Niu *et al.*
Journal of Engineering and Applied Science (2024) 71:137
https://doi.org/10.1186/s44147-024-00472-0

Journal of Engineering and Applied Science

RESEARCH Open Access

Unconfined compressive strength prediction of rock materials based on machine learning

Lihong Niu[1], Qiang Cui[2,3*], Jiangyun Luo[3], Hongbing Huang[4] and Jing Zhang[5]

*Correspondence:
cuiqiang227@126.com

[1] Xiasiwan Oil Production Plant of Yanchang Oilfield Co., Ltd, Yan'an 716000, Shaanxi, China
[2] School of Petroleum Engineering and Environmental Engineering, Yan'an University, Yan'an 716000, Shaanxi, China
[3] Engineering Research Center of Efficient Exploitation of Oil and Gas Resources and Protection Ecological Environment, Universities of Shaanxi Province, Yan'an 716000, Shaanxi, China
[4] Development Department of Yanchang Oilfield Co., Ltd, Yan'an 716000, Shaanxi, China
[5] Zhidan Oil Production Plant of Yanchang Oilfield Co., Ltd, Yan'an 716000, Shaanxi, China

Abstract

It is costly, time-consuming, and difficult to measure unconfined compressive strength (UCS) using typical laboratory procedures, particularly when dealing with weak, extremely porous, and fractured rock. By efficiently choosing the variables from a subset of the dataset that includes the Schmidt hammer rebound number (SRn), bulk density (BD), bulk tensile strength (BTS), dry density (DD) test, p-wave velocity test (Vp), and point load index test (Is(50)), this study seeks to establish predictive models for the UCS of rocks. A prediction model for UCS was prepared using K-nearest neighbor (KNN). KNN was preferred over machine learning algorithms because it is simple, versatile, and interpretable. It is particularly useful when it has limited training time, faces non-parametric data with changing distributions, or requires straightforward explanations for predictions. In order to improve KNN's prediction performance in this research, two optimization procedures (namely, Alibaba and the Forty Thieves (AFT) and Improved Manta-Ray Foraging Optimizer (IMRFO)) were used. Through comparison of KNN single modal performance with that of optimized versions, it is concluded that the KNIM (KNN model optimized with IMRFO) is an excellent possible applicant for the forecast of the UCS of rocks. This study's results showed that the KNIM model is more suitable than the KNN single model and its counterpart KNAF in terms of accuracy as its correlation of determination (R^2) values were 1.1% and 2% higher than KNN and KNAF and its root mean squared error (RMSE) values were 37.9% and 43.7% lower than KNN and KNAF. The improvement in R^2 and RMSE values for the KNIM model compared to KNN and KNAF is highly significant for the reliability and accuracy of the predictive model. R^2, measuring the proportion of variance predictable in the dependent variable (UCS of rocks) from the independent variables (model predictions), signifies a better fit to observed data. The elevated R^2 values for KNIM indicate a stronger correlation with actual UCS values, enhancing the model's accuracy in representing underlying patterns. Additionally, the reduction in RMSE values for KNIM implies that its predictions are, on average, closer to actual UCS values, contributing to a more accurate and reliable estimation of rock strength.

Keywords: Unconfined compressive strength of rock sample, Hybrid machine learning models, K-nearest neighbor, Alibaba and the Forty Thieves, Improved Manta-Ray Foraging Optimizer

Springer Open

© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

Introduction

One of the most important metrics for determining a rock's ability to support weight is its unconfined compressive strength (UCS). Inaccurate UCS calculations can be hazardous as they diminish the final bearing capacity. Rock strength is typically determined through unconfined compression tests in laboratories, following established procedures like those charted by the International Society for Rock Mechanics (ISRM). However, various challenges exist in directly measuring UCS in the lab, notably in obtaining suitable rock core specimens, particularly for rocks that are severely fractured and have a lot of lamination and foliation [1, 2]. Determining UCS directly in the early design stages is costly and time-consuming [3]. Nevertheless, alternative methods, such as regression models and machine learning techniques, offer viable options for predicting rock strength.

Numerous researchers have attempted to establish standardized methods for assessing UCS. Various techniques for predicting UCS fall into categories such as simple regression, where UCS is correlated with parameters derived from basic index tests for rocks. These tests encompass the tests for the Schmidt hammer, ultrasonic velocity (Vp), point-load index, Brazilian tensile strength, and slake durability index [4–6]. Multiple regression analysis has also been successfully employed to predict rock strength. However, it has been noted in some reports that these relationships may not consistently yield highly reliable UCS values. The clarification is that the correlation is not directly with UCS but involves parameters derived from these tests, as exemplified by not correlating UCS with the slake durability test but rather with the slake durability index [7]. Commonly, it is recommended to use these equations for specific rock types. Furthermore, these analytical prediction techniques cannot adapt to changes in data. Consequently, the equations require updating if new data is introduced [8–10].

Geotechnical researchers now favor machine learning (ML), containing support vector machines, artificial neural networks, decision trees, and neuro-fuzzy systems, due to their effectiveness in addressing complex engineering problems [11–15]. ML reduces lab testing costs for UCS determination and has broad applications in science and engineering challenges [16–18].

Previous studies have applied machine learning (ML) techniques to predict UCS [19]. Meulenkamp and Grima used a backpropagation artificial neural network (ANN) on various rock samples and found it outperformed traditional statistical methods [20]. Sonmez et al. applied a fuzzy inference system (FIS) to agglomerate samples of Ankara, yielding highly reliable UCS predictions [21]. Gokceoglu and Zorlu used regression and fuzzy models on problematic rocks, with the fuzzy model performing desirable for UCS estimation [22]. Dehghan et al. compared feed-forward regression and neural network models, concluding that ANN is a more robust model for the estimation of UCS [23]. Mishra and Basu found FIS and multiple regression approaches more efficient than simple regression for UCS prediction [24]. Cevik et al. reported the efficiency of ANN for sedimentary rock samples [25]. Yesiloglu-Gultekin et al. favored the adaptive neuro-fuzzy inference system (ANFIS) over multiple regression and ANN [26]. Skentou et al. [27] explored the prediction of UCS in granite by employing three ANN-based models. The study used three non-destructive test indicators on a collected dataset consisting of 274 entries: pulse velocity, Schmidt-hammer rebound number, and effective porosity.

Three ANN models were trained and validated: ANN-LM (built using the Levenberg–Marquardt method), ANN-PSO (ANN and particle swarm optimization), and ANN-ICA (ANN and imperialist competitive algorithm). The experimental findings revealed that the ANN-LM model exhibited the highest accuracy, achieving superior predictive performance in the validation phase with an R value of 0.9607 and RMSE of 14.8272. Comparative analysis showed that the developed ANN-LM outperformed existing models found in the literature. Additionally, the study developed a graphical user interface (GUI) for estimating UCS in granite using the ANN-LM model, enhancing practical usability. Le et al. [28] employed artificial neural networks to predict the UCS of rocks using a comprehensive database of 367 literature datasets. The study focused on input parameters such as Schmidt hammer number (Rn), Vp, and effective porosity (ne). Notably, the developed ANN effectively consolidated different Schmidt hammer numbers, exhibiting a correlation with L-type Schmidt hammer numbers within ±20% deviation from experimental data for 97.27% of specimens. Among the soft computing models considered (ANN-LM, ANN-PSO, and ANN-ICA), the highest accuracy was achieved with the ANN-ICA model. This model demonstrated strong predictive performance for UCS across various rock types and formation methods, showcasing less than ±20% deviation from experimental data for 86.36% of cases. Additionally, the study provided a user-friendly graphical interface, incorporating the closed-form equation of the ANN-ICA model, as supplementary material. Koopialipoor et al. [17] developed a novel system utilizing machine learning models such as k-nearest neighbors (KNN), multi-layer perceptron (MLP), random forest (RF), and tree. The optimal model, a stacking-tree-RF-KNN-MLP structure, integrated diverse characteristics from these models to enhance the accuracy of predicting Young's modulus. The refinement process involved optimizing influential parameters within each basic model, resulting in the development of the final model. Rock deformations were predicted using four index tests: porosity, Schmidt hammer, point load strength, and p-wave velocity. The stack-tree-RF-KNN-MLP model achieved the highest prediction accuracy ($R^2 = 0.8197$, MSE = 227.371, RMSE = 15.079, and MAE = 12.123).

Moreover, Table 1 shows the summary of some published articles.

This study addresses the imperative need for robust prediction models capable of effectively forecasting the UCS of rocks, considering the intricate relationships among various input variables. The choice of utilizing the K-nearest neighbor (KNN) algorithm stems from its intrinsic value as a versatile tool for UCS prediction, offering simplicity and adaptability. The necessity for such models arises from the complex, non-linear relationships exhibited by rocks' UCS concerning diverse geological parameters. KNN, with its instance-based approach, proves advantageous, requiring no extensive training and enabling real-time adaptability to dynamic geological conditions. Additionally, KNN provides a unique insight into feature importance, facilitating the identification of key parameters influencing rock strength. Acknowledging the complex spatial variations in rock properties, KNN's capability to capture both local and global patterns in the data aligns with the study's novel approach to modeling UCS. It serves as a crucial tool for understanding and predicting the spatial nuances of rock strength variations. However, the choice of KNN is underscored by the necessity to consider dataset characteristics and problem complexity, ensuring its suitability as a benchmark for UCS prediction in rock

Table 1 Summary of published articles

Article	Num. of dataset	Variables	Model	Evaluator	
				R^2	RMSE
Narendra et al. [29]	186	*Curingperiod*, *claywater* − *cementratio*, *cementcontent*, *liquidlimit*, *liquidityindex*, *watercontent*, *pH*, *Na*+	GP	0.9881	135
Ceryan et al. [30]	56	*Clt*, *Cly*, *Fld*, *Qrz*, *Qq*, *Bi*, *n*, *ne*, *Id*, *Vp*, *andVm*	REG	0.8837	1.108
Majdi and Rezaei [31]	93	*Rocktype*, *Schmidthardness*, *Density*, *andPorosity*	ANN	0.9725	1.113
Rezaei et al. [10]	93	*Rocktype*, *Schmidthardness*, *Density*, *andPorosity*	Mamdani fuzzy	0.9437	3.2
Mohamad et al. [32]	160	*Rocktype*, *Weatheringgrade*, *BD*, *BTS*, *Is*(50), *andVp*	ANN-PSO	0.982	0.077

mechanics. The novelty of this study lies in the enhancement of the presented models through the integration of two metaheuristic algorithms: Alibaba and the Forty Thieves (AFT) and Improved Manta-Ray Foraging Optimizer (IMRFO). This innovative approach aims to elevate the predictive performance of the models further, addressing the exigency for improved accuracy and reliability in predicting UCS in highly fractured, porous, and weak rock formations. The objective of the optimization algorithm selection in this study was to elevate the performance of the KNN model in its predictive capacity for the UCS of rocks. In pursuit of this goal, two algorithms, AFT and IMRFO, were deliberately chosen due to their demonstrated efficiency in addressing complex problems.

