Facultad de
Ingeniería de Sistemas

LATIN-AMERICAN JOURNAL OF COMPUTING

Volume 11, ISSUE 2
July 2024
ISSN: 1390-9266
e-ISSN:1390-9134

https://www.epn.edu.ec Quito - Ecuador https://lajc.epn.edu.ec

Vol XI, Issue 2 July 2024

ESCUELA POLITÉCNICA NACIONAL

MISIÓN

La Escuela Politécnica Nacional es una Universidad pública, laica y democrática que garantiza la libertad de pensamiento de todos sus integrantes, quienes están comprometidos con aportar de manera significativa al progreso del Ecuador. Formamos investigadores y profesionales en ingeniería, ciencias, ciencias administrativas y tecnología, capaces de contribuir al bienestar de la sociedad a través de la difusión del conocimiento científico que generamos en nuestros programas de grado, posgrado y proyectos de investigación. Contamos con una planta docente calificada, estudiantes capaces y personal de apoyo necesario para responder a las demandas de la sociedad ecuatoriana.

VISIÓN

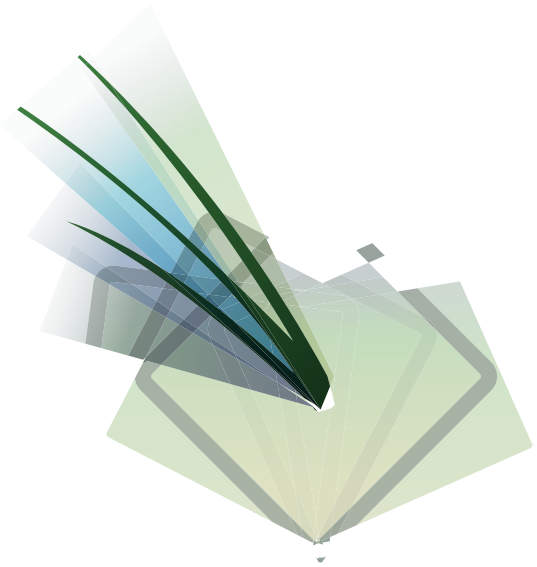
En el 2024, la Escuela Politécnica Nacional es una de las mejores universidades de Latinoamérica con proyección internacional, reconocida como un actor activo y estratégico en el progreso del Ecuador. Forma profesionales emprendedores en carreras y programas académicos de calidad, capaces de aportar al desarrollo del país, así como promover y adaptarse al cambio y al desarrollo tecnológico global. Posiciona en la comunidad científica internacional a sus grupos de investigación y provee soluciones tecnológicas oportunas e innovadoras a los problemas de la sociedad.

La comunidad politécnica se destaca por su cultura de excelencia y dinamismo al servicio del país dentro de un ambiente de trabajo seguro, creativo y productivo, con infraestructura de primer orden.

ACCIÓN AFIRMATIVA

La Escuela Politécnica Nacional es una institución laica y democrática, que garantizar la libertad de pensamiento, expresión y culto de todos sus integrantes, sin discriminación alguna. Garantiza y promueve el reconocimiento y respeto de la autonomía universitaria, a través de la vigencia efectiva de la libertad de cátedra y de investigación y del régimen de cogobierno.

https://www.epn.edu.ec



FACULTAD DE INGENIERÍA DE SISTEMAS

MISIÓN

La Facultad de Ingeniería de Sistemas es el referente de la Escuela Politécnica Nacional en el campo de conocimiento y aplicación de las Tecnologías de Información y Comunicaciones; actualiza en forma continua y pertinente la oferta académica en los niveles de pregrado y postgrado para lograr una formación de calidad, ética y solidaria; desarrolla proyectos de investigación, vinculación y proyección social en su área científica y tecnológica para solucionar problemas de transcendencia para la sociedad.

VISIÓN

La Facultad de Ingeniería de Sistemas está presente en posiciones relevantes de acreditación a nivel nacional e internacional y es referente de la Escuela Politécnica Nacional en el campo de las Tecnologías de la Información y Comunicaciones por su aporte de excelencia en las carreras de pregrado y postgrado que auspicia, la calidad y cantidad de proyectos de investigación, vinculación y proyección social que desarrolla y su aporte en la solución de problemas nacionales a través del uso intensivo y extensivo de la ciencia y la tecnología.

https://fis.epn.edu.ec

LATIN-AMERICAN JOURNAL OF COMPUTING

Vol XI, Issue 2, July 2024

ISSN: 1390-9266 e-ISSN: 1390-9134
DOI: https://doi.org/10.33333/lajc.vol11n2

Published by:
Escuela Politécnica Nacional
Facultad de Ingeniería de Sistemas

Quito – Ecuador

Indexed in

Associated institutions

Mailing Address
Escuela Politécnica Nacional,
Facultad de Ingeniería de Sistemas
Ladrón de Guevara E11-253, La Floresta
Quito-Ecuador, Apartado Postal: 17-01-2759

Web Address
https://lajc.epn.edu.ec/

E-mail
lajc@epn.edu.ec

Frequency
2 issues per year

Published by

Escuela Politécnica Nacional
Facultad de Ingeniería de Sistemas
Ecuador

EDITORIAL

Gabriela Suntaxi
PhD.

Editor LAJC
Escuela Politécnica Nacional,
Ecuador

Nos complace compartir con ustedes el Volumen 11, Número 2 de la Revista Latinoamericana de Computación (LAJC). Esta edición incluye una selección de artículos de investigación pioneros que demuestran los últimos avances en el área de las Ciencias de la Computación. Cada artículo incluido en este volumen representa una investigación académica rigurosa y métodos innovadores de resolución de problemas. Creemos que las ideas e investigaciones presentadas aquí contribuirán significativamente al área, estimularán discusiones e inspirarán futuras innovaciones.

Este número comienza con tres artículos que exploran metodologías avanzadas en la monitorización de procesos, transferencia de calor y robótica. El primer artículo investiga el uso de Redes de Estado Eco (ESNs) para crear gemelos digitales de procesos químicos dinámicos no lineales, demostrando el potencial de las ESNs en la generación de modelos sustitutos eficientes para la monitorización y control de procesos en tiempo real. El segundo artículo aborda el problema inverso en la modelación de transferencia de calor utilizando el método de la Cadena de Markov de Monte Carlo de Transición, mostrando su efectividad en la estimación de propiedades termofísicas variables en el espacio. A continuación, Janarthanan et al. exploran el potencial de los datos generados por robots, enfocándose específicamente en los archivos ROS Bag utilizados en el Sistema Operativo de Robots (ROS). El estudio destaca problemas de seguridad, como el acceso no autorizado y el robo de datos, debido a la comunicación en texto plano en los sistemas ROS heredados.

Este número también profundiza en las aplicaciones críticas de la inteligencia artificial y el aprendizaje automático en varios dominios científicos e industriales. El cuarto artículo presenta el enfoque ANN-MoC para resolver problemas inversos de transporte transitorio, mostrando su potencial en los campos de la ingeniería y la medicina mediante la estimación precisa de coeficientes de absorción a partir de mediciones de flujo escalar. A continuación, otro estudio explora el impacto del equilibrio de datos en las previsiones a corto plazo de precipitaciones utilizando Redes Neuronales Artificiales (ANNs) con datos del Observatorio de la Torre Alta del Amazonas (ATTO). Esta investigación enfatiza la necesidad de datos equilibrados para mejorar la precisión y confiabilidad de los modelos meteorológicos, destacando las implicaciones más amplias para la monitorización y predicción ambiental. Además, el volumen incluye un modelo innovador de clasificación de fallos para procesos industriales, que combina Árboles de Decisión con Programación Genética para mejorar las medidas preventivas y correctivas.

Finalmente, exploramos los mercados financieros y los avances tecnológicos. Un artículo compara el mercado de valores brasileño con criptomonedas como Bitcoin, Ethereum y Solana, utilizando la prueba no paramétrica de Kolmogorov-Smirnov para examinar sus relaciones y oportunidades de inversión potenciales. El último estudio utiliza el aprendizaje automático y la metaheurística de Optimización del Lobo Gris para predecir la demanda de electricidad en Brasil, mostrando modelos de regresión avanzados para pronosticar con precisión el consumo de energía.

Esperamos que la diversa gama de temas y enfoques innovadores presentados en este volumen inspire sus propias investigaciones. Los avances en inteligencia computacional, aprendizaje automático y análisis de datos aquí expuestos subrayan el potencial transformador de estas tecnologías para abordar desafíos del mundo real. Mientras continuamos explorando las fronteras de las ciencias de la computación, los invitamos a empujar juntos los límites del conocimiento dentro de nuestra comunidad científica. Juntos, podemos impulsar el progreso y hacer contribuciones significativas al campo.

Gabriela Suntaxi

Editora en Jefe

We are pleased to share Volume 11, Issue 2 of the Latin American Journal of Computing (LAJC) with you. This edition includes a selection of pioneering research articles that demonstrate the latest advancements in the computer science field. Each paper included in this volume represents rigorous academic research and innovative problem-solving methods. We believe that the insights and discoveries presented here will significantly contribute to the field, stimulate insightful discussions, and inspire future innovations.

This issue begins with three articles that explore advanced methodologies in process monitoring, heat transfer, and robotics. The first article investigates the use of Echo State Networks (ESNs) to create digital twins for nonlinear dynamic chemical processes, demonstrating the potential of ESNs in generating efficient surrogate models for realtime process monitoring and control. The second article addresses the inverse problem in heat transfer modeling using the Transition Markov Chain Monte Carlo method, showcasing its effectiveness in estimating spatially variable thermophysical properties. Next, Janarthanan et al. explore the potential of data generated by robots, specifically focusing on ROS Bag files used in the Robot Operating System (ROS). The study highlights security concerns, such as unauthorized access and data theft, due to plain text communication in legacy ROS systems.

This issue also delves into the critical applications of artificial intelligence and machine learning in various scientific and industrial domains. The fourth article presents the ANNMoC approach for solving inverse transient transport problems, showcasing its potential in engineering and medical fields by accurately estimating absorption coefficients from scalar flux measurements. Next, another study explores the impact of data balance on short-term rainfall forecasts using Artificial Neural Networks (ANNs) with data from the Amazon Tall Tower Observatory (ATTO). This research emphasizes the necessity of balanced data to improve the accuracy and reliability of meteorological models, highlighting the broader implications for environmental monitoring and prediction. Additionally, the volume includes an innovative fault classification model for industrial processes, merging Decision Trees with Genetic Programming to enhance preventive and corrective measures.

Finally, we explore financial markets and technological advancements. One article compares the Brazilian stock market with cryptocurrencies like Bitcoin, Ethereum, and Solana, using the Kolmogorov-Smirnov test to examine their relationships and potential investment opportunities. The last study uses machine learning and the Grey Wolf Optimization meta-heuristic to predict Brazil's electricity demand, showcasing advanced regression models for accurate energy consumption forecasting.

We hope that the diverse range of topics and innovative approaches presented in this volume will inspire your own research endeavors. The advancements in computational intelligence, machine learning, and data analysis showcased here underscore the transformative potential of these technologies in addressing real-world challenges. As we continue to explore the frontiers of computer science, we invite you to join us in pushing the boundaries of knowledge within our scientific community. Together, we can drive progress and make meaningful contributions to the field.

Gabriela Suntaxi

Editor-in-Chief

Reviewers

We are most grateful to the following individuals for their time and commitment to review manuscripts for the Latin American Journal of Computing – LAJC

Axel Rodriguez, MSc.
Universidad Tecnológica de Panamá

Carlos Montenegro, MSc.
Escuela Politécnica Nacional

Cesar Salinas, PhD.
Universidad de las Américas (UDLA)

David Nuñez, MSc.
CNT EP

Diana Martinez, PhD.
Escuela Politécnica Nacional

Edison Loza, PhD.
Escuela Politécnica Nacional

Freddy Tapia, PhD.
Universidad de las Fuerzas Armadas ESPE

Henry Paz, MSc.
Escuela Politécnica Nacional

Irene Cedillo, PhD.
Universidad de Cuenca

Jhonattan Barriga, PhD.
Escuela Politécnica Nacional

Jorge Miño, MSc.
Escuela Politécnica Nacional

José Lucio, PhD.
Escuela Politécnica Nacional

Julio Ibarra, PhD.
Universidad San Francisco de Quito

Leonardo Valdivieso, PhD.
Escuela Politécnica Nacional

Marcelo Palma, MSc.
IEEE

Marco Molina, PhD.
Escuela Politécnica Nacional

Patricio Zambrano, PhD.
Escuela Politécnica Nacional

Raphael de Oliveira Garcia, PhD.
Federal University of São Paulo

Romel Tintin, PhD.
Instituto de Altos Estudios Nacionales

Sang Yoo, PhD.
Escuela Politécnica Nacional

Veronica Segarra, PhD.
Universidad Técnica Particular de Loja

TABLE OF CONTENTS

Exploring Digital Twins of Nonlinear Systems through Meta-Modeling with Echo State Networks

ARTICLE HISTORY
Received 08 March 2024
Accepted 08 May 2024
Published 8 July 2024

Laisa Cristina Juffo Campos
Universidade Federal do Espírito Santo
Alegre, Brasil
laisacampos01@gmail.com
ORCID: 0009-0003-4427-0395

Wellington Betencurte da Silva
Universidade Federal do Espírito Santo
Alegre, Brasil
wellinton.betencurte@ufes.br
ORCID: 0000-0003-2242-7825

Ana Carolina Spindola Rangel Dias
Serviço Nacional de Aprendizagem Industrial (SENAI)
Rio de Janeiro, Brasil
acspdias@gmail.com
ORCID: 0000-0001-7376-0703

Julio Cesar Sampaio Dutra
Universidade Federal do Espírito Santo
Alegre, Brasil
julio.dutra@ufes.br
ORCID: 0000-0001-6784-4150

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

L. Campos, A.Dias, J. Dutra and W. da Silva, "Exploring Digital Twins of Nonlinear Systems through Meta-Modeling with Echo State Networks", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024.

Exploring Digital Twins of Nonlinear Systems through Meta-Modeling with Echo State Networks

Laisa Cristina Juffo Campos
Universidade Federal do Espírito Santo
Departamento de Engenharia Rural
Alegre, Brasil
laisacampos01@gmail.com

Wellington Betencurte da Silva
Universidade Federal do Espírito Santo
Departamento de Engenharia Rural
Alegre, Brasil
wellinton.betencurte@ufes.br

Ana Carolina Spindola Rangel Dias
Serviço Nacional de Aprendizagem Industrial (SENAI)
Rio de Janeiro, Brasil
acspdias@gmail.com

Julio Cesar Sampaio Dutra
Universidade Federal do Espírito Santo
Departamento de Engenharia Rural
Alegre, Brasil
julio.dutra@ufes.br

***Abstract*—** Effective process monitoring, and control rely on precise dynamic models that can capture the inherent nonlinearities of chemical systems. However, rigorous modeling of complex industrial processes can be computationally demanding. Meta modeling using machine learning methodologies offers a viable approach to generate computationally efficient surrogate representations. Specifically, Echo State Networks (ESNs) are a promising neural network approach for meta-modeling nonlinear dynamical systems. ESNs simplify training through fixed input weights while they focus learning on output weights. This study explores the development of ESN-based digital twins for a nonlinear dynamic process. An ESN is employed to construct a meta-model of a simulated continuously stirred tank reactor with biochemical kinetic. The network was trained on input-output data obtained from the simulation of an ordinary differential equation system, and the performance was evaluated both in-sample and out-of-sample. The results indicate that the ESN meta-model can successfully approximate the underlying dynamics, accurately capturing temporal evolution. A closed-loop digital twin deployment using the ESN surrogate also showed reliable behavior. This work presents initial steps toward developing digital twins of chemical processes using ESN-driven meta-modeling. The findings suggest ESNs can effectively generate computationally efficient surrogate representations of nonlinear dynamical systems. Such digital twins hold promise for online process monitoring and optimized control of industrial plants.

***Keywords*—** *Echo State Networks, Dynamic systems, Digital twins*

I. Introduction

In recent years, rapid technological progress has resulted in substantial enhancements across diverse sectors, notably in enhancing quality and safety within chemical processes. The ubiquitous incorporation of computers into process management has empowered control over various variables, that include temperature, pressure, and chemical composition, thereby generating extensive and diverse data archives [1]. Design challenges necessitating intensive computational resources are increasingly prevalent in manufacturing industries [2]. Moreover, creating tools capable of analyzing data and constructing predictive mathematical models has become imperative for real-time process monitoring and control.

Creating rigorous models that accurately capture the dynamics and nonlinearity of real systems may be impractical at plant sites, where rapid responses are crucial. One practical approach is to utilize metamodeling strategies [2][3] to tackle the challenges inherent in process systems. Widely utilized across engineering, computer science, and optimization, these strategies involve developing simplified models that approximate the behavior of complex systems or processes [4]. These simplified representations, named meta-models or surrogate models, aim to balance accuracy and computational efficiency.

In this context, digital twins emerge as virtual representations capable of reflecting the behavior of physical systems in real-time, this shows potential for online monitoring and process optimization [5]. By generating simplified yet computationally efficient models, digital twins enable dynamic data analytics and rapid decision-making to optimize industrial plant control and performance.

Expanding on recent data science research, metamodeling can draw upon various machine learning techniques [2][6]. Artificial Neural Networks (ANNs) are widely recognized for their ability to approximate complex functions [7]. Modeled after the functioning mechanism of biological neurons, ANNs comprise an input layer, a hidden layer housing artificial neurons in quantities necessary to represent the data, and an output layer. Additionally, ANNs possess memory storage and learning capabilities, making them particularly suitable for dynamic and nonlinear systems. This work precisely investigates this characteristic regarding applying neural meta-models for generating digital twins of complex chemical processes [8][9]. The aim is to develop computationally efficient representations that approximately capture the underlying dynamics of these systems.

Depending on the network architecture, various types of neural networks exist, including Feedforward Neural

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

Networks (FNNs) and Recurrent Neural Networks (RNNs). RNNs offer computational advantages for dynamic process systems owing to their inherent feedback loops. However, training traditional RNNs can be complicated due to issues like the "vanishing gradient" problem [10]. To address this, [11] introduced the Echo State Network (ESN). Unlike traditional RNNs that adjust all synaptic weights, ESNs maintain fixed input and recurrent connections, focusing solely on training output connections through a relatively simple linear regression process. This approach circumvents the complexities of training recurrent connections and mitigates gradient-related challenges. Consequently, ESNs present an effective solution for harnessing the power of RNNs while mitigating training complexities, particularly in scenarios where efficient learning is essential.

This article proposes using an Echo State Network as a meta-model to approximate dynamic nonlinear models and evaluate the performance in a closed-loop application. This work assesses the potential of this approach for this purpose, analyzing the performance of different methodologies in modeling a CSTR reactor through the construction of a digital twin. Section 2 presents a brief background on the metamodeling problem. Section 3 elaborates a case study based on a simulated bioreactor and details the data acquisition procedure. The theory, rationale, and construction of the Echo State Network are described in Section 4, followed by the discussion of simulation results.

The contribution of this article lies in presenting initial steps towards developing digital twins of chemical processes using ESN-driven meta-modeling. By demonstrating the efficacy of ESNs in generating computationally efficient surrogate representations of a classical nonlinear dynamical system, this work opens space for online process monitoring and optimized control of industrial plants.

II. The Metamodeling Problem

A meta-model (or surrogate model) can be conceived as a "model of a model" [6], functioning as a simplified representation of a high-fidelity simulation model [12]. It emulates the response by delineating the relationship between inputs (U) and outputs (Y) based on data acquired with known precision or uncertainty [13]. The importance of metamodeling lies in its ability to balance accuracy and computational efficiency. Hence, metamodeling emerges as an essential approach to navigating real-world system intricacies, especially those characterized by nonlinear relationships, numerous variables, and complex behaviors.

In industrial settings, meta-models are employed for tasks which necessitate the establishment of a (complex) relationship between the inputs and outputs of a process system. This relationship can be encapsulated by an extended meta-model equation that incorporates the feedback signal (1):

$$Y_k = f(U_k, Y_{k-1}) + \epsilon \tag{1}$$

Where Y_k represents the current output, U_k denotes the current inputs, Y_{k-1} is the previous output (feedback signal), $f(\cdot)$ is the relationship that incorporates inputs and feedback, and ϵ represents error or uncertainty in the meta-model prediction.

By offering a simplified representation of burdensome simulations, meta-models facilitate quicker evaluations and decision-making - crucial aspects in industries that demand real-time solutions. This approach enables approaching complex systems without the need of resource-intensive full-scale simulations, which can be computationally demanding and time-consuming. Some commonly used metamodeling techniques encompass polynomial surface response models, Kriging, Radial Basis Functions, Support Vector Regression, and Artificial Neural Networks [13][14]. These techniques generate approximated mappings from inputs to outputs. The choice depends on problem characteristics, available data, and required predictions.

Metamodeling using neural networks adopts a data-driven approach that harnesses the principles of ANNs to construct efficient approximations of complex systems. This methodology entails training the neural network on a dataset that reflects the system behavior under scrutiny. This dataset consists of input variables paired with corresponding output values, that facilitates the network identification of underlying patterns and correlations. Following training, the neural network can provide predictions for new input data, substantially which alleviates computational burdens compared to resource-intensive full-scale simulations.

The increased processing speed has dramatically expanded the applicability of neural network-based metamodeling. For example, [15] employed a neural network as a meta-model to approximate a copper porphyry mine comminution circuit, which leads to a significant acceleration of simulations compared to traditional phenomenological models. Additionally, [16] utilized neural networks in the metamodeling of reactive transport, and this reduces computational time for scenarios requiring multiple realizations. These studies highlight the versatility of neural network-based metamodeling in improving efficiency, accuracy, and computational performance across various domains.

Modeling and Data Generation

The mathematical model employed to generate the data was adapted from [17], outlining the dynamic behavior of a bioreactor. The equations that govern substrate balance, S, and cell balance, X, are expressed by (2) and (3), respectively, while the reaction rate, $\mu(S)$, is defined by (4), where D is defined as the dilution rate, that represents the ratio between the volumetric feed flow rate and the reactor volume, and S_f stands for the substrate feed concentration.

$$\frac{dS}{dt} = D(S_f - S) - \mu(s)\frac{X}{Y_{X/S}} \tag{2}$$

$$\frac{dX}{dt} = \mu(S)X - DX \tag{3}$$

$$\mu(S) = \frac{\mu_m S}{K_S + S} \tag{4}$$

All code implementations were developed in Python, with the free Spyder development environment (version

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

3.9.16). The code was compiled and executed on a computer system featuring 128 GB of DDR4 RAM, and an Intel® Core I7-12700k processor operating at 5.00 GHz.

This specific case study adopted a supervised training strategy to construct the neural model. This approach required the generation of input and output data. The input data was synthesized by a Random Gaussian Signal (RGS) algorithm [18]. The RGS technique is widely utilized for dynamic systems identification, which enables a thorough exploration of the input space. Consequently, it effectively stimulates the process response across diverse conditions.

The input variables were the dilution rate and substrate feed concentration, with mean values of 0.1 h^{-1} and 10.0 g L^{-1}, respectively. Each variable displayed variations of ± 0.1 h^{-1} and ± 2.5 g L^{-1}. A total of 2500 samples were generated and collected at intervals of 0.25 h. The sampling interval was modified to 8 h to generate the second dataset, while the other parameters were kept constant. As for the output data, represented by S and X, these were derived by solving the system of ordinary differential equations outlined in (2) and (3), using the solve_ivp function from the scipy.integrate library for this purpose. Gaussian random noise was added to the simulated result to make output data more complex and realistic, with a standard deviation of 5%. This makes the resulting data more complex while pushing the meta-model to discover the underlying patterns in a way that enhances its robustness against noise and variability when it transfers to actual operation. Subsequently, all datasets were organized and stored within a spreadsheet.

The generated data is showcased in Figs. 1-4 which illustrate the obtained data with higher (Figs. 1-2) and lower frequency (Figs. 3-4). The red data points indicate outputs with the addition of measurement noise, which was introduced to a better approximate reality and attenuate potential overfitting.

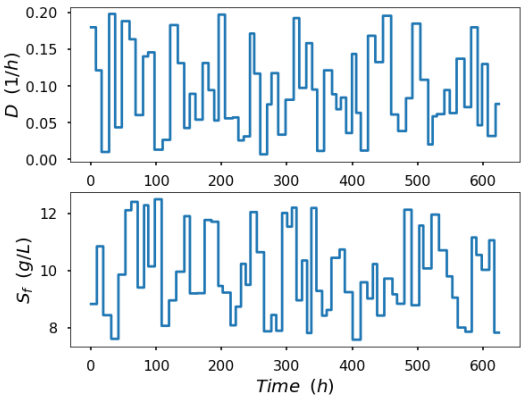


Fig. 1. Input data for the first dataset

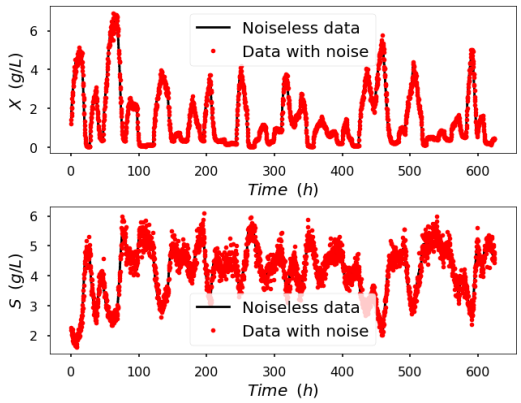


Fig. 2. Output data for the first dataset

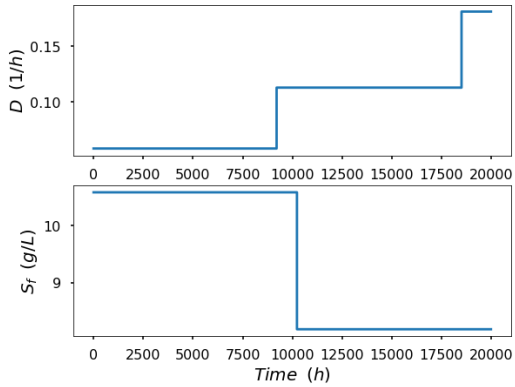


Fig. 3. Input data for the second dataset

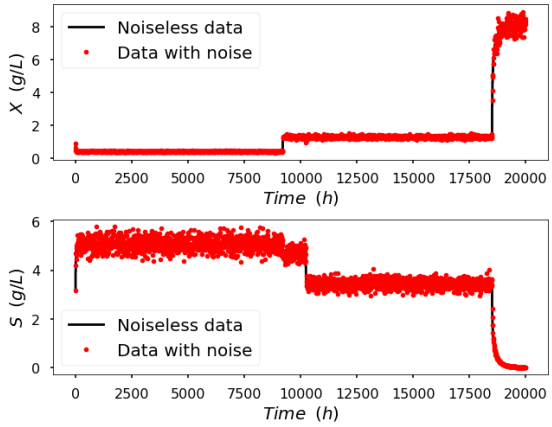


Fig. 4. Output data for the second dataset

III. Echo State Network

Acknowledging the potential of RNNs, [8] introduced a groundbreaking neural network architecture called the Echo State Network (ESN). The primary aim of this architecture is to harness the capabilities of effectively addressing complex problems while it simplifes the learning process. In the conventional training of ANNs, with the adjustment of synaptic weights across input, output, and feedback layers can impose substantial computational demands, often requiring significant computational resources. However, Jaeger's innovative network design focuses solely on training output weights, accomplished through a relatively straightforward linear regression process. This approach

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

offers significant advantages in terms of computational efficiency and streamlining the intricate task of fine-tuning complex feedback loops.

The ESN remarkably simplifies the training process by compartmentalizing the learning process into distinct stages - initially training output weights while keeping other weights fixed. This streamlined approach enhances computational efficiency and facilitates faster convergence during the training phase. Furthermore, the methodology unlocks potential applications in scenarios where efficient learning is paramount. The innovative design of the ESN offers a promising pathway to address challenges related to training complexity, which makes it well-suited for scenarios demanding both computational efficiency and enhanced learning performance.

In this implementation, the ESN network algorithm was coded following the equations outlined by [8], with specific hyperparameters maintained at fixed values (Table I). These predetermined values were determined empirically. An optimization method was utilized and implemented through Python programming to identify the optimal hyperparameters - neuron count, sparsity, and leaking rate. Following this, the resulting network was validated using the fine-tuned hyperparameters.

TABLE I. NETWORK HYPERPARAMETERS

Hyperparameter	Value
Reservoir size	1222
Leaking rate	0.6964
Sparsity	0.3536
Spectral radius	0.70
Train fraction	0.35
Ridge	4E-4
Noise level	1E-5
Random seed	13042023

IV. CONTROLLER TUNING AND CLOSED-LOOP

Another test was applied to evaluate the performance in a closed-loop simulation, allowing for the assessment of the feasibility of applying the trained network as a meta-model (that is, the digital twin). The control objective was to maintain cell concentration (X) around desired values, and it considers the substrate concentration in the feed (Sf) as the disturbance and the dilution rate (D) as the manipulated variable. For this purpose, we used a PI controller with the velocity algorithm.

A transfer function of the reactor dynamics was obtained to tune the controller, with a step test of -5% on D, performed on the differential model from its initial conditions. The steady-state response obtained was Xs = 4.5 g L^{-1} and Ss = 1.0 g L^{-1}. With the approach of [19], it was possible to approximate the process with a first-order plus dead time (FOPDT) system. Fig. 5 comparatively illustrates the original process (differential model), represented by red points, and the approximated process. The parameters obtained through such an approach are shown in Table II.

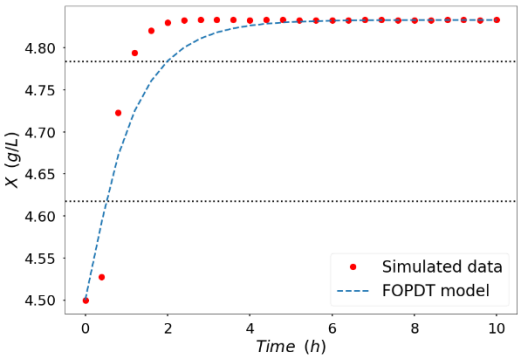


Fig. 5. Process simulation and obtained model

TABLE II. PROCESS PARAMETERS

Parameter	Value
K_P (L g^{-1} h^{-1})	-6.6642
θ (h)	0.0700
τ (h)	1.0050

After conducting tests on different controllers, three tuning techniques were applied: Internal Model Control (IMC), Integral of Time multiplied by Absolute Error for servo test (ITAE), and manual fine-tuning [17]. The parameters for each tuning technique are described in Table III. It was concluded that the manually tuned controller was the best choice for this study, even though it was a more conservative option. The manually tuned controller yielded a favorable result of less oscillation in the manipulated variable during closed-loop tests. Additionally, it demonstrated a slight difference in response time compared to the other controllers examined. The gain margin of the manually fine-tuned controller was 56.8437, which is significantly higher than the gain margins of the IMC (22.9541) and ITAE-servo test (3.0869) methods. This result suggests that the manually fine-tuned controller is more robust than the other methods. As a result, the manually fine-tuned controller was chosen due to its quick, highly stable, and oscillation-free response.

The results of the closed-loop simulation using the selected controller are presented in Figs. 6-7. Fig. 6 illustrates the behavior of the manipulated and disturbance variables, while Fig. 7 depicts the controlled variable with its setpoint, along with the other output.

TABLE III. TUNING METHODS AND CONTROLLER PARAMETERS

Parameter	Tuning method		
	IMC	*ITAE (servo test)*	*Manual*
K_C (L g^{-1} h^{-1})	-0.15164	-1.12761	-0.06123

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

τ_I (h)	1.00500	1.00752	6.70000

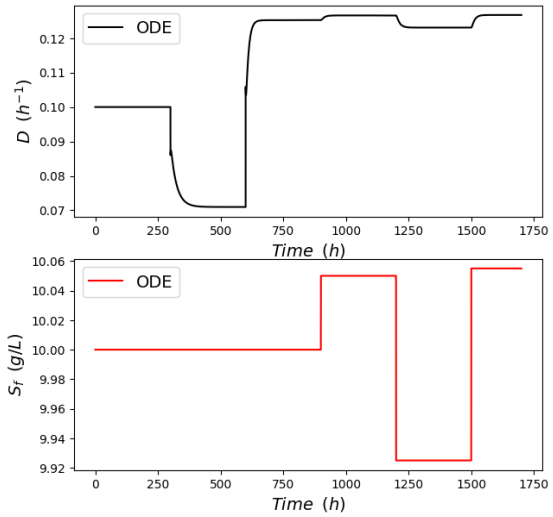


Fig. 6. Inputs of the closed-loop

To assess the neural network efficacy in accurately representing the behavior of the simulated system, as required for a digital twin, its response was evaluated within a closed-loop control framework. Within this framework, the control actions computed for the original process (based on the differential model) with the tuned proportional-integral (PI) controller were integrated as one of the network's inputs. Moreover, these inputs encompassed process disturbance information and a feedback signal generated by the network predictions rather than simulated measurements from the differential model simulation. Consequently, the neural network can autonomously adapt over time, dynamically responding to the evolving process inputs.

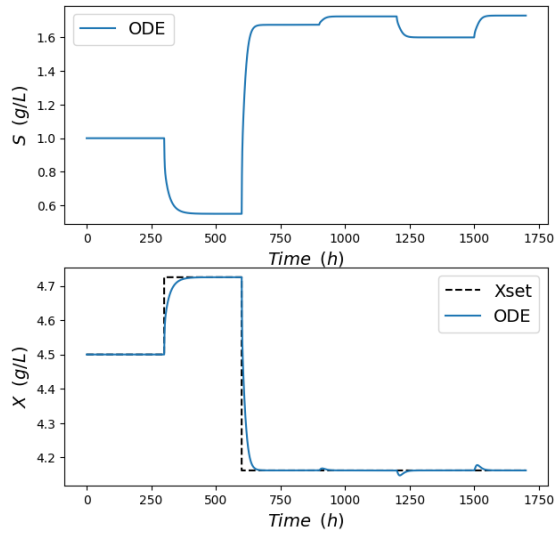


Fig. 7. Outputs of the closed-loop

V. Results

After fine-tuning the hyperparameters, the network performance was evaluated on both datasets. The higher-frequency dataset was used to assess the network predictive capacity. The neural network demonstrated exceptional training performance, accurately predicting the test data and effectively capturing the underlying dataset patterns and relationships (Fig. 8). This success highlights the robust ability of the model to generalize from complex training examples to unseen data, this showcases its deep understanding of system dynamics.

An autocorrelation analysis of the training modeling errors (residual) indicated significant autocorrelation only at lag = 0, resembling a Dirac delta function (Fig. 9), which confirms that the residual distribution follows a white noise correlogram pattern. We can see this result as an indication of the absence of systematic errors or patterns in the model predictions. Additionally, a white noise correlogram pattern suggests that the model has effectively captured all relevant information from the data, and the predictions are based on genuine signals rather than noise.

The following run evaluates the pre-trained network adaptability to a distinct scenario (second dataset), as illustrated in Fig. 10-11. As can be seen, the successful prediction of the second test dataset resulted in a residual distribution that also adheres to a white noise correlogram pattern. Remarkably, despite being trained with higher-frequency data, the model ability to accurately represent lower-frequency data underscores its robustness and versatility in capturing the system dynamics across different temporal scales.

In the closed-loop control scenario, the neural network functioned autonomously, providing its feedback signal based on the predicted outputs. However, Fig. 9 reveals a systematic deviation between the predicted and actual responses, likely stemming from the absence of feedback control dynamical effects in the training data. This discrepancy highlights the challenge of accurately capturing real-time system behavior under closed-loop control conditions.

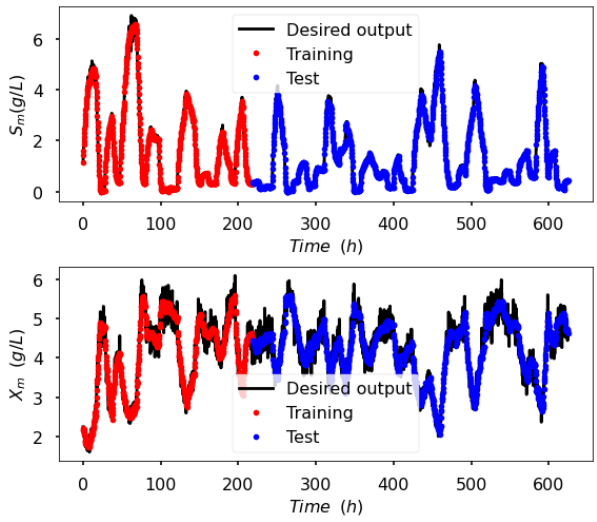


Fig. 8. Network performance for the first dataset

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

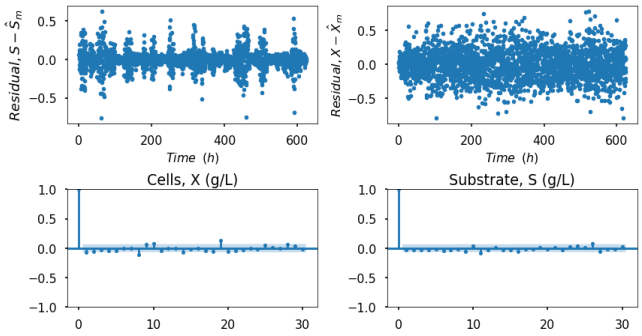


Fig. 9. Network residual analysis for the training of the first run

A bias b(k)was introduced to mitigate this issue, and this represents the disparity between the simulated process measurements, y_m(k), and the predicted outputs, \hat{y}(k). This adjustment on the predicted outputs, being \hat{y}\left(k\right)+b\left(k-1\right) with b(0)=0, yielded a maximum relative error of just 1.1%, compared to the 2.7% observed without bias. The graphical representations that depict the predictions in the absence and presence of bias correction are presented in Figs. 12 and 13, correspondingly.

Detailed performance metrics for the training, testing, and closed-loop application phases are provided in Table IV. The findings demonstrate the exceptional predictive capabilities of the network, which achieves outstanding performance in forecasting output data despite being trained on a comparatively small dataset — and contrasts with the higher training percentages commonly used in the literature. Notably, the network accurately captured the output dynamics in the first dataset with remarkable precision. Furthermore, the successful modeling of a scenario with lower variability in the second dataset suggests its versatility and robustness. Thus, inferring that the acquired meta-model fits both scenarios is reasonable. Moreover, the closed-loop results showcase the neural network potential as a virtual representation that reflects real-time process responses, thereby mimicking real-world scenarios with fidelity.

