Shahrood University of Technology

Journal of Mining and Environment (JME)

Journal homepage: www.jme.shahroodut.ac.ir

Iranian Society of Mining Engineering (IRSME)

A Hybrid Model for Back-Break Prediction using XGBoost Machine learning and Metaheuristic Algorithms in Chadormalu Iron Mine

Zohreh Nabavi[1], Mohammad Mirzehi[1], Hesam Dehghani[2], Pedram Ashtari[2*]

1. Department of Mining Engineering, Facullty of Engineering, Tarbiat Modares University, Tehran, Iran
2. Department of Mining Engineering, Hamedan University of Technology, Hamedan, Iran

Article Info

Received 9 March 2023
Received in Revised form 18 April 2023
Accepted 14 May 2023
Published online 14 May 2023

DOI:10.22044/jme.2023.12796.2323

Keywords

Back-break
Extreme gradient boosting
Particle swarm optimization (PSO)
Gray wolf optimization (GWO)
Chadormalu iron mine

Abstract

Back-break is one of the adverse effects of blasting, which results in unstable mine walls, high duration, falling machinery, and inappropriate fragmentation. Thus, the economic benefits of the mine are reduced, and safety is severely affected. Back-break can be influenced by various parameters such as rock mass properties, blast geometry, and explosive properties. Therefore, during the blasting process, back-break must be accurately predicted, and other production activities must be done to prevent and reduce its adverse effects. In this regard, a hybrid model of extreme gradient boosting (XGB) is proposed for predicting back-break using gray wolf optimization (GWO) and particle swarm optimization (PSO). Additionally, validation of the hybrid model is conducted using XGBoost, gene expression programming (GEP), random forest (RF), linear multiple regression (LMR), and non-linear multiple regression (NLMR) methods. For this purpose, the data obtained from 90 blasting operations in the Chadormalu iron ore mine are collected by considering the parameters of the blast pattern design. According to the results obtained, the performance and accuracy level of hybrid models including GWO-XGB (R2 = 99, RMSE = 0.01, MAE = 0.001, VAF = 0.99, a-20 = 0.98), and PSO-XGB (99, 0.01, 0.001, 0.99, 0.98) are better than the XGBoost (97, 0.185, 0.132, 0.98, 95), GEP (96, 0.233, 0.186, 0.967, 0.935), RF (97, 0.210, 0.156, 0.97, 0.94), LMR (96, 0.235, 0.181, 0.964, 0.92), and NLMR (96, 0.229, 0.177, 0.968, 0.93) models. Notably, the GWO-XGB hybrid model has superior overall performance as compared to the PSO-XGB model. Based on the sensitivity analysis results, hole depth and stemming are the essential effective parameters for back-break.

Highlights:

- Prediction of back-break using the hybrid XGBoost
- Comparing the real back-break with the results of more than six AI models
- Investigating the effects of the drilling pattern parameters on the back-break

1. Introduction

Drilling and blasting are cost-effective techniques for removing overburden and extraction in open-pit mines [1]. However, with the current blasting techniques, only a tiny amount (20-30%) of the blast energy is used for fragmentation. Most (70-80%) of the blast energy is lost with varying degrees of adverse consequences such as back-break, fly rock, ground vibration, and air blast [2-4]. The leading cause of energy loss in blasting operations is the mismatch and coordination between the design parameters of the blast pattern, the properties of explosives, and rock mass [5]. Among the undesirable consequences of blasting, back-break is considered a fundamental challenge primarily caused by improper design [2]. The term back-break refers to the phenomenon of breaking rocks beyond the blast zone and the last row. This can occur due to a variety of reasons such as improper fragmentation, instability of high walls, falling machinery, high duration, lower efficiency, and safety.

✉ *Corresponding author: p.ashtari@gmail.com (p. Ashtari)*

Back-break control and minimization is one of the ongoing concerns of designers, planners, and environmental specialists [6]. The review of the literature indicates that the researchers have proposed several parameters to happening back-break and its remedial actions [7]. According to Mohammadnejad *et al*., controllable and uncontrollable parameters are the two well-known parameters that contribute to back-break results during blast operations [8]. In this way, the uncontrollable parameters are related to a rock mass's physical and mechanical properties. On the other hand, controllable parameters are related to blast design and explosive parameters [9]. Also in other studies, the parameters affecting the back-break have been divided into three categories: geometric parameters of explosion model design, explosives properties, and rock mass characteristics and discontinuities [5, 10].

