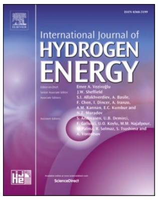
International Journal of Hydrogen Energy 87 (2024) 373–388

Contents lists available at ScienceDirect

International Journal of Hydrogen Energy

journal homepage: www.elsevier.com/locate/he



Prediction of hydrogen solubility in aqueous solution using modified mixed effects random forest based on particle swarm optimization for underground hydrogen storage

Grant Charles Mwakipunda [a], Norga Alloyce Komba [b], Allou Koffi Franck Kouassi [a], Edwin Twum Ayimadu [c], Melckzedeck Michael Mgimba [d], Mbega Ramadhani Ngata [a], Long Yu [a,*]

[a] *Key Laboratory of Theory and Technology of Petroleum Exploration and Development in Hubei Province, China University of Geosciences, Wuhan 430074, China*
[b] *Research Institute of Environmental Law, Wuhan University, No. 299, Luoyu Road, 10, Wuchang District, Wuhan City, Hubei Province, China*
[c] *School of Resource and Environmental Science, Wuhan University, No. 299, Luoyu Road, 10 Wuchang District, Wuhan City, Hubei Province, China*
[d] *Mbeya University of Science and Technology (MUST), P.O Box 131, Mbeya, Tanzania*

ARTICLE INFO

Handling Editor: Ramazan Solmaz

Keywords:
Underground hydrogen storage
Hydrogen solubility
Machine learning
Geological formations

ABSTRACT

This paper aims to enhance the prediction accuracy of hydrogen solubility in aqueous solution, which is crucial for safe and efficient underground hydrogen storage (UHS). The study developed a new hybrid machine learning (ML) algorithm, particle swarm optimization-mixed effects random forest (PSO-MERF), and compared with Extreme Gradient Boosting (XGBoost), K-Nearest Neighbors (KNN), Random Forest (RF), and Equation of State (EOS) models. PSO-MERF demonstrated superior performance, achieving a high correlation coefficient (R) of 0.9982, root means square error (RMSE) of 0.0015, and mean absolute error (MAE) of 0.00091, with less computational time (1.01 s). Among the EOS models used, Soave-Redlich-Kwong (SRK) outperformed other models. The results suggest that PSO-MERF hyperparameter optimization leads to more accurate hydrogen solubility predictions, encouraging its use in UHS design and operation for safe and sustainable hydrogen storage.

1. Introduction

Fossil fuels such as coal, natural gas, and oil dominate the global energy supply, accounting for approximately 80% of the world's primary energy consumption [1–4]. This dominant position reflects fossil fuels' central role in powering industrial processes, electricity generation, transportation, and heating. However, in 2023, approximately 75–80% of total carbon emissions in the atmosphere were from the combustion of fossil fuels for daily use [5–8]. Despite significant growth in renewable energy sources like wind, solar, hydrogen, and hydroelectric power, which have increased their share of the energy mix, the transition to a more sustainable and less carbon-intensive energy system is ongoing [9–11]. The exact percentage can fluctuate slightly based on factors such as economic growth patterns, advancements in renewable energy technologies, changes in energy policy, and shifts in consumer behavior towards more sustainable energy sources [12,13]. Hydrogen is gaining significant attention as a clean and efficient energy carrier due to its high energy density and potential for production from renewable sources [14,15]. It offers nearly three times the energy content of gasoline by weight, making it highly efficient for transportation and power generation [16]. Hydrogen boasts a remarkable energy density, packing around 140 MJ/kg compared to conventional fossil fuels. Coal comes in at 15–32 MJ/kg range, while oil and natural gas offer 47 MJ/kg and 54 MJ/kg, respectively [17].The combustion of hydrogen releases water [18,19]. The ability to produce hydrogen through electrolysis, using electricity generated from renewable sources, aligns with global sustainability goals by providing a method to store and leverage surplus renewable energy. Furthermore, hydrogen's versatility extends across various applications, from powering vehicles to generating electricity, highlighting its role in facilitating a transition to a more sustainable and resilient energy system [20,21].

Hydrogen storage can be broadly classified into surface and underground methods, each with specific advantages, applications, and limitations [22,23]. Surface storage involves storing hydrogen in tanks at

[*] Corresponding author.
E-mail address: yulong36@cug.edu.cn (L. Yu).

https://doi.org/10.1016/j.ijhydene.2024.09.054
Received 16 May 2024; Received in revised form 2 September 2024; Accepted 4 September 2024
Available online 9 September 2024

high pressure or in a liquefied state at cryogenic temperatures and storing it within materials like metal hydrides. This method is used for smaller-scale applications such as fuel cell vehicles, portable power systems, or local energy storage [24,25]. Underground hydrogen storage (UHS) emerges as a key technology for enabling large-scale, long-duration energy storage. This is particularly valuable for future energy systems aiming to significantly increase their reliance on renewable energy sources [26,27]. It involves storing hydrogen in subterranean formations, such as salt caverns [28,29], depleted oil and gas fields [29,30], or aquifers [29,31]. Each storage option has unique characteristics that make it suitable for different scenarios. While underground storage offers these advantages, it also faces challenges, including the need for significant upfront investment, potential environmental impacts, and the necessity for thorough geological assessments to ensure feasibility and safety. While challenges exist, underground hydrogen storage offers significant capacity, safety, and cost-effectiveness potential. This makes it a key element in building a sustainable and resilient future energy system centred on hydrogen as a critical energy carrier.

There are several mechanisms in which hydrogen can be stored in underground formations, such as structural trapping [32–34], capillary trapping [32], solubility (dissolution) trapping [32,34], mineralization trapping [32,35], and adsorption trapping mechanisms [32,36]. Solubility trapping, where hydrogen dissolves in underground structures' formation water or brine, offers a potentially higher efficiency for hydrogen storage with reduced operational demands. This mechanism's efficiency systems are based on the natural and passive nature of the dissolution process, which doesn't require continuous pressure or active management to maintain the stored hydrogen in its dissolved state. Once hydrogen is dissolved, it is less prone to escape or migrate, reducing the risk of leakage and the need for ongoing monitoring and maintenance compared to mechanisms like capillary trapping [27,37]. This mechanism is particularly relevant in aquifers, where large volumes of water can dissolve significant amounts of hydrogen, thus enhancing storage security through gradual dissolution [38]. When hydrogen is stored underground, its solubility in the surrounding brine or geological materials can affect how much hydrogen can be injected and retrieved. High solubility might lead to hydrogen dissolving into the surrounding fluids, potentially complicating recovery or leading to losses. Conversely, low solubility, particularly in formations like salt caverns where the hydrogen is stored in a gaseous state under high pressure, can facilitate more straightforward injection and extraction, maximizing the efficiency and effectiveness of the storage system [35]. Therefore, understanding and managing hydrogen solubility in different underground contexts is vital for optimizing these storage solutions for energy reliability, efficiency, and sustainability [39].

Storing hydrogen in water as a solution emerges as a promising option for underground hydrogen storage (UHS), offering a notable solubility rate of about 0.179 g per liter under a pressure of 10 atm and a temperature of 25 °C [40]. This approach stands out for its high energy density and affordability compared to alternative storage methods. However, accurately determining hydrogen solubility in water presents complexities, as its affected by various factors, including temperature, pressure, and the composition of the water, which can significantly alter the solubility dynamics. Researchers indicate that factors such as temperature, pressure, and the salt content of the reservoir significantly influence hydrogen solubility in the context of UHS. In general, a rise in pressure and temperature tends to enhance hydrogen dissolution, whereas higher salinity levels in the brine are associated with reduced hydrogen solubility [32].

Two main methods are used to obtain hydrogen solubility in pure water, hydrocarbons, and saline water: experimental methods and equation of state (EOS) models. These models include Soave-Redlich-Kwong (SRK) [41,42], Peng Robinson (PR) [41,42], Redlich-Kwong (RK) [42,43], Zudkevitch-Joffe (ZJ) [42], van der Waals [44], Statistical Associating Fluid Theory (SAFT) [45,46], Electrolyte Cubic [47], Non-Random Two-Liquid [48], PC-SAFT [49,50] etc. Moreover, EOS with molecular dynamic simulation is proposed for hydrogen solubility determination in heavy hydrocarbons at high pressures and temperatures [51]. However, experiments for hydrogen solubility measurements are expensive, time-consuming, and limited in scope. At the same time, EOS models can be complex, require specialized knowledge, and might not handle the varying salinity levels of underground hydrogen storage [52].

Machine learning (ML) has emerged as a highly effective tool for predicting hydrogen solubility more accurately than traditional methods due to its ability to process and learn from vast datasets encompassing diverse materials, experimental results, and conditions. By identifying complex, nonlinear relationships that are often invisible to conventional approaches, ML models can adapt to predict hydrogen solubility across various contexts with high precision [53,54]. This adaptability and rapid prediction capabilities significantly reduce the need for time-consuming and expensive experimental setups. Furthermore, as more data become available, ML models can continuously improve, refining their predictions and offering valuable insights into the underlying physics of hydrogen solubility. This makes ML an indispensable asset in accelerating the development of hydrogen storage technologies and understanding material behavior in energy applications.

There are limited literatures on estimating H_2 solubility in an aqueous solution for UHS purposes compared to other gases using ML models. Exploring this area has the potential to revolutionise UHS by enabling more efficient, safe, and cost-effective hydrogen storage solutions. For instance, Thanh et al. [55] utilized four ML algorithms to estimate H_2 solubility in an aqueous solution for optimizing the UHS process. Experimental data were collected from literature with pressure, temperature, and salinity as inputs. It was found that Adaboost outperformed other ML models such as RF, XGBoost, and GB by having a high coefficient of determination (R^2) of 0.994 and minimum errors, i.e., root mean square error (RMSE) of 0.0535 and mean absolute error of 0.0266, during the testing phase. However, Adaboost has several limitations: 1) It struggles with imbalanced datasets. The algorithm might focus too much on the minority class, leading to poor performance on the majority class.2) Adaboost performance is highly dependent on the choice of weak learners. The performance may suffer if the weak learners are not appropriate for the given problem.3) If the initial weak learners perform poorly, Adaboost may assign large weights to certain instances early on, potentially skewing the learning process. Also, Cao et al. [56] used a GA to estimate H_2 solubility in an aqueous solution for UHS purposes. The inputs for the model were pressure, temperature, and salinity, which were experimental data collected from the literature. To assess its effectiveness, the GA algorithm was compared with FBP, ANN, and RBF. It was found that GA outperformed other ML models by having R^2 of 0.9998 and 0.9184, RMSE of 0.003716 and 0.003201, standard deviation (STD) of 0.000014 and 0.00001, and absolute average relative deviation (AARD) of 183.25% and 62.13%, during training and testing, respectively. Nevertheless, GA faces several limitations: 1) GA are generally good at finding approximate solutions but may not always provide exact or optimal solutions, especially in problems where precise predictions are required.2) GA rely on stochastic processes, which means that different algorithm runs can produce different results. This variability can make it difficult to reproduce results consistently and may require multiple runs to obtain reliable outcomes.3) GA can have slow convergence rates, especially when fine-tuning solutions in the final stages of the algorithm. This can be inefficient for problems that require fast or real-time predictions, etc. Further, Ansari et al. [42] compared equations of state (EOS) and ML algorithms in predicting H_2 solubility in pure and saline water for UHS purposes. Experimental datasets were collected from the literature. RBF and LSSVM algorithms were optimized with different optimizers such as BBO, CA, ICA, and TLBO. EOSs used include SRK, PR, RK, and ZJ. It was found that RBF-CA outperformed other models with RMSE of 0.000176 and a correlation coefficient (R) of 0.972. Among EOS models, SRK performed better than other models. Though, RBF-CA has various limitations: 1) The combination usually leads to a more complex system that is harder to implement, tune, and understand than using each method individually.2) The parameters of the RBF network and the cultural algorithm interact in non-trivial ways, making the optimization process more challenging.3) The integration of cultural mechanisms can increase the computational cost of the algorithm. The management and updating of cultural knowledge require additional computational resources etc.

Furthermore, Mohammadi et al. [49] used ML algorithms and EOS models to predict hydrogen solubility in hydrocarbons for UHS storage. Experimental datasets were collected from the literature with molecular weight, critical pressure, and critical temperature of solvents, as well as pressure and temperature as inputs. The ML algorithms used in their study include XGBoost, AdaBoost-SVR, CatBoost, LightGBM, and MLP optimized by the LM algorithm, while the EOS models used include SRK, PR, RK, ZJ, and PC-SAFT. It was found that XGBoost surpassed other ML models with a minimum AAPRE of 1.81%. Among EOS models, PC-SAFT has the best performance, followed by the ZJ model. Although, XGBoost faces several limitations: 1) XGBoost is sensitive to noisy data. While it has built-in mechanisms to handle outliers and noise, the algorithm can still be influenced by noise, potentially leading to suboptimal performance.2) The algorithm has many hyperparameters that must be tuned to achieve optimal performance. This complexity can be overwhelming for beginners and requires extensive experimentation and understanding to get right.3) Despite its regularization techniques, XGBoost can still overfit the training data, especially if not properly tuned. This is particularly true if the model is too complex or has insufficient training data. Moreover, Lv et al. [57] utilized different ML models in predicting hydrogen solubility in aqueous solutions, which were compared with cubic EOS (SRK, PR, RK, and ZJ).ML methods used include AdaBoost-DT, AdaBoost-SVR, GB-DT, GB-SVR, KNN, GEP, GP, and GMDH. Experimental datasets from the literature include pressure, temperature, and salt concentration as model inputs. It was revealed that AdaBoost-SVR outperformed other ML and cubic EOS models in hydrogen solubility prediction with RMSE of 0.000115 and R^2 of 0. 9973. Still, Adaboost-SVR has various limitations: 1) Sensitivity to noise and outliers. 2) Overfitting. 3) Both AdaBoost and SVR require careful tuning of multiple hyperparameters. When combined, the hyperparameter space becomes even more complex, making the tuning process more challenging and time-consuming.