TABLE I. NETWORK PERFORMANCE METRICS

Metrics	Dataset 1		Dataset 2	Closed loop	
	Training	*Test*	*Test*	*Without bias*	*With bias*
R^2	0.9790	0.9490	0.9812	0.9930	0.9996
MSE	2.6676E-02	3.14347E-02	4.6535E-02	0.0007	0.0001
ExpVar	0.9790	0.9491	0.9812	0.9979	0.9998

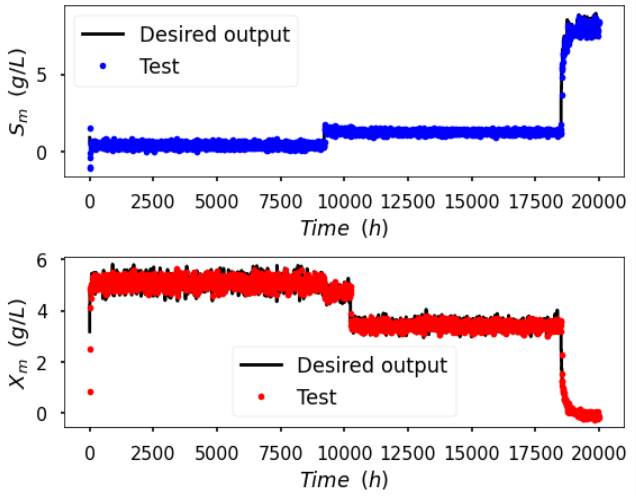


Fig. 10. Network performance for the second dataset

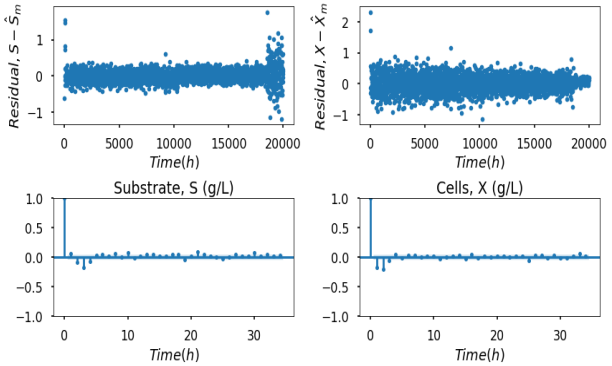


Fig. 11. Network residual analysis for the training of the second run

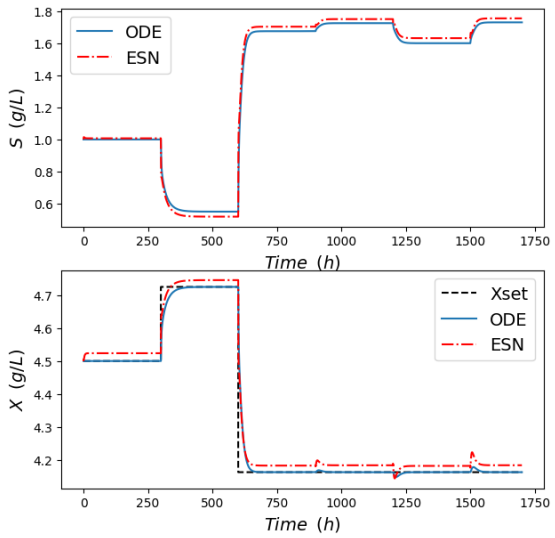


Fig. 12. Network performance for the closed-loop, without bias

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.01

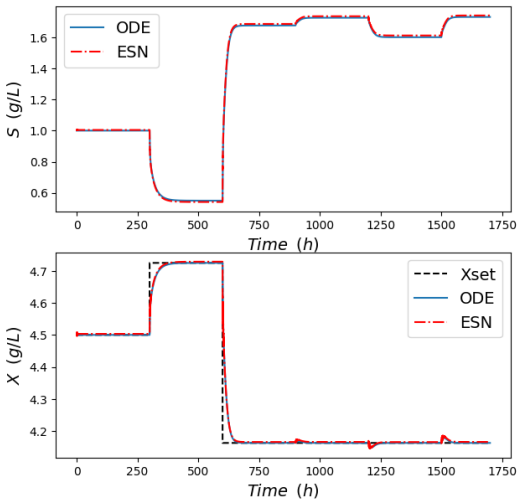


Fig. 13. Network performance for the closed-loop, with bias

VI. Conclusion

This study employed an Echo State Network (ESN) as a meta-model to tackle the complexities of a classical nonlinear bioreactor. Unlike traditional Recurrent Neural Networks, ESNs simplify learning by maintaining fixed input and recurrent connections, while training only output connections through linear regression. This approach mitigates the challenges associated with training recurrent connections.

The outcomes of our study showcase the robust predictive capabilities of the ESN, adeptly handling noisy data and limited samples across a broad spectrum of oscillations. These results underscore the ESN adaptability to the diverse scenarios commonly encountered in industrial contexts. The results of the closed-loop test validate the efficacy of ESNs, with maximum errors limited to just 3%. This underscores the potential for further exploration of ESN applications in constructing digital twins, which represents a paradigm shift from traditional models towards real-time control and monitoring contexts.

Moreover, the findings confirm the practical and effective utility of the ESN for metamodeling in industrial processes. The versatility and potential integration of ESNs into Process Control and Monitoring practices facilitate precise simulations and streamline optimization procedures, thereby enhancing the efficiency and effectiveness of industrial processes. However, it is essential to acknowledge the ongoing need for evaluating and discussing alternative strategies to enhance the network predictive accuracy, given the inherent complexity and challenges inherent in industrial process control. Continued research in this area promises to unlock further advancements in ESN applications, driving innovation and optimization within industrial processes.

Acknowledgment

This study was funded in part by the Fundação de Amparo à Pesquisa e Inovação do Espírito Santo – FAPES.

The authors also acknowledge the financial support from the CNPq and FAPERJ funding agencies.

References

[1] A. J. Silva Neto and J. C. Becceneri, “Técnicas de inteligência computacional inspiradas na natureza: Aplicação em problemas inversos em transferência radiativa,” 2009.

[2] C. P. Naveira-Cotta et al., “Eigenfunction expansions for transient diffusion in heterogeneous media,” International Journal of Heat and Mass Transfer, vol. 52, no. 21-22, pp. 5029–5039, 2009.

[3] D. C. Knupp, “Integral transform technique for the direct identification of thermal conductivity and thermal capacity in heterogeneous media,” International Journal of Heat and Mass Transfer, 2021.

[4] F. P. Incropera et al., Fundamentals of Heat and Mass Transfer, vol. 6, New York, Wiley, 1996.

[5] F. S. Mascouto et al., “Detection of contact failures employing combination of integral transforms with single-domain formulation, finite differences, and Bayesian inference,” Numerical Heat Transfer, Part A: Applications, 2020.

[6] J. Beck and S.-K. Au, “Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation,” Journal of Engineering Mechanics, vol. 128, no. 4, pp. 380–391, 2002.

[7] J. Ching and J. S. Wang, “Application of the transitional Markov chain Monte Carlo algorithm to probabilistic site characterization,” Engineering Geology, vol. 203, pp. 151–167, 2016.

[8] J. Ching and Y. C. Chen, “Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging,” Journal of Engineering Mechanics, vol. 133, no. 7, pp. 816–832, 2007.

[9] J. P. Kaipio and C. Fox, “The Bayesian framework for inverse problems in heat transfer,” Heat Transfer Engineering, vol. 32, no. 9, pp. 718–753, 2011.

[10] L. A. Da Silva Abreu et al., “Estimativa do perfil de temperatura na entrada de dutos via Método de Monte Carlo com Cadeias de Markov,” Revista Cereus, vol. 14, no. 4, pp. 129–143, 2022.

[11] M. N. Özışık and H. R. Orlande, Inverse Heat Transfer: Fundamentals and Applications, 2021.

[12] P. Gardner, C. Lord, and R. J. Barthorpe, “A unifying framework for probabilistic validation metrics,” *Journal of Verification, Validation and Uncertainty Quantification*, vol. 4, no. 3, 031005, 2019.

[13] W. Betz, I. Papaioannou, and D. Straub, “Transitional Markov chain Monte Carlo: observations and improvements,” Journal of Engineering Mechanics, vol. 142, no. 5, 04016016, 2016.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, pp.68–73, 1892.

AUTHORS

Laisa Cristina Juffo Campos

Laisa Cristina Juffo Campos is Brazilian, born in the state of Espírito Santo. She completed a technical course in Informatics at the Federal Institute of Espírito Santo (IFES), where she gained experience with various programming languages. Her dedication led her to participate in an extension project during her second year, involving Arduino programming and the development of mobile device software. Currently, she is pursuing a degree in Chemical Engineering at the Federal University of Espírito Santo (UFES). During her studies, she has applied her programming knowledge to various course-related problems, particularly in Numerical Methods and Process Control. Her research focuses on exploring the applications of artificial neural networks, with a specific emphasis on the Echo State Network (ESN) architecture. Recently, she began studying physics-informed neural networks (PINNs) and plans to combine PINNs and ESN in her future research. With a strong interest in the intersection of artificial intelligence and chemical engineering, she aims to develop innovative methodologies that can contribute to significant advancements in the field. In her free time, she enjoys drawing, playing the piano, and birdwatching.

Wellington Betencurte da Silva

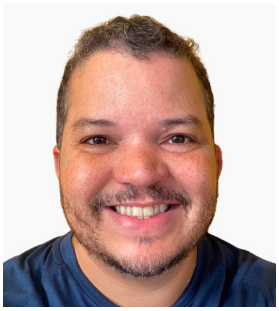
Wellington Betencurte da Silva is a Brazilian professor and researcher with a strong academic background and extensive experience in Mathematics and Mechanical Engineering. He graduated in Mathematics from the Federal Fluminense University in 2006, followed by a master's degree in Mechanical Engineering from the Military Institute of Engineering in 2008 and a Ph.D. in Mechanical Engineering from the Federal University of Rio de Janeiro in 2012. Currently, he serves as an associate professor at the Federal University of Espírito Santo, where he conducts research in the areas of Inverse Problems, Bayesian Filters, State and Parameter Estimation, and Heat Transfer. His expertise and academic contributions have been recognized in various master's dissertations and undergraduate thesis projects, where he has served as a supervisor and participated in examining boards. With a solid academic background and a commitment to excellence in research and teaching, Wellington Betencurte da Silva is a prominent figure in the Brazilian academic scene, making significant contributions to the advancement of knowledge in his field of expertise.

AUTHORS

Ana Carolina Spindola Rangel Dias

Ana Carolina Spindola Rangel Dias, a Brazilian born in Minas Gerais state, holds a Bachelor's degree in Chemical Engineering from the Federal University of Espírito Santo (2015) and a Master's degree in Chemical Engineering from the same institution (2017). Her master's dissertation focused on the control of a propylene polymerization reactor using particle filters and neural networks. A research internship was completed at the Norwegian University of Science and Technology (NTNU) from April 2019 to October 2020. She completed a Ph.D. in Chemical and Biochemical Process Engineering from the School of Chemistry at the Federal University of Rio de Janeiro in February 2023, with a thesis on developing predictive and self-optimizing controllers based on operational data. Previously, she worked as a temporary professor in the Department of Rural Engineering at the Federal University of Espírito Santo from March 2015 to December 2016. Currently, she is the lead researcher for the intelligent systems team on the Chemical Processes platform at the Senai Institute of Innovation in Biosynthetic and Fibers. Research interests include modeling, simulation, control, and optimization of processes. In her free time, she enjoys kpop music, movies and traveling.

Julio Cesar Sampaio Dutra



Julio Cesar Sampaio Dutra, born in Rio de Janeiro, Brazil, obtained his Bachelor's in Chemical Engineering from the Federal Rural University of Rio de Janeiro (UFRRJ) in 2007. He completed a direct-entry PhD in Chemical Engineering at the Federal University of Rio de Janeiro (UFRJ) in 2012, which included a research exchange at Norges Teknisk-Naturvitenskapelige Universitet (NTNU). Julio has been a faculty member since 2013 at the Federal University of Espírito Santo (UFES) as an Associate Professor. His research focuses on Mathematical Modeling, Process Simulation, and Process Control. He is particularly interested in machine learning and estimation schemes for monitoring, combined with control structure design using PID controllers, advanced process control strategies, and estimation algorithms, like Kalman filters and Sequential Monte Carlo methods. Considering such topics, he has extensive experience teaching and advising undergraduate and graduate students in Chemical Engineering. In addition to these activities, Julio is involved in administrative activities and committed to serving on deliberative committees. He enjoys cooking, drinking red wine, and traveling in his free time. In the future, Julio plans to continue exploring new techniques to address emerging challenges in Chemical Engineering.

Estimation of Spatially Dependent Coefficients in Heterogeneous Media in Diffusive Heat Transfer Problems

ARTICLE HISTORY
Received 24 February 2024
Accepted 19 April 2024
Published 08 July 2024

Lucas Lopes da Silva Costa
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
costa.lucas@iprj.uerj.br
ORCID: 0000-0003-1940-0875

Eduardo Cunha Classe
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
eduardo.classe@iprj.uerj.br
ORCID: 0000-0002-2405-3946

Lucas da Silva Asth
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
lucas.asth@iprj.uerj.br
ORCID: 0000-0002-6189-1068

Luiz Alberto da Silva Abreu
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
luiz.abreu@iprj.uerj.br
ORCID: 0000-0002-7634-7014

Diego Campos Knupp
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
diegoknupp@iprj.uerj.br
ORCID: 0000-0001-9534-5623

Leonardo Tavares Stutz
Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
ltstutz@iprj.uerj.br
ORCID: 0000-0003-3005-765X

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

Estimation of Spatially Dependent Coefficients in Heterogeneous Media in Diffusive Heat Transfer Problems

Lucas Lopes da Silva Costa
Universidade do Estado do Rio de Janeiro
Postgraduate Program in Computational Modeling, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
costa.lucas@iprj.uerj.br

Eduardo Cunha Classe
Universidade do Estado do Rio de Janeiro
Postgraduate Program in Computational Modeling, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
eduardo.classe@iprj.uerj.br

Lucas da Silva Asth
Universidade do Estado do Rio de Janeiro
Postgraduate Program in Computational Modeling, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
lucas.asth@iprj.uerj.br

Luiz Alberto da Silva Abreu
Universidade do Estado do Rio de Janeiro
Dept. de Engenharia Mecânica e Energia, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
luiz.abreu@iprj.uerj.br

Diego Campos Knupp
Universidade do Estado do Rio de Janeiro
Dept. de Engenharia Mecânica e Energia, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
diegoknupp@iprj.uerj.br

Leonardo Tavares Stutz
Universidade do Estado do Rio de Janeiro
Dept. de Engenharia Mecânica e Energia, Polytechnic Institute of Rio de Janeiro
Nova Friburgo, Brazil
ltstutz@iprj.uerj.br

***Abstract*—** This article addresses the solution to the inverse problem in a one-dimensional transient partial differential equation with a source term, commonly encountered in heat transfer modeling for diffusion problems. The equation is utilized in a dimensionless form to derive a more general solution that is applicable in various contexts. The Transition Markov Chain Monte Carlo (TMCMC) method is utilized to estimate spatially variable thermophysical properties within the equation. This approach involves transitioning between probability densities, gradually refining the prior distribution to approximate the posterior distribution. The results indicate the effectiveness of the TMCMC method in addressing this inverse problem, and it offers a robust methodology for estimating spatially variable coefficients.

Keywords*—Inverse Problem, Transition Markov Chain Monte Carlo (TMCMC), Heterogeneous Media, Estimation of Variable Coefficients, Heat Conduction*

I. Introduction

The identification of thermophysical properties is a fundamental process in various fields of science and engineering, where understanding these properties is essential to comprehend material behavior or identify them. Properties like thermal conductivity, density, and specific heat directly influence how a material responds to temperature changes [4]. When modeling the heat transfer process, these properties can be expressed through parameters within partial differential equations [3] [5] [10]. This, in turn, paves the way for varied approaches in estimating these parameters, ranging from direct methods to indirect approaches, each carrying its own advantages and disadvantages.

Within direct methods, direct experimental measurements on thermophysical properties of material samples are conducted. While recognized for their precision, these methods often prove to be costly, time-consuming, and in certain cases, intrusive to the material under analysis.

On the other hand, indirect methods offer an attractive alternative. They do not demand direct measurements of thermophysical properties but instead explore relationships between these properties and other variables that can be more easily measured [1][11]. However, indirect methods often rely on assumptions and models to establish these relationships, introducing uncertainties in the estimation.

One particular approach that has gained prominence is the utilization of Bayesian frameworks, such as the Transitional Markov Chain Monte Carlo (TMCMC) method, to estimate thermo-physical properties. The distinctive feature of Bayesian methods is the incorporation of prior information, i.e., prior knowledge about the properties in question [11]. TMCMC, for instance, constructs a probability distribution that takes into account both experimental data and prior information, resulting in more reliable estimates and quantified uncertainties [9] [11].

The aim of this work is to demonstrate the utilization of the TMCMC technique for computing unspecified parameters in a differential equation, proposing three distinct models of their spatial variation. The obtained results demonstrate the effectiveness of the TMCMC method in solving the inverse problem, providing a robust methodology for this type of problem. Furthermore, this work may validate the use of TMCMC as a reliable and versatile tool for parameter estimation in different contexts, paving the way for more advanced applications, such as characterizing new materials with different thermal properties.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

II. METHODOLOGY

In this section, the methodology employed in this study will be presented. In the subsequent subsections, the mathematical formulation of the physical problem will be explained, and the intricacies of Transition Markov Chain Monte Carlo (TMCMC) will be explored. Introduced by [8], this approach draws inspiration from the adaptive Metropolis-Hastings technique and employs Monte Carlo principles through Markov Chains. A comprehensive overview of the TMCMC method will be provided, including discussions on its fundamental principles and procedural steps. The aim of this exposition is to provide a clear understanding of how the TMCMC method operates, especially in the context of estimating coefficients in solving inverse problems.

A. Mathematical Formulation

In this section, the mathematical formulation underlying the physical phenomenon of heat transfer within a material of length L=10 will be delved into. This investigation considers Neumann boundary conditions coupled with a constant initial condition. The primary objective of this section is to model the dynamic evolution of temperature, represented as T(x,t), across space and time.

$$w(x)\frac{\partial T(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(k(x)\frac{\partial T(x,t)}{\partial x}\right)+p(x) \qquad (1a)$$

In this context, k(x) represents the thermal conductivity coefficient, a measure characterizing an intrinsic ability of a material to conduct heat. In turn, the coefficient w(x), known as the thermal diffusion coefficient, incorporates the inherent thermal diffusivity property of the material in question. The term p(x) refers to an internal heat source within the material. The spatial domain is defined in the interval $0 < x < L$, while time is restricted to positive values, $t > 0$, where L denotes the physical extent of the material. These parameters are expressed in terms of Neumann boundary conditions:

$$\left.\frac{\partial T(x,t)}{\partial x}\right|_{x=0} = 0 \qquad (1b)$$

$$\left.\frac{\partial T(x,t)}{\partial x}\right|_{x=L} = 0 \qquad (1c)$$

These expressions characterize the rates of heat transfer at the material boundaries, and the initial condition is established as shown below, where T_0 is a constant representing the initial temperature distribution within the material.

$$T(x,0) = T_0 \qquad (1d)$$

This study examines Equation (1) in three distinct scenarios: firstly, when both coefficients k(x) and w(x) are kept constant; secondly, when they are modeled as linear functions; and finally, when they follow exponential functions. The primary aim of these analyses is to assess the Transitional Markov Chain Monte Carlo (TMCMC) method ability to accurately estimate these parameters.

Transitional Markov Chain Monte Carlo (TMCMC)

The Transitional Markov Chain Monte Carlo (TMCMC) method, as proposed by [8], draws inspiration from the adaptive Metropolis-Hastings method as suggested by [6], and it is grounded on the Monte Carlo methodology through Markov Chains. The main idea is to avoid direct sampling of difficult probability distributions by sampling from a series of intermediate distributions that converge to the posterior distribution [8].

This method inherits the advantages of Adaptive Metropolis-Hastings (AMH), which is suitable for very sharp, flat, and multimodal probability density functions (PDFs), and is particularly efficient in high-dimensional PDFs. Additionally, the TMCMC method has the capability to automatically select intermediate PDFs, enhancing its versatility and effectiveness in sampling complex distributions [8].

The posterior distribution is calculated using Bayes' theorem, described by Equation (2) [9], as shown below:

$$\pi(P|Y) \propto \pi(P)\pi(Y|P) \qquad (2)$$

But, as mentioned earlier, the TMCMC method avoids computing the distribution in this way, in order to employ a series of intermediate distributions as follows:

$$f_j(P) \propto \pi(P)\pi(Y|P)^{p_j} \qquad (3)$$

The steps for the TMCMC algorithm are outlined as follows [7]:

1. Samples $\{P_{0,1}, P_{0,2}, ..., P_{0,n}\}$ are acquired from the prior distribution $f_0(P) = \pi(P)$ using Monte Carlo simulation. The process initiates with p_0 set to 0, and steps 2 and 3 are repeated for $j = \{0, 1, 2, ...\}$.
2. Likelihood distributions $\pi(Y \mid P_{j,1}), ..., \pi(Y \mid P_{j,n})$ are computed, and the weights $w_{j,k} = \pi(Y \mid P_{j,k})^{(p_{j+1} - p_j)}$ are determined. The selection of p_{j+1} ensures that the coefficient of variation (COV) of the importance weights $\{w_{j,1}, ..., w_{j,n}\}$ equals 100%. Additionally, normalized weights $\{w_{j,1}, ..., w_{j,n}\}$ are calculated.
3. Based on the normalized weights $\{w_{j,1}, ..., w_{j,n}\}$, candidates are randomly chosen from $\{P_{j,1}, P_{j,2}, ..., P_{j,n}\}$. A new candidate is proposed according to the distribution $N(P_{j,k}, \Sigma_j)$, forming the sequence $\{P_{j+1,1}, P_{j+1,2}, ..., P_{j+1}\}$. The covariance matrix Σ_j is defined by an equation.

$$\sum_j = \beta^2 \sum_{j=1}^{n_j} w_{j,k}\left[\left(P_{j,k}-\bar{P}_j\right)\times\left(P_{j,k}-\bar{P}_j\right)^T\right] \qquad (4a)$$

with

$$\bar{P}_j = \frac{\sum_{l=1}^{n_j} w_{j,l}\cdot P_{j,l}}{\sum_{l=1}^{n_j} w_{j,l}} \qquad (4b)$$

The parameter β is a factor that scales the distribution of the covariance matrix proposal [8].

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

III. Results

In this section, the study conducted a comparative analysis of the parameters w(x) and k(x) estimated in the Partial Differential Equation (PDE), as mentioned earlier. Three distinct variations were considered: constant, linear, and exponential. To obtain experimental measurements in the direct problem, three spatial measurement points were selected: x=2.5, x=5.0, and x=7.5, resulting in a total of 101 measurements for each sensor within the analyzed time interval. The standard deviations of the measurement errors σ were defined as 0.5 for the constant case, 1.0 for the linear case, and 1.5 for the exponential case. Therefore, these experimental measurements are now referred to as actual measurements. It is worth noting that these measurement errors were chosen to be proportional to the measured temperature, specifically around 2%. This choice was based on the averaging of experimental measurements for each model.

The Transition Markov Chain Monte Carlo (TMCMC) method was employed to simultaneously obtain estimates of these parameters, and the results were compared for each variation. The study was conducted with a total of 20,000 samples for the Constant model, 50,000 for the Linear model, and 50,000 for the Exponential model and β = 0.1 in all three situations. In order to simulate a source with characteristics of a smooth step curve, the following mathematical formulation for p(x) was used.

$$p(x) = 1 - \frac{1}{[1+e^{-100(x-0.5L)}]} \quad (6)$$

The specific formulations and characteristics of the analyzed models for w(x) and k(x) are detailed in the following sections, accompanied by their respective mathematical formulations and corresponding results.

It is important to emphasize that all results presented in this work were generated using the computational platform Wolfram Mathematica 12.0, operating on a desktop equipped with a Central Processing Unit (CPU) AMD Ryzen Threadripper1950x clocked at 4 GHz and 64 GB of DDR4 type RAM. The adopted operating system is Windows 10 in its 64-bit version.

A. Model with Constant Coefficients

Firstly, the TMCMC method was applied to estimate the parameters of a model with constant coefficients. In the direct problem, k(x) = 1 and w(x) = 1 were used. Table *I* below shows the exact values of the coefficients, as well as the results obtained after the method was applied.

TABLE I. Estimated Results via TMCMC – Model with Constant Coefficients

Parameter	*Exact Value*	*Estimated*	*Standard Deviation*	*Error(%)*
w_0	1.00	1.00056	0.00142	0.056
k_0	1.00	1.00641	0.01269	0.641

The table analysis reveals that the method was effective in parameter estimation, resulting in reduced relative errors and standard deviations. Fig. 1 depicts a comparison between estimated values, represented by the blue curve, and actual measurements denoted by red points, along with the 95% confidence interval depicted by the blue shaded region. On the other hand, Fig. 2 presents the residual analysis of the three utilized sensors, along with their corresponding linear regression.

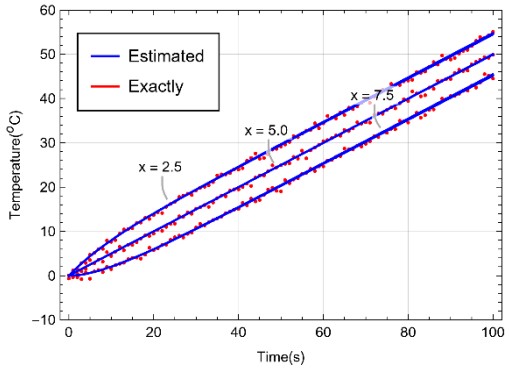


Fig. 1. Temperature Measurements with 95% confidence interval - Model with Constant Coefficients

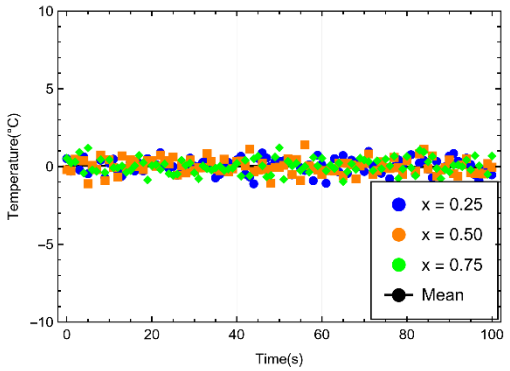


Fig. 2. Residual analysis – Model with Constant Coefficients

Through the analysis of the graphs, it is evident that the estimated measurements exhibit high agreement with the actual measurements. The residual analysis reveals that the differences between these measurements are close to zero across the entire domain, as evidenced by the linear regression. Fig.3 and Fig. 4 illustrate the histogram of the estimates for the parameters w(x) and k(x). It is important to note that all estimated samples were normalized by their respective exact values, rendering the histogram dimensionless.

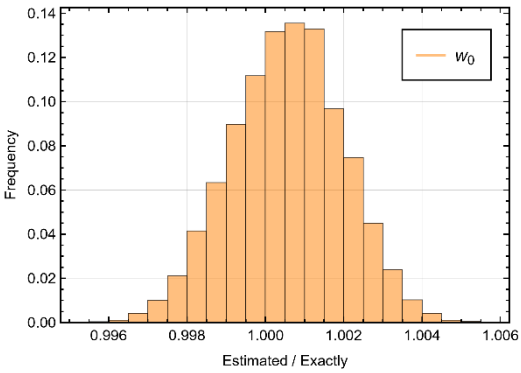


Fig. 3. Histogram of the w_0 estimated parameter - model with constant coefficients

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

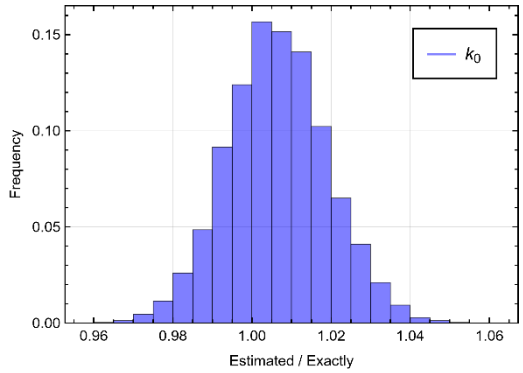


Fig. 4. Histogram of the k_0 estimated parameter - model with constant coefficients

The means of the estimated values are close to the exact values, as expected. It is noteworthy that the estimation for the parameter w(x) was more accurate than for the parameter k(x).

B. Model with Linear Coefficients

Similarly to the model with constant coefficients, Table *II* presents the values used in solving the direct problem for the case of linear coefficients in the form $w(x)=w_o x+w_1$ and $k(x)=k_0 x+k_1$. The corresponding estimates, standard deviations, and relative errors are also indicated.

TABLE II. ESTIMATED RESULTS VIA TMCMC - MODEL LINEAR WITH LINEAR COEFFICIENTS

Parameter	*Exact Value*	*Estimated*	*Standard Deviation*	*Error(%)*
w_0	0.09	0.08992	0.00185	0.091
w_1	0.10	0.10021	0.00857	0.215
w_0	0.09	0.09099	0.00552	1.101
k_1	0.90	0.09117	0.02360	8.832

Except for the parameter k_1, all estimates yielded relative errors of less than 3%. Fig. 5 illustrates the comparison between the measurements of estimated values, represented by the blue curve, and actual measurements denoted by red points, along with the 95% confidence interval depicted by the blue shaded region. Meanwhile, Fig. 6 displays the residual analysis between these two measurements and the linear regression of the points.

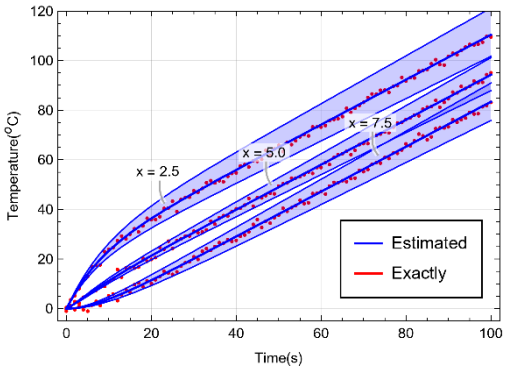


Fig. 5. Temperature Measurements with 95% confidence interval – Model with Linear Coefficients

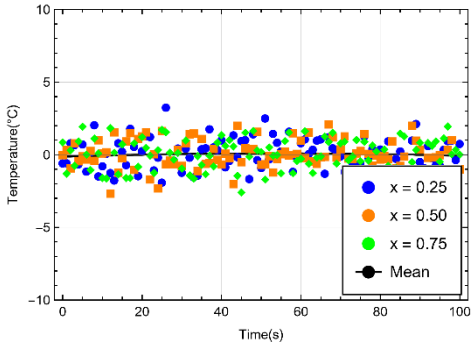


Fig. 6. Residual analysis - Model with Linear Coefficients

Once again, a remarkable resemblance is observed between the estimated measurements and the actual measurements. However, it is noticeable that for the linear case, the confidence interval encompasses all the conducted measurements. The residual analysis demonstrates that the differences between the measurements are close to zero across the entire domain, as shown by the linear regression. Fig. 7, Fig. 8, Fig. 9 and Fig. 10 displays the histograms of parameter estimates for this case.

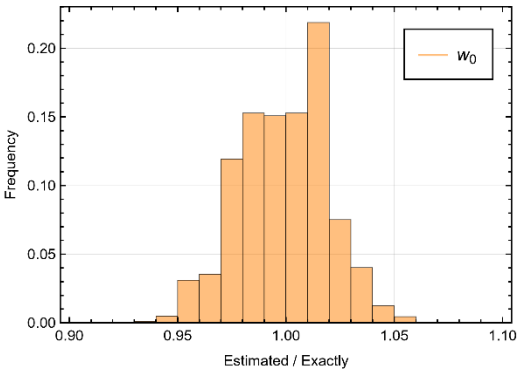


Fig. 7. Histogram of the w_0 estimated parameter - model with linear coefficients

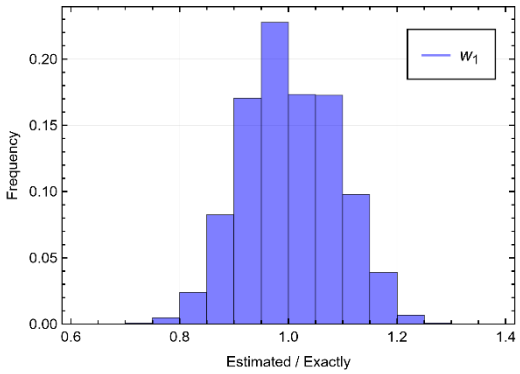


Fig. 8. Histogram of the w_1 estimated parameter - model with linear coefficients

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

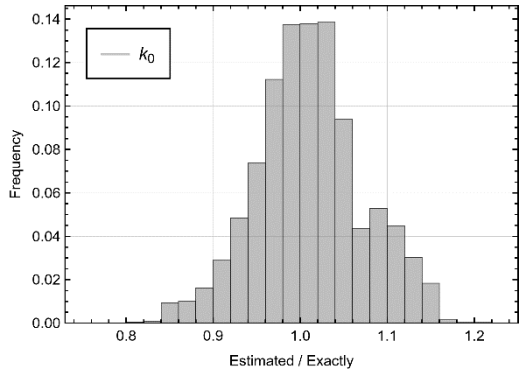


Fig. 9. Histogram of the k_0 estimated parameter - model with linear coefficients

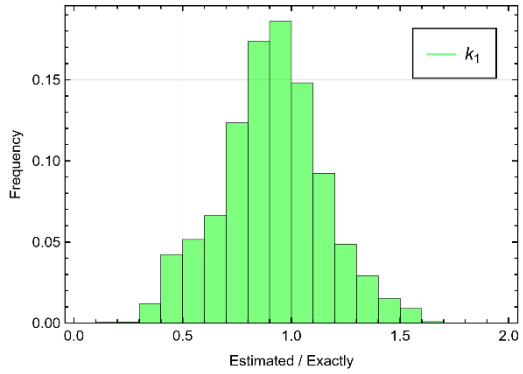


Fig. 10. Histogram of the k_1 estimated parameter - model with linear coefficients

The means of the estimated values approach the exact values, as expected, reinforcing the reliability of the TMCMC method. It is worth noting that the estimate for the parameter w(x) reveals superior precision compared to the parameter k(x), suggesting the need for a more in-depth analysis to comprehend the underlying causes of this discrepancy.

C. Model with Exponential Coefficients

Finally, Table *III* showcases the values and estimates of the parameters associated with the case of exponential coefficients in the form $w(x)=w_0e^{w_1x}$ and $k(x)=k_0e^{k_1x}$.

TABLE III. ESTIMATED RESULTS VIA TMCMC - MODEL WITH LINEAR COEFFICIENTS

Parameter	*Exact Value*	*Estimated*	*Standard Deviation*	*Error (%)*
w_0	0.10	0.10159	0.00157	1.595
w_1	0.25	0.24738	0.00249	1.046
w_0	0.10	0.10066	0.00226	0.663
k_1	0.25	0.24637	0.00487	1.450

Once again, the estimates resulted in significantly reduced relative errors and standard deviations. Fig. 11 illustrates the comparison graphs between the measurements with the estimated parameters, represented by the blue curve, and the actual measurements denoted by red points, along with the 95% confidence interval depicted by the blue shaded region. Meanwhile, Fig. 12 displays the residual analysis between these measurements with the linear regression of the points.

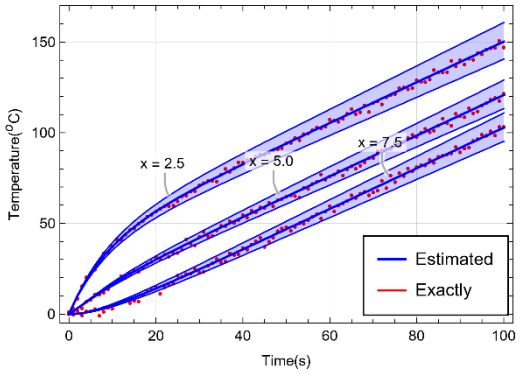


Fig. 11. Temperature Measurements with 95% confidence interval - Model with Exponential Coefficients

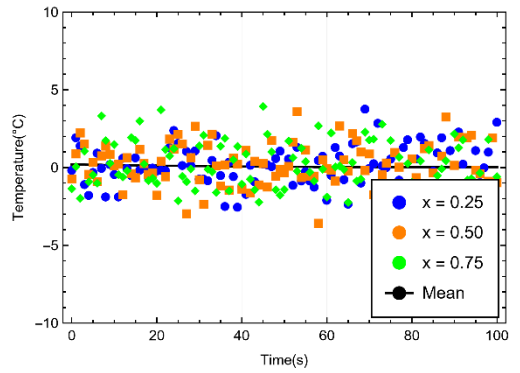


Fig. 12. Residual analysis - Model with Exponential Coefficients

Similar to the previous cases, the estimated measurements exhibit high agreement with the actual measurements. The residual analysis confirms that the differences between these measurements are close to zero across the entire domain. Figs.13, 14, 15 and 16 display the histograms of parameter estimates for this case.

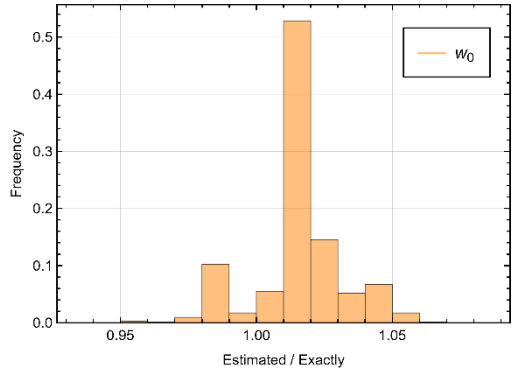


Fig. 13. Histogram of the w_0 estimated parameter - model with exponential coefficients

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

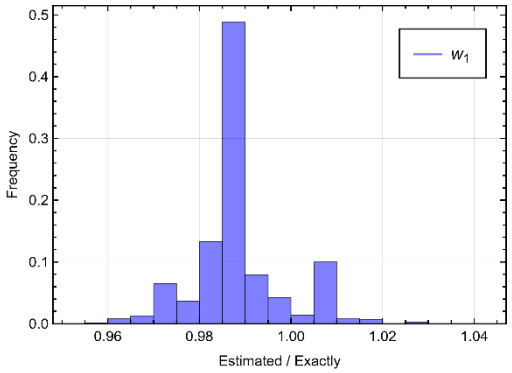


Fig. 14. Histogram of the w_1 estimated parameter - model with exponential coefficients

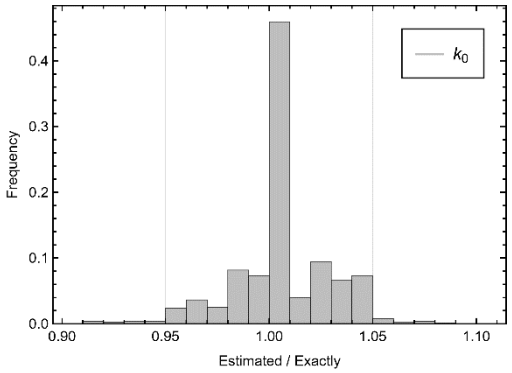


Fig. 15. Histogram of the k_0 estimated parameter - model with exponential coefficients

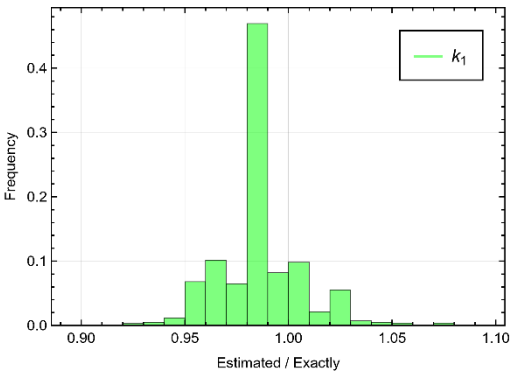


Fig. 16. Histogram of the k_1 estimated parameter - model with exponential coefficients

It is important to note that despite this change in distribution, the TMCMC method demonstrated estimating the parameters with lower relative error compared to the previous linear cases. This observation underscores the relative capability of the method in dealing with exponential coefficients, even with the loss of uniformity in histograms, indicating a relative precision in estimating these parameters.

