedding* (t-SNE) proposto por Van der Maaten & Hinton, (2008) é um método de redução de dimensionalidade não linear utilizado para análise e visualização de dados que consiste em projetar dados de alta dimensão em espaço de baixa dimensão.

O t-SNE foi desenvolvido a partir da incorporação estocástica de vizinhos e usa a distribuição de probabilidade t de *Student* (ZHOU; WANG; TAO, 2018). A distribuição de probabilidade no espaço original é representada pela Equação 12, que corresponde a probabilidade do ponto x_j ser selecionado como vizinho do ponto x_i, para cada ponto é assumida uma distribuição Gaussiana com média x_i e desvio padrão σ_i para seus vizinhos.

$$p_{j|i} = \frac{e^{-\|x_i - x_j\|^2 / 2\sigma_i^2}}{\sum_{l \neq i} e^{-\|x_i - x_l\|^2 / 2\sigma_i^2}} \tag{12}$$

A probabilidade conjunta que descreve a similaridade dos pontos y_i e y_j no espaço de baixa dimensão é representada pela Equação 13, a distribuição t de *Student* com um grau de liberdade é empregada para cálculo.

$$q_{ij} = \frac{\left(1+\|y_i - y_j\|^2\right)^{-1}}{\sum_{l \neq i}(1+\|y_i - y_l\|^2)^{-1}} \tag{13}$$

O objetivo do método t-SNE é encontrar uma representação de baixa dimensão que minimize a diferença entre p_{ij} e q_{ij}. Conforme Van der Maaten & Hinton, (2008), a proposta para comparar essas diferenças é usando a divergência de Kullback−Leibler (KL). A Equação 14 representa esse método, este é utilizado para avaliar a projeção dos dados de alta dimensão (P) para representação de baixa dimensão (Q) em função de todos os pontos de dados. Geralmente a função de custo C é minimizada usando o método de gradiente descendente (ZHU *et al.,* 2019). Um parâmetro importante no t-SNE é a perplexidade, definida na Equação 15, trata-se de uma medida de dispersão utilizada para determinar o número dos vizinhos mais próximos de um ponto x_i, é notável que o logaritmo na base 2 da perplexidade é a entropia do sistema.

$$\mathrm{C} = KL(P||Q) = \sum_{i \neq j} p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right) \tag{14}$$

$$\mathrm{Perp}(p_i) = 2^{-\sum_j p_{ij} \log_2 p_{ij}} \tag{15}$$

2.5. Métricas de avaliação dos modelos

A validação cruzada é um método estatístico comumente utilizado para avaliação de modelos no contexto de aprendizado de máquina (WANG, 2015). O objetivo desse método é avaliar a

capacidade de generalização dos modelos. O *Hold-out* é uma das metodologias frequentemente utilizadas que consiste na divisão dos dados em treino e validação em função de uma proporção predefinida dos dados (WU *et al.,* 2010). O método *GridSearchCV* é uma alternativa otimizada para estimativa dos parâmetros dos modelos, ele faz uma busca exaustiva com diversas combinações dos valores de parâmetros especificados dos modelos (AHMAD et al., 2022). Os parâmetros selecionados são aqueles que maximizam a métrica de avaliação do modelo.

Para avaliar o desempenho dos modelos ajustados e posteriormente eleger os melhores, vamos nos basear principalmente em métricas derivadas da matriz de confusão bidimensional indexada pela classe real e a outra pela classe predita do classificador (Deng et al., 2016).

As métricas *accuracy*, *precision*, *recall* e *f-score* foram utilizadas para avaliação dos modelos, conforme as Equações 16, 17, 18 e 19, respectivamente. Além disso, também utilizamos a curva de característica de operação do receptor (ROC), o eixo das abcissas contém a informação da taxa de falsos positivos, e o eixo das ordenadas a taxa de verdadeiros positivos. A área sob a curva ROC-AUC foi utilizada como critério de avaliação, trata-se de uma medida que fornece a informação do desempenho do modelo considerando todos os limites de classificação.

$$AC = \frac{\sum_{i=1}^{2} n_{ii}}{\sum_{i=1}^{2}\sum_{j=1}^{2} n_{ij}} \tag{16}$$

$$PREC_i = \frac{n_{ii}}{\sum_{j=1}^{2} n_{ji}} \tag{17}$$

$$REC_i = \frac{n_{ii}}{\sum_{j=1}^{2} n_{ij}} \tag{18}$$

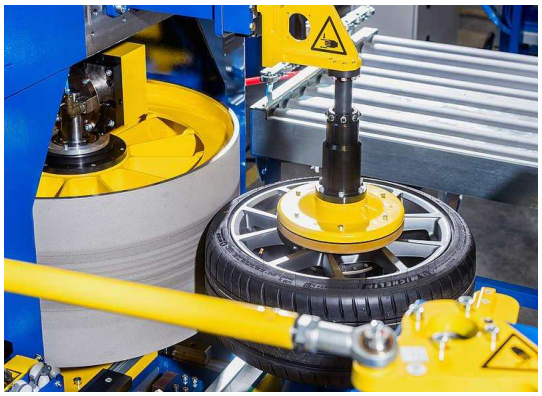
$$F\ score_i = \frac{2 \times PREC_i \times REC_i}{PREC_i + REC_i} \tag{19}$$

3. Estudo experimental

Os dados são provenientes de um sistema inteligente de manufatura de uma fábrica multinacional que fabrica pneus para carros de passeio. A variável resposta deste estudo é a conformidade ou não do pneu aos padrões de qualidade da fábrica. O pneu está condicionado aos padrões aceitos por fabricantes globais de pneus, como forma de identificar pneus com baixo desempenho para que não sejam vendidos no mercado, esse processo é denominado teste de uniformidade do pneu. O estudo inclui a variável resposta binária em que 1 corresponde ao pneu defeituoso. As variáveis explicativas são: flutuação da força radial (RFV); flutuação da força lateral (LFV); excentricidade radial (RRO); efeito de conicidade (CONICITY); efeito angular (PLY); força radial 2º harmônico (H2RFV); emenda de tampa (CAPSPLICE). O banco

de dados contém uma amostra com 61 observações defeituosas e 46 em conformidade. A Figura 1 mostra o equipamento com sensores que medem as características do pneu.

Figura 1 – Processo de uniformidade do pneu



Fonte: ("Wheel Uniformity Measuring Machines | Hofmann: the Original!", [s.d.])

4. Resultados e discussão

Para desenvolvimento desse projeto foi utilizado o programa Python®. Os dados foram separados em dois conjuntos de amostra estratificada utilizando a técnica *Hold-out,* com proporção de 70% para treinamento dos modelos e 30% para validação. Para amostra de treinamento a distribuição da quantidade de pneus conformes (C) e não conformes (NC) são: 32 (C); 42 (NC), e para a amostra de validação a quantidade é de: 14 (C); 19 (NC).

A configuração adotada na validação cruzada para o modelo *iForest* foi a *Hold-out*. O critério para seleção de parâmetros dos modelos foi o *GridSearchCV* e a métrica otimizada foi o *recall*, isto é, estamos interessados em capturar o maior número de anomalias. Para utilização desse método foi necessário criar uma variável auxiliar com os rótulos de 1 para pneus C e -1 para NC, pois os métodos de detecção de anomalias têm como resposta 1 para pontos *inliers* e -1 para *outilers*. Os parâmetros escolhidos na modelagem foram: 256 para número de árvores na floresta; 90% para o número máximo de variáveis em cada árvore; e 50% para o parâmetro de contaminação.

Os parâmetros do OCSVM foram escolhidos: *gamma* igual *scale*; o parâmetro *nu* que corresponde ao limite inferior na fração dos vetores de suporte foi de 50%; e o *kernel* não linear *rbf*. Para o LOF, a distância considerada foi a euclidiana com 4 de vizinhos mais próximos, o fator de contaminação também foi de 50%, o mesmo considerado no modelo EE.

As Tabelas 1 e 2 apresentam os resultados dos modelos não supervisionados para as etapas de treinamento e validação, respectivamente. É notável que os modelos *iForest* e LOF são os que apresentaram os melhores resultados no treinamento, isto é, a concordância entre a anomalia e

pneus NC. Nos dados de validação o modelo OCSVM foi o que apresentou o resultado mais interessante, com *iForest* em seguida. Figura 2 apresenta o gráfico da curva *ROC-AUC* de todos os modelos na base de validação, corroborando com os resultados apresentados na Tabela 2.

Tabela 1 – Métricas de avaliação na base de Treinamento

Modelos	**Métricas**			
	Precison	***Recall***	***F-score***	***Accuracy***
iForest	0,65	**0,57**	**0,61**	**0,58**
OCSVM	0,62	0,55	0,58	0,55
EE	**0,70**	0,33	0,45	0,54
LOF	0,65	**0,57**	**0,61**	**0,58**

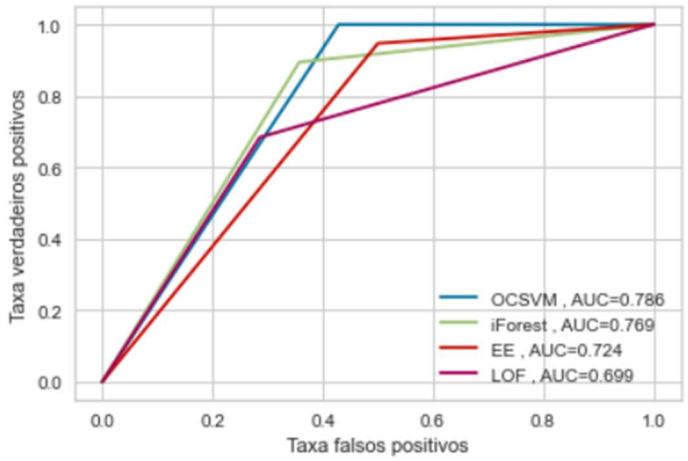
Fonte: Autor

Tabela 2 – Métricas de avaliação na base de Validação

Modelos	**Métricas**			
	Precison	***Recall***	***F-score***	***Accuracy***
iForest	**0,77**	0,89	0,83	0,79
OCSVM	0,76	**1,00**	**0,86**	**0,82**
EE	0,76	0,68	0,72	0,70
LOF	0,72	0,95	0,82	0,76

Fonte: Autor

Figura 2 – Gráfico da Curva *ROC-AUC* para os dados de validação

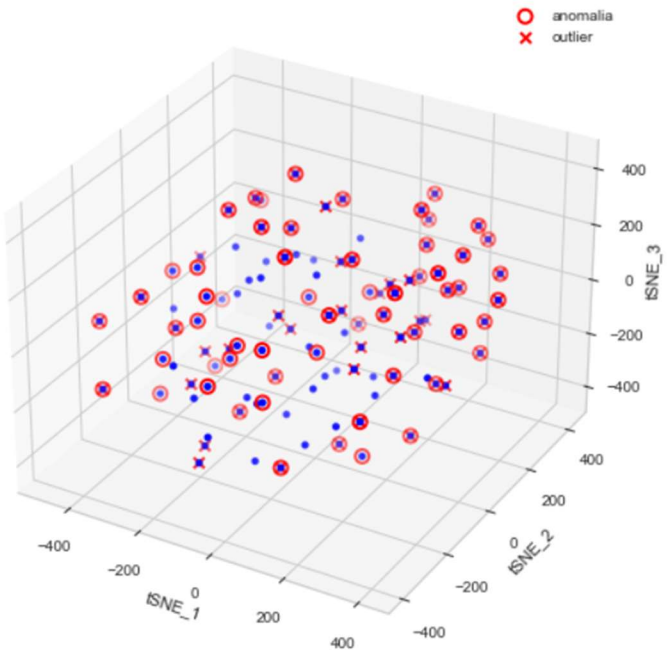


Fonte: Autor

O teste estatístico de *McNemar* para dados pareados foi empregado para comparar o desempenho dos modelos de aprendizado de máquina propostos nos dados de validação. Conforme Latah & Toker, (2018) a hipótese nula do teste corresponde a semelhança na proporção de erros dos modelos, assim a hipótese alternativa sugere que os desempenhos dos

modelos são diferentes. O resultado do teste ao nível de significância de 5 % sugere não rejeitar a hipótese nula para todas as comparações entre os modelos, portanto não há evidências significativas de que os modelos tenham desempenhos diferentes. O algoritmo t-SNE foi utilizado para reduzir a dimensionalidade das variáveis e representar estas em 3 componentes. Na Figura 3 apresentamos o resultado do modelo *iForest*, onde cada eixo do gráfico corresponde a representação de uma componente calculada com algoritmo t-SNE.

Figura 3 – Componentes do t-SNE em função do resultado do modelo *iForest*

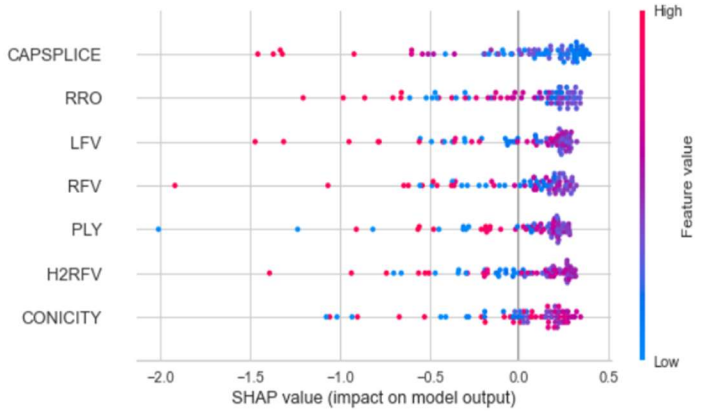


Fonte: Autor

A interpretação de modelos de aprendizado de máquina é um desafio. A abordagem *SHapley Additive exPlanations* (SHAP) proposto por Lundberg & Lee, (2017) permite interpretar as previsões de modelos de aprendizado de máquina. A metodologia é baseada na Teoria dos Jogos e consiste em calcular o impacto de cada recurso na previsão em função de uma entrada específica. O método SHAP é ideal para modelos baseados em árvores.

A Figura 4 fornece uma interpretação global dos resultados, em que cada ponto representa um exemplo do conjunto de dados de validação. Para cada variável é possível identificar o impacto dela na decisão do modelo para detecção de anomalias. Para variável CAPSPLICE valores baixos representam maior contribuição para o modelo conforme a tonalidade da cor. É notável que valores altos para as variáveis RRO e RFV contribuem de forma negativa para o *score* do modelo, portanto maior a propensão do ponto ser uma anomalia.

Figura 4 – Importância global das variáveis no resultado do modelo



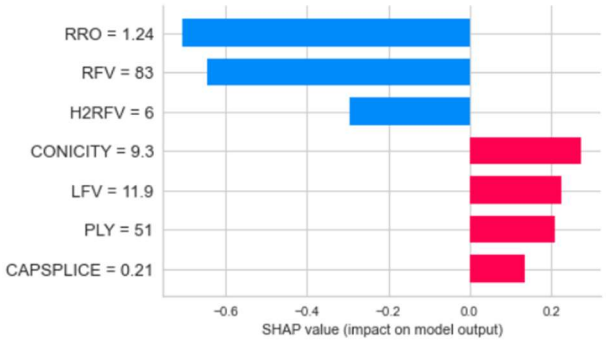
Fonte: Autor

Para uma interpretação local escolhemos um ponto aleatório do conjunto de dados de validação que foi predito pelo modelo *iForest* como anomalia, e de fato o pneu é NC. Na Figura 5 o eixo em cinza corresponde o impacto de cada variável para resposta do modelo *iForest*, o valor obtido para esse ponto foi de 7,96, portanto uma chance alta de ser uma anomalia. Notamos que valores baixos de RRO, RFV e H2RFV contribuem negativamente para o resultado do modelo, enquanto valores altos de CAPSPLICE, PLY, LFV e CONICITY impactam positivamente. Figura 6 apresenta o mesmo resultado do valor no ponto para cada variável.