Konya and Walter described some causes of back-break such as high burden, stiff benches, long stemming length, and wrong time delay [11]. Monjezi *et al*. stated that the hole depth, spacing, burden, and stemming length are the most critical parameters that affect the spread of the back-break [12]. Gate *et al*. believed that a combination of factors in blasting such as hyper-stemming and short time delays in the blasting sequence may lead to a severe back-break [7]. Roy *et al*. and Singh *et al*. stated that inappropriate lag distance between rows and longer stemming length leads to back-break in stiff benches [13, 14]. The reason for this phenomenon is the excessive trapping of gases in the last row of holes due to unacceptable delay.

As mentioned, the properties of explosives are among the influencing parameters that can be controlled on back-break. Enayatollahi and Aghajani used salt and ANFO to reduce back-break and controlled explosion [15]. The coupling ratio is a crucial parameter affecting back-break severity. This parameter shows the amount of direct contact of the explosive material with the walls of the blast hole. For this purpose, Iverson *et al*. evaluated the blast damage caused by fully coupled explosive powder. They showed that reducing the coupling ratio can reduce the back-break [16].

It is challenging to accurately and quickly predict and evaluate the back-break based on various effective parameters. Considering the diversity of effective parameters and the complexity of their interactions, using new techniques such as artificial intelligence (AI) may help solve this problem. Therefore, various studies have been conducted to predict the back-break using artificial neural networks, genetic programming, fuzzy set theory, random decision trees, support vector machines, and random forest (see Table 1). Some recent work with soft computing techniques for back-break prediction is shown in Table 1. On the other hand, many studies have been done to predict the consequences of blasting using XGBoost and GEP methods [17-22]. According to the above and Table 1, the XGBoost and GEP methods were not used to predict the back-break. On the other hand, GEP has the advantage of providing a precise relationship for prediction compared to other methods.

The study on the prediction of back-break is significant in the field of mine blasting as it can help in reducing the associated risks and costs. Back-break is a common issue in blasting operations that can lead to safety hazards, damage to equipment, and delays in production. Accurate prediction of back-break can help in optimizing blasting parameters such as hole size, explosive type, and blast design, ultimately leading to better fragmentation and reduced back-break. In addition, the use of predictive models can aid in minimizing the need for trial-and-error methods, reducing operational costs, and increasing productivity. This study proposes a hybrid model that combines the strengths of both artificial neural networks and decision trees to accurately predict back-break. The significance of this study lies in its contribution to the development of more effective and efficient blasting practices, ultimately leading to safer and more profitable mining operations.

Therefore, this study aims to develop a hybrid model for predicting back-break in open-pit blasting. The proposed model combines the power of XGBoost with the optimization capabilities of PSO and GWO algorithms. To assess the performance of the hybrid model, it will be compared with other established models including XGBoost (optimized with grid search), GEP, LMR, RF, and NLMR. This study is expected to contribute to the development of accurate and efficient models for predicting back-break, which can ultimately lead to improved safety and productivity in mining operations.

Table 1. Some recent works with soft computing techniques for back-break prediction.

Research	Input	Method	R^2
Monjezi *et al*. [23]	B, S, HD, ST, CPD, PF, SD	Fuzzy set theory	R^2 = 0.95
Monjezi *et al*. [24]	UCS, SD, W, B, S, ST, HD, BH, PF, CPD	ANN	R^2 = 0.90
Ghasemi [25]	B, S, ST, PF, K	PSO	R^2 = 0.98
Saghatforoush *et al*. [26]	B, S, ST, PF, HL	ANN	R^2 = 0.83
Ghasemi *et al*. [27]	B, S, ST, PF, K	RT, ANFIS	R^2_{RT} = 0.972 R^2_{ANFIS} = 0.998
Shirani *et al*. [28]	B, S, ST, PF, K	GP	R^2 = 0.98
Hasanipanah *et al*. [29]	B, S, ST, PF	ANFIS-PSO	R^2 = 0.92
Sharma *et al*. [6]	B, S, ST, PF, K	RFA	R^2 = 0.88
Yu *et al*. [30]	PF, B, S/B, N, CPD, LCT, ST/B, JC, UCS, W/B	SVM–MFO	R^2 = 0.985
Sirjani *et al*. [5]	HL, B, S, PF, N, UCS	ANN	R^2 = 0.83
Kumar *et al*. [31]	S/B, H/St, D, W	RF	R^2 = 0.9791
Dai *et al*. [3]	HL, B, S, ST, SD, PF	PSO–RF	R^2 = 0.9961
Li *et al*. [32]	HL, B, S, ST, SD, PF	ELM ELM–PSO ELM–FOA ELM–WOA ELM–LOA ELM–SOA ELM–SSA	R^2_{ELM} = 0.9671 $R^2_{ELM-PSO}$ = 0.9978 $R^2_{ELM-FOA}$ = 0.9760 $R^2_{ELM-WOA}$ = 0.9964 $R^2_{ELM-LSO}$ = 0.9981 $R^2_{ELM-SOA}$ = 0.9949 $R^2_{ELM-SSA}$ = 0.9971