IV. Conclusion

Throughout this study, the evaluation of the TMCMC method efficacy in estimating coefficient parameters was conducted across three distinct scenarios. The analysis of the obtained tables and histograms reveals variability in the method efficiency based on the analyzed case. Notably, it was found that the method faced more significant challenges in estimating parameters for k(x) in the second scenario, corresponding to a linear model. Despite this additional complexity, the relative error consistently remained below 9%.

A detailed analysis of the generated histograms allows for a deeper understanding of the results. In all investigated scenarios, a notable precision was observed in estimating the parameters. In the constant model case, the value distribution showed a well-defined Gaussian shape, centered around the exact value, demonstrating highly accurate estimation. However, in the linear case, there was a more significant dispersion in the probable values, especially considering the parameters associated with k(x). Lastly, in the third case, an even higher precision compared to the linear case was highlighted, along with the presence of distributions that appeared to be bimodal, indicating the occurrence of two peaks of probable values for the k(x) parameters, something that warrants further investigation.

These results offer a comprehensive insight into the applicability and performance of the TMCMC method in parameter estimation, highlighting its nuances across different model configurations. The achieved accuracy, even in the face of specific challenges, underscores the robustness and potential of this method for parameter analyses and inferences across various contexts.

Acknowledgment

The present work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Funding Code 001.

The authors also acknowledge the financial support from the CNPq and FAPERJ funding agencies.

References

[1] A. J. Silva Neto and J. C. Becceneri, "Técnicas de inteligência computacional inspiradas na natureza: Aplicação em problemas inversos em transferência radiativa," 2009.

[2] C. P. Naveira-Cotta et al., "Eigenfunction expansions for transient diffusion in heterogeneous media," International Journal of Heat and Mass Transfer, vol. 52, no. 21-22, pp. 5029–5039, 2009.

[3] D. C. Knupp, "Integral transform technique for the direct identification of thermal conductivity and thermal capacity in heterogeneous media," International Journal of Heat and Mass Transfer, 2021.

[4] F. P. Incropera et al., Fundamentals of Heat and Mass Transfer, vol. 6, New York, Wiley, 1996.

[5] F. S. Mascouto et al., "Detection of contact failures employing combination of integral transforms with single-domain formulation, finite differences, and Bayesian inference," Numerical Heat Transfer, Part A: Applications, 2020.

[6] J. Beck and S.-K. Au, "Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation," Journal of Engineering Mechanics, vol. 128, no. 4, pp. 380–391, 2002.

[7] J. Ching and J. S. Wang, "Application of the transitional Markov chain Monte Carlo algorithm to probabilistic site characterization," Engineering Geology, vol. 203, pp. 151–167, 2016.

[8] J. Ching and Y. C. Chen, "Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging," Journal of Engineering Mechanics, vol. 133, no. 7, pp. 816–832, 2007.

[9] J. P. Kaipio and C. Fox, “The Bayesian framework for inverse problems in heat transfer,” Heat Transfer Engineering, vol. 32, no. 9, pp. 718–753, 2011.

[10] L. A. Da Silva Abreu et al., “Estimativa do perfil de temperatura na entrada de dutos via Método de Monte Carlo com Cadeias de Markov,” Revista Cereus, vol. 14, no. 4, pp. 129–143, 2022.

[11] M. N. Özışık and H. R. Orlande, Inverse Heat Transfer: Fundamentals and Applications, 2021.

[12] P. Gardner, C. Lord, and R. J. Barthorpe, “A unifying framework for probabilistic validation metrics,” *Journal of Verification, Validation and Uncertainty Quantification*, vol. 4, no. 3, 031005, 2019.

[13] W. Betz, I. Papaioannou, and D. Straub, “Transitional Markov chain Monte Carlo: observations and improvements,” Journal of Engineering Mechanics, vol. 142, no. 5, 04016016, 2016.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, pp.68–73, 1892.

LATIN-AMERICAN JOURNAL OF COMPUTING (LAJC), Vol XI, Issue 2, July 2024

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.02

AUTHORS

Lucas L. da Silva Costa

Lucas Lopes da Silva Costa. (São Sebastião do Alto, Rio de Janeiro, Brazil, July 11th, 1999). B.Sc. in Applied and Computational Mathematics from Fluminense Federal University (UFF), Brazil, 2021. Currently pursuing a Master's degree in Computational Modeling at the State University of Rio de Janeiro UERJ), which began in 2022, aiming to deepen knowledge in Applied Mathematics and Scientific Computing. During his undergraduate studies, he actively participated in the Mathematics of Epidemics extension project (PEB) and contributed as a student member in course coordination. Alongside his master's studies, he is also pursuing a Teaching Degree in Mathematics at UFF, demonstrating a strong commitment to pedagogical training. Additionally, he gained practical experience as an intern in the Education sector with the Municipal Government of Macuco - RJ, applying theoretical knowledge in a real-world context between 2021 and 2022. His research interests include mathematical modeling, computational simulation, and educational methodologies in mathematics.

Eduardo Cunha Classe

Eduardo Cunha Classe, Nova Friburgo, Rio de Janeiro, Brazil, April 29, 1998. B.Sc. in Mechanical Engineering from the State University of Rio de Janeiro (UERJ), 2021. During his undergraduate studies, he participated in a scientific initiation project for the characterization of corrosion resistance of stainless steel alloys, was a member of the Baja SAE program, and a member of the SPE student chapter. Currently pursuing a Master's degree in Computational Modeling at the same university, which began in 2022, aiming to deepen his knowledge in heat transfer and computational modeling.

Lucas da Silva Asth

Lucas da Silva Asth (Nova Friburgo, Brazil, November 25th, 1996). B.Sc. in Mechanical Engineering from the State University of Rio de Janeiro, Brazil, 2021. MSc (2024) and currently pursuing a Ph.D. in Computational Modeling at the State University of Rio de Janeiro (UERJ). his research interests include heat and mass transfer, structural dynamics, inverse problems and optimization.

AUTHORS

Luiz Alberto da Silva Abreu

Luiz A. S. Abreu was born in Nova Friburgo, Brazil, on December 15th, 1982. He obtained his B.Sc. in mechanical engineering from the Rio de Janeiro State University (UERJ) in 2009, his M.Sc. in mechanical engineering from the Federal University of Rio de Janeiro (UFRJ) in 2011 and his D.Sc. in mechanical engineering from the same university in 2014. He has been an affiliate member of the ABCM—Brazilian Society of Mechanical Sciences and Engineering since 2016. He has advised or co-advised over 12 DSc and MSc theses, most of them in the Graduate Program in Computational Modeling (PPGMC) at UERJ. He is the author of over 20 articles published in major scientific journals and conference proceedings and 5 book chapters. His research interests include the solution of inverse problems, as well as the use of meshfree, numerical, analytical, and hybrid numerical–analytical methods for solving direct problems, mainly in mechanical engineering applications.

Diego Campos Knupp

Diego C. Knupp (Nova Friburgo, Brazil, November 9th, 1984). B.Sc. in Mechanical Engineering from the State University of Rio de Janeiro, Brazil, 2009. MSc (2010) and DSc (2013) in Mechanical Engineering from the Federal University of Rio de Janeiro, Brazil. Currently Professor of Mechanical Engineering at the State University of Rio de Janeiro at the Polytechnique Institute - IPRJ/UERJ, heads the Laboratory Patricia Oliva Soares of Experimentation and Numerical Simulation in Heat and Mass Transfer, LEMA. Author of around 200 articles in major journals and conferences and one book published abroad. Advisor of over 18 DSc and MSc thesis, his research interests include hybrid methods, bioheat transfer, structural dynamics, inverse problems and optimization. Prof. Knupp has been affiliate member of the Brazilian Academy of Sciences (2019-2023) and currently serves as associate editor for the Annals of the Brazilian Academy of Sciences journal.

Leonardo Tavares Stutz

Leonardo Tavares Stutz, Brazilian, born in Nova Friburgo, Rio de Janeiro. He is currently Associate Professor at the Polytechnique Institute (IPRJ) of the State University of Rio de Janeiro (UERJ), Brazil, since 2006. He has Bachelor's degree (1997), Master degree (1999) and a Ph.D. (2005) in Mechanical Engineering from the Federal University of Rio de Janeiro (UFRJ), Brazil. He is the author of over forty articles published in scientific journals and conference proceedings. He supervised over fourteen mater's dissertations and doctoral thesis in the Graduate Program in Computational Modeling (PPGMC) at IPRJ. His research interests include Structural Dynamics and Vibration, Vibration Damping, Inverse Problems, Parameter Estimation, Bayesian Inference and Computational Modelling. Much of his work has been on structural damage identification problems and on viscoelastic parameter estimation problems.

Forensic Investigation in Robots

ARTICLE HISTORY
Received 08 March 2024
Accepted 05 May 2024
Published 08 July 2024

Tharmini Janarthanan
Sheffield Hallam University
Sheffield, United Kingdom
tharmini.janarthanan@shu.ac.uk
ORCID: 0009-0008-9047-8556

Shahrzad Zargari
Sheffield Hallam University
Sheffield, United Kingdom
s.zargari@shu.ac.uk
ORCID: 0000-0001-6511-7646

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

T. Janarthanan and S.Zargari, "Forensic Investigation in Robots", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024.

Forensic Investigation in Robots

Tharmini Janarthanan
Sheffield Hallam University
Department of Computing
Sheffield, United Kingdom
tharmini.janarthanan@shu.ac.uk

Shahrzad Zargari
Sheffield Hallam University
Department of Computing
Sheffield, United Kingdom
s.zargari@shu.ac.uk

***Abstract*—**Integrating robots into industrial automation has led to a revolutionary transformation in executing complex tasks, harnessing precision and efficiency. The Robot Operating System (ROS) has played a significant role in driving this advancement. ROS Bag files in robots are crucial for preserving data, as they provide a format for recording and playing back ROS message data. These files serve as a comprehensive log of a robot's sensory inputs and operational activities, enabling detailed analysis and reconstruction of the robot's interactions and performance over time. However, there have been instances where security considerations were overlooked, giving rise to concerns about unauthorized access, data theft, and malicious actions. This research investigates the forensic potential of data generated by robots, with a particular focus on ROS Bag data. By analyzing ROS Bag data, we aim to uncover how such information can be used in forensic investigations to reconstruct events, diagnose system failures, and verify compliance with operational protocols. The components of the ROS ecosystem were examined, identifying the challenges in parsing ROS Bag files and underscoring the need for specialized tools. This analysis highlights the security risks associated with plain text communication within legacy ROS systems, emphasizing the importance of encryption. While providing valuable insights, this research calls for further exploration, tool development, and enhanced security practices in robotics and digital forensics, aiming to lay the foundation for effective crime resolution involving robots.

Keywords*—Robot forensics, forensics, ROS, Cybersecurity*

I. Introduction

In recent years, researchers have been increasingly focused on enhancing industrial automation processes by integrating advanced robotic technologies. A key aspect to this progress is the Robot Operating System (ROS), which significantly boosts the speed and capabilities of robots, positioning them as indispensable assets to the industry. Particularly, robots play an indispensable role in critical sectors such as healthcare as they significantly contribute to patient care, medication administration, and surgical procedures. In fact, the data generated by robots holds valuable insights that assist medical professionals in making informed decisions, ultimately improving patient outcomes [1]. However, amidst the pursuit of these advancements, security considerations have frequently been overlooked, which can expose significant vulnerabilities that malicious actors might exploit [2]. For instance, unauthorized access to robot control systems, sensors, and data poses a substantial threat, potentially allowing adversaries to seize control, manipulate actions, or steal sensitive information. Moreover, inadequate security protocols may lead to data breaches, not only endangering the privacy of collected information, but also the intellectual property of the robotic system's critical functionality. Also, insecure communication channels could enable eavesdropping due to the lack of trust mechanisms which might allow unauthorized modifications to ROS software/firmware. Such insecure practices can lead to physical tampering, and unpredictable robot behavior with severe legal and reputational repercussions [3].

Since the robots' interaction with machinery can generate important digital traces, valuable insights can be retrieved and analyzed as potential evidence in forensic investigations. Particularly, analyzing *ROS bag data* can uncover significant artefacts, retrace robot movements, and reconstruct events [4], using traces produced by diverse sensor inputs like visual, audio, and environmental data [5]. In this study, we propose two primary objectives to enhance the investigation of cybercrimes involving robots: Firstly, developing an understanding of the Robot Operating System (ROS) and its underlying structure. Secondly, exploring the potential forensic artefacts that can be extracted from a ROS via scenario-based simulations.

The rest of the paper is organized as follows: Section II presents a literature review on ROS communication process, addressing its inherent security challenges while exploring the emerging field of ROS forensics. In Section III, an overview of the research methodology is provided. In Section IV the experimental setup for ROS-based forensic evidence retrieval is described. Later, Section V discusses the findings and the results, highlighting the significance of the artefacts discovered. Finally, conclusions and directions for future work are outlined in Section VI.

II. Literature Review

In this section, we discuss the characteristics of the Robot Operating System, its communication process as well as its security challenges, and the emerging field of ROS Forensics.

A. Robot Operating System (ROS)

ROS is a framework for developing robotic software, offering an extensive suite of tools and libraries. Its powerful features enable developers to facilitate message passing, perform distributed computations, reuse code, and implement algorithms for various robotic applications. A key objective of ROS is to create a standardized programming approach for robots, providing off-the-shelf software components that can be seamlessly integrated into custom robotic projects. Today, ROS has become the preferred platform for many leading robotics companies. This shift is also evident in industrial robotics, where companies increasingly transition from proprietary robotic applications to ROS [5].

ROS manages multiple distributed functional entities known as *nodes*, each representing an autonomous process with its own lifecycle within an application context. Central to a ROS's architecture is a dedicated entity operating on a specific host within the ROS network also known as a *master* which is responsible for overseeing and mediating operations. The master maintains a directory of all existing nodes and their corresponding data [6]. At the architecture's core, there

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.03

is the *publish-subscribe communication model*, which main purpose is to effectively simplify complex components and establish precise interfaces for their connections. This model uses a *topic-based approach*, creating virtual channels (or topics) for individual instances. Thus, subscribers can use such topics to access the transmitted information. For example, in the ROS environment, a sensor node capturing images from a camera would publish this visual data on a specific topic, allowing any node requiring this information to subscribe to the relevant topic. Within the publish-subscribe framework, the specific identities of the publisher and the subscriber are relatively unimportant, facilitating seamless swapping within a ROS network. This feature also streamlines the addition of existing nodes or their adaptation for new applications.

B. ROS Communication

In a ROS environment, the entire communication process adheres to the publish-subscribe paradigm for each action-related topic [5] [7]. The master keeps a comprehensive catalogue of all available services. ROS also supports client-server communication through services, enabling a service client to request connection details for a specific service [5] [6]. Since ROS communication can flexibly use both TCP (ROSTCP) and UDP (ROSUDP) protocols, a service, identified by a unique name, can be accessed interactively and synchronously by a client, serving various purposes, such as obtaining one-time information. During its initialization, a publisher node contacts the master to declare the topics it plans to publish [4]. Then, a subscriber communicates its topic requirements to the master. When the master finds a compatible match between a publisher and a subscriber, it informs the subscriber about potential publishers for the designated topic. Subsequent communication between these nodes occurs directly, bypassing the ROS master.

For complex tasks, such as directing a mobile robot, ROS uses a communication pattern utilizing five topics. In this case, the process begins with the client sending a goal to the server, which in turn provides continuous updates and feedback through dedicated topics, including the robot's location. The outcome of the task is communicated via a result topic, while a cancel topic allows for the termination of the task.

C. Security Challenges in Robots

Industry experts have already noted that although manufacturers initially prioritized the physical safety of human operators, and their interaction with robotic systems, robot cybersecurity is currently critical due to their exposure to a broader range of vulnerabilities [8]. Likewise, according to ABI Research [9], the number of connected industrial robots will reach 4.3 million units by 2025, highlighting their expanding attack surface, making them increasingly prone to cyberattacks, physical tampering, and ethical issues. While becoming more interconnected, autonomous, and capable of managing critical tasks, it is clearer that robot widespread adoption has significantly increased the complexity of managing cybersecurity-related challenges, affecting both industries and individuals. Particularly, individuals without formal training may unintentionally introduce security risks [3] [8] while developing robot platforms, applications, hardware, and sensors. In fact, the landscape of robot cybersecurity is defined by many attack surfaces, including the physical robot, operating system, software or firmware, remote control technologies, vendor Internet services, cloud services, and networks [10].

Conversely, hackers may target robots for various impactful reasons, such as manipulating them to introduce defects in manufactured parts or assemblies; thereby sabotaging production processes. They may also use ransomware tactics to coerce manufacturers into paying substantial ransoms to prevent the exposure of compromised production lots. In some cases, hackers cause physical damage to the robots or robotic cells themselves, posing a direct threat to human workers. The stakes are further heightened by the theft of critical information, intellectual property, and data manipulation, leading to erroneous decision-making [3]. Moreover, robots are significantly vulnerable to cybersecurity concerns due to limitations such as the absence of proper authorization or authentication, encryption deficiencies, and insufficient physical protection measures [11].

In contrast, ROS, serving as the foundational infrastructure for numerous robots, could become a target in operating system attacks aimed at exploiting vulnerabilities. An in-depth analysis involving 176 threats from the robot vulnerability database revealed that 92.6 per cent are predominantly linked to software-related issues. This highlights the elevated threat level posed by software in robotics systems compared to hardware components [12]. Besides, in [13], researchers illustrate how genuine attacker profiles targeting ROS-based robots can be discerned, providing valuable insights for experts in selecting appropriate ROS security solutions for mitigating man-in-the-middle (MITM) attacks. Furthermore, vulnerabilities extend to multi-robot active surveillance systems, where intruders can manipulate or disable any ROS node within the network using shutdown commands. Another study highlights the potential for attackers to misguide robots by tampering with velocity commands [14]. ROS environments also face significant risks due to unsecured communication ports. In [15], ROS-based attacks involving plain-text communication over unsecured ports are illustrated, potentially leading to unauthorized access. Also, while [16] showcased the exploitation of APIs by attackers to undermine ROS applications, [17] emphasized the interception, manipulation and disruption of communication between two ROS nodes which may cause not only poor performance and disruption in operations, but also a potential disclosure of sensitive information.

Therefore, as robots become more appealing targets for hackers, the need to understand the ROS file structure and the locations of data artefacts is essential for conducting forensic investigations in this domain in order to reduce the impact of security breaches and prevent future incidents.

D. ROS Forensic Investigation

ROS Forensics is an emerging field that rapidly explores digital traces to extract valuable insights about security vulnerabilities present in robots. These insights serve various purposes, including investigating cybercrimes, uncovering malicious activities, and providing evidence for legal proceedings. Despite the increasing importance of ROS Forensics in the context of robotics and the Internet of Things, research in this area remains relatively undeveloped due to the lack of understanding of the ROS file structure for identifying artefacts within these systems for effective incident response and thorough forensics examinations. Compared to studies on

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.03

ROS security, research on ROS forensics is sparse. An early notable contribution to this domain is the work of [18], which was focused on assessing ROS cyber-physical security. The authors identified vulnerabilities and potential threats to ROS-based systems through various attacks and scenarios, laying the groundwork for digital investigations in this field.

Conversely, researchers in [19] introduced a framework aimed at aiding investigators in retrieving digital evidence from such systems. The authors noted that ROS presents investigative challenges, particularly in real-time scenarios, due to the complexities of communication between robot components and the ROS framework. These complexities introduce supplementary data during the investigation process, thereby complicating the gathering of accurate evidence from ROS. The complex communication between robot components and the ROS framework in real-time scenarios introduces additional data during investigations. A deep understanding of the ROS file structure enables investigators to identify and extract relevant data streams efficiently. This capability aids in extracting relevant information while eliminating unnecessary data noise.

On the other hand, in [4], researchers conducted a forensic examination of ROS, describing the tools required for ROS memory acquisition. The authors used three memory forensics tools: the *DD command*, *LiME*, and the *Volatility memory*

framework. Their Kali Linux experiment involved executing fundamental breaches to acquire live memory data. The memory images collected with DD and LiME were verified using the Bdsum5 command. Subsequently, the analysis of memory images was compared with live response analysis. According to the report, live response activities disrupted ROS operational processes. Although the impact was less pronounced in the case of image forensics, it contributed to enhanced digital evidence integrity. Likewise, research carried out in [20], employed memory forensics in ROS using the Volatility tool. The authors utilized the *Linux_rosnode plugin*, facilitating memory investigation within ROS. Their study focused on collecting digital evidence from robot memory, specifically focusing on the cognition device, a part of ROS, to assess the possibility of tampering within the system. Unlike the study conducted in [20], this work presents a well-defined method and definitive outcomes.

As opposed to the previous approaches, [21] provided a comprehensive overview of ROS forensics, focusing on ROS 2 security features, security challenges, and vulnerabilities. This study revealed the difficulties distinguishing between a software bug and a deliberate attack within ROS systems. Such challenges arise from the goal of forensics investigations to identify potential attacks and their subsequent outcomes. The authors highlighted the research gap and the limited availability of in-depth ROS forensics investigation studies.

III. Methodology

To explore the challenges of forensics investigation in robotics domains, a simulated ROS environment was utilized to replicate real-life scenarios. Data was generated by engaging the robot in various movement activities. Although ROS 2 is the latest version of ROS and surpasses its predecessor, ROS 1, the latter has been used for over a decade and many robots still operate on ROS 1. Actually, ROS 1 is still supported by a substantial user community and a comprehensive repository of libraries, tools, and resources. Thus, in this study, ROS 1 was chosen due to the higher probability of having ROS 1 robots being targeted by potential attackers compared to their ROS 2 counterparts. After configuring the environment and recording robot activities to generate data, Autopsy and FTK Imager were employed to extract and analyze forensic artefacts from different locations. This aids in gaining a clearer understanding of the type of information and the location of valuable artefacts.

IV. Experimental Work

In this section, our proposed ROS experimental setup is explained.

A. ROS Environmental Setup

In this research, ROS was deployed on a Linux-based virtual machine to establish the laboratory environment. The machine operated on Windows 11, using VMware for virtualization. Within VMware, Ubuntu 20.04.06 (Focal Fossa) was installed as the guest operating system. The configuration included the installation of *ROS Noetic Ninjemys (ROS 1)* distribution systems on Ubuntu. Figure 1 illustrates the proposed experimental setup visually.

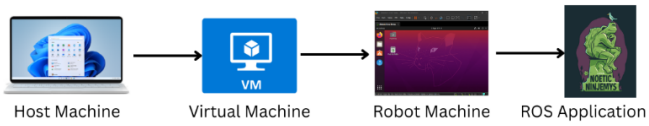


Fig. 1. Experimental setup of the ROS environment.

Additionally, the date and time settings on the Ubuntu virtual machine were adjusted to Pacific Daylight Time (Los Angeles, United States) to align with the default settings.

B. TurtleBot3 Experimentation and Data Generation

The next step involved setting up a simulated environment using the *TurtleBot3 model Waffle_Pi*. This simulation was designed to closely replicate the robot's behavior within a controlled virtual setting called *House World*. Additionally, the *teleoperation package* was employed to manually control the robot's movements within this environment.

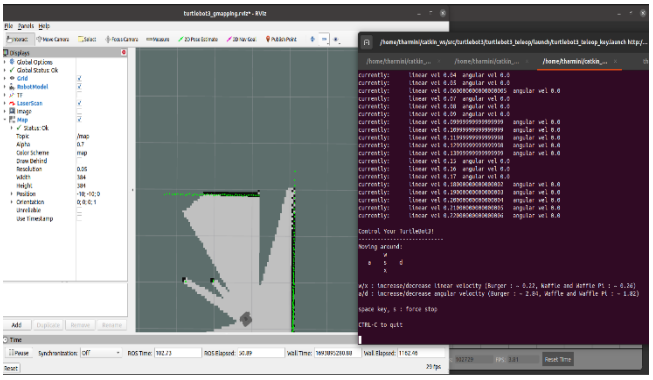


Fig. 2. Creating a SLAM map of the environment.

A key aspect of this experiment was the use of *Simultaneous Localization and Mapping (SLAM) algorithms*, which were instrumental in generating detailed maps of the environment while simultaneously tracking the robot's precise position (see Figure 2). This was mainly used to enable the robot to explore partially known environments using SLAM technology autonomously.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.03

An interesting observation during the experiment was the robot's ability to detect and adapt to new obstacles, including other robots or human entities within the evolving map. This ability allowed the robot to autonomously and dynamically adjust its navigation. In the mapping process, walls or obstacles were depicted in black, unobstructed areas in white, and uncharted or ambiguous regions in varying shades of grey or transparency.

Rviz, the ROS visualization tool, was employed to acquire real-time insights into sensor data, mapping progress, and the robot's precise position. This tool proved invaluable in monitoring and assessing robot interactions with their environment. However, we encountered performance-related issues, leading to frustrating instances of lag and reduced responsiveness. These issues had tangible consequences, notably slowing down the robot's movements and introducing complexities into the mapping process.

By default, the ROS tool saves the map into the home directory in two formats (unless specified otherwise): (i) *the map.pgm file* and (ii) *the map.yaml file*. The file (i) visualized the environment, featuring distinct white, grey, and black regions. These regions denoted various aspects of the mapped space, with black indicating walls or obstacles, white representing unobstructed areas, and grey or transparent sections signifying uncertainty regions. On the other hand, the file (ii) held essential configuration data for the *map.pgm image* [22]. This configuration data provided critical information about the map's scale, resolution, and other parameters for effectively interpreting and utilizing the map. This usually benefits the developers or researchers for future navigation undertakings and analysis. To end the SLAM process, the terminal operating the SLAM node was shut down. This completed the mapping and location process. With this capability, we could make the most of the TurtleBot3 for various SLAM-connected purposes, like self-governing navigation, exploration, and automation. Also, cameras and sensors were incorporated into the simulation environment to enhance the dataset and introduce a visual dimension to data collection, as shown in Figure 3.

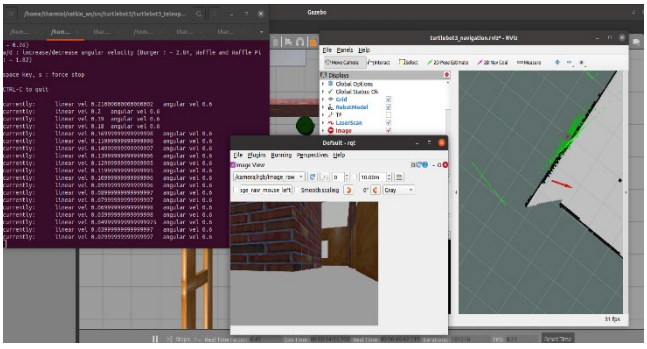


Fig. 3. Capturing of camera data by Turtlebot3.

This effort aimed to provide a more comprehensive understanding of the robot's interactions with its environment. The process of configuring and utilizing these sensors was simplified by the *rqt package* which offered user-friendly tools for setup. The cameras, acting as the robot's eyes, diligently recorded its visual observations while the author efficiently stored this data using ROS bags. These bags functioned as comprehensive repositories, capturing a wide array of sensor information and detailed accounts of the robot's movements. Moreover, the recorded data's adaptability was highlighted by the ease of playback and analysis, achievable through ROS commands or the convenient rqt tool (see Figure 4). This gave us a powerful means to revisit and scrutinize robot behaviors and sensor data.

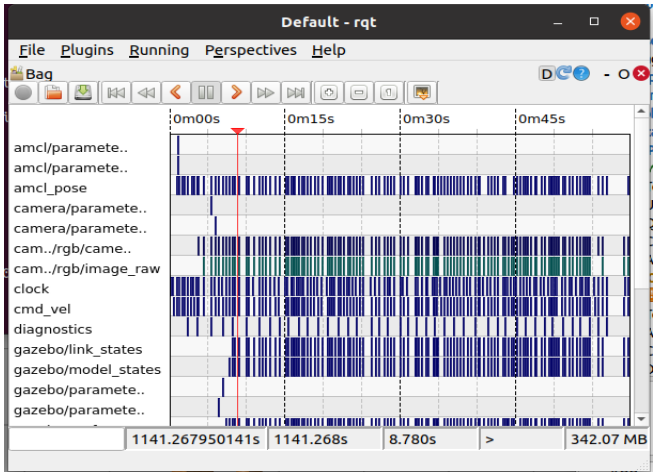


Fig. 4. An example of rqt bag.

Recording TurtleBot3 data using ROS bags, especially when incorporating SLAM, provides a comprehensive dataset that captures the robot's interactions with the environment. Additionally, it captures its self-estimated localization and mapping. This recorded data is crucial for robotic developers and researchers, facilitating in-depth analysis, algorithm development, and refinement of robot behavior.

However, several challenges were encountered. The simulation environment was resource-intensive, and limited knowledge hindered the ability to obtain extensive autonomous data. Issues with lagging and reduced performance were also faced. Moreover, we had to fine-tune Gazebo profile configurations using the command *gz physics -s* to balance performance and data quality [23]. Accurate calibration of sensors, such as cameras, and synchronization with robot movements required meticulous attention to detail. Researchers should be prepared to address these challenges to ensure the quality and reliability of recorded data.

Despite these obstacles, our research provided valuable insights into the complexities of robot data collection. It also highlighted the importance of addressing performance issues to ensure the acquisition of high-quality autonomous data. We strongly emphasize the significance of having a deeper understanding of the ROS and Gazebo ecosystem as well as optimizing the virtual environment for efficient data generation and collection.

V. Findings and Discussion

In this section, findings regarding the analysis of the retrieved artefacts and their significance in robot forensics are discussed.

A. ROS Bag Analysis

An analysis using the Autopsy tool was conducted to examine the structure and contents of a *ROS Bag file* extracted from the virtual disk (.vmdk). The .vmdk image was loaded into Autopsy and FTK Imager for forensics analysis and

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.03

verification. The findings from this analysis revealed crucial information about the internal organization and metadata of this file. The analysis primarily focused on hexadecimal lines extracted from the file which contained critical details regarding the file's structure and metadata (see Figure 5).

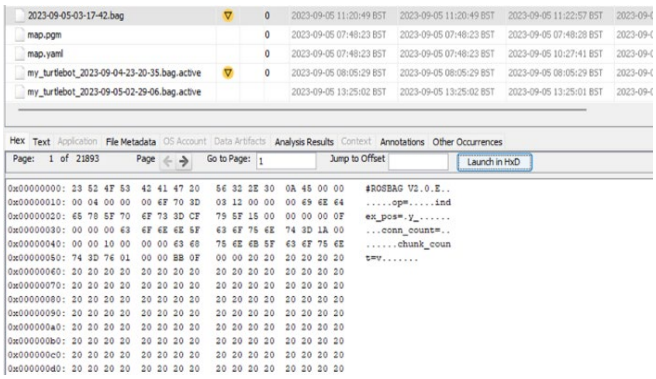


Fig. 5. ROS bag analysis.

These findings explain several key aspects. Firstly, the initial set of hexadecimal values *#ROSBAG V2.0\nE\0\0* indicate the ROS Bag file's format version, which is crucial for determining the file's compatibility with ROS tools.

Additionally, the hexadecimal lines included data fields labelled *chunk_count*, *conn_count*, *index_pos* and *op*. These fields represent various parameters or properties related to the data stored within the bag file. Further analysis identified specific field-value pairs, such as *chunk_count = 0x00000010* and *conn_count = 0x0000000F*, which provide essential metadata about the bag file, including the number of data chunks and the count of connections or channels recorded within the file.

Furthermore, two extra fields were identified: *index_pos* with a value of *0x00000004* and *op* with a hexadecimal value of *0x03BB0F00*. These fields contain crucial information about the bag file's structure or content, which are integral to the *Header Section*. The analysis of this section of the ROS Bag File structure is a critical component containing metadata about the bag file, including *format version*, *compression specifications*, and *other file-level characteristics*.

Summing up, understanding and interpreting these parameters are essential for ensuring the file's integrity and compatibility with ROS tools which are of upmost importance in robotic data analysis and forensics.

B. Significance of ROS Bag Files in Digital Forensics

The analysis of a ROS Bag file using the Autopsy tool revealed critical insights into its structure and metadata. Key findings include identifying the file's format version and the presence of essential data fields such as *chunk_count* and *conn_count*. Additional fields like *index_pos* and *op* were also discovered. These findings primarily relate to the *Header Section* of the ROS Bag File Structure, underscoring its role in storing vital metadata about this file.

On the other hand, identifying the *ROS Bag file's format version* is crucial for determining its compatibility with ROS tools, ensuring seamless data processing and analysis. Data fields such as *chunk_count* and *conn_count* offer valuable metadata about the file, including the number of data chunks and connections recorded. This provides essential context for forensic investigators in interpreting the file's content and structure. Additionally, the fields *index_pos* and *op* have critical information related to the file's internal organization, which highlights the importance of the Header Section.

These findings hold substantial implications for digital forensic investigations and cybercrime in robotics. They empower forensic experts to thoroughly analyze, validate, and interpret data captured within ROS Bag files, enhancing their ability to detect potential cyber threats, malicious activities, or unauthorized operations. Especially, *understanding the format version* ensures that ROS tools can accurately process the file, maintaining the investigation's integrity. Additionally, the metadata on data chunks and connections assists investigators in reconstructing events and contextualizing the recorded data; thereby providing a clearer understanding of the situation under scrutiny.

Another crucial aspect to consider in cybercrime involving robots is the prevalent use of plain text communication within legacy systems utilizing ROS. Such communication methods can introduce vulnerabilities since data transmitted in an unencrypted format is prone to interception and tampering. To mitigate this risk, it is essential for organizations employing ROS in legacy systems to adopt encryption and robust security measures to safeguard data both in transit and at rest. Neglecting to implement these measures may leave robotic systems vulnerable to security breaches and unauthorized disclosure.

However, while the analysis offers valuable insights into the Header Section of the ROS Bag File Structure, it is important to acknowledge its limitations. The findings focus on the file's metadata and structure and may not unveil the precise content or significance of the recorded data. Furthermore, the analysis overlooks potential encryption, compression, or security measures that could impact the interpretation of the file's content. Further investigation may be necessary to explore these aspects thoroughly. Specifically, a significant challenge when working with ROS Bag files is the limitation of tools capable of efficiently parsing and analyzing their contents. While examining the file's structure and metadata are feasible tasks, accessing and deciphering the data they contain can pose significant challenges. This limitation brings up the necessity for developing specialized tools and techniques dedicated to extracting valuable insights from ROS Bag files.

VI. Conclusion and Future Work

This research aimed to explore and demonstrate the potential of utilizing robot-generated data, such as ROS Bag data, as valuable sources of evidence in digital forensic investigations. By identifying relevant artefacts and comprehensively understanding the behavior of robotic systems, we aimed to enhance the ability to successfully resolve crime cases involving robots.

In this work, we answered the research question of how robot-generated data can be used effectively in this context. For this purpose, the methods used included simulating robot activities, collecting structured data, and conducting direct observations, which resulted in a rich dataset. Also, the detailed analysis of a ROS Bag file using the Autopsy tool uncovered valuable insights into its structure and metadata, improving the understanding of how ROS Bag files can be used in digital forensics. We were also able to identify challenges associated with parsing and analyzing ROS Bag files, emphasizing the need for specialized tools. Additionally,

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.03

we highlighted the vulnerability introduced by plain text communication within legacy ROS systems and recommended implementing encryption and security measures for data at rest and in transit. This research demonstrates the potential of robot-generated data as forensic evidence, making significant contributions to digital forensics, setting the foundation for future investigations and tool development in this emerging domain.

Finally, to further enhance the understanding of ROS Bag files in digital forensics investigations, future work could focus on refining methodologies for analyzing robot-generated data, including decrypting and decompressing the file's content, if applicable. Additionally, examining the timestamps and their synchronization with the forensic workstation's time zone could provide insights into the accuracy of temporal data within the file. Exploring the impact of encryption and security measures on data accessibility and interpretation could be also a valuable venue for further research for exploring legal and ethical aspects while conducting case studies to validate current findings in real-world scenarios. In general, any ongoing exploration of ROS Bag files and any other robot-generated data is essential to continually improve forensic practices in robotics as well as addressing the challenges of data parsing and security not only in legacy ROS systems, but also in related evolving technologies.

REFERENCES

[1] M. Javaid, A. Haleem and R. S. Pratap, "Substantial capabilities of robotics in enhancing industry 4.0 implementation," *Cognitive Robotics,* vol. 1, pp. 58-75, 2021. https://doi.org/10.1016/j.cogr.2021.06.001

[2] B. Dieber, B. Breiling and S. Taur, "Security for the Robot Operating System," *Robotics and Autonomous Systems,* vol. 98, pp. 192-203, 2017. https://doi.org/10.1016/j.robot.2017.09.017

[3] J.-P. A. Yaacoub, H. N. Noura, O. Salman and A. Chehab, "Robotics cyber security: vulnerabilities, attacks, countermeasures, and recommendations," *International Journal of Information Security,* vol. 21, pp. 115-158, 2022. https://doi.org/10.1007/s10207-021-00545-8

[4] I. Abeykoon, X. Feng and R. Qiu, "A Forensic Investigation of Robot Operating System," in *IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing (DASC/PiCom/DataCom/CyberSciTech)*, 2017. https://doi.org/10.1109/dasc-picom-datacom-cyberscitec.2017.77

[5] L. Joseph, "Mastering ROS for Robotics Programming," October 2021. [Online]. Available: https://learning.oreilly.com/library/view/mastering-ros-for/9781801071024/B17104_01_Epub_AM.xhtml#_idParaDest-29. [Accessed 15 February 2024].

[6] U. Shirode, A. Aher, P. Bale and A. M. Kadam, "A robotic framework for simulation and control of SCARA robot based on ROS," 2019. [Online]. Available: https://doi.org/10.2139/ssrn.3418758.