Figura 5 – Força de impacto local das variáveis

Fonte: Autor

Figura 6 – Valores SHAP para importância local



Fonte: Autor

5. Conclusões

Os modelos de aprendizado de máquinas são técnicas amplamente utilizadas para detecção de anomalias. O presente trabalho apresentou os algoritmos de aprendizado de máquina não supervisionados desenvolvidos para detectar anomalias em pneus que não estão em conformidade com os padrões de fabricação. Os algoritmos não supervisionados desenvolvidos com estruturas distintas foram capazes de aprender as estruturas não lineares das variáveis preditoras provenientes do processo de uniformidade dos pneus.

Conforme o teste estatístico de *McNemar*, não houve diferença entre a proporção de erros dos modelos, portanto os modelos apresentaram performance semelhante. Os valores de *accuracy, precision, recall e f-score* apresentaram valores satisfatórios no conjunto de dados de validação, comprovando a robustez desses métodos. Também foi possível identificar a importância das variáveis preditoras para detecção de anomalias, vimos que, valores pequenos de RRO, RFV e H2RFV impactam de forma negativa para a performance de detecção de anomalias do melhor modelo desenvolvido.

Os resultados desse trabalho são de grande importância para o processo de manufatura de pneus. O monitoramento da manufatura de produtos para detecção de falhas é essencial para garantir a eficiência dos processos de qualidade dos produtos de uma fábrica. Para pesquisas futuras, recomendamos estudar outros métodos de detecção de *outlier*s como: modelos de *clusters* e uma análise das fronteiras de decisão das variáveis preditoras com intervalos de confiança para cada variável preditora. Isso pode viabilizar a implementação de políticas para a fabricação de pneus com melhores desempenhos, e consequentemente, diminuindo custos operacionais, aumentando o tempo de vida útil e diminuindo o impacto ambiental desse produto enquanto resíduo.

REFERÊNCIAS

ACOSTA, S. M., MARCEL, R., OLIVEIRA, A., & SANT'ANNA, A. M. O. Machine learning algorithms applied to intelligent tyre manufacturing. **International Journal of Computer Integrated Manufacturing**. p. 1–11, 10 fev. 2023.

ACOSTA, S. M., AMOROSO, A. L., SANT'ANNA, A. M. O., & CANCIGLIERI, J, O. Relevance vector machine with tuning based on self-adaptive differential evolution approach for predictive modelling of a chemical process. **Applied Mathematical Modelling**, v. 95, p. 125–142, 1 jul. 2021.

ALGHUSHAIRY, O., ALSINI, R., SOULE, T., & MA, X. A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. **Big Data and Cognitive Computing 2021, Vol. 5, Page 1**, v. 5, n. 1, p. 1,

29 dez. 2020.

ALMUHTARAM, H.; ZAMYADI, A.; HOFMANN, R. Machine learning for anomaly detection in cyanobacterial fluorescence signals. **Water Research**, v. 197, p. 117073, 1 jun. 2021.

ALSINI, R., ALMAKRAB, A., IBRAHIM, A., & MA, X.. Improving the outlier detection method in concrete mix design by combining the isolation forest and local outlier factor. **Construction and Building Materials**, v. 270, p. 121396, 8 fev. 2021.

ASHRAFUZZAMAN, M., DAS, S., JILLEPALLI, A. A., CHAKHCHOUKH, Y., & SHELDON, F. T. Elliptic Envelope Based Detection of Stealthy False Data Injection Attacks in Smart Grid Control Systems. **2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020**, p. 1131–1137, 1 dez. 2020.

BREUNIG, M. M., KRIEGEL, H.-P., NG, R. T., & SANDER, J. LOF: Identifying Density-Based Local Outliers. 2000.

CHEN, Y., WANG, S., ZHAO, Q., & SUN, G. Detection of Multivariate Geochemical Anomalies Using the Bat-Optimized Isolation Forest and Bat-Optimized Elliptic Envelope Models. **Journal of Earth Science**, v. 32, n. 2, p. 415–426, 1 abr. 2021.

CHENG, Z.; ZOU, C.; DONG, J. Outlier detection using isolation forest and local outlier. **Proceedings of the 2019 Research in Adaptive and Convergent Systems, RACS 2019**, p. 161–168, 24 set. 2019.

CORTES, C.; VAPNIK, V.; SAITTA, L. Support-vector networks. **Machine Learning 1995 20:3**, v. 20, n. 3, p. 273–297, set. 1995.

GUTIERREZ-GOMEZ, L.; PETRY, F.; KHADRAOUI, D. A Comparison Framework of Machine Learning Algorithms for Mixed-Type Variables Datasets: A Case Study on Tire-Performances Prediction. **IEEE Access**, v. 8, p. 214902–214914, 2020.

HARIRI, S.; KIND, M. C.; BRUNNER, R. J. Extended Isolation Forest. **IEEE Transactions on Knowledge and Data Engineering**, v. 33, n. 4, p. 1479–1489, 1 abr. 2021.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning**. Springer Series in Statistics. 2009.

KAMOONA, A. M., GOSTAR, A. K., BAB-HADIASHAR, A., & HOSEINNEZHAD, R. Point Pattern Feature-Based Anomaly Detection for Manufacturing Defects, in the Random Finite Set Framework. **IEEE Access**, v. 9, p. 158672–158681, 2021.

KO, D., KANG, S., KIM, H., LEE, W., BAE, Y., & PARK, J. Anomaly Segmentation Based on Depth Image for Quality Inspection Processes in Tire Manufacturing. **Applied Sciences 2021, Vol. 11, Page 10376**, v. 11, n. 21, p. 10376, 4 nov. 2021.

LATAH, M.; TOKER, L. Towards an efficient anomaly-based intrusion detection for software-defined networks. **IET Networks**, v. 7, n. 6, p. 453–459, 1 nov. 2018.

LI, K. L., HUANG, H. K., TIAN, S. F., & XU, W. Improving one-class SVM for anomaly detection. **International Conference on Machine Learning and Cybernetics**, v. 5, p. 3077–3081, 2003.

LIU, F. T.; TING, K. M.; ZHOU, Z. H. Isolation forest. **Proceedings - IEEE International Conference on Data Mining, ICDM**, p. 413–422, 2008.

LUNDBERG, S. M.; LEE, S. I. A Unified Approach to Interpreting Model Predictions. **Advances in Neural Information Processing Systems**, v. 2017- December, p. 4766–4775, 22 maio 2017.

MUÑOZ-MARÍ, J., BOVOLO, F., GÓMEZ-CHOVA, L., BRUZZONE, L., & CAMP-VALLS, G. Semisupervised one-class support vector machines for classification of remote sensing data. **IEEE Transactions on Geoscience and Remote Sensing**, v. 48, n. 8, p. 3188–3197, ago. 2010.

ROUSSEEUW, P. J.; VAN DRIESSEN, K. A fast algorithm for the minimum covariance determinant estimator. **Technometrics**, v. 41, n. 3, p. 212–223, 1999.

SCHÖLKOPF, B., PLATT, J. C., SHAWE-TAYLOR, J., SMOLA, A. J., & WILLIAMSON, R. C. Estimating the Support of a High-Dimensional Distribution. **Neural Computation**, v. 13, n. 7, p. 1443–1471, 1 jul. 2001.

TAX, D. M. J.; DUIN, R. P. W. Support Vector Data Description. **Machine Learning**, v. 54, n. 1, p. 45–66, jan. 2004.

VAN DER MAATEN, L.; HINTON, G. Visualizing Data using t-SNE. **Journal of Machine Learning Research**, v. 9, p. 2579–2605, 2008.

WANG, Y., ZHANG, Y., ZHENG, L., YIN, L., CHEN, J., & LU, J. Unsupervised Learning with Generative Adversarial Network for Automatic Tire Defect Detection from X-ray Images. **Sensors 2021, Vol. 21, Page 6773**, v. 21, n. 20, p. 6773, 12 out. 2021.

Wheel Uniformity Measuring Machines | Hofmann: the Original! Disponível em: <https://www.hofmannmaschinen.com/en/business-segments/wheel-assembly-lines/single-systems/wheel-uniformity-measuring-machines/>. Acesso em: 5 mar. 2023.

ZHAN, P., WANG, S., WANG, J., QU, L., WANG, K., HU, Y., & LI, X. Temporal anomaly detection on IIoT-enabled manufacturing. **Journal of Intelligent Manufacturing**, v. 32, n. 6, p. 1669–1678, 1 ago. 2021.

ZHOU, H.; WANG, F.; TAO, P. t-Distributed Stochastic Neighbor Embedding Method with the Least Information Loss for Macromolecular Simulations. **J. Chem. Theory Comput**, v. 14, n. 11, p. 5499-5510, 2018.

ZHU, W., WEBB, Z. T., MAO, K., & ROMAGNOLI, J. A Deep Learning Approach for Process Data Visualization Using t-Distributed Stochastic Neighbor Embedding. **Ind. Eng. Chem. Res**, v. 58, n. 22, p. 9564-9575, 2019.

5 Conclusão

A presente dissertação apresentou diferentes abordagens de modelos de aprendizado de máquina supervisionados e não supervisionados para reconhecimento de padrões no processo de manufatura em uma indústria multinacional fabricante de pneus. Os algoritmos com estruturas distintas foram capazes de aprender estruturas não lineares de variáveis provenientes do processo de uniformidade dos pneus, para classificação de pneus conformes e aqueles que não estão em conformidade com os padrões de fabricação.

Em relação ao objetivo específico **propor modelos de aprendizado de máquina para classificação de atributos multiclasses**, o artigo 1 apresentou diversas abordagens de modelos de aprendizado de máquina supervisionado para classificação multiclasse. O modelo *Random Forest* apresentou o melhor desempenho no conjunto de validação, com maiores valores de *accuracy*, e com auxílio da análise dos valores de SHAP foi possível identificar que as variáveis CONICITY, RRO e RFV possuem os maiores impactos globais para o desempenho do pneu. Os resultados apontam que valores SHAP pequenos de CONICITY, RRO, RFV e H2RFV têm grande importância para a categoria A do pneu.

Em relação ao objetivo específico **propor modelos de aprendizado de máquina para classificação de atributos binários**, o artigo 2 apresentou as abordagens de modelos de aprendizado de máquina supervisionado para classificação binária e um *framework* para interpretabilidade dos modelos desenvolvidos. Segundo o teste estatístico de *McNemar*, não houve diferença entre a proporção de erros dos modelos. Os modelos *Random Forest* e *Logistic Regression* apresentaram os melhores resultados de *accuracy* e *AUC-ROC* no conjunto de dados de validação. O modelo *Random Forest* foi considerado para identificar a importância das variáveis preditoras na detecção de defeitos. A análise global do método SHAP permitiu afirmar que valores grandes de RRO, CONICITY, RFV e H2RFV impactam de forma positiva a probabilidade do pneu ser defeituoso. Na análise local com os métodos LIME e SHAP foi possível identificar a influência de cada variável preditora para classificação final do desempenho do pneu. Também foi apresentado uma metodologia para identificar valores referência das variáveis preditoras, a proposta foi embasada na abordagem de otimização linear em função dos valores dos coeficientes do resultado apresentado do modelo *Logistic Regression*. Essa metodologia permitiu encontrar

valores de referência para as variáveis RRO, CONICITY e H2RFV para classificação de pneus em conformidade.

Em relação ao objetivo específico **propor modelos de aprendizado de máquina para detecção de anomalias**, o artigo 3 apresentou abordagens de modelos não supervisionados para detecção de anomalias. Com teste estatístico de *McNemar* foi possível afirmar que modelos apresentaram performance semelhante. A importância das variáveis preditoras foi avaliada em função do modelo *Isolation Forest* para detecção de anomalias, os valores de SHAP indicam que valores pequenos de RRO, RFV e H2RFV impactam de forma negativa para a performance de detecção de anomalias do modelo.

Os resultados desta dissertação são de grande importância para o processo de manufatura de pneus. Abordagens de modelos de aprendizado de máquina podem contribuir para o processo de monitoramento de processos industriais. No contexto da Indústria 4.0 isso é essencial para garantir a eficiência dos processos de qualidade dos produtos de uma fábrica. A tomada de decisão pautada em dados pode viabilizar a implementação de políticas para manufatura de produtos com melhores desempenhos, contribuindo para diminuir custos operacionais, aumentar o tempo de vida útil e diminuir o impacto ambiental desses produtos enquanto resíduos na natureza. Os códigos desenvolvidos durante a pesquisa estão disponíveis no repositório do `Github` dedicado a esse projeto: ⟨https://github.com/roaraujo/machine-learning-manufacturing-processes⟩.

5.1 Pesquisas futuras

Esta pesquisa buscou aplicar modelos de aprendizado de máquina supervisionado e não supervisionado analisar e definir valores de referência para as variáveis de processo da manufatura de pneus em conformidade com os padrões de fabricação, a fim de possibilitar as variações no produto final e reduzir as perdas no processo. A partir da abordagem proposta, algumas sugestões podem ser investigadas em trabalhos futuros: *(i)* propor modelos lineares generalizados como uma alternativa para encontrar associação entre as variáveis de processo e os padrões de defeitos; *(ii)* propor modelos não supervisionados para identificar grupos e analisar a concordância destes com os desempenhos dos pneus pode trazer informações relevantes, levar a abordagens robustas e permitir um maior

entendimento das relações entre as variáveis; *(iii)* propor cartas de controle baseadas em modelos para monitorar e detectar anomalias.

Referências[1]

ACOSTA, S. M. et al. Relevance vector machine with tuning based on self-adaptive differential evolution approach for predictive modelling of a chemical process. *Applied Mathematical Modelling*, Elsevier, v. 95, p. 125–142, 7 2021. ISSN 0307-904X. Citado na página 10.

ACOSTA, S. M. et al. Machine learning algorithms applied to intelligent tyre manufacturing. *International Journal of Computer Integrated Manufacturing*, Taylor & Francis, v. 10, p. 1–11, 2 2023. ISSN 0951-192X. Disponível em: ⟨https://www.tandfonline.com/doi/abs/10.1080/0951192X.2023.2177734⟩. Citado 2 vezes nas páginas 10 e 13.

CARVALHO, J. L.; FARIAS, P. C.; FILHO, E. F. S. Global Localization of Unmanned Ground Vehicles Using Swarm Intelligence and Evolutionary Algorithms. *Journal of Intelligent and Robotic Systems: Theory and Applications*, Institute for Ionics, v. 107, n. 3, p. 1–21, 3 2023. ISSN 15730409. Disponível em: ⟨https://link.springer.com/article/10.1007/s10846-023-01813-6⟩. Citado na página 13.

CARVALHO, T. P. et al. A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, Pergamon, v. 137, p. 106024, 11 2019. ISSN 0360-8352. Citado na página 10.

CHO, J. R. et al. Optimum design of run-flat tire insert rubber by genetic algorithm. *Finite Elements in Analysis and Design*, Elsevier, v. 52, p. 60–70, 5 2012. ISSN 0168-874X. Citado na página 15.

DU, M. et al. A Study on the Influence of Tire Speed and Pressure on Measurement Parameters Obtained from High-Speed Tire Uniformity Testing. *Vehicles 2020, Vol. 2, Pages 559-573*, Multidisciplinary Digital Publishing Institute, v. 2, n. 3, p. 559–573, 9 2020. ISSN 2624-8921. Disponível em: ⟨https://www.mdpi.com/2624-8921/2/3/31/htmhttps://www.mdpi.com/2624-8921/2/3/31⟩. Citado na página 15.

