Contemporary Mathematics
https://ojs.wiserpub.com/index.php/CM/

Research Article

Error Analysis and Model Efficiency for the Jaulent-Miodek System Using Physics-Informed Neural Networks

Ahmad Shafee

PAAET, College of Technological Studies, Laboratory Technology Department, Shuwaikh, 70654, Kuwait
E-mail: as.zada@paaet.edu.kw

Received: 15 June 2025; **Revised:** 15 September 2025; **Accepted:** 19 September 2025

Abstract: This paper focuses on approximate solution of the Jaulent-Miodek nonlinear-coupled partial differential equations. Physics-Informed Neural Network (PINN) architecture is implemented to find an approximate solution against the independent variables. Loss function is defined using PDEs and initial conditions. The PINN is set up to meet both the governing PDEs and the starting conditions. It does this by using automated differentiation to enforce the system's residuals in the loss function. The results are compared with the exact solution using graphical representations. The model is tested for specific values of ψ. The error is illustrated in table, which shows the model efficiency and accuracy.

***Keywords*:** Jaulent-Miodek system, physics-informed neural network, approximate solution

MSC: 65M99, 68T07, 35Q53

1. Introduction

The scientific community is looking into whether neural networks are suitable for computational tasks due to their remarkable success in machine learning tasks like game theory [1], natural language processing [2], and computer vision [3]. They are also exploring the possibility of using specialized hardware, like Apple's Neural Engine and Google's Tensor Processing Units, to efficiently execute neural networks [4, 5]. As a result, there is now a fascinating new area of study called scientific machine learning, where traditional applied mathematics issues are tackled using methods like deep neural networks and statistical learning. Using methods from machine learning and artificial intelligence, in this research, we aim to provide a concise summary of recent developments in the use of numerical solutions to nonlinear and linear Partial Differential Equations (PDEs) [6–10].

By offering strong information processing capabilities for improved assimilation of vast volumes of quantitative and experimental data, Machine Learning (ML) techniques possess the capacity to produce more general and practical constitutive models in recent years. They have the potential to be another paradigm for multiscale bridging because of their computing efficiency, versatility in application, and lack of restriction by particular equation forms [11–15]. In other words, by avoiding tasks associated with compact equation-based visualization of information, machine learning may more effectively use huge data to simulate genuine continuous multiscale physics [16].

There are certain risk factors associated with the concept of using Deep Neural Network (DNN) to obtain numerical solutions of PDEs. The piecewise continuous objective function is a prerequisite for the DNN approximation's

DOI: https://doi.org/10.37256/cm.6620257474

dependability, per the universal approximation theorem [17, 18]. Nevertheless, the regularity criterion is not always met by PDE solutions, especially nonlinear PDEs [19]. As a result, the numerical precision of the solutions derived from the direct DNN prediction is typically insufficient. As an alternative to direct prediction, semi-analytic models are the subject of a large number of published neural network-based studies. To solve PDEs, for instance, Raissi et al. first created the Physics-Informed Neural Network (PINN) [20] by including integral forms into the loss function. The hp-Variational Physics-Informed Neural Network (VPINN), a more accurate form of PINN, was created by Kharazmi et al. [21]. The DNN was used by the author of [22] to address variational issues resulting from PDEs. These methods transform the original PDEs into parametric models and include penalty functions to ensure that the DNN models are consistent with the original PDEs, even if the accuracy of the DNN-based solutions is usually limited.

The automated integration [23] is another technique now in use for computing integrals in DNNs. Using this approach, the author uses the high-order Taylor series expansion of the integrand around a specified point in the integration domain to approximate it. After that, an analytical calculation is made of the integrals. Automatic differentiation, often known as autodiff, is used to calculate the derivatives required for the Taylor series expansion [24]. Overfitting can readily happen since the derivatives' information is local. Adaptive integration techniques provide an additional option for the quadrature rule. The Deep Least Squares (DLS) approach [25], which employs an adaptive mid-point quadrature rule with local error indications, is one example of how writers have chosen these techniques. It is unclear what quadrature error results from these indications, which are based on the residual value at randomly chosen places.

Kovacs et al. suggested conditional physics-informed neural networks [26] for partial differential equation eigenvalue problems. Mao et al. used PINN in their work to estimate the Euler equations [27]. To address multi-scale issues, Wang et al. developed a novel architecture that makes use of spatiotemporal and multi-scale random Fourier features [28].

