Developing an XGBoost Regression Model for Predicting Young’s Modulus of Intact Sedimentary Rocks for the Stability of Surface and Subsurface Structures

Niaz Muhammad Shahani1,2, Xigui Zheng1,2,3,4*, Cancan Liu1,2, Fawad Ul Hassan1,5 and Peng Li1,2

*Correspondence:
Xigui Zheng
cumt_ckzxg@126.com

OPEN ACCESS

Edited by:
Zetian Zhang,
Sichuan University, China

Reviewed by:
Harsha Vardhan,
National Institute of Technology, India
Yingchun Li,
Dalian University of Technology, China

Specialty section:
This article was submitted to Geohazards and Georisks, a section of the journal Frontiers in Earth Science

Received: 20 August 2021
Accepted: 12 October 2021
Published: 26 October 2021

Citation:
Shahani NM, Zheng X, Liu C, Hassan FU and Li P (2021) Developing an XGBoost Regression Model for Predicting Young’s Modulus of Intact Sedimentary Rocks for the Stability of Surface and Subsurface Structures. Front. Earth Sci. 9:761990. doi: 10.3389/feart.2021.761990

Young’s modulus (E) is essential for predicting the behavior of materials under stress and plays an important role in the stability of surface and subsurface structures. E has a wide range of applications in mining, geology, civil engineering, etc.; for example, coal and metal mines, tunnels, foundations, slopes, bridges, buildings, drilling, etc. This study developed a novel machine learning regression model, namely an extreme gradient boosting (XGBoost) to predict the influences of four inputs such as uniaxial compressive strength in MPa; density in g/cm³; p-wave velocity (Vp) in m/s; and s-wave velocity in m/s on two outputs, namely static Young’s modulus (Es) in GPa; and dynamic Young’s modulus (Ed) in GPa. Using a series of basic statistical analysis tools, the accompanying strengths of each input and each output were systematically examined to classify the most prevailing and significant input parameters. Then, two other models i.e., multiple linear regression (MLR) and artificial neural network (ANN) were employed to predict Es and Ed. Next, multiple linear regression and ANN were compared with XGBoost. The original dataset was allocated as 70% for the training stage and 30% for the testing stage for each model. To improve the performance of the developed models, an iterative 10-fold cross-validation method was used. Therefore, based on the results XGBoost model has revealed the best performance with high accuracy (Es: correlation coefficient (R²) = 0.998; Ed: R² = 0.999 in the training stage; Es: R² = 0.997; Ed: R² = 0.999 in the testing stage), root mean square error (RMSE) (Es: RMSE = 0.0652; Ed: RMSE = 0.0062 in the training stage; Ed: RMSE = 0.071; Ed: RMSE = 0.027 in the testing stage), RMSE-standard deviation ratio (RSR) index value (Es: RSR = 0.00238; Ed: RSR = 0.00023 in the training stage; Es: RSR = 0.00304; Ed: RSR = 0.001 in the testing stage) and variance accounts for (VAF) (Es: VAF = 99.71; Es: VAF = 99.99 in the training stage; Es: VAF = 99.83; Ed: VAF = 99.94 in the testing stage) compared to the other developed models in this study. Using a novel machine learning approach, this study was able to deliver substitute elucidations for predicting Es and Ed parameters with suitable accuracy and runtime.

Keywords: dynamic Young’s modulus, k-fold crosses validation, machine learning, predictive modeling, static Young’s modulus, XGBoost
INTRODUCTION

Young's modulus (E) is important for predicting the behavior of materials under load and plays a key part in the stability of surface and subsurface structures. E has a broad application in mining, geology, civil engineering, etc., i.e., coal and metal mines, tunnels, foundations, slopes, bridges, buildings, drilling, etc. Computation of accurate rock deformation properties, especially E is essential to the design of any rock engineering or rock mechanics project. Several researchers have studied the deformation and behavior of various types of rocks (Zhao et al., 2017; Rahimi and Nygaard, 2018; Davarpanah et al., 2019; Xiong et al., 2019). Generally, there are two common techniques, static and dynamic, employed to measure E. Static Young's modulus (E_s) is generally acquired as the digression of the stress-strain curve at 50% of the maximum strength of the rock core sample. The dynamic Young's modulus (E_d) can be determined if the density of the rock along with the velocities of compressional and shear waves is known. In rock engineering, the variation between E_s and E_d has been broadly investigated (Brotons et al., 2016). Normally, the value of E_d is slightly greater than the E_s studied by various researchers (Zhang, 2006; Kolesnikov, 2009). The ratio between E_d and E_s was calculated to range between 1 and 20 (Wang, 2000).