GUTIERREZ-GOMEZ, L.; PETRY, F.; KHADRAOUI, D. A Comparison Framework of Machine Learning Algorithms for Mixed-Type Variables Datasets: A Case Study on Tire-Performances Prediction. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 8, p. 214902–214914, 2020. ISSN 21693536. Citado na página 10.

GUTIERREZ-GOMEZ, L.; PETRY, F.; KHADRAOUI, D. A Comparison Framework of Machine Learning Algorithms for Mixed-Type Variables Datasets: A Case Study on Tire-Performances Prediction. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 8, p. 214902–214914, 2020. ISSN 21693536. Citado na página 13.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The Elements of Statistical Learning. Springer New York, New York, NY, 2009. Disponível em: ⟨http://link.springer.com/10.1007/978-0-387-84858-7⟩. Citado 2 vezes nas páginas 12 e 13.

KAMOONA, A. M. et al. Point Pattern Feature-Based Anomaly Detection for Manufacturing Defects, in the Random Finite Set Framework. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 9, p. 158672–158681, 2021. ISSN 21693536. Citado na página 10.

[1] De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

LAGARINHOS, C. A.; TENÓRIO, J. A. Logística reversa dos pneus usados no Brasil. *Polímeros*, Associação Brasileira de Polímeros, v. 23, n. 1, p. 49–58, 2013. ISSN 0104-1428. Disponível em: ⟨https://www.scielo.br/j/po/a/qZbJvJkKWw5LnZfcyM8FhGj/?format=html⟩. Citado na página 12.

LEE, S. K. et al. Prediction of tire pattern noise in early design stage based on convolutional neural network. *Applied Acoustics*, Elsevier Ltd, v. 172, 1 2021. ISSN 1872910X. Citado na página 11.

LI, X.; GUO, M.; ZHOU, X. A multivariate multiple regression analysis of tire-road contact peak triaxial stress by using machine learning methods. *Mechanics of Advanced Materials and Structures*, Taylor and Francis Ltd., 2021. ISSN 15376532. Citado na página 11.

LIU, Z. et al. Stochastic analysis for in-plane dynamic responses of low-speed uniformity of tires due to geometric defects. *Mechanical Systems and Signal Processing*, Academic Press, v. 197, p. 110377, 8 2023. ISSN 0888-3270. Citado na página 15.

MAO, S. et al. Opportunities and Challenges of Artificial Intelligence for Green Manufacturing in the Process Industry. *Engineering*, Elsevier, v. 5, n. 6, p. 995–1002, 12 2019. ISSN 2095-8099. Citado na página 12.

PEDRAM, A. et al. Integrated forward and reverse supply chain: A tire case study. *Waste Management*, Pergamon, v. 60, p. 460–470, 2 2017. ISSN 0956-053X. Citado na página 12.

PRODANOV, C. C.; FREITAS, E. C. d. *Metodologia do Trabalho Científico: métodos e técnicas da pesquisa e do trabalho acadêmico*. 2ª edição. ed. 2013, 2013. 277 p. ISBN 978-85-7717-158-3. Disponível em: ⟨https://www.researchgate.net/publication/359118465_Metodologia_do_Trabalho_Cientifico_metodos_e_tecnicas_da_pesquisa_e_do_trabalho_academico⟩. Citado na página 14.

RAJESWARI, M. et al. Detection of tyre defectsusingweighted quality-based convolutional neural network. *Soft Computing*, Springer Science and Business Media Deutschland GmbH, v. 26, n. 9, p. 4261–4273, 5 2022. ISSN 14337479. Citado na página 11.

RUAN, G. et al. Exploring the transferability of wheat nitrogen status estimation with multisource data and Evolutionary Algorithm-Deep Learning (EA-DL) framework. *European Journal of Agronomy*, Elsevier B.V., v. 143, 2 2023. ISSN 11610301. Citado na página 13.

SANTOS, J. I. et al. Explainable machine learning for project managemenSantos, J. I., Pereda, M., Ahedo, V., & Galán, J. M. (2023). Explainable machine learning for project management control. Computers & Industrial Engineering, 180, 109261. https://doi.org/10.1016/J.CIE.2023. *Computers & Industrial Engineering*, Pergamon, v. 180, p. 109261, 6 2023. ISSN 0360-8352. Citado na página 13.

WANG, C. et al. Direct conversion of waste tires into three-dimensional graphene. *Energy Storage Materials*, Elsevier, v. 23, p. 499–507, 12 2019. ISSN 2405-8297. Citado na página 12.

XU, N. et al. Tire Force Estimation in Intelligent Tires Using Machine Learning. *IEEE Transactions on Intelligent Transportation Systems*, Institute of Electrical and Electronics Engineers Inc., v. 23, n. 4, p. 3565–3574, 4 2022. ISSN 15580016. Citado na página 11.

ZHAN, P. et al. Temporal anomaly detection on IIoT-enabled manufacturing. *Journal of Intelligent Manufacturing*, Springer, v. 32, n. 6, p. 1669–1678, 8 2021. ISSN 15728145. Disponível em: ⟨https://link.springer.com/article/10.1007/s10845-021-01768-1⟩. Citado na página 10.

ZHU, J.; HAN, K.; WANG, S. Automobile tire life prediction based on image processing and machine learning technology:. *https://doi-org.ez10.periodicos.capes.gov.br/10.1177/16878140211002727*, SAGE PublicationsSage UK: London, England, v. 13, n. 3, p. 1–13, 3 2021. ISSN 16878140. Disponível em: ⟨https://journals-sagepub-com.ez10.periodicos.capes.gov.br/doi/10.1177/16878140211002727⟩. Citado na página 12.

ZHU, Q.; LIU, Z.; YAN, J. Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks. *Computational Mechanics*, Springer Science and Business Media Deutschland GmbH, v. 67, n. 2, p. 619–635, 2 2021. ISSN 14320924. Citado na página 11.

Anexos

INTERNATIONAL JOURNAL OF COMPUTER INTEGRATED MANUFACTURING
https://doi.org/10.1080/0951192X.2023.2177734

Machine learning algorithms applied to intelligent tyre manufacturing

Simone Massulini Acosta [a], Rodrigo Marcel Araujo Oliveira [b] and Ângelo Márcio Oliveira Sant'Anna [b]

[a]Academic Department of Electronics, Federal University of Technology Paraná, Curitiba, Brazil; [b]Polytechnic School, Federal University of Bahia, Salvador, Brazil

ABSTRACT

Intelligent manufacturing is a way to expand industrial manufacturing by integrating artificial intelligence and device technologies to provide great solutions to solve complex problems and improve industrial processes. Artificial intelligence has been used in intelligent manufacturing for monitoring and optimization processes, focusing on improving efficiency. This paper examines the predictive performance of six machine learning algorithms for modeling tyre weight in smart tire manufacturing from real data. The main contribution of this research is developing a scheme solution that uses machine learning algorithms to industrial processes in stored data large manufacturing processes, allowing the process engineer to manage the finished products and the process parameters. The proposed relevance vector machine is compared with other algorithms such as support vector machine, artificial neural network, k-nearest neighbors, random forest, and model trees. RVM algorithm presented the smallest measures of squared error and better performance than the other algorithms. This novel approach accurately predicts tyre weight patterns during production using machine learning algorithms to analyze relevant features and detect anomalies based on predicted process data.

ARTICLE HISTORY
Received 30 October 2021
Accepted 1 February 2023

KEYWORDS
Artificial intelligence; machine learning; intelligent manufacturing; tyre; industrial process

1. Introduction

Industry 4.0 is the recent movement on intelligent technology with the challenge of integrating several concepts such as the internet of things, cyber-physical systems, artificial intelligence, cloud-based manufacturing, and Big Data. Sanchez, Exposito, and Aguilar (2020) asserted that Industry 4.0 is a concept in increasing development that integrate intelligent technologies and with many challenges to solve problems in the manufacturing industry. Intelligent manufacturing is a way to expand industrial manufacturing, integrating sensors and automation to provide great solutions to solve complex problems to improve industrial processes.

Recent research with machine learning applications for intelligent process systems has been proposed in the literature. Onan (2015) used a fuzzy-rough nearest neighbor algorithm to classify breast cancer diagnosis. The performance metrics such as F-measure and kappa statistics indicated good results compared to other literature methods. Onan (2016) presented a comparative analysis of ensemble methods for web page classification using four different algorithms (boosting, bagging, dagging, and random subspace) with other machine learning algorithms. The results showed that this study achieved better predictive performance. Onan (2016) presented an empirical analysis of five statistical extraction methods and ensemble learning methods for the predictive performance of text classification. The study indicated that classification algorithms integrated with ensemble methods presented the highest average predictive performance but with the highest average running time analysis. Onan (2017) proposed a hybrid supervised clustering based on cuckoo search algorithm and k-means method for text document classification. The results indicated the good predictive performance of the proposed classifier ensemble model compared to the conventional classification algorithms and ensemble learning methods for text classification benchmarks. Onan and Korukoglu (2017) presented an ensemble approach for feature selection based on naive bayes and K-nearest neighbor algorithms integrated with genetic algorithm. These authors asserted that this approach is an efficient method for sentiment analysis. Onan (2018) presented a comparative analysis using different machine learning algorithms such as

CONTACT Ângelo Márcio Oliveira Sant'Anna angelo.santanna@ufba.br Polytechnic School, Federal University of Bahia, Salvador, Brazil

naive bayes, support vector machines, logistic regression, k-nearest neighbor, and random forest integrated with the ensemble learning methods as boosting, bagging, and random subspace for classification of different feature engineering. Peres et al. (2018) developed a framework based on machine learning algorithm to analyze and supervised real-time data from cyber-physical production system. The k-means and logistic regression algorithms were used to process the gathered shop-floor data to provide decision-making in predictive maintenance.

Soto, Tavakolizadeh, and Gyulai (2019) proposed a framework for detection of failures of surface electronic devices. They used a neural network, random forest, and gradient boosting algorithms to identify product failure and presented a better flexible learning solution to the production processes. Onan (2019a) proposed a recurrent neural network model for opinion mining on instructor evaluation reviews. The results indicated good performance using deep learning model when compared with ensemble learning and supervised learning for sentiment analysis. Onan (2019b) proposed a framework for extraction methods from scientific literature. They presented a two-stage procedure in conjunction with cluster method for extracting the word embedding schemes, and it showed better predictive performance when compared to the baseline clustering method. Onan (2019c) used different algorithms such as naive bayes, support vector machines, logistic regression, k-nearest neighbor, and random forest with ensemble learning methods such as boosting, bagging, and random subspace for consensus clustering contexts with an imbalance data problem.

Zhu, Yan, and Lu (2020) presented a fault diagnosis approach for online detection in disc slitting machines based on wavelet transform and support vector machine algorithms. The results showed that this method could effectively identify the faults with good accuracy. Francisco, Canciglieri Junior, and Sant'anna (2020) presented a review combined learning algorithms with different metaheuristics as genetic algorithms, particle swarm optimization, firefly algorithm, cuckoo search algorithm, and bat algorithm for biomedical text classification. Onan (2020) proposed a deep learning-based approach for sentiment analysis using a convolutional neural network and long short-term memory. This method showed better predictive performance than the traditional deep learning algorithm. Cardin et al. (2020) proposed a framework to implement energy-aware resources digital twin on injection molding machines. They presented a procedure to estimate the best design for thermoplastic machines' physical and geometric parameters, energy consumption, materials, and setpoint temperature. Redelinghuys, Kruger, and Basson (2020) used a connected architecture of multiple digital twins to evaluate power consumption and variations of each component in a manufacturing cell scenario. The proposed architecture allowed to obtain the flow of information and the interconnections with digital twins.Morariu et al. (2020) presented a prediction method for production planning and predictive maintenance based on machine learning algorithms. The long short-term memory algorithm was used to predict the energy consumption patterns from collected data about operation execution time and operation energy consumption for each resource. The results indicated good predictive production from deep learning algorithms compared to traditional production planning.

Onan and Toçoglu (2021) used a long short-term memory learning algorithm to identify sarcastic text documents. The results demonstrated that this deep learning algorithm could be used in natural language processing for the sarcasm identification task. Acosta et al. (2021) combined a relevance vector machine algorithm with a novel self-adaptive differential evolution metaheuristic for predictive modeling in a steelmaking process. This study described the performance of the proposed algorithm with the random forest, artificial neural network, and k-nearest neighbors algorithms. Farahani et al. (2021) used ten machine learning algorithms to analyze quality monitoring systems from in-mold sensors and machine data. They proposed an approach to predict the best weight, diameter, and thickness value from injection molding processes. Acosta and Sant'anna (2022) presented an approach for monitoring product failures based on machine learning algorithms integrated with the control chart method. The results showed that the relevance vector machine performs better than the support vector machine and artificial neural network.

Although there have been several studies on the applications of machine learning algorithms for different knowledge areas, this paper examines the predictive performance of six machine learning

algorithms for tyre weight modeling in intelligent tyre manufacturing from real data. The main contribution of this research is developing a scheme solution that uses machine learning algorithms to industrial processes in stored data large manufacturing processes, allowing the process engineer to manage the finished products and the process parameters. The proposed relevance vector machine is compared with other algorithms such as support vector machine, artificial neural network, k-nearest neighbors, random forest, and model trees.

2. Machine learning algorithms

2.1. Artificial neural networks

Artificial Neural Networks (ANN) is a machine learning algorithm for classification and regression most used based on the behavior of the human nervous system. ANN are neural network architectures that process data categorized into output and hidden layers, whereby a special form of a mathematical process as a decision rule to minimize the error (Onan and Toçoglu 2021). The hidden layers consist of several layers connected as feature networks obtained from learning any input/output variables of the continuous form (Morariu et al. 2020). In ANN, each output is calculated by function over each layer from input/output data. This way, the performance is calculated based on all the layers' calculations. The ANN architecture is obtained based on the length of the input data (Onan 2020)

2.2. Support vector machine

Support Vector Machine (SVM) is a machine learning algorithm for classifying linear and nonlinear regression, in which the input data are mapped via a nonlinear function. SVMs aim to identify an optimal decision region for partitioning the data into classes (Onan 2017). The performance of SVM generalization depends on the correct specification of the free hyperparameters, the value of the ε-insensitivity, regularization parameter C, and kernel parameters. SVMs have relatively high predictive performance, and it is less vulnerable to the overfitting problem (Onan 2018).

2.3. K-nearest neighbors

K-nearest neighbor (KNN) is a machine learning algorithm for classification based on instance in which such nearest neighbors are conceptually straightforward approaches to approximating real-valued or discrete-valued target functions. The K-nearest neighbor algorithm presents a simple structure and uses a limited set of parameters. The KNN builds the classification based on the similarity between the k closest training instances of a particular instance (Onan and Korukoglu 2017). N-dimensional features represent the training instances, and their correspond to a single point in n-dimensional space (Onan, 2019a).

2.4. Random forest

Random Forest algorithm (RF) is a machine learning algorithm for classification and regression tasks that combines tree predictors induced from random vector samples (Breiman 2001). In the RF, each tree is built based on a bootstrap sample drawn randomly from the original dataset. In the algorithm, the classifier's error depends on the power of the individual trees and their correlations. The tree classification process consists of a random feature selection, increasing the model's capacity to predict original and noisy samples (Onan, Korukoglu, and Bulut 2016). The random feature selection method is used in the tree construction phase, which enables predictive performance (Onan, 2019b).

2.5. Model trees

Model Trees (ML) is a statistical learning algorithm for regression or classification. It predicts the parameter space into areas (subspaces) and builds a local specialized linear regression model in each of them. ML algorithm was originally proposed by Quinlan (1992). The algorithm provides a scheme to apply linear regression based on tree model, which can predict continuous numeric attributes (Sant'anna 2015). Model trees aim to optimize the tree structure function. This model uses the decision tree rule as a regression classifier for training and choosing parameters to create a hybrid regression model that produces good estimates and insights applied to the patterns which accomplish that rule (Abolfathi et al. 2016).