AFT, known for its exploratory nature, was specifically selected to contribute to the refinement of the KNN model parameters. The algorithm's inherent ability to explore the solution space was deemed advantageous in systematically adjusting the model's configuration to better align with the intricacies of the dataset, thereby enhancing its predictive accuracy for UCS. On the other hand, IMRFO was chosen for its advanced exploration capabilities, with the primary aim of fine-tuning parameters and further improving the overall predictive performance of the KNN model. IMRFO's capacity for comprehensive exploration, considering multiple objectives, was considered a valuable asset in the quest for optimal parameter settings that could significantly elevate the model's efficacy in capturing the underlying patterns governing rock strength.

By strategically combining the strengths of AFT and IMRFO, the study aimed to not only refine the KNN model but also harness the relation between these optimization algorithms to achieve a more robust and accurate predictive framework for UCS in rock mechanics. The thoughtful integration of these algorithms aimed to navigate the complexities inherent in rock strength prediction and contribute to the advancement of reliable modeling techniques in geotechnical applications.

Five statistical indices, which included R^2, RMSE, MSE, RSR, and FB, were utilized to assess the precision of the models. Finally, the optimal model is introduced to be used in geotechnical applications.

Laboratory tests and methodology

Rock sample data

In geotechnical analysis, critical rock properties encompass physical attributes and strength parameters. These include bulk density (BD), pivotal for UCS measurements by aiding in dry weight calculation; bulk tensile strength (BTS), indicating resistance to tensile forces; dry density (DD) for assessing soil compaction quality; p-wave velocity test (Vp) for measuring seismic wave speed; Schmidt hammer rebound number (SRn) for surface hardness evaluations; and point load index test (Is(50)), especially useful in core rock sample testing. Each of these variables has been widely recognized in the study as a significant contributor to the mechanical properties of rocks.

- BD: represents the mass per unit volume of the rock, offering insights into its overall density and compactness.
- BTS: reflects the rock's ability to withstand tensile stresses, providing crucial information on its structural integrity.

- DD test: measures the density of the rock without considering water content, contributing to a more accurate assessment of its composition.
- Vp: indicates the speed at which compressional waves travel through the rock, offering insights into its elastic properties.
- SRn: provides an estimate of the rock's surface hardness, which correlates with its overall strength.
- Is(50): measures the rock's strength under point loading conditions, offering valuable data on its resistance to applied forces.

In this study, 106 datasets are used from the published paper [32–34], of which 70% (74 samples) are related to training and 30% (32 samples) are related to the testing section. The rock specimens were also subjected to measurements of their bulk densities. The test procedure recommended by the International Society for Rock Mechanics (ISRM) was implemented. The weathering grade categorization employed adheres to the methodology proposed by the ISRM. As indicated in this table, the samples vary in weathering degree, ranging from slightly weathered to extensively weathered. The BD values of the rock samples vary between 2089 and 3534 ($kg/m^{3)}$. The rock samples in this investigation exhibit a range of indirect tensile strengths, or BTSs, from around 0.7 to4.2(MPa). The UCS measurements, however, vary between 5.5 and61.1(MPa). As anticipated, the UCS falls proportionally with the increase in weathering intensity. As an example, the average UCS for grade IV is10.6(MPa), whereas for gradeII, this number rises to60(MPa). The results for Vp demonstrate that Vp values are elevated for shales that are denser and have lower porosity. However, the Vp values vary between 1247(m/s) for rock samples that are extensively weathered and 2910(m/s) for barely weathered rock samples. The results indicate that the maximum compressive strength of rock samples is 4.1(MPa) for gradeII. However, for extensively weathered shale rock samples (grade IV), this value reduces to0.1(MPa). Moreover, 70% of the dataset belonged to the training set and 30% to the test set. To explore the influence of these input variables on UCS outcomes, experimental records are provided in Table 2. In addition, the dataset is mentioned in Appendix 1.

Scatter plots in Fig. 1 visually represent data points on a Cartesian plane, where the horizontal axes represent the input variables, and the vertical axes correspond to the output variable (UCS). The distribution of data points, their concentration, and any discernible patterns or trends within the plots can offer valuable insights into the relationship between these variables. Evidently, among the seven variables under consideration, there is a uniform dispersion of data points for three variables: sample

Table 2 Statistical properties of input and UCS

Features	Dataset components							
	Sample set	BD (kg/m^3)	BTS (MPa)	DD (kg/m^3)	Vp (m/s)	SRn (MPa)	Is(50) (MPa)	UCS (MPa)
Min	1	0	0	0	1247	0	0.1	5.5
Max	106	3.535	4.2	3011	6440	45.4	6.07	108.68
Mean	53.5	0.970	0.827	1669.7	4092.1	23.754	2.508	47.930
St.Dev	30.744	1.276	1.227	1308.1	1722.2	18.847	1.568	26.849

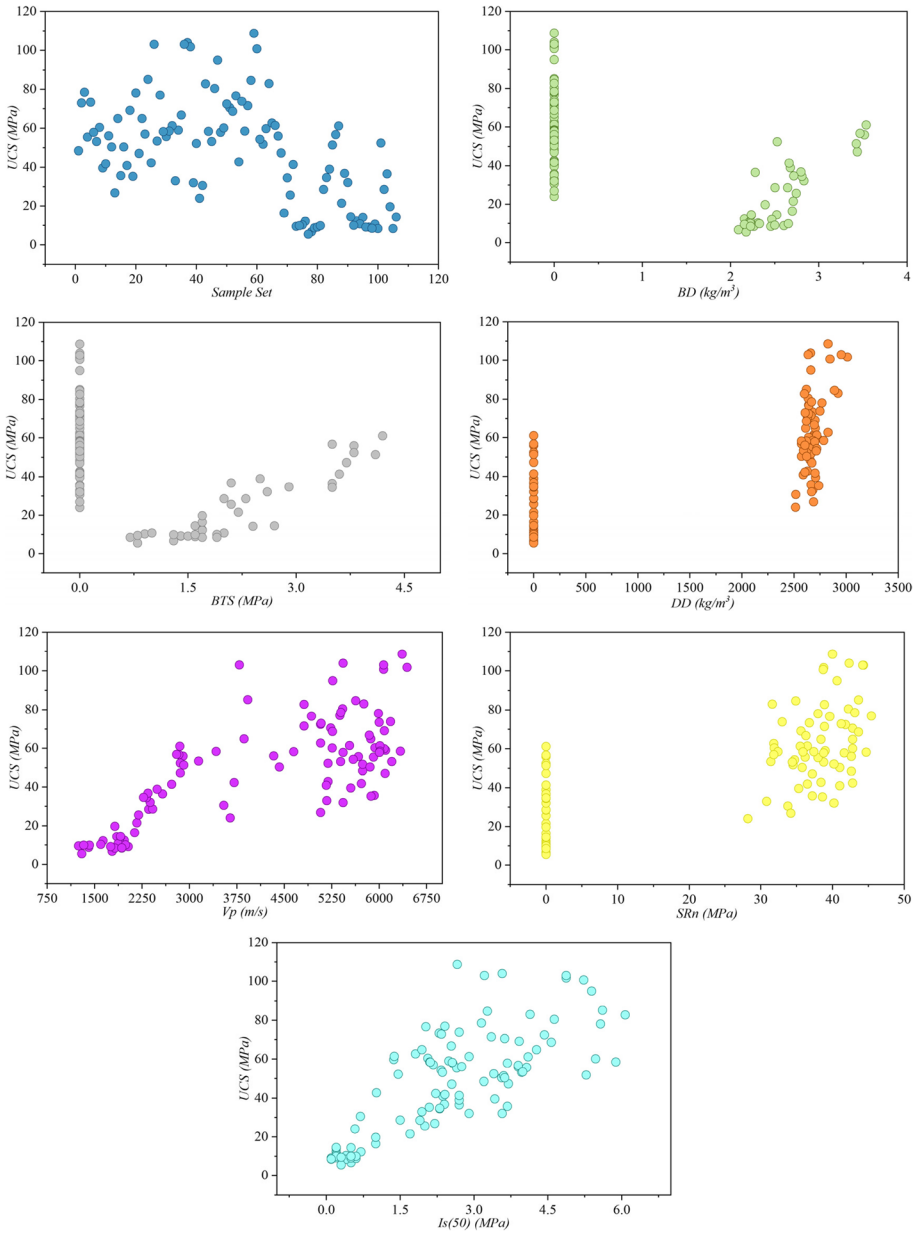

Fig. 1 The scatter plot between input and output

number, Vp, and Is(50). In contrast, the remaining four variables exhibit a notable concentration of approximately half of the data points at zero for each variable.

Figure 2 indicates that the correlation matrix provides insights into the linear relationships between different variables in the dataset. Notably, strong positive correlations, such as the 0.9250 correlation coefficient between BD and BTS, indicate that as one variable increases, the other tends to increase as well. Conversely, strong negative correlations, like the − 0.9792 correlation between BD and DD, suggest that as one variable increases, the other tends to decrease. The matrix also reveals interesting patterns, such as the strong negative correlation (− 0.9669) between SRn and BD, indicating an inverse relationship between these two variables. Similarly, the strong

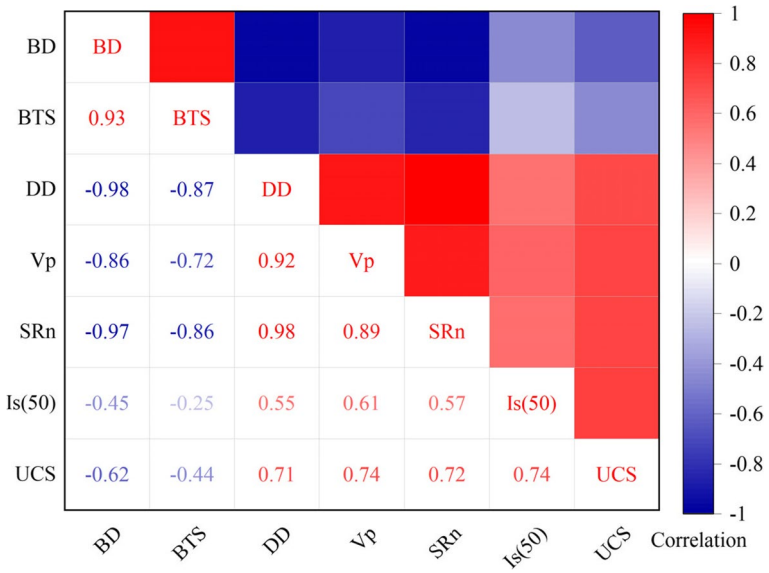

Fig. 2 Correlation between the input and output variables

negative correlation (0.9830) between SRn and DD suggests a robust inverse connection. These findings can be valuable for understanding the interplay between geological or engineering parameters represented by the variables. Additionally, moderate negative correlations, like the − 0.6226 correlation coefficient between UCS and BD, provide further insights into the relationships within the dataset.