Further, Zhou et al. [58] used the ML algorithm to predict H_2 solubility in different alcoholic solvents from experimental data. The model's inputs include pressure, temperature, critical pressure, and acentric factors. It was found that ANFIS2 outperformed other models such as LSSVM, ANN, SRK, PC-SAFT, and PR with R^2 of 0.998896, MSE of 6.9×10^{-4}, and RAD of 3.32%. Similarly, Jiang et al. [59] utilized ANFIS to estimate H_2 solubility in aromatic compounds, which was compared with LSSVM, ANN, MLPNN, CFFNN, GRNN, and RBFNN. The models' input includes temperature, pressure, critical pressure, critical temperature, and acentric factor. It was revealed that ANFIS outperformed other models with R^2 of 0.99664, RMSE of 0.0052, RD of 7.88%, RAE of 5.05%, MSE of 2.75×10^{-5}, and MAE of 0.0023. However, ANFIS faces various limitations: 1) Highly sensitive to the initialization of membership functions and rule parameters. Poor initialization can lead to suboptimal training and performance. 2) Overfitting.3) Inherently designed for unsupervised learning etc. Moreover, Tatar et al. [60] employed four ML algorithms (DT, RF, GB, and ET) to predict H_2 solubility in hydrocarbons. The models' input includes pressure, dimensionless pressure, dimensionless temperature, critical pressure, critical temperature, type of n-alkane, boiling point, and acentric factor. It was revealed that GB surpassed other ML models with high accuracy with R^2 of 0.9826 and RMSE of 0.0086. Though, GB has several limitations:1) Have many hyperparameters (like learning rate, number of trees, maximum depth of trees, etc.) that need to be tuned carefully. 2) Prone to overfitting, especially when the number of trees is large or when the trees are deep.3) Sensitive to noisy data etc. In addition, Hadavimoghaddam et al. [61] used GMDH to estimate H_2 solubility in hydrocarbons, which was compared with the GP algorithm. The models' input includes pressure, temperature, carbon weight percentage, H_2 weight percentage, molecular weight, and hydrogen/carbon ratio. It was found that GMDH outperformed GP in H_2 solubility estimation with R^2 of 0.9641 and RMSE OF 0. 053,302.Further, it was found that pressure, temperature, and H_2 weight percentage have highest impacts on H_2 solubility in hydrocarbons. Although, GMDH faces several limitations:1) Overfitting the data, especially when the dataset is small or has a lot of noise. 2) Choosing the right parameters (such as the number of layers, neurons per layer, and selection criteria) is difficult and require significant trial and error. Improper parameter selection can lead to suboptimal models.3) GMDH uses a local optimization approach, which means it may find local rather than global optima. This can sometimes result in suboptimal models that do not represent the best possible solution.

Hence, this paper developed a new hybrid ML algorithm, i.e., particle swarm optimization - mixed effects random forest (PSO-MERF), to predict H_2 solubility in aqueous solution for UHS purposes to solve prior ML limitations. This innovative methodology synergistically combines the robust predictive capabilities of MERF with the global optimization strength of PSO, addressing the multifaceted and complex nature of hydrogen solubility dynamics in variable salinity conditions. The MERF component of the model incorporates both fixed and random effects, enabling it to account for the inherent variability and hierarchical structure of environmental data, such as temperature, pressure, and salinity gradients, which traditionally challenge solubility models. Concurrently, the PSO algorithm optimizes the hyperparameters of the MERF model, ensuring the model is finely tuned to the peculiarities of the dataset, thus enhancing predictive accuracy and model generalizability. Further, another uniqueness of the PSO-MERF model is its ability to systematically navigate and model the complex, nonlinear relationships inherent in UHS systems, leveraging the collective intelligence of the PSO swarm to explore the hyperparameter space efficiently and the MERF's capacity to model complex data structures. This dual-strength strategy offers a significant advancement in the predictive modelling of hydrogen solubility, contributing to the enhancement of UHS design and management by providing reliable, high-fidelity predictions crucial for the operational efficiency and safety of storage facilities. To assess its effectiveness, PSO-MERF was compared with EOS and other ML models, which are KNN, XGBoost, and RF. This paper comprises four sections: introduction, methodology, results and discussion, and conclusions and recommendations.

2. Methodology

This section discusses four methods utilized in this paper to predict hydrogen solubility in aqueous solution, specifically K-Nearest Neighbors (KNN), Extreme Gradient Boosting (XGBoost), Random Forest (RF), and newly developed method, i.e., Particle Swarm Optimization-Mixed Effects Random Forest (PSO-MERF) algorithm.

2.1. Data collection

The data utilized in this study to predict H_2 solubility were collected from published literature. A comprehensive review of peer-reviewed journals, conference proceedings, and reputable industry reports was conducted to identify relevant studies pertaining to H_2 solubility in aqueous solutions. The search strategy involved keywords such as "H_2 solubility", "H_2 storage", "experiments H_2 solubility," and related terms to ensure the inclusion of pertinent literature. Additionally, databases such as Scopus, Web of Science, and Google Scholar were extensively searched to gather various studies from various sources. Several inclusion criteria were applied during the literature selection process to ensure the robustness and reliability of the data. Firstly, only studies that explicitly reported H_2 solubility values or provided sufficient data for H_2 solubility prediction were considered. This criterion ensured that the selected literature directly contributed to developing the H_2 solubility prediction model. Secondly, emphasis was placed on selecting studies that employed experimental techniques or rigorous simulation methods for H_2 solubility determination, thus prioritising accuracy and credibility in the collected data. Each identified study underwent a thorough review to extract relevant data points. Careful attention was paid to ensure data consistency and validity, with any discrepancies or ambiguities resolved by referring back to the original literature sources. In this paper, a total of 350 experimental datasets were collected from published literature [55,62–72], consisting of pressure (P) in bar, temperature (T) in K, and salinity (S) in (%wt) as inputs and hydrogen solubility (HS) in mole as output.The detailed data sources are summarized in Table 1. The selection of pressure, temperature, and salinity as input variables for predicting hydrogen solubility is firmly rooted in fundamental chemical principles and corroborated by extensive research in previously published literature [37,55,69,73]. These factors are the primary determinants of gas solubility in aqueous solutions, and their inclusion ensures a comprehensive and accurate predictive model. Our study aims to build upon this established knowledge base to provide a robust and reliable tool for predicting hydrogen solubility in varying conditions.

2.2. Data preprocessing

The collected data contains outliers detected by the box plot technique, as shown in Fig. 1. Addressing outliers effectively can ensure the ML models learn from the most representative data and achieve better performance on unseen data. In this paper, the Z-score method was used to remove outliers by capping technique, which involves replacing outlier values in the datasets within a predefined threshold value, i.e., the maximum and minimum outliers in the datasets were replaced with upper and lower limits, respectively, calculated by Z-score method. The capping technique is a better choice than the trimming technique when preserving all data points, which is essential and can mitigate their impact. In contrast, trimming involves completely removing outlier data points from the dataset [83,84]. Normalization is a vital preprocessing step in ML, paving the way for better model performance, interpretability, and overall robustness. Firstly, it helps to ensure that features are on a similar scale, preventing certain features from dominating others during training. Secondly, normalization aids in speeding up the convergence of iterative optimization algorithms, leading to faster training times. Moreover, it can improve the performance of models by making them more robust to outliers and noise in the data. Additionally, normalization facilitates the interpretation of model parameters since the scale of the features no longer affects the magnitude of the weights [85,86].In this paper, all the data were normalized from 0 to 1 using Eq. (1) before model training. Descriptive statistics for the data used for training and testing the models are shown in Table 2. The correlation heat map of the data used in model training and testing is given in Fig. 2, in which pressure and temperature positively correlate with hydrogen solubility. In contrast, salinity has a negative correlation with hydrogen solubility.

$$y_i' = \frac{y_i - y_{\min}}{y_{\max} - y_{\min}} \tag{1}$$

Where y_i', y_i, $y_{\min}$, $y_{\max}$ are the normalized value of y_i, the value to be normalized, the minimum value of y_i, and the maximum value of y_i, respectively.

2.3. K-nearest neighbors (KNN)

KNN is a supervised ML technique utilized for regression and classification problems [87–89]. It was first introduced by Fix and Hodges [90]. It's a lazy, non-parametric learning approach that doesn't learn a particular model throughout training or make any assumptions about the distribution of the underlying data. Rather, it records all accessible data points and generates outputs by comparing fresh data points to pre-existing ones [91–93]. Given a dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ where x_i represents the feature vector of the ith sample and y_i represents its corresponding label, then the KNN algorithm works as follows: For classification to classify new data (x_{new}), the algorithm computes the distance between x_{new} and all other data points in the training set. It then selects the k nearest data points (nearest neighbors) to x_{new} based on some distance metric (usually Euclidean distance). After that, it assigns the class label by a majority vote among its k nearest neighbors. In contrast, for regression, it computes the average (or weighted average) of the predicted values of its k nearest neighbors. The most commonly used distance metric in KNN is the Euclidean distance, which is calculated as [94,95]:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{m} (x_{i,l} - x_{j,l})^2} \tag{2}$$

Where $x_{i,l}$ and $x_{j,l}$ are the components of the feature vectors x_i and x_j, respectively, and m is the dimensionality of the feature space.

However, this method has several disadvantages, such as being computationally expensive during testing, especially for large datasets, sensitive to the choice of distance metric, and performance can degrade with high-dimensional data.

2.4. Extreme gradient boosting (XGBoost)

XGBoost stands out as a powerful and efficient implementation of gradient boosting, a ML technique known for its effectiveness in various prediction tasks. Its speed and strong performance have made it a popular choice, particularly in competitions focused on structured data analysis. It is designed to be distributed efficiently and has the flexibility to handle various types of predictive modelling problems. It builds a group of decision trees (ensemble learning) to create a final model, with each tree focusing on areas where previous ones struggled (gradient boosting). This approach helps prevent overfitting (regularization) and

Table 1
Details on utilized data sources.

Pressure range (bar)	Temperature range (K)	Salinity range (% wt)	Solubility range (mol fraction)	Data points	Reference (s)
29.272–121.706	323.18–372.73	0–1	0.000309–0.001544	37	Chabab et al. [74]
19.884–216.205	323.18–372.78	1–3	0.000201–0.002132	17	Torín-Ollarves and Trusler [75]
11.64–229.72	323.15–423.15	2.5–5	0.000127–0.004557	25	Jáuregui-Haza et al. [64]
5–110	273.15–423.15	0	0.002016–0.9131	40	Kling and Maurer [65]
5–110	333.15–363.15	0	0.03828–0.68979	18	Ruetschi and Amlie [76]
5–110	273.15–423.15	0–1	0.027–0.97881	54	Wiebe and Gaddy [77]
5–25	273.15–373.15	1–3	0.01336–0.14601	23	Crozier and Yamamoto [78]
5–25	273.15–373.15	3–5	0.00668–0.09292	23	Gordon et al. [79]
4.6–101.4	273.15–373.15	0–5	0.01291–0.791533	53	Morrison and Billett [80]
1.01325	278.2–298.2	0–3	0.00061–0.00084	42	Braun [81]
1.01325	274.15–303.15	0–4	0.000681–0.000817	18	Wiesenburg and Guinasso [82]
Total				350	

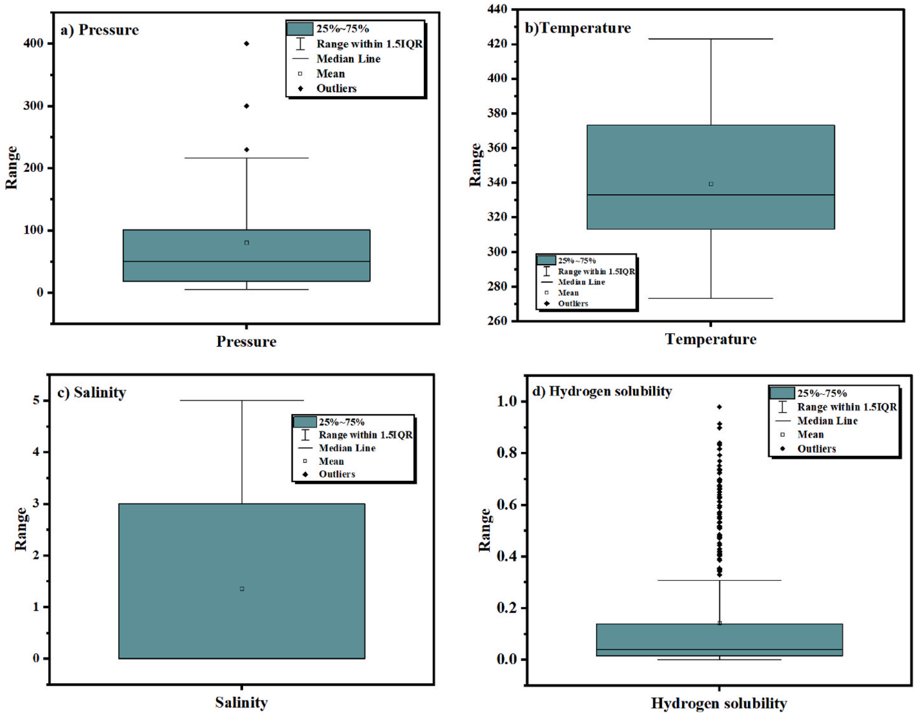


Fig. 1. Data outliers in using boxplots.

Table 2
Descriptive statistics of the utilized data.

