Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P5695-cd

Verification of Bayesian Physics-Informed Neural Networks

Zhen Yang

Civil and Environmental Engineering, University of Strathclyde, Glasgow, UK. E-mail: z.yang.2021@uni.strath.ac.uk

Yu Chen

School of Engineering, University of Liverpool, UK. E-mail: lesliechen666@outlook.com

Edoardo Patelli

Civil and Environmental Engineering, University of Strathclyde, Glasgow, UK. E-mail: edoardo.patelli@strath.ac.uk

With the rapid advancement of machine learning technology, its applications are becoming increasingly vital across various critical systems and domains. However, the effectiveness of machine learning models heavily depends on high-quality data, which is often costly to obtain and affected by inherent uncertainty. To address this challenge, we propose a robust Bayesian physics-informed neural network (BPINN) that enables the analysis of limited datasets while incorporating uncertainty quantification, all while maintaining the physical interpretability of predictions. In this study, we develop a verification problem to systematically assess and verified the effectiveness and robustness of our approach and shown the performance by predicting the fracture time of a steel alloy based on very limited dataset.

Keywords: Machine Learning, Uncertainty Quantification, Benchmark Problems, Bayesian Physics-Informed Neural Networks

1. Introduction

With the rapid advancement of machine learning (ML) techniques and the increasing availability of data science methodologies, it has become possible to address complex challenges across various engineering domains, including aerospace, nuclear energy, and renewable energy. However, in many critical applications, data availability remains limited or highly sparse, significantly impacting the performance and reliability of ML methods [1].

In real-world engineering scenarios, data acquisition is often challenging due to factors such as loss, corruption, or difficulties in measurement under extreme conditions. For instance, in fields like aerospace, nuclear energy, and certain renewable energy sectors, data collection is costly and infeasible under high-risk or extreme environments (see e.g. [11]). Additionally, some physical systems, such as environmental hazard monitoring and disaster prediction, naturally suffer from data sparsity due to the infrequency of catastrophic events.

The availability of limited data has a profound impact on ML model training and predictive performance [12]. Traditional ML techniques rely on large datasets for learning, making it difficult for these models to generalize effectively when trained on sparse datasets, often resulting in overfitting and poor extrapolation beyond the training range. Even when ML models perform well within the training data distribution, they can fail in unseen physical scenarios, limiting their practical applicability.

Numerical simulations offer a potential solution by generating synthetic probabilistic data, which can supplement ML training. However, these simulations come with high computational costs and often introduce noise, reducing overall data quality and reliability [16]. As a result, while simulated data can enhance ML model training, it does not fully resolve the challenges associated with data sparsity.

Physics-Informed Neural Networks (PINNs) integrate physical laws with conventional neural network models, offering a promising approach to addressing problems constrained by underlying physical principles during the training process [2]. However, also PINNs exhibit limited capabilities in handling uncertainty, which can hinder their effectiveness in real-world applications [3].

To overcome this limitation, the recently developed Bayesian Physics-Informed Neural Network (B-PINN) has emerged as a powerful tool for managing uncertainty and sparse data across a wide range of applications.

In this work, we propose a set of benchmark problems designed to systematically evaluate and verified machine learning models under different scenarios of increasing complexity. These scenarios vary in terms of dimensionality, uncertainty levels, prior knowledge availability, and model complexity. By utilizing these benchmark problems, researchers and engineers can assess and compare the accuracy, stability, and computational efficiency of their algorithms in diverse physical modelling scenarios, ensuring consistent evaluation under standardized conditions.

2. Methodology

In our deep learning framework, we integrate the methodologies of Bayesian Neural Networks (BNNs) and Physics-Informed Neural Networks (PINNs) to develop a more robust and interpretable model. BNNs combine Bayesian inference with neural networks, where the model weights are represented as probability distributions, enabling uncertainty quantification in predictions. This probabilistic approach enhances the robustness of BNNs, making them well-suited for complex and high-dimensional tasks [5].