[7] M. Quigley, K. Conley, B. P. Gerkey and A. Y. Ng, "ResearchGate," ROS: an open-source Robot Operating System., 2009. [Online]. Available: https://www.researchgate.net/publication/233881999_ROS_an_open-source_Robot_Operating_System. [Accessed 25 April 2024]

[8] "What Is ROS?," 1 February 2023. [Online]. Available: https://roboticsbackend.com/what-is-ros/. [Accessed 15 May 2024].

[9] E. Fosch-Villaronga and T. Mahler, "Cybersecurity, safety and robots: Strengthening the link between cybersecurity and safety in the context of care robots.," *Computer Law & Security Review,* 2021. https://doi.org/10.1016/j.clsr.2021.105528

[10] *ABI Research and Data. "50,000 warehouses will be used by robots by 2025 as barriers to entry fall and AI innovation accelerates",* 2019. https://www.abiresearch.com/press/50000-warehouses-use-robots-2025-barriers-entry-fall-and-ai-innovation-accelerates/ [Accessed 25 June 2024]

[11] "Rogue Robots: Testing the limits of an industrial robot's security.," 3 May 2017. [Online]. Available: https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/rogue-robots-testing-industrial-robot-security. [Accessed 15 June 2024].

[12] A. Botta, S. Rotbei, S. Zinno and G. Ventre, "Cyber security of robots: A comprehensive survey.," *Intelligent Systems With Applications,* no. 18, 2023. https://doi.org/10.1016/j.iswa.2023.200237

[13] K. Cottrell, D. B. Bose, H. Shahriar and A. Rahman, "An Empirical Study of Vulnerabilities in Robotics.," in *IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC).,* 2021. https://doi.org/10.1109/compsac51774.2021.00105

[14] N. Goerke, D. Timmermann and I. Baumgart, "Who Controls Your Robot? An Evaluation of ROS Security Mechanisms," in *7th International Conference on Automation, Robotics and Applications (ICARA),* 2021. https://doi.org/10.1109/icara51699.2021.9376468

[15] D. Portugal, S. S. Pereira and M. S. Couceiro, "The role of security in human-robot shared environments: A case study in ROS-based surveillance robots," in *26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN),* 2017. https://doi.org/10.1109/roman.2017.8172422

[16] R. Toris, C. A. Shue and S. Chernova, "Message authentication codes for secure remote non-native client connections to ROS-enabled robots," in *IEEE International Conference on Technologies for Practical Robot Applications (TePRA).,* 2014. https://doi.org/10.1109/tepra.2014.6869141

[17] B. Dieber, R. White, S. Taurer, B. Breiling, G. Caiazza, H. I. Christensen and A. Cortesi, "Penetration Testing ROS," *In Studies in computational intelligence,* pp. 183-225, 2019. https://doi.org/10.1007/978-3-030-20190-6_8

[18] R. R. Teixeira, I. P. Maurell and P. Drews, "Security on ROS: analysing and exploiting vulnerabilities of ROS-based systems.," in *Latin American Robotics Symposium (LARS).,* 2020. https://doi.org/10.1109/lars/sbr/wre51543.2020.9307107

[19] J. R. McClean, C. J. Stull, C. R. Farrar and D. Mascareñas, "A preliminary cyber-physical security assessment of the Robot Operating System (ROS)," *Proceedings of SPIE - Defense, Security and Sensing,* 2013. https://doi.org/10.1117/12.2016189

[20] I. Abeykoon and X. Feng, "Challenges in ROS Forensics," in *IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI),* 2019. https://doi.org/10.1109/smartworld-uic-atc-scalcom-iop-sci.2019.00299

[21] V. Vilches, "Volatile memory forensics for the Robot Operating System.," arXiv., 2018. https://doi.org/10.48550/arXiv.1812.09492

[22] M. M. Basheer and A. Varol, "An overview of robot operating system forensics.," in *1st International Informatics and Software Engineering Conference (UBMYK).,* 2019. https://doi.org/10.1109/ubmyk48245.2019.8965649

[23] map_server, "ROS Wiki.," 23 March 2020. [Online]. Available: http://wiki.ros.org/map_server. [Accessed 18 August 2023].

AUTHORS

Shahrzad Zargari

Shahrzad is the principal lecturer of the cyber security and computer networks subject group lead at Sheffield Hallam University. Shahrzad has worked in the IT industry for over 15 years and gained a great deal of experience in computer hardware, software, and business management. Shahrzad's passion lies in the realm of digital forensics and security. She advocates collaboration among the government, industry, and academia. Her mission involves sharing information and nurturing the next generation of cybersecurity experts through education. Shahrzad is the director and steering committee member of the Yorkshire Cyber Security Cluster and the vice chair of BCS South Yorkshire Branch. Shahrzad is a member of the UKC3 cyber skills working group. Shahrzad is an experienced researcher (CENTRIC), having published book chapters and many papers in conferences, journals, and magazines. Additionally, she is the associate editor of the Information Security Journal: A Global Perspective at Taylor and Francis.

Tharmini Janarthanan

Tharmini, a Lecturer in Cybersecurity and Digital Forensics at Sheffield Hallam University is a dedicated Information Security Professional and a Certified ISO 27001:2022 Lead Auditor with over five years of experience. Tharmini's expertise encompasses information, technology, privacy risk management, and digital forensics. Her passion for cybersecurity drives her academic research in digital forensics, where she has undertaken numerous research projects and published papers in peer-reviewed international conferences, journals, and book chapters. In addition to her academic achievements, Tharmini is a member of CIISec and serves on the British Computing Society (BCS) committee in South Yorkshire, further demonstrating her credibility and involvement in the industry.

ANN-MoC Method for Inverse Transient Transport Problems in One-Dimensional Geometry

ARTICLE HISTORY
Received 5 March 202
Accepted 19 April 2024
Published 08 July 2024

Nelson Garcia Roman
Universidade Federal de Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
ngroman1992@gmail.com
ORCID: 0009-0006-8794-9500

Pedro Costas dos Santos
Universidade Federal de Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
pedro.costa4137@gmail.com
ORCID: 0009-0001-9927-2860

Pedro Henrique de Almeida Konzen
Universidade Federal de Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
pedro.konzen@ufrgs.br
ORCID: 0000-0002-0411-1563

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

N. Roman, P. dos Santos and P. Konzen, "ANN-MoC Method for Inverse Transient Transport Problems in One-Dimensional Geometry", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024.

ANN-MoC Method for Inverse Transient Transport Problems in One-Dimensional Geometry

Nelson Garcia Roman
Universidade Federal de Rio Grande do Sul (UFRGS)
Instituto de Matemática e Estatística (IME)
Porto Alegre, Brazil
ngroman1992@gmail.com

Pedro Costas dos Santos
Universidade Federal de Rio Grande do Sul (UFRGS)
Instituto de Matemática e Estatística (IME)
Porto Alegre, Brazil
pedro.costa4137@gmail.com

Pedro Henrique de Almeida Konzen
Universidade Federal de Rio Grande do Sul (UFRGS)
Instituto de Matemática e Estatística (IME)
Porto Alegre, Brazil
pedro.konzen@ufrgs.br

***Abstract*—**Transport problems of neutral particles have important applications in engineering and medical fields, from safety and quality protocols to optical medical procedures. In this paper, the ANN-MoC approach is proposed to solve the inverse transient transport problem of estimating the absorption coefficient from scalar flux measurements at the boundaries of the model domain. The central idea is to fit an Artificial Neural Network (ANN) using samples generated by direct solutions computed by a Method of Characteristics (MoC) solver. The direct solver validation is performed on a manufactured solution problem. Two inverse problems are then presented for testing the ANN-MoC method. In the first, a homogeneous medium is assumed, and, in the second, the medium is heterogeneous with a piecewise constant absorption coefficient. We show that the method can achieve good estimates, with accuracy depending on that of the direct solver. We also include a test of sensibility by studying the propagation of noise on the input data. The results highlight the potential of the proposed method to be applied to a broader range of inverse transport problems.

Keywords*—*artificial neural network, method of characteristics, particle neutral transport, inverse problem

I. Introduction

Neutral particle transport problems have many important applications in engineering and medical fields. The main fields of radiative heat transfer and neutron transport share the fundamental model based on the linear Boltzmann equation [1], [2]. Applications include engineering at high temperatures, such as glass and ceramic manufactures [3], combustion chambers [4], solar energy production [5], nuclear energy production [6], and optical medicine [7], [8]. Related inverse problem solutions can enhance the development of safety protocols, quality control procedures, and technological innovations.

We consider the time-dependent linear Boltzmann equation with initial and boundary conditions and with isotropic scattering

$$\forall \mu: \frac{1}{c}\frac{\partial}{\partial t} I(t,x,\mu) + \mu \frac{\partial}{\partial x} I(t,x,\mu) + \sigma_t I(t,x,\mu) = \sigma_s \Psi(t,x) + q(t,x,\mu),\ (t,x) \in \left(0, t_f\right] \times D, \tag{1.1}$$

$$\forall \mu: I(0,x,\mu) = I_0(x,\mu),\ x \in D, \tag{1.2}$$

$$\forall \mu > 0: I(t,a,\mu) = I_{in,a}(t,\mu),\ t \in \left(0, t_f\right], \tag{1.3}$$

$$\forall \mu < 0: I(t,b,\mu) = I_{in,b}(t,\mu),\ t \in \left(0, t_f\right], \tag{1.4}$$

$I(t,x,\mu)$ (W/sr) denotes the radiation intensity at time $0 \leq t \leq t_f$ (s) at point $x \in D = [a,b]$ (m), and in the direction $-1 \leq \mu \leq 1, \mu \neq 0$. The average speed of light in the medium is denoted by c (m/s). The total absorption coefficient is denoted by $\sigma_t = \kappa + \sigma_s$, while $(1/m)$ and σ_s $(1/m)$ are, respectively, the absorption and scattering coefficients. The sources are denoted by $q(t,x,\mu)$ $(W/(msr))$ in the domain and $I_{in,a}(t,\mu)$, $I_{in,b}(t,\mu)$ (W/sr) at boundaries. At $t = 0$, initial condition $I = I_0(x,\mu)$ (W/sr) is assumed. The average scalar flux (W/sr) is defined as

$$\Psi(t,x) := \frac{1}{2}\int_{-1}^{1} I(t,x,\mu) d\mu. \tag{2}$$

Inverse transport problems have been the subject of important research for many decades. The books of [9] and [10] discuss the fundamental methods applied to the solution of inverse problems. Concerning the problems of parameter estimation, the main approaches consist of estimating parameters as solutions to an associated minimization problem. The problem can then be solved by optimization methods, which usually require a good initial approximation of the solution. When this is not known, meta-heuristic algorithms can be applied to this end (see, for instance [11]). Alternatively, Deep Learning [12] techniques are also applied [13], [14]. A well-known approach is to fit an Artificial Neural Network (ANN, [15]) with samples built from solutions to the associated direct problem.

In this context, we introduce the ANN-MoC approach to the inverse transport problem of the absorption coefficient estimation from the scalar flux measured at the boundaries of the model domain. The core concept is to fit an ANN using data derived from direct solutions of Eq. (1) computed by a solver based on the Method of Characteristics (MoC) [16]. The designed methodology is here presented together with

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

selected test cases. After testing the direct solver, two inverse problems are considered. The first is a transport problem in a homogeneous medium. In the second, the medium has two regions with different absorption coefficients.

In the following, the methodology of the MoC direct solver and the ANN model are presented. Numerical experiments with the proposed approach are then presented. They include the selection of ANN architectures, data preprocessing, and model sensibility tests. Conclusions are then presented.

II. The ANN-MOC Method

The ANN-MoC approach consists of solving the inverse transport problem by an Artificial Neural Network (ANN) trained from samples generated by directly solving a set of transport problems by the Method of Characteristics (MoC).

A. MoC direct solver

The MoC direct solver computes an approximation of Eq. (1) built with the Discrete Ordinates Method (DOM) [1] followed by an implicit Euler time discretization [17]. The raised system of ordinary differential equations is decoupled by a Source Iteration (SI, [1]) scheme and then, solved with the Method of Characteristics (MoC, [15]).

Discrete ordinates formulation. The following DOM form of Eq. (1) is obtained by assuming the Gauss-Legendre quadrature $\{(\mu_j, w_j)\}_{j=1}^{n_q}$, with even $n_q > 1$,

$$\forall \mu: \frac{1}{c}\frac{\partial}{\partial t} I_j(t,x) + \mu_j \frac{\partial}{\partial x} I_j(t,x) + \sigma_t I_j(t,x) = \sigma_s \Psi(t,x) + q_j(t,x), \ (t,x) \times D, \tag{3.1}$$

$$\forall \mu: I_j(0,x) = I_{j,0}(x), \ x \in D, \tag{3.2}$$

$$\forall \mu > 0: I_j(t,a) = I_{j,in,a}, \ \forall t \in (0, t_f], \tag{3.3}$$

$$\forall \mu < 0: I_j(t,b) = I_{j,in,b}, \ \forall t \in (0, t_f], \tag{3.4}$$

where the notation $I_j(t,x) \approx I(t,x,\mu_j)$ (analogous to the others) is assumed with $j = 1,2,\dots,n_q$. The scalar flux is approximated by

$$\Psi(t,x) \approx \frac{1}{2}\sum_{j=1}^{n_q} I_j \, w_j. \tag{4}$$

Time discretization. For the time discretization, it is assumed that $t^{(k)} = kh_t, k = 0,1,2,\dots,n_t, h_t = t_f/n_t$ (see Fig. 1). The implicit Euler formulation of Eq. (3) gives an iterative procedure with initialization

$$\forall \mu_j: I_j^{(0)}(x) = I_{j,0}(x), \ x \in D, \tag{5}$$

$j = 1,2,\dots,n_q$, and the following steps

$$\forall \mu_j: \frac{1}{c}\frac{I_j^{(k+1)}(x) - I_j^{(k)}(x)}{h_t} + \mu_j \frac{\partial I_j^{(k+1)}(x)}{\partial x} + \sigma_t I_j^{(k+1)}(x) = \sigma_s \Psi^{(k+1)}(x) + q_j^{(k+1)}(x), \tag{6.1}$$

$$\forall \mu_j > 0: I_j^{(k+1)}(a) = I_{j,in,a}^{(k+1)}, \tag{6.2}$$

$$\forall \mu_j < 0: I_j^{(k+1)}(b) = I_{j,in,b}^{(k+1)}, \tag{6.3}$$

where the notation $I_j^{(k)}(x) \approx I(t^{(k)}, x, \mu_j)$ (analogous to the others) is assumed with $k = 0,1,2,\dots,n_t - 1$ and $j = 1,2,\dots,n_q$. For the sake of simplicity, in the following the index k will be suppressed, with $I_j^{(1)}$ denoting $I_j^{(k+1)}$ and $I_j^{(0)} = I_j^{(k)}$ (analogous to the others).

Source iteration. The decoupling of system Eq. (6) is performed with the Source Iteration (SI) technique. From a given initial scalar flux $\Psi^{(0,0)}(x)$, successive approximations $\Psi^{(1,l)}(x)$are iteratively computed from

$$\forall \mu_j: \frac{1}{c}\frac{I_j^{(1,l+1)}(x) - I_j^{(0)}(x)}{h_t} + \mu_j \frac{\partial I_j^{(1,l+1)}(x)}{\partial x} + \sigma_t I_j^{(1,l+1)}(x) = \sigma_s \Psi^{(1,l)}(x) + q_j^{(1)}(x), \tag{7.1}$$

$$\forall \mu_j > 0: I_j^{(1,l+1)}(a) = I_{j,in,a}^{(1,l+1)}, \tag{7.2}$$

$$\forall \mu_j < 0: I_j^{(1,l+1)}(b) = I_{j,in,b}^{(1,l+1)}, \tag{7.3}$$

where

$$\Psi^{(1,l)}(x) := \frac{1}{2}\sum_{j=1}^{n_q} I_j^{(1,l)}(x) \, w_j, \tag{8}$$

for $j = 1,2,\dots,n_q$, and $l = 0,1,2,\dots$ until some given stop criteria are fulfilled.

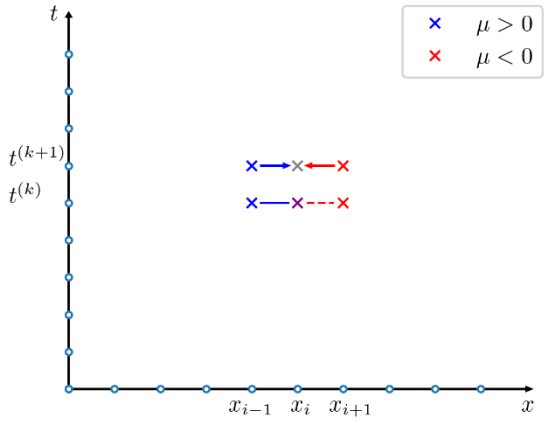


Fig. 1. Scheme of the space-time discretization. Points (×) and intervals (lines and sets) for directions μ > 0 (blue) and μ < 0 (red)

Method of characteristics. At each time step and each source iteration, Eq. (7) is solved by the Method of Characteristics (MoC). First, it is observed that Eq. (7a) can be rewritten as

$$\mu_j \frac{\partial I_j^{(1,l+1)}(x)}{\partial x} + \left(\sigma_t + \frac{1}{ch_t}\right) I_j^{(1,l+1)}(x) = \sigma_s \Psi^{(1,l)}(x) + q_j^{(1)}(x) + \frac{1}{ch_t} I_j^{(0)}(x), \tag{9}$$

$j = 1,2,\dots,n_q$ and $l = 0,1,2,\dots$. Again, for the sake of simplicity, the index j is suppressed in the following.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

The MoC form of Eq. (9) is obtained by assuming $x(s) = x_0 + s\mu, s \in \mathbb{R}$, from where Eq. (9) is rewritten as

$$\frac{d}{ds}I^{(1,l+1)}(s) + \left(\sigma_t + \frac{1}{ch_t}\right)I^{(1,l+1)}(s) = \sigma_s \Psi^{(1,l)}(s) + q^{(1)}(s) + \frac{1}{ch_t}I^{(0)}(s), \quad (10)$$

$l = 0, 1, 2, \ldots$. This linear first-order differential equation can now be solved using an integration factor, which gives the solution from

$$I^{(1,l+1)}(s) = I^{(1,l+1)}(0)e^{-\int_0^s \tilde{\sigma}_t ds'} + \int_0^s S^{(l)}(s')e^{-\int_{s'}^s \tilde{\sigma}_t ds''} ds', \quad (11)$$

where $\tilde{\sigma}_t := \sigma_t + \frac{1}{ch_t}$ and

$$S^{(l)}(s) := \sigma_s \Psi^{(1,l)}(s) + q^{(1)}(s) + \frac{1}{ch_t}I^{(0)}(s), \quad (12)$$

$l = 0, 1, 2, \ldots$.

One observes that choosing $x_0 = a$, Eq. (11) gives the particle intensity $I^{(1,l+1)}\big(x(s)\big)$ at each domain $x(s)$ for a given direction $\mu > 0$. Analogously, by choosing $x_0 = b$, one obtains the particle intensity point for a given direction $\mu < 0$.

Direct solver algorithm. Assuming a spatial mesh with n_x nodes $x_i = a + ih_x$, and mesh size $h_x = (b-a)/n_x, i = 0,1,2,\ldots,n_x$, see Fig. 1, the direct solver algorithm can be summarized as follows:

1. Set time, mesh and quadrature parameters
2. From initial condition, set

$$I_{i,j}^{(0)} \leftarrow I(0, x_i, \mu_i), \ \forall i, j, \quad (13.1)$$

$$\Psi_i^{(0)} \leftarrow \frac{1}{2}\sum_{j=1}^{n_q} I_{i,j}^{(0)} w_j, \ \forall i. \quad (13.2)$$

3. (Time loop). For $k = 0,1,2,\ldots,n_t$:

   a. (SI loop) For $l = 0,1,2,\ldots,n_{s.i.}$:

   a.1. For $j = 0,1,2,\ldots,n_q$ and $\mu_j > 0$:

   For $i = 0,1,2,\ldots,n_x - 1$:

$$I_{i+1,j}^{(1,l+1)} \leftarrow I_{i,j}^{(1,l+1)}e^{-\int_0^s \tilde{\sigma}_t ds'} + \int_0^s S^{(l)}(s')e^{-\int_{s'}^s \tilde{\sigma}_t ds''} ds'. \quad (14)$$

   a.2. For $j = 1,2,\ldots,n_q$ and $\mu_j < 0$:

   For $i = n_x, n_x - 1, \ldots, 1$:

$$I_{i-1,j}^{(1,l+1)} \leftarrow I_{i,j}^{(1,l+1)}e^{-\int_0^s \tilde{\sigma}_t ds'} + \int_0^s S^{(l)}(s')e^{-\int_{s'}^s \tilde{\sigma}_t ds''} ds'. \quad (15)$$

   a.3. Compute new scalar flux

$$\Psi_i^{(1+l)} \leftarrow \frac{1}{2}\sum_{j=1}^{n_q} I_{i,j}^{(1,l+1)} w_j, \ \forall i. \quad (16)$$

   a.4. SI stop criterion

B. *ANN inverse model*

The inverse problem is solved by fitting a Multilayer Perceptron network (MLP, [14]) from a data set $\left\{\left(\boldsymbol{\psi}_{train}^{(s)}, \boldsymbol{\kappa}_{train}^{(s)}\right)\right\}_{s=1}^{n_{train}}$ generated from computed solutions of the direct problem for several values of the absorption coefficient. The MLP of $n_h + 2$ layers is written as

$$\tilde{\boldsymbol{\kappa}} = \mathcal{N}\left(\boldsymbol{\psi}; \left\{\left(\boldsymbol{f}^{(l)}, \boldsymbol{b}^{(l)}, \boldsymbol{W}^{(l)}\right)\right\}_{l=1}^{n_h+1}\right), \quad (17)$$

where, in the l-th network layer with $n_n^{(l)}$ neuron units, $\left(\boldsymbol{f}^{(l)}, \boldsymbol{b}^{(l)}, \boldsymbol{W}^{(l)}\right)$ denotes the triple of the activation function, the bias $n_n^{(l)}$-vector, and the weights $n_n^{(l)} \times n_n^{(l+1)}$-matrix. By denoting the input $\boldsymbol{y}^{(\boldsymbol{0})} = \boldsymbol{\psi}$ of detector measurements, its forward propagation through the network layers $l = 1, 2, \ldots n_h + 1$ is given by

$$\boldsymbol{y}^{(l)} = \boldsymbol{f}^{(l)}\left(W^{(l)}\boldsymbol{y}^{(l-1)} + \boldsymbol{b}^{(l)}\right), \quad (18)$$

and the output is the estimated absorption coefficient $\tilde{\boldsymbol{\kappa}} = \boldsymbol{y}^{(\boldsymbol{n_h+1})}$ (see Fig. 2).

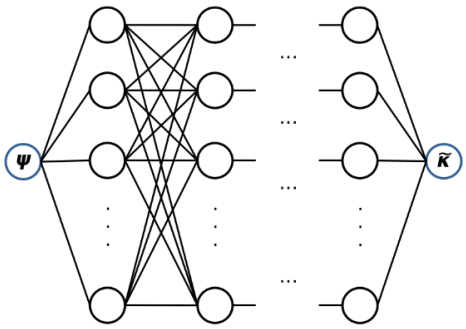


Fig. 2. Architecture of a MLP neural network with n_n neurons on each $n_h$hidden layer

Basic training algorithm. The basic training algorithm can be summarized as follows:

1. Set the MLP architecture.

Sets n_h, n_n, $\boldsymbol{f}^{(l)}$, and initial $b^{(l)}$, $W^{(l)}$ and a global learning rate $l_r > 0$.

2. Loop over epochs $e \leftarrow 1,2,\ldots,n_e$:

   2.a. Forward the training set.

$$\tilde{\boldsymbol{\kappa}}_{train} \leftarrow \mathcal{N}(\boldsymbol{\psi}_{train}). \quad (19)$$

   2.b. Compute the loss function.

$$\mathcal{L} \leftarrow \frac{1}{n_{train}}\sum_{s=1}^{n_{train}}\left|\tilde{\boldsymbol{\kappa}}_{train}^{(s)} - \boldsymbol{\kappa}_{train}^{(s)}\right|^2. \quad (20)$$

   2.c. Backward the loss function to compute the gradients

$$\frac{\partial \mathcal{L}}{\partial W^{(l)}}, \frac{\partial \mathcal{L}}{\partial b^{(l)}}, l = 1, 2, \ \ldots n_l. \quad (21)$$

   2.d. Perform an optimizer gradient based step.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

$$(W^{(l)}, \boldsymbol{b}^{(l)}) \leftarrow (W^{(l)}, \boldsymbol{b}^{(l)}) - l_r \frac{\partial \mathcal{L}}{\partial (W^{(l)}, \boldsymbol{b}^{(l)})}, \quad (22)$$

where $l = 1, 2, \ldots n_l$, and l_r is a given learning rate.

The MLPs reported in this paper have been implemented with the help of the machine learning package PyTorch [18], and trained with the Adam method [19]. The learning rate has been set to $l_r = 10^{-2}$.

ANN model test. The test of the trained neural network model consists of verifying its performance for a new data set $\left\{\left(\boldsymbol{\psi}_{test}^{(0)}, \boldsymbol{\kappa}_{test}^{(s)}\right)\right\}_{s=1}^{n_{test}}$ which has not been used for training. The test data set has also been computed by solving the direct problem for several values of the absorption coefficient. The accuracy of the network estimated values $\tilde{\boldsymbol{\kappa}}_{test}^{(s)}$ can be measured by the squared error $\mathcal{L}_{test}$ and the coefficient of determination.

C. Data preprocessing

Data preprocessing for deep learning may reduce generalization errors and reduce the size of the model needed to fit the training set [12]. There are many available techniques [20], and we have chosen to work with the preprocessing now as Standard Scaler. This function transforms the features to have zero mean and unit standard deviation. The general formula for the transformation is:

$$X_{scaled} = \frac{X - mean(X)}{std(X)}, \quad (23)$$

where X is the original value of the feature, $mean(X)$ is the mean and $std(X)$ the standard deviation over the data set X. It ensures that features have comparable scales, which is known to enhance training gradient-based methods.

III. Results

Numerical experiments with the proposed ANN-MoC approach are presented. First, the direct solver validation is presented on a manufactured solution problem. Two inverse problems are then discussed. In the first, a homogeneous medium is assumed, and, in the second, it is considered a two-region heterogeneous medium.

A. Direct solver test

In order to test the direct solver, we have considered the manufactured solution

$$\hat{I}(t, x, \mu) := e^{-\sigma_t |x-t|^2}, \; x \in (0, t_f] \times [0,1]. \quad (24)$$

By substituting Eq. (24) into Eq. (1.1), the source is found to be

$$q(t, x, \mu) = [2\sigma_t(1-\mu)(x-t) + \kappa] e^{-\sigma_t |x-t|^2}, \quad (25)$$

and from the definition of the scalar flux Eq. (2), one has $\hat{\Psi} = \hat{I}$.

After numerical tests, we have chosen the solver parameters $h_t = 0.01, n_x = n_q = 100$, and $tol = 1.49 \times 10^{-8}$ as the absolute L^2-norm tolerance for the SI stopping criterion. Table I shows a comparison between the direct solver approximations and the exact scalar flux solutions at $t_f = 1.0$ for different absorption coefficients. The relative L^2-error is denoted by ε_{rel} and indicates that the chosen parameters were enough for the direct solver to produce an accurate solution with $\varepsilon_{rel} < 10^{-2}$.

TABLE I. COMPARISON BETWEEN THE DIRECT SOLVER APPROXIMATIONS AND THE EXACT SOLUTION AT $t_f = 1.0$

κ	$\Psi(0.0)$	$\Psi(0.5)$	$\Psi(1.0)$	ε_{rel}
0.9	$3.667e-1$	$7.748e-1$	$9.974e-1$	$4.5e-3$
0.5	$3.664e-1$	$7.740e-1$	$9.971e-1$	$5.3e-3$
0.1	$3.660e-1$	$7.730e-1$	$9.968e-1$	$6.4e-3$
Exact	$3.679e-1$	$7.788e-1$	$1.000e+0$	--x--

B. Inverse problem 1 – homogeneous medium

In the inverse problem 1, we assume a homogeneous medium with a constant absorption coefficient. The problem consists of estimating $0.1 < \kappa < 0.9$ from detectors measurements of the scalar fluxes at $x_{d,0} = 0, x_{d,1} = 1$ and at time $t_{d,3} = 3.0$. Boundary conditions are taken as $I(t, 0, \mu) = 1$, for all $\mu > 0$, and $I(t, 1, \mu) = 0$, for all $\mu < 0$. The source is considered null, and the initial condition is $I(0, 0, \mu) = 1, \mu > 0$, and $I(0, x, \mu) = 0$ for all $x > 0$.

The ANN inverse model has the detectors' measurements $d_0 = \Psi(t_{d,3}, 0)$, $d_1 = \Psi(t_{d,3}, 1)$ as inputs and outputs the estimated absorption coefficient $\tilde{\kappa}$. For its training, we have used the direct solver to build a training set $\left\{\left(\boldsymbol{d}_{train}^{(s)}, \kappa_{train}^{(s)}\right)\right\}_{s=1}^{n_{train}}$ of $n_{train} = 17$ samples (patterns) with $\kappa^{(s)} = 0.1 + (s-1)h_s$, $h_s = 0.05$. The test set $\left\{\left(\boldsymbol{d}_{test}^{(s)}, \kappa_{test}^{(s)}\right)\right\}_{s=1}^{n_{test}}$ has been generated with $n_{test} = 32$ with uniformly distributed random choices $0.1 < \kappa^{(s)} < 0.9$ (see Fig. 3).

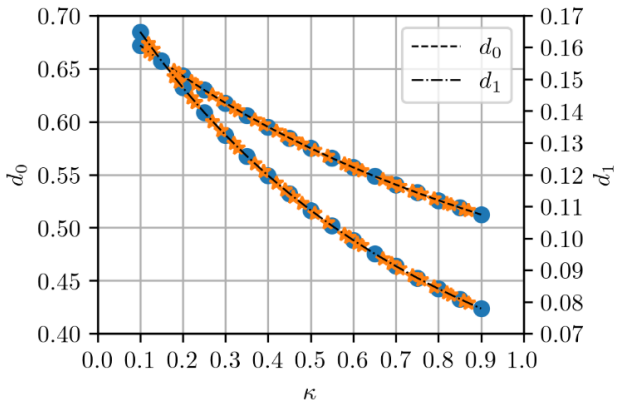


Fig. 3. Inverse problem 1. Training (*circles*) and test (*stars*) samples

We have performed several numerical tests to choose an adequate MLP architecture. Here, we tried architectures $2 - n_n \times n_h - 1$ (2 inputs, n_n neurons on each n_h hidden layer, and 1 output). Training has been stopped when the loss function $\mathcal{L} < 10^{-5}$. Due to the stochasticity of the training method, each test has been repeated three times. Table II presents the results with the hyperbolic tangent ($tanh$) and the identity as activation functions in the hidden and in the output layers, respectively. The demanded averaged total

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

number of epochs n_e and computational time t_c are tabulated. Table III presents results for similar numerical test, but with the $ReLU$ as activation function in hidden layers. We observe that, if MLP, with $tanh$ have demanded last resources to train with small architectures, the $ReLU$ in an $2-30\times4-1$ MLP was even better.

TABLE II. INVERSE PROBLEM 1. TRAINING TESTS FOR MLP ARCHITECTURES WITH TANH AS ACTIVATION FUNCTION

n_h/n_n	10	15	20	25	30
1	3807/ 5,52 s	3281/ 3.66 s	3160/ 3.66s	4711/ 6.55s	9070/ 5.01 s
2	986/ 2.3 s	818/ 1.68 s	920/ 1.27 s	764/ 1.40 s	895/ 1.24 s
3	1294/ 0.95 s	628/ 1.05 s	737/ 2.03 s	885/ 1.01 s	710/ 0.77 s
4	1603/ 1.41 s	1500/ 3.18 s	1041/ 1.94 s	1696/ 2.51 s	634/ 0.76 s

To enhance the training, we have then performed trials with data preprocessing. Inputs of the training samples have been scaled with the Standard Scaler. Setting the $ReLU$ as activation function in hidden layers, several MLP architectures have been tested, and the results can be found in Table IV. The enhancement with preprocessing is notable, with the $2-30\times4-1$ providing the best results.

TABLE III. INVERSE PROBLEM 1. TRAINING TESTS FOR MLP ARCHITECTURES WITH RELU AS ACTIVATION FUNCTION

n_h/n_n	10	15	20	25	30
1	15670/ 21.09 s	11368/ 12.83 s	11884/ 10.9 s	12697/ 15.40 s	16679/ 9.11 s
2	8040/ 4.64 s	14567/ 8.5 s	956/ 0.56 s	2398/ 1.46 s	1577/ 0.99 s
3	2140/ 1.42 s	521/ 0.27 ss	859/ 0.55 s	698/ 0.45 s	226/ 0.15 s
4	1992/ 1.37 s	1015/ 0.83 s	912/ 0.64 s	396/ 0.28 s	196/ 0.24 s

TABLE IV. INVERSE PROBLEM 1. TRAINING TESTS OF MLP ARCHITECTURES WITH DATA PREPROCESSING

n_h/n_n	10	15	20	25	30
1	3280/ 1.48 s	1842/ 0.85 s	374/ 0.18 s	237/ 0.13 s	230/ 0.1 s
2	647/ 0.38 s	238/ 0.2 s	333/ 0.21 s	170/ 0.1 s	93/ 0.06 s
3	553/ 0.34 s	230/ 0.14 s	214/ 0.14 s	145/ 0.1 s	113/ 0.07 s
4	507/ 0.33 s	227/ 0.13 s	212/ 0.14 s	174/ 0.13 s	83/ 0.06 s

Following the previous numerical tests, we have chosen to work with an $2-30\times4-1$ MLP model (two inputs, four hidden layers with 30 neurons each, and one output neuron), the $ReLU$ and the identity as activation functions in the hidden and in the output layers, respectively. With approximately $n_e = 83$, the model reaches a mean squared error $\mathcal{L}_{train} < 10^{-5}$ and coefficient of determination $R^2_{train} = 0.9998$. The application of the trained model to the test data gave results with $\mathcal{L}_{test} < 10^{-5}$ and $R^2_{test} = 0.9998$ (see Fig. 4).

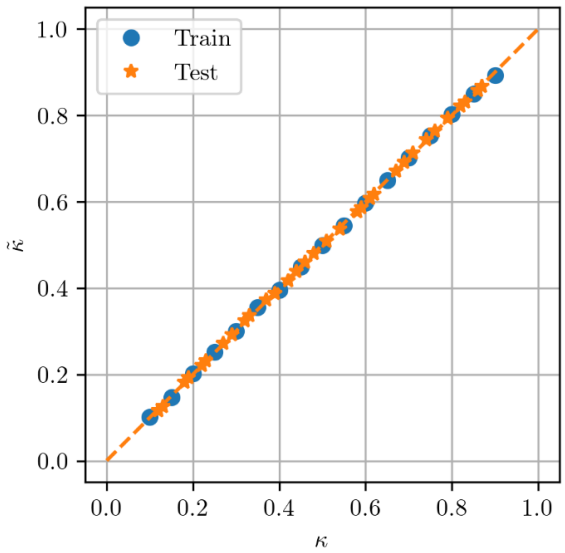


Fig. 4. Inverse problem 1. Expected κ versus estimated $\tilde{\kappa}$. Train: circles. Test: starts. Line fitted to test data results: dashed line

C. *Inverse problem 2 – heterogeneous medium*

In the inverse problem 2, we assume a heterogeneous medium with piecewise constant absorption coefficients

$$\kappa(x) = \begin{cases} \kappa_1\ , & 0 \le x \le 0.5, \\ \kappa_2\ , & 0.5 < x\ \le 1. \end{cases} \tag{26}$$

The inverse problem consists of estimating $0.1 \le \kappa_1, \kappa_2 \le 0.9$ from detectors measurements of the scalar fluxes at $x_{d,0} = 0$, $x_{d,1} = 1$ and at the times $t_{d,2} = 2.0$ and $t_{d,3} = 3.0$. The initial and boundary conditions, as well as the source, are the same as for inverse problem 1.

The ANN inverse model has the detector measurements $\boldsymbol{d}_0 = \big(\Psi(t_{d,2},0),\Psi(t_{d,3},0)\big)$, $\boldsymbol{d}_1 = \big(\Psi(t_{d,2},1),\Psi(t_{d,3},1)\big)$ as inputs and outputs the estimated absorption coefficients $\tilde{\kappa}_1$ and $\tilde{\kappa}_2$. For its training, we have used the direct solver to compute the training set $\left\{\left(\boldsymbol{d}^{(s)}_{train}, \boldsymbol{\kappa}^{(s)}_{train}\right)\right\}_{s=1}^{n_{train}}$ of $n_{train} = 81$ samples (patterns) with $\kappa^{(s)}_{1,2} = 0.1 + (s-1)h_s$, $h_s = 0.1$. The test set $\left\{\left(\boldsymbol{d}^{(s)}_{test}, \boldsymbol{\kappa}^{(s)}_{test}\right)\right\}_{s=1}^{n_{test}}$ has been generated with $n_{test} = 64$ uniformly distributed random choices $0.1 < \kappa^{(s)}_{1,2} < 0.9$.

For this inverse problem, we tested MLP architectures $4-n_n\times n_h-2$ (4 inputs, n_n neurons in each hidden layer n_h, and 2 outputs) with Standard Scaler preprocessing the input data. The training was stopped when the loss function $\mathcal{L} < 10^{-5}$. Due to the stochasticity of the training method, each test has been repeated three times. Table V presents the results with the $ReLU$ and the identity as activation functions in the hidden and in the output layers, respectively. It is tabulated the required average total number of epochs n_e and computational time t_c. Like the inverse problem 1, the MLP architecture $4-30\times4-2$ provided the best results, which we now set to report the results to follow.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

TABLE V. INVERSE PROBLEM 2. TRAINING TESTS FOR MLP ARCHITECTURES WITH DATA PREPROCESSING

n_h/n_n	10	15	20	25	30
1	7516/ 8.60 s	2662/ 3.86 s	2628/ 3.93 s	1480/ 1.79 s	1292/ 1.83 s
2	1753/ 3.30 s	1336/ 2.29 s	1062/ 1.47 s	467/ 0.85 s	448/ 0.54 s
3	3581/ 6.76 s	1177/ 2.06 s	752/ 1.72 s	513/ 1.07 s	417/ 0.49 s
4	3434/ 7.05 s	1748/ 3.81 s	565/ 0.99 s	554/ 0.92 s	266/ 0.40 s

With approximately $n_e = 266$, the model reaches a mean squared error $\mathcal{L}_{train} < 10^{-5}$ and coefficient of determination $R^2_{train} = 0.9998$. The application of the trained model to the test data gave results with $\mathcal{L}_{test} < 10^{-5}$ and $R^2_{test} = 0.9998$. Figs. 5 and 6 show the expected versus estimated absorption coefficients for the training and test samples. The fitted least square line is also shown for the test data.