Typically, there are two common techniques, such as destructive and non-destructive, to estimate the strength and deformation of rocks. According to the recommended standards of the International Society of Rock Mechanics (ISRM) and the American Society for Testing Materials (ASTM), the use of

| Serial No | UCS (MPa) | Density (g/cm³) | Vp (m/s) | Vs (m/s) | E_s (GPa) | E_d (GPa) |
|-----------|-----------|----------------|----------|----------|-----------|-----------|
| 1         | 11.37     | 2.22           | 2,500    | 1,500    | 3.24      | 12.18     |
| 2         | 11.31     | 2.31           | 2,351    | 1,293    | 3.41      | 9.91      |
| 3         | 23.87     | 2.31           | 2,338    | 1,241    | 4.19      | 9.28      |
| 4         | 20.16     | 2.35           | 2,515    | 1,302    | 5.04      | 13.29     |
| 5         | 48.72     | 2.35           | 2,163    | 1,297    | 5.28      | 9.92      |
| ...       | ...       | ...            | ...      | ...      | ...       | ...       |
| 60        | 81.24     | 2.65           | 2,975    | 1,935    | 17.13     | 22.49     |
| 61        | 97.39     | 2.69           | 5,195    | 2,953    | 55.68     | 60.96     |
| 62        | 91.4      | 2.68           | 5,534    | 3,012    | 36.05     | 62.71     |
| 63        | 46.51     | 2.64           | 3,811    | 2,122    | 17.31     | 30.32     |
| 64        | 40.68     | 2.73           | 1,838    | 951      | 13.74     | 6.5       |
| Mean      | 56.19     | 2.51           | 4,005.06 | 2,173.33 | 29.35     | 36.73     |
| Min       | 10.24     | 2.04           | 1,626    | 900      | 0.77      | 4.98      |
| Max       | 143.09    | 2.92           | 6,539    | 3,420    | 90.49     | 83.89     |
| Std. D    | 32.78     | 0.21           | 1,599.18 | 840.15   | 27.35     | 27.63     |

FIGURE 1 | Pairwise relationship of input parameters and outputs (A) E_s and (B) E_d in the dataset.
destructive testing in the laboratory to directly estimate $E$ is complex, time-consuming and expensive process. At the same time, sample preparation is quite difficult in the case of fragile, internally damaged, thin and highly foliated rocks (Jing et al., 2020). Thus, attention must be paid to the indirect evaluation of $E$ through the use of rock index tests. Many researchers have established prediction models to overcome these shortcomings by employing soft computing methods such as artificial neural network (ANN), multiple regression analysis (MRA) and other novel machine learning approaches (Lindquist et al., 1994; Singh and Dubey, 2000; Tiryaki, 2008; OzcelikBayram et al., 2013; Abdi et al., 2018; Teymen and Mengüç, 2020; Cao et al., 2021; Yang et al., 2020; Duan et al., 2020). Waqas et al. used linear and nonlinear regression, regularization and ANFIS (using neuro-fuzzy inference system) to predict the $E_d$ of sedimentary rocks (Waqas and Ahmed, 2020). Abidi et al. proposed the ANN and MRA

FIGURE 2 | Correlation plot of input parameters and outputs (A) $E_s$ and (B) $E_d$ in the dataset.