2.6. Relevance vector machine

Relevance vector machine (RVM) is a machine learning algorithm based on Bayesian sparse kernel function for regression and classification. The algorithm can build relevance learning regressors that can be used to solve linear and nonlinear problems. RVMs can be successfully applied in various regression tasks in several knowledge areas (Acosta et al. 2021). SVM can perform well with a high-dimensional feature dataset to determine the best direction of the kernel functions and their parameters. In addition, RVM adopts a fully probabilistic structure and introduces a prior over-the-model weight oriented by a set of hyperparameters. It's associated with each weight, whose most probable values are iteratively estimated from the dataset (Acosta and Sant'anna 2022).

2.7. Performance measures

The performance of machine learning models depends heavily on the choice of your hyperparameters. In recent research, many practitioners have selected the hyperparameters for machine learning empirically or used a grid search technique together with a cross-validation method. Apart from consuming enormous amounts of time, such procedures for selecting the hyperparameters may not result in the best performance.

This study uses grid search and cross-validation methods to select the optimal parameters for the RVM and SVM models with the Gaussian RBF and Laplacian kernel functions. We use grid search and cross-validation methods to choose the optimal parameters for the ANN, RF, MT, and K-NN models. Regression models with good fits present little discrepancies between the observed and predicted values. The adequacy of a model is also essential because the relation between the response and the factors should be significant and independent of the number or type of input variables. The standard regression model assumes that the residuals are independent and identically distributed (i.i.d) normal random variables with zero mean and constant variance.

The generalization of the machine learning algorithms is based on minimization strategies: the root mean square error (RMSE) (Eq. 1), mean absolute error (MAE) (Eq. 2), mean absolute percentage error (MAPE) (Eq. 3), where y_i is the observed value, $\hat{y}_i$ is the predicted value and n is the number of observations. The best model will have the lowest RMSE, MAE and MAPE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2} \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}| \tag{2}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{\hat{y}_i}\right| \times 100 \tag{3}$$

3. Real case study

The present work was carried out in a global tyre industry that has been in the market since the 19th century and is in 53 countries. This factory manufactures tyres for passenger cars and trucks. The study variable is tyre weight which is an indicator of great importance for the tyre company due to rolling resistance, drivability, and legal issues. In the last years, the development of tyres in the world has been moving towards control over your overall weight.

To develop this study, we used a database collected from the information system in intelligent manufacturing of the company. Figure 1 illustrates the tyre anatomy (a) and equipment with sensors (b) that measure the tyre features. The study includes the response variable and six control variables: tyre weight (y_1). The control variables are: tread (x_1), steel belt radial (x_2), tyre sidewall (x_3), body ply (x_4), inner linear (x_5), and bead wire (x_6).

The measurement time of tyre attributes is every ten seconds. The company's information system stored these measures in a database to be analyzed, allowing the process engineer to manage the finished products and the process parameters. The selected database covers a sample of 178 observations. The statistical summary of tyre weight (Kg) were minimum = 55.75, maximum = 58.35, mean = 56.80, standard deviation = 0.5598, and coefficient of variation = 0.96%. Simulations and calculations were performed with the open-source software R® (R 2021). All programs ran on a personal computer with an Intel Core i7-2670QM, 2.2 GHz, 8 GB DDR3–1333 SDRAM, Windows 7 Professional 64-bit.

The 178 observations were normalized into the interval [0,1]. The observation set was then randomly

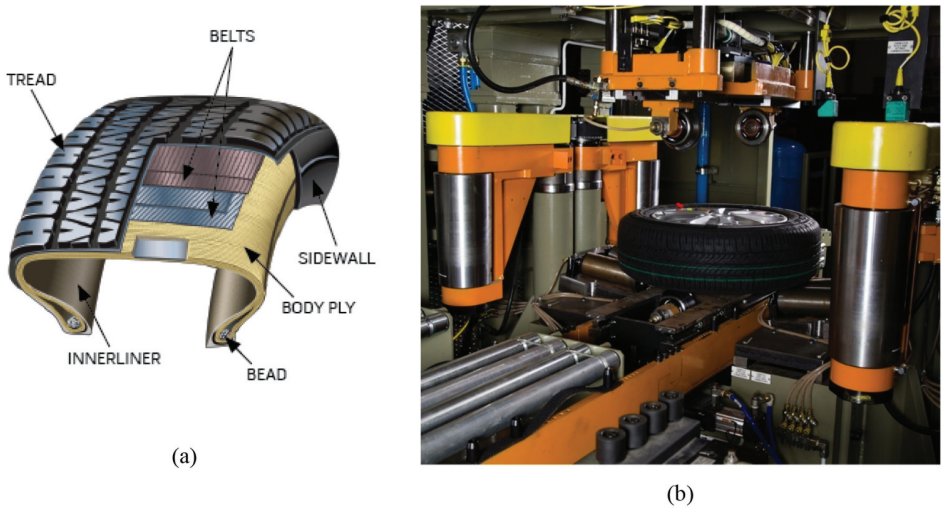


Figure 1. Tyre anatomy (a) and equipment with read sensors (b).

divided into two parts: a training dataset composed of 142 (80%) observations and a test dataset composed of 36 (20%) observations. The size of the training dataset was divided into two subsets (training and validation), with the validation dataset size set to be 20% of the original training dataset. The training dataset was used to estimate the machine learning models in the tyre manufacturing process, and the test dataset was used to evaluate and compare the predictive power of the machine learning models.

The search space of the control parameters for the RVM is $\gamma \in [0.001; 1]$, $u \in [0; 10]$ and $d \in [1; 5]$. The RVM model selected is the model optimized by the DE algorithm using the Laplacian kernel with the selected optimal kernel parameter $\gamma = 0.01576$, and the number of RVs = 26. The number of RVs is 18.3% of the training dataset, which can indicate the model's goodness of fit and adequacy.

To tune the SVM parameters with the RBF kernel using the training data, we first used the procedure that Cherkassky and Ma (2004) proposed4) to obtain the three free hyperparameters (C, ε and RBF kernel parameter γ) and identify the best search region. The search space of the control parameters for the SVM is: $C \in [1; 50]$, $\varepsilon \in [0.001; 1]$, $\gamma \in [0.001; 1]$, $u \in [0; 10]$ and $d \in [1; 5]$. The SVM model selected is the model optimized by the DE algorithm using RBF kernel with the selected optimal parameters: $C = 3.7873$, $\varepsilon = 0.1148$ and $\gamma = 0.2591$, and the number of SVs = 113.

The RVR model produced fewer RVs (26) compared to the number of SVs (113) in the SVM model. The number of SVs is 79.5% of the training dataset, and the number of SVs is nearly four times greater than the number of RVs. For the ANN model, we selected the multilayer perceptron network (MLP) with resilient backpropagation (Rprop) learning function, 74 neurons in the one hidden layer, and the logistic activation function. Haykin (2009) suggests a decision rule to minimize the error in the training data based on the general induction principle.

For the random forest (RF) model, we selected four hyperparameters using a grid search technique and cross-validation method to find the optimal hyperparameters that have the smallest RMSE of the OOB data prediction, such as: the number of features (mtry = 6) randomly chosen at each split, the number of trees (ntree = 390) in the forest, the number of samples to train on (sampsize = 113), and the minimum number of samples within the terminal nodes (nodesize = 6).

For the K-nearest neighbors (K-NN) model, we used a cross-validation method to select the optimal parameter of K-NN, as k = 5. This K-NN algorithm was based on the proximity of the nearest neighbor search with the lowest Bayes error rate. The model tree (MT) used is the M5 rule-based model with boosting and corrections based on nearest neighbors in the training dataset. The Model tree tunable hyperparameters were selected by cross-validation method using the number of training committees was 3, and the number of neighbors for prediction was 5.

This study evaluated the machine learning models' performance using residual analysis and error minimization strategies. We tested the normality of the

residuals of the fitted models using the Shapiro – Wilk test for the training data. We obtained a *p*-value higher than 0.4 for the two models, which indicates that the residuals are normally distributed. We examined the autocorrelation of the residuals using the Durbin-Watson test for the training data, and the results indicate no significant correlations in the models' residuals. We used the Levene test to check homoscedasticity and obtained a *p*-value higher than 0.5, which means that residuals can be considered as having constant variances. After these tests, we concluded that the residuals of the RVM and SVM models are independent and identically distributed (i.i.d) normal random variables with constant variances. It is evidence of the goodness of the fits and shows that the models are appropriate for the observations.

Table 1 summarizes the statistical measures of the error minimization results (RMSE, MAE, MAPE) for the training dataset. In Table 1 can be seen that the RVR model has smaller values of $RMSE_{Avg}$, MAE_{Avg}, and $MAPE_{Avg}$. Table 2 summarizes the statistical measures of the error minimization results (RMSE, MAE, MAPE) for the test dataset. These results note substantial agreement between the training results (Table 1) and the test results (Table 2), indicating no overfitting problems with models. The most hit rate suggests a better performance of the model. Based on the ascending order are: to $RMSE_{Avg}$, MAE_{Avg}, and $MAPE_{Avg}$ is: RVM < SVM < ANN < MT < K-NN < RF.

The RVR model is 75% when the error bound is within ±0.2 kg, and 97.22% when the error bound is within ±0.4 kg. The verification result reveals that the RVR model has higher prediction accuracy than other models.

Figure 2 shows the predicted values against the observed values for the training and test datasets for the models. In Figure 2, we note that there is a substantial agreement between the training and test results, indicating no overfitting problems with the models and that the models possessed high accuracy and generalization ability. This goodness of fit graphics confirms the good predictive modeling performance, and the RVR model had better performance than other models. Based on the results of the tables ($RMSE_{Avg}$, MAE_{Avg}, $MAPE_{Avg}$) and the figure, we can define the ascending order of models' performance in the prediction of the tyre weight: RVM < SVM < ANN < MT < RF < K-NN.

There are some advantages associated with RVM. The generalization performance of RVM is comparable to an equivalent SVM. Furthermore, RVM produces probabilistic output, and there is no need to tune the regularization parameter C and the insensitivity parameter ε necessary in SVM. RVM yields sparse models with fewer relevance vectors (Tipping 2000). Compared with traditional ANNs, SVR possesses some advantages: it has high generalization capability and avoids local minima, it always has a solution, does not need the network topology to be determined in advance, and it has a simple geometric interpretation and provides a sparse solution. SVR provides good performance when the model parameters are well-tuned (Wang et al. 2003; Tipping 2000).

Statistical tests are employed to give a detailed analysis of the performance differences among all the regression models. Parametric tests assume a series of hypotheses on the data applied to independence, normality, and homoscedasticity. If such

Table 1. Statistical measures of error minimization results for training dataset.

Model	$RMSE_{Avg}$	$RMSE_{SD}$	MAE_{Avg}	MAE_{SD}	$MAPE_{Avg}$	$MAPE_{SD}$
RVR	0.163399	0.0107693	0.125834	0.0092774	0.222101	0.0163967
SVR	0.173906	0.0068925	0.128568	0.0049625	0.227011	0.0088135
ANN	0.173692	0.0159460	0.133582	0.0124746	0.235976	0.0220565
MT	0.182805	0.0093905	0.138738	0.0055125	0.244992	0.0097566
RF	0.275179	0.0117261	0.211048	0.0085683	0.372507	0.0151278
K-NN	0.240877	0.0093025	0.192042	0.0075796	0.338737	0.0133910

Table 2. Statistical measures of error minimization results for test dataset.

Model	$RMSE_{Avg}$	$RMSE_{SD}$	MAE_{Avg}	MAE_{SD}	$MAPE_{Avg}$	$MAPE_{SD}$
RVR	0.218828	0.0318997	0.162923	0.0234802	0.287043	0.0415885
SVR	0.249866	0.0255882	0.205831	0.0173129	0.363514	0.0305360
ANN	0.250582	0.0261979	0.209722	0.0193658	0.370352	0.0343186
MT	0.265495	0.0391840	0.221143	0.0274856	0.389902	0.0485732
RF	0.297848	0.0488134	0.243925	0.0287436	0.430206	0.0509702
K-NN	0.287906	0.0489148	0.221756	0.0286839	0.391523	0.0508863

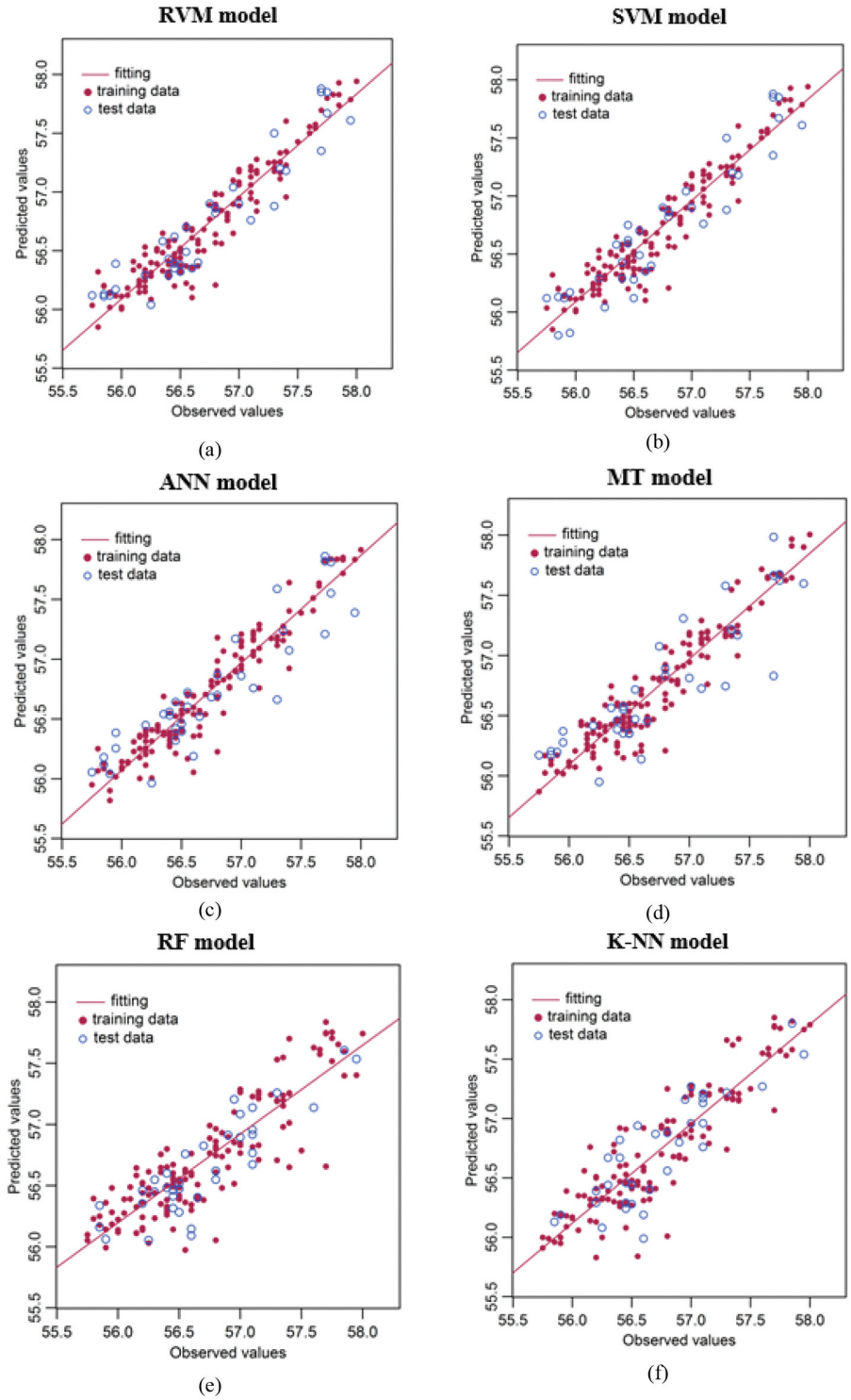


Figure 2. Predicted against observed values for the training and test dataset: (a) RVM, (b) SVM, (c) ANN, (d) MT, (e) RF, (f) K-NN models.

assumptions do not hold, the reliability of the tests is not guaranteed. Nonparametric tests can perform two classes of analysis: pairwise and multiple comparisons (Derrac et al. 2011; Latorre et al. 2020).