On the other hand, PINNs are designed to solve challenging partial differential equations (PDEs) by incorporating physical knowledge into the neural network architecture. A common technique involves embedding physical constraints into the loss function or enforcing them as output constraints, ensuring that predictions remain physically consistent and interpretable [6], [14].

Existing research PINNs often incorporates physical constraints into the loss function using a Mean Squared Error (MSE)-like penalty, similar to conventional neural networks [7] [8][9]. While this approach provides a straightforward method for integrating physics-based terms, it introduces inconsistencies when combined with a Bayesian formulation. Typically, BNNs define the loss function using the negative log-likelihood or the Evidence Lower Bound reflecting a probabilistic interpretation of the data likelihood and priors as follow:

$$\mathcal{L}_{BNN} = \frac{1}{N}\sum_{i=1}^{N}\big(-\log P\,(\,y_i \mid \theta\,)\big) + \mathrm{KL}\big(q(\theta) \parallel P(\theta)\big) \quad (1)$$

where N represents the number of (training) samples, y_i the observed (true) data, θ the set of parameters of the BNN. The first term is negative log-likelihood while the second term is Kullback-Leibler (KL) term acting as a prior regularisation representing which quantifies the difference between the posterior distribution of the network parameters learned from data $q(\theta)$, and the prior distribution over parameters $P(\theta)$.

In contrast, loss functions based on MSE or Mean Absolute Error (MAE), commonly used in standard neural networks, focus on pointwise errors (i.e. difference between the predicted output of the model y_i and y_i) rather than modelling full probability distributions

$$\mathcal{L}_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (2)$$

To resolve this inconsistency, we propose reformulating the physical constraint term into a negative log-likelihood framework, ensuring that both the data-fitting term and the physics-informed term share a consistent probabilistic interpretation within the BNN loss function. Specifically, we assume that the model's prediction for the physical constraints follows a Gaussian distribution with learnable mean and variance, resulting in a per-point NLL penalty (see Eqs. (3)–(5)). This adjustment ensures dimensional consistency between the physics likelihood and the data likelihood, both conforming to the Bayesian negative log-likelihood (NLL) formulation.

$$Loss_{Total} = (1-\lambda)\,Loss_{Data} + \lambda\,Loss_{phys} \quad (3)$$

$$Loss_{Data} = \frac{(y-\mu)^2}{2\sigma_{\text{data}}^2} + \log\big(\sqrt{2\pi}\sigma_{\text{data}}\big) \quad (4)$$

$$Loss_{phys} = \frac{\left(s - \mu_{\text{phys}}\right)^2}{2\sigma_{\text{phys}}^2} + \log\left(\sqrt{2\pi}\sigma_{\text{phys}}\right) \quad (5)$$

Here, μ or μ_{phys} and σ or σ_{phys} represent the mean and standard deviation of the model outputs under their respective distributions while λ is a parameter that define the weight between the physics likelihood and the data likelihood Unlike methods that measure the full distribution-to-distribution distance (e.g., KL divergence between arbitrary distributions), this approach treats each physics or data point as a sample drawn from a univariate Gaussian distribution, aligning with the BNN's negative log-likelihood framework.

Our proposed method follows the workflow illustrated in Fig. 1. The process begins with data preparation, where the input dataset and the corresponding physical constraint formulation are defined. Next, the neural network architecture is constructed, with the prior distribution of weights initialized. The physical constraints are then reformulated into a negative log-likelihood representation, specifically modelled as a Gaussian NLL for the physics residuals. Once these components are established, the combined loss function is formulated (as described in Eq. (3)), followed by the training and optimization process.