FIGURE 3 | Flow chart of the study.
The proposed approach is suitable for predicting E. The algorithm (FA) to predict E. The results showed that the gradient boosting (XGBoost) combined with the approach of supervised machine learning, namely an extreme novel study, XGBoost is developed for predicting Es and Ed using corresponding data of tests performed in the laboratory. In this approaches plays a key role in determining the accuracy of the predictive methods, the prediction of E with machine learning model of E of rocks. Due to limitations of the conventional used, especially in combination with ANN and MLR, for predictive knowledge, XGBoost machine learning method has rarely been estimated Es (Elkatatny et al., 2019). Cao et al. employed the novel SVM. The established correlations improved the accuracy of the prediction results from different models such as ANN, ANFIS and results (Elkatatny, 2021). Elkatatny et al. was from the drilling parameters. The study showed encouraging events in the testing phase. The developed ANN model predicted network with 409 data events in the training phase and 183 data events in the testing phase. The developed ANN model predicted Es with a high correlation coefficient (R² = 0.999) and minimum mean absolute percentage error (AAPE = 0.998%) (Mahmoud et al., 2019). Elkatatny developed an ANN network for predicting Ed from the drilling parameters. The study showed encouraging results (Elkatatny, 2021). Elkatatny et al. was first to correlate E, prediction results from different models such as ANN, ANFIS and SVM. The established correlations improved the accuracy of the estimated E (Elkatatny et al., 2019). Cao et al. employed the novel approach of supervised machine learning, namely an extreme gradient boosting (XGBoost) combined with the firefly algorithm (FA) to predict E. The results showed that the proposed approach is suitable for predicting E.

Based on the above literature and to the best of author knowledge, XGBoost machine learning method has rarely been used, especially in combination with ANN and MLR, for predictive modeling of E of rocks. Due to limitations of the conventional predictive methods, the prediction of E with machine learning approaches plays a key role in determining the accuracy of the corresponding data of tests performed in the laboratory. In this novel study, XGBoost is developed for predicting E, and E using four input parameters, i.e., uniaxial compressive strength (UCS) in MPa; density in g/cm³; P-wave velocity (Vp) in m/s; and s-wave velocity (Vs) in m/s, complimented with ANN and MLR. Then, the original dataset of 64 data points is split as 70% for the training stage and 30% for the testing stage. To improve the performance of the machine learning model, an iterative 10-fold cross-validation method is used.

MATERIALS AND METHODS

Construction of a Dataset

Several multivariate parameters of intact sedimentary rocks (marlstone, sandstone, and limestone) are already reported (Moradian and Behnia, 2009) to have been used as inputs to predict the static Young’s modulus (Es) and dynamic Young’s modulus (Ed), which include uniaxial compressive strength (UCS) in MPa; density in g/cm³; P-wave velocity (Vp) in m/s; and s-wave velocity (Vs) in m/s. There were a total of 64 events with no missing data. Table 1 shows the original dataset and statistical distribution in this study.

In this study, to visualize the original dataset of E, the seaborne module in Python was used. Figures 1A,B illustrate a pairwise scatter of the kernel density estimation (KDE). The purpose of building a KDE pairwise plot is to examine the association between any two influencing parameters in the original dataset. Based on Figures 1A,B, all the input parameters have a moderate to strong positive correlation with both Es and Ed. Next, Figures 2A,B highlight the diagonal correlations between the input and output parameters. The seaborne module in Python was used for diagonal correlation heatmaps to develop the correlation coefficients of multiple inputs with E, and Ed. Correlation coefficient values are specified in the light red to dark red color for Es and light purple to dark purple for Ed. According to Figures 2A,B, the overall correlation coefficients between the input and output parameters are relatively high. Therefore, all parameters were incorporated to improve the accuracy of the final probabilistic prediction framework in the Es and Ed circumstance. Figure 3 depicts the flow chart of the study.

Methods

Multiple Regression Analysis

Multiple regression analysis (MRA) can be classified into linear and nonlinear regression. However, this study has implemented the multiple linear regression (MLR) as a result of multiple variables using SPSS (version 23). MLR is a numerical method that uses multiple descriptive parameters to estimate the output of a reporting parameter. The MLR method is used to obtain the best-fit relationship between the parameters. Generally, MLR can be employed to establish the association between independent (input) and dependent (output) parameters. In this study, the MLR technique was used to predict Es and Ed, respectively. The MLR relationship between the inputs and output can usually be expressed by Eq. 1.

\[ D = a + b_1X_1 + b_2X_2 + b_3X_3 + \ldots + b_nX_n \]  

(1)
where, $D$ depicts the output parameter, $a$ denotes the regression constant, $B_1 - B_n$ are the coefficients of regression and $X_1 - X_n$ are the input variables.