K-nearest neighbor (KNN)

The KNN technique is known for its simplicity, effectiveness, and ease of implementation [35]. Like artificial neural networks (ANN) and random forests (RF), KNN can be applied for classification and regression tasks. Several advantages are associated with the utilization of this method:

1. It is straightforward and easily understandable, making it accessible for practical implementation.
2. When applied to regression and classification, it can train non-linear decision boundaries and offers flexibility in defining them by adjusting the K value. These features enhance its versatility.
3. Unlike some other architectures, KNN does not involve a specific training step.
4. The method involves only one hyperparameter, denoted as K, simplifying the adjustment of other hyperparameters.

The fundamental principle underlying KNN is identifying a set of K samples, often determined using distance functions, that exhibit proximity to unknown samples within the calibration data. It is achieved by identifying sample groups that share similarities. Subsequently, KNN ascertains the category of unfamiliar samples by computing the mean of response variables and then contrasting these outcomes

with those of the K-selected samples [36]. Consequently, the choice of the value for K plays a pivotal role in the effectiveness of the KNN algorithm [37]. KNN operates on the fundamental principle of proximity, predicting a target variable by considering the majority class or average value of its k-nearest neighbors in a multi-dimensional feature space. In the context of this study, KNN leverages the similarity between rock samples in this feature space to estimate their UCS.

The methodology of KNN involves several key steps. Firstly, distance calculation is performed, computing the distance between the target rock sample and all other samples in the dataset using a specified distance metric, such as the commonly used Euclidean distance. Subsequently, neighbor selection identifies the k-nearest neighbors of the target sample based on the calculated distances. For classification tasks, the algorithm employs majority voting among the neighbors to assign the class to the target sample. In regression tasks, it calculates the weighted average of the target variable based on the distances to these neighbors.

Two crucial considerations in the KNN methodology are hyperparameter selection and feature scaling. The choice of the hyperparameter "*k*," representing the number of neighbors, significantly influences the model's performance. A smaller "*k*" provides a more flexible model, whereas a larger "*k*" results in a smoother decision boundary. Additionally, proper scaling of features is essential as KNN is sensitive to the magnitude of input variables.

In the context of this study, KNN is specifically applied to predict the UCS of rocks. The algorithm utilizes a carefully selected subset of variables for this purpose. Its suitability for the complex task of modeling UCS in highly fractured, porous, and weak rock formations is attributed to its simplicity, interpretability, and ability to capture non-linear relationships. This comprehensive introduction to KNN sets the stage for its role in predicting UCS in rocks, emphasizing its operational principles, methodology, and key considerations in the study's context.

In the context of regression tasks, 3 distance functions, which assess the distances among neighboring points and are represented by Eqs. (1) to (3), are employed for this purpose:

$$F(e) = \sqrt{\sum_{i=0}^{f} (x_i - y_i)^2} \tag{1}$$

$$F(ma) = \sum_{i=0}^{f} \left| x_i - y_i \right| \tag{2}$$

$$F(mi) = \left(\sum_{i=0}^{f} (\left| x_i - y_i \right|)^q \right)^{\frac{1}{q}} \tag{3}$$

where $F(e)$ represents the Euclidean distance function, $F(ma)$ corresponds to the Manhattan distance function, and $F(mi)$ represents the Minkowski distance function. Here, x_i and y_i refer to the ith dimension of the data points x and y, and q represents the order parameter governing the distance calculation between these points.

Optimization algorithms

Alibaba and the Forty Thieves (AFT)

The framework of AFT encompasses three discernible states, each of which can be analyzed and delineated as follows [38, 39]:

First state The modeling of the pursuit of Ali Baba by the thieves, utilizing data acquired from an information source, can be effectively illustrated by employing Eq. (4). This equation serves as a means to represent the positions held by the individual thieves during the pursuit.

$$x_i^{t+1} = gbest^t + \left[Td^t\left(best_i^t - y_i^t\right)r_1 + Td^t\left(y_i^t - m_{a(i)}^t\right)r_2\right] sgn(rand - 0.5), p \geq 0.5, q > P_{p^t} \tag{4}$$

Where x_i^{t+1} indicates the location of the ith theft during the subsequent time step $(t+1)$. $m_{a(i)}^t$ shows the degree of cunning Marjaneh employed to trick the burglar i, at time t. $best_i^t$ symbolizes the optimal location attained by thief I till the current time step (t). $gbest^t$ is the greatest worldwide rank attained by a thief as of the current t. r_1, r_2, $rand$, p, and q are values that are created at random and fall between [0,1] $p \geq 0.5$ indicates either a value of 0 or 1. y_i^t shows Ali Baba's location with respect to thief i at time t. The definition of a is given by using Eq. (7). One of two values may be assigned to $sgn(rand - 0.5) : -1 \text{or} 1$. Td^t stands for the thieves' tracking distance, as determined by Eq. (4). P_{p^t} reflects the thieves' possible capacity for perceptual detection of Ali Baba, as determined by Eq. (6).

$$Td^t = \tau_0 e^{-\tau_1\left(\frac{t}{T}\right)_1^\tau} \tag{5}$$

$$P_{p^t} = \lambda_0 \log\left(\lambda_1\left(\frac{t}{T}\right)^{\lambda_0}\right) \tag{6}$$

$$a = [(n-1).rand(n,1)] \tag{7}$$

where $\tau_0(\tau_0 = 1)$ serves as a preliminary tracking distance estimate. $\tau_1(\tau_1 = 2)$ is used to control how much exploration and exploitation are combined. t and T refer to the maximum and current iteration values, respectively. $\lambda_0(\lambda_0 = 1)$ signifies the ultimate estimation of the likelihood that the thieves will successfully achieve their objective after the search. $\lambda_1(\lambda_1 = 1)$ represents a constant used to regulate the balance between exploration and exploitation. $rand(n,1)$ is produced by generating a series of random numbers between 0 and 1.

$$m_{a(i)}^t = \begin{cases} x_i^t & if\ f(x_i^t) \geq f(m_{a(i)}^t) \\ m_{a(i)}^t & if\ f(x_i^t) < f(m_{a(i)}^t) \end{cases} \tag{8}$$

Here, $f(0)$ indicates the fitness function's value or score.

Second state: The thieves have the potential to realize that they have been misled, which could prompt them to venture into previously unexplored and unforeseen areas.

$$x_i^{t+1} = Td^t\left[\left(u_j - l_j\right)r + l_j\right]; p \geq 0.5, q \leq P_{p^t} \tag{9}$$

In this case, the bounds of the dimension j search space are denoted by u_j (the upper bound) and l_j (the lower bound). A random variable in the interval [0, 1] is called r.

Third state: To enhance both the exploration and exploitation aspects of the AFT algorithm, the thieves may extend their exploration to additional search positions beyond those determined by Eq. (4). This scenario can be formally represented by Eq. (10):

$$x_i^{t+1} = gbest^t - \left[Td^t\left(best_i^t - y_i^t\right)r_1 + Td^t\left(y_i^t - m_{a(i)}^t\right)r_2\right]sgn(rand - 0.5) \tag{10}$$

The basic AFT algorithm's iterative pseudo-code steps can be precisely presented as follows:

Define and begin the control parameters.
Begin and evaluate the initial, best, and global positions of all thieves
Begin Marjane's wit level concerning all thieves
Set $t \leftarrow 1$
While $(t \leq T)$ *do*
Update the parameter P_{p^t} *using Eq.* (6).
For each thief, do
if $(p \geq 0.5)$ *then*
if $(q \geq P_{p^t})$ *then*
Update the thieves' position using Eq. (5).
else
Update the thieves' position using Eq. (9).
end if
else
Update the thieves' position using Eq. (10).
end if
end for
Update the new, best, and global positions of all thieves
Update Marjane's wit plans using Eq. (8).
$t = t + 1$
end while
Return the best global solution

Improved Manta-Ray Foraging Optimization (IMRFO)

In the original Manta Ray Foraging Optimizer (MRFO) [40], the exploitation phase involves individuals updating their positions based on the best fitness individual. This can lead to reduced stagnation in local optima and population diversity. Additionally, MRFO exhibits weak solution stability due to its limited fine-tuning capacity. An improved version called IMRFO is introduced to address these limitations. IMRFO introduces a controlling factor of exploring to enhance search, employs a coefficient of adaptive weight with Levy flight to maximize diversity and maintain population balance, and incorporates Morlet wavelet mutation with the fine-tuning capability to prevent premature convergence to local optima and ensure solution stability.

Factor of exploring control

In MRFO, exploration is limited due to a low exploration probability, especially in the first half of the optimization process, governed by the value of t/T. In contrast, the IMRFO algorithm employs a factor of exploring control (p_s), to enhance exploration. This factor increases the likelihood of exploration in the latter optimization stages, offering improved search capabilities.

$$p_s(t) = \left(1 - \frac{t}{T}\right)\sqrt{\frac{5}{r}} \tag{11}$$

where r represents a random number within the interval $[0, 1]$. When the value of p_s exceeds 0.5, the IMRFO algorithm engages in exploration; conversely, it prioritizes exploitation when p_s falls below 0.5. The factor of exploring control exhibits a reducing trend coupled with random oscillations, compelling the algorithm to emphasize exploration during later iterations [41].

By defining θ as $\theta = 1 - \frac{t}{T}$, the expression for $p_s(t)$ becomes $p_s(t) = \sqrt{\frac{5}{r}}.\theta$. To determine the probability of p_s being greater than 0.5, it can be calculated as follows:

$$P\{A(t) > 0.5\} = 1 - \frac{\dfrac{1}{\dfrac{\int_0^{2\sqrt{5}} \int^1 5\, dr d\theta}{\left(\dfrac{0.5}{\theta}\right)^2}}}{1 \times 1} \approx 0.8509 \tag{12}$$

Consequently, the exploration probability within the IMRFO amounts to 0.8509×0.5, yielding a value of 0.4254 over the optimization process.

Coefficient of adaptive weight with Levy flight

Inspired initially by natural foraging behaviors, Levy flight has become a valuable tool for efficient exploration in unknown spaces, widely used in various metaheuristic algorithms. It enhances search behavior by incorporating Levy flight's characteristic of short and occasional long steps into the MRFO's cyclone foraging strategy. This adaptation fosters diversity among exploration single and guards against premature convergence to local optima.