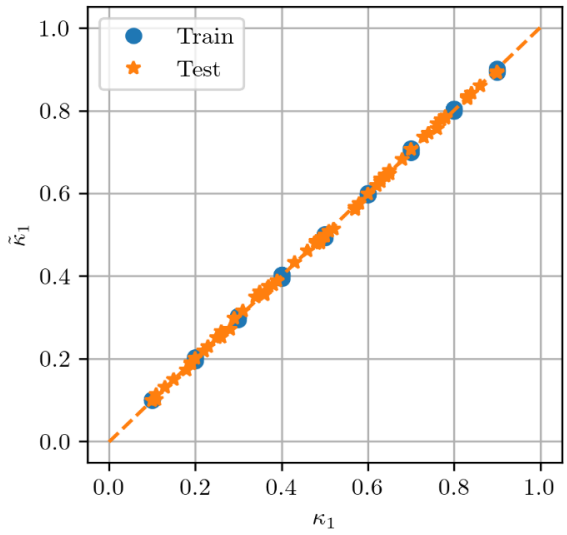


Fig. 5. Inverse problem 2. Expected κ versus estimated $\tilde{\kappa}_1$. Train: circles. Test: starts. Line fitted to test data results: dashed line

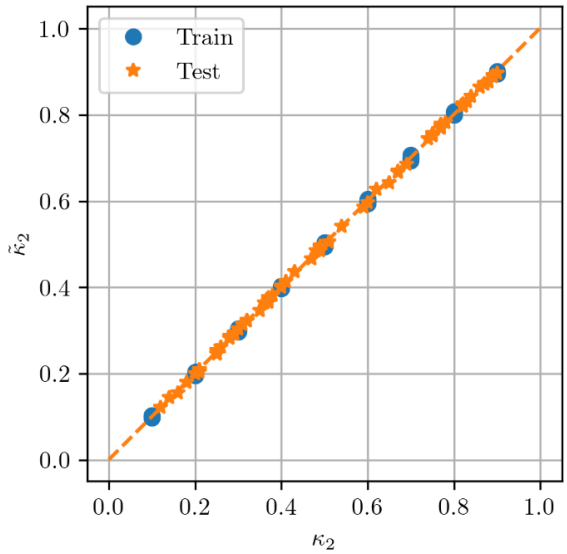


Fig. 6. Inverse problem 2. Expected κ versus estimated $\tilde{\kappa}_2$. Train: circles. Test: starts. Line fitted to test data results: dashed line

Sensitivity test. To validate the robustness and stability of the proposed MLP model in this problem, a sensitivity test was applied, which involves adding uniformly distributed noise into the input data, more specifically in detectors $\boldsymbol{d}_0$ and $\boldsymbol{d}_1$. Table VI shows the results of the mean squared error R^2 and the mean absolute squared error ($MAPE$) for different levels of noise.

The results indicate that the MLP model is relatively robust to low and moderate levels of noise in the input data. The noise is propagated to the output by a factor of 3.4 times. The $R^2 > 0.85$ is reached even with a noise level up to 5%. Figs. 7 and 8 show the expected *versus* estimated κ_1 and κ_2 of the test data set with noise levels of 2%, 3% and 4%. In the figures, the identify line is plotted as a dashed line as a guide. We observe the absence of outliers, which also indicates a good generalization of the ANN-MoC method.

TABLE VI. INVERSE PROBLEM 2. SENSITIVITY TESTS

Noise (%)	R^2	$MAPE$ (%)
1	0.994	3.84
2	0.981	7.10
3	0.948	11.06
4	0.929	13.27
5	0.873	16.95
6	0.809	20.43
7	0.726	25.26
8	0.758	23.15
9	0.483	35.52
10	0.658	33.31

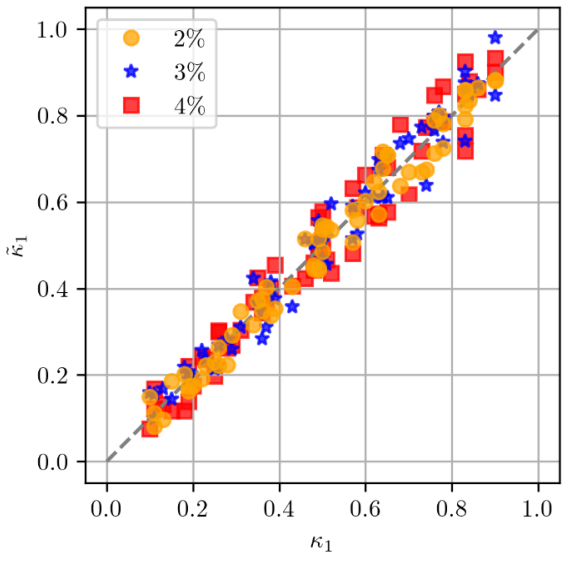


Fig. 7. Sensibility test for inverse problem 2. Comparison between expected κ_1 *versus* estimated $\tilde{\kappa}_1$ for different levels of input data noise

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.04

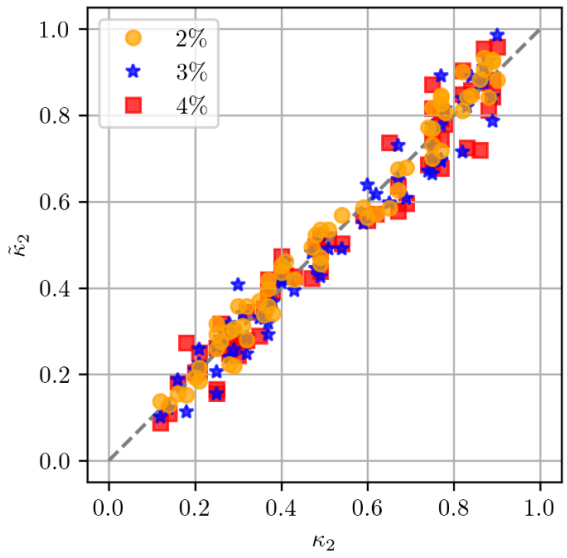


Fig. 8. Sensibility test for inverse problem 2. Comparison between expected κ_2 *versus* estimated $\tilde{\kappa}_2$ for different levels of input data noise

IV. Conclusions

In this paper, the ANN-MoC approach has been proposed to solve the inverse transient transport problem of estimating the absorption coefficient from scalar flux measurements at the boundaries of the model domain. The central idea is to fit an Artificial Neural Network (ANN) using samples generated by direct solutions computed by a Method of Characteristics (MoC) solver.

Applications of two different inverse transport problems were reported, one with homogenous medium and the other two region medium with piecewise constant absorption coefficient. After several numerical tests, we found that small MLPs could provide good estimations. Better results were reached by preprocessing the input data with the Standard Scaler. A sensitivity test was also reported for the second problem. The results highlight the potential of the proposed method to be applied to a broader range of inverse transport problems.

Further developments should aim to improve the direct solver. Improvements in the solution accuracy and, primarily, in computational performance are important to provide the ANN model with a higher-quality dataset. Solutions to more complex inverse transport problems could also benefit from the proposed approach, but once again, it will require additional improvements in the direct solver. Finally, the use of the proposed methodology for realistic problems depends on how good the direct transport model is for the intended application.

Acknowledgment

The authors thank the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) for partially financing this research (Finance Code 001).

References

[1] M. F. Modest, *Radiative Heat Transfer*, 3rd. New York: Elsevier, 2013.

[2] E. E. Lewis and W. F. Miller, *Computational Methods of Neutron Transport*. New York: John Wiley & Sons, Inc., 1984.

[3] E. W. Larsen, G. Thömmes, A. Klar, M. Seaïd, and T. Götz, "Simplified P_N approximations to the equations of radiative heat transfer and applications," *Journal of Computational Physics*, vol. 183, pp. 652–675, 2002.

[4] M. Frank, M. Seaïd, A. Klar, R. Pinnau, G. Thömmes, and J. Janicka, "A comparison of approximate models for radiation in gas turbines," *Progress in Computational Fluid Dynamics, an International Journal*, vol. 4, pp. 191–197, 2004.

[5] W. Fuqiang, M. Lanxin, C. Ziming, T. Jianyu, H. Xing, and L. Linhua, "Radiative heat transfer in solar thermochemical particle reactor: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 73, pp. 935–949, 2017.

[6] W. M. Stacey, *Nuclear Reactor Physics*, 2nd ed. Weinheim: Wiley-VCH, 2007.

[7] L. V. Wang and H. Wu, *Biomedical Optics: Principles and Imaging*. John Wiley & Sons, 2007.

[8] G. Bal, "Inverse transport theory and applications," *Inverse Problems*, vol. 25, p. 053001, 2009.

[9] M. N. Özisik and H. R. B. Orlande, *Inverse Heat Transfer. Fundamentals and Applications*, 2nd ed. CRC press, 2021.

[10] F. D. Moura Neto and A. J. Silva Neto, *An Introduction to Inverse Problems with Applications*, 1st ed. Heidelberg: Springer, 2014.

[11] F. S. Lobato, Jr. V. Steffen, and A. J. Silva Neto, "A comparative study of the application of differential evolution and simulated annealing in radiative transfer problems," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 32, pp. 518–526, 2010.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. London: MIT Press, 2016.

[13] J. C. Bokar, "The estimation of spatially varying albedo and optical thickness in a radiating slab using artificial neural networks," *International Communications in Heat and Mass Transfer*, vol. 26, pp. 359–367, 1999.

[14] Jr. J. Lugon, A. J. Silva Neto, and C. C. Santana, "A hybrid approach with artificial neural networks, Levenberg–Marquardt and simulated annealing methods for the solution of gas–liquid adsorption inverse problems," *Inverse Problems in Science and Engineering*, vol. 17, pp. 85–96, 2009.

[15] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. New Jersey: Pearson, 2009.

[16] L. C. Evans, *Partial Differential Equations*, 1st ed. Providence: AMS (American Mathematical Society), 1997.

[17] J. Stoer, R. Bulirsch, R. Bartels, W. Gautschi, and C. Witzgall, *Introduction to Numerical Analysis*, 3rd ed. New York: Springer, 1980.

[18] PyTorch Developers, "PyTorch." [Online]. Available: https://pytorch.org/

[19] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[20] A. Zheng and A. Casari, *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*, 1st ed. O'Reilly Media, Inc.

LATIN-AMERICAN JOURNAL OF COMPUTING (LAJC), Vol XI, Issue 2, July 2024

AUTHORS

Nelson Garcia Roman

Nelson García Román was born on September 20, 1992, in Pinar del Río, Cuba. He graduated as a Mechanical Engineer from the University of Pinar del Rio (2011-2016). During his studies, he was involved as a student assistant in Calculus and received two Honorable Mention awards for the presentation of two papers on Applied Mathematics in Engineering at the Scientific Conferences. After graduation, he became a professor at the same university from 2016 to 2018, and subsequently as a mathematics professor at the José Antonio Echeverría Technological University (CUJAE) until 2022. Currently, he is pursuing a Master's degree in the Graduate Program in Applied Mathematics (PPGMAp) at the Federal University of Rio Grande do Sul (UFRGS), where he holds a scholarship from CAPES, focusing on numerical methods, computational modeling, and deep learning for the numerical solution of particle neutral transport inverse problems.

Pedro Costas dos Santos

Pedro Costa dos Santos was born on September 23, 1998, in Rio de Janeiro - RJ, Brazil. During his secondary education, he successively received Honorable Mentions in the Brazilian Public School Mathematics Olympiad (OBMEP), and in 2015, he was awarded the Silver Medal in the competition. From 2016 to 2018, he attended an undergraduate course in Industrial Chemistry at the Federal University of Rio Grande do Sul(UFRGS). Since 2018, he has been a student in the undergraduate course of Applied Mathematics at UFRGS. Since 2022, he has been granted a scholarship for research initiation in Applied Mathematics, with an aim on Deep Learning applications to the numerical solution of Inverse Particle Neutral Transport problems. In 2023, he received the Honorable Mention for his research developments presented in the Scientific Initiation Week (SIC) at UFRGS.

Pedro Henrique de Almeida Konzen

Pedro Henrique de Almeida Konzen was born on June 12, 1981, in Santa Cruz do Sul - RS, Brazil. Doctor in Applied Mathematics from the Federal University of Rio Grande do Sul (UFRGS, 2010), having conducted doctoral research at Ruprecht-Karls-Universität Heidelberg/Germany (Uni-HD, 2008-2010). Currently, Adjunct Professor at the Department of Pure and Applied Mathematics (DMPA), Institute of Mathematics and Statistics (IME), Federal University of Rio Grande do Sul (UFRGS, since 2014). Permanent member of the Graduate Program in Applied Mathematics (PPGMAp-UFRGS, since 2022). Has experience in the field of applied mathematics, with emphasis on numerical methods, computational simulation, mathematical modeling and deep learning.

A study on the impact of data balance on rainfall prediction through artificial neural networks using surface microwave radiometers

ARTICLE HISTORY
Received 14 February 2024
Accepted 19 April 2024
Published 08 July 2024

Lourenço José Cavalcante Neto
Postgraduate Program in Applied Computing National Institute for Space Research
São José dos Campos, Brazil
lourenco.cavalcante@ifto.edu.br
ORCID: 0000-0001-9915-7726

Alan James Peixoto Calheiros
Postgraduate Program in Applied Computing National Institute for Space Research
São José dos Campos, Brazil
alan.calheiros@inpe.br
ORCID: 0000-0002-6408-3410

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

L. Cavalcante, A. Calheiros, "A study on the impact of data balance on rainfall prediction through artificial neural networks using surface microwave radiometers", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024

A study on the impact of data balance on rainfall prediction through artificial neural networks using surface microwave radiometers

Lourenço José Cavalcante Neto
National Institute for Space Research
Postgraduate Program in Applied Computing
São José dos Campos, Brazil
lourenco.cavalcante@ifto.edu.br

Alan James Peixoto Calheiros
National Institute for Space Research
Postgraduate Program in Applied Computing
São José dos Campos, Brazil
alan.calheiros@inpe.br

***Abstract*—**The National Institute for Space Research (INPE) has been a partner in significant projects that conduct atmospheric investigations impacting various sectors, such as the Amazon Tall Tower Observatory (ATTO) project. Since 2009, the project has conducted studies on the interactions between climate and the Amazon forest. ATTO has played an essential role in providing large volumes of data obtained by meteorological sensors, contributing to a deeper understanding of the atmospheric dynamics of the region. In a landscape where Artificial Intelligence-based rainfall forecast models gain prominence, this study explores the imbalance of data from the ATTO Campina field experiment and its influence on short-term rainfall forecasts using Artificial Neural Networks (ANNs). Metrics such as MAE, RMSE, and POD, as well as FAR indices, were applied in the assessment and revealed the connection between data balance and forecast results. More balanced data or data with greater weights for different rainfall ranges yield better results. The study emphasizes the importance of reliable data for training rain forecast models, aiming to improve the dexterity of these models. This approach is fundamental to increase the reliability of these models in real environments.

Keywords*—*Rainfall prediction, Data balancing, Machine learning, Amazon, ATTO Campina

I. Introduction

The Amazon region is home to the world's largest tropical forest and it has an equatorial and tropical climate. It is a complex and unique environment for cloud and precipitation research [1], and one of the few continental areas where primitive atmospheric conditions can still be observed [2]. The Amazon Tall Tower Observatory (ATTO) project, located approximately 150 km north of Manaus and in partnership with INPE, is an international collaboration that has focused on the interactions between climate and the Amazon rainforest since 2009. At the site, measurements of various micrometeorological and atmospheric chemical variables are conducted, covering elements such as temperature, wind, precipitation, water and energy fluxes, turbulence, soil temperature, heat fluxes, radiation, and visibility [3]. This project substantially contributes to the understanding of atmospheric processes and their global impact [2].

These advances in data collection are particularly relevant in the context of developments in Artificial Intelligence (AI) driving prediction models, including those based on Machine Learning (ML) and atmospheric data. However, measures are needed to enhance the reliability and accuracy of research results benefiting from this data. The application of AI techniques in predictions has been widely explored, as evidenced by research such as [4], [5], [6]. In 1990, during the 16th Conference on Local Severe Storms, [4] presented a study on the use of AI in storm prediction. This study stimulated the development of new research in the field, especially with the use of Artificial Neural Networks (ANNs). ANNs are systems inspired by the brain´s ability to perform calculations in parallel and distributed, enabling the accomplishment of complex tasks such as pattern recognition [7]. [8] highlighted ANNs as promising for predicting rainfall, emphasizing the need for reliable data for the effectiveness of these models.

In the literature, there are several studies that have also taken a similar approach to estimate surface rainfall. For example, [9] developed a "nowcasting" technique to analyze intense convective activities in the southeast of India, using microwave radiometers. The term "nowcasting" refers to very short-term weather forecasts, which can range from minutes to six hours. Furthermore, [10] integrated data from meteorological sensors, such as temperature, humidity, water vapor, and droplet size, into rainfall prediction models using Machine Learning (ML), representing AI approaches for weather event forecasting. In contrast, [11] developed a short-term rainfall prediction model using radiometric measurements, atmospheric parameters, and water content. However, its application in other regions is limited due to uncertainties in input data and observed results.

In this study, a simple Multilayer Perceptron (MLP) type Neural Network was configured and trained to predict short-term rainfall - one hour ahead ($t + 1$), where t represents the current moment before the event is observed on the surface 1 hour later. The model was fed with data from four meteorological instruments (rain gauge, two disdrometers and radiometer) dataset.

It is important to emphasize that the focus of this study is not on evaluating the model but on the impact that data imbalance can have on its proficiency. Metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE),

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

and additional parameters like Probability of Detection (POD) and False Alarm Rate (FAR) were used for evaluation, highlighting the relationship between observations and predictions.

II. Materials And Proposed Methodology

The research was conducted in five stages: data acquisition and preprocessing, exploratory data analysis (EDA) and variable selection, training of the neural network model, evaluation and testing, and analysis of results. Data were collected by four meteorological sensors: a Joss-Waldvogel impact disdrometer (RD-80), a PARticle SIze and VELocity laser disdrometer (PARSIVEL2), an automatic weighing-bucket rain gauge, and a ground-based microwave radiometer (MWR) model MP3000A.

These sensors are installed at the ATTO-Campina field experiment (2°10'53.7"S/59°01'18.7" W), located approximately 4 km northwest of the ATTO research site (2°08'38"S/58°59'59"W), as illustrated in Fig. 1. More detailed information about ATTO can be find at [2].

The RD-80, based on the principle established by [12], measures the size distribution of raindrops through the force applied on a transducer. According to [13], this equipment provides estimates of the Drop Size Distribution (DSD) considering relationships between size, velocity, and shape of the drops, which is essential for calculating the rainfall rate and parameters related to precipitation microphysics.

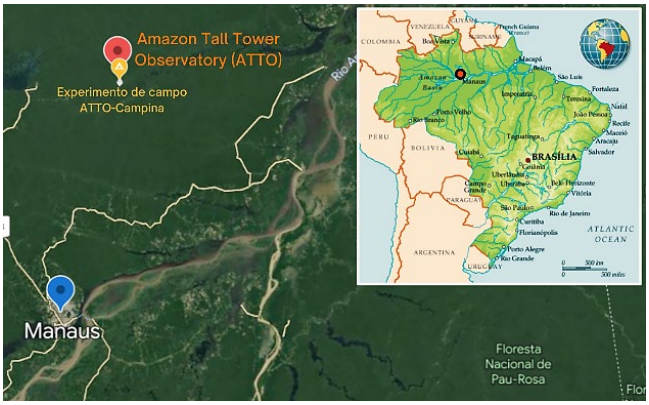


Fig. 1. Location of the Amazon Tall Tower Observatory (ATTO) project and the ATTO-Campina field experiment

In turn, the PARSIVEL is widely used in precipitation studies due to its ability to provide detailed and accurate information on drop sizes and velocities. This instrument measures the drops by interrupting the beam of a horizontally projected laser by the disdrometer [14]. Additionally, rain gauges are essential instruments for measuring the amount of precipitation in a given location, with their records often employed in the calibration and verification of remote rain sensors and weather radars [15], [16], [17].

According to [18], the MWR is an instrument used to measure radiance in the microwave spectrum and thereby estimate some atmospheric parameters such as temperature, humidity, and water vapor. It consists of two radiofrequency subsystems that use brightness temperature observations in channels between 51 and 59 GHz (V band) and between 22 and 30 GHz (K band) to estimate atmospheric vertical profiles of temperature and water, as well as integrated vapor and liquid water contents in the atmosphere up to 10 km vertically.

In the context of this study, near-surface precipitation data originate from the rain gauge and the two disdrometers, while observations of brightness temperature in different channels of the microwave spectrum are from an MWR. It is worth noting that, in this investigation, atmospheric parameters derived from the K band of the MWR were used.

The data were accessed and acquired through a public FTP server provided by the University of São Paulo (USP). After data acquisition, the data underwent a preprocessing and integration process, resulting in a single dataset. The application of the integration technique was crucial since the data were of meteorological nature and collected by different instruments. Given the nature of the time series, it was necessary to ensure the alignment of these data, thus facilitating the necessary analyses.

Next, an Exploratory Data Analysis (EDA) was conducted to investigate and select the best variables to be taken as input in the neural network training process, based on correlation analysis. A correlation matrix was constructed during the EDA, revealing relevant correlations between MWR data attributes and information about the occurrence of rainfall observed by other instruments. In other words, several MWR attributes were well correlated with the target variable, in this case, rainfall events.

To validate these observations, a case study was conducted with data collected on February 18, 2022. During this study, a significant response of brightness temperature at the 22.234 GHz channel, recorded by the MWR, was observed in relation to water accumulation in the clouds prior to rainfall.

Fig. 2 graphically depicts these data, with the red line representing the time series of brightness temperature (TB) in (K) at the 22.234 GHz channel of the MWR, while the blue line represents the precipitation rate (mm/h) recorded by precipitation sensors. Following these observations, as mentioned earlier, in the exploratory data analysis (EDA) phase, the data were integrated for better handling.

After integrating the data, as illustrated in Fig. 3, a direct analysis of the relationship between changes in brightness temperature and rainfall occurrence becomes evident. Notably, in the highlighted yellow region of the graph, there is a notable increase in temperature (red line) hours before rainfall events (blue line), which recorded a local precipitation rate exceeding 50 millimeters per hour (mm/h). Similar phenomena were also observed on other investigated rainy days. Additionally, other attributes measured by the MWR showed significant correlations with rainfall occurrence; however, only those mentioned earlier were used as input data for model training.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

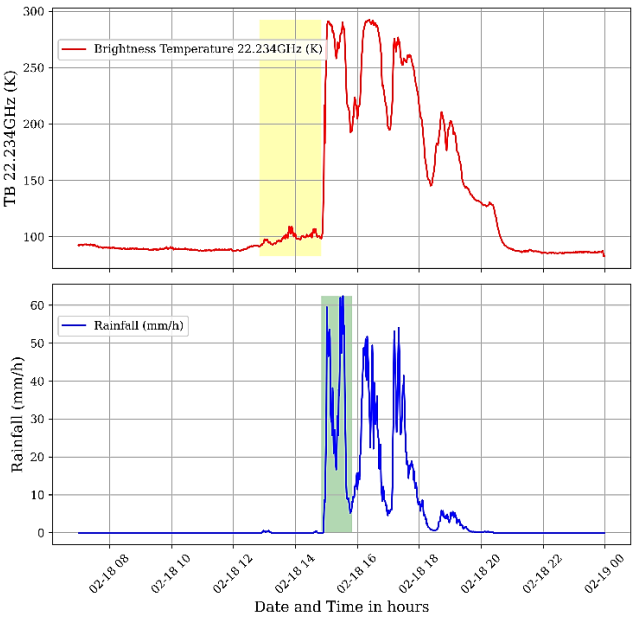


Fig. 2. Variation in brightness temperature (TB) in the 22.234 GHz channel (K) of the MWR and rainfall rate (mm/h) recorded by precipitation instruments on February 18, 2022

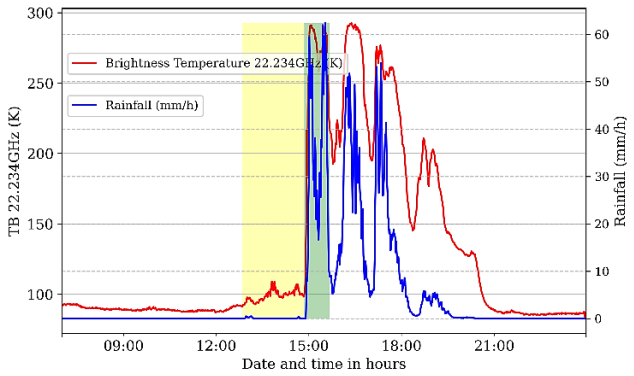


Fig. 3. Brightness temperature (TB) (K) variation in the 22 GHz channel of the MWR alongside rainfall rate (mm/h) recorded by precipitation instruments on February 18, 2022, with data co-location

As highlighted by [19], the use of radiometric brightness temperature as an observed parameter is an additional advantage since rainfall initiation is directly related to the presence of saturated water vapor and liquid water in the atmosphere, which is reflected in the increase in brightness temperature at frequencies 23 and 30 GHz [20].

Therefore, harnessing the MWR's ability to measure radiance in the microwave spectrum, particularly through brightness temperature observations in the K band, proved to be fundamental for conducting the investigations.

III. Investigated Scenarios

The investigations addressed the imbalance in the samples, emphasizing that the presence of this imbalance in the data can lead to biased estimates for certain types of rainfall. For example, there may be a tendency to favor weaker precipitation events over heavier rainfall, especially considering that heavy precipitation events are less frequent. Fig. 4 displays the unbalanced distribution of rainfall data in the data used, illustrating the disparity in the frequency of different rainfall intensities.

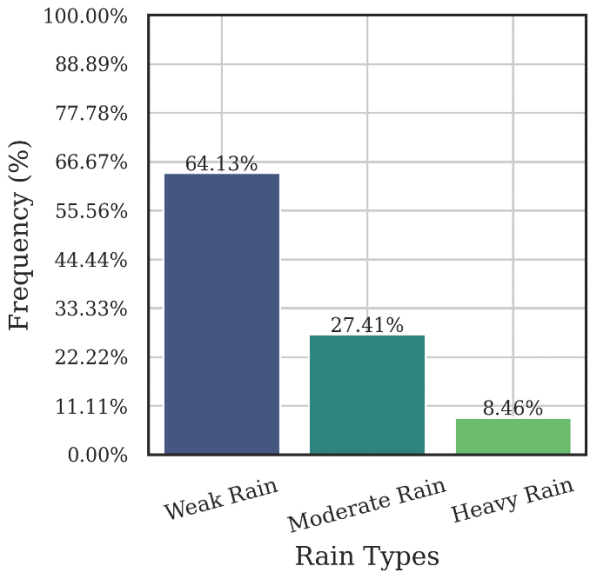


Fig. 4. Analysis of the distribution of rainfall intensity in the precipitation data used

This imbalance in the data can arise from various sources, including differences in the geographical distribution of weather events, seasonal variations in the frequency of different types of rainfall, and even limitations in data collection methods. Therefore, understanding and properly addressing this imbalance is essential to ensure accurate analyses and meaningful insights into weather patterns and rainfall events. In this context, the model was trained, evaluated, and tested in three different scenarios, as detailed below:

A. *All occurrences of rain*

Dataset containing all instances of observed systems with rainfall rates equal to or greater than 0.1 mm/h (minimum disdrometer rainfall detection) during the observed period. This approach allowed evaluating the model ability to make predictions in a range of scenarios with different rainfall intensities.

B. *Imbalance by rainfall intensity*

Dataset of defined rainfall rates, ranging from 0.1 to 50 mm/h, distributed as 64.13% for weak rain, 27.41% for moderate rain, and 8.46% for heavy rain. This test allowed us to assess the direct impact of data imbalance on the model predictions, with known data imbalances.

C. *Application of adjustments to the weights of less representative samples*

The same data from Scenario B were used; however, weights were applied to each sample point in the model training process. Higher weights were applied to less representative samples, following an approach similar to the technique proposed by [21]. This weight adjustment aimed to mitigate the impact of imbalance and investigate how considering rainfall intensity during training influences predictions.

The intensity scale of precipitation is provided in Table I for reference.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

TABLE I. INTENSITY SCALE OF PRECIPITATION (MM/H)

Rain Type	Cumulative precipitation (mm/h)
Weak rain	*≥ 0.1 and < 2.5*
Moderate rain	*≥ 2.5 and < 10*
Heavy rain	*≥ 10 and < 20*
Rainstorm	*≥ 20*

Regarding the neural network used in this study, the approach adopted was similar to that proposed by [22], with adjustments tailored to the context of this investigation. Next, we will briefly describe the architecture defined for the model and the process of assigning weights to the samples.

To calculate the sample weights, an interval-based approach was employed. Initially, all weights were initialized as equal for all samples. Then, three distinct intervals in the target sample values were identified: the majority, the intermediate, and the minority. These intervals were defined based on the values (Table I) of the training data. We assigned differentiated weights to each interval based on a specific calculation function, aiming to enhance the model's performance, especially concerning minority samples.

The structure of the MLP neural network was defined in terms of its layers and corresponding activations. The network consisted of an input layer with a number of neurons equal to the number of input attributes. Two hidden layers are utilized, with 64 and 32 neurons, respectively, both activated by the ReLU (Rectified Linear Unit) activation function. Dropout, a regularization technique, is applied after the first and second hidden layers to prevent overfitting. Finally, an output layer with a single neuron and linear activation is employed to generate predictions.

The analysis of comparisons between actual samples and predictions for 1 hour ahead represents a crucial strategy to investigate the effects of identified imbalances and the model's ability to handle short-term variations. To assess the results, the RMSE and MAE metrics were employed, where RMSE is defined as (1).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \quad (1)$$

Where n represents the total number of observations, y_i denotes the i-th actual observation, and $\hat{y}_i$ represents the i-th prediction. The MAE metric calculates the average of the absolute differences between actual observations and predictions, providing a direct measure of the average magnitude of prediction errors, and is given in (2):

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \quad (2)$$

Where $|y_i - \hat{y}_i|$ denotes the absolute value of the difference between the actual observation and the corresponding prediction.

Additionally, probabilistic parameters POD (3) and FAR (4) were also applied to examine the correspondence between observed and predicted rainfall rates.

$$POD = \frac{a}{a + c} \quad (3)$$

$$FAR = \frac{b}{a + b} \quad (4)$$

According to [23], these parameters classify events as hits, misses, false positives, or false negatives. These parameters depend on the relationship between hits and misses, where:

- *a*: Number of observed events predicted by the model.
- *b:* Events not observed but were predicted.
- *c:* Events observed but the model did not predict.

Additionally, we also have *d*, which represents events not predicted and that also do not occur but not used in the previous metrics. In the POD parameter, the score of the operation ranges from zero to one, where the maximum value of 1 reflects the most ideal performance, while 0 represents the opposite situation. In turn, the FAR evaluation scale starts from 0 to indicate the most favorable result possible. The combination of these two indices allowed an amplified evaluation of the predictions in each of the tests. The training process of the neural network was carried out with data from the period September 2021 to May 2023 (i.e. a total of 1.793 observations), divided in the proportion of 70% for training, 15% for validation, and 15% for testing.

IV. RESULTS AND DISCUSSIONS

In this study, the analysis focused on evaluating the effects of imbalanced data distribution on the prediction of rainfall rate using ANN. It is important to note that the scope of this investigation does not cover the verification of the predictive capacity of the model itself but rather explores the connection between data imbalance and its effects on model efficacy.

We acknowledge that the model has not yet achieved its optimal performance. To conduct this analysis, a series of tests was carried out using distinct datasets, and this allows the observation of how each dataset influenced the prediction outcomes. Fig. 5 presents a comparative analysis of test results in various scenarios, using new data from April 6th, 2023, covering only 12 hours when rainfall events were recorded by precipitation sensors.

These data constitute a sample that was not exposed to the model during the training process. The graph illustrates the discrepancies between the forecasts and the observed values for the specific case.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

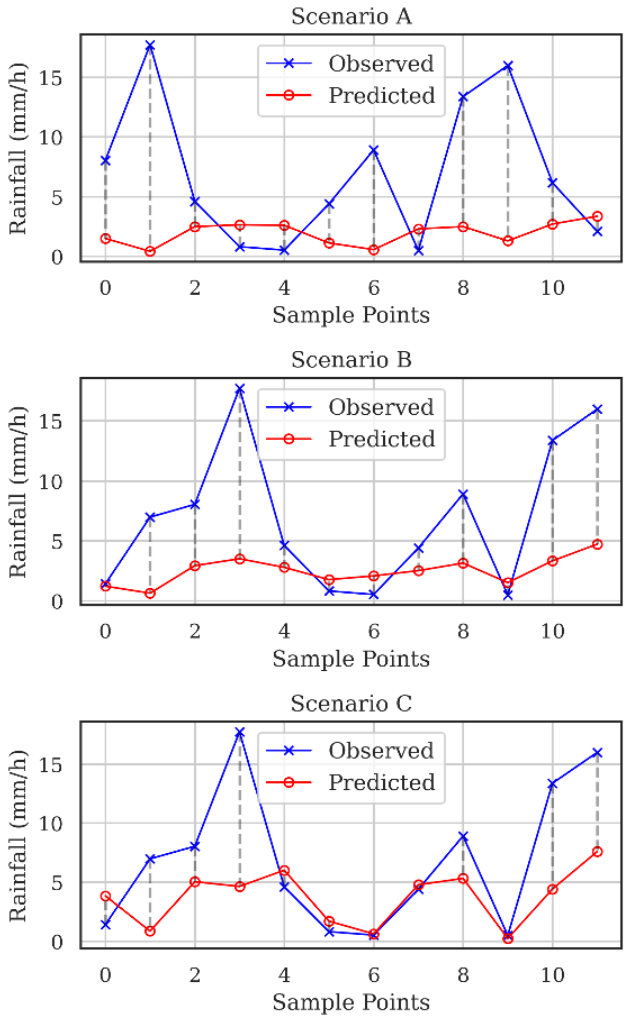


Fig. 5. Plot of discrepancies between predictions and observed values in tests with data from the observed day

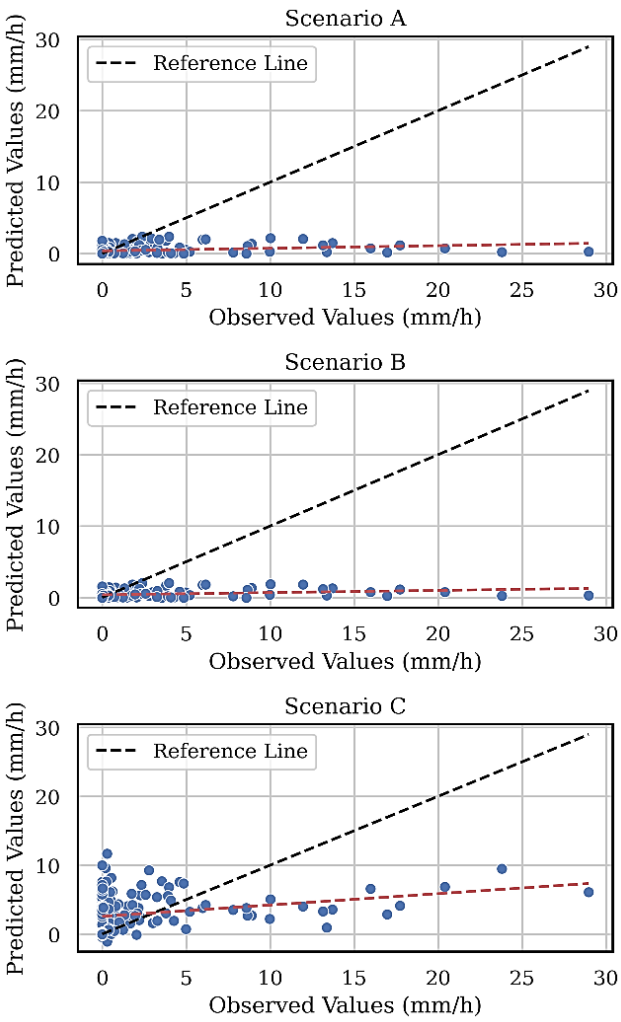


Fig. 6. Scatter plots illustrating the relationship between observed and predicted values, providing a comprehensive assessment of predictions

Upon objective analysis of Fig. 5, we emphasize how data imbalances impacted the predictions. This becomes evident when examining the discrepancies in Scenario C, where the differences between the forecasts (red line) and the actual values (blue line) decreased over the analyzed period after applying weights to the samples.

Each point of the red line above the blue points indicates an overestimation, while points below indicate an underestimation. When the lines coincide, it indicates a correct forecast. Notably, after applying weights to the less representative samples, a significant improvement in the predictions of Scenario C was observed, as evidenced by the closer alignment between the red and blue lines.

Additionally, for a more comprehensive assessment of predictions, we turn to scatterplots (Fig. 6), providing a visual representation that establishes the relationship between observed and predicted values. It is worth mentioning that in Fig. 4, the results presented show distinctions between the three tests performed (Scenarios A, B and C).

In the case of Scenario C, in the scatter plot where adjustments were applied to the sample weights based on the intensity of the rain during training, a notable improvement in the dispersion is evident with a notable alignment of the dashed red line with the reference line (black line dashed line) compared to Scenarios A and B.

Note that the model showed a better ability to predict more intense rainfall. This suggests that the adjustments made in Scenario C had a positive impact on the forecasts, allowing for a closer alignment between observed and predicted values. We emphasize that in this visual representation of the results, the closer the red line gets to the black line, the better the predictions. Despite the differences in approaches, the results in Scenario C are similar to class balancing techniques, indicating a convergence of performance between these types of strategies. This emphasizes the relevance of including rainfall intensity and balance approaches, reinforcing the search for more accurate and reliable forecasts.