Based on the consequences of MLR, $E_s$ and $E_d$ are predicted by the established linear expressions as shown in Eqs 2, 3.

$E_s = 0.086UCS + 26.55\text{Density} + 0.014V_p - 0.004V_s - 89.47$  

$E_d = 14.568\text{Density} + 0.008V_p - 0.15V_s - 0.034UCS - 64$  \(2\)  \(3\)

where, $E_s$ and $E_d$ are static and dynamic Young’s modulus in GPa, respectively. UCS represents uniaxial compressive strength in MPa, $V_p$ and $V_s$ are the p-wave and s-wave velocities in m/s, respectively.

Artificial Neural Network

Artificial neural network (ANN) is among many supervised machine learning methods and has found wide application in a variety of fields. An ANN consists of three components, i.e., input layers, hidden layers, and an output layer. The structure of ANN and the choice of hidden layers and neurons play a crucial part in ascertaining its performance (Chester, 1990). The feedforward back-propagation (FFBP) neural network, a multilayer perceptron network, was used owing to its simple process and wide applicability. Back-propagation (BP) is one of the most efficient and commonly employed learning algorithms in multi-layer networks (Cevik et al., 2011; Hajihassani et al., 2014). Each network must contain sufficient neurons depending on the application of ANN. Neurons of a given layer are connected to the neurons of the subsequent consecutive layer with every connection having a certain weightage (Atkinson and Tatnall, 1997).

Equation 4 is employed to estimate the approximate number of neurons in the hidden layer ($N_h$), since the inappropriate selection of the neurons in the hidden layer often leads to “under-fitting” and “over-fitting” and must be prevented. Figure 4 represents the basic structure of the ANN network for predicting $E_s$ and $E_d$ in this study.

$N_h \leq 2N_1 + 1$  \(4\)

where, $N_1$ denote the total number of inputs.

In order to build the net input $n$, the weighted input $\omega p_i$ is connected to a scalar bias $b$, $f$ denotes the transfer function, and $O$ denotes the scalar output. If the neuron has $Z$ input parameters, the output can be computed by Eq. 5.

$O = f \sum_{i=1}^{Z} (\omega p_i + b)$  \(5\)
This study used a sigmoid transfer function in the hidden layer and a linear output function in the output layer. To achieve the number of neurons in the hidden layer, this study used some provisions, since there is no specific approach to providing the desired results. In addition, fifty epochs were used for training the ANN network and the least error of validation is considered as a stop to avoid overfitting.

**Extreme Gradient Boosting**

The extreme gradient boosting (XGBoost) algorithm was created by Chen and Guestrin (Chen and Guestrin, 2016). Being an effective tree-based ensemble learning algorithm, it is considered a powerful tool among data science researchers. XGBoost is based on gradient boosting architecture (Friedman, 2001), which uses various complement functions to estimate the results using Eq. 6.

\[
\overline{y_i} = y_i^0 + \eta \sum_{k=1}^n f_k (U_i)
\]  

(6)

where, \(\overline{y_i}\) indicates the predicted output for \(i\)th data with the parameter vector \(U_i\); \(n\) denotes the number of estimators corresponding to independent tree structures for each \(f_k\) (i.e. \(k = 1\) to \(n\)); and \(y_i^0\) displays the primary hypothesis, which is actually the mean of the original parameters in the training data. \(\eta\) represents the rate of learning connected to improving the performance of the model, whether connecting the additional trees to prevent overfitting. The statistical model has to be developed with less overfitting which is one of the genuine problems that often conflicts in machine learning. In the XGBoost model, the training phase is determined in a complementary way.