** Symbols are explained in the Abbreviation section.

2. Materials and Methodology

2.1. Field study and data collection

Chadormalu open-pit iron mine is located 120 km northeast of the Yazd city in the center of Iran. Figure 1 shows a view of the Chadormalu mine and its location in Iran relative to Tehran (the capital of Iran) and Yazd. The total amount of mineable ore reserves are about 400 million tons. According to mineralogical studies, magnetite, hematite, and apatite are the main components of the deposit. The primary explosive used in mine blasting is ANFO.

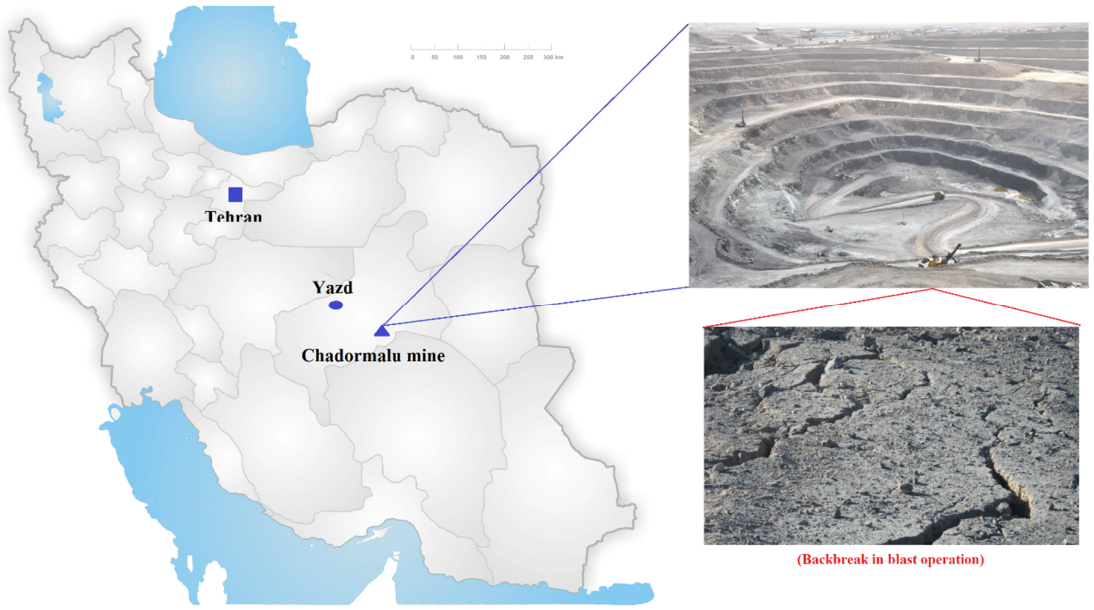


Figure 1. Location of the Chadormalu mine in Iran relative to Tehran (the capital) and Yazd city.

In this research work, the dataset used in the developed model consists of input and output parameters. The input parameters used for back-break prediction are given in Table 2. The input

parameters burden (B), spacing (S), stemming (ST), hole diameter (D), hole depth (H), specific charge (SC), and the number of rows (NR) have been applied as the output parameters of the models to predict back-break (BB). Table 2 shows the input and output parameters' details, statistical descriptions, and symbols.

Chadormalu iron ore mine was used as a case study to collect the input parameters in this research work. A database related to back-break prediction was formed based on 90 data pairs for model development. From the organized database, 20% of the dataset was selected to test the model to ensure consistency.

Table 2. Input and output data with details, statistical descriptions, and symbols.