The evolution equation known as the Jaulent-Miodek (JM) equations [29] has been used in several science domains, including fluid mechanics, enhanced matter physics, and optical phenomena. The JM equation is frequently used to characterize the influence of energy Schrodinger ability [30]. These equation systems are utilized as a model to solve complex problems in many technological and natural science domains (see [31–33]). Because these nonlinear PDEs are so important in the previously listed fields, it is essential to comprehend both the numerical and analytic-approximate solutions. The coupled JM equations are the focus of much mathematical study and are used in many scientific and technological domains, such as condensed matter physics [34, 35] and plasma physics [36]. Consider the following system of JM equations:

$$\frac{\partial}{\partial\psi}\Phi_1(\sigma,\ \psi)+\frac{9}{2}\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi)\frac{\partial^2}{\partial\sigma^2}\Phi_2(\sigma,\ \psi)+\frac{\partial^3}{\partial\sigma^3}\Phi_1(\sigma,\ \psi)+\frac{3\Phi_2(\sigma,\ \psi)}{2}\frac{\partial^3}{\partial\sigma^3}\Phi_2(\sigma,\ \psi)$$

$$-6\Phi_1(\sigma,\ \psi)\Phi_2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi)-6\Phi_1(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_1(\sigma,\ \psi)-\frac{3}{2}\Phi_2^2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_1(\sigma,\ \psi)=0, \tag{1}$$

$$\frac{\partial}{\partial\psi}\Phi_2(\sigma,\ \psi)-6\Phi_1(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi)-6\Phi_2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_1(\sigma,\ \psi)$$

$$-\frac{15}{2}\Phi_2^2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi)+\frac{\partial^3}{\partial\sigma^3}\Phi_2(\sigma,\ \psi)=0.$$

Initial conditions

$$\Phi_1(\sigma,\ 0)=\frac{q^2}{8}\left(1-\operatorname{sech}^2\left(\frac{q\sigma}{2}\right)\right),$$

$$\Phi_2(\sigma,\ 0) = q\ \text{sech}\left(\left(\frac{q\sigma}{2}\right)\right). \tag{2}$$

For the first time, deep learning is used in this research to estimate the solution of the PDE system. Using the PDEs system and the problems' starting values, a loss function is created. Backpropagation is used by the model itself to learn, optimize the value in each iteration, and offer the solution. Next, the solution is contrasted with the PDE system's precise solution. The efficiency of the work offered is thoroughly examined in the outcomes section.

The framework associated with the proposed PINN is not only applied to the Jaulent-Miodek system, but also to a broad range of nonlinear coupled PDEs. Applications comprise the study of liquid flow, non-linear wave propagation, plasma physics, and optical systems, where analytical solutions are seldom obtainable. This demonstrates the wider applicability and flexibility of the approach toward real-world issues involving complex dynamical systems.

2. Methodology

A popular machine-learning approach for approximating ODEs and PDEs is called PINN. For the first time, the study uses PINN to estimate the solution of the PDE system. Two neurons that receive the independent variables σ and ψ make up the PINN input layer. Fully connected layers (hidden layers) are connected to the input layer. Features are extracted from the initial conditions and backpropagation using hidden layers. Equation (3) expresses nonlinearity, which is added using the tanh function as an activation function.

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{3}$$

2.1 *Loss function*

The loss function is constructed from the initial condition loss and the PDE system residual loss:

$$\mathscr{L}(\theta) = \mathscr{L}_{PDEs} + \lambda \mathscr{L}_{IC} \tag{4}$$

where λ is the weighting coefficient balancing the two terms contributions.

$$\mathscr{L}(\theta) = \frac{1}{N_r}\sum_{i=1}^{N_r}\left(\mathscr{R}_1^2 + \mathscr{R}_2^2\right) + \frac{1}{N_0}\sum_{j=1}^{N_0}\left\|\mathscr{N}_\theta(\sigma_j,\ 0) - \begin{bmatrix}\Phi_1(\sigma_j,\ 0)\\ \Phi_2(\sigma_j,\ 0)\end{bmatrix}\right\|^2 \tag{5}$$

where

$$R_1 = \frac{\partial}{\partial\psi}\Phi_1(\sigma,\ \psi) + \frac{9}{2}\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi)\frac{\partial^2}{\partial\sigma^2}\Phi_2(\sigma,\ \psi) + \frac{\partial^3}{\partial\sigma^3}\Phi_1(\sigma,\ \psi) + \frac{3\Phi_2(\sigma,\ \psi)}{2}\frac{\partial^3}{\partial\sigma^3}\Phi_2(\sigma,\ \psi)$$