For evaluating our model performance, our research uses R^2 (coefficient of determination) as the primary evaluation metric

$$\mathrm{R}^2 = 1 - \frac{\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{y_i} - \widehat{\mathrm{y}_\mathrm{i}})^2}{\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{y_i} - \bar{\mathrm{y}})^2} \quad (6)$$

We selected this evaluation metric due to its clarity and interpretability [17]. It provides a straightforward measure of how well the model captures variability, making it easier to compare different models: a value of R^2 equal to 1 indicates a perfect prediction, where the model captures all variability in the data. A value around 0 suggests that the model performs no better than the mean prediction, meaning it fails to explain variance in the data while $R^2 < 0$ implies that the model performs worse than simply predicting the mean, indicating poor generalization. However, we also recognize certain limitations in using R^2 as the primary evaluation metric for our model. When applying to scenarios with sparse data, R^2 reliability may be compromised since it is computed based on variance. In cases where only a few data points are available, the sample variance may not accurately reflect the true variability of the dataset. For instance, in small datasets, extreme values can have a disproportionate influence on R^2 making it challenging to consistently assess the model's true predictive performance [15]. Therefore, we also incorporated additional metrics to ensure a more comprehensive assessment of our model's performance.

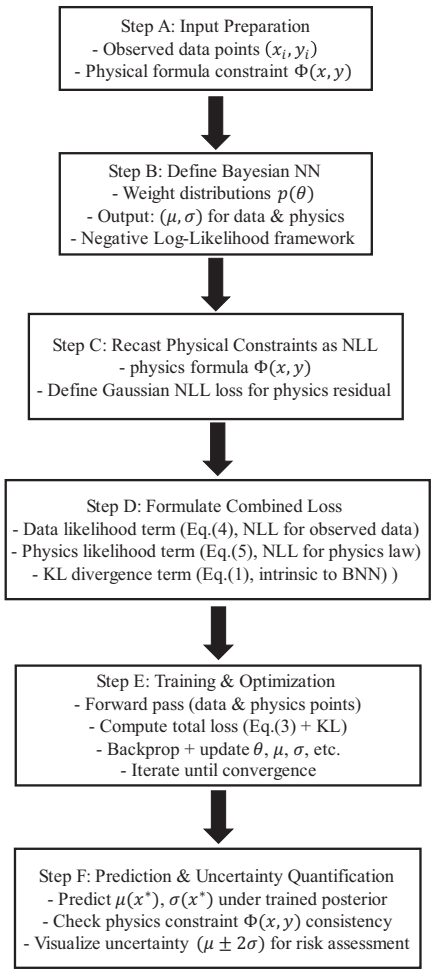

Fig. 1 Model workflow

3. Verification problem

A key aspect of our investigation is evaluating whether our model can maintain satisfactory performance when physical constraints are missing or incomplete. This is particularly

important as, in many practical applications, obtaining complete and accurate physical equations to describe the relationships between inputs and outputs is often challenging [10].

To systematically examine this issue, we design a controlled experiment in which certain physical information is deliberately omitted, partially simplified, or modified. This allows us to assess the robustness and adaptability of our model. Our objective is to determine whether the proposed approach can effectively infer missing physical information from the available data, and whether the learned representations remain consistent with known physical principles despite incomplete constraints.

Thus, a verification problem has been created to simulate the structural deformation of a beam-based systems subjected to various loading conditions and material properties. The characteristic equations define the maximum displacement (D) and stress (S) as function of load and material properties derived from classical beam bending theory:

$$D = \frac{4L^3}{Ewt}\sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{X}{w^2}\right)^2} \tag{7}$$

$$S = \frac{600Y}{wt^2} + \frac{600X}{w^2t} \tag{8}$$

where the symbols are defined as in Table 3. These equations coupled with the constraints $S \leq R \; and \; D \leq D_0$, adapted from [19], consider the combined effects of horizontal and vertical loads, which shows that both components contribute to the total deflection.

Parameter [unit]	Distribution
L (Length) [m]	$N(100{,}100 \times 0.05)$
D_0(Displacement tolerance) [m]	$N(2.2535{,}2.2535 \times 0.05)$
R (yield stress) [Pa]	$N(40000{,}2000)$
E (Young's modulus) [Pa]	$N(2.9E7{,}1.45E6)$
w (Width) [m]	$U(4{,}8)$
T (Thickness) [m]	$U(2{,}4)$
Y (Vertical load) [N]	$N(1000{,}100)$
X (Horizontal load) [N]	$N(500{,}100)$

Table. 1 Parameter of the verification problem. Normal distributions are indicated as $N(\mu, \sigma)$ with mean μ and variance σ^2. Uniform distribution as $U(a, b)$ a and b are the bounds of the distribution.