Statistical parameters	Inputs			Output
	Pressure (P) in (bar)	Temperature (T) in (K)	Salinity (S) in (% wt)	Hydrogen solubility (HS) in (mole fraction)
Mean	49.6743	378.5691	1.6321	0.1931
Standard deviation	51.3967	47.2391	1.8954	0.2793
Minimum	3.9	273.15	0	0.000121
25%	15	303.15	0	0.01857
50%	25	333.15	1	0.0703
75%	66.6345	372.75	3	0.2883
Maximum	233.6523	473.14	5	0.9788

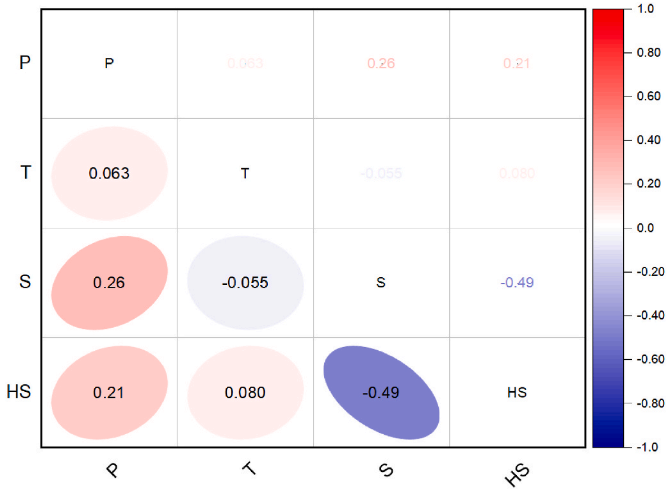


Fig. 2. The correlation heatmap between inputs and output.

works well with big data (scalability). XGBoost can handle missing values and offers some interpretability through feature importance, making it a flexible and user-friendly tool for various ML projects to tackle regression, classification, and ranking tasks. The mathematical foundation of XGBoost, incorporating gradient boosting with regularized objectives, provides a robust framework for predictive modelling.

Let’s consider a dataset (D) with n data points and m features

$$D = \{(x_i, y_i : i = 1, \ldots, n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R})\} \tag{3}$$

Suppose $\widehat{y}_i$ is the predicted value given as [96,97]:

$$\widehat{y}_i = \sum_{i=1}^{K} f_k(x_i), f_k \in F \tag{4}$$

Where K represent regression number trees, x_i stands for sample i features, F is regression trees space are, and f_k is the weight of the leaf for node j. In XGBoost, the goal is to minimize an objective function, which can be expressed as [96,97]:

$$Obj = \sum_{i=1}^{n} l(y_i, \widehat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{5}$$

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2 \tag{6}$$

Where $\sum_{i=1}^{n} l(y_i, \widehat{y}_i)$ stands for training loss function, γ is the degree of regularization, K stands for the number of trees, λ is the regularization coefficient, ω represents leaf weight, and $\Omega(f)$ serves to constrain the model’s complexity, mitigating the risk of overfitting. To achieve the best model performance, f_t is added in the objective function in Eq. (5), in which $\widehat{y}_i^t$ and *i-th* stands for model output and the number of iterations, as presented in Eq. (7) [96,97].

$$Obj^t = \sum_{i=1}^{n} l\left(y_i, \widehat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \tag{7}$$

Incorporating the greedy algorithm enhances the model’s effectiveness by integrating feature transformation. Subsequently, the performance of the model is optimized in every iteration through the reduction of the fitness function expressed as [96,97]:

$$\widehat{y}_i^{(t)} = \widehat{y}_i^{(t-1)} + f_t(X_i) \tag{8}$$

2.5. Random forest (RF)

RF is a powerful ensemble learning method that leverages the combined strength of multiple decision trees to enhance prediction accuracy and achieve greater robustness in its results [98–101]. It belongs to the family of bagging algorithms and is known for its effectiveness in classification and regression tasks [102–104]. It was first discovered by Breiman [105] and later modified by Liaw and Wiener [106]. It achieves its accuracy by combining the predictions of multiple decision trees, overcoming the limitations of individual trees by introducing randomness and averaging. RF builds multiple decision trees during training. Each tree is trained on a random subset of the training data, drawn with replacement (bootstrap samples). Bootstrap sampling involves randomly selecting data points from the original dataset to form a new dataset of the same size. Since sampling is done with replacement, some data points may be selected multiple times, while others may not. The model's performance can be estimated using these Out-of-Bag (OOB) samples, eliminating the need for a dedicated validation set. This randomness in the training process helps to mitigate overfitting by introducing variation among the trees. It also contributes to the unpredictability of the model, making it more robust and less prone to memorising the training data [99,107–109]. In addition to the bootstrap sampling strategy, random forest introduces randomness in the training process in two main ways:1) Random selection of features: At each decision tree node, only a random subset of features is considered for splitting. This helps to decorate the trees in the forest and prevent them from all making the same splits.2) Random generation of training samples: The training samples for each tree are randomly generated with replacements from the bootstrap sample. This means that each tree sees a slightly different subset of the data, leading to diversity among the trees in the forest. To create robust and accurate predictions, RF leverages a combination of techniques: bootstrap aggregating (bagging), random feature selection, and random training sample generation. This ensemble approach builds a collection of diverse decision trees, each trained on a random subset of features and data points. By combining the predictions from these trees, RF achieves improved generalizability and reduces the risk of overfitting [110–112]. For classification tasks, the class predicted by each tree is tallied, and the class with the most votes is assigned as the final prediction [103]. In regression problems, the model combines the predictions from all trees to arrive at a final prediction, typically by averaging them by Eq. (9) [95,113].

$$\overline{f}_{rf}^{B} = \frac{1}{B}\sum_{b=1}^{B} T(x, O_b) \tag{9}$$

Where $\overline{f}_{rf}^{B}$ stands for average tree output, B is the trees, and $T(x, O_b)$ is the output of each tree.

2.6. Particle swarm optimization-mixed effects random forest (PSO-MERF)

The PSO-MERF model represents an advanced integration of optimization and machine learning techniques designed to enhance predictive modelling in complex datasets. This combination leverages the global search capability of particle swarm optimization (PSO) to fine-tune the parameters of a mixed effect random forest (MERF). This hybrid model integrates the strengths of Random Forests and Mixed Effects models. In the PSO-MERF model, PSO is used to optimize the hyperparameters of the MERF model. This includes, but is not limited to, the number of trees in the Random Forest, the depth of the trees, the minimum samples required to split an internal node, and the parameters governing the mixed effects model component, such as the variance components of the random effects. These two components and functionalities of PSO-MERF are discussed in detail in the following subsections.

2.6.1. Particle swarm optimization (PSO)

PSO is a well-regarded optimization algorithm that draws inspiration from the collective movement patterns observed in bird flocks and fish schools [114–118]. PSO involves a population of candidate solutions, referred to as particles, which move through the search space, looking for the optimal solution. The algorithm starts with an initial population of particles randomly distributed across the search space. Each particle acts as a candidate solution in the optimization problem. These particles are equipped with a position vector representing their current exploration point in the hyperparameter space and a velocity vector indicating their direction and magnitude of movement for the next iteration. Each particle is represented by a position vector within the search space. This position vector indicates the current solution being evaluated. The particle also possesses a velocity vector, influencing the direction and magnitude of its movement in the search space during the iterative optimization process. During each iteration (or generation) of the algorithm, each particle adjusts its velocity and position based on its own experience and the experiences of its neighbors. This adjustment is guided by two key components: the particle's best-known position (personal best) and the best-known position found by its neighbors (global best) [119–121].

Each particle's velocity at iteration i is determined by the following equation [117,119,122]:

$$v_i(t+1) = w.v_i(t) + c_1.r_1.(p_{i,best} - x_i(t)) + c_2.r_2.\left(p_{g,best} - x_i(t)\right) \tag{10}$$

Where $v_i(t)$ is the velocity of particle i at iteration t,w is the inertia weight that controls the impact of the previous velocity,c_1 and c_2 are acceleration coefficients representing the cognitive and social components, respectively, r_1 and r_2 are random values sampled from the uniform distribution in the range [0,1], $p_{i,best}$ is the personal best position of particle i, $p_{g,best}$ is the global best position found by any particle, and $x_i(t)$ is the position of particle i at iteration t. After updating the velocities, the positions of the particles are updated using the following equation [118,119,122,123]:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{11}$$

The algorithm runs for a set number of iterations or until a stopping condition is reached (e.g., achieving a satisfactory solution).

2.6.2. Mixed effects random forest (MERF)

MERF combines the predictive power of random forests with the ability to account for both fixed and random effects in hierarchical or grouped data, making it highly suitable for complex datasets where individual and group-level variability must be considered. This approach can be formulated as [124–129]:

$$\begin{aligned} &y_i = f(X_i) + Z_i u_i + \varepsilon_i, \\ &u_i \sim N(0, G), \varepsilon_i \sim N(0, R_i), i = 1, 2, \ldots, K \end{aligned} \tag{12}$$

Where $y_i = (y_{i1}, \ldots, y_{in})$ stands for vector output for cluster i, X_i and Z_i represents design matrices for fixed effects and random forest, respectively, u_i represents unknown vector for random effects, and ε_i stands for residual vector. The constant part $f(X_i)$ is computed by an RF. In the MERF model it is assumed that the data from the clusters are independent and u_i as well as ε_i. In addition, a diagonal matrix ($R_i = \sigma^2 I_{in}$) is needed to ensure that the residual structures and sizes are identical across all clusters. Steps for MERF implementation are as follows [125–127,130]:

Step 1: Initialization of random effects coefficients at zero, $\sigma_R^2 = 1$, G as identity matrix (G = I_m). Iteration number k is zero [124,128,131].

Step 2: i) Updating k to k = k+1, then the random part is subtracted from the output: $y^*_{i(k)} = y_i - Z_i\widehat{u}_{i(k-1)}$ [124,131].

ii) The RF model is trained based on $y^*_{i(k)}$ utilizing the bagging method.

iii) Predict for unseen data points in each observation j using trees that haven't seen j during training [124,131].

iv) Then update u_i [124,131]:

$$\widehat{u}_{i(k)} = \widehat{G}_{(k-1)} Z_i^T V_{i(k-1)}^{-1} \left(y_i - \widehat{f}(X_i)_{(k)}\right), i = 1, \ldots, n \tag{13}$$

Where $V_{i(k-1)} = Z_i \widehat{G}_{(k-1)} Z_i^T + \widehat{\sigma}^2_{R(k-1)} I_{ni}$.

Step 3: Refine the covariance matrix G and the estimate σ_R^2 using the latest residual values [124,131].

$$\widehat{\sigma}^2_{R(k)} = \frac{1}{N}\sum_{i=1}^{n} \widehat{\varepsilon}^T_{i(k)} \widehat{\varepsilon}_{i(k)} + \widehat{\sigma}^2_{R(k-1)} \left(n_i - \widehat{\sigma}^2_{R(k-1)}.trace\left(V_{i(k-1)}\right)\right) \tag{14}$$

$$\widehat{G}_{(k)} = \frac{1}{n}\sum_{i=1}^{n} u^T_{i(k)} u_{i(k)} + \widehat{G}_{(k-1)} - \widehat{G}_{(k-1)} Z_i^T V^{-1}_{i(k-1)} Z_i \widehat{G}_{(k-1)} \tag{15}$$

Where $\varepsilon_{i(k)} = y_i - \widehat{f}(X_i)_{(k)} - Z_i\widehat{u}_{i(k)}, i = 1, 2, \ldots, K$ and is not defined by random forest and random effects estimates.

Step 4: The process continues by iteratively performing steps 2 and 3 until a stopping criterion, based on a measure of model fit, is met [124,129,131].

$$GLL(f, u|y) = \sum_{i=1}^{n} (y_i - f(X_i) - Z_i u_i)^T R_i^{-1} (y_i - f(X_i) - Z_i u_i) + u_i^T D^{-1} u_i + \log|G| + \log|R_i|\bigg) \tag{16}$$

If we consider the generalized likelihood criterion (denoted as GLL_k) after the kth iteration, the algorithm is said to have converged when [124,127,129,131]:

$$\frac{|GLL_k - GLL_{k-1}|}{GLL_{k-1}} < \delta \tag{17}$$

For some $\delta > 0$.

In this context, a relative convergence criterion is preferred over an absolute one. The absolute value of the generalized likelihood criterion (GLL) can vary significantly depending on the specific problem and dataset. Therefore, basing the convergence solely on the actual GLL value would be meaningless. The steps for PSO-MERF implementation are outlined as follows.