$$-6\Phi_1(\sigma,\ \psi)\Phi_2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_2(\sigma,\ \psi) - 6\Phi_1(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_1(\sigma,\ \psi) - \frac{3}{2}\Phi_2^2(\sigma,\ \psi)\frac{\partial}{\partial\sigma}\Phi_1(\sigma,\ \psi),$$

$$R_2 = \frac{\partial}{\partial \psi}\Phi_2(\sigma,\ \psi) - 6\Phi_1(\sigma,\ \psi)\frac{\partial}{\partial \sigma}\Phi_2(\sigma,\ \psi) - 6\Phi_2(\sigma,\ \psi)\frac{\partial}{\partial \sigma}\Phi_1(\sigma,\ \psi)$$

$$-\frac{15}{2}\Phi_2^2(\sigma,\ \psi)\frac{\partial}{\partial \sigma}\Phi_2(\sigma,\ \psi) + \frac{\partial^3}{\partial \sigma^3}\Phi_2(\sigma,\ \psi). \tag{6}$$

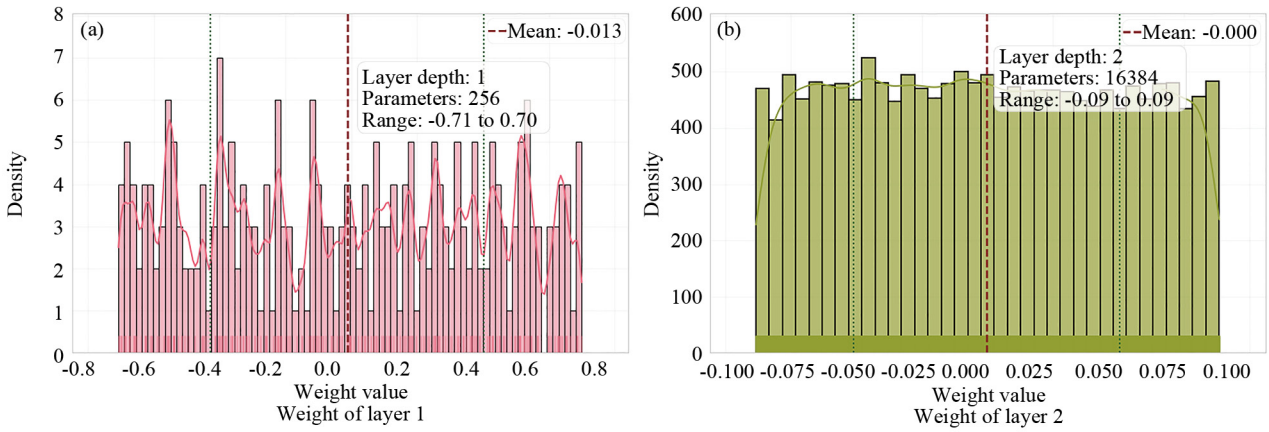
2.2 *Back propagation*

Gradients of the loss function computed with regard to the neural network parameters are obtained by backpropagation. The Adam optimizer minimizes the loss function. The solution is evaluated against the initial condition and compared with the solution of previous iterations to notice the convergence.

3. Results

The presented model is implemented in Python and evaluated on Google Colab. We used the feed forward neural network with 4 hidden layers. The initial layer comprised 256 neurons that were randomly set to initial values of weights between -0.71 and 0.70 so as to provide a wide range of representation at the input layer. The second layer of the network consisted of 128 units, and the 3rd layer comprised of 128 neurons, weight distributions falling within the range of -0.09 to 0.09 and almost equal means, due to which the training was stable, and the approximation symmetrical in the area. The last layer involved one or two neurons, depending on the system variable under study and the weights were in the range of -0.09 to 0.09. The activation function employed in the entire network is hyperbolic tangent (tanh), because of its smoothness to capture the non linear dynamics in stability. Figure 1 shows the weight distributions of hidden layers, showing the histograms shaped like Gaussians. This means that the weights were set up correctly and the learning process is steady.

The convergence behavior of the model is illustrated in Figure 2. The total loss, PDE loss and Initial Condition (IC) loss are decreasing with the epochs, indicating the effectiveness of the model.