Synthetic data have been generated for training and testing by sampling the values of each parameter from its distribution and further perturb each computed D and S by adding a small (0.5 – 1%) Gaussian noise.

The ML model has 6 inputs, 2 hidden layers with 16 and 12 neurons and 2 outputs. We use multitask learning to predict D and S as outputs at the same time in models, because D and S share the same input parameters and have some physical coupling. It help with enhancing model efficiency by enabling shared feature extraction，improving generalization through task-related knowledge transfer, and ensures physical consistency between the predicted D and S.

$$f: \{E, R, X, Y, w, t\} \rightarrow \{D, S\}$$

The created dataset allows to systematically test our model's performance under different training set sizes, with a particular focus on small datasets (10–50 samples) to evaluate the model's effectiveness in data-limited scenarios from a comprehensive perspective. We also assume three different levels on knowledge are available to define the physical:

1. No physical knowledge at all. In this case the only choice is to use a BNN or other different kinds of conventional models.

2. Partial prior physical knowledge available, based on the incomplete formula :

$$D_2 = f\left(\frac{L^3}{Ewt} \cdot \frac{Y}{t^2}\right) \tag{9}$$

$$S_2 = f\left(\frac{Y}{wt^2}\right) \tag{10}$$

We assume that the exact functional forms are unknown. Therefore a 50 data points are randomly selected from the synthetic datasets to fit Regression models as shown Fig.2 and Fig.3.

3. Complete physical knowledge is available. We use the physical information used to generate the synthetic data as a full physical constraint.

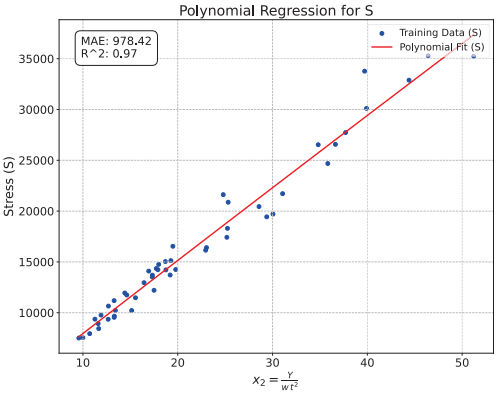

Fig. 2 Regression for fitting incomplete physical constraint for stress (fitting equation $-2.031 \cdot x_2^2 + 7.256 \cdot x_2 + 7.142$

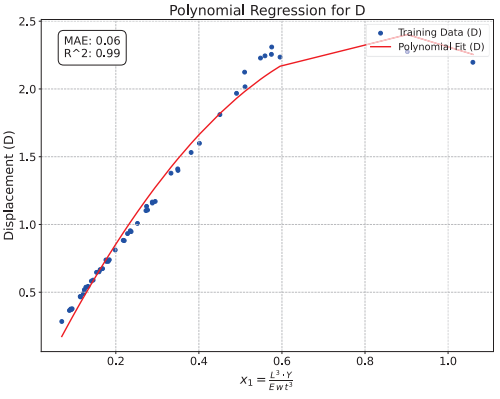

Fig. 3 Regression for fitting incomplete physical constraint for displacement (fitting equation $D=-3.643x_1^2 + 6.218x_1 - 2.433$)

The performance of the neural network is shown in Fig.4 and Fig.5 for the stress and displacement, respectively for different size of training data (10,20,30,50,100, 200, and 500). From the results of S and D, it is seems that for small data set the partial physical knowledge provides the best results, while complete knowledge is in the middle, and as expected no knowledge model does not provide satisfying results. Generally, our models with different levels of physical prior show improvement than traditional none physical data driven model.