The Friedman test is a nonparametric test analog to the parametric two-way variance analysis (Garcia et al. 2010). The null hypothesis states that all the models have the same performance (Carrasco et al. 2020). Once Friedman's test rejects the null hypothesis, we

can proceed with the Bergmann-Hommel post-hoc test to find the pairs of models that produce differences ($N \times N$ comparisons) (Derrac et al. 2011). We used the Friedman test with the six models (RVR, SVR, ANN, MT, RF, and K-NN), and the p-value reported by this test is 0.0058, which is significant at the significance level ($\alpha = 0.05$). Then, we perform the Bergmann-Hommel post-hoc test to determine the location of the differences between these models.

Figure 3 shows the adjusted p-values using the Bergmann-Hommel post-hoc test for multiple comparisons. The null hypothesis is rejected if the adjusted p-value is less than the significance level ($\alpha = 0.05$). The p-values below 0.05 indicate that the respective models differ significantly in prediction errors. Figure 4 shows predicted against observed values for the test dataset for RVM, SVM, ANN, MT, RF, K-NN models. We can observe statistically significant differences between RVM with MT, K-NN, and RF. Nonetheless, the difference between RVM with SVM and ANN is not statistically significant.

3.1. Discussion

The results show the machine learning applications that can be used for several intelligent manufacturing. The case study tries to resolve the prediction problem, and the insights obtained can be used in the pattern recognition or fault detection process. Moreover, the prediction problem in intelligent manufacturing is

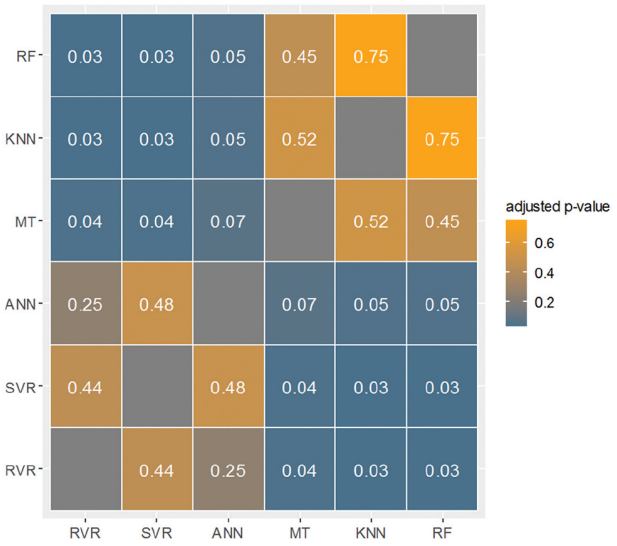


Figure 3. Adjusted p-values using the Bergmann-Hommel post-hoc test for multiple comparisons.

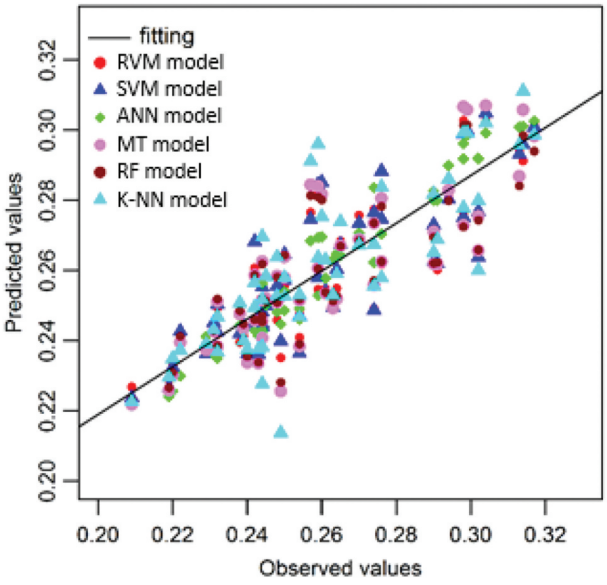


Figure 4. Predicted values against observed values for the test dataset for the proposed machine learning.

characterized by relevant uses of machine learning algorithms to learn patterns and to detect variations in the industrial processes, i.e. tyre failures or nonconforming products, and then use the machine learning to make predictions for that data.

One of the major causes of unconformity in the tyre manufacturing process is the tyre weight. The main product attributes found in tyre industrial process were studied: tread, steel belt radial, tyre sidewall, body ply, inner linear, and bead wire, and they can severely affect the physical integrity of the tyres. The overall proposed machine learning algorithms for predicting tyre weight were achieved based on the following target values to control variables with a tolerance of 0.825 mm. Another important result of this work was to define the optimal value to output variable: tyre weight 57.00 (±0.04184) Kg, improving the manufacturing process performance. This study was concerned with predicting the tyre weight in the final production process of one tyre industry.

4. Conclusions

In the last years were published applications of machine learning for modeling and predicting different intelligent manufacturing. Artificial intelligence has been used in advanced manufacturing for detection and optimization processes, focusing

on improving efficiency. The most compelling feature of the relevant vector machine is that it typically utilizes fewer kernel functions while capable of generalization performance superior to other machine learning algorithms. This paper examines the predictive performance of six machine learning algorithms for tyre weight modeling in intelligent manufacturing from real data. The main contribution of this research is developing a scheme solution that uses machine learning algorithms to industrial processes in stored data large manufacturing processes, allowing the process engineer to manage the finished products and the process parameters. The proposed relevance vector machine is compared with other algorithms such as support vector machine, artificial neural network, k-nearest neighbors, random forest, and model trees.

Building machine learning models can be used to predict outputs variables based on various process variables and to explain the phenomenon underlying the collected data. The RVM algorithm performance is compared with the SVM, ANN, MT, K-NN, and RF models. RVM has nearly the same performance as SVM, but RVM produced almost four times fewer RVs than the SVs produced by SVM. Furthermore, the tuning of RVM involves only the kernel parameters, whereas SVM has more parameters for tuning (C, ε and kernel parameters). As observed in the tables, compared with the statistical results of the other models, the RVM model achieves better results on the $RMSE_{Avg}$, MAE_{Avg}, $MAPE_{Avg}$. This means that the RVM model had better performance than other models.

The Friedman and Bergmann-Hommel post-hoc tests were used, and the results showed significant differences between the RVM with MT, K-NN and RF. The difference is not significant between RVM, SVM, and ANN. This novel approach accurately predicts tyre weight patterns during production using real-time machine learning algorithms to analyze the tire's relevant features and detect anomalies based on predicted process data.

Future work is focused on further developing the machine learning algorithm presented in this paper to allow the modeling and prediction of tyre features in intelligent manufacturing. From univariate to multivariate analysis has the potential to show some interesting correlations between the predictive performance on multiple dimensions can be a good research direction. By taking the machine learning algorithms and performance measures reported in this paper as a basis, it should be beneficial to development of internet of things and cloud-based manufacturing of resources and manufacturing operations and embedding them in the machine learning algorithms for process monitoring and predictive maintenance.

Disclosure statement

The authors declare that no known competing financial interests or personal relationships could have appeared to influence the work reported in this paper.

Funding

The work was supported by the CNPQ [309812/2021-6].

ORCID

Simone Massulini Acosta http://orcid.org/0000-0003-3674-5576
Rodrigo Marcel Araujo Oliveira http://orcid.org/0000-0002-2646-2882
Ângelo Márcio Oliveira Sant'Anna http://orcid.org/0000-0001-8332-8877

References

Abolfathi, S., A. Yeganeh-Bakhtiary, S. M. Hamze-Ziabari, and S. Borzooei. 2016. "Wave Runup Prediction Using M5′ Model Tree Algorithm." *Ocean Engineering* 112: 76–81. doi:10.1016/j.oceaneng.2015.12.016.

Acosta, S. M., A. L. Amoroso, A. M. O. Sant Anna, and O. Canciglieri Junior. 2021. "Relevance Vector Machine with Tuning Based on Self-Adaptive Differential Evolution Approach for Predictive Modelling of a Chemical Process." *Applied Mathematical Modelling* 95: 125–142. doi:10.1016/j.apm.2021.01.057.

Acosta, S. M., and A. M. O. Sant'anna. 2022. "Machine Learning-Based Control Charts for Monitoring Fraction Nonconforming Product in Smart Manufacturing." *International Journal of Quality & Reliability Management* in press. doi:10.1108/IJQRM-07-2021-0210.

Breiman, L. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32. doi:10.1023/A:1010933404324.

Cardin, O., P. Castagna, D. Couedel, C. Plot, J. Launay, N. Allanic, Y. Madec, and S. Jegouzo. 2020. "Energy-Aware Resources in

Digital Twin: The Case of Injection Moulding Machines. In *SOHOMA'19, Studies in Computational Intelligence*, edited by Borangiu, T, Trentesaux, D, Leitão, P, Giret Boggino, A, Botti, V , Vol. 853, 183–194. Cham: Springer. doi:10.1007/978-3-030-27477-1_14

Carrasco, J., S. García, M. M. Ruedab, S. Dasc, and F. Herrera. 2020. "Recent Trends in the Use of Statistical Tests for Comparing Swarm and Evolutionary Computing Algorithms: Practical Guidelines and a Critical Review." *Swarm and Evolutionary Computation* 54: 1–20. doi:10.1016/j.swevo.2020.100665.

Cherkassky, V., and Y. Ma. 2004. "Practical Selection of SVM Parameters and Noise Estimation for SVM Regression." *Neural Networks* 17 (1): 113–126. doi:10.1016/S0893-6080(03)00169-2.

Derrac, J., S. García, D. Molina, and F. Herrera. 2011. "A Practical Tutorial on the Use of Nonparametric Statistical Tests as a Methodology for Comparing Evolutionary and Swarm Intelligence Algorithms." *Swarm and Evolutionary Computation* 1 (1): 3–18. doi:10.1016/j.swevo.2011.02.002.

Farahani, S., B. Xu, Z. Filipi, and S. Pilla. 2021. "A Machine Learning Approach to Quality Monitoring of Injection Molding Process Using Regression Models." *International Journal of Computer Integrated Manufacturing* 34 (11): 1223–1236. doi:10.1080/0951192X.2021.1963485.

Francisco, M. G., O. Canciglieri Junior, and Â. M. O. Sant'anna. 2020. "Design for Six Sigma Integrated Product Development Reference Model Through Systematic Review." *International Journal of Lean Six Sigma* 11 (4): 767–795. doi:10.1108/IJLSS-05-2019-0052.

Garcia, S., A. Fernández, J. Luengo, and F. Herrera. 2010. "Advanced Nonparametric Tests for Multiple Comparisons in the Design of Experiments in Computational Intelligence and Data Mining: Experimental Analysis of Power." *Information Sciences* 180 (10): 2044–2064. doi:10.1016/j.ins.2009.12.010.

Haykin, S. 2009. *Neural Networks and Learning Machine*. 3rd ed. New York: Prentice Hall.

Latorre, A., D. Molina, E. Osaba, F. Del Ser, and J. Herrera. 2020. "Fairness in Bio-Inspired Optimization Research: A Prescription of Methodological Guidelines for Comparing Meta-Heuristics." *Neural and Evolutionary Computing* 67: 100973. arXiv:2004.09969, doi:10.1016/j.swevo.2021.100973.

Morariu, C., O. Morariu, S. Răileanu, and T. Borangiu. 2020. "Machine Learning for Predictive Scheduling and Resource Allocation in Large Scale Manufacturing Systems." *Computers in Industry* 120: 103244. doi:10.1016/j.compind.2020.103244.

Onan, A. 2015. "A Fuzzy-Rough Nearest Neighbor Classifier Combined with Consistency-Based Subset Evaluation and Instance Selection for Automated Diagnosis of Breast Cancer." *Expert Systems with Applications* 42 (20): 6844–6852. doi:10.1016/j.eswa.2015.05.006.

Onan, A. 2016. "Classifier and Feature Set Ensembles for Web Page Classification." *Journal of Information Science* 42 (2): 150–165. doi:10.1177/0165551515591724.

Onan, A. 2017. "Hybrid Supervised Clustering-Based Ensemble Scheme for Text Classification." *Kybernetes* 46 (2): 330–348. doi:10.1108/K-10-2016-0300.

Onan, A. 2018. "An Ensemble Scheme Based on Language Function Analysis and Feature Engineering for Text Genre Classification." *Journal of Information Science* 44 (1): 28–47. doi:10.1177/0165551516677911.

Onan, A. 2019a. "Consensus Clustering-Based Undersampling Approach to Imbalanced Learning." *Scientific Programming* 5901087: 1–14. doi:10.1155/2019/5901087.

Onan, A. 2019b. "Mining Opinions from Instructor Evaluation Reviews: A Deep Learning Approach." *Computer Applications in Engineering Education* 28 (1): 117–138. doi:10.1002/cae.22179.

Onan, A. 2020. "Sentiment Analysis on Product Reviews Based on Weighted Word Embeddings and Deep Neural Networks." *Concurrency and Computation: Practice and Experience* 33 (23): e5909. doi:10.1002/cpe.5909.

Onan, A., and S. Korukoglu. 2017. "A Feature Selection Model Based on Genetic Rank Aggregation for Text Sentiment Classification." *Journal of Information Science* 43 (1): 25–38. doi:10.1177/0165551515613226.

Onan, A., S. Korukoglu, and H. Bulut. 2016. "Ensemble of Keyword Extraction Methods and Classifiers in Text Classification." *Expert Systems with Applications* 57: 232–247. doi:10.1016/j.eswa.2016.03.045.

Onan, A., and M. A. Toçoglu. 2021. "A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification." *IEEE Access* 9: 7701–7722. doi:10.1109/ACCESS.2021.3049734.

Peres, R. S., A. D. Rocha, P. Leitao, and J. Barata. 2018. "IDARTS – Towards Intelligent Dataanalysis and Real-Time Supervision for Industry 4.0." *Computers in Industry* 101: 138–146. doi:10.1016/j.compind.2018.07.004.

Quinlan, J. R. 1992. "Learning with Continuous Classes." Proceedings of the 5th Australian Joint Conference on Artificial Intelligence, 16-18 November, Hobart. edtited by, Adams, A. and Sterling, L. Singapore: World Scientific. 343–348.

R 2021. "R: A Language and Environment for Statistical Computing." R Foundation for statistical computing. ISBN 3-900051-07-0. Available at http://www.r-project.org .

Redelinghuys, A., K. Kruger, and A. Basson. 2020. "A Six-Layer Architecture for Digital Twins with Aggregation. In *SOHOMA'19, Studies in Computational Intelligence*, edited by Borangiu, T, Trentesaux, D, Leitão, P, Giret Boggino, A, Botti, V.Vol. 853, 171–182Cham: Springer. doi:10.1007/978-3-030-27477-1_13

Sanchez, M., E. Exposito, and J. Aguilar. 2020. "Industry 4.0: Survey from a System Integration Perspective." *International Journal of Computer Integrated Manufacturing* 33 (10): 1017–1041. doi:10.1080/0951192X.2020.1775295.