Tables II to V present the results of the metrics applied in the evaluation and the outcomes obtained with the probabilistic parameters used to examine the correspondence between observed and predicted rainfall rates in the investigated scenarios.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

TABLE II. RESULTS OF EVALUATION METRICS WITH EVALUATION DATA AND TEST DATA

Scenarios	**Values for MAE and RMSE obtained in the conducted investigations**		
	Dataset	***MAE***	***RMSE***
A	*Validation data*	2.83	7.78
	Test data	2.88	7.77
B	*Validation data*	1.30	2.59
	Test data	**1.31**	**2.65**
C	*Validation data*	**1.19**	**2.51**
	Test data	**1.27**	**2.54**

TABLE III. RESULTS OF THE POD AND FAR PARAMETERS IN THE INVESTIGATIONS CONDUCTED IN SCENARIO A

Scenario A	**POD (Probability of Detection) and FAR (False Alarm Rate) obtained in the conducted investigations**			
	Rain Type	***POD***	***FAR***	***Interpretation***
Evaluate the model's ability to make predictions in a range of scenarios with varying occurrences of rainfall of different intensities.	*Weak rain*	*0.70*	*0.30*	***Good detection, moderate false alarms***
	Moderate rain	*0.65*	*0.35*	***Good detection, moderate false alarms***
	Heavy rain	*0.39*	*0.61*	*Low detection and/or high false alarms*
	All	***0.41***	***0.24***	***Reasonable detection, significant false alarms***

TABLE IV. RESULTS OF THE POD AND FAR PARAMETERS IN THE INVESTIGATIONS CONDUCTED IN SCENARIO B

Scenario B	**POD (Probability of Detection) and FAR (False Alarm Rate) obtained in the conducted investigations**			
	Rain Type	***POD***	***FAR***	***Interpretation***
Evaluate the direct impact of data imbalance on the model's predictions, with the data imbalances previously known.	*Weak rain*	*0.87*	*0.13*	***High detection, low false alarms.***
	Moderate rain	*0.69*	*0.31*	***Good detection, moderate false alarms.***
	Heavy rain	*0.19*	*0.81*	*Low detection and/or high false alarms.*

Scenario B	**POD (Probability of Detection) and FAR (False Alarm Rate) obtained in the conducted investigations**			
	Rain Type	***POD***	***FAR***	***Interpretation***
	All	***0.44***	***0.17***	***Reasonable detection, low false alarms.***

TABLE V. RESULTS OF THE POD AND FAR PARAMETERS IN THE INVESTIGATIONS CONDUCTED IN SCENARIO C

Scenario C	**POD (Probability of Detection) and FAR (False Alarm Rate) obtained in the conducted investigations**			
	Rain Type	***POD***	***FAR***	***Interpretation***
Investigate how considering rainfall intensity during training influences the model's predictions by applying higher weights to less representative samples.	*Weak rain*	*0.72*	*0.28*	***Good detection, moderate false alarms***
	Moderate rain	*0.85*	*0.15*	***High detection, low false alarms***
	Heavy rain	*0.62*	*0.38*	***Good detection, moderate false alarms***
	All	***0.43***	***0.21***	***Reasonable detection, low false alarms***

The metrics highlight the impact of data imbalance on predictions. Table 1 shows that in Scenario *C*, the best results were recorded in terms of MAE and RMSE, while Scenario *A* reveals the lowest performance. In Scenario *C*, with the application of weight adjustment to less representative samples, it outperforms both Scenario *A* and Scenario *B*.

Although the difference compared to Scenario *B* is relatively low, the results in Scenario *C* for validation metrics (MAE: 1.19, RMSE: 2.51) and test metrics (MAE: 1.27, RMSE: 2.4) suggest that the weight adjustment technique may be a good alternative for this type of context.

As presented, it is possible to observe important insights into the model behavior in the three scenarios studied. Scenarios *A*, *B*, and *C* highlight the importance of data representativeness and balance. Scenario A shows limitations in predictions for all types of rainfall but still performs better for lighter rains. Scenario *B* reveals high detection for both light and moderate rains, but predictions for heavy rains need improvement. In Scenario *C,* there is a good detection for light and moderate rains, but the detection of heavy rains remains limited, despite improvements compared to previous tests.

V. CONCLUSIONS

The prediction of weather events, such as heavy rainfall, has gained global prominence. These events are often linked to natural disasters such as floods and landslides, impacting lives and economies. Accurately forecasting rainfall and its intensity in the short term can minimize risks and damages,

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.05

proving crucial in sectors such as agriculture and logistics. The results of this study underscored the importance of data balance in the construction and effectiveness of prediction models. The sensitivity of these models to data details highlights the need to consider the representativeness of the data used.

The focus was on evaluating the impact of meteorological data imbalance on rainfall prediction, with the use of data from multiple sensors and Artificial Neural Networks (ANNs). Investigations in three scenarios, related to the imbalance in training and validating the model data, highlighted the importance of data balance for accurate detection and a reduced number of false alarms. Strategies such as adjusting weights on samples proved to be alternatives to enhance rainfall predictions, especially in intense events where imbalance can compromise accuracy. Weighted sampling techniques also proved effective in dealing with imbalances, improving the model performance in the investigated scenario.

REFERENCES

[1] L. Pires, K. Sušelj, L. Rossato, and J. Teixeira, "Analyses of Shallow Convection over the Amazon Coastal Region Using Satellite Images, Data Observations and Modeling," *Rev. Bras. Meteorol.*, vol. 33, pp. 366–379, Jun. 2018, doi: 10.1590/0102-7786332009.

[2] M. O. Andreae *et al.*, "The Amazon Tall Tower Observatory (ATTO): overview of pilot measurements on ecosystem ecology, meteorology, trace gases, and aerosols," *Atmospheric Chem. Phys.*, vol. 15, no. 18, pp. 10723–10776, Sep. 2015, doi: 10.5194/acp-15-10723-2015.

[3] V. A. Costa, "A narrative review of papers developed on the Amazon Tall Tower Observatory experimental site," *Res. Soc. Dev.*, vol. 10, no. 14, p. e73101421749, Oct. 2021, doi: 10.33448/rsd-v10i14.21749.

[4] W. R. Moninger *et al.*, "Summary of the First Conference on Artificial Intelligence Research in Environmental Sciences (AIRIES)," *Bull. Am. Meteorol. Soc.*, vol. 68, no. 7, pp. 793–800, Jul. 1987.

[5] D. W. McCann, "A Neural Network Short-Term Forecast of Significant Thunderstorms," *Weather Forecast.*, vol. 7, no. 3, pp. 525–534, Sep. 1992, doi: 10.1175/1520-0434(1992)007<0525:ANNSTF>2.0.CO;2.

[6] A. P. Almeida, "Uso de redes neurais para previsão de descargas elétricas nuvem-solo a partir de dados de radar: Um estudo de caso na Amazônia", Dissertation (Masters in Applied Computing), National Institute for Space Research, São José dos Campos-SP, Brazil, Sep. 2021.

[7] S. Haykin, *Neural Networks: A Comprehensive Foundation (2nd Edition) Neural Networks: A Comprehensive Foundation*. 1998.

[8] Z. Basha, N. Bhavana, P. Bhavya, and S. V, *Rainfall Prediction using Machine Learning & Deep Learning Techniques*. Aug. 2020, p. 97. doi: 10.1109/ICESC48915.2020.9155896.

[9] M. Akkisetti, M. Rajeevan, V. R. Madineni, J. Bhate, and c. v Naidu, "Nowcasting severe convective activity over southeast India using ground-based microwave radiometer observations," *J. Geophys. Res. Atmospheres*, vol. 118, pp. 1–13, Jan. 2013, doi: 10.1029/2012JD018174.

[10] M. Darji, V. Dabhi, and H. Prajapati, *Rainfall forecasting using neural network: A survey*. Jul. 2015. doi: 10.1109/ICACEA.2015.7164782.

[11] A. Maitra and R. Chakraborty, "Rain prediction using radiometric observations at a tropical location," in *2017 XXXIInd General Assembly and Scientific Symposium of the International Union of Radio Science (URSI GASS)*, Aug. 2017, pp. 1–4. doi: 10.23919/URSIGASS.2017.8105124.

[12] J. Joss and A. Waldvogel, "Ein Spektrograph fur Niedersch Pagsmessungen mit automatischer Auswertung," *Pure Appl. Geophys.*, vol. 68, pp. 240–246, Dec. 1967, doi: 10.1007/BF00874898.

[13] C. Caracciolo, F. Prodi, and R. Uijlenhoet, "Comparison between Pludix and impact/optical disdrometers during rainfall measurement campaigns," *Atmospheric Res.*, vol. 82, pp. 137–163, Nov. 2006, doi: 10.1016/j.atmosres.2005.09.007.

[14] M. Löffler-Mang and J. Joss, "An Optical Disdrometer for Measuring Size and Velocity of Hydrometeors," *J. Atmospheric Ocean. Technol. - J ATMOS OCEAN TECHNOL*, vol. 17, Feb. 2000, doi: 10.1175/1520-0426(2000)017<0130:AODFMS>2.0.CO;2.

[15] E. Habib, W. Krajewski, and A. Kruger, "Sampling Errors of Tipping-Bucket Rain Gauge Measurements," *J. Hydrol. Eng.*, vol. 6, Apr. 2001, doi: 10.1061/(ASCE)1084-0699(2001)6:2(159).

[16] G. Upton and A. R. Rahimi, "On-line detection of errors in tipping-bucket raingauges," *J. Hydrol.*, vol. 278, pp. 197–212, Jul. 2003, doi: 10.1016/S0022-1694(03)00142-2.

[17] S. Ochoa-Rodriguez, L.-P. Wang, P. Willems, and C. Onof, "A Review of Radar-Rain Gauge Data Merging Methods and Their Potential for Urban Hydrological Applications," *Water Resour. Res.*, vol. 55, Aug. 2019, doi: 10.1029/2018WR023332.

[18] Radiometrics Corporation, "MP3000A Profiler Operator's Manua." Boulder, CO, USA, 2008.

[19] K. Bhattacharyya and M. Islam, "SHORT TERM RAIN FORECASTING FROM RADIOMETRIC BRIGHTNESS TEMPERATURE DATA," *J. Mech. Contin. Math. Sci.*, vol. 15, Feb. 2020, doi: 10.26782/jmcms.2020.02.00007.

[20] E. R. Westwater, "The accuracy of water vapor and cloud liquid determination by dual-frequency ground-based microwave radiometry," *Radio Sci.*, vol. 13, no. 4, pp. 677–685, Aug. 1978, doi: 10.1029/RS013i004p00677.

[21] Y. Chen *et al.*, "Improving the heavy rainfall forecasting using a weighted deep learning model," *Front. Environ. Sci.*, vol. 11, p. 1116672, Feb. 2023, doi: 10.3389/fenvs.2023.1116672.

[22] L. C. P. Velasco, R. P. Serquiña, M. S. A. Abdul Zamad, B. F. Juanico, and J. C. Lomocso, "Week-ahead Rainfall Forecasting Using Multilayer Perceptron Neural Network" *Procedia Comput. Sci.*, vol. 161, pp. 386–397, Nov. 2019, doi: 10.1016/j.procs.2019.11.137.

[23] R. Levine, "Statistical Methods in the Atmospheric Sciences by Daniel S. Wilks," *J. Am. Stat. Assoc.*, vol. 95, pp. 344–345, Mar. 2000, doi: 10.2307/2669579.

AUTHORS

Lourenço José Cavalcante Neto

He is currently a student in the Master's Program in Applied Computing at the National Institute for Space Research - INPE, located in São José dos Campos (SP), Brazil. He obtained a specialization in Information Technology in Education from the Faculdade União Cultural do Estado de São Paulo - UCESP in 2016. He completed his degree in Computing from the Federal Institute of Education, Science and Technology of Tocantins - IFTO, Araguatins campus, in 2015. In addition, he has technical training in Internet Computing at IFTO, Palmas campus, completed in 2014. During the period from March 2016 to May 2019, he was part of the permanent teaching staff at the Federal Institute of Education, Science and Technology of Mato Grosso - IFMT, on Campus Forward from Guarantã do Norte. He is currently part of the faculty at IFTO, Araguatins campus. Its main areas of activity are centered on data science and Artificial Intelligence applied to atmospheric sciences, as well as the development of systems using the PHP and Python languages. Since 2022, he has been dedicated to applying computational techniques, such as machine learning, to understand rain patterns in the Amazon region.

Alan James Peixoto Calheiros

He holds a bachelor's degree in Meteorology from the Federal University of Alagoas (2006), a master's degree (2008), and a Ph.D. (2013) in Meteorology from the National Institute for Space Research (INPE). Currently, he is a Technologist and Professor in the applied computing graduate program at INPE, specializing in Precipitation, Storm Forecasting, Satellite and RADAR Rainfall Estimation, and Cloud Microphysics. He develops products for storm monitoring and forecasting, participating in projects to characterize clouds across the entire Brazilian region. Since 2018, he has been focusing on the application of computational techniques, such as machine learning and complex networks, to understand rainfall regimes in South America. Currently, he leads new international partnership projects at INPE focused on global satellite rainfall monitoring.

Classification of Failure Using Decision Trees Induced by Genetic Programming

ARTICLE HISTORY
Received 2 January 2024
Accepted 19 April 2024
Published 08 July 2024

Rogério Costa Negro Rocha
Graduate Program in Computational Modeling and Systems
State University of Montes Claros
Montes Claros, Brazil
rogeriocostanegro@hotmail.com
ORCID: 0009-0002-4667-9656

Laércio Ives Santos
Campus Montes Claros Federal Institute of Norte de Minas Gerais
Montes Claros, Brazil
laercio.ives@gmail.com
ORCID: 0000-0001-6504-7692

Rafael Almeida Soares
Graduate Program in Computational Modeling and Systems
State University of Montes Claros
Montes Claros, Brazil
rafael.almeida.soares2012@gmail.com
ORCID: 0009-0006-5544-9798

Franciele Alves Barbosa
Graduate Program in Computational Modeling and Systems
State University of Montes Claros
Montes Claros, Brazil
francielealvesb10@gmail.com
ORCID: 0009-0005-3964-7391

Marcos Flávio Silveira Vasconcelos D'Angelo
Departament of Computer Science State University of Montes Claros
Montes Claros, Brazil
marcos.dangelo@unimontes.br
ORCID: 0000-0001-5754-3397

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

R. Rocha, L. Santos, R. Soares, F. Barbosa and M. D'Angelo, "Classification of Failure Using Decision Trees Induced by Genetic Programming", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024.

Classification of Failure Using Decision Trees Induced by Genetic Programming

Rogério Costa Negro Rocha
State University of Montes Claros
Graduate Program in Computational Modeling and Systems
Montes Claros, Brazil
rogeriocostanegro@hotmail.com

Laércio Ives Santos
Federal Institute of Norte de Minas Gerais
Campus Montes Claros
Montes Claros, Brazil
laercio.ives@gmail.com

Rafael Almeida Soares
State University of Montes Claros
Graduate Program in Computational Modeling and Systems
Montes Claros, Brazil
rafael.almeida.soares2012@gmail.com

Franciele Alves Barbosa
State University of Montes Claros
Graduate Program in Computational Modeling and Systems
Montes Claros, Brazil
francielealvesb10@gmail.com

Marcos Flávio Silveira Vasconcelos D'Angelo
State University of Montes Claros
Departament of Computer Science
Montes Claros, Brazil
marcos.dangelo@unimontes.br

***Abstract*—**Fault classification in industrial processes is of paramount importance, as it allows the implementation of preventive and corrective measures before catastrophic failures occur, which can result in significant repair costs and production loss, for example. Therefore, the purpose of this study was to develop a classification model by merging the concepts of Decision Trees with Genetic Programming. To accomplish this, the proposed model randomly generates a set of decision trees using the adapted Tennessee Eastman dataset. The generation of these trees does not rely on classical construction logic; instead, they employ an approach where the structure and characteristics of the trees are randomly determined and adjusted throughout the evolutionary process. This approach enables a broader exploration of the search space and may lead to diverse solutions. The results obtained were moderate, largely due to the high number of target classes for classification (21 classes), resulting in the creation of complex trees. The average accuracy on the test data was 0.75, indicating the need to implement new alternatives and enhancements in the algorithm to improve the results.

Keywords*—*decision trees, multiclass classification, fault detection, genetic programming

I. Introduction

Fault Classification Problems have been widely explored in various fields of engineering and science, being of utmost relevance for the detection and prevention of anomalies in systems and processes. Early fault detection plays a fundamental role in proactive maintenance and ensuring efficient and safe operations in complex and interconnected environments, such as industrial production systems, telecommunications networks, and transportation systems.

In this context, fault classification involves the development of models and algorithms capable of identifying subtle patterns in data, distinguishing between normal situations and anomalous behaviors. Advanced machine learning and data mining techniques have been applied to address these challenges.

Precise fault detection and classification not only reduce costs associated with unplanned downtime, but also contribute to process optimization and worker safety.

To achieve this, models based on Machine Learning (ML) techniques can be used for fault classification. ML techniques are interesting because they emulate the human way of thinking and decision-making, analyze large datasets containing many features in a reasonable time, and can handle complex relationships between data, making them more accurate than human experts in some specific situations [1].

Given the above, decision trees are considered, a ML technique used for classification problems. The use of decision trees has proven to be a promising approach in the context of fault classification problems. Decision trees offer an intuitive and interpretable way to model complex patterns in data, allowing the identification of relevant features for the classification of different types of faults in industrial systems. Additionally, as highlighted by [2], the hierarchical nature of decision trees mirrors human decision-making processes, making them suitable for analyzing anomalous behaviors in interconnected systems.

The application of decision trees for fault classification can also be enhanced with data preprocessing techniques, such as relevant feature selection. Thus, the use of decision trees offers a versatile and effective approach to address the challenges of fault detection and classification in complex systems. However, such algorithms often use a greedy strategy and tend to fall into local optima. Moreover, the recursive partitioning policy in the construction phase can result in datasets with low cardinality for the attribute selection process in deeper tree nodes, causing data overfitting.

Furthermore, researchers have considered the application of evolutionary algorithms to induce decision trees, specifically through Genetic Programming (GP). GP is an evolutionary algorithm that evolves a set of individuals represented in the form of trees [3]. When GPs are applied to the induction of decision trees, it is possible to deal with multiple attributes simultaneously, reducing the dependence on feature selection methods in preprocessing and still providing a global search strategy [4]. This is an interesting approach to be tested, given that in recent years, it is not

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

common to find works that use evolutionary computation techniques to induce decision trees.

Therefore, this work aims to build a multiclass classification algorithm based on decision trees induced by genetic programming, with the purpose of classifying faults in the adapted database of the Tennessee Eastman Process Simulation and analyzing its accuracy results. Experiments were conducted to assess the quality and complexity of the solutions found. The results obtained indicated that the model presents moderate results for fault classification in the chosen database and results in complex trees; therefore, new strategies need to be applied to the algorithm to achieve better results and performance.

II. Literature Review

A. Decision Trees

Decision Trees are widely used algorithms in machine learning to solve classification and regression problems. Data is organized in a tree-like structure, wherein each inner node signifies a decision derived from a particular attribute, and each terminal node, or leaf, corresponds to either a classification label or a regression value [5].

One of the advantages of decision trees is their interpretability. Their representation, especially when viewed graphically, is easily understandable. One can follow the logic of each node and interpret it until reaching a leaf node, which indicates the class of the instance, for example. Additionally, decision trees have the ability to handle both numerical and categorical data. They can represent complex relationships between attributes and classes, making them suitable for modeling nonlinear data [6].

To evaluate a decision tree, the Misclassification Error criterion can be used [6]. In this criterion, the number of correct predictions is measured by comparing predicted outputs with true outputs, resulting in accuracy. Accuracy assesses the ratio of correctly classified examples to the total number of evaluated examples. Higher accuracy indicates a greater number of accurate predictions.

B. Genetic Programming

Genetic Programming (GP) is an artificial intelligence technique that uses principles inspired by biological evolution to evolve solutions for complex problems. In this approach, a set of random solutions is represented as genetic structures that can be combined and mutated over several generations, generating new individuals representing new solutions, with the aim of finding optimal or approximate solutions to a problem [3].

Genetic programming starts with an initial population of potential solutions (good or bad), known as individuals. In each generation, these individuals are evaluated based on a fitness function that quantifies how well they solve the given problem. Individuals with higher fitness are more likely to be selected for reproduction, where crossover (recombination) and mutation operations occur, similar to the processes of genetic evolution [3].

The genetic programming approach allows the exploration of a broad solution space in search of effective solutions for complex and multidimensional problems. It is applied in various fields, including optimization, machine learning, and modeling.

C. Genetic Programming Applied to Decision Trees

Genetic programming (GP) applied to decision trees represents an innovative approach in the field of artificial intelligence. In this paradigm, decision trees are portrayed as chromosomes, enabling the evolution of effective solutions for multiclass classification problems. [7] emphasizes that this genetic representation facilitates the application of evolutionary operators, such as crossover and mutation, to generate new generations of decision trees, allowing the discovery of novel and improved solutions to the addressed problem.

The evolutionary process unfolds over iterations, where trees are selected for a reproduction pool, forming pairs that crossbreed to produce new individuals. Trees that are more adapted, as per a fitness function, have higher chances of being chosen for reproduction. This evolutionary approach aims to find decision trees that optimally fit the data patterns. Nguyen et al. [8] underscore the importance of a well-defined fitness function to efficiently guide the evolutionary process.

The advantages of this approach include the ability to handle complex problems and the flexibility to evolve decision tree structures without the need for manual definition. However, challenges such as the potential uncontrolled growth of the tree (overfitting) need to be addressed. [9] discuss strategies, such as penalties in the fitness function, to mitigate these challenges and ensure more generalized solutions.

In summary, genetic programming applied to decision trees offers a promising approach to solve multiclass classification problems, combining the flexibility of genetic evolution with the structured representation of decision trees. However, the careful selection of parameters and strategies to prevent overfitting is crucial in the development and implementation of this technique.

III. Methodology

A. Used Database

The Tennessee Eastman Process Simulation database is widely recognized as a benchmark in the field of process engineering and fault detection. Developed by the Oak Ridge National Laboratory in the United States, this database was designed to allow the evaluation and comparison of fault detection, diagnosis, and prediction algorithms and methods in a simulated environment of a complex chemical process [10]. Researchers employ this dataset to test and compare anomaly detection algorithms, pattern identification, and diagnosis in a chemical process scenario, fostering advancements in the field [11].

In total, the original database has 55 columns, with 54 input attributes and 1 output attribute. The column that presents the output attribute is called "faultNumber", representing the fault number, ranging from 0 to 21. This expresses a classification of 22 fault classes, where class 0 means no fault, and the other classes (1 to 21) represent the fault classification number.

In the present study, a database derived from a fault detection model based on Qualitative Trend Analysis (QTA),

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

proposed by [12], was used. This model adopts a two-step process for fault detection in the Tennessee Eastman database. The first step uses the fuzzy set theory, while the second one relies on a Bayesian approach for detecting change points in time series, providing an indication of a possible fault. If this indication is established, the proposed model takes responsibility for identifying the specific nature of the fault.

Additionally, 22 variables were eliminated from the dataset in two phases by [12]. In the first phase, a correlation matrix obtained from the 52 input variables in the author's used database was employed. Variables with a correlation below 0.6 were eliminated, resulting in a reduction of 15 variables.

In the second phase, 7 more variables that showed no indications of faults by the Fuzzy/Bayesian approach were discarded. In other words, in the calculation of the new probability vector for change points for these variables, no changes were detected. Consequently, the vectors were zeroed out, resulting in the variables having only zero values, which does not affect fault classification.

Thus, 30 input variables were retained, which were used to train and test the classifier proposed in this work.

In the end, the dataset proposed by [12] presented 4180 instances, with 30 input attributes and 1 output attribute. In this sampling, all classes have 200 instances, except for classes 1, 9, 15, 19, 18, and 20, which have 199, 190, 198, 195, 199, and 199 instances, respectively.

B. Developed Algorithm

Using the concepts of decision trees and genetic programming, a predictive model was developed using Python. Three classes were created in total: 'Node', which stores the nodes of the tree, 'DecisionTree', that represents the classification trees, and finally, the 'PG' class (Genetic Programming). This contains the genetic operators that create the tree population and evolve them with the aim of finding better solutions for the fault classification problem.

When executed, the algorithm takes training parameters and the maximum number of iterations as inputs. It can also receive additional parameters such as the maximum depth of the trees, population size, crossover rate, mutation rate, elitism, the number of individuals participating in tournaments during the selection phase, and the number of split points for individuals during the crossover phase.

The pseudocode of the developed algorithm can be seen in Fig. 1.

Algorithm 1 Genetic Programming(*TrainingData, MaxIt*)

1: $Pop_1 \leftarrow$ *GenerateInitialPopulation*(*TrainingData*);
2: $g \leftarrow 1$;
3: **while** $g <$ *MaxIt* **do**
4: $Pop_g \leftarrow$ *CalculateFitness*(Pop_g);
5: $Selected \leftarrow$ *Selection*(Pop_g);
6: $NewGeneration \leftarrow$ *Crossover*($Selected$);
7: $NewGeneration \leftarrow$ *Mutation*($NewGeneration$);
8: $NewGeneration \leftarrow$ *Elitism*($NewGeneration$);
9: $Pop_{g+1} \leftarrow NewGeneration$;
10: $g \leftarrow g + 1$;
11: **end while**
12: **Return** Pop_g

Fig. 1. Developed algorithm

In line 1 of the pseudocode, we have the first function to be called, which is *GenerateInitialPop()* that takes the training data as a parameter, aiming to generate the initial population randomly.

After the initial population is formed, the algorithm enters a loop, which lasts for the specified number of generations. In each generation within the loop, the population goes through the *Evaluation()* function (line 4), which calculates the fitness level of each individual. Right after, the *Selection()* (line 5) occurs, to select individuals for the *Crossover()* phase (line 6) through tournaments, creating a new generation, where individuals may undergo the *Mutation()* process (line 7). After the new generation is formed, the *Elitism()* function (line 8) is called, with the goal of saving the best individual from the previous generation and placing it in the new generation. Finally, the current population is replaced by the new generation (line 9), and the current generation number is incremented (line 9).

At the end of the loop, the final population found by the algorithm is returned (line 12), containing the last generation of individuals found by the model. From this, it is possible to select the best individual or individuals from this population to perform tests using test data, evaluating the test accuracy of the tree found, representing how well the tree performed in predicting fault classifications.

C. Generation of The Initial Population

The generation of the initial population is done through the *GenerateInitialPop()* function, which takes the training data as a parameter. In this function, a number of decision tree individuals equivalent to the user-specified number are created.

The construction of a tree is based on the training input data (instances/input attributes) and output (fault class). From this, nodes are randomly generated, where the input attribute related to this node, the split threshold, and the data split operator are randomly chosen. The function also checks various stopping conditions, such as the maximum tree depth, the minimum number of samples for a split, and whether all samples belong to a single class. If the stopping condition is met, the next node to be generated will be a leaf node, representing a class to be predicted.

Fig. 2 represents a decision tree generated by the function. In this example, it can be observed that the root node has the attribute 24 (equivalent to column 24 of the database) randomly selected, where the threshold was randomly chosen as 0.03, and the operator was <. When analyzing the set of training instances, if the value of column 24 of the instance is < 0.03, the instance proceeds to the left node; otherwise, it proceeds to the node on the right, and this process repeats until the instance reaches a leaf node, where the predicted class will be determined.

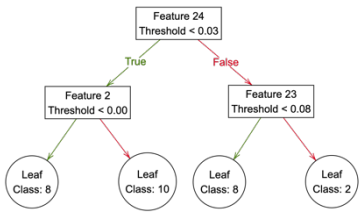


Fig. 2. Example of generated decision tree.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

D. Fitness Function

To calculate the fitness of each decision tree in the population, the Misclassification Error criterion was used to measure the accuracy of the individual. This accuracy serves as its fitness. For this purpose, the *CalculateFitness()* function utilizes the training data, where the tree uses the input data to predict the outputs (fault classes). After the prediction, the predicted outputs are compared with the true outputs of the training data. In other words, it quantifies the ratio of correctly classified instances compared to the total instances in the dataset, providing an overall measure of the model accuracy in predicting the correct classess. This accuracy value is the fitness of the individual. This calculation is performed for each individual in the population.

E. Selection and Crossover

The selection operator chosen is tournament selection. In this method, there will be a number of tournaments equivalent to the size of the population. For each tournament, two individuals from the current population are randomly selected and compete against each other. The one with the higher fitness wins the tournament, and a copy of it is added to a list of winners. An individual may be drawn for competition more than once. Moreover, the number of competing individuals per tournament can be changed by the user.

After the tournaments and the list of winners are complete, the individuals undergo crossover. In this phase, two random individuals are taken from the list of winners to undergo crossover. If the crossover probability is equal to or greater than the defined crossover rate, crossover occurs; otherwise, the function returns the two randomly selected individuals. If crossover occurs, the two individuals are fragmented at one or more random points, and these segments are exchanged between the individuals, generating two new offspring individuals, which are then returned by the function. The number of crossovers that occur is equivalent to half the population size. Each return from the crossover function (two resulting individuals) is added to the list of the new generation. Consequently, a new generation is formed.

An example of a cross between two individuals can be seen in Fig. 3. In this example, in Individual 1, the node 'Feature 9' was segmented from the individual. Meanwhile, in Individual 2, it was the 'Feature 2' node that underwent segmentation. Following the segmentation process, the individuals crossbreed, giving rise to two new children, thus exchanging segments between them. Consequently, 'Child 1' is a copy of 'Individual 1,' but it now includes the 'Feature 2' node where 'Feature 9' used to be. On the other hand, 'Child 2' is a copy of 'Individual 2,' but with the 'Feature 9' node now in the place of 'Feature 2.' These two new individuals represent fresh solutions to the problem.

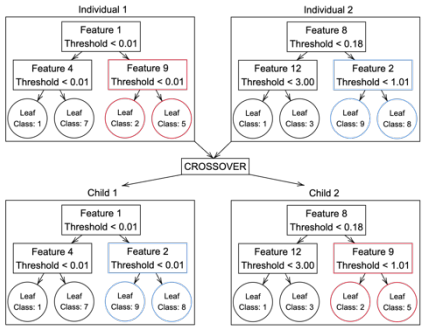


Fig. 3. Example of crossover between trees.

F. Mutation

In this phase, each individual in the new generation has the possibility of undergoing mutation. Observing the mutation rate defined by the user, if a randomly generated decimal number is greater than or equal to this rate, the individual undergoes mutation. In this process, a node of the tree is randomly selected, and its attribute, threshold, and split operator are randomly modified. If the selected node is a leaf node, the target class of that node is randomly modified among the possibilities, which range from 1 to 21.

Fig. 4 illustrates an example of an individual that was selected for mutation. In this individual, the node 'Feature 9' was chosen and underwent changes. Previously, this node used input attribute 9, with a threshold of 0. After the mutation, this node now uses input attribute 7, with a threshold of 2.31, thus becoming the 'Feature 7' node.

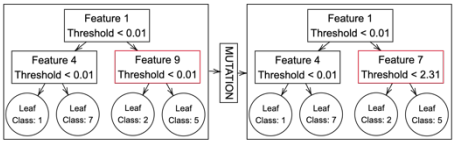


Fig. 4. Tree mutation example.

G. Elitism

After the new generation is formed, to prevent the loss of the best individual from the previous population, the elitism technique is applied. The individual with the lowest fitness in the new generation is replaced by the individual with the highest fitness from the current generation. This prevents the population from degrading rapidly in quality.

IV. Tests and Results

After the algorithm implementation, tests were initiated. To achieve this, the database was divided using a stratified sampling strategy into training and testing sets, allocating 70% of the data for training and 30% for testing purposes. Additionally, the following parameters were defined:

- Maximum tree depth: 100.
- Number of generations: 1,000 generations.
- Population size: 200 individuals.
- Crossover rate: 0.9.
- Mutation rate: 0.6.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

- Elitism enabled.
- Tournament selection (2 competing individuals).
- Single-point crossover.

With the above settings, 100 algorithm executions were performed, and the average of the obtained results was calculated. The results were:

- Average run time: 02:54:41.
- Average number of features used: 29.4.
- Medium depth: 41.4.
- Average number of nodes: 157.
- Average training accuracy: 0.772.
- Average test accuracy: 0.755.

The Fig. 5 presents the average test accuracies for each class, with a standard deviation of 0.2682 in this case. It can be observed that the prediction for classes 3 and 9 had an average accuracy below 0.2, with class 9 being the worst predicted by the model. Additionally, classes 20 and 21 had averages lower than 0.6, and classes 4, 11, 13, 16, and 18 achieved an average accuracy below 0.8. On the other hand, the remaining classes (12 in total) achieved accuracies higher than 0.8.

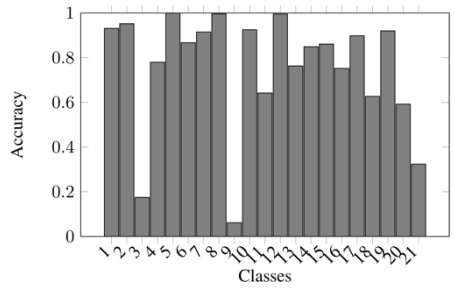


Fig. 5. Average test accuracy by class

Among the 10 tests conducted, the one with the best performance showed a training accuracy of 0.804 and a test accuracy of 0.799. Analyzing the Fig. 5 and the average of the results obtained, it is possible to notice that the average training accuracy obtained was 0.772, and the average test accuracy was 0.755, indicating these results as moderate. Regarding the average accuracy obtained per class, there were occurrences of extremely low accuracies, especially for classes 3 and 9, including accuracies equal to 0 in some of their executions, meaning that the resulting genetic algorithm made errors in all predicted classifications. On the other hand, 12 out of the 21 classes achieved accuracies higher than 0.8, indicating promising results.

V. Conclusions

This work aimed to present an approach for fault detection and classification, evaluating its performance when applied to the Tennessee Eastman Process. Decision trees induced by genetic programming were used to build and train the predictive classification model. The results of this application were collected and analyzed.

It is important to highlight that approaches based on decision trees can provide interpretable models, and the application of such models in the Tennessee Eastman Process has not been found in the previous literature. In this sense, this work stands as one of the first to use interpretable approaches in fault classification for the Tennessee Eastman Process dataset.

Another point to be discussed concerns the reduction of input attributes. By default, the dataset has 30 such attributes, and in a few executions, the proposed model managed to reduce this quantity to a maximum of 28 attributes. Although there was a reduction in some tests, this number is not significant or consistent.

Despite decision trees being simple models to understand and interpret, as their decisions are represented in a hierarchical structure that is easily comprehensible, facilitating explanations to non-technical users, the interpretability of the trees obtained by the model was hindered by their size. The trees had an average depth of 41.4 and an average number of nodes of 157. In light of these results, it identifies a greater difficulty in interpreting the resulting trees due to their size. Such size is also due to the complexity of the 21-class fault classification problem, which is an extensive issue.

Through the aforementioned ideas, it is concluded that, despite the model not achieving satisfactory results for all classes, a good part of the classes was predicted reasonably or adequately. Moreover, it is the first study that uses an interpretable model applied to the Tennessee Eastman dataset. However, the model needs changes and refinements for better results.

For future work, it is necessary to apply optimization techniques to improve the algorithm performance, aiming to reduce its execution time. Additionally, implementing functionalities and strategies that make the trees more interpretable and provide better accuracy results is crucial. The intention is to apply niche techniques, specifically fitness sharing based on Hamming distance, to increase the population diversity, and implement pruning techniques to reduce the size of the trees and make them more interpretable.

References

[1] A. Rajkomar, J. Dean, e I. Kohane, “Machine Learning in Medicine,” New England Journal of Medicine, vol. 380, no. 14, pp. 1347-1358, Abr. 2019.

[2] E. F. Brown et al., “Hierarchical decision trees for anomaly detection in interconnected systems,” in Proceedings of the International Conference on Industrial Engineering, pp. 126–132, 2020.

[3] J. R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992.

[4] R. K. DeLisle and S. L. Dixon, “Induction of decision trees via evolutionary programming,” Journal of Chemical Information and Computer Sciences, vol. 44, no. 3, pp. 862–870, 2004.

[5] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, Classification and Regression Trees. Chapman & Hall, 1984.

[6] A. Silva, T. Killian, I. D. Jimenez Rodriguez, S. Son, e M. Gombolay, “Optimization Methods for Interpretable Differentiable Decision Trees in Reinforcement Learning,” arXiv, 2019.

[7] J. R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1985.

[8] Q. U. Nguyen, M. Zhang, K. Zhang, and S. Li, “Evolutionary construction of decision trees for multiclass classification,” IEEE Transactions on Evolutionary Computation, vol. 19, no. 6, pp. 822–834, 2015.

[9] N. Javed, F. Gobet, e P. Lane, “Simplification of genetic programs: a literature survey,” *Data Mining and Knowledge Discovery*, vol. 36, no. 4, pp. 1279-1300, Abr. 2022.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

[10] R. W. J. Westerhout, F. J. J. Verhagen, and P. M. J. van den Hof, "Monitoring and diagnosis of industrial processes using chemomet- ric techniques," Computers & Chemical Engineering, vol. 27, no. 9, pp. 1259–1273, 2003.

[11] P. Wang and H. Wang, "A review of data-driven approaches for process systems fault detection and diagnosis," Computers & Chemical Engi- neering, vol. 94, pp. 188–200, 2016.

[12] M. F. D'Angelo, R. M. Palhares, R. H. Takahashi, and R. H. Loschi, "Fuzzy/bayesian change point detection approach to incipient fault detection," Control Theory & Applications, IET, vol. 5, pp. 539–551, 2011

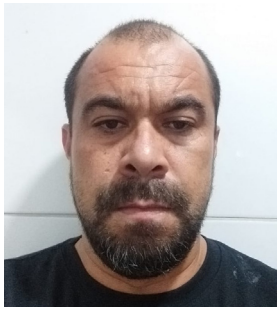
ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.06

AUTHORS

Rogério Costa Negro Rocha

Rogério Costa Negro Rocha graduated with a Bachelor's degree in Information Systems in 2021 from the Federal Institute of Northern Minas Gerais, Salinas Campus. Currently, he is pursuing a Stricto Sensu Graduate Program in Computational Modeling and Systems at the State University of Montes Claros – UNIMONTES, Brazil, with an expected completion date by November 2024. He has prior experience in software development, web development, and mobile development, particularly focusing on systems tailored for enterprise and industrial management, as well as retail management and e-commerce projects. Additionally, he has worked on projects in the field of photometry, following agile Scrum methodologies. Moreover, he has academic experience in machine learning, natural computing, and evolutionary computation. His interests encompass software and mobile development, utilizing Java and Dart; web development with PHP, HTML, CSS, and JavaScript; evolutionary computing, leveraging Python, particularly employing genetic algorithms and genetic programming for resource allocation and optimization problems; and pattern recognition in images using Convolutional Neural Networks, also utilizing Python.

Laércio Ives Santos



Laércio Ives Santos graduated in Information Systems in 2008, obtained a Master's degree in Computational Modeling and Systems, and a Doctorate in Health Sciences in 2021, all from the State University of Montes Claros - UNIMONTES. Currently, he is a professor at the Federal Institute of Education, Science, and Technology of Northern Minas Gerais, in Montes Claros, mainly teaching in the courses of Computer Science, Information Technology, and Electrical Engineering. His main disciplines include: Database, Computer Programming, Artificial Intelligence, and Natural Computing. He has been a Visiting Professor in the Computational Modeling Program at UNIMONTES since 2022. His experience includes research using Machine Learning and Artificial Intelligence techniques for diagnosing engine and process failures, as well as predicting events related to the medical field. His areas of interest include: Evolutionary Computing, Swarm Intelligence, Artificial Neural Networks, ensemble learning, as well as relational and NoSQL databases, and software development in languages such as Python, MATLAB, and JavaScript.