1. **Data partitioning:** After data normalization, the data were divided into training and testing groups.70% of the data was used for training, while 30% was used for testing. K-fold cross-validation is a widely used method for assessing the performance of machine learning models. The dataset is divided into k distinct subsets, or folds, for this technique. Common choices for k are 5 or 10. The process involves using k-1 folds to train the model while the remaining fold is set aside for validation. This procedure is repeated k times, with each fold being used exactly once as the validation set. The choice of k significantly impacts the evaluation process. Fewer folds (e.g., k = 5) result in a higher bias and lower variance but reduce computational cost. Conversely, more folds (e.g., k = 10) decrease bias and increase variance, leading to more computationally intensive evaluations. Tenfold cross-validation is frequently chosen to strike a good balance between bias and variance [11]. The final performance metric is calculated by averaging the results obtained from each iteration, providing a comprehensive measure of the model's effectiveness across different subsets of the data. For this paper, tenfold cross-validation was utilized.
2. **Initialization of PSO parameters:** PSO parameters such as the number of particles, maximum iterations, inertia weight, acceleration coefficients, and bounds for the search space were defined for PSO execution. Proper initialization of parameters is essential for PSO-MERF to achieve efficient exploration, avoid local optima, and adapt to diverse datasets. This allows the algorithm to learn optimal parameter settings and achieve accurate predictions for specific applications.
3. **Initialization of MERF parameters:** MERF parameters such as the kernel type, regularization parameter (C), and kernel parameters (gamma for RBF kernel) were defined for PSO execution. This initialization of MERF parameters randomly is a valuable strategy by aiding in exploration, adaptation, and robustness
4. **Initialization of particle positions and velocities:** Particle positions and velocities were initialized randomly for model development. This is essential for efficiently navigating and utilizing the search area. Ultimately, the best initialization method depends on your specific problem and its characteristics. Experimentation with different approaches might be necessary to find the most effective strategy for your application
5. **Evaluation of fitness function:** This stage involves evaluating the fitness of each particle's position using MERF, followed by training the MERF model with the training data using the parameters represented by each particle's position. After that calculation, the fitness of each particle based on the performance of the MERF model on the training data such as RMSE and MAE
6. **Update personal and global best positions:** Following the evaluation by a fitness function, both the personal best position (pbest) and the global best position (gbest) are updated for each particle. The pbest update considers the particle's current fitness value compared to its previous best. In contrast, the gbest update compares the fitness of all particles to identify the overall best position encountered so far.
7. **Update particle velocities and positions:** In this stage, the position and velocity of each particle based on its current velocity, personal best position, and global best position were updated. Then, after reaching a maximum number of iterations, the model stopped.
8. **Final model training:** After optimization, the MERF model was trained using the best parameters obtained from the global best position.
9. **Model evaluation:** The developed model was evaluated by the unseen data based on three selected criteria: R, RMSE, and MAE. In case the initial result is unsatisfactory, steps 2 through 7 are iterated upon until a desirable result is achieved
10. **Analysis and interpretation:** The results were analyzed and interpreted, and a conclusion was made regarding the relationship between the input variables and the target variable. The flowchart for PSO-MERF is summarized in Fig. 3.

2.7. Hyperparameter tuning

Hyperparameter tuning, also known as model tuning, is a critical step in the ML aimed to find the optimal settings for a model's hyperparameters [95,132,133]. These are essentially dials and levers that control the learning process and overall behavior of the model, but unlike regular parameters that are learned during training, they need to be set before training starts [133,134]. There are several methods used for hyperparameter tuning:1) Random search: Samples random

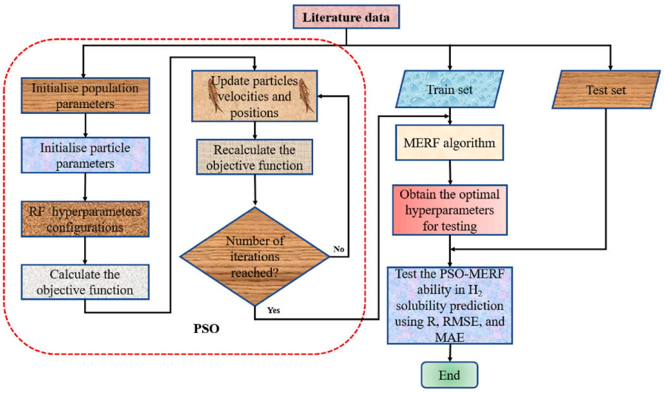

Literature data
Initialise population parameters
Initialise particle parameters
RF hyperparameters configurations
Calculate the objective function
Update particles velocities and positions
Recalculate the objective function
Number of iterations reached?
No
Yes
PSO
Train set
Test set
MERF algorithm
Obtain the optimal hyperparameters for testing
Test the PSO-MERF ability in H_2 solubility prediction using R, RMSE, and MAE
End

Fig. 3. PSO-MERF flowchart.

combinations of parameters, which is faster but less likely to find the best settings.2) Grid search: Tries every possible combination of parameter values, which can be computationally expensive for large grids.3) Bayesian optimization: Uses statistical methods to identify promising regions of the parameter space, focusing the search on areas with higher chances of finding optimal settings [135,136]. This paper used the random search for hyperparameter tuning because it searches samples of random combinations from the parameter space, avoiding the exhaustive evaluation of every possible combination. This makes it significantly faster, especially for models with many hyperparameters or large ranges. Also, unlike other methods like Bayesian optimization, random search only requires defining the ranges for each parameter, making it less prone to overfitting. Further, random search explores a wider variety of parameter combinations than grid search, which its predefined grid can limit. This allows it to discover unexpectedly suitable configurations potentially. Hyperparameters utilized in this paper are shown in Table 3.

3. Results and discussion

3.1. Model performance indicators

Model performance indicators are metrics utilized to assess and quantify the accuracy, reliability, and efficacy of a predictive model. These indicators help in assessing how well a model is performing, based on its ability to make accurate predictions or classifications. The choice of performance indicators often depends on the model type (e.g., regression, classification) and the specific objectives of the modelling exercise. In this paper, three models' performance indicators, which are correlation coefficient (R), root mean square error (RMSE), and mean absolute error (MAE), expressed in Eqs. (18)–(20) [137], respectively, were utilized in models' comparisons during hydrogen solubility prediction. Usually, when MAE and RMSE values are near zero, it typically indicates high model accuracy, while for R, when values are close to one, it indicates the model performed excellently. Hence, the best model in this study was chosen based on minimum errors (RMSE and MAE) and high R during training and testing. In this paper, the Python 3.12.2 version was used for the model development.

Table 3
Optimal hyperparameters for model developments.

ML models	Hyperparameters	Range values	Used values
KNN	n_neighbors	[1,30]	22
RF	n_estimators	[500,2500]	1300
	max_depth	[1,30]	15
	min_samples_leaf	[1,70]	45
	min_samples_split	[1150]	97
XGBoost	n_estimators	[500,2700]	1800
	max_depth	[1,50]	25
	subsample	[0.1,2]	0.9
	colsample_by tree	[0,1]	0.9
	learning_rate	[0,1]	0.7
MERF	min_samples_leaf	[1,20]	9
	min_samples_split	[1,15]	13
	max_depth	[1,30]	10
	Number of trees	[100,1500]	350

$$R = \frac{\sum_{i=1}^{N}\left(y_{act} - \overline{y_{act}}\right)\left(Y_{prd} - \overline{Y_{prd}}\right)}{\left(\sqrt{\sum_{i=1}^{N}\left(y_{act} - \overline{y_{act}}\right)^2}\right)\left(\sqrt{\sum_{i=1}^{N}\left(Y_{prd} - \overline{Y_{prd}}\right)^2}\right)} \tag{18}$$

$$RMSE = \sqrt{\left(\frac{1}{N}\sum_{i=1}^{N}\left(y_{act} - Y_{prd}\right)^2\right)} \tag{19}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{n}\left|y_{act} - Y_{prd}\right| \tag{20}$$

Where y_{act} is actual value, $\overline{y_{act}}$ is average actual value, Y_{prd} is predicted value, $\overline{Y_{prd}}$ represents average forecasted value, and N stands for the amount of data.

3.2. Models' statistical analysis

Table 4 shows the performance metrics of four different ML models applied to the estimation of hydrogen solubility, including PSO-MERF, XGBoost, KNN, and RF. During training, PSO-MERF, XGBoost, KNN, and RF exhibit exceptionally high R values of 0.9997,0.9968,0.9899, and 0.9819, respectively, which implies that the models have captured the underlying data generation process with high fidelity during training. At the same time, in testing, there was a minor drop of R by 0.15% for PSO-MERF, indicating that the model generalizes well to unseen data. For other models, the decrease of R values between training and testing was 0.85% for XGBoost,5.73% for KNN, and 7.61% for RF. For errors, PSO-MERF has RMSE of 0.00033 on training data, which is the lowest among all models, suggesting very accurate predictions with minimal error, while in testing, the RMSE increases slightly to 0.0015 for PSO-MERF, which is still the lowest among the models. For XGBoost, KNN, and RF, the RMSE were 0.00099 and 0.0031,0.0025 and 0.0297, 0.0067 and 0.0636, throughout the training and testing phases, respectively. Also, the training MAE of 0.00041 for PSO-MERF indicates minimal average prediction error. This low MAE shows the model has high accuracy in the training phase. At the same time, it achieves a testing MAE of 0.00091, which again is the lowest compared to other models, reinforcing its ability to generalize from training to testing data. For other models, MAE were 0.00099 and 0.00252,0.0025 and 0.00614, 0.0067 and 0.00918, for XGBoost, KNN, and RF, during training and testing, respectively. Furthermore, PSO-MERF took less computational time (1.01s) than other models, as shown in Tables 4 and ie., 59.9% less than XGBoost, 69.7% less than KNN, and 80.8% less than RF. This result suggests that the PSO-MERF exhibits superior convergence speed and accuracy compared to the other ML models. PSO-MERF generally outperforms the other models in all metrics for training and testing. It has the highest R, indicating the best fit to the data; the lowest RMSE (Fig. 4), showing the smallest average error in predictions; and the lowest MAE (Fig. 5), demonstrating its ability to maintain prediction accuracy consistently. The minimal differences in metrics between training and testing for PSO-MERF suggest superior generalizability and indicate an effective balance between bias and variance, likely due to the optimized hyperparameters via PSO. In contrast, other models, especially RF, show a pronounced decline in performance from training to testing, which can indicate overfitting to the training data and a lack of generalization to new data.

Table 4
Models' statistical results.

Models	Training data			Testing data			Computational time (Seconds)
	R	RMSE	MAE	R	RMSE	MAE	
PSO-MERF	**0.9997**	**0.00033**	**0.00041**	**0.9982**	**0.0015**	**0.00091**	**1.01**
XGBoost	0.9968	0.00099	0.00053	0.9883	0.0031	0.00252	2.52
KNN	0.9899	0.0025	0.00070	0.9332	0.0297	0.00614	3.33
RF	0.9819	0.0067	0.00095	0.9072	0.0636	0.00918	5.25

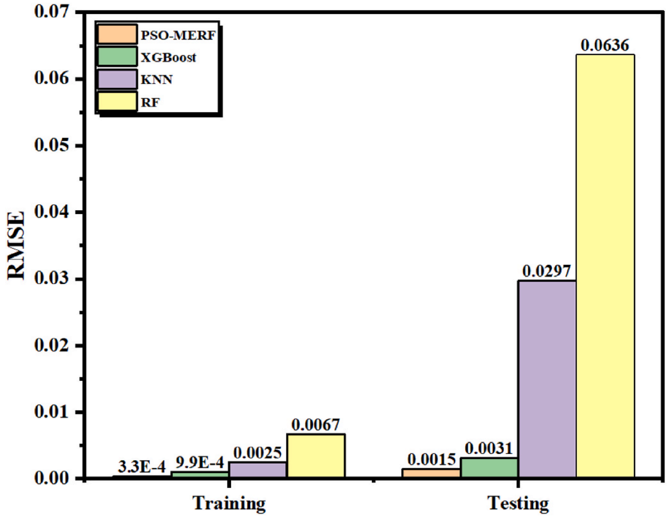


Fig. 4. RMSE errors for the utilized models.

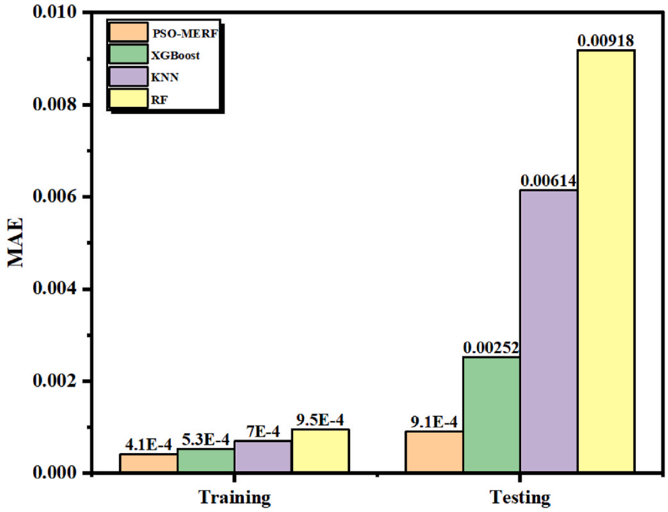


Fig. 5. MAE errors for the utilized models.

3.3. Models' comparisons

From Fig. 6, the x-axis represents the experimental data and the y-axis represents the predicted H_2 solubility. A perfect fit would be a straight diagonal line from the bottom left (0,0) to the top right (1,1). The closer the fitting line is to this diagonal, the better the model's performance. In Fig. 6 (a), the PSO-MERF model shows a nearly perfect fit to the data during training, with an R-value of 0.9997, indicating that the model captures 99.97% of the variance in the training data. During testing, the R-value was 0.9982, slightly lower, but still indicates a perfect fit to the unseen data. The proximity of the training and testing fitting lines to the fitting lines suggests that the model has excellent predictive accuracy and generalises the data well. For the XGBoost model in Fig. 6 (b), the XGBoost training fitting line is close to the diagonal line. This indicates that XGBoost performed well on the training data. However, the XGBoost testing fitting line deviates more from the diagonal line than the PSO-MERF testing fitting line. This suggests that XGBoost may have to overfit the training data more than PSO-MERF. For the KNN model in Fig. 6 (c), the KNN training fitting line is close to the diagonal line but deviates more than PSO-MERF and XGBoost in the range. This indicates that KNN performed well on the training data but not as well as PSO-MERF or XGBoost. The KNN testing fitting line deviates significantly from the diagonal line, particularly for values above 0.6 on the x-axis. This suggests that KNN performed poorly on the testing data and significantly overfitted the training data. For the RF model in Fig. 6 (d), the RF training fitting line deviates considerably from the diagonal line across the entire x-axis range compared to other models. This indicates that RF did not perform well on the training data. The RF testing fitting line also deviates significantly from the diagonal line and to a similar extent as the training fitting line. This suggests that RF performed poorly on the training and testing data and did not learn the underlying relationship between the experimental data and the predicted solubility, especially for small H_2 solubility values. In general, based on the fitting lines, PSO-MERF appears to be the best-performing model. It has the closest fitting lines to the diagonal line on both the training and testing data, indicating good performance on both sets. XGBoost also performed well on the training data, but its performance appears to have degraded more on the testing data than PSO-MERF. KNN and RF performed poorly, with KNN showing significant overfitting and RF failing to learn the relationship between the data.PSO-MERF likely achieved superior performance due to a combination of factors. PSO, the optimization algorithm within PSO-MERF, excels at searching through complex spaces for optimal solutions. This might have helped PSO-MERF capture the underlying relationships within the solubility data during training. Additionally, MERF, the regression component, might have been particularly adept at modelling these relationships, leading to accurate predictions on both the training and testing sets.