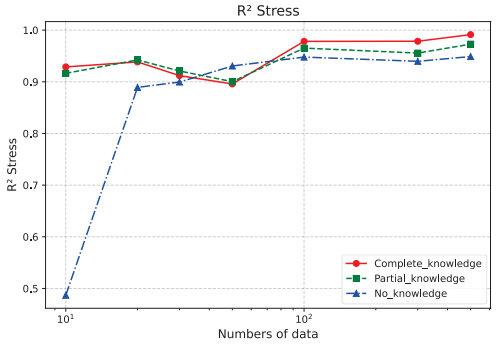

Fig. 4 Stress prediction using complete, partial and no physical prior knowledge under different numbers of training data

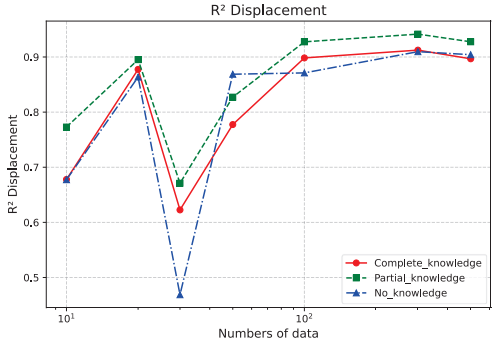

Fig. 5 Displacement prediction using complete, partial and no physical prior knowledge under different numbers of samples

4. Case study PRMOAP Research

The proposed approach has been successfully adopted to predict the creep rupture characteristics of SCH24, a specialized steel alloy used in nuclear reactor construction. [4][11] The primary challenge in this application was the extremely limited dataset due to the high cost and difficulty of experimental data collection (creep rupture process is highly complex and difficult to control, introducing significant uncertainty into the process). The dataset [18] contains the features shown in Table. 1. The database contains 8005 entries for with 281 different material cast codes represent different material type and production batch. We analysed a specific steel batch (cast code 221) containing only 13 data points. The cast code defines the material composition and therefore the remaining input

variable were the stress and temperature and the quantity of interest the fracture time.

Input feature	Output feature
Material code	Fracture Time (FT)
Cast code	Elongation
stress	Reduction of Area (RA)
temperature	
Composition (19 elements)	

Table. 2 Features of PROMAP datasets

We use the approach shown in Fig.6. A dataset of generic SCH24 steel alloy (dataset B) is used for pretraining the neural network while the original datasets (dataset A) is used to fine training the neural networks. Finally physical constraint is used as shown in the verification example.

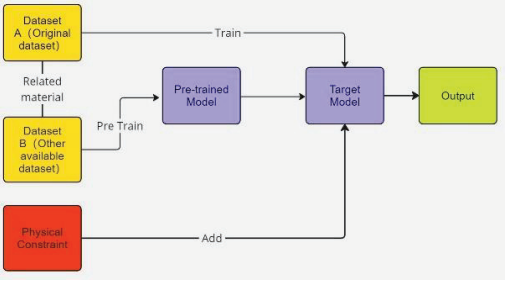

Fig. 6 PRMOAP flowchart

The physical loss constraint in our model comes from the empirical Larson-Miller Parameter formula

$$\sigma = f[T(C + log\ (t))] \qquad (11)$$

where σ is stress, T is temperature, C is a constant fitting parameter and t the fracture time. However, the functional form of f is unknown. We used the information from other related cast codes (222-227) to fit a non-linear physical constrain as shown in Fig.7.

We testes different physical loss weights (see Fig.8) and identifying the best model for a value of $\lambda = 0.2$. The performance of the best network trained using out approach is shown in Fig.9.

The proposed approach outperformed traditional approaches for the PROMAP case study as shown in Table.3. where RMSA represents the Root Mean Squared Error. The results have demonstrated the effectiveness and feasibility of the proposed model able to yield reliable and accurate results.

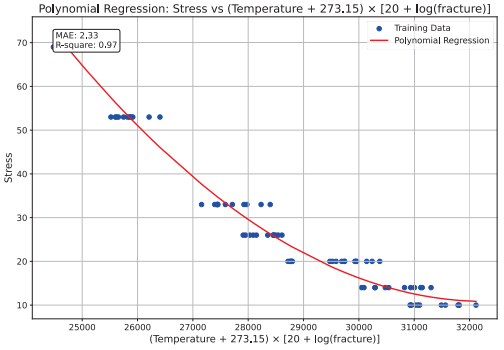

Fig. 7 Regression analysis for physical constraint.