Type of data	Variables	Symbol	Minimum	Maximum	Mean	Std. Deviation
Input	Burden	B	5	6	5.24	0.43
	Spacing	S	6	8	6.86	0.64
	Stemming	ST	1.37	6.68	3.92	1.36
	Hole diameter	D	6.5	7.5	6.76	0.44
	Hole depth	H	3.74	16.91	10.29	3.59
	Specific charge	SC	0.239	0.49	0.34	0.04
	Number of rows	NR	2	11	5.26	2.23
Output	Back-break	BB	2	7	4.59	1.2

2.2. Hybrid method

A machine learning model requires past experience parameters based on datasets. Many studies have shown heuristic algorithms improve machine learning accuracy and stability [33, 34]. Thus this study aims to predict back-break by combining the extreme gradient boosting (XGB) framework with optimization algorithms including gray wolf optimization (GWO) and particle swarm optimization (PSO). The hyperparameters of the regression model can be optimized using GWO and PSO. In addition, the intelligent optimization algorithm adjusts three critical parameters of the XGB model (learning_rate, maximum_depth, and n_estimator) to achieve a higher accuracy. Maximum depth, learning_rate, and n_estimator represent a tree's maximum depth, the tree's shrinkage coefficient, and the number of trees, respectively.

2.2.1. Extreme gradient boosting (XGBoost)

This method is proposed based on the gradient boosting decision [35-38] by Chen and He [39]. XGBoost has been applied in many engineering fields for classification and regression problems [40]. Due to the advantages of regularization, parallel processing, and efficient tree pruning, it has performed very well. Many data science problems can be solved with XGBoost quickly and accurately with parallel tree boosting. Optimizing the objective function is the core of this algorithm [41].

XGBoost uses the residual in each iteration to calibrate the previous predictor. This process is related to loss function (LOF) optimization. On the other hand, in the calibration process, regularization is used to reduce the overfitting of the objective function. With this description, the objective function, according to Equation (1), consists of two parts: regularization and training loss.

$$Obj(\Theta) = L(\Theta) + \Omega(\Theta) \quad (1)$$

where Θ is the parameter trained from the data, Ω is related to regularization. Regularization is intended to avoid overfitting as it can control the complexity of the model [42]. Training loss functions L are shown, which measure how well the model fits the training data. There are different ways to define complexity. However, the complexity of each tree is often calculated using Equation (2).

$$\Omega(\mathcal{F}) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} \omega_j^2, \quad (2)$$

where the complexity of each leaf is denoted by γ, T is the number of DT leaves, λ scales the penalty, and ω is the vector of scores on the leaves. Next, XGBoost applies the second-order Taylor expansion to the loss function (LOF) general gradient boosting. Equation (3) can obtain the objective function when the mean squared error is considered LOF.

$$J^{(t)} \approx \gamma T + \sum_{j=1}^{T}\left[\left(\sum_{i\in i_j} g_i\right)\omega_j + 0.5\left(\sum_{i\in i_j} h_i + \lambda\right)\omega_j^2\right], \quad (3)$$

The first and second derivatives of the MSE loss function are g_i and h_i, respectively. Also q is a function that assigns a data point to the corresponding leaf. Finally, the XGBoost objective function is calculated from Equation (4).

$$Obj = \sum_{j=1}^{T}\left[G_j\omega_j + \frac{1}{2}(H_j + \lambda)\omega_j^2\right] + \gamma T, \quad (4)$$

Here, ω_j are independent of each other. The definition of the two terms G_j and H_j are given in Equation (5).

$$G_j = \sum_{i\in i_j} g_i \ , H_j = \sum_{i\in i_j} h_i, \quad (5)$$

In general, in XGBoost, the optimization of the objective function can be transformed into a process for at least one quadratic function. An objective function is used to assess the change in model performance after splitting a particular node in the DT. This division will be accepted if the model's performance is higher than before; otherwise, the division will stop. XGBoost has a more vital ability to prevent over-installation, due to the addition of regularization phenomena. Figure 2 shows the structure of XGBoost.