According to Eq. 3 in the \(k\)th stage, the \(k\)th estimator is connected to the model and the prediction of the \(k\)th \(\overline{y_i}^{(k-1)}\) is calculated from the estimated output \(y_i^{(k-1)}\) in the next step, and the established \(f_k\) of the \(k\)th complementary estimator is shown in Eq. 7.

\[
y_i^{(k)} = y_i^{(k-1)} + \eta f_k
\]

(7)

whereas \(f_k\) represents the leaves weight that is established by reducing the objective function of the \(k\)th tree and is given by Eq. 8.

\[
f_{vk} = y Z + \sum_{a=1}^{Z} \left[ g_a \omega_a + \frac{1}{2} (h_a + \lambda) \omega_a^2 \right]
\]

(8)

where, \(Z\) denotes the quantity of leaf nodes, \(y\) denotes the complexity parameter, \(\lambda\) denotes constant coefficient, and \(\omega_a^2\)

| Performance ranking | Poor | Good | Very good | Best |
|---------------------|------|------|-----------|------|
| RSR index value     | >0.7 | 0.6  | 0.5       | 0.00 |

**TABLE 2 | Performance ranking and the corresponding RSR index values.**
denotes the leaf weight from 1 to Z. \( y \) and \( \lambda \) are regularization parameters employed to improve the model to keep away from the over-fitting. \( g_a \) and \( h_a \) are the summation parameters for the entire dataset associated with a leaf of the initial and previous loss function gradient, correspondingly. In order to build the \( k \)th tree, a leaf is distributed into several leaves. Such a system is implied by using the gain parameters expressed in Eq. 9.

\[
G = \frac{1}{2} \left[ \frac{O_L^2}{P_L + \lambda} + \frac{O_R^2}{P_R + \lambda} + \frac{(O_L + O_R)^2}{P_L + P_R + \lambda} \right] \tag{9}
\]

where, \( G \) denotes the gain parameters, \( O_R \) and \( P_R \) denote the right leaf, respectively. \( O_L \) and \( P_L \) denote the subsequent division of the left leaf, respectively. When the gain parameter is approximated to zero, the division criteria are generally assumed. \( y \) and \( \lambda \) are regularization parameters that are indirectly reliant on the gain parameters. For example, a larger regularization parameter can significantly reduce the gain parameter, thus avoiding the leaf convolution phenomenon. However, this will reduce the performance of the model to adapt to the training data. Figure 5A demonstrates the basic structure of the level-wise XGBoost tree model.

**Grid Search Cross Validation**

The grid search method is used for the adjustment of hyperparameters (Bergstra and Bengio, 2012). The technique approves a search in an identified range of hyperparameters and defines the desired results leading to the best outcomes of the assessment criteria, i.e., \( R^2 \), MAE, MSE and RMSE. GridSearchCV() has been carried out in scikit-learn Python programming language to process this strategy. This method simply calculates the score of CV for all hyperparameters integrated with a particular reach. In this study, a 10-fold iterated arbitrary arrangement practice was incorporated in the CV command as specified in Figure 5B. GridSearchCV() allows not only to compute the desired hyperparameters, but also to evaluate the metric values to their desired outcomes. This study used all the remaining features of the Python programming language by default to perform Grid Search CV.

**Performance Criterion**

Typically, the performance of a model must be estimated when approaching the steadiness of a prediction framework, using an extensive range of performance criteria to select a highly accurate model. Therefore, this study proposes a unique performance criterion as follows:

**Correlation coefficient**

The correlation coefficient \( (R^2) \) is the key to the execution of the regression survey. The computation of \( R^2 \) can be expressed by Eq. 10.

\[
R^2 = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2(Y_i - \bar{Y})}} \tag{10}
\]
Mean Square Error

Mean square error (MSE) is the mean of the square of all errors and is one of the important metrics for evaluating the performance of the corresponding models. The computation of MSE can be expressed by Eq. 11.

\[
MSE = \frac{\sum_{i=1}^{n} (X_i - Y_i)^2}{n}
\]  

(11)

Root Mean Square Error

Root mean square error (RMSE) is the square root of the mean of the square of all errors and is measured as significant metric for mathematical predictions. The computation of RMSE can be expressed by Eq. 12.