The length of the random step in the Levy flight is determined via the Levy distribution, which is expressed as follows [42]:

$$L(s) \sim t^{-\lambda}(1 \leq \lambda \leq 3) \tag{13}$$

Here, λ represents the tail index or stability, while s denotes the length of the step. Following the algorithm of Mantegna, the length of step for the Levy flight is defined in the following:

$$s = u/|v|^{\frac{1}{\beta}} \tag{14}$$

The variables u and v follow normal distributions, and specifically:

$$u \sim N\left(0, \sigma_u^2\right), v \sim N(0,1) \tag{15}$$

$$\sigma_u = \left(\frac{\Gamma(1+\beta).\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right).\beta.2^{\frac{\beta-2}{2}}} \right)^{1/\beta} \tag{16}$$

Γ is employed in the computation, with a default value assigned to parameter β set at 1.5. Thus, the formulation of a coefficient of adaptive weight, integrating the Levy flight into the cyclone foraging techniques, is as follows:

$$\beta_L = e^{2(T-t+1)/T}.s \tag{17}$$

Observing Eq. (14) can discern two critical effects. Firstly, the frequent generation of multiple short steps by the Levy flight enhances the exploitation capacity of the algorithm. In contrast, occasional long steps bolster exploration, effectively ensuring local optima avoidance. Secondly, the function $e^{2(T-t+1)/T}$ exhibits a reducing trend by iterations, thus offering a larger exploration scope during early iterations and gradually narrowing it in later iterations. This characteristic enhances the algorithm's search efficiency and ensures that step lengths remain within the variable boundaries.

The cyclone foraging strategy employed in the IMRFO algorithm is defined as follows:

$$X_i^d(t+1) = \begin{cases} x_{best} + r.(x_{best}(t) - x_i(t)) + \beta_L.(x_{best}(t) - x_i(t)) i = 1 \\ x_{best} + r.(x_{i-1}(t) - x_i(t)) + \beta_L.(x_{best}(t) - x_i(t)) i = 2, \ldots, N \end{cases} p_s < 0.5 \tag{18}$$

$$X_i^d(t+1) = \begin{cases} x_{rand} + r.(x_{rand}(t) - x_i(t)) + \beta_L.(x_{rand}(t) - x_i(t)) i = 1 \\ x_{rand} + r.(x_{i-1}(t) - x_i(t)) + \beta_L.(x_{rand}(t) - x_i(t)) i = 2, \ldots, N \end{cases} p_s \geq 0.5 \tag{19}$$

Wavelet mutation strategy

The MRFO algorithm may encounter challenges in getting stuck in local optima, leading to an inefficient exploration of the instability of solutions and the global optimum. IMRFO incorporates Morlet wavelet mutation to enhance the algorithm's ability to break free from stagnation, improve convergence rates, and ensure solution stability. This wavelet mutation involves dynamically adjusting the mutation process by integrating wavelet function translations and dilations [42, 43]. In pursuit of fine-tuning objectives, control over the wavelet function's dilation parameters is exercised to reduce its amplitude, consequently constraining the mutation space as iterations progress.

Given that p_m represents the mutation probability and r_4 presented a random number within the [0, 1] range, the integration of wavelet mutation enhances the somersault foraging techniques as follows:

$$x_i(t+1) = \begin{cases} \begin{cases} x_i(t) + \sigma_w.(x_i(t) - Low)\sigma_w < 0 \\ x_i(t) + \sigma_w.\big(U_p - x_i(t)\big)\sigma_w \geq 0 \end{cases} r_4 < p_m \\ x_i(t) + S.(r_2.x_{best} - r_3.x_i(t))r_4 \geq p_m \end{cases} \tag{20}$$

where p_m represents the mutation probability, set to a value of 0.1, and σ_w denotes the wavelet function's dilation parameters, which can be defined in the following:

$$\sigma_w = \frac{1}{\sqrt{a}}\psi\left(\frac{\varphi_i}{a}\right) \tag{21}$$

where $\psi(x)$ corresponds to the wavelet function of Morlet, determined in the following:

$$\psi(x) = e^{-\frac{x^2}{2}}\cos(5x) \tag{22}$$

Over 90% of the overall energy of the wavelet function is concentrated within the range of $[-2.5, 2.5]$. Consequently, σ_w can be stochastically generated from the interval $[-2.5a, 2.5a]$, with "a" representing the dilation parameter. This parameter scales progressively from 1 to "s" as the iterations' number grows. To prevent overlooking the global optimum, a monotonically growing function is defined in the following:

$$a = e^{-\ln(g).\left(1-\frac{t}{T}\right)+\ln(g)} \tag{23}$$

Here, g is a fixed constant, with a value set to 100,000.

The fundamental steps of the IMRFO algorithm can be accurately represented through the following iterative pseudo-code [41]:

Initialize N (the size of the population) and T (the iteration maximal number), and initialize the rand
= 1, … , N.
Compute the fitness of each individual $f_i = f(x_i)$ *and obtain the best solution found so far* x_{best}.
For $i = 1$ *to N*
If rand < 0.5 *Then// cyclone foraging*
Calculate the search control factor p_s *using Eq.* (11);
If $p_s < 0.5$ *Then*
Update the individual position using Eq. (18);
Else
Update the individual position using Eq. (19);
End If
Else //Chain foraging
Update the individual position using Eq. (11);
End If
Compute the fitness of each individualf$(x_i(t+1))$;.
If $f(x_i(t+1)) < f(x_{best})$
Then
$x_{best} = x_i(t+1)$;
//Somersault foraging
For $i = 1$ *to N, Do*
Calculate the dilation parameter σ_w *using Eqs.* $(21-23)$;
Update the individual position using Eq. (20);
Compute the fitness of each individualf$(x_i(t+1))$;.
If $f(x_i(t+1)) < f(x_{best})$
Then
$x_{best} = x_i(t+1)$;
End For.
End While.
Return the best solution found so far, x_{best}.

Results assessment criteria

In this study, the effectiveness of prediction algorithms was rigorously evaluated using a comprehensive set of nine key metrics. These metrics served as performance indicators and included:

$$R^2 = \left(\frac{\sum_{i=1}^{n}(T_i - \overline{T})(P_i - \overline{P})}{\sqrt{\left[\sum_{i=1}^{n}(T_i - \overline{P})^2\right]\left[\sum_{i=1}^{n}(P_i - \overline{P})^2\right]}} \right)^2 \tag{24}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(T_i - P_i)^2}{n}} \tag{25}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(T_i - P_i)^2 \tag{26}$$

$$RSR = \frac{RMSE}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(T_i - \overline{T})^2}} \tag{27}$$

$$FB = \frac{1}{n}\sum_{i=1}^{n}\frac{2 \times (P_i - T_i)}{P_i + T_i} \tag{28}$$

$$SI = \frac{RMSE}{mean(T_i)} \tag{29}$$

$$NSE = 1 - \frac{\sum_{i=1}^{N}(T_i - P_i)^2}{\sum_{i=1}^{N}(P_i - \overline{P})^2} \tag{30}$$

$$n20 - index = \frac{n20}{n} \tag{31}$$

$$OBJ = \left(\frac{n_{train} - n_{test}}{n_{train} + n_{test}}\right)\frac{MAE_{test} + RMSE_{train}}{1 + R^2_{train}} + \left(\frac{2n_{train}}{n_{test} + n_{train}}\right)\frac{RMSE_{test} - MAE_{test}}{1 + R^2_{test}} \tag{32}$$

where T_i and P_i are actual and predicted values, respectively. $\overline{T}$ is the average of all the tested results, while n represents the number of samples in the analyzed dataset. $\overline{P}$ shows the mean of predicted value. n_{test} and n_{train} indicate the sample number of test and train, respectively.

Hybridization

The hybridization procedure integrates the KNN model with two distinct metaheuristic optimization algorithms: IMRFO for KNIM and AFT for KNAF. The detailed discussion of the hybridization process unfolds as follows:

➢ KNIM (KNN optimized with IMRFO):

- Initialization: The process begins by initializing the KNN model with a predefined set of hyperparameters, representing the starting configuration before optimization.
- Optimization with IMRFO: IMRFO is applied to the initialized KNN model, orchestrating an iterative optimization process. This involves refining the hyperparameters to augment the model's performance over successive iterations.
- Final KNIM model: The outcome of the IMRFO optimization process yields the optimized KNN model, denoted as KNIM. This refined version incorporates improved parameter configurations, enhancing its predictive capabilities for UCS prediction in rocks.

➢ KNAF (KNN optimized with AFT):

- Initialization: KNAF commences with the initialization of the KNN model, initially set with default hyperparameters, providing a baseline for subsequent optimization.
- Optimization with AFT: The AFT algorithm is employed to optimize the hyperparameters of the KNN model. AFT, with its exploratory nature, iteratively explores the solution space to pinpoint optimal hyperparameter configurations.
- Final KNAF model: The conclusion of the AFT optimization process results in the finalized KNN model, identified as KNAF. This optimized version reflects superior hyperparameter settings, enhancing the KNN model's effectiveness in predicting UCS in challenging geological conditions.

Metaheuristic optimization hyperparameters

➢ IMRFO:

- Parameters: IMRFO involves configuring parameters such as population size, maximum iterations, and the exploration–exploitation trade-off. The specific values are contingent upon the study's implementation details.

➢ AFT:

- Parameters: AFT encompasses parameters like population size, maximum iterations, and potential exploration–exploitation parameters. The actual values utilized in the study are explicitly defined during the AFT optimization process.

This meticulous consideration of metaheuristic optimization algorithms and their associated hyperparameters ensures a systematic and effective approach to enhancing the KNN models (KNIM and KNAF) for precise UCS prediction in the domain of rock

Table 3 Hyperparameters of developed models

Model	Hyperparameter		
	n_neighbors	leaf_size	*p*
KNN	5	30	2
KNIM	1	1	2
KNAF	31	306	958

mechanics. The hyperparameters for the KNN models, including the default and optimized versions, are reported in Table 3.

These hyperparameters represent the key configurations of the KNN models within the study, both before and after optimization using IMRFO and AFT.

Convergence

In the study, convergence is assessed based on the RMSE, a key metric for evaluating the accuracy of regression models. The metaheuristic optimization algorithms, IMRFO and AFT, iteratively refine the hyperparameters of the KNN model. Convergence is observed through the RMSE, which ideally decreases over iterations, signifying improved alignment between predicted and actual UCS values. The process stabilizes when further iterations cease to enhance model performance significantly. The final RMSE at convergence serves as a crucial indicator of optimized model accuracy in predicting UCS in rocks, aligning with the study's goal of improving predictive capabilities. Figure 3 shows the convergence of developed hybrid models.