Also, as shown in Fig. 7, Taylor's diagram was utilized in the models' comparison performance. Based on Fig. 7 it indicates that the PSO-MERF model (triangle symbol), followed closely by the XGBoost model (square symbol), demonstrates the highest correlation and a standard deviation most close to the reference data, with both models also indicating a lower root mean square difference (RMSD) compared to KNN and RF models. This suggests that PSO-MERF and XGBoost have the most accurate and reliable predictions regarding both pattern and magnitude of variability. The RF model (diamond symbol) shows a lower correlation and a more significant deviation from the reference data than the PSO-MERF and XGBoost, suggesting it is the least accurate model among those presented. The KNN model (circle symbol) has a correlation and standard deviation lower than the PSO-MERF and XGBoost but better than the RF model. Therefore, based on this diagram, the PSO-MERF model performs the best in predicting H_2 solubility, closely followed by XGBoost, with both significantly outperforming the KNN and RF models.

A violin plot is a data visualization tool that combines aspects of a box plot and a kernel density plot [138]. It is particularly useful for visualizing data distribution and comparing multiple groups or models.

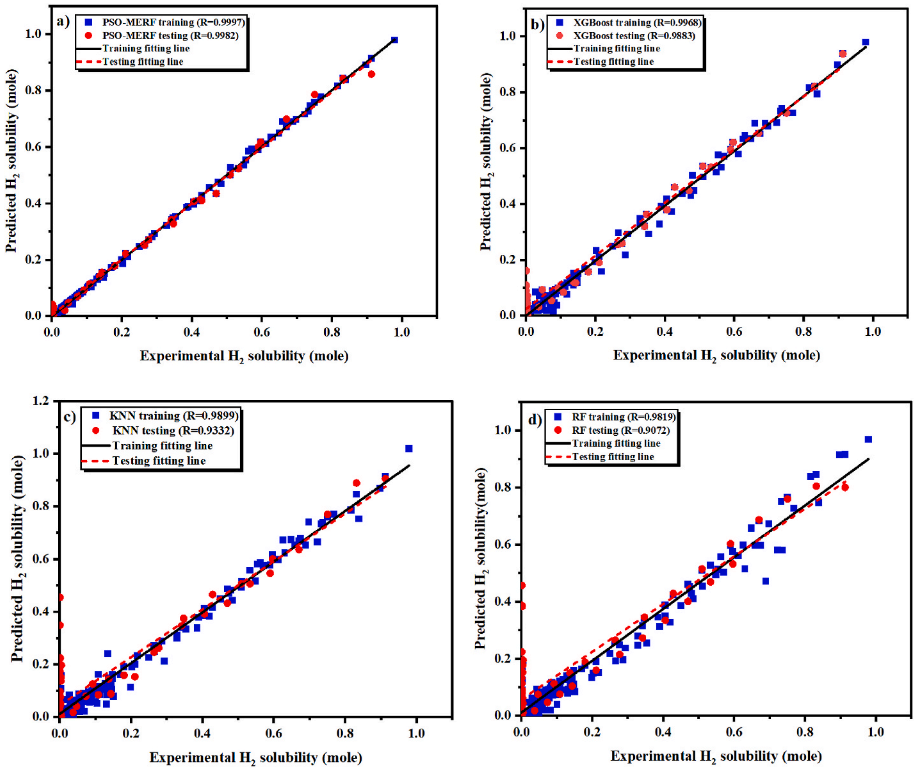


Fig. 6. Crossplots for the utilized models.

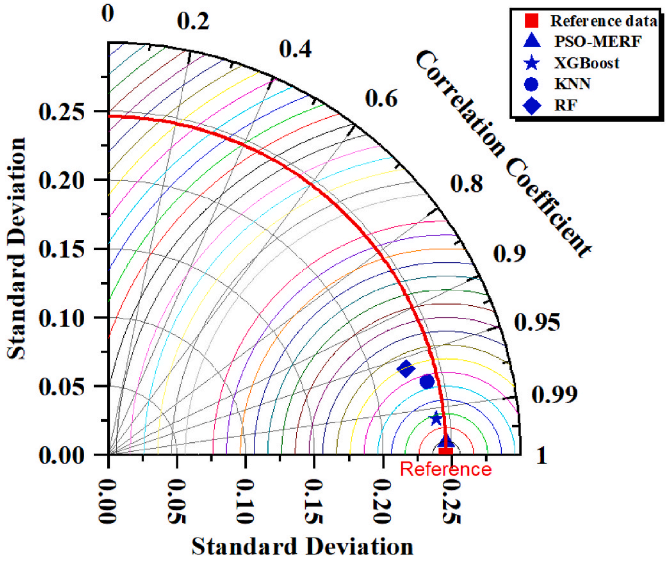


Fig. 7. Taylors diagram for models' assessments.

Violin plots provide a more detailed view of the distribution, showing multimodal distributions and the shape of the data. The key components and features of a violin plot include:1) Kernel density estimate (KDE): The width of the violin plot represents the density of the data at different values. Wider sections indicate a higher density of data points, while narrower sections indicate a lower density.2) Median and quartiles: Similar to a box plot, a violin plot typically includes a marker for the median of the data and lines or dots indicating the quartiles (25th and 75th percentiles). 3) Symmetry: The plot is symmetric, with the KDE mirrored on both sides of the central axis.4) Multiple groups: Violin plots can compare data distribution across multiple categories or groups. Each group is represented by a separate violin [139,140].In this paper, a violin plot assessed the models' performances in H_2 solubility prediction. Notably, the PSO-MERF model achieved an exceptional match, as evidenced by the precise alignment of the median (depicted as white dots within the violin plot) with experimental data. From Fig. 8, the PSO-MERF model predictions for the lower and upper percentiles (5th and 95th, marked by thin black lines) and for the quartiles (25th and 75th, indicated by thicker lines) displayed the most accurate reflection of the experimental data distribution, surpassing other models. XGBoost model showcased comparable proficiency, closely emulating the PSO-MERF model performance across all statistical indicators. Conversely, the KNN tended to overpredict the lower tail and underpredict the upper tail of the H_2 solubility range, while RF models generally underestimated the lower percentile. Regarding the distribution shape illustrated by the violin plot, representing the probability

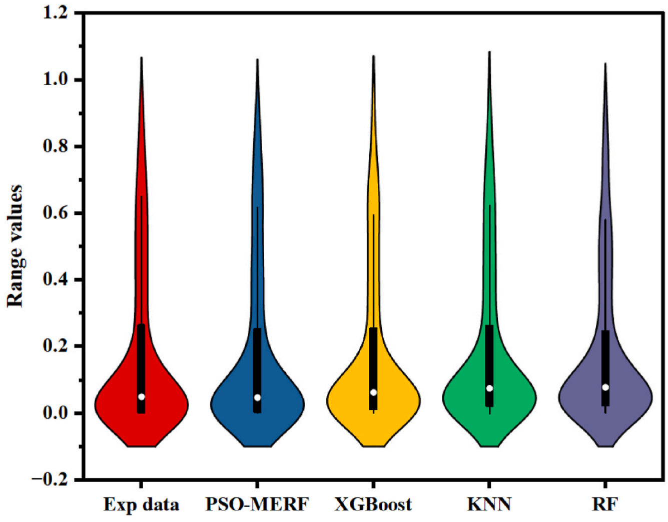


Fig. 8. Models' comparison using violin plot.

density function (PDF), the PSO-MERF model's PDF conformed most closely to the experimental H_2 solubility data distribution. This was followed by the XGBoost and KNN models. In contrast, the RF model PDF deviated significantly from the experimental H_2 solubility data distribution, resulting in a markedly poorer fit.

Moreover, PSO-MERF was compared with empirical EOS models' to predict hydrogen solubility in an aqueous solution. The utilized EOS models' include SRK [42,141], PR [42,141], RK [42,142], and ZJ [42,143]. As shown in Table 5, PSO-MERF outperformed the EOSs by having high R and minimum errors. The R obtained by PSO-MERF was 0.9997, RMSE of 0.000091 and MAE of 0. 00043. Among the EOS models in predicting hydrogen solubility in aqueous solution, PR EOS surpassed other models with R of 0.9992, RMSE of 0.00043, and MAE of 0. 0000011. Further, the ZJ EOS model has poor results compared to other EOS models with R of 0.9900, RMSE of 0.0063, and MAE of 0. 0011. The order rank of performance of all the models was PSO-MERF > PR > SRK > RK > ZJ. With these results, it can be recommended that PSO-MERF can be adopted as an alternative way of predicting hydrogen solubility in aqueous solution.

3.4. Trend analysis

Trend analysis was conducted to gain valuable insights into the model behavior and ensure its continued reliability and effectiveness. It examines whether the models" predictions maintain accuracy, reliability, and consistency as the underlying data changes. This can be particularly important in dynamic environments where data distributions can shift or in cases where models are used for long-term predictions. For this paper, the trend analysis was conducted for variations of pressure and temperature with reference to predicted hydrogen solubility by the PSO-MERF model and experimental hydrogen solubility so that their variations can be analyzed. Fig. 9 shows the changes in hydrogen solubility by increasing temperature up to 473 K under different pressures of 50 bar, 100 bar, 180 bar, and 233.65 bar. It was found that the model hydrogen solubility decreases with an initial increase in temperature due to the exothermic nature of hydrogen absorption, where lower temperatures favour the formation of stable hydrogen-material bonds, releasing energy. However, as the temperature rises, various factors contribute to increased solubility. These include enhanced atomic vibrations creating more interstitial spaces for hydrogen, changes in the material's phase that may offer additional hydrogen storage capacity, and overcoming kinetic barriers that impede hydrogen absorption at lower temperatures. Additionally, at high temperatures, especially under high-pressure conditions, hydrogen may exhibit different behaviors, such as transitioning to a plasma state, which can also influence its solubility in the host material. This complex interplay of thermodynamic and kinetic factors results in various materials' observed nonlinear temperature dependence of hydrogen solubility. Moreover, Fig. 10 shows the variations of hydrogen solubility by increasing pressures up 233.65 bar at different temperatures of 273.15 K,373.15 K, and 473 K. It shows that the hydrogen solubility predicted by the model increases with an increase in pressure, which obeys Henry's law and matching the experimental hydrogen solubility. Further, Fig. 10 shows that the model hydrogen solubility decreases with increased salinity, which obeys physical laws and matches the experimental data. From these observations, it can be concluded that the developed model obeys the physical laws and ensures the model's robustness and sensitivity to different input variations, which match findings from previous researchers such as Lv et al. [57] and Ansari et al. [42].

Table 5
PSO-MERF comparisons with EOS models.

Models	Performance indicators		
	R	RMSE	MAE
PSO-MERF	**0.9997**	**0.000091**	**0.0000011**
SRK	0.9970	0.0051	0.00087
PR	0.9986	0.0033	0.00058
RK	**0.9990**	**0.0020**	**0.000099**
ZJ	0.9900	0.0063	0.0011

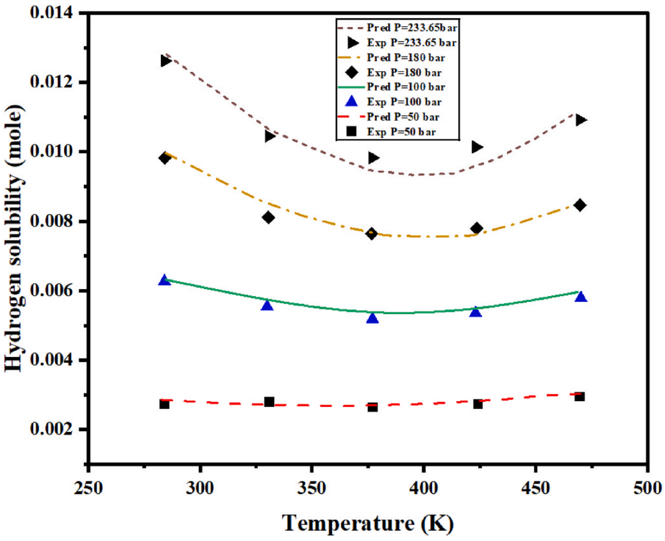


Fig. 9. Predicted hydrogen solubility by PSO-MERF with experimental hydrogen solubility at different pressures.

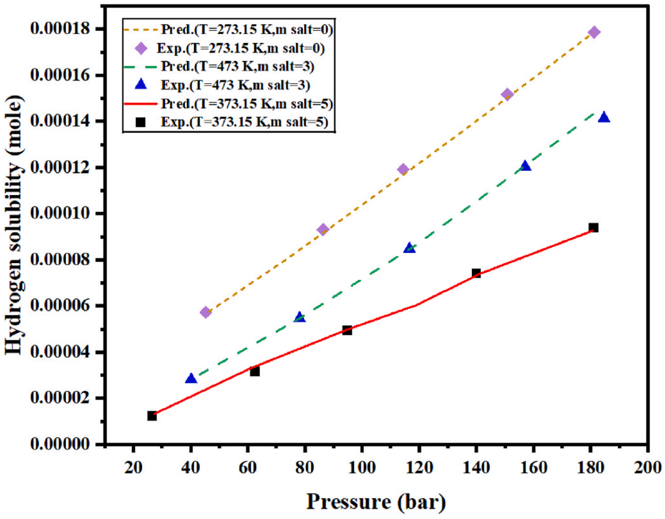


Fig. 10. Predicted hydrogen solubility by PSO-MERF with experimental hydrogen solubility at different temperatures and salt concentrations.