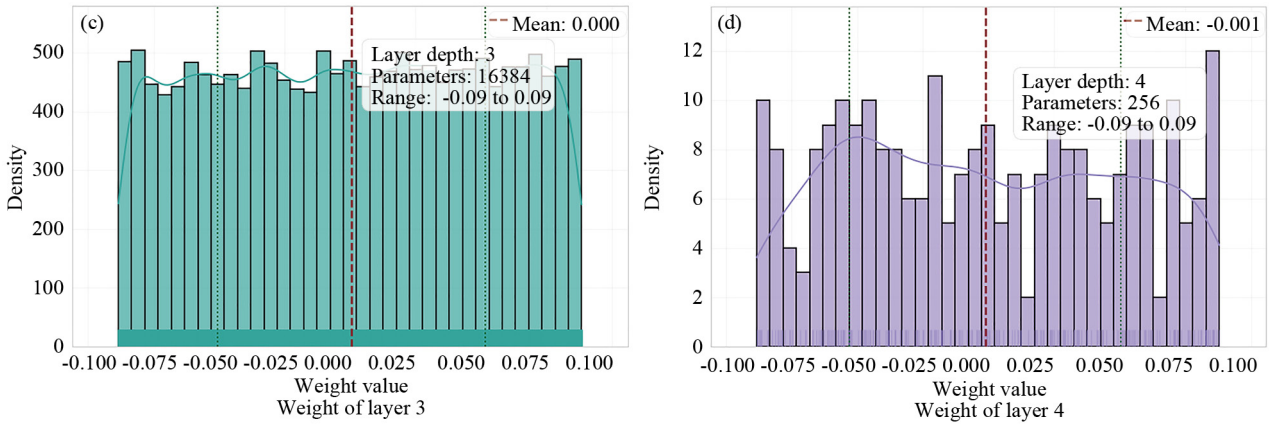


Figure 1. Weight distribution of PINN layers

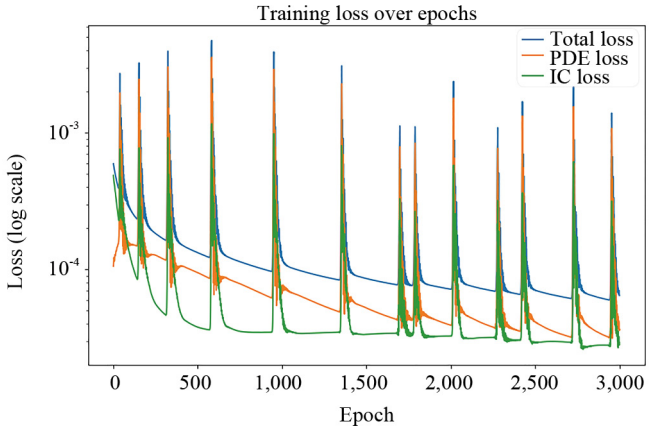


Figure 2. Convergence of the model

Figure 3a-b show the PINN approximation and exact solution for ϕ_1 respectively. Similarly, Figure 3c-d illustrate the comparison of PINN approximation and exact solution for ϕ_2 respectively. The exact solution is given as

$$\Phi_1(\sigma,\ \psi) = \frac{q^2}{8}\left(1 - \operatorname{sech}^2\left(\frac{q}{2}\left(\sigma - \frac{q^2\psi}{2}\right)\right)\right),$$
$$\Phi_2(\sigma,\ \psi) = q\left(1 - \operatorname{sech}\left(\frac{q}{2}\left(\sigma - \frac{q^2\psi}{2}\right)\right)\right). \tag{7}$$