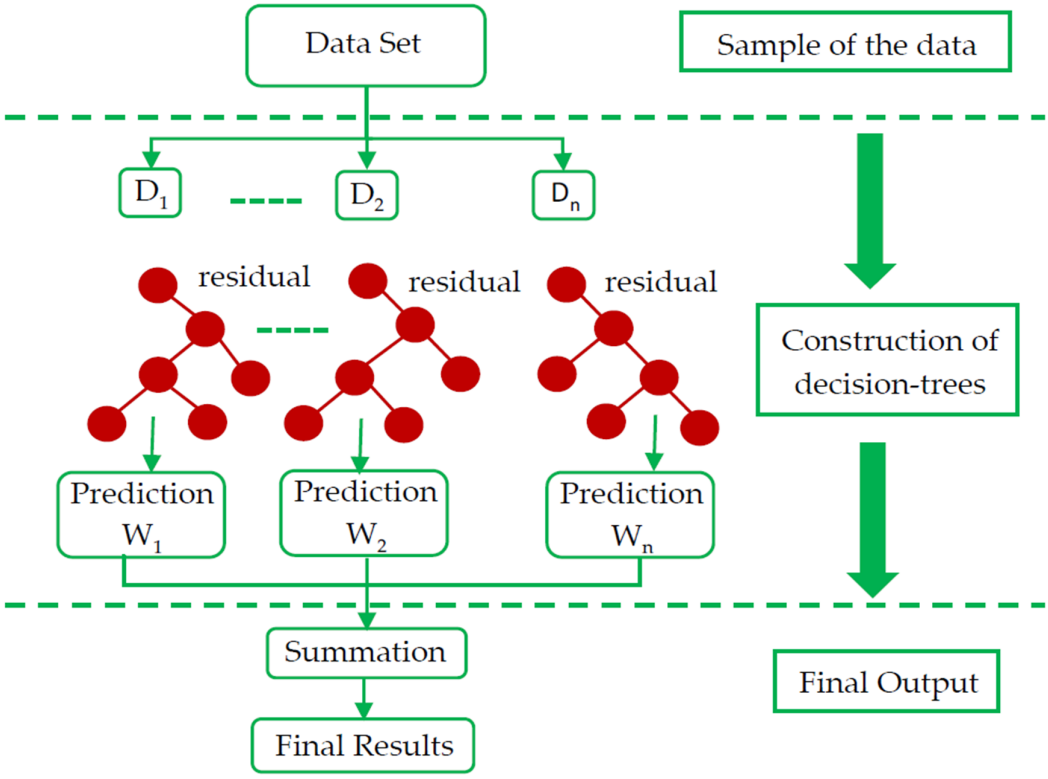


Figure 2. Structure of the XGBoost method.

2.2.2. Particle swarm optimization (PSO)

PSO is an established population-based metaheuristic algorithm for solving problems like function optimization and network training [43]. PSO is derived from two aspects of bird flocks' movement behavior: their position and velocity [44]. Equation (6) can be used to write the PSO formula based on the optimization procedure in the lth iteration.

$$P_{l+1} = P_l + V_{l+1}, \quad (6)$$

being V_l and P_l the particle velocity and particle position, respectively. Based on Equation (6), the updated velocity can be calculated according to finding the most optimal swarm position (Pg) and the personal best value (Pb) using Equation (7).

$$V_{l+1} = a.V_l + c_1.r_1\,(P_l - P_b) + c_2.r_2\,(P_l - P_g) \quad (7)$$

where c_1 and c_2 are the acceleration coefficients, a is the inertia weight, and r_1 and r_2 are randomly generated coefficients at each iteration. Until the target criterion is reached, the repetitive procedure continues.

2.2.3. Gray wolf optimization (GWO)

Similar to PSO, the GWO algorithm is also based on natural inspiration. The main idea is developed by imitating gray wolves' hunting behavior and social hierarchy [45, 46]. Simpleness, flexibility, and avoidance of local optima are some of the positive features of the GWO when dealing with nonlinear and multivariable functions [47].

There are five stages in the GWO modeling process: 1) social hierarchy, 2) encircling prey, 3) tracking and searching for prey (exploration), 4) capturing prey (exploitation), and finally, 5) hunting (optimization). In wolves, packs usually consist of five to twelve individuals grouped into alphas (α), betas (β), deltas (δ), and omegas (ω). Alpha (the highest rank in the hierarchy) makes the final decisions for the pack with the help of betas. A delta wolf submits to an alpha and beta wolf, while an omega wolf always submits to another dominant wolf (the last group permitted to consume prey).

Alpha, beta, and delta agents reach the optimal solution in the GWO [48]. Assuming G represents the wolf's position and Gp represents the prey's position; the updated position of a single wolf is calculated using Equations (8) and (9).

$$G(t+1) = G_p(t) - A.D, \qquad D = \left|C.G_p(t) - G(t)\right| \quad (8)$$

$$A = A \times (2 \times r_1 - 1), \quad C = 2 \times r_2 \quad (9)$$

where A and C are coefficient vectors, and t is the current iteration. r_1 and r_2 are random numbers ranging from [0, 1], and a is a coefficient that varies from 2 to 0. In mathematics, the most optimal solution is called the alpha (G_α), and the second and third are called the beta (G_β) and delta (G_δ), respectively. In addition to the three most appropriate solutions, other solutions are connected to omegas ($G\omega$). Thus Equation (10) can update one wolf's position.

$$G(t+1) = \frac{G_1 + G_2 + G_3}{3}, \quad (10)$$

Alpha, beta, and delta wolves are defined by G1, G2, and G3, respectively. During position updating, they are calculated (Equations (11) and (12)).

$$D_\alpha = \left|C_1.G_\alpha - G\right| \qquad D_\beta = \left|C_1.G_\beta - G\right| \qquad D_\delta = \left|C_1.G_\delta - G\right| \quad (11)$$

$$G_1 = G_\alpha - A_1.D_\alpha \qquad G_2 = G_\beta - A_2.D_\beta \qquad G_3 = G_\delta - A_3.D_\delta \quad (12)$$

Attacking the prey concludes the pack's hunting. The prey must be reached and approached close to do this. Regularizing the attack typically involves reducing a from 2 to 0 in Equation (9). In [45], you can find more details about the GWO algorithm. Figure 3 illustrates the proposed hybrid models for predicting back-break.