3.5. Shapley additive exPlanations (SHAP) analysis

SHAP analysis is a method used to explain the output of ML models [95,144,145]. It is based on the concept of Shapley values from cooperative game theory. It provides a way to fairly distribute the prediction among the inputs based on their marginal contribution to the model's prediction. The explanations provided by SHAP values are intuitive and can be visualized in various ways, such as force plots, waterfall plots, beeswarm plots, mean SHAP, etc. One of the main advantages of SHAP over other model interpretation methods is that it accounts for the interaction effects between features, as opposed to only considering the individual feature effects. This can provide a more accurate picture of how features contribute to model predictions, especially in complex models where features do not contribute independently to the output. In practice, calculating the exact SHAP values can be computationally expensive, especially for models with many features, because it requires evaluating the model for all possible subsets of features. Therefore, various approximation methods and algorithms have been developed to estimate SHAP values efficiently [144,146–148].

Consider a prediction model function f and an input feature set X with N features for a ML model. The SHAP value for the ith feature is given by Ref. [144].

$$\phi_i(f)=\sum_{S\subseteq X\setminus\{x_i\}}\frac{|S|!(N-|S|-1)!}{N!}[f(S\cup\{x_i\})-f(S)] \tag{21}$$

Where $\phi_i(f)$ is the SHAP value for the ith feature, S is a subset of features, and the sum goes over all subsets of features that do not include x_i, $|S|$ is the number of features in subset S, N is the total number of features, $f(S\cup\{x_i\})$ is the prediction of the model using the features in set S along with feature x_i, $f(S)$ is the prediction of the model using the features in set S without feature x_i, $\frac{|S|!(N-|S|-1)!}{N!}$ represents the weight for the contribution of feature x_i when considering the subset S, taking into account all the possible orderings of the features. Eq. (21) effectively distributes the prediction among the features, such that the sum of all SHAP values for a given prediction sums up to the difference between the prediction for the instance and the average prediction over the dataset (the base value). In this section, SHAP analysis based on the best method (PSO-MERF) was conducted for all data types, as presented in the following subsections.

Fig. 11 shows that hydrogen solubility in the aqueous solution generally increases as salinity decreases because there's less competition for hydrogen bonding with water molecules. This allows more hydrogen molecules to form these bonds, ultimately leading to an increase in hydrogen solubility in the aqueous solution, while for pressure is vice versa, in which an increase in pressure results in an increase in hydrogen solubility in the aqueous solution. This obeys Henry's law, which states that "the amount of a gas dissolved in a liquid is directly proportional to the partial pressure of the gas above the liquid". For temperature, as temperature increases, the hydrogen solubility in an aqueous solution decreases because, at low temperatures, water has stronger attractive forces that hold the hydrogen molecules in, but as temperature increases, these forces get weaker, allowing more hydrogen to escape and reducing its overall solubility. Moreover, Fig. 12 shows that salinity greatly impacted hydrogen solubility predicted by the PSO-MERF model with a SHAP value of +0.12, followed by pressure with +0.11. In contrast, temperature had a small contribution to model output compared to other parameters with a SHAP value of +0.02.

In general, SHAP is a valuable tool for explaining ML models, but it's essential to be aware of its limitations. By understanding these limitations, you can use SHAP effectively with other techniques to better understand your models'.

a) Conceptual limitations

❖ Feature independence: SHAP assumes features are independent or have minimal interaction effects. In reality, features often interact with each other, and these interactions can influence the model's predictions. SHAP might not fully capture these complex interactions.

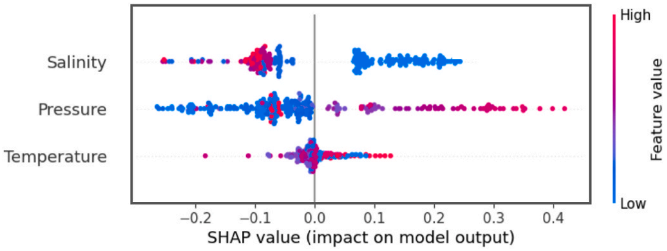


Fig. 11. Effects of input parameters on the model output (PSO-MERF).

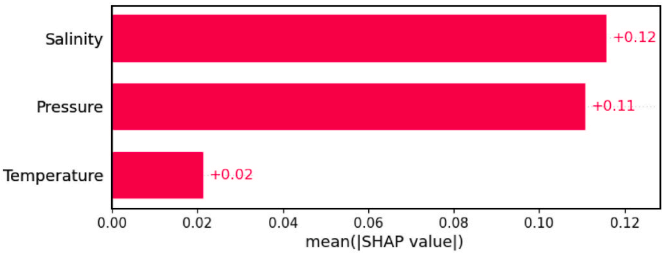


Fig. 12. Relative influence of input parameters on the model output (PSO-MERF).

❖ Causal vs. Correlational: SHAP explains feature contributions to the model's prediction, but it doesn't necessarily imply causality. Just because a feature has a high SHAP value doesn't necessarily mean it causes a change in the output variable.

b) Computational limitations

❖ Complexity with large datasets: Calculating SHAP values can be computationally expensive for large datasets with many features. This can make it impractical for real-time explanations.

❖ Approximation methods: For larger datasets, SHAP implementations often rely on approximation methods to calculate SHAP values efficiently. These approximations might not be as accurate as the exact calculation, introducing a potential source of error.

c) Interpretability limitations

❖ Meaning of SHAP value magnitude: The interpretation of the exact magnitude of a SHAP value depends on the specific implementation of SHAP and the model being used. It's essential to understand how SHAP values are calculated in a particular case for accurate interpretation.

❖ Global vs. local explanations: While SHAP can provide both local and global explanations, understanding complex models with many features through SHAP visualizations can be challenging.

4. Conclusions and recommendations

4.1. Conclusions

This paper explored the application of ML techniques in predicting hydrogen solubility in the context of underground hydrogen storage. Four ML models were employed: PSO-MERF, XGBoost, KNN, and RF. The results revealed that PSO-MERF outperformed the other three and EOS models' prediction accuracy. The findings encourage further exploration of ML techniques for optimizing UHS design and operation, ultimately contributing to the advancement of safe and sustainable hydrogen storage solutions with PSO-MERF recommended as an alternative for hydrogen solubility prediction in aqueous solutions for efficient UHS. The key findings are as follows.

a) PSO-MERF emerged as the most effective method for predicting hydrogen solubility, achieving superior performance compared to XGBoost, KNN, and RF with R of 0.9997 and 0.9982, RMSE of 0.0.00033 and 0.0015, and MAE of 0.00041 and 0.00091, during training and testing, respectively. This suggests that combining PSO's optimization capabilities and MERF's symbolic regression abilities can effectively capture the complex relationships between influencing factors and hydrogen solubility. XGBoost demonstrated promising results, indicating its potential for accurate prediction in this domain. Its ability to handle complex nonlinear relationships makes it a viable alternative, mainly when interpretability is less critical. KNN and RF, while achieving acceptable accuracy, were surpassed by PSO-MERF and XGBoost. These methods might be preferable for applications where interpretability is paramount, as their simpler models offer more precise insights into the relationships between variables. Moreover, PSO-MERF used less computational time (1.01s) followed by XGBoost 2.52 s, KNN 3.33 s, and RF 5. 25s.This confirms the robustness and fast convergence of the newly developed model in hydrogen solubility prediction utilizing less computational time <50% than other models.

b) From SHAP analysis, it has been shown that an increase in pressure results in an increase in hydrogen solubility in an aqueous solution, while a decrease in salinity results in an increase in hydrogen solubility, further, as temperature increases, the hydrogen solubility in an aqueous solution decrease. Moreover, salinity has a significant contribution to the model output compared to other parameters followed by pressure. In contrast, temperature has the least contribution compared to other parameters in model output.

c) Among the EOS models' used to predict hydrogen solubility in an aqueous solution, the RK model outperformed other EOS models with R of 0.9990, RMSE of 0.0020, and MAE of 0.000099. The order rank of performance for EOS models was RK > PR > SRK > ZJ.

4.2. *Recommendations for future study*

a) Data augmentation and feature engineering: Expanding the dataset with additional data points or incorporating new features derived from existing data could potentially improve the performance of all models. Techniques like data imputation or dimensionality reduction might also be beneficial.

b) Uncertainty quantification: Implement methods to quantify the uncertainty associated with the predictions. This would provide valuable insights into the model's confidence in its results and allow for a more comprehensive risk assessment in real-world applications.

c) Real-world validation: Validate the developed model using experimental data from underground storage facilities. This would enhance confidence in the model's generalizability and pave the way for practical implementation.

d) In future research, it is crucial to validate the PSO-MERF model to ensure its robustness. This can be achieved through external validation in which the model can use independent datasets not used during the model development process with the same inputs but missing output. This helps in evaluating the model's ability to generalize new data.

The application of ML techniques in UHS has profound implications, enhancing prediction accuracy, operational efficiency, and safety. ML models like PSO-MERF in predicting hydrogen solubility accurately can optimize storage conditions and efficiency, enable real-time monitoring, and predict maintenance needs, by reducing costs and improving economic viability. These advancements support the integration of hydrogen as a renewable energy source, minimize environmental impacts, and inform policy and regulatory standards. By continuously learning and adapting, ML enhances the reliability and sustainability of UHS, paving the way for safer and more efficient hydrogen storage solutions.

Data availability statement

The data sources are provided within the paper and attached as supplementary material.

CRediT authorship contribution statement

Grant Charles Mwakipunda: Writing – original draft, Methodology, Conceptualization. **Norga Alloyce Komba:** Writing – review & editing, Visualization, Methodology. **Allou Koffi Franck Kouassi:** Validation, Software, Methodology. **Edwin Twum Ayimadu:** Resources, Methodology, Investigation. **Melckzedeck Michael Mgimba:** Validation, Investigation, Data curation. **Mbega Ramadhani Ngata:** Writing – review & editing, Visualization, Investigation. **Long Yu:** Writing – review & editing, Validation, Supervision.

Declaration of competing interest

The authors declare that there is no conflict of interest.

Acknowledgements

The authors thank the Chinese Scholarship Council for their support (2022GXZ005733).

Nomenclature

RF	Random Forest
BBO	Biogeography-Based Optimization
XGBoost	Extreme Gradient Boosting
MLP	Multi-Layer Perceptron
DT	Decision Trees
GB	Gradient Boosting
FBP	Feedback Propagation
ICA	Imperialist Competitive Algorithm
ANN	Artificial neural networks
LightGBM	Light Gradient Boosting Machine
GEP	Gene Expression Programming
LM	Levenberg–Marquardt
GP	Genetic Programming
TLBO	Teaching-Learning-Based Optimization
LSSVM	Least Square Support Vector Machine
RBF	Radial Basis Function
GRNN	Generalized Regression Neural Networks
RAD	Relative Absolute Deviation
CA	Cultural Algorithm
RD	Relative Deviation
CFFNN	Cascade Feed-Forward Neural Networks
GMDH	Group Method of Data Handling
RAE	Relative Absolute Error
AdaBoost-SVR	Adaptive Boosting Support Vector Regression
PC-SAFT	Perturbed chain statistical associating fluid theory

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ijhydene.2024.09.054.

References

[1] Petrovic S. World energy overview. In: Petrovic S, editor. World energy handbook. Cham: Springer International Publishing; 2023. p. 3–10.

[2] Vo DTH, Trinh HH. Financing renewable energy transition toward sustainable development goals: policies and economic implications in the era of climate change. Green finance and sustainable development goals. World Scientific; 2024. p. 269–97.

[3] Yang R, Liu Z, Liu J. The methodology of decoupling fuel and thermal nitrogen oxides in multi-dimensional computational fluid dynamics combustion simulation of ammonia-hydrogen spark ignition engines. Int J Hydrogen Energy 2024;55:300–18.

[4] Masi M, Danieli P, Lazzaretto A. Effect of solidity and aspect ratio on the aerodynamic performance of axial-flow fans with 0.2 hub-to-tip ratio. J Turbomach 2023;145.

[5] Kohse-Höinghaus K. Combustion, chemistry, and carbon neutrality. Chem Rev 2023;123:5139–219.

[6] Azarpour A, Mohammadzadeh O, Rezaei N, Zendehboudi S. Current status and future prospects of renewable and sustainable energy in North America: progress and challenges. Energy Convers Manag 2022;269:115945.

[7] McCulloch N. Ending fossil fuel subsidies: the politics of saving the planet. Oxford, UK: Practical Action Publishing; 2023.

[8] Chen G, Jia G. A hybrid causal machine learning to reveal driving factors responsible coal market: case of the Chinese industry. J Clean Prod 2024;434:140249.

[9] Hussain A, Arif SM, Aslam M. Emerging renewable and sustainable energy technologies: state of the art. Renew Sustain Energy Rev 2017;71:12–28.

[10] Nastasi B, Markovska N, Puksec T, Duić N, Foley A. Ready solutions for today and tomorrow-Renewable and sustainable energy systems. Elsevier; 2024, 114341.

[11] Khalid MS, Mansour AS, Desouky SE-DM, Afify WSM, Ahmed SF, Elnaggar OM. Improving permeability prediction via machine learning in a heterogeneous carbonate reservoir: application to middle miocene nullipore, ras fanar field, gulf of suez, Egypt. Environ Earth Sci 2024;83:244.

[12] Hu X, Wang C, Elshkaki A. Material-energy Nexus: a systematic literature review. Renew Sustain Energy Rev 2024;192:114217.