AUTHORS

Rafael Almeida Soares

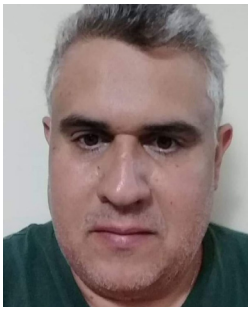
Rafael Almeida Soares completed his Bachelor's degree in Information Systems in 2022 at the Federal Institute of Education, Science, and Technology of Northern Minas Gerais, Salinas Campus. Currently, he is enrolled in the Stricto Sensu Graduate Program in Computational Modeling and Systems at the State University of Montes Claros, pursuing a Master's degree, expected to be completed by December 2024. Professionally, Rafael holds experience in backend development and mobile app development. He collaborated with the doctoral program in Health Sciences at the State University of Montes Claros to develop an application for childhood vaccination. His professional interests lie in technology applied to agriculture, livestock farming, and healthcare. Rafael's academic and professional pursuits extend to pattern recognition in images using Convolutional Neural Networks. He has already developed a system for license plate recognition. Moreover, he is keenly interested in machine learning, optimization, evolutionary computing, and the development of web and mobile applications.

Franciele Alves Barbosa

Franciele Alves Barbosa graduated with a Bachelor's degree in Information Systems in 2021 from the Federal Institute of Northern Minas Gerais, Salinas Campus. Currently, she is pursuing a Stricto Sensu Graduate Program in Computational Modeling and Systems at the State University of Montes Claros – UNIMONTES, Brazil, with an expected completion date by December 2024. She has prior experience in data mining, machine learning, time series forecasting, software development, web development, and mobile development. She has worked on a research project involving clustering reference evapotranspiration time series for the state of Minas Gerais using the K-means and Ward algorithms. Currently, she has worked on a project that utilizes artificial intelligence for healthcare, with a focus on cervical cancer diagnosis. Moreover, she has academic experience in machine learning, time series forecasting, deep learning and data science. Her interests encompass data mining, deep learning, time series forecasting and in the python programming language.

AUTHORS

Marcos F. Silveira V. D'Angelo



Marcos Flávio Silveira Vasconcelos D'Angelo joined the State University of Montes Claros in 2000 as an Associate Professor in Information Science. He earned his B.S. and M.S. degrees in Electrical Engineering from the Pontifical Catholic University of Minas Gerais in 1998 and 2000, respectively, and his Ph.D. in Electrical Engineering from the Federal University of Minas Gerais in 2010. His main research interests encompass dynamic systems, optimization theory and applications, and soft computing. Specifically, his work has focused on maintenance engineering. D'Angelo served as the coordinator of the systems engineering course and was a member of the university council at the State University of Montes Claros. Currently, he serves as a reviewer for journals and conferences, both nationally and internationally. To date, he has published approximately 49 full papers in journals, 4 chapters in books, and 45 full papers in conference proceedings. Additionally, he has coordinated research projects funded by grants and technological innovation projects.

R. Rocha, L. Santos, R. Soares, F. Barbosa and M. D'Angelo, "Classification of Failure Using Decision Trees Induced by Genetic Programming", Latin-American Journal of Computing (LAJC), vol. 11, no. 2, 2024.

A Comparative Study Between the Brazilian Stock Market and Cryptocurrencies

ARTICLE HISTORY
Received 01 April 2024
Accepted 21 May 2024
Published 08 July 2024

Marjori Klinczak
UFPR, Unifatec-PR
Curitiba, Brazil
marjori.klinczak@unifatecpr.com.br
ORCID: 0009-0009-2028-8359

Egon Wildauer
UFPR, Unifatec-PR
Curitiba, Brazil
egon@ufpr.br
ORCID: 0000-0003-2340-8984

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

A Comparative Study Between the Brazilian Stock Market and Cryptocurrencies

Marjori Klinczak
UFPR, Unifatec-PR
Information Management
Curitiba, Brazil
marjori.klinczak@unifatecpr.com.br

Egon Wildauer
UFPR, Unifatec-PR
Information Management
Curitiba, Brazil
egon@ufpr.br

***Abstract*—**The Brazilian Stock Market has been experiencing an increase in trading volume, and this shows an improvement in indices. This phenomenon is due to the adoption of Corporate Governance practices, improvement in institutional environments, and greater liquidity in national markets. In this scenario, blockchain technology has become popular in recent years, with various applications, ensuring transaction identification, authenticity, reliability, transparency, equity, and interoperability, along with the emergence of smart contracts. However, the most well-known cryptocurrency is Bitcoin, followed by Ethereum, which was the first to allow the use of smart contracts, and Solana, created in 2018, already holds the fourth position, with great expectations for future growth. The popularization of this asset class may represent an investment opportunity; on the other hand, there is research on its possible relationship with other markets and assets, such as gold, the dollar, or even the Dow Jones index. However, the literature on this subject lacks broader research regarding the Brazilian economy, which, being less stable than those markets known as strong, may present different results. This is the aim of the research to compare three cryptocurrencies (Bitcoin, Ethereum, and Solana) with the Brazilian stock market by means of the non-parametric statistical test Kolmogorov-Smirnov.

Keywords*—Bitcoin, Kolmogorov-Smirnov, Brazilian Stock Market, Cryptocurrencies*

I. Introduction

According to [2012], blockchain technology has become popular in recent years due to its potential applications in various areas. It offers the advantage of being a decentralized method, what allows transactions to be made directly between parties, eliminating central banks or intermediaries, and this method ensures data integrity, anonymity, and immutability. Among these applications, digital currencies or cryptocurrencies have gained massive popularity due to their market values (volatility), ongoing regulatory efforts by some countries, and the proliferation of new currencies.

These digital currencies are encrypted, operate on a peer-to-peer network to facilitate digital exchange, and were developed in 2008, and propose a digital revolution in the payment system, as stated by [5]. Transactions can be completed in minutes, which aid in emergency responses, for example.

Bitcoin, the most well-known and first-created cryptocurrency, proposes a shift from a centralized to a decentralized payment system, with no backing from any central bank. This eliminates territorial barriers and transaction fees, allowing people without bank accounts to conduct transactions with just a mobile device and internet connection.

Indeed, Ethereum allowed the idea of Bitcoin to be extended to other sectors of the economy through the creation of smart contracts, making it the second largest cryptocurrency today [23]. Smart contracts consist of a series of rules that run on the blockchain [24], and through them, it is possible to reduce intermediaries and bureaucracy, as they allow the execution of contracts that were previously done physically, in a digital form, which ensures transparency and immutability. Some areas where these smart contracts have already been successfully applied include: healthcare, the Internet of Things, the insurance industry, notary and registry offices, the financial system, reduction in operacional costs, among others [24, 25], what allows for significant gains in sustainability and efficiency is the monitoring of data, as it enables the reduction of operational costs, minimizes environmental impact, and fosters the development of innovative applications and services [25].

On the other hand, this rapid growth has created some scalability issues, such as the transaction execution time, the block size limit of transactions that can be created, a potential increase in transaction fees, and the increasing complexity of mining as the number of transactions grows, which leads to a higher demand for resources and specialized hardware for processing. To address some of these problems, Solana was launched in 2018, and compared to older cryptocurrencies and its short period of existence, it is already the fourth largest cryptocurrency in the world, with great potential for growth and appreciation in the coming years [22].

As they are not a physical product, their value are generated as users engage in various transactions, such as trading or store of value. Examples include the situation in Argentina when the population faced limitations on converting currency to dollars [9], or during the Brexit vote for the exit of the UK from the European Union [3]. This ease of exchange, without the need to visit authorized agents or research exchange rates, coupled with the ability to use digital currencies online, makes them a faster and more agile solution [5].

Research has been conducted on whether Bitcoin correlates with other indices or currencies. For instance, [10] investigate correlations between Bitcoin value fluctuations

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

and the indices of G7 (Germany, Canada, France, Italy, United States, Japan, and United Kingdom) and BRICS (Brazil, Russia, India, China, and South Africa) stock exchanges. [15] examine cryptocurrency efficiency by creating an index of the 30 largest digital currencies and comparing it with the American Dow Jones index. [14] study risk propagation between the Bitcoin market, crude oil, and six other traditional markets (American stocks, Chinese currency, US Treasury bonds, gold, bonds, and US currency).

Thus, the general objective of this study is to compare Bitcoin, Ethereum and Solana volatility and correlation with the Brazilian stock market and its local currency, the real, by means of the Kolmogorov-Smirnov non-parametric statistical test, a statistical method where the data or population lacks characteristic structures or parameters.

This research is relevant because most existing studies compare cryptocurrencies with already strong and established economies, while the Brazilian economy, like that of many other countries, it is still under development, and it faces internal issues such as corruption, social problems, and low education levels in many regions, which are not commonly present in already developed countries. Consequently, the country tends to feel changes in the macro and microeconomic scenario more intensely, bringing greater volatility to the local currency and stock market. Cryptocurrencies could represent an opportunity for store of value during times of high instability if they were to demonstrate greater stability. Additionally, few studies analyze cryptocurrency market behavior in relation to other parts of the economy.

One key difference between comparing the cryptocurrency market with the traditional stock market is that the traditional market operates within a specific schedule and operating days, as well as having regulatory bodies and central banks, some of the major traditional stock markets are: New York Stock Exchange (NYSE), Nasdaq, Shanghai Stock Exchange, EuroNext, Japan Exchange Group, Shenzhen, Hong Kong, Bombay Stock Exchange, London Stock Exchange, and Toronto Stock Exchange. On the other hand, cryptocurrencies can trade 24/7 and there are no regulatory bodies or central banks. This continuous operation leads to greater volatility with regard to events, which may be reflected in quotes on days when the traditional market is closed.

The methodology used is exploratory, where data extraction from the Ibovespa is performed through the Python library yfinance, and its grouping with the data of the cryptocurrencies Bitcoin, Ethereum, and Solana. It is necessary to preprocess these data, as they do not have opening and closing times, making it possible to trade every day, unlike stock markets which have specific days and times for trading. After grouping and filtering the data to only include those with movements on the same days, the Kolmogorov-Smirnov test is performed for each cryptocurrency in relation to the Brazilian index Ibovespa.

The choice of cryptocurrencies was made considering Bitcoin as the largest and most famous, Ethereum as the second largest, and Solana due to its rapid growth and future potential.

II. Literature Review

The theoretical framework addresses the particularities of the Brazilian stock market, Bitcoin, Ethereum e Solana and probability distributions using the Kolmogorov-Smirnov (KS) test. Thus, the section on the Brazilian stock market addresses its growth since its inception, making a comparison with the cryptocurrencies discussed. The parts related to each cryptocurrency cover their particularities, creation, and purpose. Finally, the part about the KS test explains its functioning and usage.

A. Brazilian Stock Market

The Brazilian stock market, also known as B3 (B3 Brasil Bolsa Balcão S.A.), is the main financial exchange in Brazil. The establishment of the stock market and shares in Brazil dates back to 1817, as referenced in [17], and in the 1990s, there were several exchanges in the country, gradually unified into a single one to facilitate transactions and regulations. Today, it counts with more than 400 listed companies, that represents various sectors of the economy such as finance, education, healthcare, agribusiness, among others.

It plays a crucial role in the economy of the country, facilitating the trading of stocks, commodities, and other financial instruments. To understand the dynamics, regulations, and trends of the Brazilian stock market is essential for evaluating its performance and interactions with other financial assets [17].

[8] mentions that economic development is fundamental for the growth of any country, as it creates liquidity and enables the financing of companies and businesses. According to [4], the capital market in Brazil experienced expansion in the 1990s, and the number of investors has been growing every year due to the ease of investing, reduced brokerage fees, and the possibility of higher gains compared to savings accounts, for example. In 2023, the number of investors in B3 reached 19 million, an increase of 46% compared to 2021 [16], with daily transactions amounting to approximately R$ 36.981 billion in January [22], which demonstrates a significant year-on-year increase in transactions due to the influx of new investors. To facilitate investment in a basket of assets and also to track daily trading volume in the Brazilian market, an index called Ibovespa (Ibov) was created, which currently consists

of the 91 main Brazilian stocks, used to demonstrate the overall market volatility. There are some rules for companies to be included, such as transaction volume, market value, level of corporate governance, and each one has a weight, with the index being updated from time to time.

According to [20], the Ibovespa is calculated in real-time and represents approximately 80% of the trading volume on the Brazilian stock exchange, which reflects not only the daily fluctuations in buying and selling of stocks, but also reflects the local macroeconomic and political scenario.

In order to compare with the proposed cryptocurrencies, Table I shows how much the Ibovespa index has risen since its inception, as well as Bitcoin, Ethereum, and Solana, where it can be observed that Ibovespa took more than 50 years to double in value, whereas the cryptocurrencies, in a much

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

shorter time, have doubled their value by approximately 50 times, as is the case with Bitcoin.

TABLE I. % APPRECIATION OF ASSETS SINCE THEIR INCEPTION

	Year of Creation	*%*
Ibovespa	1968	+200.59
Bitcoin	2008	+30,684.82
Ethereum	2013	+52,403.27
Solana	2018	+479,40

B. Bitcoin

According to [10], Bitcoin is a liberal decentralized financial system that allows financial transactions to be carried out without the intermediary of banks, brokers or regulatory entities such as the Central Bank. It was created by [11], a name that to this day is unknown whether it belongs to a company, a person or a group of programmers, who presented the idea as a payment system based on cryptography where transactions would be made based on the trust of network nodes.

Its value is then based on the number of available coins (which are created as transactions and blocks are made), within a finite limit, and on the digital transactions that are being executed. This is carried out within a blockchain network, similar to a ledger, keeping all transactions transparent and immutable, according to [1].

Since then, several other currencies have emerged, with Bitcoin having established itself as the largest, with the highest volume of transactions and which has already reached the highest market value, above 65 thousand dollars in 2021.

C. Ethereum

Ethereum was created in 2015 and is considered the second largest cryptocurrency in the world, right behind Bitcoin [23]. Its prominence stems from the possibility of creating smart contracts, which allow two or more parties make agreements digitally and without intermediaries, extending the function of Bitcoin to other sectors.

It also enables the creation of other decentralized applications (dApps), and its transaction completion time is much shorter than with Bitcoin, which takes about 10 minutes, while Ethereum takes about 20 seconds.

On the other hand, it faces scalability issues and often charges high fees during periods of high demand, a problem that has been investigated for the launch of future versions [23].

D. Solana

According to [21], Solana is a public blockchain platform that was launched in April 2018, aiming to increase scalability compared to other cryptocurrencies without compromising their security and decentralization. Like Ethereum, it supports smart contracts. Unlike Ethereum, smart contracts on Solana can be written in any programming language, which also contributes to its rapid growth.

Thus, despite being created relatively recently (compared to other cryptocurrencies), it is seen as the fourth largest cryptocurrency with a great potential for appreciation, having appreciated by more than 34% in just one week [22]. Therefore, this cryptocurrency was chosen for analysis to determine if the launch time has any influence on the proposed tests.

E. Probability Distribution

According to [13], a probability distribution can be understood as a function that indicates the possibility of different events occurring within a set of observations, and it can be either discrete or continuous. Discrete distributions can be counted, while continuous distributions occur within a certain range and cannot be presented in a tabular form.

They can also be of the normal or non-parametric type, with normal distributions generally having a bell-shaped curve and being more commonly found in nature. As [13] state, they are typically defined by a mean and a standard deviation. On the other hand, non-parametric distributions are often encountered, for example, in the financial market, such as the application of the Kolmogorov-Smirnov test.

Based on this, the correct identification of distributions allows for the selection of the best analysis according to the objective of the study, as applying the wrong method to a data set can yield unsatisfactory and unreliable results.

Probability distributions are also used to try to predict asset prices using time series and linear equations, as well as for portfolio modeling and decision-making, where data mining or artificial intelligence techniques can also be employed.

F. Kolmogorov-Smirnov Test (KS)

The Kolmogorov-Smirnov test (or KS test, named after the Russian mathematicians Andrei Kolmogorov and Nikolai Smirnov) is used to test the equality of probability distributions, being employed for comparison of 2 samples (bivariate) or of a sample with a reference value (univariate) [6].

Thus, the objective of the test is to quantify the distance between the distributions, with the null hypothesis (H0) being that the sample is drawn from the distribution, in the case of the univariate, or that both are part of the same distribution, in the case of the bivariate [6]. This test can be applied in various software packages or by developing routines by means of programming languages, such as Python, where the test can be applied using the scipy.stats.kstest function[1].

The choice to use the KS test was made because for samples with a size equal to or greater than 30, it is advisable to use the KS test, whereas the Shapiro-Wilk test, for example, is recommended to use with smaller data dimensions, as referenced in [18].

The formula to calculate the KS is:

$$D = \max|Fn(x) - F(x)|$$

Where:

- D is the value of the test statistic,
- Fn(x) is the empirical distribution function of the sample,

1 https://docs.scipy.org/doc/scipy/reference/generated/scipy.tats.kstest.html

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

- F(x) is the theoretical distribution function, usually the CDF (Cumulative Distribution Function) of the distribution being tested.

The value of D is compared with the table of critical values to determine if there is a sufficient evidence to reject the null hypothesis that the two samples come from the same distribution.

In practice, the KS test allows us to verify if the volatility of the Ibovespa has any similarity with the volatility of Bitcoin, Ethereum, or Solana, which enables investors to make better-informed decisions regarding the allocation of their assets.

III. Similar works

Among some of the similar works found, the highlight is on the comparison of cryptocurrencies, typically Bitcoin, with some other index or indicator, such as the dollar, stock market indices like the Dow Jones, as seen in the aforementioned works by [15] and [10].

In addition to these studies, [14] examined the risk propagation among the Bitcoin market, crude oil, and six other traditional markets (US stocks, Chinese currency, US Treasury bonds, gold, bonds, and US currency) between 2019 and 2020, a period that also included the Covid-19 pandemic. Among other methods, they used the Kolmogorov-Smirnov test, and the authors found that during this period, the risk of all markets increased, suggesting caution to investors during times of uncertainty.

[2] and [7] compared the correlation between cryptocurrencies and different currencies such as the dollar, euro, yen, pound, among others. They concluded that the correlation between the assets is practically zero and that there is no dependence between the groups.

The Kolmogorov-Smirnov test has also been used in the verification of criminal transactions, as seen in the work of [19], where the Kolmogorov-Smirnov test, Anderson-Darling test, and Crame-von Mises criterion were used to verify if transactions on the Bitcoin blockchain network originate from illegal sources. The BABD-13 database was used to identify these addresses and serve as a test point. Of the three applications, the Kolmogorov-Smirnov test had the best result in detecting illegal addresses, while the Anderson-Darling test performed better in detecting legal addresses.

These studies are relevant as they allow us to see other comparisons that have already been made and by what method, besides enabling a better understanding of cryptocurrencies and how they relate to the traditional stock and exchange markets. Moreover, knowing the correlation or lack thereof between these means may enable investors to choose investments with lower risk during times of political or economic instability.

Furthermore, it can be seen that the Kolmogorov-Smirnov test has been considered relevant by other authors in comparing data that is non-parametric, such as those produced by the fluctuation of asset and currency values.

IV. Methodology

The methodology used is exploratory, where data acquisition and pre-processing were performed, followed by the bivariate application of the KS test. All development was done using the Python programming language and the libraries numpy [2], pandas [3], scipy [4], datetime [5], matplotlib [6], and yfinance[7]. The numpy library handles large data in formats such as dataframes and arrays, matplotlib is used for generating graphs, datetime was used to convert the timestamp to a readable format, pandas is responsible for reading data from text files, scipy has the implementation of the KS method, and yfinance was used to obtain data for the Brazilian index Ibovespa and other assets.

The data preparation was done independently for each cryptocurrency with which we worked, as their creation dates are different. They needed to be prepared to have the same dates as the Ibovespa database.

The Ibovespa data was obtained via the yfinance library from the Yahoo Finance website, representing the official quotes of the index, and the choice of this method of obtaining data was due to the site already having an API that easily provides the information. This eliminates the need to create a webcrawler for the official pages of the Brazilian stock exchange B3, as the API already returns the following data: Date, Open, High, Low, Close, Volume, and Adj Close. From the information obtained, only the adjusted closing value (Aj Close), which represents the closing value of the asset on the day, was used, and the other values were discarded.

Since the quotes presented by Yahoo Finance are identical to the official quotes, there are no null or blank values. Therefore, no value from the Ibovespa needed to be discarded, grouped, or treated.

A. Bitcoin and Ibovespa

Data collection was carried out in 2 stages. First, historical Bitcoin data was obtained, followed by the Ibovespa index data for the same period, as mentioned above. Bitcoin data was obtained directly from the Kaggle website[8] (which is a site that has various databases already compiled in csv format, therefore, it was not necessary to manually acquire the data from any cryptocurrency exchange), which already has the compiled historical database, covering the period from Jan 2012 to May 2021, with minute-by-minute updated data. The dataset includes Timestamp (Unix time), Open, High, Low, Close, and Volume, with some values as NaN, indicating a possible API failure in capturing the data at that moment. In total, 4,857,377 data points were obtained, with null (NaN) values disregarded, leaving 3,613,769 data points.

Since the data was obtained from Kaggle, in this case no additional cleaning was required other than the exclusion of

2 https://numpy.org/
3 https://pandas.pydata.org/
4 https://scipy.org/
5 https://docs.python.org/3/library/datetime.html
6 https://matplotlib.org/
7 https://pypi.org/project/yfinance/
8 https://www.kaggle.com/datasets/mczielinski/bitcoin-historical-data

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

null values, which is the main advantage of using the dataset provided by the site.

The Ibovespa data was obtained via the yfinance library from the Yahoo Finance website, considering the same period, totaling 2,263 data points, and only the date and adjusted closing value (Adj Close) columns were kept, discarding the others.

Due to a much larger amount of Bitcoin price data, as it represents minute-by-minute asset acquisition, it was necessary to aggregate values by date and keep the value of the median daily quotation to normalize the dataset with the same pattern as the Ibovespa index, resulting in 3,376 data points. The difference with the Ibovespa is that Bitcoin operates every day of the week, 24 hours a day, while the Brazilian stock market operates only on business days during a certain period (usually from 10 a.m to 5 p.m), not trading on weekends, holidays, and overnight.

To solve this problem, the two datasets were merged, considering only the days when both had quotation values, resulting in a total of 2,260 data points as final population to the follow tests. However, when generating the initial graph, a significant interval gap between the assets was observed because the Ibovespa data is in Brazilian real, while Bitcoin price is linked to the dollar.

To solve this problem, the Brazilian real versus dollar exchange rate data was obtained through the yfinance library for the same period mentioned, and its median was calculated. All quotation values were then multiplied by the obtained median value to approximate the Ibovespa value to that of the dollar within the proposed period. The preliminary graph with the values can be viewed in Figure 1, where the green line corresponds to the Bitcoin value, the orange line to the adjusted Ibovespa, and the blue line to the Ibovespa in real.

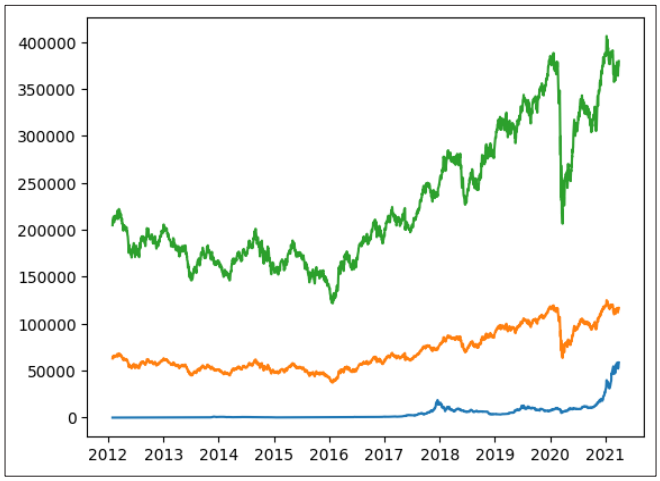


Fig. 1. Generation of the Bitcoin x Ibovespa x Adjusted Ibovespa quotation chart. Legend: Green: corresponds to the value of Bitcoin; Orange: corresponds to the value of the adjusted Ibovespa; Blue: corresponds to the value of the Ibovespa in real terms.

B. *Ethereum and Ibovespa*

The Ethereum data was obtained from the Kaggle[9] platform, which already has it in compiled csv format. Upon downloading the database, it comes in 3 files: one with daily movements, one with minute-by-minute movements, and one with hourly movements, that covers the period from May 9, 2016, to April 15, 2020. We opted to work with the daily data, resulting in 1,438 rows and 8 attributes: Date, Symbol, Open, High, Low, Close, Volume ETH, and Volume USD.

We removed all columns except for Date and Close, which represent the daily closing value of the asset. The dataset contains no null values, leaving us with 1,438 records after this initial preprocessing step.

Since the data was obtained also from Kaggle, in this case no additional cleaning was required, and the dataset did not have null values, so no data treatment was necessary.

The steps for obtaining the Ibovespa data are the same as described above, with only the collection period changed to start from May 9, 2016, to April 15, 2020. The acquisition also considered the adjusted Ibovespa base and in Brazilian real.

After unifying the databases, considering common days across all databases, we were left with a population of 975 data points, as shown in Figure 2. The green line corresponds to the Ethereum value, the orange line to the adjusted Ibovespa, and the blue line to the Ibovespa in Brazilian real.

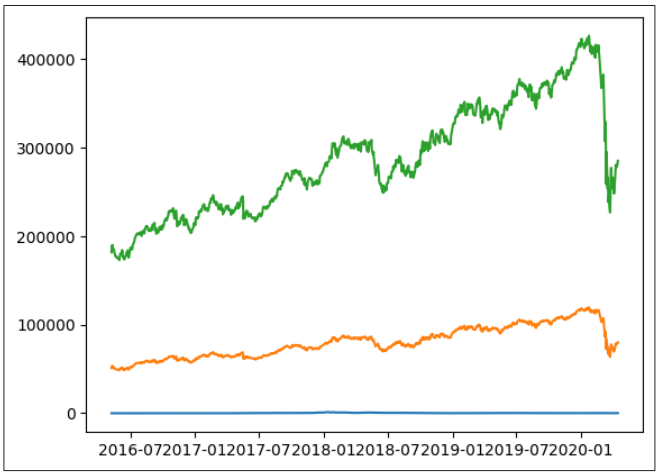


Fig. 2. Generation of the Ethereum x Ibovespa x Adjusted Ibovespa quotation chart. Legend: Green: corresponds to the value of Ethereum; Orange: corresponds to the value of the adjusted Ibovespa; Blue: corresponds to the value of the Ibovespa in real terms.

C. *Solana and Ibovespa*

For Solana, a database provided also by Kaggle[10] was also utilized, resulting in 1,402 data points spanning from April 17, 2020, to February 17, 2024, compiled on a daily basis. The dataset contains the following information: Date, Open, High, Low, Close, Adj Close, and Volume. Only the Adj Close and Date columns were retained, and as with the Ethereum dataset, there were no null values and no additional data treatment was necessary.

For the Ibovespa, the same steps of acquisition mentioned previously were followed, considering the same period as the Solana data. After merging the dates present in the databases, 952 data points remained, and the preliminary result is presented in Figure 3. The green line corresponds to the value

9 https://www.kaggle.com/datasets/prasoonkottarathil/ethereum-historical-dataset

10 https://www.kaggle.com/datasets/ahmadalijamali/cryptourrenciesprices

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

of Solana, the orange line to the adjusted Ibovespa, and the blue line to the Ibovespa in Brazilian real.

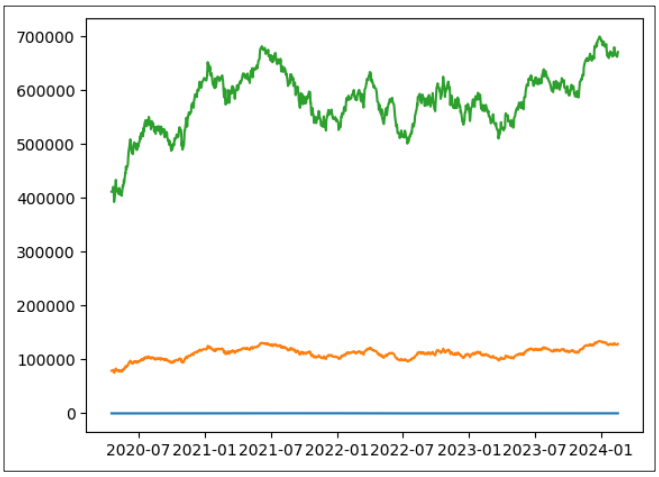


Fig. 3. Generation of the Solana x Ibovespa x Adjusted Ibovespa quotation chart. Legend: Green: corresponds to the value of Solana; Orange: corresponds to the value of the adjusted Ibovespa; Blue: corresponds to the value of the Ibovespa in real terms.

Finally, the KS test was performed for all sets of data, Bitcoin and Ibovespa, Ethereum and Ibovespa and Solana and Ibovespa, considering Ibovespa in real terms and for the adjusted Ibovespa. We use the ktest function from the SciPy library in the Python language.

This test was chosen because the data does not follow a normal distribution, as it exhibits a distribution different from the bell curve. Therefore, solely obtaining means or standard deviations may not be entirely effective in interpreting the information. Thus, the Kolmogorov–Smirnov test is used to compare two samples with each other to verify their equality, which in our case implies they have similar volatility. We did not consider using the Shapiro-Wilk test because it is typically used for smaller datasets.

Finally, the significance test allows for a decision to be made between two or more hypotheses, as it indicates the probability of rejecting the null hypothesis when it is true, considering a p-value of 0.05.

V. *Results*

Just like in the methodology, the results are separated by the sets of databases worked on: Bitcoin Ethereum and Ibovespa, all being compared with the Ibovespa index.

The static value corresponds to the percentage value of the KS test, the static location corresponds to the distance between the empirical distribution function and the measure in the observation, and the p-value is the probability of the value being less than 5%, indicating that the variables have a probability of being part of the same model, contributing to its solution.

A. *Bitcoin*

Thus, Table 2 summarizes the results considering the null hypothesis that the distributions are equal, Table 4 shows the results where the Bitcoin distribution is greater, and Table 3 where the null hypothesis states that the Bitcoin distribution is smaller.

TABLE II. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTIONS ARE EQUAL.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	58901.8
Ibovespa in brazilian real	98.23	0.0	37393.49

TABLE III. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS SMALLER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	0	1.0	406412.696
Ibovespa in brazilian real	0	1.0	125077.0

TABLE IV. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS GREATER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	58901.8
Ibovespa in brazilian real	98.230	0.0	37393.49

Observing Table 3, it is noted that the p-value is equal to 1, meaning it is equal to the level of significance, where the probability of any element from the sample participating and impacting the model is low, thus lying outside the confidence interval, as it neither impacts nor contributes to the model.

On the other hand, the results from Tables 2 and 4 are identical, demonstrating that Bitcoin may have a distribution greater than or equal to that of the Ibovespa, both in its adjusted version and in Brazilian currency (real), indicating that the study holds a valid significance.

B. *Ethereum*

Table 5 summarizes the results considering the null hypothesis that the distributions are equal, Table 7 shows the results where the Ethereum distribution is greater, and Table 6 shows the results under the null hypothesis that the Ethereum distribution is smaller.

TABLE V. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTIONS ARE EQUAL.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	1292.25
Ibovespa in brazilian real	100	0.0	1292.25

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

TABLE VI. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS SMALLER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	0	1.0	119528.0
Ibovespa in brazilian real	0	1.0	426021.68

TABLE VII. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS GREATER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	1292.25
Ibovespa in brazilian real	100	0.0	1292.25

Similarly to the experiment involving Bitcoin and the Ibovespa index, Table 6 shows a p-value equal to 1, demonstrating that the probability of its participation and impact on the model is low. Also similar to the previous results, Tables 5 and 7 demonstrate that Ethereum has a distribution greater than or equal to that of the Ibovespa, both in its Brazilian real form and in the adjusted form, indicating that the study has a valid degree of significance.

C. Solana and Ibovespa

Table 8 summarizes the results considering the null hypothesis that the distributions are equal, Table 10 presents the results where the Solana distribution is greater, and Table 9 shows where the null hypothesis states that the Solana distribution is smaller.

TABLE VIII. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTIONS ARE EQUAL.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	248.46
Ibovespa in brazilian real	100	0.0	248.46

TABLE IX. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS SMALLER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	0	1.0	134194.0
Ibovespa in brazilian real	0	1.0	698466.32

TABLE X. SUMMARY OF RESULTS CONSIDERING THE NULL HYPOTHESIS THAT THE DISTRIBUTION OF BITCOIN IS GREATER THAN THAT OF THE IBOVESPA.

	Static (%)	*pvalue*	*Static Location*
Adjusted Ibovespa	100	0.0	248.46
Ibovespa in brazilian real	100	0.0	248.46

Similarly to the previous studies involving Bitcoin and Ibovespa, and Ethereum and Ibovespa, the result shows that Solana has a greater distribution than or equal to that of the Ibovespa, both in its Brazilian real form and in the adjusted form, with a valid significance (Tables 8 and 10). Meanwhile, in Table 9, the p-value equal to 1 indicates that the contribution of the data to the model is low or of little importance.

This means that the distributions are weakly correlated, which from a practical standpoint, means that if the Ibovespa index is experiencing internal or external pressures and declining, Bitcoin, Ethereum, or Solana could be an option to avoid asset loss or be used as a store of value. Since they are weakly related, this downward volatility would not necessarily impact cryptocurrencies.

Conversely, if Bitcoin, Ethereum, or Solana were experiencing downward volatility due to possible regulation or bans, the Ibovespa index would not necessarily be affected for the same reasons, potentially being used strategically to maintain invested capital with lower risk.

Therefore, knowing whether assets of different types have any correlation can be important for investors to make good decisions not only for profit but also to protect their capital, especially in times of great economic or political instability, such as during wars, uncertainties, or pandemics.

Additionally, Brazil being a developing country may experience events from external sources with varying degrees of intensity, especially significant events like Brexit or the Russia-Ukraine war. This could be considered a positive point as it puts the country outside the radar of macroeconomic uncertainties, that makes it a lower-risk investment possibility in some scenarios, unlike cryptocurrencies, which, is traded globally 24/7, may experience higher volatility during periods of uncertainty.

VI. *CONCLUSIONS*

Blockchain technology has become quite popular, and one of its most well-known applications is Bitcoin, a decentralized virtual currency that ensures transparency and integrity of transactions. As a counterpoint to this popularization, its volatility tends to be high in various periods.

Based on this, the study sought to understand the distance of its curve with the Brazilian stock market, represented by the Ibovespa index, which comprises the main Brazilian stocks. The KS test was then applied under 3 null hypotheses for 3 cryptocurrencies (Bitcoin, Ethereum and Solana): that the distribution of the cryptocurrency is equal to that of the Ibovespa, smaller or larger, considering both the index adjusted in dollars and in Brazilian real.

As final results, the focus of the work is on the distance that the curves represent through the KS test, that is, the greater the distance, the greater the dispersion of the data, leading to the interpretation that they are weakly correlated, or the adherence of one may not influence the other, as the market would like (or desire) it to follow (in the trend of value), ending up with lower (or higher) market values. This is because the calculated value (p-value) is less than 0.05 or 5%, demonstrating that the null hypothesis that cryptocurrencies has a similar distribution to that of the Brazilian market or larger is true, which can be interpreted as markets still in development.

The same result occurs for all three analyzed cryptocurrencies: Bitcoin, Ethereum, and Solana. The test considering that they would be smaller than the Brazilian

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.07

market showed to be unlikely, and for the tests taking into account that the Brazilian market would be greater than or equal to them, similar results were obtained.

As a practical result, the fluctuation of one does not necessarily imply the fluctuation of the other, which can allow investors to protect their capital during times of crisis. Given that Brazil is still a developing country, it usually does not have a significant participation in external events such as wars or international political disputes. On the other hand, when the country experiences instabilities, cryptocurrencies can be a good source of profit or store of value, again enabling investors to protect their wealth.

Based on this, the contribution of the study has a social bias, allowing for greater decision-making and risk management by providing a better understanding of the correlation or lack thereof between different assets. And the choice of the KS test was made because it applies to continuous distributions (as is the case with stocks and their values) and its values are more sensitive near the center of the distribution than to the tails.

As future work, we intend to continue the research by applying other tests focused on non-parametric distributions and also focusing on developing markets, but seeking to extend the analysis to explore additional factors, such as global economic indicators, money supply, inflation rates, public perception, confidence in cryptocurrencies, among others, thereby enriching and broadening the analysis, using this study as an initial basis.

REFERENCES

[1] M. Andoni and V. Robu and D. Flynn and S. Abram and D. Geach and D. Jenkins and P. Mccallum and A. Peacock, "Blockchain technology in the energy sector: A systematic review of challenges and opportunities", in Elsevier, 2019.

[2] E. Baumöhl, E, "Are cryptocurrencies connected to forex? A quantile cross-spectral approach", in Finance Research Letters, 2019, doi: https://doi.org/10.1016/j.frl.2018.09.002.

[3] C. Bovaird, "Bitcoin Rollercoaster Rides Brexit As Ether Price Holds Amid DAO Debacle", 2016, doi: http://www.coindesk.com/bitcoin-brexit-ether-price-rollercoaster

[4] A. G. Carvalho, "Ascensão e declínio do mercado de capitais no Brasil - a experiência dos anos 90", in Economia Aplicada in v.4, n.3, 2000.

[5] P. D. DeVries, "An Analysis of Cryptocurrency, Bitcoin, and the Future", in International Journal of Business Management and Commerce, in v.1, n.2, September, 2016.

[6] L. P. L. Fávero and P. P. Belfiore and F. L. Silva and B. L. Chan, "Análise de Dados: modelagem multivariada para tomada de decisões", in Rio de Janeiro: Elsevier, 2009.

[7] W. Kristjanpoller and E. Bouri. "Asymmetric multifractal cross-correlations between the main world currencies and the main cryptocurrencies", in Physica A: Statistical Mechanics and its Applications, 2019, pp. 1057-1071, 2019, doi: https://doi.org/10.1016/j.physa.2019.04.115.

[8] R. Levine and S. Zervos, "Stock markets banks and economic growth", in American Economic Review n. 88, 1998, p. 537-58.

[9] P. Magro, "What Greece can learn from bitcoin adoption in Latin America", 2016, doi: http://www.ibtimes.co.uk/what-greece-can-learn-bitcoin-adoption-latin-america-1511183.

[10] A. M. Mota and L. de M. Queiroz, "Bitcoin x Mercado de ações: uma análise da variação dos índices das bolsas de valores tradicionais diante da maior alta histórica da criptomoeda", in XXIII SEMEAD (Seminários em Administração), 2020, ISSN 2177-3866.

[11] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system, 2008, doi: https://bitcoin.org/en/bitcoin-paper.

[12] K. S. Tanwar and R. E. Parekh, "Blockchain-based electronic healthcare record system for healthcare 4.0 applications", in Journal of Information Security and Applications, 2021.