2.3. Gene expression programming (GEP)

First, the GEP method, which is a combination of genetic algorithm (GA) and genetic programming (GP), was introduced by Ferreira [49]. GEP, as an AI evolutionary approach, has corrected some of the weaknesses of GA and GP such as tree systems. Individuals or answers are the key reason for the modification of these methods. Individuals are known as fixed-length chromosomes, defined as binary in the GEP algorithm. In the GP method, the answers can include tree systems of different sizes. Since the GEP method combines both GA and GP, it has fixed-length chromosomes and tree systems of different shapes and sizes, known as Expression Trees (ET). The structure of the GEP algorithm includes distinct parts. These include terminal, stop criteria, function, operator, and fitness [49].

Chromosomes have two parts: head and tail, making them a fixed length. The head part includes functions and terminals, and the end part also includes terminals. Due to the problems' complexity, the head part's length, introduced as the input of the GEP algorithm, cannot be obtained by a specific equation, and the only solution is trial and error [50]. By Equation (13) can be obtained length of the tail part.

$$t = h(n_{max} - 1) + 1, \quad (13)$$

where t, h, and n_{max} represent the length of the tail, head, and the number of arguments of the functions.

Each solitary chromosome in the initial population is evaluated using a fitness function determined for gap problems. The considered chromosomes are adapted using several genetic operators. Each chromosome can contain functions, positive and negative constants, and terminals that depend on the problem conditions [51, 52]. The general process of the GEP algorithm is presented as follows:

Step 1: A certain number of chromosomes should be defined (randomly) according to the conditions (size) of the problem under study.