System configuration

The system configuration for model development involved a hardware setup with an Intel Core i7-3770K CPU running at 3.50 GHz, complemented by 16.0 GB of RAM and a 1-terabyte hard drive. The operating system used was Windows 11 Pro, designed for a 64-bit

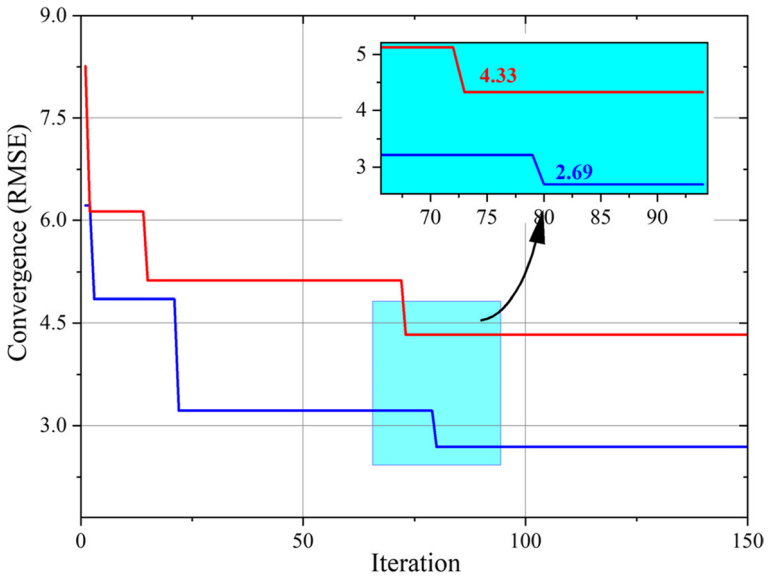

Fig. 3 Convergence of developed hybrid models

architecture. An NVIDIA GeForce GT 640 GPU facilitated graphics processing. In terms of software, Python served as the primary programming language for conducting experiments and developing models. The scikit-learn machine learning framework played a central role in implementing various machine learning algorithms and models. Additionally, for data analysis and visualization tasks, the study leveraged popular Python libraries, including Pandas, NumPy, and Matplotlib.

Run time

Table 4 displays the run time of the developed models, showcasing the computational efficiency of each model. The results indicate that KNN has the shortest run time at 0.1629 s, followed by KNIM with a run time of 148.73 s, and KNAF with the longest run time at 354.57 s.

Results and discussion

In the current study, the research aims to select the optimal UCS estimation model by examining the performance of KNN-based models. These models, including single KNN and hybrid models optimized with AFT and IMRFO, are organized into training, validation, and testing sets. The evaluation employs nine metrics to assess their performance, with results presented in Table 5 and visualized in various graphs to facilitate comparative analysis and model selection.

Table 4 Run time of developed models

Model	Run time (second)
KNN	0.1629
KNIM	148.73
KNAF	354.57

Table 5 The result of developed KNN-based models

Model	Phase	Index values								
		RMSE (MPa)	R^2	MSE (MPa)	RSR	FB (MPa)	SI	NSE	n20-index	OBJ
KNN	Train	4.221	0.974	17.823	0.048	0.0011	0.095	0.973	0.750	4.910
	Validation	6.445	0.953	41.543	0.052	0.0059	0.101	0.930	0.792	
	Test	5.234	0.970	27.392	0.053	0.0055	0.106	0.967	0.757	
	All	4.778	0.971	22.829	0.050	0.0008	0.100	0.968	1.000	
KNIM	Train	1.899	0.995	3.602	0.021	0.0005	0.043	0.994	0.932	2.792
	Validation	3.457	0.986	11.950	0.027	0.0026	0.054	0.980	1.000	
	Test	4.387	0.986	19.245	0.045	0.0041	0.089	0.977	1.000	
	All	2.688	0.991	7.223	0.028	0.0004	0.056	0.990	0.953	
KNAF	Train	3.297	0.985	10.871	0.037	0.0007	0.075	0.983	1.000	4.267
	Validation	6.358	0.970	40.425	0.050	0.0048	0.100	0.932	1.000	
	Test	5.766	0.980	33.248	0.059	0.0049	0.117	0.959	1.000	
	All	4.326	0.980	18.710	0.045	0.0006	0.090	0.974	1.000	

Table 5 provides a comprehensive overview of the performance metrics, including R^2, RMSE, MSE, FB, SI, NSE, n20-index, OBJ, and RSR, for all prediction models within the training, validation, and testing sets. A detailed examination of the model's predictive capabilities in estimating the UCS of rock samples is presented in the subsequent analysis:

- The KNIM hybrid model demonstrated remarkable performance with maximum R^2 values of 0.995 during the training phase and 0.986 each for both the validation and testing phases. These high R^2 values signify that this model effectively explains a substantial portion of the variance in the UCS through the incorporated input variables. In essence, this shows that the model and the data fit together well, highlighting the validity of the selected input variables as reliable indicators of the anticipated result.
- Regarding error values, the KNIM model is the most accurate among the developed models, exhibiting approximately twofold and threefold lower RMSE values than the KNAF and KNN models. This suggests that KNIM provides superior predictive accuracy and is associated with significantly more minor discrepancies between predicted and actual values than the other models, emphasizing its effectiveness in UCS estimation.
- A minimum RSR value of 0.021, observed in the KNIM model, indicates an excellent fit where the model's predictions closely match actual data. This suggests that the model effectively captures the variability in UCS while keeping standardized residuals relatively small, signifying its accuracy and reliability.
- The minimum FB value of 0.0005 (MPa) observed in the KNIM model during the training phase indicates minimal bias in its predictions. This suggests that the model offers highly accurate and unbiased estimates, aligning closely with the data.
- The KNIM model's ability to consistently achieve the highest NSE and n20-index values further reinforces its superior performance in capturing the underlying patterns and variability in the data.
- The lower OBJ and SI values for KNIM reinforce its superior overall performance and robustness in handling extreme values compared to the other KNN-based models (KNN and KNAF).

In the assessment of the models, it is evident that overfitting did not occur. Overfitting manifests when there is a substantial disparity between the training and test outputs, a phenomenon conspicuously absent in the outcomes of these models.

Figure 4 provides graphical representations illustrating the relationships between experimentally determined UCS values and their corresponding predictions. This study employs advanced quantitative data analysis techniques, focusing on two key evaluation metrics: RMSE and R^2. RMSE helps govern data dispersion, with reduced values indicating denser and more concentrated findings. The R^2 evaluation brings data points closer to the central axis, enhancing alignment. The diagram features critical elements such as the central line at $Y=X$, a linear regression model, and four boundary lines at $Y=0.9X$ and $Y=1.1X$, denoting 10% underestimation and overestimation, respectively. The KNIM model exhibits an optimal concentration of predicted UCS values around the

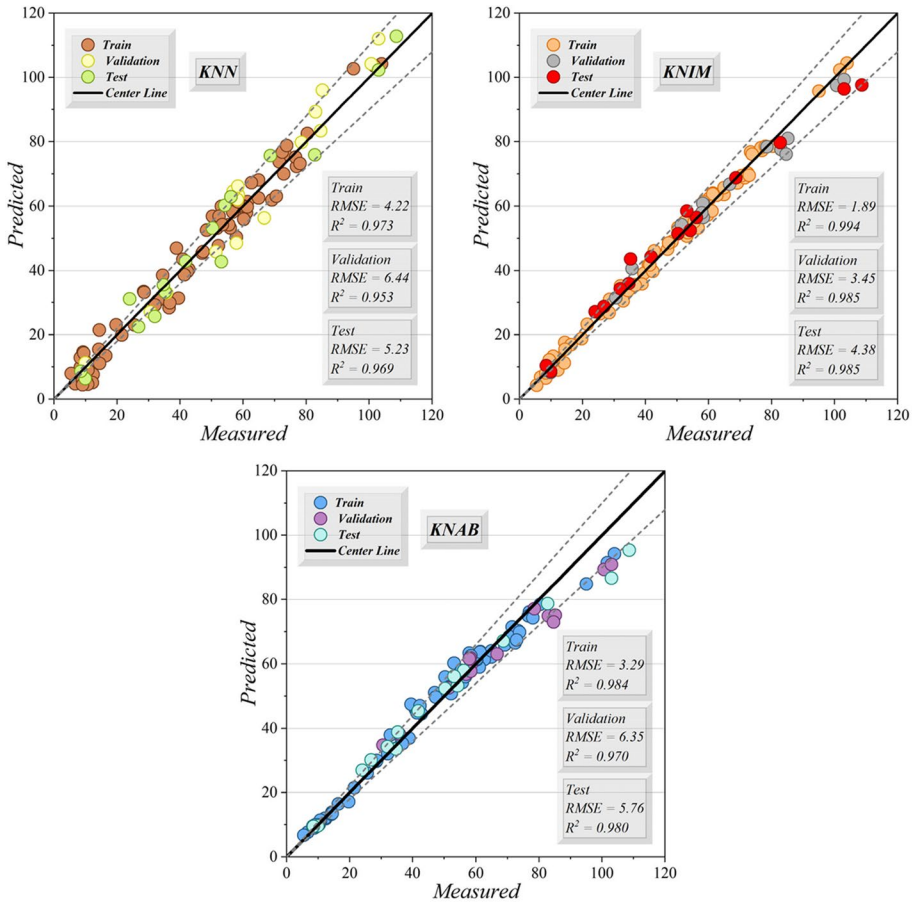

Fig. 4 The scatter plot for developed hybrid models

central line, outperforming the single KNN model and another hybrid model (KNAF), which show scattered data dispersion beyond an acceptable range.

Figure 5 highlights the excellent alignment between observed and predicted UCS values across all three models, with the KNIM model excelling. In this figure, the dashed line shown in the figure connects the prediction points to each other. At the same time, the maximum prediction-measurement difference is attributed to the KNAF model. All three models experienced maximum variation between measured and estimated values in their testing phase.

Figures 6 and 7 demonstrate the error percentage in histogram density and half-box plots. According to Fig. 5, for KNIM and KNAF, the percentage of errors near zero percent is almost twice the KNN model, indicating that optimizing the KNN model with two selected optimizers decreased error values. When comparing the range of error values of the models provided in Fig. 6, it can be observed that KNN in the training phase and KNN in the validation phase have the most and most minor broad range of error values. The KNIM hybrid model performs better in all three phases, while KNAF with marginal variation ranks second.

Furthermore, the best model (KNIM) and measured values are mentioned in Appendix 2.