Sant'anna, A. M. O. 2015. "Framework of Decision in Data Modeling for Quality Improvement." *The TQM Journal* 27 (1): 135–149. doi:10.1108/TQM-06-2013-0066.

Soto, J. A. C., F. Tavakolizadeh, and D. Gyulai. 2019. "An Online Machine Learning Framework for Early Detection of Product Failures in an Industry 4.0 Context." *International Journal of Computer Integrated Manufacturing* 32 (4): 452–465. doi:10.1080/0951192X.2019.1571238.

Tipping, M. E. 2000. "The Relevance Vector Machine." In *Advances in Neural Information Processing Systems*, edited by S. A. Solla, T. K. Leen, and K. Müller, 652–658. Vol. 12. Cambridge: MIT Press.

Wang, W., Z. Xu, W. Lu, and X. Zhang. 2003. "Determination of the Spread Parameter in the Gaussian Kernel for Classification and Regression." *Neurocomputing* 55 (3–4): 643–663. doi:10.1016/S0925-2312(02)00632-X.

Zhu, Y., Q. Yan, and J. Lu. 2020. "Fault Diagnosis Method for Disc Slitting Machine Based on Wavelet Packet Transform and Support Vector Machine." *International Journal of Computer Integrated Manufacturing* 33 (10): 1118–11128. doi:10.1080/0951192X.2020.1795927.

Modelo híbrido de aprendizado de máquina para extração de características em imagens de radiografia de tórax

Rodrigo Marcel Araujo Oliveira
PEI - Escola Politécnica
Universidade Federal da Bahia
Salvador, Bahia
email:rodrigomarcel@ufba.br

Caroline Ribeiro Figueredo
PPGEE - Escola Politécnica
Universidade Federal da Bahia
Salvador, Bahia
email: carolinefigueredo@ufba.br

Luciano Almeida da Silva
PPGEE - Escola Politécnica
Universidade Federal da Bahia
Salvador, Bahia
email:lucianoas@ufba.br

Paulo César Machado de Abreu Farias
PPGEE - Escola Politécnica
Universidade Federal da Bahia
Salvador, Bahia
email:paulo.farias@ufba.br

Ângelo Márcio Oliveira Sant'Anna
PEI - Escola Politécnica
Universidade Federal da Bahia
Salvador, Bahia
email: angelo.santanna@ufba.br

***Resumo*—O diagnóstico de doenças com imagens e auxílio de sistemas computacionais de Inteligência Artificial vêm sendo desenvolvidos e aplicados em vários contextos clínicos. A Indústria 4.0 tem um papel fundamental nesse processo e pesquisas recentes demonstram que transferência de aprendizado de redes neurais convolucionais pré-treinadas como ResNet-50, VGG-16 e Inception podem contribuir significativamente no resultado das predições para classificação de imagens. Contudo, na maioria das vezes, o treinamento de redes neurais requer um tempo de execução considerável. Este artigo propõe um método rápido e eficiente para treinamento com menor tempo de execução baseado em algoritmos de aprendizado de máquina para modelagem preditiva de imagens de radiografia de tórax de pacientes saudáveis e com pneumonia. O trabalho consiste em desenvolver uma modelagem híbrida usando a ResNet-50 para extração de características, a análise de componentes principais para reduzir a dimensão das variáveis preditoras e o modelo *Random Forest* para classificar as imagens. Algoritmos genéticos foram utilizados para otimizar os parâmetros do modelo *Random Forest*. Os resultados dessa abordagem permitiu alcançar valores relevantes de desempenho quando comparado com os encontrados na literatura por arquiteturas robustas de redes neurais. O classificador híbrido *Random Forest* proposto alcançou 93%, 79%, 88% de ROC-AUC nos conjuntos de treino, teste e validação, respectivamente.**

Palavras-chave—ResNet-50, Random Forest, radiografia de tórax, Algoritmos Genéticos, PCA.

I. Introdução

A pneumonia é uma doença infecciosa que se instala nos pulmões, no mundo aproximadamente 1 milhão de crianças morrem de pneumonia, ela é a principal causa de morte de crianças menores de cinco anos [1]. No Brasil, há 4 milhões de casos de pneumonia infantil a cada ano, em que 11% das mortes são em crianças abaixo de um ano e 13% entre crianças de 1 a 4 anos. As taxas de mortalidade infantil por pneumonias variam por região, sendo mais altas nos estados do Norte e Nordeste e mais baixas no Sul [2] [3] [4].

O diagnóstico de pneumonia em crianças pode ser feito a partir do histórico do paciente e do exame clínico, podendo ser confirmado com uma radiografia de tórax. A radiografia de tórax é capaz de fornecer grande quantidade de informações anatômicas e fisiológicas, e é considerada o padrão ouro para diagnóstico de pneumonia aguda [5]. Apesar de ser um exame simples, de ampla disponibilidade, a interpretação dos resultados muitas vezes é difícil, por diversos fatores, tais como a capacidade de interpretação dos profissionais, a qualidade dos equipamentos utilizados, as variações de idade e fisiológicas diversas dos pacientes.

A Indústria 4.0 [6] é caracterizada pela integração de diferentes tecnologias como Inteligência Artificial (IA), robótica, computação em nuvem, etc. As Redes Neurais Convolucionais (CNNs) são métodos eficazes para extrair características de imagens [7]. A aprendizagem profunda é um ramo da IA e tem aplicações em diversas áreas, como na área da saúde para detecção de anomalias em imagens médicas, e pode auxiliar profissionais da saúde na tomada de decisões, permitindo um diagnóstico mais rápido e preciso [8].

A radiografia de tórax é uma ferramenta fundamental para o diagnóstico de doenças, pois permite que os médicos monitorem e avaliem as condições físicas do paciente. Algoritmos de aprendizado de máquina estão cada vez mais sendo utilizados na interpretação e detecção de doenças [9]. Isso permite que médicos e profissionais de saúde tenham previsões sobre o diagnóstico do paciente, com isso realizar triagens de doenças e tomar decisões sobre o tratamento, melhorando a sensibilidade e a especificidade dos resultados.

Atualmente pesquisas apontam que a transfêrencia de aprendizado de redes neurais pré-treinadas são alternativas eficientes

para modelagem preditiva de alto desempenho [7], e entre as redes neurais amplamente utilizadas estão: ResNet-50, VGG-19, DenseNet201, Inception-V3, MobileNet, Xception, etc.

A necessidade de desenvolver métodos computacionais rápidos, eficientes e precisos para diagnosticar doenças como a pneumonia é de interesse público. O objetivo deste trabalho foi classificar imagens de radiografia de tórax para identificar pacientes com pneumonia. Para isso foi utilizada a rede neural ResNet-50 para a extração de características e o modelo de aprendizado de máquina Random Forest para classificação.

Na Seção II, são discutidos tópicos relevantes de pesquisa. Na Seção III, são apresentados os referenciais teóricos utilizados no artigo, abrangendo redes neurais, modelagem preditiva, técnicas de otimização de parâmetros e metodologias de avaliação de modelos. A Seção IV trata do conjunto de dados e da origem das imagens empregadas. Os resultados do modelo nos processos de validação cruzada, treino, teste e validação estão expostos na Seção V. Na Seção VI, são delineadas as conclusões do estudo, bem como as possibilidades para pesquisas futuras.

II. Revisão da literatura

O desenvolvimento de sistemas baseado em IA capazes de identificar padrões e diagnosticar pacientes com diferentes patologias utilizando imagens é um tema presente na literatura. Em [10] os autores desenvolveram uma metodologia para diagnosticar pacientes com pneumopatia pediátrica com auxílio de imagens de radiografia de tórax. Esse sistema também é capaz de fazer a triagem de pacientes com doenças tratáveis comuns da retina que causam cegueira. O conjunto de dados consiste em imagens de Tomografia de Coerência Óptica e de tórax de pacientes. Neste trabalho eles utilizam uma abordagem de aprendizado por transferência da CNN Inception-V3. O desempenho da rede neural desenvolvida é comparável à classificação de especialistas humanos para diagnóstico dessas doenças de degeneração macular relacionada à idade e edema macular diabética. No trabalho [11] os autores desenvolveram uma CNN baseada na arquitetura da VGG-16 para classificação de imagens de Tomografia de Coerência Óptica, a técnica demonstrou a capacidade de distinguir imagens de degeneração macular relacionada à idade comparando com a de indivíduos saudáveis.

Em [6] os autores utilizaram o modelo Random Forest para detecção de anomalias e previsão do peso do pneu de uma fábrica. Nesse estudo um algoritmo de evolução diferencial foi utilizado para otimizar parâmetros de modelos de aprendizado de máquina e isso possibilitou melhores resultados comparado a outros métodos. O modelo Random Forest foi utilizado para classificar imagens com base em características extraídas da CNN ResNet-50 [12]. Esse estudo utilizou dados de imagens de retinopatia diabética e os autores relatam que a proposta adotada superam arquiteturas como: ResNet-50, VGG-19, Inception-v3, MobileNet, Xception e VGG-16.

No artigo [9] é apresentado um método não supervisionado para agrupar imagens de radiografia de tórax com base em medições de similaridade e dissimilaridade. As informações das imagens foram extraídas usando uma CNN, e essas variáveis foram utilizadas para projeção dos dados com algortimo Self-Organizing Map (SOM) em função de variáveis provenientes do processo de redução de dimensionalidade com a análise de componentes principais (PCA). Os autores adotaram o algoritmo K-means para criar os grupos. Isso permitiu o desenvolvimento de uma plataforma de triagem para detecção de tuberculose.

A ResNet-18 foi utilizada para extrair características de imagens de radiografia de tórax de pacientes com tuberculose pulmonar [13]. Os autores desenvolveram uma metodologia baseada em máquinas de comitê (*ensemble*) para classificação de imagens com diferentes configurações e padrões de qualidade. Nesse estudo os autores adotaram a técnica PCA para projeção dos dados em dimensão menor. A rede neural *multilayer perceptron* (MLP) foi adotada como classificador e o treinamento do modelo foi realizado com processo de validação cruzada usando k-folds.

O aumento de dados sintéticos utilizando técnicas de *data-augmentation* com auxílio de redes adversárias generativas (GANs) foi tema dos autores [14]. Nesse trabalho foi realizado uma comparação entre o desempenho de uma CNN com modelo de aprendizado de máquina *One Class Support Vector Machine* para classificação de imagens de pacientes com tuberculose. Os autores relatam ganhos no desempenho do treinamento quando os dados sintéticos foram gerados pela GAN *Pix2Pix*.

Os trabalhos [15], [16], [17], [18] apresentam diversas arquiteturas de redes neurais pré-treinadas para classificação de radiografia de tórax de indivíduos com COVID-19. A transferência de aprendizado com a rede ResNet-50 foi a arquitetura adotada para detecção de COVID-19 em [19]. Nesse artigo foram testadas várias composições de pesos diferentes da rede. Os resultados dos melhores modelos são da ordem de 99% de *accuracy*, *precision* e *recall*. No trabalho [20] a ResNet-50 teve o menor desempenho para classificação radiografia de tórax de pacientes com COVID-19. Os autores relatam que a arquitetura da VGG-16 forneceu o melhor desempenho com 80% de *accuracy*. Esse resultado foi comparado com as redes VGG-19, ResNet-50, DenseNet e Inception-V3.

Os autores do trabalho [21] também apresentaram uma metodologia baseada na técnica GAN para melhorar a triagem de pacientes com COVID-19. O conjunto de dados consiste em imagens de radiografia de tórax. Os autores discutem vários métodos de filtros para pré-processamento dos dados, como: Sobel, Laplaciano do Gaussiano e Gabor. Os resultados do treinamento da rede com a estratégia dos filtros foram comparados com as arquiteturas: AlexNet, GoogleNet, VGG-19, ShuffleNet-V2, DenseNet-121 e DenseNet-201. A metodologia com filtro de Gabor atingiu o melhor desempenho, com apenas 45 épocas o modelo foi capaz de elevar a precisão em até 32%.

Para a classificação de imagens de ressonância magnética de pacientes com COVID-19 com tecidos infectados, os autores [22] propuseram uma metodologia para o pré-processamento das imagens baseada em fractais, que consiste em reduzir a

dimensão e selecionar características relevantes das imagens. Essas informações serviram de entrada para o treinamento de uma rede neural profunda (DNN) e uma CNN proposta pelos autores. A classificação dos resultados mostra que a arquitetura da CNN apresentou os melhores resultados, com *accuracy* de 93,2%, comparado com *accuracy* de 83,4% do método DNN.

III. Metodologia

Nesta sessão serão discutidos os referenciais teóricos das técnicas utilizadas para o desenvolvimento deste artigo. A Figura 1 representa a abordagem sequencial do tratamento dos dados. Os métodos foram implementados com auxílio de uma máquina com processador Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz, intel core i7 7th Gen, 8 GB de RAM e uma placa de vídeo NVIDIA GEFORCE 2 GB. A linguagem de programação adotada foi Python versão 3.9, com auxílio de ferramentas de software livre como Jupyter Notebook. As bibliotecas TensorFlow, scikit-learn e sklearn-genetic-op foram utilizadas para aplicação da ResNet-50, modelagem com Random Forest e otimização de parâmetros, respectivamente.

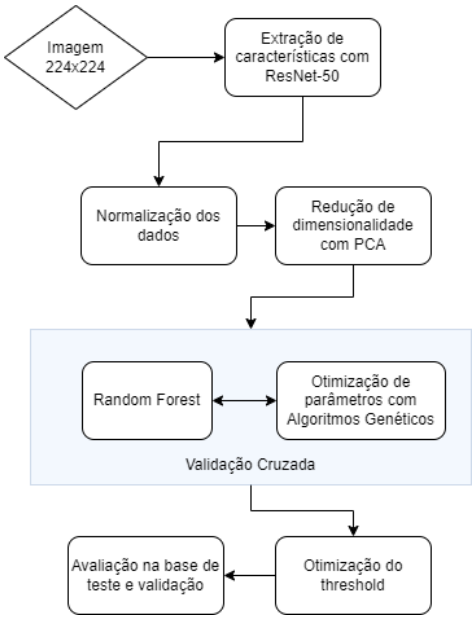


Figura 1. Fluxograma da metodologia.

A. Extração de características

A extração de características é fundamental no processo de classificação de imagens, e esse processo pode ser realizado por meio de uma CNN [19]. O treinamento de redes neurais profundas é um desafio para comunidade científica, e conforme o número de camadas de um modelo aumenta, o número de parâmetros no modelo aumenta significativamente, isso gera uma complexidade e requer recursos computacionais eficientes associados a processamento e memória [23]. Um dos problemas comuns em CNNs é a explosão e o desaparecimento dos gradientes. No processo de retropropagação, a diferenciação da regra da cadeia pode gerar valores de gradientes próximos de zero, e quando multiplicados por mesma magnitude esse valor fica menor, analogamente o mesmo acontece para valores grandes, quando multiplicados os pesos podem gerar valores na ordem dos milhões, consequentemente gera instabilidade numérica dos parâmetros da rede.