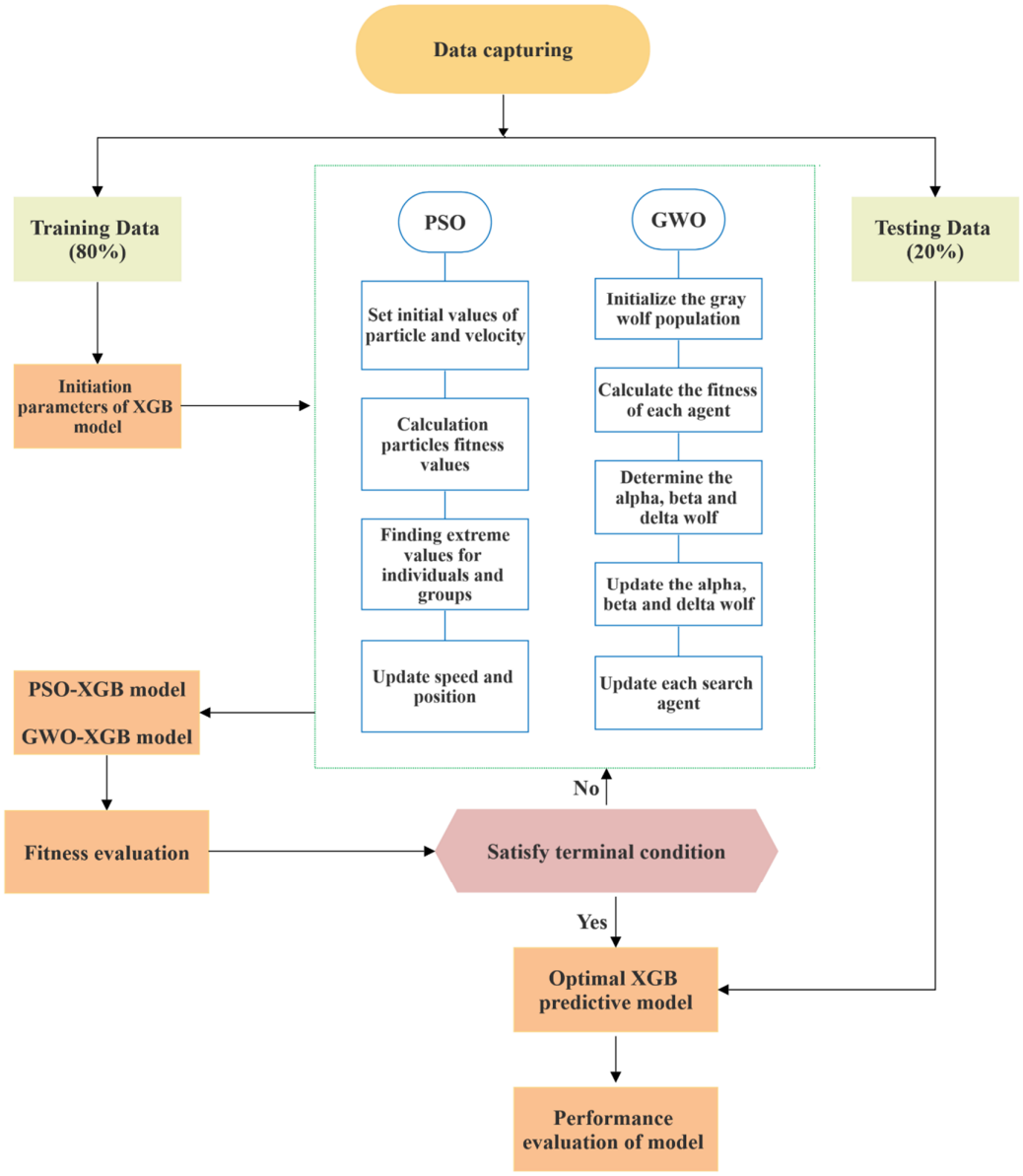


Figure 3. Structure of hybrid model based on XGB for predicting back-break.

Step 3: Fit the chromosomes according to the overall fitness function (root mean square error (RMSE) and coefficient of determination (R^2)). If the stopping criterion is not met, other methods such as the roulette wheel method are used to select the best initial generation.

Step 4: In this step, the genetic operators (identified as the main of the GEP

algorithm) must be linked to the other remaining chromosomes.

Step 5: Finally, the creation of the next generation begins, and the process is repeated to create new structures.

A new language called Karva (K-Expression) to express the codes in the chromosomes was invented to decode the programs in the chromosomes. Genetic operators introduced so far that are used for chromosome modification are inversion, mutation, triple recombination operators, and triple transposition operators [51]. Figure 4 shows the general flowchart of the GEP method.

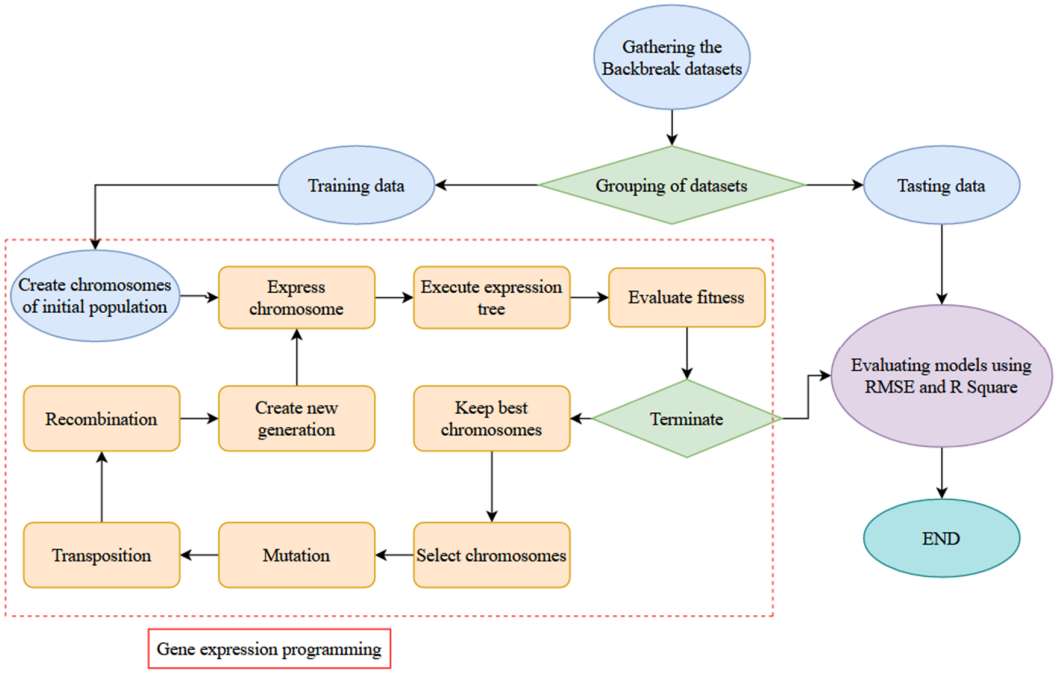


Figure 4. Flowchart of gene expression programming (GEP) algorithm.