\[
RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X_i - Y_i)^2}{n}}
\]  

(12)

Root Mean Square Error Standard Deviation Ratio

Root Mean Square Error Standard Deviation Ratio (RSR) is employed in this study for the comparison of significant models, which can be executed to predict \( E_s \) and \( E_d \). RSR plays an important role as a valuable metric for testing
analytical models. The computation of RSR can be expressed by Eq. 13.

$$RSR = \frac{\sqrt{\sum_{i=1}^{n} (X_i - Y_i)^2}}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{Y})^2}}$$

(13)

Variance Accounts For

Variance accounts for (VAF) is also considered as one of the important metrics for evaluating the overall performance of the model. Higher the VAF value, the greater will be the performance of the model. The computation of VAF can be expressed by Eq. 14.

$$VAF = \left[ 1 - \frac{\text{var} (X_i - Y_i)}{\text{var} (X_i)} \right] \times 100$$

(14)

where, $X$ and $Y$ are the mean of the original and predicted data values, respectively, $X_i$ and $Y_i$ are the original and predicted data values, respectively, and $n$ shows the number of datasets. $Y_i$ is the mean value of the original data. Table 2 signifies the performance ranking and the corresponding RSR values.
FIGURE 10 | Change of relative error between predicted and original data of proposed models in the (A) training stage and (B) testing stage for $E_s$ and $E_p$. 
RESULTS AND DISCUSSION

In this study, a novel machine learning regression XGBoost model was developed and compared with two other models, namely MLR and ANN, to confirm the accuracy of predicting $E_s$ and $E_d$. To avoid overfitting of these models, the original dataset was partitioned into 70% for the training stage and 30% for the testing stage of 64 events. The ANN and XGBoost models are trained on training data and then validated by testing data. The 50 epochs were used for training the ANN model and the least error of validation is considered as a stop to avoid overfitting.

According to Eq. 4, a total of nine neurons are selected in the hidden layer, which is connected to four input neurons and two output neurons, as shown in Figure 4. In this study, an XGBoost model with the default features of the XGBoost module was executed, i.e., $M = 50$ estimators, the regularization properties of $\gamma = 0$, $\lambda = 1$, and a learning rate of $\eta = 0.3$. Moreover, a 10-fold iterated arbitrary arrangement practice was incorporated to substantiate the models.

The original and predicted output values were then arranged and represented in scattered plots in order to ease the performance and correlation analysis of the developed models. The input parameters are UCS (MPa); Density (g/cm$^3$); $V_p$ (m/s); and $V_s$ (m/s). The predicted output parameters are $E_s$ and $E_d$. The final output was evaluated by using performance criteria such as $R^2$, RMSE and VAF, and the developed models were compared to estimate the appropriate model with higher accuracy of prediction results in this study.

Figures 6A,B to Figures 7A,B depict the scatter plots of the predictions of the proposed models (a) MLR, (b) ANN and (c) XGBoost for $E_s$ and $E_d$ versus the original data in the training and testing stages, respectively. The prediction performance accuracy of the proposed models is (a) MRA ($E_s$: $R^2 = 0.928$; $E_d$: $R^2 = 0.981$ in the training stage, and; $E_s$: $R^2 = 0.717$; $E_d$: $R^2 = 0.985$ in the testing stage), (b) ANN ($E_s$: $R^2 = 0.958$; $E_d$: $R^2 = 0.998$ in the training stage; and $E_s$: $R^2 = 0.849$; $E_d$: $R^2 = 0.998$ in the testing stage) and (c) XGBoost ($E_s$: $R^2 = 0.998$; $E_d$: $R^2 = 0.999$ in the training stage, and; $E_s$: $R^2 = 0.997$; $E_d$: $R^2 = 0.999$ in the testing stage).

Simultaneously, to comprehend the good visualization of the predicted values aggregated with the original data of $E_s$ and $E_d$, Figures 8A,B demonstrate the performance of MLR, ANN and XGBoost models in the training stage, respectively. Figures 9A,B demonstrate the performance of MLR, ANN and XGBoost models in the testing stage for (a) $E_s$ and (b) $E_d$, respectively.