A ResNet-50 é uma rede treinada com milhares de imagens do conjunto de dados ImageNet e contém 50 camadas [19]. As características extraídas das camadas mais profundas codificam propriedades específicas, como forma, textura e cor. A ResNet-50 trabalha com conceito de rede residual, permitindo conexões de saltos de alguma camada para outra camada oculta de modo a incorporar o fluxo de informações [12]. O processo permite que as camadas se encaixem em um mapeamento residual, conforme a Figura 2. O diferencial da ResNet comparada a outras redes é que esse mapeamento permite um menor custo computacional, além de demostrar que o resultado da rede não necessariamente está associado a um número grande de camadas ocultas [24]. Os neurônios das camadas ocultas são transformados usando alguma função de ativação, as funções comumente utilizadas no contexto de classificação são: sigmoid; ReLu; tangente hiperbólica; softmax [15]. A equação 1 corresponde a forma como a derivada da função Relu é implementada, onde x representa o valor do dado.

$$ReLu'(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \tag{1}$$

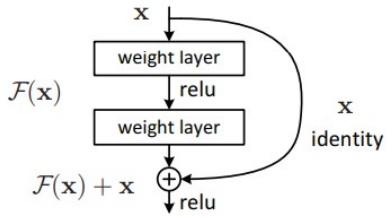


Figura 2. Rede residual da ResNet [24].

A Figura 3 representa a arquitetura da rede, ela contém uma série de camadas convolucionais (conv), as imagens de entrada foram padronizadas para dimensão (224, 224), as camadas intermediárias são processadas de 3 a 6 vezes. A camada de *average pool* é responsável por reduzir os dados de entrada, para isso é construído um mapa de recursos com a finalidade de agrupar as informações das camadas anteriores. No presente trabalho para extração das características são consideradas as transformações até o *average pool* da ResNet-50.

B. Análise de Componentes Principais

A análise de componentes principais (PCA) é uma técnica estatística que pode ser utilizada para explicar relações entre variáveis [25]. O método consiste em transformar os dados originais em outro conjunto de dimensão igual ou menor. O conjunto de componentes principais é formado por combinações lineares das variáveis originais [26]. No caso bidimensional,

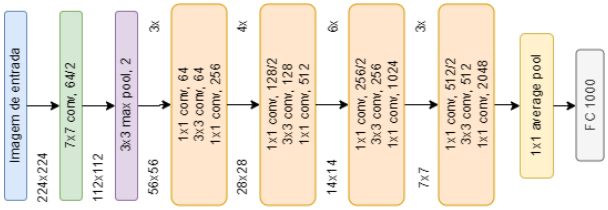


Figura 3. Arquitetura da ResNet-50

a primeira componente principal pode ser expressa conforme a equação 2 de modo a maximizar a variância expressa pela equação 3. O vetor $\mathbf{X}$ corresponde às variáveis explicativas, a matriz de convariância é dada por $\mathbf{\Sigma}$, e $\boldsymbol{\beta}$ são os coeficientes. A segunda componente principal é obtida por a equação 4, de modo a maximizar $\boldsymbol{\beta_2}^T\mathbf{\Sigma}\boldsymbol{\beta_2}$, com as restrições $\boldsymbol{\beta_2}^T\boldsymbol{\beta_2}$=1 e $\boldsymbol{\beta_1}^T\boldsymbol{\beta_2}$=0. Portanto, a soma das variâncias das componentes principais é igual a variância total do sistema conforme a equação 5, e as variâncias denotadas por $\hat{\lambda}_i$ correspondem autovalores da matriz $\mathbf{K}$, e os coeficientes $\hat{\beta}_i$ são os autovetores [27].

$$PC_1 = \boldsymbol{\beta_1}^T\boldsymbol{X} = \beta_{1,1}X_1 + \cdots + \beta_{1,n}X_n. \tag{2}$$

$$V(PC_1) = \boldsymbol{\beta_1}^T\mathbf{\Sigma}\boldsymbol{\beta_1}. \tag{3}$$

$$PC_2{=}\boldsymbol{\beta_2}^T\boldsymbol{X}. \tag{4}$$

$$tr(\boldsymbol{K}) = tr(\sum_{i=1}^{2}\hat{\lambda}_i\hat{\beta}_i\hat{\beta}_i^T). \tag{5}$$

C. Random Forest

Os métodos de árvores de decisão são eficazes para análise preditiva e são mais fáceis de interpretar. Árvores de decisão são modelos de aprendizado de máquina supervisionado que representam regras de decisão baseadas nos valores dos atributos. O algoritmo de *Random Forest* (RF) proposto por [28] é composto por uma combinação de árvores de decisão, que cada árvore é sintetizada a partir de um vetor aleatório amostrado de forma independente com mesma distribuição de probabilidade.

A construção de uma árvore de classificação consiste em determinar as regiões em que o espaço das variáveis preditoras é particionado, com a finalidade de predizer ou classificar uma variável resposta a partir de um conjunto de variáveis preditoras. As árvores do ensemble são construídas por processo de amostragem bootstrap. As florestas aleatórias adotam o bagging proposto com objetivo de reduzir a variância através da predição de vários modelos separados usando diferentes conjuntos dos dados de treinamento. É adotada amostragem com reposição com dois terços dos dados, para depois calcular a média dos resultados das predições. Os dados out-of-bag (OOB) são usados para validação e para definição de relevância dos atributos de entrada, e correspondem a um terço dos dados. As métricas de Gini e de entropia, conforme as equações 6 e 7, são utilizadas como critério para escolha da variável que compõe o nó raiz de cada árvore, onde $\hat{p}_k$ é a probabilidade da classe k. A entropia é uma medida de informação que indica a desordem do sistema, o Gini é uma medida de impureza, ambos possuem valores de mínimo igual a zero, isso indica que o nó da árvore é puro.

$$Entropia = \sum_{i=1}^{k} -\hat{p}_k log(\hat{p}_k). \tag{6}$$

$$Gini = \sum_{i=1}^{k} \hat{p}_k(1-\hat{p}_k). \tag{7}$$

Para problemas de classificação, o erro de generalização depende da ponderação atribuída às árvores individuais e da correlação de saída que há entre essas árvores. No processo de votação das árvores a classe majoritária vence entre as classes previstas [25]. Trata-se de um algoritmo robusto e eficiente para lidar com *overfitting*, o que o torna um modelo excelente para lidar com padrões não lineares como características extraídas da ResNet-50 [12].

D. Algoritmo Genético

O Algoritmo Genético (AG) é um tipo de algoritmo inspirado na evolução natural que é usado em problemas de otimização e busca em geral [29]. Os cromossomos representam essas soluções e podem ser codificados por uma matriz de bits. O conceito de geração, função *fitness*, cruzamento, mutação e elitismo são elementos fundamentais do AG. A geração é caracterizada por uma interação que contém uma população de cromossomos. A seleção dos cromossomos com melhores características está associado a função *fitness* [30]. O cruzamento consiste em selecionar dois cromossomos e gerar um cromossomo com uma nova configuração. O conceito de mutação serve para diversificar os indivíduos da população de modo que os cromossomos gerados no processo de cruzamento não fiquem presos em mínimos locais, esse operador altera com uma probabilidade baixa um dos genes do cromossomo aleatoriamente [31]. O elitismo consiste em selecionar os melhores indivíduos de cada geração, a seleção dos cromossomos pais para o processo de cruzamento está associado a uma distribuição de probabilidade com indivíduos com maior *fitness* da geração. O método GASearchCV [32] da biblioteca *sklearn-genetic-op* foi utilizado para estimar parâmetros do modelo *Random Forest* para maximizar a função *fitness*. Esse método seleciona conjuntos aleatórios de parâmetros e ajusta um modelo para cada conjunto de hiperparâmetros, para cada processo da validação cruzada é calculado o valor da função *fitness*. O algoritmo cria novas gerações combinando a última geração com diferentes abordagens, o processo é repetido até que há convergência considerando o número de gerações ou algum método de parada. A Figura 4 representa o fluxograma desse processo.

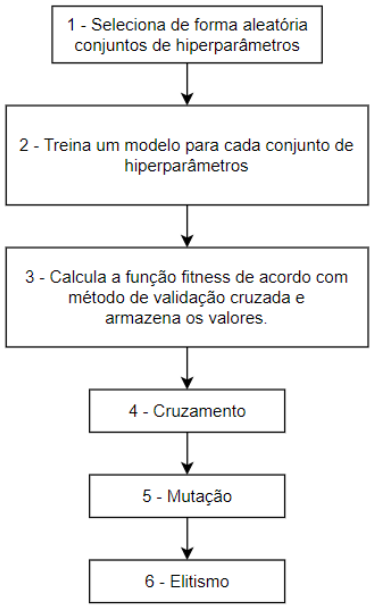


Figura 4. Fluxograma do GASearchCV.

E. Métricas de Desempenho

A avaliação do desempenho do modelo ajustado será baseado nas métricas derivadas da matriz de confusão M [33]. *Accuracy* conforme a equação 8 é correspondente a proporção de predições corretas do modelo. *Precision* expressa pela equação 9 é definida como o número de verdadeiros positivos (*True Positives* - TP) sobre o número de verdadeiros positivos somado ao número de falsos positivos (*False Positives* - FP). *Recall* conforme a equação 10 é definido como o número de verdadeiros positivos sobre o número de verdadeiros positivos somado o número de falsos negativos (*False Negatives* - FN). *Specificity* expressa pela equação 11 corresponde ao número de verdadeiros negativos (*True Negatives* - TN) sobre número de verdadeiros negativos somado ao número de falsos positivos. O F_1-*score* conforme a equação 12 é a média harmônica do *precision* e *recall*. A área sob a curva característica de operação do receptor ROC-AUC foi utilizada como critério de avaliação, em que o eixo das abcissas contém a informação da taxa de falsos positivos, e o eixo das ordenadas a taxa de verdadeiros positivos.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

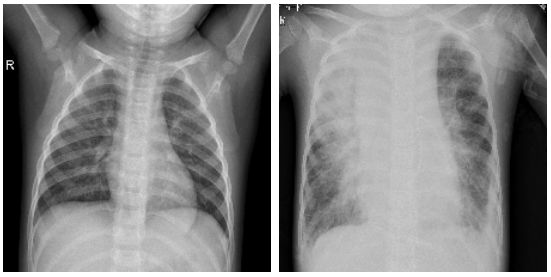
$$Specificity = \frac{TN}{TN + FP} \tag{11}$$

$$F_1score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{12}$$

A técnica k-fold foi utilizada no estudo para validação cruzada. Essa metodologia divide o conjunto de dados em k partes, em cada ajuste uma dessas partes é considerada como conjunto de testes e as demais como treinamento [25].

IV. Estudo de Caso

O conjunto de dados utilizado está disponível em [34], e este possui 5863 imagens de Raios X com três categorias, normal, pneumonia bacteriana e pneumonia viral. As imagens de radiografia de tórax foram selecionadas de cortes retrospectivas de pacientes pediátricos do *Guangzhou Women and Children's Medical Center*. Os diagnósticos das imagens foram classificados por dois médicos especialistas. Para desenvolvimento desse estudo, foi feita a separação do conjunto em duas categorias, normal e pneumonia. O conjunto de amostras de 5216 imagens foi separado para treinamento do modelo, com 74,3% classificadas com pneumonia e 25,7% normal. O conjunto de teste contém 624 imagens com 62,5% e 37,5%, respectivamente, pneumonia e tórax normal. O conjunto de validação contém com 16 imagens, com 8 de cada categoria. Na Figura 5 temos um exemplo de imagens do conjunto de dados.



(a) Pulmão normal (b) Pulmão com pneumonia

Figura 5. Imagem 5(a) corresponde a uma radiografia de tórax normal e mostra pulmões claros, a imagem 5(b) exibe um pulmão com pneumonia.

V. Resultados e Discussão

A padronização nos dados foi aplicada devido a diferença entre os intervalos das variáveis preditoras, conforme a equação 13. O método consiste em escalar as variáveis de modo que a média seja zero e desvio padrão igual a 1.

$$Z = \frac{x - \mu}{\sigma} \tag{13}$$

O PCA permitiu que o estudo fosse realizado com um número de dimensão menor, com 600 componentes foi possível explicar 91,5% da variância total explicada pelas 2048 características extraídas da ResNet-50. Isso permitiu um tempo de processamento menor para o treinamento do modelo, sem perda de informações relevantes.

Na otimização de parâmetros do RF com AG foi considerado: uma população de tamanho 8; 5 gerações; probabilidade

de 0,8 de crossover; probabilidade de 0,1 de mutação; função fitness definida pela ROC-AUC. O método de validação cruzada k-fold consiste em amostras aleatórias estratificadas, ou seja, preservando a porcentagem de amostras para cada classe, com $k = 5$.

O intervalo de busca dos parâmetros do RF foram: número de árvores na floresta ficou entre $[20, 80]$, a profundidade máxima da árvore em $[4, 8]$, para medir a qualidade de uma divisão os critérios gini e entropy foram considerados, as funções 'sqrt' e 'log2' foram utilizadas para determinar a quantidade de variáveis preditoras na amostra, os intervalos para o número mínimo de amostras necessárias para divisão de um nó interno e para estar em um nó folha foram, respectivamente, $[80, 150]$ e $[200, 350]$.

A Figura 6 representa a quantidade de modelos ajustados e para cada interação o valor da média da métrica ROC-AUC nos conjuntos de treinamentos e validações no processo de validação cruzada. A margem sobre as linhas do gráfico corresponde ao desvio padrão de cada modelo em função dos k-fold's. A Figura 7, corresponde ao ROC-AUC em função do número de gerações do processo de otimização de parâmetros do AG, é possível notar que a partir da terceira geração o desempenho do modelo não tem diferença significativa, com ROC-AUC de 97,8% na base de treinamento. Os parâmetros do melhor modelo no processo de validação cruzada foram: a entropia para o critério de divisão; profundidade da árvore igual a 6; número mínimo de amostras de 88 para nó folha; 218 para divisão de um nó interno; e 75 para número de árvores.

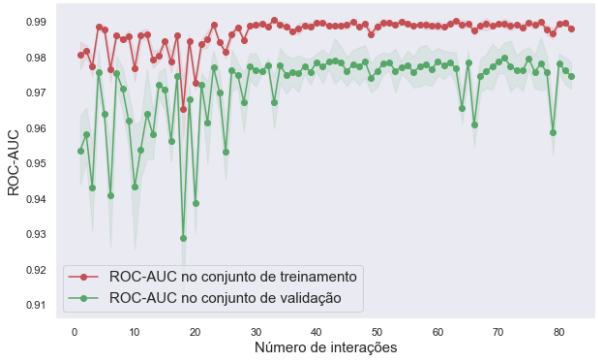


Figura 6. ROC-AUC em função do número de interações do processo de validação cruzada.

Para avaliar a capacidade de generalização do modelo a métrica de desempenho foi analisada em função do número de amostras para treinamento, para isso foi selecionado o melhor modelo do resultado de otimização com AG e foi avaliado em outro processo de validação cruzada. Na Figura 8 representa as curvas de aprendizado do RF e funciona como um mecanismo para diagnosticar o problema de viés e variância do modelo. Nota-se que conforme o número de amostras aumenta os valores de ROC-AUC do conjunto de validação cruzada convergem para os valores de inferência no conjunto de treinamento. Podemos observar que não há

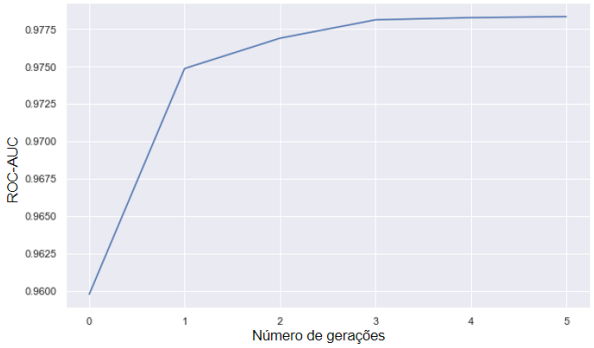


Figura 7. ROC-AUC em função do número de gerações.

muita diferença entre as curvas, o que corrobora na tese de que o modelo consegue generalizar sua inferência para outros conjuntos de dados.