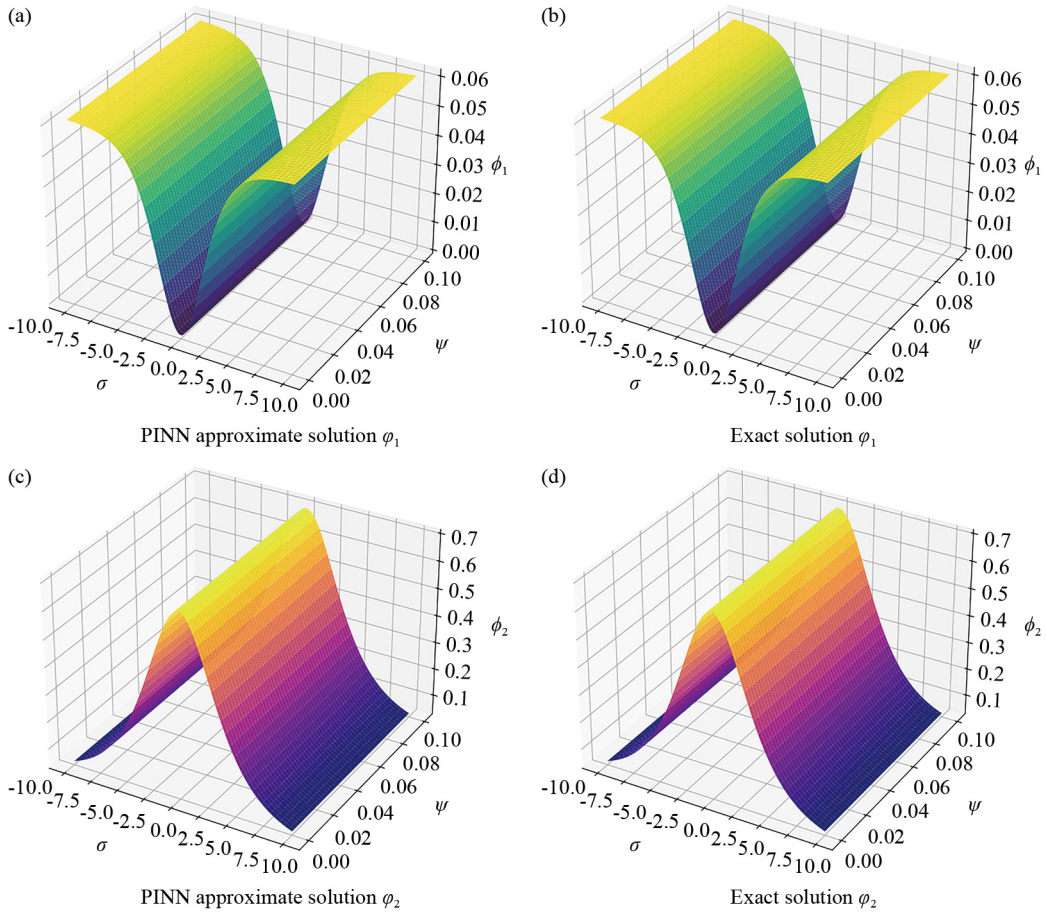


Figure 3. PINN approximations and exact solutions for system of PDEs

Both the surfaces seems alike with no visual error. However, we compute the absolute error between the exact solution and PINN approximation, which is shown in Figure 4. The 3D surface of absolute error shows that there is very little error between the two solutions.

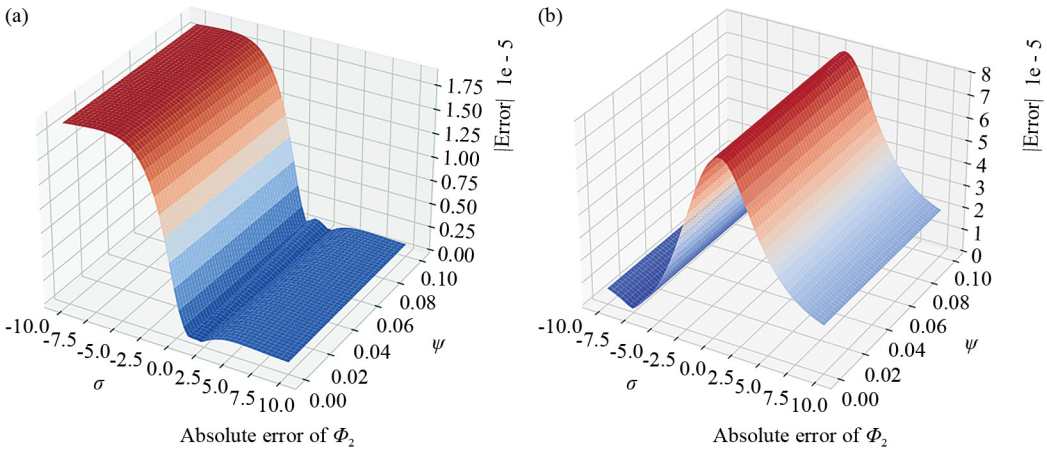
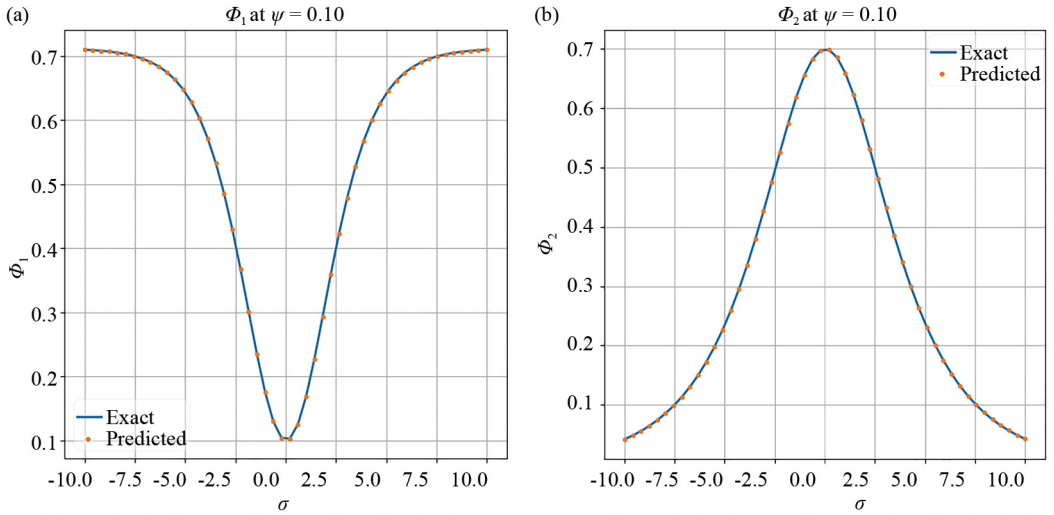