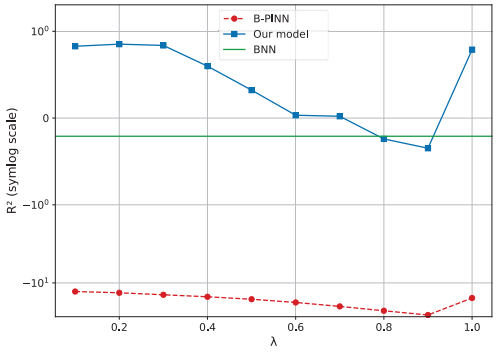

Fig. 8. Comparison of R^2 across methods under different weights for physical loss

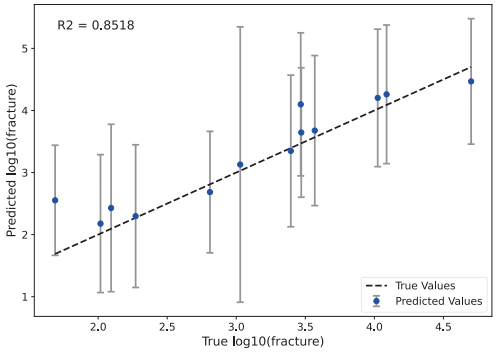

Fig. 9 Performance of B-PINN model with $\lambda = 0.2$.

Model	MAE	MSE	RMSE	R^2
BNN	0.7853	0.9228	0.9606	-0.2096
B-PINN	2.7141	8.0842	2.8433	-12.8897
Our method	0.2426	0.1131	0.3363	0.8518

Table. 3 Comparison of different metrics for predicting fracture time from PROMAP database.

However, in the field of deep learning, validation through a single application is insufficient to establish the versatility and transferability of the approach. Therefore, to ensure the robustness of our approach, it is essential to conduct further evaluations across multiple domains, datasets, and problem settings. Expanding the range of test cases will allow us to assess the model's generalization capabilities and performance stability under diverse conditions.

5. Discussion

The results presented demonstrate the successful application of the proposed approach in both a synthetic case study and a complex real-world scenario, consistently outperforming traditional methods. Moreover, comparative experiments conducted under different levels of physical constraint knowledge—which simulate varying degrees of understanding or uncertainty in the physical prior—further highlight the advantages of our approach over a conventional Bayesian Neural Network.

These findings indicate that even with partial physical information (relationship among parameters), the approach can still effectively utilise such information to its predictive performance. And in our experiments, the model with partial physical knowledge even behaves better than the model with full prior knowledge. These surprising results need further investigation, but it may suggest that a moderate physical priori can provide sufficient guidance to the model to avoid overfitting while retaining a certain degree of flexibility when data are limited. When the data size is small, the full knowledge may impose 'too strong' constraints and not allowing accounting for the uncertainty in the data.

It is also important to notice, that full knowledge is never available (and if it is there is no need of a machine learning approach); instead, physical constraints are often simplified or derived under idealized assumptions. This suggests that proposed approach can effectively leverage incomplete or approximate physical priors, making it a reliable solution for applications where exact governing equations are unavailable or uncertain.

However, further tests and applications are required to validate and refine the approach especially applicability across diverse domains needs further evaluation.

6. Conclusions

In this paper, we propose a novel Bayesian Physics-Informed Neural Network framework that directly incorporates physical constraints into the negative log-likelihood loss function, ensuring consistency within the Bayesian Neural Network framework. Our approach was evaluated on two distinct problems: a numerical example used for verification purpose in order the quantify the effect of each assumption and the effect of data size and a complex nuclear material property prediction task. Experimental results demonstrate that our method significantly improves both accuracy and stability compared to traditional PINN and B-PINN methods.