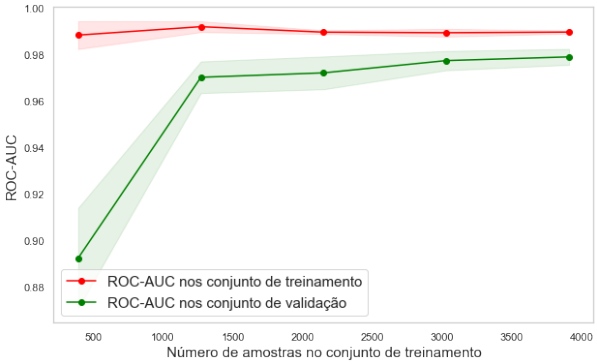


Figura 8. ROC-AUC do conjunto de treino e validação em função do número de amostras.

Para inferência nos conjuntos de dados de teste e validação foi optado por otimizar o ponto de corte do modelo em função da distribuição de probabilidade. Conforme a Figura 9 pode-se observar que a distribuição de probabilidade é assimétrica, a classe de radiografia de tórax com pneumonia é concentrada próximo ao intervalo de [0,8; 1,0], a classe de imagens normais está distribuída no intervalo [0,5;0,6]. A fim de obter melhores métricas, escolhemos o ponto de corte de 0,7 para probabilidade que distingue ambas as classes conforme a frequência observada no conjunto de treinamento.

As matrizes de confusão estão representadas na Figura 10, as métricas obtidas podem ser verificadas na Tabela I. O classificador RF obteve o desempenho de ROC-AUC de 93%, 79% e 88% nos conjuntos de treino, teste e validação, respectivamente. O tempo de processamento para o treinamento do modelo foi de 12,3 minutos. O modelo obteve os menores desempenhos da métrica *specificity* nos conjuntos de teste e validação, esse resultado indica que a proporção de falsos positivos aumentou em relação a proporção observada no conjunto de treinamento, o que não é um problema pois é preferível que o modelo tenha alto *recall* de modo a encontrar a maior parte dos pacientes doentes, mesmo que classifique

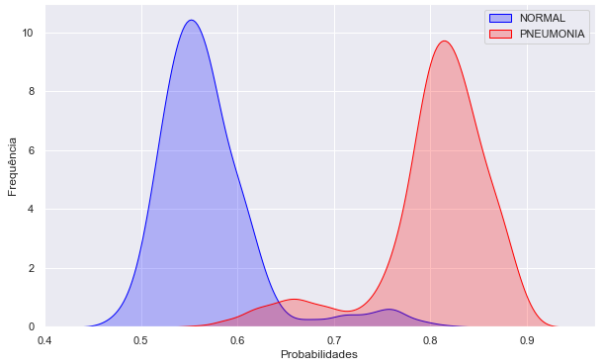
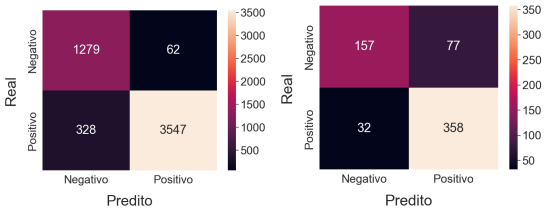


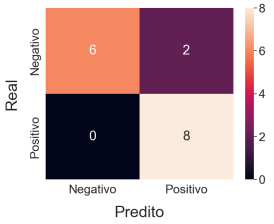
Figura 9. Distribuição da probabilidade do modelo Random Forest.

alguns pacientes saudáveis como doentes.



(a) Conjunto de treino

(b) Conjunto de teste



(c) Conjunto de validação

Figura 10. Matriz de confusão para cada conjunto de dados

Tabela I
MÉTRICAS DE DESEMPENHO

Métricas de avaliação	**Conjunto de dados**		
	Treino	***Teste***	***Validação***
Accuracy	93%	83%	88%
Precision	98%	82%	80%
Recall	92%	92%	100%
Specificity	95%	67%	75%
F_1-*score*	95%	87%	89%
ROC-AUC	93%	79%	88%

Comparando os resultados obtidos por outros trabalhos que abordam a classificação de imagens de radiografia de tórax, nota-se que o desempenho alcançado pelo modelo proposto é comparável com resultados vistos na literatura. Em [10] os autores utilizaram o mesmo conjunto de dados, e alcançaram um valor de *accuracy* de 93%, com *specificity* de 90% e *recall* de 93% no conjunto de teste. Os autores [15] relatam que o tempo de treinamento das redes pré-treinadas como ResNet-50, Inception-V3 e ResNet-101 são em média da ordem de 4 horas. É evidente que a transferência de aprendizado dessas redes pré-treinadas contribuem significativamente no desempenho da classificação das imagens. Os resultados vistos em [15], [16], [17], [18] são da ordem de 99% de *accuracy*. Contudo, o tempo de execução para treinamento é relativamente maior comparado a abordagem adotada nesse trabalho. A estratégia de usar PCA para reduzir a dimensão das variáveis preditoras se demonstra factível para inferência do modelo. O resultado obtido nesse trabalho demonstra que a metodologia é robusta e comparável aos resultados da literatura, embora os valores de algumas métricas de desempenho sejam inferiores comparado a alguns trabalhos elaborados para classificação de imagens de radiografia de tórax, o modelo consegue distinguir com alto desempenho imagens de indivíduos saudáveis e com pneumonia. A proposta tem um tempo de treinamento relativamente baixo, o que pode ser interessante para testar novas arquiteturas de redes neurais e modelos de aprendizado de máquina que confronte os resultados apresentados pela literatura.

VI. CONCLUSÃO

Os modelos de aprendizado de máquinas são técnicas amplamente utilizadas para classificação e detecção de padrões não lineares. O presente trabalho apresentou algoritmos de aprendizado de máquina utilizado para classificação de imagens de radiografia de tórax de indivíduos saudáveis e com pneumonia. A rede neural pré-treinada ResNest-50 foi utilizada para extrair características das imagens. Esse processo resultou em 2048 variáveis preditoras. Com auxílio da PCA foi possível reduzir a dimensão para 600 componentes principais, com 91% da variância total. Essas informações foram utilizadas para classificação das imagens usando o modelo RF em conjunto com AG no processo de validação cruzada para otimização dos parâmetros. Foi realizado uma análise de otimização do melhor limiar de probabilidade para classificação das imagens.

Os resultados desse trabalho contribuem para desenvolvimento de sistemas de IA com alto desempenho. O processamento dessas informações pode contribuir para agilizar o diagnóstico de doenças tratáveis como a pneumonia, facilitando o tratamento precoce. O modelo proposto obteve um desempenho de F_1-*score* de 95%, 87% e 89% nos conjuntos de treino, teste e validação, respectivamente, com tempo de execução baixo. O treinamento do modelo foi realizado com uma máquina comum, o que torna o experimento alcançado seja facilmente reproduzível. Essa foi a razão para não usar uma CNN para classificação, pois o treinamento da rede neural requer recursos computacionais para processamento de alto desempenho, conforme os resultados vistos na literatura as tecnologias como *Graphics Processing Unit* (GPU) são fundamentais para reduzir o tempo de treinamento das CNNs [15]. Para pesquisas futuras se pretende avaliar outros algoritmos de aprendizado de máquina, em conjunto com outras arquitetura de redes neurais pré-treinadas, como VGG-19, DesNet-101, CoroNet, ResNest-101, etc [21] [20]. A avaliação

do desempenho do modelo entre os grupos com pneumonia viral e bacteriana pode contribuir para demonstrar a robustez dos métodos abordados, embasando o desenvolvimento de modelos para classificar o tipo de pneumonia em indivíduos doentes.

REFERÊNCIAS

[1] R. k. Seramo, S. M. Awol, Y. A. Wabe, and M. M. Ali, "Determinants of pneumonia among children attending public health facilities in Worabe town," *Scientific Reports 2022 12:1*, vol. 12, pp. 1–9, 4 2022.

[2] J. C. Rodrigues, L. V. F. Da Silva Filho, and A. Bush, "Diagnóstico etiológico das pneumonias: uma visão crítica," *Jornal de Pediatria*, vol. 78, pp. 129–140, 11 2002.

[3] R. A. M. Menezes, D. R. Pavanitto, and L. F. C. Nascimento, "Distribuição espacial das taxas de internação de crianças por pneumonia no Sistema Único de Saúde, nos municípios do estado de São Paulo," *Revista Brasileira de Epidemiologia*, vol. 22, p. e190053, 12 2019.

[4] R. d. O. Ferraz, J. K. Oliveira-Friestino, and P. M. S. B. Francisco, "Pneumonia mortality trends in all Brazilian geographical regions between 1996 and 2012," *Jornal Brasileiro de Pneumologia*, vol. 43, pp. 274–279, 7 2017.

[5] S. Raoof, D. Feigin, A. Sung, S. Raoof, L. Irugulpati, and E. C. Rosenow, "Interpretation of plain chest roentgenogram," *Chest*, vol. 141, pp. 545–558, 2 2012.

[6] S. M. Acosta, R. M. A. Oliveira, and M. O. Sant'Anna, "Machine learning algorithms applied to intelligent tyre manufacturing," *International Journal of Computer Integrated Manufacturing*, pp. 1–11, 2 2023.

[7] A. Abdollahi, B. Pradhan, N. Shukla, S. Chakraborty, and A. Alamri, "Deep learning approaches applied to remote sensing datasets for road extraction: A state-of-the-art review," *Remote Sensing*, vol. 12, 5 2020.

[8] G. Silva, L. Duarte, M. Shirassu, S. Peres, M. de Moraes, and A. Chiavegatto Filho, "Machine learning for longitudinal mortality risk prediction in patients with malignant neoplasm in São Paulo, Brazil," *Artificial Intelligence in the Life Sciences*, vol. 3, p. 100061, 12 2023.

[9] F. Ferreira, P. Gaspar, L. M. Oliveira, R. Torres, M. V. Araújo, C. E. Covas, M. Bastos, A. Trajamn, and J. M. Seixas, "Machine Learning based sampling of X-Ray images for a computer-aided detection of Tuberculosis," *Associacao Brasileira de Inteligencia Computacional*, pp. 1–7, 10 2021.

[10] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, J. Dong, M. K. Prasadha, J. Pei, M. Ting, J. Zhu, C. Li, S. Hewett, J. Dong, I. Ziyar, A. Shi, R. Zhang, L. Zheng, R. Hou, W. Shi, X. Fu, Y. Duan, V. A. Huu, C. Wen, E. D. Zhang, C. L. Zhang, O. Li, X. Wang, M. A. Singer, X. Sun, J. Xu, A. Tafreshi, M. A. Lewis, H. Xia, and K. Zhang, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, pp. 1122–1131, 2 2018.

[11] C. S. Lee, D. M. Baughman, and A. Y. Lee, "Deep Learning Is Effective for Classifying Normal versus Age-Related Macular Degeneration OCT Images," *Ophthalmology Retina*, vol. 1, pp. 322–327, 2017.

[12] M. K. Yaqoob, S. F. Ali, M. Bilal, M. S. Hanif, and U. M. Al-Saggaf, "ResNet Based Deep Features and Random Forest Classifier for Diabetic Retinopathy Detection," *Sensors (Basel, Switzerland)*, vol. 21, pp. 1–6, 6 2021.

[13] C. A. S. Silva, L. A. A. Cardoso, D. D. Ferreira, J. M. Seixas, M. L. S. Bastos, and A. Trajman, "Ensemble de especialistas para avaliação de adesão ao procedimento operacional padrão de fotografias de radiografias de tórax," *Associacao Brasileira de Inteligencia Computacional*, pp. 1–7, 10 2021.

[14] O. T. Nascimento, J. M. Seixas, and A. Trajman, "Data-augmentation de dados de radiografia de torax no contexto de aprendizagem profunda," *Associacao Brasileira de Inteligencia Computacional*, pp. 1–7, 10 2021.

[15] A. Narin, C. Kaya, and Z. Pamuk, "Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks," *Pattern Analysis and Applications*, vol. 24, pp. 1207–1220, 3 2020.

[16] A. I. Khan, J. L. Shah, and M. M. Bhat, "CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images," *Computer methods and programs in biomedicine*, vol. 196, 11 2020.

[17] M. E. H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. Al-Emadi, M. B. I. Reaz, and T. I. Islam, "Can AI help in screening Viral and COVID-19 pneumonia?," *IEEE Access*, vol. 8, pp. 132665–132676, 3 2020.

[18] R. d. S. Farias, K. R. d. S. Oliveira, C. D. M. Regis, C. d. M. Costa, and J. M. Seixas, "Redes Neurais Convolucionais para Classificação da COVID-19 em Imagens de Raio-x de Tórax," *Anais do 15. Congresso Brasileiro de Inteligência Computacional*, pp. 1–8, 1 2021.

[19] M. B. Hossain, S. M. S. Iqbal, M. M. Islam, M. N. Akhtar, and I. H. Sarker, "Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images," *Informatics in Medicine Unlocked*, vol. 30, p. 100916, 1 2022.

[20] K. Sahinbas and F. O. Catak, "Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images," *Data Science for COVID-19*, p. 451, 1 2021.

[21] A. H. Barshooi and A. Amirkhani, "A novel data augmentation based on Gabor filter and convolutional deep learning for improving the classification of COVID-19 chest X-Ray images," *Biomedical Signal Processing and Control*, vol. 72, p. 103326, 2 2022.

[22] S. Hassantabar, M. Ahmadi, and A. Sharifi, "Diagnosis and detection of infected tissue of COVID-19 patients based on lung x-ray image using convolutional neural network approaches," *Chaos, Solitons and Fractals*, vol. 140, 2020.

[23] S. Haykin, "Neural Networks and Learning Machines," *Pearson Prentice Hall New Jersey USA 936 pLinks*, vol. 3, p. 906, 2008.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 770–778, 12 2015.

[25] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer Series in Statistics, New York, NY: Springer New York, 2nd ed ed., 2009.

[26] M. M. Silva, E. F. Simas Filho, P. C. Farias, M. C. Albuquerque, I. C. Silva, and C. T. Farias, "Intelligent embedded system for decision support in pulsed eddy current corrosion detection using Extreme Learning Machine," *Measurement*, vol. 185, p. 110069, 11 2021.

[27] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning : with Applications in R.* New York, NY: Springer, 2nd ed ed., 2013.

[28] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 10 2001.

[29] J. L. Carvalho, P. C. Farias, and E. F. Simas Filho, "Global Localization of Unmanned Ground Vehicles Using Swarm Intelligence and Evolutionary Algorithms," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 107, pp. 1–21, 3 2023.

[30] A. L. Pereira, M. Kohler, and M. A. C. Pacheco, "Evolutionary Convolutional Neural Network: a case study," *Associacao Brasileira de Inteligencia Computacional*, pp. 1–6, 11 2021.

[31] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, pp. 8091–8126, 2 2021.

[32] G. Ruan, U. Schmidhalter, F. Yuan, D. Cammarano, X. Liu, Y. Tian, Y. Zhu, W. Cao, and Q. Cao, "Exploring the transferability of wheat nitrogen status estimation with multisource data and Evolutionary Algorithm-Deep Learning (EA-DL) framework," *European Journal of Agronomy*, vol. 143, 2 2023.

[33] A. K. Santra and C. J. Christy, "Genetic Algorithm and Confusion Matrix for Document Clustering," 2012.

[34] D. Kermany, K. Zhang, and M. Goldbaum, "Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification," vol. 2, 2018.

UFBA
UNIVERSIDADE FEDERAL DA BAHIA
ESCOLA POLITÉCNICA

PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA INDUSTRIAL - PEI

Rua Aristides Novis, 02, 6º andar, Federação, Salvador BA
CEP: 40.210-630
Telefone: (71) 3283-9800
E-mail: pei@ufba.br
Home page: http://www.pei.ufba.br