Figures 10A,B demonstrates the variation of the relative error of the proposed models, i.e., MLR, ANN, and XGBoost, for the prediction of $E_s$ and $E_d$ and the original data in (a) training stage.

### TABLE 3 | Performance criterion of the MRA, ANN, and XGBoost.

| Model  | Training | Testing |
|--------|----------|---------|
|        | $R^2$    | RMSE    | RSR    | VAF | $R^2$    | RMSE    | RSR    | VAF |
| MRA    | $E_s$    | 0.928   | 0.3080 | 0.0114 | 98.68 | 0.717   | 1.6176  | 0.06894 | 96.23 |
|        | $E_d$    | 0.981   | 0.0117 | 0.00044 | 99.96 | 0.985   | 0.2815  | 0.01038 | 99.43 |
| ANN    | $E_s$    | 0.958   | 1.0044 | 0.03680 | 95.67 | 0.849   | 0.5274  | 0.02248 | 98.77 |
|        | $E_d$    | 0.998   | 0.1723 | 0.00651 | 99.45 | 0.998   | 0.1270  | 0.00468 | 99.74 |
| XGBoost| $E_s$    | 0.998   | 0.0652 | 0.00238 | 99.71 | 0.997   | 0.0711  | 0.00304 | 99.83 |
|        | $E_d$    | 0.999   | 0.0062 | 0.00023 | 99.99 | 0.999   | 0.0274  | 0.001   | 99.94 |

**FIGURE 11** | The comparative Taylor diagrams of the proposed models MRA, ANN, and XGBoost in the (A) training stage and (B) testing stage for $E_s$ and $E_d$. 
and (b) testing stage, respectively. The prediction performance of the proposed models for variation in relative mean square error (MSE) is (a) MLR ($E_s$: MSE = 0.095; $E_d$: MSE = 0.00014 in the training stage, and; $E_s$: MSE = 2.617; $E_d$: MSE = 0.079 in the testing stage), (b) ANN ($E_s$: MSE = 1.009; $E_d$: MSE = 0.030 in the training stage, and; $E_s$: MSE = 0.278; $E_d$: MSE = 0.079 in the testing stage) and (c) XGBoost ($E_s$: MSE = 0.0043; $E_d$: MSE = 0.00003 in the training stage, and; $E_s$: MSE = 0.0051; $E_d$: MSE = 0.0007 in the testing stage).

Table 3 shows the performance criterion of the MRA, ANN and XGBoost determined using Eq. 10–14. Figure 11 depicts the comparative Taylor diagrams of the proposed models MRA, ANN and XGBoost in the (a) training stage and (b) testing stage for $E_s$ and $E_d$ to further estimate the performance of the models more extensively. The standard deviation values associated with one another by the circular lines are shown as horizontal and vertical coordinates in the diagrams. Two performance metrics, one is the $R^2$ value, indicated by the blue radial lines from the starting of the coordinates, and the other is the RMSE value, specified by the green circular line. Observed values were used as the base model with zero errors, i.e., RMSE = 0 in the diagrams, the maximum $R^2$ = 1, and the computed standard deviations. Next, the $R^2$, RMSE and standard deviation of the other models were compared with the observed values. A best model is one with highest degree of similarity to observed data model. As shown in Figure 11, the XGBoost model has been able to approach the observed data and outperformed the MRA and ANN models in both the training and testing phases.

Moreover, according to Table 3 and the results in Figure 11, the XGBoost model has revealed the best performance with high accuracy ($E_s$: $R^2 = 0.998$; $E_d$: $R^2 = 0.999$ in the training stage, and; $E_s$: $R^2 = 0.997$; $E_d$: $R^2 = 0.999$ in the testing stage), RMSE ($E_s$: RMSE = 0.0652; $E_d$: RMSE = 0.0062 in the training stage, and; $E_s$: RMSE = 0.071; $E_d$: RMSE = 0.027 in the testing stage), RSR index value ($E_s$: RSR = 0.00238; $E_d$: RSR = 0.00023 in the training stage, and; $E_s$: RSR = 0.00304; $E_d$: RSR = 0.001 in the testing stage) and VAF ($E_s$: VAF = 99.71; $E_d$: VAF = 99.99 in the training stage, and; $E_s$: VAF = 99.83; $E_d$: VAF = 99.94 in the testing stage) compared to the other developed models in this study.