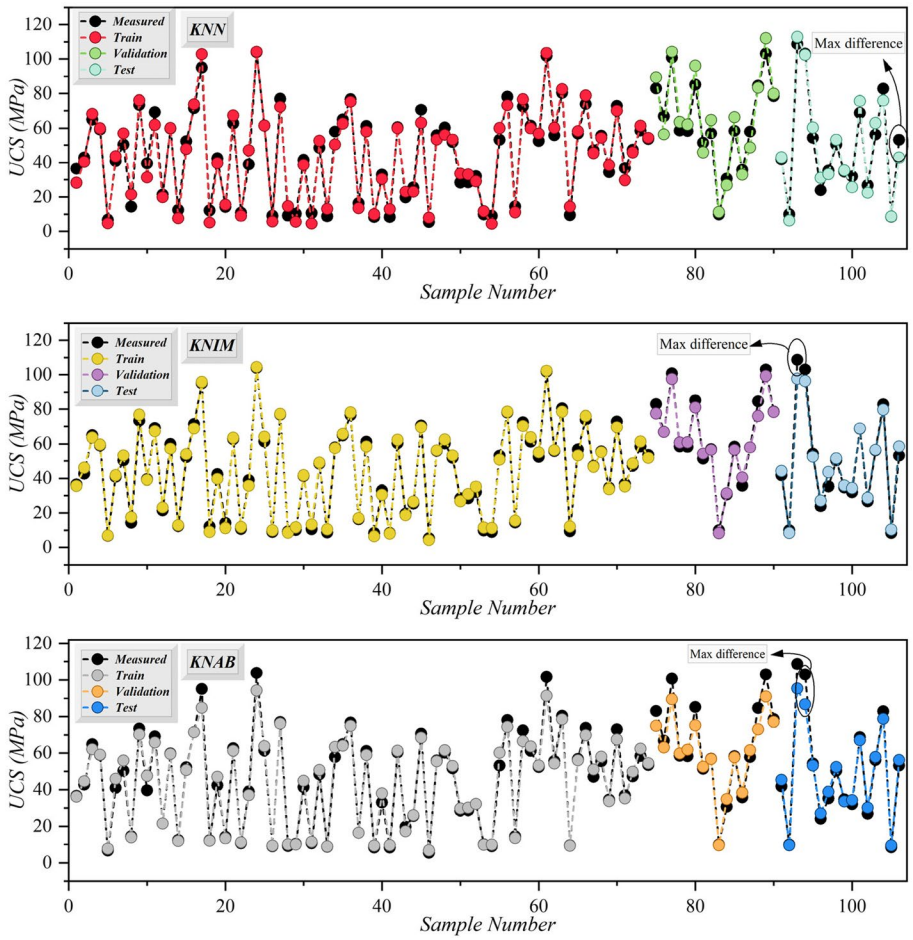

Fig. 5 The comparison of predicted and measured values

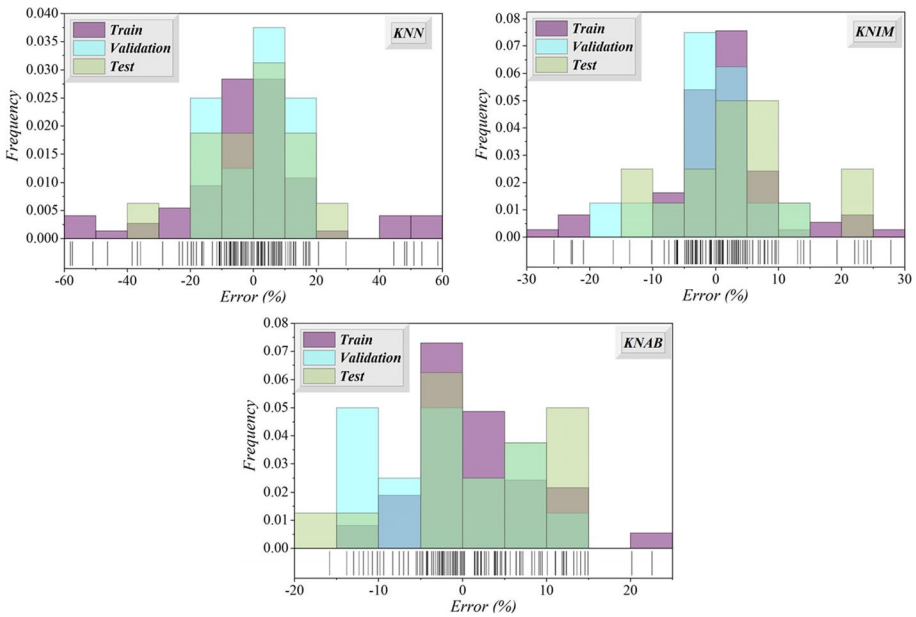

Fig. 6 Error percentage for the hybrid models based on a histogram density plot

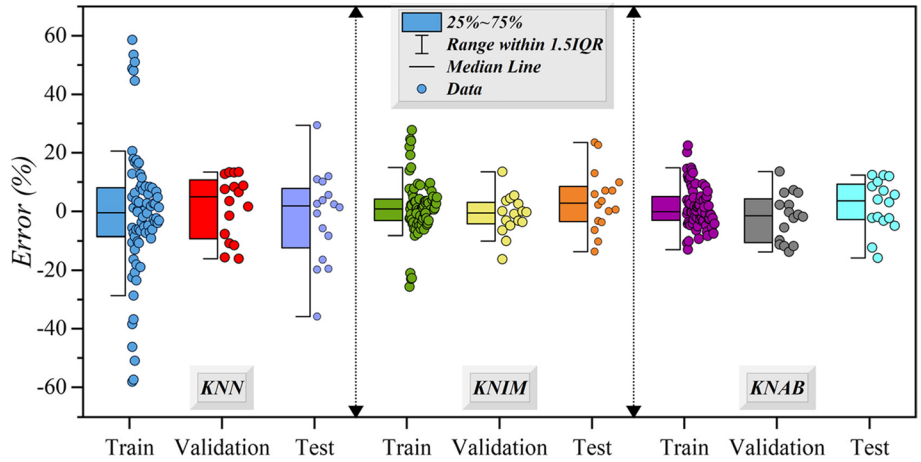

Fig. 7 The half box of errors among the developed models

Discussion

Main limitations

Despite the promising outcomes of this study, certain limitations need consideration. Firstly, the predictive models developed, while effective in the context of UCS prediction for specific rock types, may exhibit variability in performance across diverse geological formations. The models' reliance on selected variables might limit their generalizability to rock types not represented in the current dataset. Additionally, the reliance on laboratory test data for model training raises questions about the applicability of the models to real-world field conditions.

Wider applicability of methods

The methods employed in this study, particularly the use of K-nearest neighbors (KNN) and optimization algorithms (Alibaba and the Forty Thieves, Improved Manta-Ray Foraging Optimizer), offer a robust framework for UCS prediction. However, the wider applicability of these methods extends beyond rock mechanics. Similar methodologies could find utility in diverse fields such as geotechnical engineering, materials science, and environmental studies, where predicting material strength properties is paramount. The simplicity and interpretability of KNN, coupled with the optimization algorithms, contribute to the versatility of the proposed approach.

Findings and implications

The findings indicate that the optimized KNN model, particularly KNIM, outperforms the baseline KNN and KNAF models in terms of accuracy and predictive capabilities for UCS prediction in certain rock types. The marginal differences observed in the Wilcoxon test between KNAF and KNIM warrant attention, signaling potential avenues for model refinement. The efficient run time of KNN makes it a favorable choice for real-time applications, while the optimization algorithms enhance its performance.

Comparison

Table 6 provides a comprehensive comparison between the presented study and previously published articles in terms of the model used, evaluation metrics (R^2 and RMSE), and their respective performance. The inclusion of diverse models and evaluators highlights the broader context of UCS prediction methodologies in the literature. The present study, utilizing the KNIM model, demonstrates competitive results with a high R^2 of 0.991 and an RMSE of 2.688.

Wilcoxon test

The Wilcoxon test was employed to assess the comparative performance of three models: KNN, KNIM, and KNAF. The test results, considering p values and statistics for each pair of models, provide insights into their statistical significance. Table 7 shows the result of the Wilcoxon test.

The Wilcoxon test results reveal that there is no statistically significant difference in performance between KNN and KNIM (p value = 0.9585, statistic = 2819) as well as between KNN and KNAF (p value = 0.7135, statistic = 2719). These findings suggest comparable performance between these model pairs. However, the comparison between KNAF and KNIM indicates a marginally significant difference (p value = 0.0902, statistic = 2298). While not reaching conventional levels of significance, this result suggests a potential difference that may warrant further investigation or consideration. In summary, the Wilcoxon test suggests comparable performance between KNN and KNIM and between KNN and KNAF. The KNAF and KNIM pair shows a marginally significant difference, indicating the need for cautious interpretation and potential further exploration.

Table 6 Comparison between the presented and published articles

Article	Model	Evaluator	
		R^2	RMSE
Narendra et al. [29]	GP	0.9881	135
Ceryan et al. [30]	REG	0.8837	1.108
Majdi and Rezaei [31]	ANN	0.9725	1.113
Rezaei et al. [10]	Mamdani fuzzy	0.9437	3.2
Mohamad et al. [32]	ANN-PSO	0.982	0.077
Present study	KNIM	0.991	2.688

Table 7 Result of Wilcoxon test

Difference of models	Parameter	
	p value	Statistic
Def. between KNN and KNIM	0.9585	2819
Dif. between KNN and KNAF	0.7135	2719
Dif. between KNAF and KNIM	0.0902	2298

Sensitivity analyses

SHAP value

The analysis utilizing SHAP (SHapley Additive exPlanations) values offers a comprehensive understanding of how various variables influence the model's output. Illustrated in Fig. 8, the SHAP values elucidate the nuanced impact of inputs on the model's predictions. The findings underscore the significant influence of Vp, DD, and Is on the model's UCS predictions, aligning closely with geological principles. Conversely, BD exhibits a comparatively lesser impact on UCS. Such insights not only enhance the interpretability of the model but also furnish invaluable guidance for researchers and practitioners in the field, facilitating informed decision-making and furthering understanding of geological processes.

Conclusions

In summary, this research used hybrid machine learning models to predict unconfined compressive strength (UCS) in rock samples. The study addresses the challenges of sample preparation by developing and evaluating these hybrid models, which incorporate the K-nearest neighbor (KNN) model optimized with Alibaba and the Forty Thieves (AFT), as well as Improved Manta-Ray Foraging Optimizer (IMRFO). Based on the assessment findings, the KNIM model emerges as the most optimal choice, boasting the following advantages:

- Improvements over KNN single model: KNIM demonstrates substantial enhancements over the KNN single model, featuring a 2% higher R^2 value, a notable 43.7% reduction in RMSE, and a substantial 68.4% decrease in MSE values. These results underscore KNIM's superior predictive accuracy and ability to reduce prediction errors significantly.