[13] Liu J, Liu Z. In-cylinder thermochemical fuel reforming for high efficiency in ammonia spark-ignited engines through hydrogen generation from fuel-rich operations. Int J Hydrogen Energy 2024;54:837–48.

[14] Zhang L, Jia C, Bai F, Wang W, An S, Zhao K, et al. A comprehensive review of the promising clean energy carrier: hydrogen production, transportation, storage, and utilization (HPTSU) technologies. Fuel 2024;355:129455.

[15] Abdalla AM, Hossain S, Nisfindy OB, Azad AT, Dawood M, Azad AK. Hydrogen production, storage, transportation and key challenges with applications: a review. Energy Convers Manag 2018;165:602–27.

[16] Rasul M, Hazrat M, Sattar M, Jahirul M, Shearer M. The future of hydrogen: challenges on production, storage and applications. Energy Convers Manag 2022; 272:116326.

[17] Li D, Beyer C, Bauer S. A unified phase equilibrium model for hydrogen solubility and solution density. Int J Hydrogen Energy 2018;43:512–29.

[18] Hosseini SE, Wahid MA. Hydrogen production from renewable and sustainable energy resources: promising green energy carrier for clean development. Renew Sustain Energy Rev 2016;57:850–66.

[19] Muhammed NS, Gbadamosi AO, Epelle EI, Abdulrasheed AA, Haq B, Patil S, et al. Hydrogen production, transportation, utilization, and storage: recent advances towards sustainable energy. J Energy Storage 2023;73:109207.

[20] Ishaq H, Dincer I, Crawford C. A review on hydrogen production and utilization: challenges and opportunities. Int J Hydrogen Energy 2022;47:26238–64.

[21] Singh S, Jain S, Venkateswaran P, Tiwari AK, Nouni MR, Pandey JK, et al. Hydrogen: a sustainable fuel for future of the transport sector. Renew Sustain Energy Rev 2015;51:623–33.

[22] Amirthan T, Perera M. The role of storage systems in hydrogen economy: a review. J Nat Gas Sci Eng 2022;108:104843.

[23] Tarhan C, Çil MA. A study on hydrogen, the clean energy of the future: hydrogen storage methods. J Energy Storage 2021;40:102676.

[24] Bosu S, Rajamohan N. Recent advancements in hydrogen storage-Comparative review on methods, operating conditions and challenges. Int J Hydrogen Energy 2023.

[25] Xu Y, Deng Y, Liu W, Zhao X, Xu J, Yuan Z. Research progress of hydrogen energy and metal hydrogen storage materials. Sustain Energy Technol Assessments 2023; 55:102974.

[26] Fatah A, Al Ramadan M, Al-Yaseri A. Hydrogen impact on cement integrity during underground hydrogen storage: a minireview and future outlook. Energy Fuels 2024.

[27] Zivar D, Kumar S, Foroozesh J. Underground hydrogen storage: a comprehensive review. Int J Hydrogen Energy 2021;46:23436–62.

[28] Minougou JD, Gholami R, Andersen P. Underground hydrogen storage in caverns: challenges of impure salt structures. Earth Sci Rev 2023:104599.

[29] Tarkowski R. Underground hydrogen storage: characteristics and prospects. Renew Sustain Energy Rev 2019;105:86–94.

[30] Sekar LK, Kiran R, Okoroafor ER, Wood DA. Review of reservoir challenges associated with subsurface hydrogen storage and recovery in depleted oil and gas reservoirs. J Energy Storage 2023;72:108605.

[31] Raad SMJ, Leonenko Y, Hassanzadeh H. Hydrogen storage in saline aquifers: opportunities and challenges. Renew Sustain Energy Rev 2022;168:112846.

[32] Gbadamosi AO, Muhammed NS, Patil S, Al Shehri D, Haq B, Epelle EI, et al. Underground hydrogen storage: a critical assessment of fluid-fluid and fluid-rock interactions. J Energy Storage 2023;72:108473.

[33] Ershadnia R, Singh M, Mahmoodpour S, Meyal A, Moeini F, Hosseini SA, et al. Impact of geological and operational conditions on underground hydrogen storage. Int J Hydrogen Energy 2023;48:1450–71.

[34] Al-Yaseri A, Yekeen N, Al-Mukainah H, Sarmadivaleh M, Lebedev M. Snap-off effects and high hydrogen residual trapping: implications for underground hydrogen storage in sandstone aquifer. Energy Fuels 2024.

[35] Navaid HB, Emadi H, Watson M. A comprehensive literature review on the challenges associated with underground hydrogen storage. Int J Hydrogen Energy 2023;48:10603–35.

[36] Thiyagarajan SR, Emadi H, Hussain A, Patange P, Watson M. A comprehensive review of the mechanisms and efficiency of underground hydrogen storage. J Energy Storage 2022;51:104490.

[37] Gholami R. Hydrogen storage in geological porous media: solubility, mineral trapping, H2S generation and salt precipitation. J Energy Storage 2023;59: 106576.

[38] Aftab A, Hassanpouryouzband A, Xie Q, Machuca LL, Sarmadivaleh M. Toward a fundamental understanding of geological hydrogen storage. Ind Eng Chem Res 2022;61:3233–53.

[39] Diamantakis N, Peecock A, Shahrokhi O, Pitchaimuthu S, Andresen JM. A review of analogue case studies relevant to large-scale underground hydrogen storage. Energy Rep 2024;11:2374–400.

[40] Shokouhi M, Adibi M, Jalili AH, Hosseini-Jenab M, Mehdizadeh A. Solubility and diffusion of H2S and CO2 in the ionic liquid 1-(2-hydroxyethyl)-3-methylimidazolium tetrafluoroborate. J Chem Eng Data 2010;55:1663–8.

[41] Moysan J, Huron M, Paradowski H, Vidal J. Prediction of the solubility of hydrogen in hydrocarbon solvents through cubic equations of state. Chem Eng Sci 1983;38:1085–92.

[42] Ansari S, Safaei-Farouji M, Atashrouz S, Abedi A, Hemmati-Sarapardeh A, Mohaddespour A. Prediction of hydrogen solubility in aqueous solutions: comparison of equations of state and advanced machine learning-metaheuristic approaches. Int J Hydrogen Energy 2022;47:37724–41.

[43] Bender E, Klein U, Schmitt WP, Prausnitz JM. Thermodynamics of gas solubility: relation between equation-of-state and activity-coefficient models. Fluid Phase Equil 1984;15:241–55.

[44] Rahbari A, Brenkman J, Hens R, Ramdin M, Van Den Broeke LJ, Schoon R, et al. Solubility of water in hydrogen at high pressures: a molecular simulation study. J Chem Eng Data 2019;64:4103–15.

[45] Blas FJ, Vega LF. Prediction of binary and ternary diagrams using the statistical associating fluid theory (SAFT) equation of state. Ind Eng Chem Res 1998;37: 660–74.

[46] Alanazi A, Bawazeer S, Ali M, Keshavarz A, Hoteit H. Thermodynamic modeling of hydrogen–water systems with gas impurity at various conditions using cubic and PC-SAFT equations of state. Energy Convers Manag X 2022;15:100257.

[47] Sun L, Kontogeorgis GM, von Solms N, Liang X. Modeling of gas solubility using the electrolyte cubic plus association equation of state. Ind Eng Chem Res 2019; 58:17555–67.

[48] Kwaterski M, Herri J-M. Modelling of gas clathrate hydrate equilibria using the electrolyte non-random two-liquid (eNRTL) model. Fluid Phase Equil 2014;371: 22–40.

[49] Mohammadi M-R, Hadavimoghaddam F, Pourmahdi M, Atashrouz S, Munir MT, Hemmati-Sarapardeh A, et al. Modeling hydrogen solubility in hydrocarbons using extreme gradient boosting and equations of state. Sci Rep 2021;11:17911.

[50] Arianti P, Siti Mahmudah M. PERTUBED-CHAIN statistical associating fluid theory (PC-SAFT) equation of state untuk prediksi vapor-liquid equilibria sistem SOLVEN+ anti-solven CO2. 2010.

[51] Yuan H, Gosling C, Kokayeff P, Murad S. Prediction of hydrogen solubility in heavy hydrocarbons over a range of temperatures and pressures using molecular dynamics simulations. Fluid Phase Equil 2010;299:94–101.

[52] Anitescu C, Atroshchenko E, Alajlan N, Rabczuk T. Artificial neural network methods for the solution of second order boundary value problems. Comput Mater Continua (CMC) 2019;59:345–59.

[53] Liu Z, Liu J. Machine learning assisted analysis of an ammonia engine performance. J Energy Resour Technol 2022;144.

[54] Samaniego E, Anitescu C, Goswami S, Nguyen-Thanh VM, Guo H, Hamdia K, et al. An energy approach to the solution of partial differential equations in computational mechanics via machine learning: concepts, implementation and applications. Comput Methods Appl Mech Eng 2020;362:112790.

[55] Thanh HV, Zhang H, Dai Z, Zhang T, Tangparitkul S, Min B. Data-driven machine learning models for the prediction of hydrogen solubility in aqueous systems of varying salinity: implications for underground hydrogen storage. Int J Hydrogen Energy 2024;55:1422–33.

[56] Cao Y, Ayed H, Dahari M, Sene N, Bouallegue B. Using artificial neural network to optimize hydrogen solubility and evaluation of environmental condition effects. Int J Low Carbon Technol 2022;17:80–9.

[57] Lv Q, Zhou T, Zheng H, Amiri-Ramsheh B, Hadavimoghaddam F, Hemmati-Sarapardeh A, et al. Modeling hydrogen solubility in water: comparison of adaptive boosting support vector regression, gene expression programming, and cubic equations of state. Int J Hydrogen Energy 2024;57:637–50.

[58] Zhou Z, Nourani P, Karimi M, Kamrani E, Anqi AE. Relying on machine learning methods for predicting hydrogen solubility in different alcoholic solvents. Int J Hydrogen Energy 2022;47:5817–27.

[59] Jiang Y, Zhang G, Wang J, Vaferi B. Hydrogen solubility in aromatic/cyclic compounds: prediction by different machine learning techniques. Int J Hydrogen Energy 2021;46:23591–602.

[60] Tatar A, Esmaeili-Jaghdan Z, Shokrollahi A, Zeinijahromi A. Hydrogen solubility in n-alkanes: data mining and modelling with machine learning approach. Int J Hydrogen Energy 2022;47:35999–6021.

[61] Hadavimoghaddam F, Ansari S, Atashrouz S, Abedi A, Hemmati-Sarapardeh A, Mohaddespour A. Application of advanced correlative approaches to modeling hydrogen solubility in hydrocarbon fuels. Int J Hydrogen Energy 2023;48: 19564–79.

[62] Chabab S, Theveneau P, Coquelet C, Corvisier J, Paricaud P. Measurements and predictive models of high-pressure H2 solubility in brine (H2O+ NaCl) for underground hydrogen storage application. Int J Hydrogen Energy 2020;45: 32206–20.

[63] Torín-Ollarves GA, Trusler JM. Solubility of hydrogen in sodium chloride brine at high pressures. Fluid Phase Equil 2021;539:113025.

[64] Jáuregui-Haza U, Pardillo-Fontdevila E, Wilhelm A, Delmas H. Solubility of hidrogen and carbon monoxide in water and some organic solvents. Lat Am Appl Res 2004;34:71–4.

[65] Kling G, Maurer G. The solubility of hydrogen in water and in 2-aminoethanol at temperatures between 323 K and 423 K and pressures up to 16 MPa. J Chem Therm 1991;23:531–41.

[66] Ruetschi P, Amlie R. Solubility of hydrogen in potassium hydroxide and sulfuric acid. Salting-out and hydration. J Phys Chem 1966;70:718–23.

[67] Wiebe R, Gaddy V. The solubility of hydrogen in water at 0, 50, 75 and 100 from 25 to 1000 atmospheres. J Am Chem Soc 1934;56:76–9.

[68] Crozier TE, Yamamoto S. Solubility of hydrogen in water, sea water, and sodium chloride solutions. J Chem Eng Data 1974;19:242–4.

[69] Chabab S, Kerkache H, Bouchkira I, Poulain M, Baudouin O, Moine É, et al. Solubility of H2 in water and NaCl brine under subsurface storage conditions: measurements and thermodynamic modeling. Int J Hydrogen Energy 2024;50:648–58.

[70] Zhu Z, Cao Y, Zheng Z, Chen D. An accurate model for estimating H2 solubility in pure water and aqueous NaCl solutions. Energies 2022;15:5021.

[71] Alvarez J, Crovetto R, Fernández-Prini R. The dissolution of N2 and of H2 in water from room temperature to 640 K. Ber Bunsen Ges Phys Chem 1988;92:935–40.

[72] Pray HA, Schweickert C, Minnich BH. Solubility of hydrogen, oxygen, nitrogen, and helium in water at elevated temperatures. Ind Eng Chem 1952;44:1146–51.

[73] Sidi-Boumedine R, Horstmann S, Fischer K, Provost E, Fürst W, Gmehling J. Experimental determination of hydrogen sulfide solubility data in aqueous alkanolamine solutions. Fluid Phase Equil 2004;218:149–55.

[74] Chabab S, Théveneau P, Coquelet C, Corvisier J, Paricaud P. Measurements and predictive models of high-pressure H2 solubility in brine (H2O+NaCl) for underground hydrogen storage application. Int J Hydrogen Energy 2020;45:32206–20.

[75] Torín-Ollarves GA, Trusler JPM. Solubility of hydrogen in sodium chloride brine at high pressures. Fluid Phase Equil 2021;539:113025.

[76] Ruetschi P, Amlie RF. Solubility of hydrogen in potassium hydroxide and sulfuric acid. Salting-Out and hydration. J Phys Chem 1966;70:718–23.