[13] R. Walpole and R. Myers, "Probability and Statistics for Engineers and Scientists", Macmillan, 2º ed., 1978.

[14] R. Zha and L. Yu and H. Yin, "Dependences and risk spillover effects between Bitcoin, crude oil and other traditional financial markets during the COVID-19 outbreak" in Environmental Science and Pollution Research 30:40737–40751, 2023.

[15] W. Zhang and P. Wang and X. Li and D. Shen, "The inefficiency of cryptocurrency and its cross-correlation with Dow Jones Industrial Average", in Physica A, 2018, pp. 658-670, doi:https://doi.org/10.1016/j.physa.2018.07.032.

[16] J. Kirinata. "Investidores na B3 chegam a 19,1 milhões; número de PF em renda fixa, FIIs e ações cresce." XPI. https://conteudos.xpi.com.br/conteudos-gerais/b3-investidores-renda-fixa-variavel-fiis-dez-2023/. (accessed Mar. 30, 2024).

[17] S. Nyasha and NM. Odhiambo, "The brazilian stock market development: a critical analysis of progress and prospects during the past 50 year", in Risk governance & control: financial markets & institutions, v.3, Issue 3, 2013.

[18] C. M. Martins, "Testes não paramétricos", in Curso de Estatística Computacional – Universidade de Coimbra, http://www.mat.uc.pt/~cmtm/Ecwww/TestesNP.pdf, (accessed Mar. 30, 2024).

[19] A. J. Maheshwari and J. Calles and S. K. Waterton, "Engineering tRNA abundances for synthetic cellular systems", in Nat Commun 14, 4594, 2023, doi: https://doi.org/10.1038/s41467-023-40199-9.

[20] P. H. M. Silveira, "O que move o preço da ação? Um estudo sobre as maiores variações diárias do ibovespa na década de 2010", in universidade federal de minas gerais faculdade de ciências econômicas, 2019, https://repositorio.ufmg.br/bitstream/1843/34940/1/Monografia%20Pedro%20Silveira.pdf, (accessed Mar. 30, 2024).

[21] G. A. Pierro and R. Tonelli, "Can Solana be the Solution to the Blockchain Scalability Problem?", in *IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, Honolulu, HI, USA, 2022, pp. 1219-1226, doi: 10.1109/SANER53432.2022.00144.

[22] M. Pechman, "Solana valoriza 34,5% em uma semana e métricas da rede sustentam potenciais ganhos adicionais", in Cointelegraph, https://br.cointelegraph.com/news/solana-gains-34-5-percent-week-network-metrics-support-further-gains (accessed Mar. 31, 2024).

[23] Portal Infomoney, "Ethereum: ¿como surgiu a segunda criptomoeda mais valiosa do mundo?", in Infomoney. https://www.infomoney.com.br/guias/o-que-e-ethereum/ (accessed Mar. 23, 2024).

[24] Mohanta, Bhabendu and Panda, Soumyashree & Jena, Debasish. (2018). "An Overview of Smart Contract and Use Cases in Blockchain Technology", IEEE, 2018. 10.1109/ICCCNT.2018.8494045.

[25] Rodriguez, Axel and Jipsion, Armando. "Análisis de la operación de unión en redes de sensores inalámbricos". Prisma Tecnológico, 2019. 10. 44-47. 10.33412/pri.v10.1.2173.

AUTHORS

Marjori Klinczak

PhD student in Information Management at UFPR, Master's degree in Applied Computing from UTFPR, holds the following specializations in the field of computer science: Specialization in Software Development in International Markets from UFPR, Postgraduate studies in Internet Law from FAEL, Postgraduate studies in Ethical Hacking and Cyber Security from Vincit, Postgraduate studies in Offensive Security and Cyber Intelligence from Vincit. Bachelor's degree in Internet Systems Development from FAE. Working since 2007 with web and mobile development both on the front-end (HTML, CSS, JavaScript, jQuery, Angular) and on the back-end (PHP, ASP.NET, VB.NET, Solana, Python, C#, Ionic, React, MySQL, PostgreSQL, SQL Server), and since 2012 owns a company in the field of full-stack web and mobile development services (Mosaic Web). Additionally, has been a Professor in programming, artificial intelligence, and data science at Unifatec-PR since 2019. Holds the following international certifications: ISO 27005, Ethical Hacking, LGPD, and ISO/IEC 38507 from Itcerts.

Egon Wildauer

Bachelor in Informatics - Federal University of Paraná, specialist in Computer Science PUC-PR, improvement in Pedagogy PUC-PR, master in Production and Quality Engineering UFSC and doctor in Forestry Engineering, Forest Management - Computational Production Systems (UFPR, with studies at the Albert Ludwig Freiburg Universität, Freiburg – Germany). Bureau Manager at Schlumberger Inc. in the IT area, Coordinator and Director of the Computer Science area and holds a Bachelor's degree in Information Systems from Centro Universitário Campos de Andrade in Curitiba. Was professor at CEPROTEC in Mato Grosso, Sinop-MT. Since 2005 have been professor at UFPR, member of the CPPD; Deputy Coordinator of the Postgraduate Program in Information Management - PPGGI; was AGTI - Technology and Information Governance Advisor at UFPR, Head of the Information Management Department; participates in the Strictu sensu Postgraduate Program, line 2, of the Master's Course in Information Science and Management-PPCGTI-UFPR where he teaches the IoT and Data Analysis discipline; Statistics and Data Analysis. Coordinator of the enGlobe, AWARE, UFPR and Technische Hochschule Ingolstadt - Germany project and Coordinator of the MBA Management in Engineering specialization course at UFPR.

Electricity Energy Demand Prediction Using Computational Intelligence Techniques

ARTICLE HISTORY
Received 06 February 2024
Accepted 19 April 2024
Published 08 July 2024

Bruno da S. Macêdo
Federal University of Lavras
Lavras, Brazil
bruno.macedo2@estudante.ufla.br
ORCID: 0009-0009-4375-8464

Camila Martins Saporetti
Polytechnic Institute, Rio de Janeiro State University
Nova Friburgo, Brazil
camila.saporetti@iprj.uerj.br
ORCID: 0000-0002-8145-7074

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024

Electricity Energy Demand Prediction Using Computational Intelligence Techniques

Bruno da S. Macêdo
Federal University of Lavras
Systems and Automation Engineering Graduate Program
Lavras, Brazil
bruno.macedo2@estudante.ufla.br

Camila Martins Saporetti
Rio de Janeiro State University
Department of Computational Modeling, Polytechnic Institute
Nova Friburgo, Brazil
camila.saporetti@iprj.uerj.br

***Abstract*—** Energy is an important pillar for the economic development of a country. The demand for electricity is something that continues to grow, one of the contributing factors is the emergence of various technological equipment and the consequent use by the population. There are several resources that can be exploited to generate electricity, with hydroelectric power stations being one of the most used resources. As electrical energy cannot be stored, there is a need to estimate its consumption, looking for a way to meet this energy demand. In this context, this study seeks to apply machine learning techniques, using the Grey Wolf Optimization (GWO) meta-heuristic to optimize regression models, to predict the demand for electricity in Brazil, and it aims to estimate how much energy should be produced. For the predictions, the period between the years 2017 to 2022 was used, totaling around 2,190 samples. The methodology involves pre-processing, crossvalidation, parameters optimization and regression. The results show that Random Forest performed well in the experiments carried out, presenting a coefficient of determination (R^2) of 0.8751, Root Mean Squared Error (RMSE) of 0.0554 and Mean Absolute Error (MAE) of 0.0348 in the best model.

Keywords*—Electric Energy, Machine Learning, Meta-Heuristic, Grey Wolf Optimization*

I. Introduction

Energy is a fundamental input in the current economy. The economic and social development of countries is deeply related to the growth and increase in the supply of electrical energy [1]. It is estimated that global electrical energy generation will increase by approximately 77% between 2006 and 2030 [2].

The demand for electrical energy is something that continues to grow. One of the contributing factors is the emergence of various technological equipment and the consequent use by the population. Brazil has several resources that can be explored to generate electrical energy, and one of the most used resources is hydroelectric plants, which are renewable energy sources [1].

The contribution of energy from hydroelectric plants is around 63% in Brazil, being responsible for generating approximately 70% of all energy used in the country. Despite incentives for the use of other energy sources, it is estimated that in the following years at least 50% of the energy consumed will still come from hydroelectric plants [3].

Although Brazil has large sources of renewable energy, the country has problems in terms of electricity supply, and issues to be observed regarding energy-related investment [4]. There is a segment of the Brazilian population that has difficulty accessing electricity, with the majority of problems being in the way energy is distributed.

As electrical energy cannot be stored, its production and consumption must be accurately idealized in order to avoid circumstances of energy insufficiency as well as overproduction. Simultaneously, load and demand projections serve as the foundation for several decisions made in the energy markets, enabling the planning and operation in a way that is secure, clear, effective, and meets industry demands. Thus, one can observe the need to estimate energy consumption, looking for some way to meet this energy demand. In this context, computational tools can assist in the prediction process and when it comes to the use of data, machine learning techniques appear as an alternative. Given the above, this study seeks to apply machine learning methods to predict electricity demand in Brazil. Thus, it will be possible to assist in decision-making for the distribution of electrical energy.

When using machine learning techniques, a very important factor is to define attributes of the methods to maximize performance. To overcome this situation, metaheuristics can be applied to optimize the models, seeking the best parameters to obtain estimates with the lowest error.

The aim of this study is to use energy load data made available by ONS in order to predict demand based on consumption from the previous seven days. To this end, we propose the use of machine learning techniques commonly used in other applications, such as MultiLayer Perceptron (MLP), Random Forest (RF) and Support Vector Machine (SVM), and the metaheuristic Grey Wolf Optimizer (GWO). The results show that the computational methodology developed performs well in prediction and can assist specialists, providing direction for strategic management and it will anticipate future needs, that will serve as a roadmap for the development and execution of strategies. For businesses in

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

the energy sector, selecting the forecasting model and approach for demand prediction is a crucial decision. Companies operating in the energy sector can set their strategic goals and have the opportunity to improve performance based on this study.

The present study is divided into five sections: section 2 presents the research related to this study. Section 3 discusses the study area, as well as the methodology used. In section 4, there is a discussion of the results obtained and, finally, in section 5, the conclusions of this study are presented.

II. RELATED RESEARCHES

Forecasting energy demand in Brazil is a topic that has been studied by many researchers. In these studies, analysis is carried out and it is proposed tools in the context of Data Mining to understand the problem and seek solutions to predict the results.

Ruas *et al*. [5] carried out a study on predicting short-term energy demand in the state of Paraná between 2004 and 2006. Artificial Intelligence methods were used to predict the results, such as Recurrent Artificial Neural Networks (RNNs) and Support Vector Machine (SVM). The SVM algorithm, with 84 days of input, with sub-bands for the forecast, was the one that obtained the best result.

Alves [6] conducted a study on short-term electrical load forecasting, with historical data from periods of 24 and 48 hours forward, from a company in the electrical sector. Multiple Linear Regression (MLR) and Multilayer Perceptron (MLP) algorithms were used. The MLP was the one that achieved the optimal results.

In the research by Drebes [7], the energy demand for a given day was forecast for the Certel Cooperativa Operations Center Company, responsible for the operation of distribution systems, operation of substations and responsible for controlling active demand. The algorithms used were the MLP, Linear Regression (LR) and Random Forest (RF). The LR algorithm was the one that presented really good results.

Schreiber *et al*. [8] made a prediction of the performance of transformers at the State Electricity Distribution Company in the city of Porto Alegre, Rio Grande do Sul. The MLR algorithm was used. The best results showed an average relative error of 0.050 of the real and estimated yield.

In Marcos and Júnior's work [1], machine learning techniques were used to predict electricity consumption in the Northeast region of Brazil, between the years 2004 and 2019. MLP and Convolutional Neural Networks (CNN) were those that obtained the best outcomes.

Oliveira [9] used the GWO meta-heuristic to minimize the objective function total cost of a shell and tube heat exchanger project, which are used to heating and cooling in various applications such as petroleum refineries, chemical processing, among other applications.

In Pizzolato *et al*. [10], the GWO meta-heuristic was used to obtain the optimal configuration of relay actuation and optimize relay time, which allows faults to be identified, locate and alert the operation of an electrical system so that circuit breakers are open, isolating a given defect. Using GWO, it was possible to coordinate the relays, maintaining the adjustments to the protection system.

The papers found do not forecast energy demand for Brazil as a whole, but rather for specific regions, in addition to not using approaches to find the optimal model. The application of machine learning algorithms is very promising and employing meta-heuristics will help to find the best model, making it possible to predict demand with less error.

III. METHODOLOGY

A. Database

The National Electric System Operator (ONS) has diverse information about energy in Brazil. In this study, the variable Energy Load (EL) was used, which indicates the population's demand, that is, how much energy is used.

The database has daily records of the energy load across the country, where this information is separated by regions. As the objective is to analyze the entire country, a sum of information from all regions was carried out to obtain the demand of the Brazilian population as a whole. The period used for predictions is between the years 2017 and 2022, around 8,764 samples.

B. Pre-Processing

There are four attributes available: id_subsistema, nom_subsistema, din_instante and val_cargaenergiamwmed. The id_subsistema attribute contains the initial letter of each region of Brazil. For example, for the North region the representation is N. The nom_subsistema attribute represents the name of the regions of Brazil, being North, South, Southeast and Northeast. Information for the Central-West region is not available on the base. Furthermore, the din_instante attribute indicates a respective date, in the format (YYYY-MM-DD). Finally, the val_cargaenergiamwmed attribute presents the load value in milliwatts (MW).

To predict energy demand, only the variables din_instante and val_cargaenergiamwmed were considered. They were renamed to DATE and EL, respectively. The DATE variable represents a single date and the EL variable represents the sum of energy loads between the North, South, Southeast/Central-West and Northeast regions. After summing up the energy load of the regions, the database had 2,191 samples. Furthermore, a normalization of the EL variable was performed, resulting in values from 0.10 to 0.90. The attributes that refer to the energy load value of each region, as well as those that identify a specific region, were excluded, as the DATE and EL attributes, which contain the sum of the loads between the regions, will be taken into consideration for the analysis. A lag was also created in the database, creating 7 variables: EL1, EL2, EL3, EL4, EL5, EL6 and EL7. EL1 has a charge from the second day of EL and as it contains one less piece of information, this remains as NaN. EL2, from the third day onwards, contains two NaN information and so on. These samples containing NaN were excluded, 7 in total. After exclusion, 2,184 samples were obtained. This way, it will be possible to predict the energy load for the eighth day considering the previous seven. With the creation of these variables, a new base was created, having only the following

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

attributes: DATE, EL, EL1, EL2, EL3, EL4, EL5, EL6 and EL7 and this data was used in this study.

McNemar's statistical test [11] was used to verify the dependence between variables. When applying the test, some factors must be considered, such as the variables are of the same nature; are identical; have the same values; each variable was entered only once in the sample. If the result is less than (0.05) which is the significance level, the null hypothesis (HO) is rejected, that is, there is an association between variables.

C. Cross Validation

Cross Validation (CV) is used to analyze how much a method can generalize across a set of data. CV is widely used in problems whose aim is to make predictions. Using this approach, it is possible to divide the database into training and test sets. The training set is applied to define the parameters that will be used in the model and the test set is to evaluate the model after training the method.

To evaluate the performance of the regression methods, the K-Fold (KF) technique was used [12]. When using KF, k subsets are divided from N samples, where K>1. After separating the subsets, the k-1 subsets are used to train the methods, and the rest of the sets is used to perform them. Thus, at the end of the process, the validation error is calculated. This procedure is repeated K times, using a different test set for each iteration. In order for regression methods to be able to predict future inputs, tests are repeated several times to best train the models.

D. Methods

Random Forest (RF) [13] is a learning algorithm that works as an ensemble. Builds k decision trees on a training data set in k iterations. During each iteration, a set of samples is randomly selected first. To construct a decision tree from this subset, attributes are randomly chosen by the RF. In this case, as the variable used is the EL and the lags, the decision trees are created based on the randomly chosen lags. The McNemar statistical test was applied where the null hypothesis was not rejected, indicating that there is no statistical evidence that there is an association between pairs of variables in a way that allows the use of RF. Each decision tree is constructed by considering independent random subsets of features and samples. The prediction of a new sample is made using an average or median of the individual tree predictions.

SVR is the Support Vector Machine method for regression [14]. The SVR can be linear or non-linear according to the kernel functions employed. Given the data set of points $\{(x_1, y_1), \ldots (x_l, y_l)\}$, where $x_i \in \Re^n$ is a vector of features and $y_i \in \Re^1$ is the vector of target values. It has the parameters > 0 and $C > 0$ and the SVR is formulated as ε the optimization issue in (1).

$$min \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i + C \sum_{i=1}^{l} \xi_i' \qquad (1)$$

subject to

$$w^T \phi(x_i) + b - y_i \leq \varepsilon + \xi_i,$$

$$y_i - w^T \phi(x_i) - b \leq \varepsilon + \xi_i',$$

$$\xi_i, \xi_i' \geq 0, i=1,\ldots,l.$$

The dual form of the optimization problem can be written as (2).

$$min \frac{1}{2} (\alpha - \alpha')^T K(x_i, x_j)(\alpha - \alpha') + \text{¿} \qquad (2)$$

$$+ \sum_{i=1}^{l} (y_i + \varepsilon)(\alpha_i - \alpha_i')$$

subject to

$$e^T (\alpha - \alpha') = 0,$$

$$0 \leq \alpha_i, \alpha_i' \leq C, i=1,\ldots,l.$$

where $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)^T$, and $\phi(.)$ is the kernel function.

Solving (2) allows determining the parameters to build the SVR approximation.

Then, SVR estimates are given by

$$\hat{y}_i = \varepsilon \sum_{i=1}^{l} (\alpha_i - \alpha_i') K(x_i, x) + b.$$

Multilayer Perceptron (MLP) neural networks were developed to solve problems that are non-linearly separable, in which they cannot be separated by a hyperplane. The structure of a MLP is composed of layers of neurons, where the first layer is for data input, followed by one or more hidden layers to process the information, which uses activation functions, and a last output layer to return the result. In the MLP training phase, the neuron weights are updated to minimize errors using the Backpropagation algorithm [15]. Fig. 1 illustrates an MLP neural network, with a tgh activation function, with six and two neurons in the first and second hidden layers, respectively.

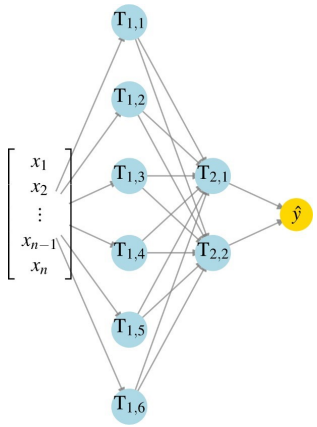


Fig. 1. Architecture of a MLP

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

One of the more actual meta-heuristic swarm intelligence techniques is the Grey Wolf Optimizer (GWO). Because of its remarkable advantages over other swarm intelligence techniques, namely, that it requires no derivation information during the initial search and has very few parameters, it has been extensively used for a wide range of optimization problems. Additionally, it is straightforward, easy to apply, adaptable, scalable, and has the unique ability to balance exploration and exploitation in a way that promotes favorable convergence during the search.

The GWO meta-heuristic is based on the social behavior of grey wolves, seeking to simulate the social hierarchy of wolves in a pack [16]. In the GWO, the inhabitants are separated in alpha (α), beta (β), delta (δ) and omega (ω). Theα wolves that have more capability to survive environment are the first denominated, β and δ that lead other wolves ωα towards promising locations in the search space. During the process, the wolves advance and modify their, β or δα positions as follows:

$$\vec{M}=|\vec{L}\,\vec{X}_p(t)-\vec{X}(t)| \quad (3)$$

$$\vec{X}_{(t+1)}=\vec{X}_p(t)-\vec{J}\,\vec{M} \quad (4)$$

where t is the most recent epoch, $\vec{J}=2a\,\vec{r}_1 a$, $\vec{L}=2\vec{r}_2$, $\vec{X}_p$ is the prey position vector, $\vec{X}$ is the position vector of a grey wolf, $\vec{r}_1$, $\vec{r}_2$ are random vectors in [0,1] and the parameter a decreases linearly from two to zero. The GWO supposes that, β and δ are the assumed (optimal) preyα position. The three best solutions obtained so far are considerate, β and δ respectively. Other wolves are thenα indicated as ω and with the capacity to reposition themselves in relation to, β and δ. The proposed mathematical modelα that reports the location to be readjusted of the ω wolves:

$$\vec{M}_\alpha=\left|\vec{L}_1\vec{X}_\alpha-\vec{X}\right| \quad (5)$$

$$\vec{M}_\beta=\left|\vec{L}_2\vec{X}_\beta-\vec{X}\right| \quad (6)$$

$$\vec{M}_\delta=\left|\vec{L}_3\vec{X}_\delta-\vec{X}\right| \quad (7)$$

where $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$ present the position of, β and δα respectively, $\vec{L}_1$, $\vec{L}_2$, $\vec{L}_3$ are random vectors and the $\vec{X}$ is the position of the current solution.

Equations (5), (6) and (7) calculate the distance between the current solution and, β and δ. The final position of the α actual solution is calculated as follows:

$$\vec{X}_1=\vec{X}_\alpha-\vec{J}_1\vec{M}_\alpha \quad (8)$$

$$\vec{X}_2=\vec{X}_\beta-\vec{J}_2\vec{M}_\beta \quad (9)$$

$$\vec{X}_3=\vec{X}_\delta-\vec{J}_3\vec{M}_\delta \quad (10)$$

$$\vec{X}_{(t+1)}=\frac{\vec{X}_1+\vec{X}_2+\vec{X}_3}{3} \quad (11)$$

where $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$ show the current location of the wolves, β and δ, α $\vec{J}_1$, $\vec{J}_2$, $\vec{J}_3$ are randomly generated vectors and *t is* the number of epochs.

As shown above, (5), (6) and (7) represent the step of the wolf ω toward, β and δ respectively. Equations (8), (9), α (10) and (11) are the final location of the ω wolves. There are two vectors too, as can be seen: $\vec{J}$ and $\vec{L}$.

E. Metrics

The following metrics were used to evaluate performance: Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Coefficient of Determination (R^2).

RMSE calculates the square root of the mean of the difference between the true value and the estimated value for the data set. The calculation of the difference is squared. The higher the RMSE value in the calculation, the worse the model will be [17]. In (12), y_i the true value, $\hat{y}_i$ the estimated value and n the data set number.

$$RMSE=\sqrt{\left(\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\hat{y}_i\right)^2\right)} \quad (12)$$

In MAE, the average difference between the true value and the estimated value for the data set is calculated, but as there may be negative values in the difference, the value in the module is considered. The lower the value obtained in the MAE calculation, the better the predicted results will be [17]. In (13), y_i the true value, $\hat{y}_i$ the estimated value and n the data set number.

$$MAE=\frac{1}{n}\sum_{i=1}^{n}\left|y_i-\hat{y}_i\right| \quad (13)$$

R^2 measures the variance of a model data. The variability value is between 0 and 1. The value obtained in the R^2 calculation indicates that the higher the value, the prediction is closer to what is expected according to the original data [17]. In (14), y_i represents the true value, $\hat{y}_i$ the value to be predicted and $\bar{y}$ the average value for y.

$$R^2=\frac{\sum_{i=1}^{n}\left(y_i-\hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i-\bar{y}\right)^2} \quad (14)$$

IV. RESULTS AND DISCUSSIONS

The computational methodology was implemented using the programming language Python, which is a language used to perform data analysis and which has several libraries with optimized functions [18]. All experiments were run on a computer with the following specifications: Intel(R) Core(TM) i5-1135G7, 8 GB RAM, and Windows 10 operating system. The Scikit-Learn and PyGMO libraries were used. Scikit-Learn is a library that allows you to work with machine learning, it has a set of resources, such as algorithms to perform data analysis, metrics for prediction,

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

among others [19]. PyGMO is a library used to work with optimization problems, it has several optimization algorithms to be used in conjunction with machine learning algorithms for a better performance [20]. As for the library, 30 independent iterations were carried out to evaluate the methodology. A KF with a value of K equal to 5 was used.

Table 1 presents the description of the models used in the GWO meta-heuristic, indicating the method, the parameters for the model to perform well, the description of these parameters and their configuration, indicating the values used during the executions. In MLP, the activation function settings to be used in GWO were represented as follows: 0: Identity, 1: Logistic, 2: Tanh and 3: ReLU, the configuration being [0, 3]. In this configuration, for example, it means that the values will be generated randomly in the range from 0 to 3. The máximum number of GWO generations was 30.

TABLE I. DESCRIPTION OF THE MODELS

Methods	Parameters	Description	Settings
MLP	hidden_layer_sizes	Number of hidden layers and Number of neurons in each layer	[1, 4] and [1, 50]
	activation	Activation Function	0: Identity; 1: Logistic; 2: Tanh; 3: ReLU
RF	n_estimators	Number of trees in the forest	[100, 200]
	max_depth	Maximum depth of the tree	[2, 10]
SVM	C	Adjusts the penalty for regression errors	[20, 200]
	gamma	Defines how far the influence of a single training example extends	[0.001, 0.1]
	epsilon	Sets a limit on insensitivity to errors in predictions	[0.001, 0.1]

Table 2 presents the results obtained using the average and standard deviation, the R^2 metrics of the iterations, RMSE and MAE. Using the GWO meta-heuristic, it was possible to find the ideal parameters to maximize the performance of the regression models. The best model is highlighted in bold.

The model that achieved the best performance was RF, with a R^2 of 0.8704, MAE of 0.0352 and RMSE of 0.0564, thus indicating that it was the method that obtained the best predictions in relation to the EL variable. The SVM also showed good results, with a R^2 of 0.8618, MAE of 0.0353 and RMSE of 0.0583. The closer the value of R^2 is to 1 and the lower the values of MAE and RMSE, the better the model performance.

TABLE II. MODELS PERFORMANCE

Methods	R^2	RMSE	MAE
MLP	0.7982 ± 0.0288	0.0703 ± 0.0047	0.0496 ± 0.0042
RF	**0.8704 ± 0.0016**	**0.0564 ± 0.0003**	**0.0352 ± 0.0003**
SVM	0.8618 ± 0.0005	0.0583 ± 0.0001	0.0353 ± 0.0003

Table 3 shows the optimal MLP, RF and SVM parameters to maximize performance. The best result obtained by RF has a number of trees equal to 130, a maximum tree depth equal to 10, R^2 of 0.8751, RMSE of 0.0554 and MAE of 0.0348. The results show that using ensemble methods generates good results, as the method combines several models.

TABLE III. BEST MODELS

Methods	Parameters	R^2	RMSE	MAE
MLP	hidden_layer_sizes = (47, 49, 48, 44), activation = ReLU	0.8244	0.0656	0.0450
RF	n_estimators: 130, max_depth: 10	0.8751	0.0554	0.0348
SVM	C = 100, gamma = 0.1, epsilon = 0.0229	0.8626	0.0580	0.0353

Fig. 2 presents boxplots for the R^2, RMSE and MAE metrics, showing the performance of the RF, SVM and MLP regression models in predicting the load during the 30 iterations. It can be seen that RF was the model that presented the best results, with the lowest values for RMSE and MAE and the highest for R^2. The MLP, considering the network topology adopted, presented some variations in the prediction of the load variable, resulting in higher values for MAE. For the MLP, some MAE values were not concentrated like the other predictions, thus indicating that for some data the predictions were not correct. Such analysis is appropriate considering the search range for the parameters used and the meta-heuristic employed.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

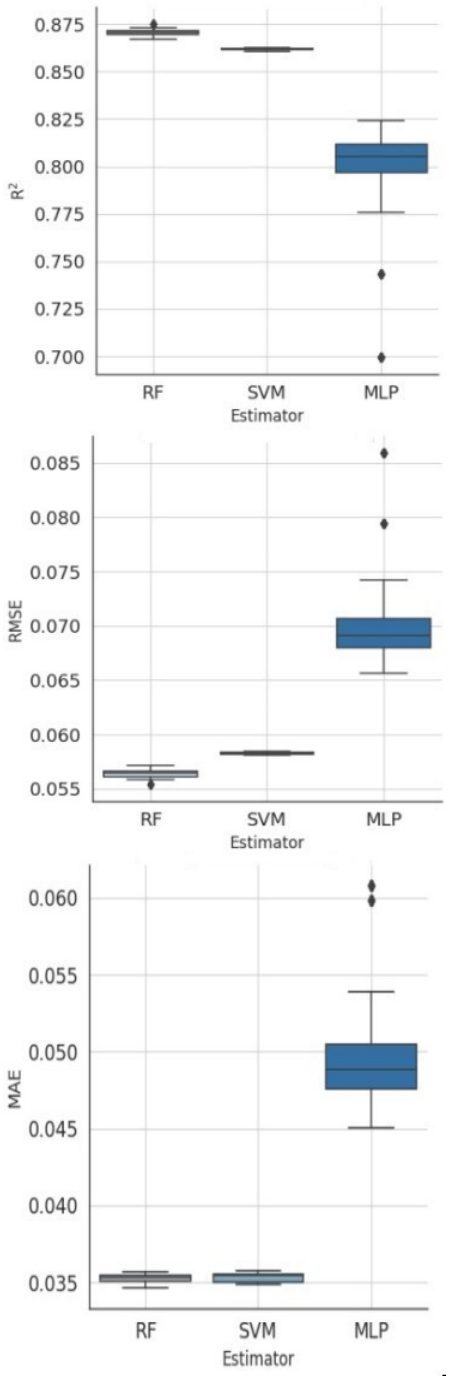


Fig. 2. Boxplot for R^2, RMSE and MAE metrics

Fig. 3 shows a comparison of the true energy load already normalized with that predicted by the best MLP, RF and SVM models in the analyzed period. It can be observed that the error between the predicted and the true is less for the RF comparing to other models. The time series formed by these values have similar behavior. This shows that the RF model has the capacity to assist in the energy load prediction process, which helps to verify population demand.

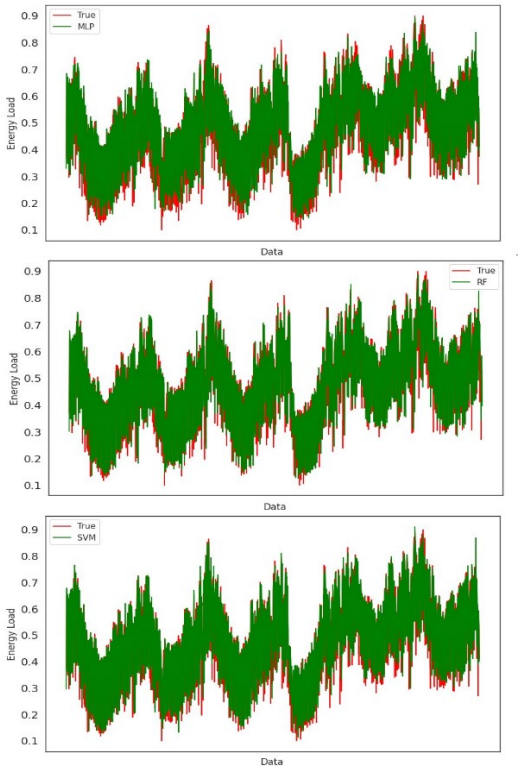


Fig. 3. Energy Load True x Energy Load Predict – MLP, RF and Svm respectively

Fig. 4 illustrates the parameter distribution of the MLP, RF and SVM models. One can see that for MLP the ReLU was the activation func chosen in all runs and 4 hidden layers in the most runs. For RF, the max depth was 10 in all runs and No. estimators around 130 in 8 runs followed by 200 in 5 runs. In the SVM case, C equal 100 in all runs, γ equal 0.1 and ε values were well distributed between 0.001 and 0.0275.

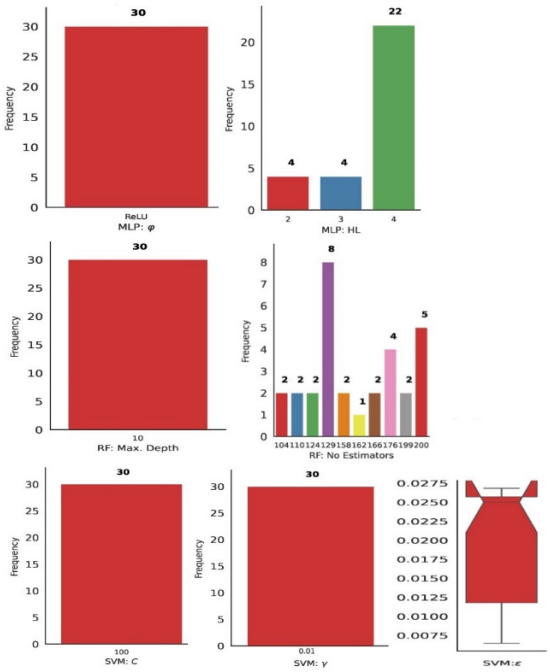


Fig. 4. MLP, RF and SVM parameters distribution

V. CONCLUSION

In this paper, the performance of three regression methods (MLP, RF, SVM) for forecasting electricity demand in Brazil, it was analyzed for the algorithms to have good results, the GWO meta-heuristic was used to improve their performance. The RF and SVM methods showed the best results. However, the RF was the one that presented the best result, with R^2 = 0.8751, RMSE = 0.0554 and MAE = 0.0348 and, which shows that using ensemble methods generates good results, by combining a set of models. For the MLP, some predictions were not correct, due to the fact that some metric values were not concentrated. This analysis was carried out considering the search space for the parameters adopted and the meta-heuristic used.

Future research includes applying deep learning techniques, such as Long Short-Term Memory, to evaluate whether methods considered more robust will perform better in predicting energy demand. In addition, analyzing the insertion of other variables that affect daily energy consumption and that can assist in prediction.

Furthermore, the proposed approach proved to be effective, as it used cross-validation techniques to enable the model ability to generalize data from the tests carried out. Moreover, by using the GWO meta-heuristic, it was possible to search for the best parameters to maximize the performance of the regression models, as well as by the use of an ensemble algorithm that combines multiple models.

REFERENCES

[1] I. P. Marcos and A. P. P. Júnior," Forecast of Electricity Consumption in the Northeast Region of Brazil". Journal of Engineering and Applied Research. v. 6, n. 3, p. 21-30, 2021.

[2] L. C. Morais, Study on the panorama of electrical energy in Brazil and future trends. Dissertation - Electrical Engineering. UNESP. 2015.

[3] B., Stearns, F. Rangel, F. Firmino F, Rangel and J. Oliveira, Predicting performance of enem candidates through socioeconomic data. In: Proceedings of the XXXVI SBC
Scientific Initiation Paper Competition. SBC, 2017.

[4] Energy research company, Dea Technical Note 22/16, Projection of electricity demand for the next 10 years (2016 – 2026). 2016.

[5] G. I. S. Ruas, T. A. C. Bragatto, M. V. Lamar, A. R. Aoki and S. M. Rocco," Forecasting electrical energy demand using artificial neural networks and support vector regression". VI National Artificial Intelligence Meeting. p. 1262-1271, 2007.

[6] M. F. Alves, Forecasting electrical load demand by stepwise variable selection and artificial neural networks. Dissertation Electrical Engineering. UNESP. 2013.

[7] F. Drebes, Electricity demand forecast using artificial intelligence. 2020. Monograph (Graduation in Electrical Engineering) – University of Vale do Taquari - Univates, Lajeado, 03 Dec. 2020.

[8] J. F. Schreiber, I. E. M. Kühne, L. A. Destefani, A. T. Z. R. Sausen, M. Campos and P. S. Sausen, "Intelligent Networks: Data Mining as a Support Tool for the Analysis of Large Volumes of Data in Underground Energy Substations". In: Brazilian Automatic Congress-CBA. 2020.

[9] C. T. Oliveira, Optimization of a shell and tube heat exchanger using the grey wolf algorithm. Dissertation - Mechanical Engineering. Unisinos. 2015.2018.

[10] G. P. Pizzolato, E. M. dos Santos, A. R. Fagundes, J. O. dosSantos and H. Hasselein, Optimizing the Operating Time of Overcurrent Relays Using the Grey Wolf Algorithm. Brazilian Symposium on Electrical Systems, 1(1). 2020.

[11] P. A. Lachenbrunch, McNemar test, Wiley StatsRef: StatisticsReference Online. 2014.

[12] R., Kohavi, "A study of cross-validation and bootstrap foraccuracy estimation and model selection", Ijcai. Vol. 14. No. 2. 1995.

[13] L. Breiman, Random forests. Machine learning, v. 45, p. 5-32,2001.

[14] A. J. Smola, Learning with Kernels, PhD Thesis. TechnicalUniversity of Berlin, 1998.

[15] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998.

[16] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey wolf optimizer".Advances in engineering software, v. 69, p. 46-61, 2014.

[17] B., Stearns, F. Rangel, F. Firmino F, Rangel and J. Oliveira,Predicting performance of enem candidates through socioeconomic data. In: Proceedings of the XXXVI SBC
Scientific Initiation Paper Competition. SBC, 2017.

[18] A. Boschetti and L. Massaron, *Python data science essentials*. Packt Publishing Ltd, 2016.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B.Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrotand and É. Duchesnay, "Scikit-learn: Machine learning in Python". the Journal of machine Learning research, v. 12, p. 2825-2830, 2011.

[20] F. Biscani and D, Izzo, "A parallel global multiobjectiveframework for optimization: pagmo". Journal of Open Source Software, v. 5, n. 53, p. 2338, 2020.

ISSN:1390-9266 e-ISSN:1390-9134 LAJC 2024
DOI: https://doi.org/10.33333/lajc.vol11n2.08

AUTHORS

Bruno da S. Macêdo

Master's student in Systems and Automation Engineering at the Federal University of Lavras. He holds a Bachelor's degree in Computer Engineering from the State University of Minas Gerais and a professional technical degree in Internet Computing from the Federal Center for Technological Education of Minas Gerais (2018). He has developed work in the field of Computational Intelligence.

Camila Martins Saporetti

PhD in Computational Modeling from the Federal University of Juiz de Fora. She also holds a Master's degree in Computational Modeling from the same institution. She earned a Bachelor's degree in Exact Sciences from the Institute of Exact Sciences at the Federal University of Juiz de Fora, as well as degrees in Computational Engineering from the Faculty of Engineering and in Computer Science from the Institute of Exact Sciences at the same university.

She is currently an Associate Professor at the Polytechnic Institute of the State University of Rio de Janeiro and a permanent member of the Graduate Program in Computational Modeling at the Polytechnic Institute. She has experience in the field of Computational Intelligence, working mainly on machine learning and metaheuristics.

LATIN-AMERICAN JOURNAL OF COMPUTING

Published by

Escuela Politécnica Nacional
Facultad de Ingeniería de Sistemas

Quito-Ecuador

https://lajc.epn.edu.ec/
lajc@epn.edu.ec

July 2024

Vol XI, Issue 2, July 2024

LATIN-AMERICAN JOURNAL OF COMPUTING