Despite its advantages, our approach has certain limitations that require further investigation:

- The current physical loss function is still based on a pointwise Gaussian assumption and our method has not yet incorporated more comprehensive probability distribution metrics

To further refine and enhance the proposed framework, we outline three key research directions:

- Extending Probabilistic Modelling of Physical Losses:
- Investigate non-Gaussian distributions, correlation structures, and complex noise models:
- Integrating Advanced Probability Distribution Matching Techniques:

We anticipate that these improvements will contribute to a more comprehensive and adaptable framework, enabling its application to a broader range of benchmark problems and real-world scenarios.

Acknowledgement

This work has been supported by the Strathclyde Centre for Doctoral Network in Digital technologies for Resilient and Sustainable Infrastructures and the Research Innovation Scotland - International Collaboration Fund

References

[1] Dou, B., et al. (2023). Machine learning methods for small data challenges in molecular science. *Chemical Reviews*, 123, 8736–8780.

[2] Cuomo, S., et al. (2022). Scientific machine learning through physics-informed neural networks: Where we are and what's next. *Journal of Scientific Computing*, 92, 88.

[3] de la Mata, F. F., et al. (2023). Physics-informed neural networks for data-driven simulation: Advantages, limitations, and opportunities. *Physica A: Statistical Mechanics and its Applications*, 610, 128415.

[4] Lye, Adolphus, Nawal Prinja, and Edoardo Patelli. 2022. 'Probabilistic Artificial Intelligence Prediction of Material Properties for Nuclear Reactor Designs'. In 32nd European Safety and Reliability Conference. Dublin, Ireland.https://rpsonline.com.sg/rps2prod/esrel22-epro/html/S24-02-306.xml.

[5] Jospin, L. V., et al. (2022). Hands-on Bayesian neural networks—A tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17, 29–48.

[6] Farea, A., O. Yli-Harja, and F. Emmert-Streib (2024). Understanding physics-informed neural networks: Techniques, applications, trends, and challenges. *AI*, 5, 1534–1557.

[7] Yang, L., X. Meng, and G. E. Karniadakis (2021). B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 425, 109913.

[8] Stock, S., et al. (2024). Bayesian Physics-informed Neural Networks for System Identification of Inverter-dominated Power Systems. *arXiv preprint arXiv:2403.13602*.

[9] Pensoneault, A., and X. Zhu (2024). Efficient Bayesian Physics Informed Neural Networks for inverse problems via Ensemble Kalman Inversion. *Journal of Computational Physics*, 508, 113006.

[10] Willard, J., Jia, X., Xu, S., et al. (2020). Integrating physics-based modeling with machine learning: A survey. arXiv preprint arXiv:2003.04919, 1(1), 1–34.

[11] Chen, Yu, and Edoardo Patelli. 2024. 'Towards Robust Prediction of Material Properties for Nuclear Reactor Design under Scarce Data - a Study in Creep Rupture Property'. In Reliability Computations in Sustainable and Resilient Development, 100–109. Beijing, China. https://drive.google.com/drive/folders/1QfP3Q-yO5o-E5qgPY6-3ghFVLgABqLw5?usp=sharing.

[12] Kumar, A., Boehm, M., & Yang, J. (2017). Data management in machine learning: Challenges, techniques, and systems. Proceedings of the ACM International Conference on Management of Data, 1717–1722.

[13] Dou, B., Zhu, Z., Merkurjev, E., et al. (2023). Machine learning methods for small data challenges in molecular science. Chemical Reviews, 123(13), 8736–8780.

[14] Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378, 686–707.

[15] Barrett, J. P. (1974). The coefficient of determination—some limitations. The American Statistician, 28(1), 19–20.

[16] Jordon, J., Szpruch, L., Houssiau, F., et al. (2022). Synthetic data—what, why and how? arXiv preprint arXiv:2205.03257.

[17] Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Computer Science, 7, e623.

[18] Lye, Adophuls. (2022). PROMAP dataset. Available at: https://github.com/Adolphus8/Project_PROMAP.

[19] Surjanovic, S. & Bingham, D. (2013). Virtual Library of Simulation Experiments: Test Functions and Datasets. Retrieved March 4, 2025, from http://www.sfu.ca/~ssurjano.