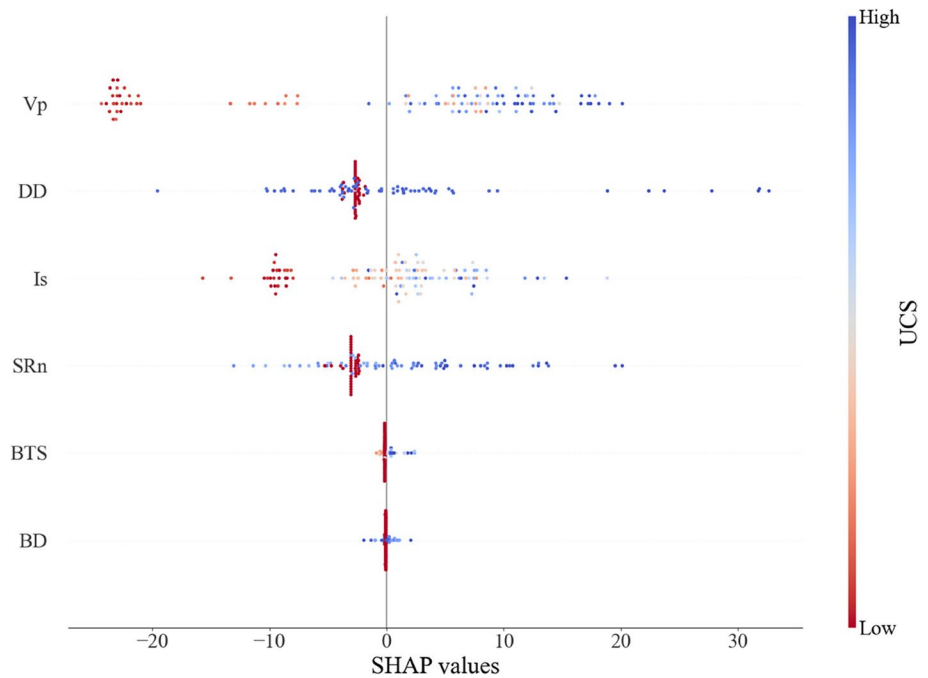

Fig. 8 Impact of inputs on model's output

- Enhancements over KNAF hybrid model: Compared to another hybrid model, KNAF, KNIM exhibits noteworthy enhancements, including a 1.1% higher R^2 value, indicative of a better data fit. Additionally, KNIM achieves a considerable 37.8% reduction in RMSE and a substantial 61.4% decrease in MSE values, underscoring its superiority in predictive accuracy and error reduction.

These findings collectively establish KNIM as a highly effective and promising model for UCS prediction in rock samples, offering valuable insights for applications in civil engineering. The study's findings apply primarily to the specific dataset and context of predicting UCS in rocks. The transferability of the proposed optimization algorithms, AFT and IMRFO, requires validation across diverse datasets and applications. The study's relevance to different rock types and engineering scenarios needs careful consideration. Limitations include the reliance on a specific dataset, the sensitivity of optimization algorithms to conditions and hyperparameters, assumptions of data stationarity, and the exclusion of potentially relevant variables. Robustness testing, external validation, and consideration of real-world variability are suggested for future research to address these limitations. The study's findings suggest several potential directions for future research to enhance predictive models for UCS. The recommendations include exploring additional variables (geological, geophysical, or mineralogical), conducting temporal variability analysis, assessing regional specificity, exploring ensemble models, incorporating advanced machine learning techniques, validating models on diverse datasets, performing real-world testing and field validation, exploring alternative optimization algorithms, and integrating remote sensing data. These avenues aim to advance the accuracy and reliability of predictive models for UCS in rocks beyond the current study.

Appendix 1

Full dataset						
BD (kg/m^3)	**BTS (MPa)**	**DD (kg/m^3)**	**Vp (m/s)**	**SRn (MPa)**	**Is(50) (MPa)**	**UCS (MPa)**
2.28	3.5	0	2576	0	2.7	36.5
0.00	0	2628	5193	38.4	1.02	42.79
0.00	0	2612	3863	42.8	4.27	64.91
0.00	0	2693	6103	39	2.49	58.99
2.09	1.3	0	1776	0	0.5	6.7
0.00	0	2589	5163	41	2.37	40.9
0.00	0	2572	4423	41	3.56	50.32
2.52	1.6	0	1852	0	0.5	14.4
0.00	0	2679	6000	36.8	2.29	73.38
0.00	0	2708	5550	35.3	3.42	39.53
0.00	0	2702	6083	35.6	3.92	69.14
2.71	2.2	0	2167	0	1.7	21.5
0.00	0	2728	6080	36	1.37	59.73
2.16	1.7	0	1967	0	0.2	12.4
0.00	0	2637	5190	40.2	1.46	52.2
0.00	0	2663	4813	38.9	3.35	71.5

Full dataset						
BD (kg/m^3)	BTS (MPa)	DD (kg/m^3)	Vp (m/s)	SRn (MPa)	Is(50) (MPa)	UCS (MPa)
0.00	0	2662	5267	40.6	5.39	95.03
2.47	1.7	0	1634	0	0.7	12.2
0.00	0	2606	3707	42.8	2.22	42.31
2.24	2.4	0	1912	0	0.2	14.2
0.00	0	2826	5073	31.8	1.81	62.68
2.21	2	0	1901	0	0.2	10.8
2.68	2.5	0	2489	0	2.7	38.9
0.00	0	2660	5430	42.3	3.57	103.93
0.00	0	2691	5530	36.5	1.38	61.42
2.18	1.4	0	2030	0	0.1	9.1
0.00	0	2641	5380	45.4	2.41	76.97
2.20	1.4	0	1947	0	0.2	9.2
2.31	0.9	0	1596	0	0.4	10.3
2.66	3.6	0	2717	0	2.7	41.4
2.23	1	0	1871	0	0.2	10.7
0.00	0	2654	5737	42.6	3.2	48.43
2.61	1.6	0	1406	0	0.6	8.8
0.00	0	2645	5430	41.6	2.57	57.85
0.00	0	2706	5867	38.4	1.94	64.91
0.00	0	2644	4930	39.6	2.02	76.67
2.70	1.7	0	2134	0	1	16.4
3.54	4.2	0	2847	0	4.1	61.1
2.26	0.7	0	1928	0	0.1	8.4
0.00	0	2682	5173	30.8	1.94	32.95
2.46	1.7	0	1820	0	0.4	8.4
0.00	0	2672	5933	31.9	2.06	60.35
2.39	1.7	0	1820	0	1	19.7
2.75	2.1	0	2194	0	2	25.6
0.00	0	2628	5233	42.8	3.62	70.56
2.18	0.8	0	1297	0	0.3	5.5
3.52	3.8	0	2897	0	3.9	55.9
0.00	0	2641	5257	42.8	5.47	60.2
0.00	0	2664	5740	34.5	5.28	51.73
2.65	2.3	0	2356	0	1.9	28.5
2.51	2	0	2417	0	1.5	28.6
2.83	2.6	0	2378	0	2.9	32.1
2.22	1.9	0	1988	0	0.2	10
2.50	1.5	0	1754	0	0.5	9.1
0.00	0	2622	5390	38.8	3.97	53.14
0.00	0	2769	5987	38	5.57	78.09
2.24	2.7	0	1909	0	0.2	14.5
0.00	0	2641	5073	41.8	4.43	72.44
0.00	0	2718	6013	35.5	2.9	61.3
2.53	3.8	0	2857	0	3.4	52.4
0.00	0	3011	6440	38.8	4.87	101.7
0.00	0	2697	5670	36.2	4.07	55.67
0.00	0	2636	5420	42.2	4.63	80.44
2.16	0.8	0	1247	0	0.3	9.5
3.47	3.5	0	2832	0	3.9	56.7

Full dataset						
BD (kg/m^3)	**BTS (MPa)**	**DD (kg/m^3)**	**Vp (m/s)**	**SRn (MPa)**	**Is(50) (MPa)**	**UCS (MPa)**
0.00	0	2750	6180	33	2.7	73.84
0.00	0	2671	6093	37.2	2.55	47.02
0.00	0	2677	5910	38	2.65	55.49
2.81	3.5	0	2320	0	2.3	34.5
0.00	0	2607	5083	41.2	2.34	72.91
2.80	2.1	0	2344	0	2.4	36.8
3.44	3.7	0	2857	0	3.7	47.3
0.00	0	2786	6337	32.4	5.88	58.47
0.00	0	2592	3143	31.4	3.99	53.45
0.00	0	2922	5757	31.6	4.14	83
0.00	0	2696	5843	36.3	2.54	66.74
0.00	0	2846	6070	38.7	5.23	100.68
0.00	0	2682	6013	37.4	2.1	58.51
0.00	0	2620	4650	44.7	2.56	58.2
0.00	0	2620	3923	43.6	5.61	85.15
3.42	4.1	0	2910	0	3.6	51.4
0.00	0	2573	2797	31.8	2.17	56.91
2.65	1.6	0	1417	0	0.6	9.9
0.00	0	2518	3543	33.8	0.69	30.59
0.00	0	2573	3420	35.9	2.11	58.32
0.00	0	2664	5923	37.2	3.68	35.72
0.00	0	2701	6003	38.8	3.68	57.85
0.00	0	2887	5630	34.9	3.27	84.62
0.00	0	2952	6070	44.3	4.87	102.99
0.00	0	2669	5393	43.1	3.15	78.56
0.00	0	2703	5720	36.5	2.41	41.8
2.33	1.3	0	1330	0	0.5	9.9
0.00	0	2825	6363	40	2.66	108.68
0.00	0	2637	3790	44.2	3.21	102.99
0.00	0	2719	5590	34.7	2.33	54.24
0.00	0	2516	3643	28.2	0.58	24
0.00	0	2737	5870	38.6	2.09	35.25
0.00	0	2622	5857	35.8	3.62	50.32
2.72	2.9	0	2275	0	2.3	34.7
0.00	0	2670	5430	40.2	3.57	32.01
0.00	0	2620	5263	43.6	4.57	68.67
0.00	0	2688	5073	34.2	2.2	26.83
0.00	0	2606	4333	42.6	2.75	56.14
0.00	0	2600	4813	38.9	6.07	82.79
2.23	1.9	0	1928	0	0.1	8.5
0.00	0	2715	6200	34.4	2.36	53.14

Appendix 2

Obtained value based on measured and predicted UCS	
UCS106_Measured	**UCS106_ Predicted (KNIM)**
36.5	35.618
42.79	46.0905
64.91	63.5905
58.99	59.5185
6.7	6.85E + 00
40.9	41.698
50.32	53.2985
14.4	17.5815
73.38	76.8065
39.53	39.1485
69.14	67.152
21.5	23.187
59.73	57.182
12.4	12.806
52.2	53.9115
71.5	6.88E + 01
95.03	95.7335
12.2	9.071
42.31	39.7445
14.2	1.12E + 01
62.68	63.359
10.8	11.8185
38.9	35.7255
103.93	104.462
61.42	64.0415
9.1	9.92E + 00
76.97	77.2535
9.2	8.6295
10.3	11.742
41.4	41.757
10.7	1.33E + 01
48.43	49.018
8.8	1.05E + 01
57.85	57.4335
64.91	65.6765
76.67	78.126
16.4	1.69E + 01
61.1	58.4645
8.4	6.475
32.95	30.507
8.4	8.0955
60.35	62.1945
19.7	18.7375
25.6	26.531
70.56	69.4785

Obtained value based on measured and predicted UCS	
UCS106_Measured	**UCS106_ Predicted (KNIM)**
5.5	4.25E + 00
55.9	56.1135
60.2	62.377
51.73	5.32E + 01
28.5	26.792
28.6	30.973
32.1	35.194
10	11.5015
9.1	11.2945
53.14	50.711
78.09	7.86E + 01
14.5	15.4835
72.44	70.2455
61.3	63.708
52.4	55.0425
101.7	102.309
55.67	56.2965
80.44	78.5815
9.5	12.1425
56.7	53.2
73.84	76.087
47.02	46.6665
55.49	55.133
34.5	33.6845
72.91	69.557
36.8	35.3195
47.3	48.553
58.47	61.2945
53.45	51.798
83	77.5968
66.74	66.7774
100.68	97.5208
58.51	60.611
58.2	60.807
85.15	81.0558
51.4	54.183
56.91	56.4282
9.9	8.294
30.59	31.371
58.32	56.461
35.72	40.5512
57.85	57.8742
84.62	76.1126
102.99	99.2918
78.56	78.3818
41.8	44.2552
9.9	8.5514
108.68	97.6316

Obtained value based on measured and predicted UCS	
UCS106_Measured	**UCS106_ Predicted (KNIM)**
102.99	96.4054
54.24	52.4734
24	27.1262
35.25	43.5572
50.32	51.442
34.7	35.9276
32.01	34.2694
68.67	68.7514
26.83	28.7308
56.14	56.4766
82.79	79.7298
8.5	10.4272
53.14	58.4376

Acknowledgements
I would like to take this opportunity to acknowledge that there are no individuals or organizations that require acknowledgment for their contributions to this work.