[77] Wiebe R, Gaddy VL. The solubility of hydrogen in water at 0, 50, 75 and 100° from 25 to 1000 atmospheres. J Am Chem Soc 1934;56:76–9.

[78] Crozier TE, Yamamoto S. Solubility of hydrogen in water, sea water, and sodium chloride solutions. J Chem Eng Data 1974;19:242–4.

[79] Gordon LI, Cohen Y, Standley DR. The solubility of molecular hydrogen in seawater. Deep-Sea Res 1977;24:937–41.

[80] Morrison TJ, Billett F. 730. The salting-out of non-electrolytes. Part II. The effect of variation in non-electrolyte. J Chem Soc 1952:3819–22.

[81] Braun L. Über die Absorption von Stickstoff und von Wasserstoff in wässerigen Lösungen verschieden dissociierter Stoffe. Z Phys Chem 1900;33:721–39.

[82] Wiesenburg DA, Guinasso Jr NL. Equilibrium solubilities of methane, carbon monoxide, and hydrogen in water and sea water. J Chem Eng Data 1979;24:356–60.

[83] García-Escudero LA, Mayo-Iscar A. Robust clustering based on trimming. Wiley Interdisciplinary Reviews: Comput Stat 2024;16:e1658.

[84] Krishna NS, Kumar YP, Prakash KP, Reddy GP. Machine learning and statistical techniques for outlier detection in smart home energy consumption. 2024 IEEE open conference of electrical, electronic and information sciences (eStream). IEEE; 2024. p. 1–4.

[85] Mkono CN, Chuanbo S, Mulashani AK, Mwakipunda GC. Deep learning integrated approach for hydrocarbon source rock evaluation and geochemical indicators prediction in the Jurassic-Paleogene of the Mandawa basin, SE Tanzania. Energy 2023;284:129232.

[86] Majid A, Mwakipunda GC, Guo C. Solution gas/oil ratio prediction from pressure/volume/temperature data using machine learning algorithms. SPE J 2024;29:999–1014.

[87] Hasan N, Ahmed N, Ali SM. Improving sporadic demand forecasting using a modified k-nearest neighbor framework. Eng Appl Artif Intell 2024;129:107633.

[88] Steinbach M, Tan P-N. kNN: k-nearest neighbors. The top ten algorithms in data mining. 2009. p. 151–62.

[89] Lahmiri S. Integrating convolutional neural networks, kNN, and Bayesian optimization for efficient diagnosis of Alzheimer's disease in magnetic resonance images. Biomed Signal Process Control 2023;80:104375.

[90] Fix E, Hodges JL. Discriminatory analysis. Nonparametric discrimination: consistency properties. International Statistical Review/Revue Internationale de Statistique 1989;57:238–47.

[91] Sotiropoulou KF, Vavatsikos AP, Botsaris PN. A hybrid AHP-PROMETHEE II onshore wind farms multicriteria suitability analysis using kNN and SVM regression models in northeastern Greece. Renew Energy 2024;221:119795.

[92] Kohli S, Godwin GT, Urolagin S. Sales prediction using linear and KNN regression. Advances in machine learning and computational intelligence: proceedings of ICMLCI 2019. Springer; 2020. p. 321–9.

[93] Guo G, Wang H, Bell D, Bi Y, Greer K. KNN model-based approach in classification. On the move to meaningful internet systems 2003: CoopIS, DOA, and ODBASE: OTM confederated international conferences, CoopIS, DOA, and ODBASE 2003, catania, sicily, Italy, november 3-7, 2003 proceedings. Springer; 2003. p. 986–96.

[94] Chakravarthy SS, Bharanidharan N, Rajaguru H. Deep learning-based metaheuristic weighted K-nearest neighbor algorithm for the severity classification of breast cancer. IRBM 2023;44:100749.

[95] Nadege MN, Jiang S, Mwakipunda GC, Kouassi AKF, Harold PK, Roland KYH. Brittleness index prediction using modified random forest based on particle swarm optimization of Upper Ordovician Wufeng to Lower Silurian Longmaxi shale gas reservoir in the Weiyuan Shale Gas Field, Sichuan Basin, China. Geoenergy Science and Engineering 2024;233:212518.

[96] Liu B, Rostamian A, Kheirollahi M, Mirseyed SF, Mohammadian E, Golsanami N, et al. NMR log response prediction from conventional petrophysical logs with XGBoost-PSO framework. Geoenergy Science and Engineering 2023;224:211561.

[97] Alabdullah AA, Iqbal M, Zahid M, Khan K, Amin MN, Jalal FE. Prediction of rapid chloride penetration resistance of metakaolin based high strength concrete using light GBM and XGBoost models by incorporating SHAP analysis. Construct Build Mater 2022;345:128296.

[98] Hastie T, Tibshirani R, Friedman JH, Friedman JH. The elements of statistical learning: data mining, inference, and prediction. Springer; 2009.

[99] Siqueira RG, Moquedace CM, Fernandes-Filho EI, Schaefer CE, Francelino MR, Sacramento IF, et al. Modelling and prediction of major soil chemical properties with Random Forest: machine learning as tool to understand soil-environment relationships in Antarctica. Catena 2024;235:107677.

[100] Rigatti SJ. Random forest. J Insur Med 2017;47:31–9.

[101] Li H, Lin J, Lei X, Wei T. Compressive strength prediction of basalt fiber reinforced concrete via random forest algorithm. Mater Today Commun 2022;30:103117.

[102] Nafouanti MB, Li J, Nyakilla EE, Mwakipunda GC, Mulashani A. A novel hybrid random forest linear model approach for forecasting groundwater fluoride contamination. Environ Sci Pollut Control Ser 2023;30:50661–74.

[103] Biau G, Scornet E. A random forest guided tour. Test 2016;25:197–227.

[104] Sun Z, Wang G, Li P, Wang H, Zhang M, Liang X. An improved random forest based on the classification accuracy and correlation measurement of decision trees. Expert Syst Appl 2024;237:121549.

[105] Breiman L. Random forests. Mach Learn 2001;45:5–32.

[106] Liaw A, Wiener M. Classification and regression by randomForest. R News 2002;2:18–22.

[107] Rey-Blanco D, Zofío JL, González-Arias J. Improving hedonic housing price models by integrating optimal accessibility indices into regression and random forest analyses. Expert Syst Appl 2024;235:121059.

[108] Leo GL, Jayabal R, Srinivasan D, Das MC, Ganesh M, Gavaskar T. Predicting the performance and emissions of an HCCI-DI engine powered by waste cooking oil biodiesel with Al2O3 and FeCl3 nano additives and gasoline injection–A random forest machine learning approach. Fuel 2024;357:129914.

[109] Genuer R, Poggi J-M, Genuer R, Poggi J-M. Random forests. Springer; 2020.

[110] Nguyen J-M, Jézéquel P, Gillois P, Silva L, Ben Azzouz F, Lambert-Lacroix S, et al. Random forest of perfect trees: concept, performance, applications and perspectives. Bioinformatics 2021;37:2165–74.

[111] Antoniadis A, Lambert-Lacroix S, Poggi J-M. Random forests for global sensitivity analysis: a selective review. Reliab Eng Syst Saf 2021;206:107312.

[112] He B, Armaghani DJ, Lai SH. Assessment of tunnel blasting-induced overbreak: a novel metaheuristic-based random forest approach. Tunn Undergr Space Technol 2023;133:104979.

[113] Chowdhury MS. Comparison of accuracy and reliability of random forest, support vector machine, artificial neural network and maximum likelihood method in land use/cover classification of urban setting. Environmental Challenges 2024;14:100800.

[114] Kennedy J, Eberhart RC. A discrete binary version of the particle swarm algorithm. 1997 IEEE International conference on systems, man, and cybernetics Computational cybernetics and simulation: ieee. 1997. p. 4104–8.

[115] Marini F, Walczak B. Particle swarm optimization (PSO). A tutorial. Chemometr Intell Lab Syst 2015;149:153–65.

[116] Houssein EH, Gad AG, Hussain K, Suganthan PN. Major advances in particle swarm optimization: theory, analysis, and application. Swarm Evol Comput 2021;63:100868.

[117] Suriyan K, Nagarajan R. Particle swarm optimization in biomedical technologies: innovations, challenges, and opportunities. Emerging Technologies for Health Literacy and Medical Practice; 2024. p. 220–38.

[118] Tudorică B-G, Bucur C, Panait M, Oprea S-V, Bâra A. Energetic Equilibrium: optimizing renewable and non-renewable energy sources via particle swarm optimization. Util Pol 2024;87:101722.

[119] Wang D, Tan D, Liu L. Particle swarm optimization algorithm: an overview. Soft Comput 2018;22:387–408.

[120] Gad AG. Particle swarm optimization algorithm and its applications: a systematic review. Arch Comput Methods Eng 2022;29:2531–61.

[121] Hemalatha N, Venkatesan S, Kannan R, Kannan S, Bhuvanesh A, Kamaraja A. Sensorless speed and position control of permanent magnet BLDC motor using particle swarm optimization and ANFIS. Measurement: Sensors 2024;31:100960.

[122] Wu W, Chen K, Tsotsas E. Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model. Powder Technol 2024;434:119365.

[123] Nayak J, Swapnarekha H, Naik B, Dhiman G, Vimal S. 25 years of particle swarm optimization: flourishing voyage of two decades. Arch Comput Methods Eng 2023;30:1663–725.

[124] Rutten T, Cqm JWB, van den Heuvel E, Cqm IMP. Mixed-effects random forest model for quantifying relations in clustered data. 2021.

[125] Hajjem A, Bellavance F, Larocque D. Mixed effects regression trees for clustered data. Stat Probab Lett 2011;81:451–9.

[126] Hajjem A, Larocque D, Bellavance F. Generalized mixed effects regression trees. Stat Probab Lett 2017;126:114–8.

[127] Katreddi S, Thiruvengadam A, Thompson GJ, Schmid NA. Mixed effects random forest model for maintenance cost estimation in heavy-duty vehicles using diesel and alternative fuels. IEEE Access 2023.

[128] Yang S-I, Brandeis TJ, Helmer EH, Oatham MP, Heartsill-Scalley T, Marcano-Vega H. Characterizing height-diameter relationships for Caribbean trees using mixed-effects random forest algorithm. For Ecol Manag 2022;524:120507.

[129] Krennmair P, Schmid T. Flexible domain prediction using mixed effects random forests. J Roy Stat Soc C Appl Stat 2022;71:1865–94.

[130] Mayapada R, Susetyo B, Sartono B. A comparison between random forest and mixed effects random forest to predict students' math performance in Indonesia. Int J Sci Basic Appl Res 2021;57:1–8.

[131] Hajjem A, Bellavance F, Larocque D. Mixed-effects random forest for clustered data. J Stat Comput Simulat 2014;84:1313–28.

[132] Yang L, Shami A. On hyperparameter optimization of machine learning algorithms: theory and practice. Neurocomputing 2020;415:295–316.

[133] Bischl B, Binder M, Lang M, Pielok T, Richter J, Coors S, et al. Hyperparameter optimization: foundations, algorithms, best practices, and open challenges. Wiley Interdisciplinary Reviews: Data Min Knowl Discov 2023;13:e1484.

[134] Pannakkong W, Thiwa-Anont K, Singthong K, Parthanadee P, Buddhakulsomsiri J. Hyperparameter tuning of machine learning algorithms using response surface methodology: a case study of ANN, SVM, and DBN. Math Probl Eng 2022;2022:1–17.

[135] Bartz E, Bartz-Beielstein T, Zaefferer M, Mersmann O. Hyperparameter tuning for machine and deep learning with R: a practical guide. Springer Nature; 2023.

[136] Bartz-Beielstein T. Hyperparameter tuning. Online machine learning: a practical guide with examples in Python. Springer; 2024. p. 125–40.

[137] Mgimba MM, Jiang S, Nyakilla EE, Mwakipunda GC. Application of GMDH to predict pore pressure from well logs data: a case study from southeast sichuan basin, China. Nat Resour Res 2023;32:1711–31.

[138] Hintze JL, Nelson RD. Violin plots: a box plot-density trace synergism. Am Statistician 1998;52:181–4.

[139] Li Q, Li M, Safaei MR. Development of various machine learning and deep learning models to predict glycerol biorefining processes. Int J Hydrogen Energy 2024;52:669–85.

[140] Tasneem S, Ageeli AA, Alamier WM, Hasan N, Goodarzi M. Development of machine learning-based models for describing processes in a continuous solar-driven biomass gasifier. Int J Hydrogen Energy 2024;52:718–38.

[141] Pedersen KS, Christensen PL, Shaikh JA, Christensen PL. Phase behavior of petroleum reservoir fluids. CRC press; 2006.

[142] Zudkevitch D, Joffe J. Correlation and prediction of vapor-liquid equilibria with the redlich-kwong equation of state. AIChE J 1970;16:112–9.

[143] Ronze D, Fongarland P, Pitault I, Forissier M. Hydrogen solubility in straight run gasoil. Chem Eng Sci 2002;57:547–53.

[144] Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, et al. From local explanations to global understanding with explainable AI for trees. Nat Mach Intell 2020;2:56–67.

[145] Shapley LS. A value for n-person games. 1953.

[146] Lundberg SM, Lee S-I. A unified approach to interpreting model predictions. Adv Neural Inf Process Syst 2017;30.

[147] Janzing D, Minorics L, Blöbaum P. Feature relevance quantification in explainable AI: a causal problem. International Conference on artificial intelligence and statistics. PMLR; 2020. p. 2907–16.

[148] Zhang J, Ma X, Zhang J, Sun D, Zhou X, Mi C, et al. Insights into geospatial heterogeneity of landslide susceptibility based on the SHAP-XGBoost model. J Environ Manag 2023;332:117357.