Figure 4. Absolute error between PINN approximations and exact solutions

The model is also simulated for the specific value of $\psi = 0.1$. Figure 5 shows the output of ϕ_1 and ϕ_2 respectively, for specific values of ψ. The PINN solution almost overlap the exact solution. Table 1 shows the comparison of the PINN approximation, exact solution, and their absolute error for both the functions.



The analysis of exact solution and predicted solution $\Phi_1(\sigma, \psi)$

The analysis of exact solution and predicted solution $\Phi_2(\sigma, \psi)$

Figure 5. The comparison of PINN predication and exact solutions ($\psi = 0.10$)

Table 1. Tabulated results for varying σ

σ	ψ	$\Phi_{1\,\text{exact}}$	$\Phi_{1\,\text{pred}}$	$\lvert\Phi_{1\,\text{error}}\rvert$	$\Phi_{2\text{exact}}$	$\Phi_{2\text{pred}}$	$\lvert\Phi_{2\,\text{error}}\rvert$
−10	0.1	0.0610	0.0610	5.7594×10^{-5}	0.0419	0.0419	3.0475×10^{-5}
−08	0.1	0.0603	0.0603	5.2457×10^{-5}	0.0853	0.0853	2.9226×10^{-5}
−06	0.1	0.0575	0.0575	4.8262×10^{-5}	0.1722	0.1722	2.4527×10^{-5}
−04	0.1	0.0472	0.0472	4.4975×10^{-5}	0.3354	0.3354	1.5065×10^{-5}
−02	0.1	0.0201	0.0201	8.2533×10^{-5}	0.5739	0.5739	4.2315×10^{-5}
00	0.1	0.0004	0.0004	1.5057×10^{-5}	0.6978	0.6978	1.2534×10^{-5}
02	0.1	0.0193	0.0193	1.0483×10^{-5}	0.5795	0.5795	8.0451×10^{-5}
04	0.1	0.0468	0.0468	4.7354×10^{-5}	0.3405	0.3405	6.1573×10^{-5}
06	0.1	0.0574	0.0574	2.2872×10^{-5}	0.1751	0.1751	2.0012×10^{-5}
08	0.1	0.0603	0.0603	4.3873×10^{-5}	0.0868	0.0868	2.3171×10^{-5}
10	0.1	0.0610	0.0610	5.3993×10^{-5}	0.0426	0.0426	3.0000×10^{-5}

4. Conclusion

In this paper, a PINN is implemented to obtain approximate solutions for the system of JM equations, which is a nonlinear coupled PDE system. The PINN approximate the solution using back propagation techniques, which is a unsupervised technique. Adam optimizer is used to minimizes the loss function, which is designed using the PDEs and ICs. The model accuracy and efficiency is evaluated by different graphical comparison and error metrics. The convergence of model and the absolute error between the PINN approximation and exact solution show that the model is highly accurate. In future, this model can be used for ODE system of infectious disease for prediction.

Conflict of interest

The author declares no conflict of interests.

References

[1] Zhu M, Anwar AH, Wan Z, Cho JH, Kamhoua CA, Singh MP. A survey of defensive deception: Approaches using game theory and machine learning. *IEEE Communications Surveys and Tutorials*. 2021; 23(4): 2460-2493. Available from: https://doi.org/10.1109/COMST.2021.3102874.