Authors' contributions
All authors contributed to the study's conception and design. Data collection, simulation, and analysis were performed by LN, QC, JL, HH, and JZ. The first draft of the manuscript was written by LN and all authors commented on previous versions of the manuscript. All authors have read and approved the manuscript.

Funding
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Availability of data and materials
Data can be shared upon request.

Declarations

Competing interests
The authors declare no competing interests.

Received: 28 November 2023 Accepted: 12 June 2024
Published online: 19 June 2024

References

1. Singh R, Kainthola A, Singh TN (2012) Estimation of elastic constant of rocks using an ANFIS approach. Appl Soft Comput 12:40–45
2. Monjezi M, Amini Khoshalan H, Razifard M (2012) A neuro-genetic network for predicting uniaxial compressive strength of rocks. Geotech Geol Eng 30:1053–1062
3. Kahraman S (2014) The determination of uniaxial compressive strength from point load strength for pyroclastic rocks. Eng Geol 170:33–42
4. Azarafza M, Hajialilue Bonab M, Derakhshani R (2022) A deep learning method for the prediction of the index mechanical properties and strength parameters of marlstone. Materials (Basel) 15:6899
5. Luo J, He J (2022) Constitutive model and fracture failure of sandstone damage under high temperature–cyclic stress. Materials (Basel) 15:4903
6. Tandon RS, Gupta V (2015) Estimation of strength characteristics of different Himalayan rocks from Schmidt hammer rebound, point load index, and compressional wave velocity. Bull Eng Geol Environ 74:521–533
7. Ghanizadeh AR, Abbaslou H, Amlashi AT, Alidoust P (2019) Modeling of bentonite/sepiolite plastic concrete compressive strength using artificial neural network and support vector machine. Front Struct Civ Eng 13:215–239
8. Amlashi AT, Alidoust P, Ghanizadeh AR, Khabiri S, Pazhouhi M, Monabati MS (2022) Application of computational intelligence and statistical approaches for auto-estimating the compressive strength of plastic concrete. Eur J Environ Civ Eng 26:3459–3490. https://doi.org/10.1080/19648189.2020.1803144
9. behnam Sedaghat, Tejani GG, Kumar S. Predict the maximum dry density of soil based on individual and hybrid methods of machine learning. Adv Eng Intell Syst 2023;002. https://doi.org/10.22034/aeis.2023.414188.1129.

10. Rezaei M, Majdi A, Monjezi M (2014) An intelligent approach to predict unconfined compressive strength of rock surrounding access tunnels in longwall coal mining. Neural Comput Appl 24:233–241
11. Su C, Xu S, Zhu K, Zhang X (2020) Rock classification in petrographic thin section images based on concatenated convolutional neural networks. Earth Sci Informatics 13:1477–1484
12. Saporetti CM, Goliatt L, Pereira E (2021) Neural network boosted with differential evolution for lithology identification based on well logs information. Earth Sci Informatics 14:133–140
13. Di Y, Wang E, Li Z, Liu X, Li B (2021) Method for EMR and AE interference signal identification in coal rock mining based on recurrent neural networks. Earth Sci Informatics 14:1521–1536
14. Moradi M, Basiri S, Kananian A, Kabiri K (2015) Fuzzy logic modeling for hydrothermal gold mineralization mapping using geochemical, geological, ASTER imageries and other geo-data, a case study in Central Alborz. Iran Earth Sci Informatics 8:197–205
15. Amlashi AT, Abdollahi SM, Goodarzi S, Ghanizadeh AR (2019) Soft computing based formulations for slump, compressive strength, and elastic modulus of bentonite plastic concrete. J Clean Prod 230:1197–1216
16. Chen L, Asteris PG, Tsoukalas MZ, Armaghani DJ, Ulrikh DV, Yari M (2022) Forecast of airblast vibrations induced by blasting using support vector regression optimized by the grasshopper optimization (SVR-GO) technique. Appl Sci 12:9805
17. Koopialipoor M, Asteris PG, Mohammed AS, Alexakis DE, Mamou A, Armaghani DJ (2022) Introducing stacking machine learning approaches for the prediction of rock deformation. Transp Geotech 34:100756
18. Zhu W, Rad HN, Hasanipanah M (2021) A chaos recurrent ANFIS optimized by PSO to predict ground vibration generated in rock blasting. Appl Soft Comput 108:107434
19. Ghanizadeh AR, Safi Jahanshahi F, Naseralavi SS. Intelligent modelling of unconfined compressive strength of cement stabilised iron ore tailings: a case study of Golgohar mine. Eur J Environ Civ Eng n.d.:1–29. https://doi.org/10.1080/19648189.2023.2276133.
20. Meulenkamp F, Grima MA (1999) Application of neural networks for the prediction of the unconfined compressive strength (UCS) from Equotip hardness. Int J Rock Mech Min Sci 36:29–39
21. Sonmez H, Tuncay E, Gokceoglu C (2004) Models to predict the uniaxial compressive strength and the modulus of elasticity for Ankara agglomerate. Int J Rock Mech Min Sci 41:717–729
22. Gokceoglu C, Zorlu K (2004) A fuzzy model to predict the uniaxial compressive strength and the modulus of elasticity of a problematic rock. Eng Appl Artif Intell 17:61–72
23. Dehghan S, Sattari G, Aliabadi MA (2010) Prediction of uniaxial compressive strength and modulus of elasticity for travertine samples using regression and artificial neural networks. Min Sci Technol 20:41–6. https://doi.org/10.1016/S1674-5264(09)60158-7
24. Mishra DA, Basu A (2013) Estimation of uniaxial compressive strength of rock materials by index tests using regression analysis and fuzzy inference system. Eng Geol 160:54–68
25. Cevik A, Sezer EA, Cabalar AF, Gokceoglu C (2011) Modeling of the uniaxial compressive strength of some clay-bearing rocks using neural network. Appl Soft Comput 11:2587–2594
26. Yesiloglu-Gultekin N, Gokceoglu C, Sezer EA (2013) Prediction of uniaxial compressive strength of granitic rocks by various nonlinear tools and comparison of their performances. Int J Rock Mech Min Sci 62:113–122
27. Skentou AD, Bardhan A, Mamou A, Lemonis ME, Kumar G, Samui P et al (2023) Closed-form equation for estimating unconfined compressive strength of granite from three non-destructive tests using soft computing models. Rock Mech Rock Eng 56:487–514. https://doi.org/10.1007/s00603-022-03046-9
28. Le T-T, Skentou AD, Mamou A, Asteris PG (2022) Correlating the unconfined compressive strength of rock with the compressional wave velocity effective porosity and schmidt hammer rebound number using artificial neural networks. Rock Mech Rock Eng 55:6805–6840. https://doi.org/10.1007/s00603-022-02992-8
29. Narendra BS, Sivapullaiah PV, Suresh S, Omkar SN (2006) Prediction of unconfined compressive strength of soft grounds using computational intelligence techniques: a comparative study. Comput Geotech 33:196–208
30. Ceryan N, Okkan U, Kesimal A (2013) Prediction of unconfined compressive strength of carbonate rocks using artificial neural networks. Environ Earth Sci 68:807–819
31. Majdi A, Rezaei M (2013) Prediction of unconfined compressive strength of rock surrounding a roadway using artificial neural network. Neural Comput Appl 23:381–389
32. Mohamad ET, Jahed Armaghani D, Momeni E, Alavi Nezhad Khalil Abad SV (2015) Prediction of the unconfined compressive strength of soft rocks: a PSO-based ANN approach. Bull Eng Geol Environ 74:745–57
33. Sulukcu S, Ulusay R (2001) Evaluation of the block punch index test with particular reference to the size effect, failure mechanism and its effectiveness in predicting rock strength. Int J Rock Mech Min Sci 38:1091–111. https://doi.org/10.1016/S1365-1609(01)00079-X
34. Kohno M, Maeda H (2012) Relationship between point load strength index and uniaxial compressive strength of hydrothermally altered soft rocks. Int J Rock Mech Min Sci 50:147–157
35. Wu X, Kumar V, Ross Quinlan J, Ghosh J, Yang Q, Motoda H et al (2008) Top 10 algorithms in data mining. Knowl Inf Syst 14:1–37
36. Akbulut Y, Sengur A, Guo Y, Smarandache F (2017) NS-k-NN: Neutrosophic set-based k-nearest neighbors classifier. Symmetry (Basel) 9:179
37. Qian Y, Zhou W, Yan J, Li W, Han L (2014) Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. Remote Sens 7:153–168
38. Braik M, Ryalat MH, Al-Zoubi H (2022) A novel meta-heuristic algorithm for solving numerical optimization problems: Ali Baba and the forty thieves. Neural Comput Appl 34:409–455. https://doi.org/10.1007/s00521-021-06392-x
39. Sharma P, Thangavel S, Raju S, Prusty BR (2022) Parameter estimation of solar PV using Ali Baba and forty thieves optimization technique. Math Probl Eng 2022:5013146. https://doi.org/10.1155/2022/5013146
40. Zhao W, Zhang Z, Wang L (2020) Manta ray foraging optimization: an effective bio-inspired optimizer for engineering applications. Eng Appl Artif Intell 87:103300
41. Liao Y, Zhao W, Wang L (2021) Improved manta ray foraging optimization for parameters identification of magnetorheological dampers. Mathematics 9:2230

42. Mantegna RN (1994) Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes. Phys Rev E 49:4677–4683. https://doi.org/10.1103/PhysRevE.49.4677
43. Ling S-H, Iu HHC, Chan KY, Lam H-K, Yeung BCW, Leung FH (2008) Hybrid particle swarm optimization with wavelet mutation and its industrial applications. IEEE Trans Syst Man, Cybern Part B 38:743–763

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.