Therefore, XGBoost is an applicable machine learning regression model that can be applied to accurately predict the $E_s$ and $E_d$.

**CONCLUSION**

Young’s modulus (E) plays an important role in the stability of surface and subsurface structures. Therefore, an accurate estimation of E is mandatory. This study developed a novel machine learning XGBoost regression model with four input parameters, i.e., UCS (MPa), density (g/cm³), $V_p$ (m/s) and $V_s$ (m/s) for predicting $E_s$ (GPa) and $E_d$ (GPa). In addition, the MRA and ANN models were included to compare their results with the proposed model. To avoid overfitting of these models, the original dataset was partitioned into 70% for the training stage and 30% for the testing stage of 64 data points. The study concludes that the proposed XGBoost regression model performed more accurately than the other studied models in predicting $E_s$ and $E_d$. Employing a novel machine learning approach, this study was able to provide substitute elucidations to predict $E_s$ and $E_d$ parameters with appropriate accuracy and runtime. Future work can be extended using various datasets to further confirm the reliability of the proposed model.

**DATA AVAILABILITY STATEMENT**

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

**AUTHOR CONTRIBUTIONS**

NS: Conceptualization, Methodology, Writing Original Draft, and Run Code. XZ: Project Supervisor, Funding, and Review and Editing. CL: Run Code and Visualization. FH: Review and Editing. PL: Data Curation and Run Code.

**FUNDING**

This research was supported by the Science and Technology Innovation Project of Guizhou Province (Qiankehe Platform Talent (2019) 5620 to XZ). No additional external funding was received for this study.

**REFERENCES**

Abdi, Y., Garavand, A. T., and Sahamieh, R. Z. (2018). Prediction of Strength Parameters of Sedimentary Rocks Using Artificial Neural Networks and Regression Analysis. *Austral J. Geosci.* 11 (19), 1–11. doi:10.1007/s12517-018-3929-0

Aboutaleb, S., Behnia, M., Bagherpour, R., and Bluekian, B. (2018). Using Non-destructive Tests for Estimating Uniaxial Compressive Strength and Static Young’s Modulus of Carbonate Rocks via Some Modeling Techniques. *Build. Eng. Geol. Environ.* 77 (4), 1717–1728. doi:10.1016/s1006-0417-015-003

Atkinson, P. M., and Tatnall, A. R. L. (1997). Introduction Neural Networks in Remote Sensing. *Int. J. Remote Sensing* 18 (4), 699–709. doi:10.1080/014311697218700

Bergstra, J., and Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *J. Machine Learn. Res.* 13 (2), 281–305. doi:10.1016/j.chemolab.2011.12.002

Brotons, V., Tomás, R., Iborra, S., Greidiaga, A., Martínez-Martínez, I., Benavente, D., et al. (2016). Improved Correlation between the Static and Dynamic Elastic Modulus of Different Types of Rocks. *Mater. Struct.* 49 (8), 3021–3037. doi:10.1617/s11527-015-0702-7

Cao, J., Gao, J., Rad, H. N., Mohammed, A. S., Hasanipanah, M., and Zhou, J. (2021). A Novel Systematic and Evolved Approach Based on XGBoost-Firefly Algorithm to Predict Young’s Modulus and Unconfined Compressive Strength of Rock. *Eng. Comput.* 10.1007/s10660-020-01241-2 Published online Gevik, A., Seier, E. A., Cabalar, A. F., and Gokceoglu, C. (2011). Modeling of the Uniaxial Compressive Strength of Some clay-bearing Rocks Using Neural Network. *Appl. Soft Comput.* 11 (2), 2587–2594. doi:10.1016/j.asoc.2010.08.008

Chen, T., and Guestrin, C. (2016). ”Xgboost: A Scalable Tree Boosting System,” in Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, August 13–17, 2016, 785–794.