[2] Lauriola I, Lavelli A, Aiolli F. An introduction to deep learning in natural language processing: Models, techniques, and tools. *Neurocomputing*. 2022; 470: 443-456. Available from: https://doi.org/10.1016/j.neucom.2021.05.103.

[3] Islam AR. Machine learning in computer vision. In: *Applications of Machine Learning and Artificial Intelligence in Education*. USA: IGI Global Scientific Publishing; 2022. p.48-72.

[4] Tran MQ, Doan HP, Vu VQ, Vu LT. Machine learning and IoT-based approach for tool condition monitoring: A review and future prospects. *Measurement*. 2023; 207: 112351. Available from: https://doi.org/10.1016/j.measurement.2022.112351.

[5] Pratama DA, Bakar MA, Ismail NB. ANN-based methods for solving partial differential equations: A survey. *Arab Journal of Basic and Applied Sciences*. 2022; 29(1): 233-248. Available from: https://doi.org/10.1080/25765299.2022.2104224.

[6] Sartanpara PP, Meher R, Nikan O, Avazzadeh Z. Solution of generalized fractional Jaulent-Miodek model with uncertain initial conditions. *AIP Advances*. 2023; 13(12): 125303. Available from: https://doi.org/10.1063/5.0166789.

[7] Prajapati VJ, Meher R. Solution of time-fractional Rosenau-Hyman model using a robust homotopy approach via formable transform. *Iranian Journal of Science and Technology, Transactions A: Science*. 2022; 46(5): 1431-1444. Available from: https://doi.org/10.1007/s40995-022-01347-w.

[8] Sartanpara PP, Meher R. Solution of generalised fuzzy fractional Kaup-Kupershmidt equation using a robust multi parametric approach and a novel transform. *Mathematics and Computers in Simulation*. 2023; 205: 939-969. Available from: https://doi.org/10.1016/j.matcom.2022.10.020.

[9] Dhirawat K, Meher R. Semi-analytical approach to nonlinear partial differential equations using homotopy analysis technique (HAM). *Contemporary Mathematics*. 2023: 4(4): 721-732. Available from: https://doi.org/10.37256/cm.4420232467.

[10] Sartanpara PP, Meher R. A robust computational approach for Zakharov-Kuznetsov equations of ion-acoustic waves in a magnetized plasma via the Shehu transform. *Journal of Ocean Engineering and Science*. 2023; 8(1): 79-90. Available from: https://doi.org/10.1016/j.joes.2021.11.006.

[11] Nikan O, Rashidinia J, Jafari H. Numerically pricing American and European options using a time fractional Black-Scholes model in financial decision-making. *Alexandria Engineering Journal*. 2025; 112: 235-245. Available from: https://doi.org/10.1016/j.aej.2024.10.083.

[12] Avazzadeh Z, Nikan O, Nguyen AT, Nguyen VT. A localized hybrid kernel meshless technique for solving the fractional Rayleigh-Stokes problem for an edge in a viscoelastic fluid. *Engineering Analysis with Boundary Elements*. 2023; 146: 695-705. Available from: https://doi.org/10.1016/j.enganabound.2022.11.003.

[13] Luo M, Qiu W, Nikan O, Avazzadeh Z. Second-order accurate, robust and efficient ADI Galerkin technique for the three-dimensional nonlocal heat model arising in viscoelasticity. *Applied Mathematics and Computation*. 2023; 440: 127655. Available from: https://doi.org/10.1016/j.amc.2022.127655.

[14] Alkan A. Improving homotopy analysis method with an optimal parameter for time-fractional Burgers equation. *Karamanoglu Mehmetbey University Journal of Engineering and Natural Sciences*. 2022; 4(2): 117-134. Available from: https://doi.org/10.55213/kmujens.1206517.

[15] Alkan A, Akturk T, Bulut H. The traveling wave solutions of the conformable time-fractional zoomeron equation by using the modified exponential function method. *Eskisehir Technical University Journal of Science and Technology A-Applied Sciences and Engineering*. 2024; 25(1): 108-114. Available from: https://doi.org/10.18038/estubtda.1370631.

[16] Brunton SL, Noack BR, Koumoutsakos P. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*. 2020; 52(1): 477-508. Available from: https://doi.org/10.1146/annurev-fluid-010719-060214.

[17] Cybenko G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*. 1989; 2(4): 303-314. Available from: https://doi.org/10.1007/BF02551274.

[18] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural Networks*. 1989; 2(5): 359-366. Available from: https://doi.org/10.1016/0893-6080(89)90020-8.

[19] Ezzinbi K, Fu X. Existence and regularity of solutions for some neutral partial differential equations with nonlocal conditions. *Nonlinear Analysis: Theory, Methods and Applications*. 2004; 57(7-8): 1029-1041. Available from: https://doi.org/10.1016/j.na.2004.03.027.

[20] Raissi M, Perdikaris P, Karniadakis GE. Machine learning of linear differential equations using Gaussian processes. *Journal of Computational Physics*. 2017; 348: 683-693. Available from: https://doi.org/10.1016/j.jcp.2017.07.050.

[21] Kharazmi E, Zhang Z, Karniadakis GE. *hp*-VPINNs: Variational physics-informed neural networks with domain decomposition. *Computer Methods in Applied Mechanics and Engineering*. 2021; 374: 113547. Available from: https://doi.org/10.1016/j.cma.2020.113547.

[22] Yu B. The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems. *Communications in Mathematics and Statistics*. 2018; 6(1): 1-12. Available from: https://doi.org/10.1007/s40304-018-0127-z.

[23] Liu K. Automatic integration. *arXiv:2006.15210*. 2020. Available from: https://doi.org/10.48550/arXiv.2006.15210.

[24] Baydin AG, Pearlmutter BA, Radul AA, Siskind JM. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*. 2018; 18(153): 1-43.

[25] Cai Z, Chen J, Liu M, Liu X. Deep least-squares methods: An unsupervised learning-based numerical method for solving elliptic PDEs. *Journal of Computational Physics*. 2020; 420: 109707. Available from: https://doi.org/10.1016/j.jcp.2020.109707.

[26] Kovacs A, Exl L, Kornell A, Fischbacher J, Hovorka M, Gusenbauer M, et al. Conditional physics informed neural networks. *Communications in Nonlinear Science and Numerical Simulation*. 2022; 104: 106041. Available from: https://doi.org/10.1016/j.cnsns.2021.106041.

[27] Mao Z, Jagtap AD, Karniadakis GE. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*. 2020; 360: 112789. Available from: https://doi.org/10.1016/j.cma.2019.112789.

[28] Wang S, Wang H, Perdikaris P. On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*. 2021; 384: 113938. Available from: https://doi.org/10.1016/j.cma.2021.113938.

[29] Alkhezi Y, Shafee A. Analytical techniques for studying fractional-order Jaulent-Miodek system within algebraic context. *Fractal and Fractional*. 2025; 9(1): 50. Available from: https://doi.org/10.3390/fractalfract9010050.

[30] Jaulent M, Miodek I. Nonlinear evolution equations associated with energy-dependent Schrödinger potentials. *Letters in Mathematical Physics*. 1976; 1: 243-250. Available from: https://doi.org/10.1007/BF00417611.

[31] Atangana A, Alabaraoye E. Solving a system of fractional partial differential equations arising in the model of HIV infection of CD_4^+ cells and attractor one-dimensional Keller-Segel equations. *Advances in Difference Equations*. 2013; 2013: 1-14. Available from: https://doi.org/10.1186/1687-1847-2013-94.

[32] Atangana A, Kilicman A. Analytical solutions of the space-time fractional derivative of advection dispersion equation. *Mathematical Problems in Engineering*. 2013; 2013: 853127. Available from: https://doi.org/10.1155/2013/853127.

[33] Atangana A, Botha JF. Analytical solution of the groundwater flow equation obtained via homotopy decomposition method. *Journal of Earth Science and Climatic Change*. 2012; 3(2): 115. Available from: https://doi.org/10.4172/2157-7617.1000115.

[34] Tao H, Yu-Zhu W, Yun-Sheng H. Bogoliubov quasiparticles carried by dark solitonic excitations in nonuniform Bose-Einstein condensates. *Chinese Physics Letters*. 1998; 15(8): 550. Available from: https://doi.org/10.1088/0256-307X/15/8/002.

[35] Ma WX, Li CX, He J. A second Wronskian formulation of the Boussinesq equation. *Nonlinear Analysis: Theory, Methods and Applications*. 2009; 70(12): 4245-4258. Available from: https://doi.org/10.1016/j.na.2008.09.010.

[36] Das GC, Sarma J, Uberoi C. Explosion of soliton in a multicomponent plasma. *Physics of Plasmas*. 1997; 4(6): 2095-2100. Available from: https://doi.org/10.1063/1.872545.