2.4. Random forest (RF)

The Decision Tree (DT) is a particular area of Artificial Intelligence (AI), and within this branch lies the Random Forest (RF), which was created by Breiman [53]. RF is a sturdy DT model that can handle both classification and regression issues. Vigneau *et al.* recommend this model as a way to achieve precise predictions based on the varying results of the individual trees [54]. An optimal outcome is established by taking into account the results of each tree in the forest as a whole. Each tree has a say in the final decision-making process, acting as a voter in RF's ensemble predictions [55]. To estimate regression problems like back-break, RF follows a three-step approach: (i) generating bootstrap samples as the number of trees in the forest based on the database; (ii) expanding a suitable regression tree for each bootstrap instance through the random selection of estimators (mtry), then determining the best fit based on a particular variable; and (iii) estimating recent perception by combining the estimated outcomes from each tree in the forest (ntree), with the mean value of estimates being applied to the single tree [56].

2.5. Non-linear multiple regression (NMLR)

In the case of multiple input variables, multiple regression techniques are available to obtain the best-fit equation. A relationship between the input and output parameters is generally the goal of such techniques. The nonlinear multiple regression method (NLMR) finds non-linear relationships between input and output parameters. Statistically, the line of closest fit represents a line whose sum of squares of deviations from the line is the lowest. In the NLMR technique, non-linear and linear relationships, e.g. exponential and power, can be incorporated. In contrast to traditional linear regression, NLMR predicts models with arbitrary relationships between inputs and outputs [28]. There is little focus and effort in research in

predicting BB resulting from blasting using the NLMR model.

2.6. Linear multiple regression (MLR)

This method is based on linear combinations of dichotomous, dummy or interval-independent variables. Multiple regressions aim to gain more information about the relationship between several predictor or independent variables and a criterion or dependent variable [57]. On the other hand, regression analysis aims to determine the values of function parameters that make the function best match a set of data observations. Multiple regression analysis solves the dataset by performing least squares fit, so that it creates and solves the simultaneous equations by forming the regression matrix and solving the coefficient using the backslash operator. The multivariate linear regression prediction model with n regression variables is shown in Equation (14).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n , \quad (14)$$

where β_0 is a constant coefficient, indicating where the regression line intersects the y-axis. It also represents the value of y or the dependent variable (when all of them are 0). The regression coefficients are β_1, β_2, …, β_n. These coefficients show the value of the dependent variable y with a corresponding change of 1 unit of the independent variables [58, 59].

3. Results and Discussion

The datasets used in this research work are divided into training and testing. This way, 80% (72 explosion events) of the entire dataset is used for the training process. For the testing process, the rest (18 observations) are used. The training data and the test dataset are used, respectively, to develop the models and evaluate the performance of the built models. Five statistical measures, the variance accounted for (VAF), mean absolute error (MAE), a-20 index (A20), root mean square error (RMSE), and determination coefficient (R^2), are applied to evaluate the performance of the constructed models [60]. These parameters are calculated using Equations (15) to (19).

$$VAF\ (\%) = \left(1 - \frac{var(y_i - \hat{y}_i)}{var(y_i)}\right) \times 100 \quad (15)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (16)$$

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y}_i)^2} \quad (17)$$

$$MAE = \frac{\sum_{i=1}^{n} \hat{y}_i - y_i}{n} \quad (18)$$

$$A20 = \frac{m20}{M} \quad (19)$$

3.1. XGBoost result

It is necessary to choose appropriate parameter values for an XGBoost model to fit the data. A model's hyperparameters should be selected optimally to prevent overfitting and underfitting. These hyper-parameters are learned and adjusted based on the data to achieve the most suitable fit. Considering the complexity and the length of time of the problem to search for values for these hyperparameters, grid search techniques are applied. The grid search considers a possible set of values. The model is run using all these values, and the predictive accuracy is evaluated with different statistical metrics including MAE and RMSE. The combination of hyper-parameters would be selected that produces the lowest error value. An accurate model can be generated with grid search since it covers all possible combinations of hyper-parameters [61, 62].

Hyper-parameter optimization is performed using the grid search technique in this model. Therefore, to avoid complexity in the modeling of the XGBoost method, three stopping criteria, namely learning rate (shrinkage coefficient of tree), maximum depth (maximum depth of a tree), and n estimators (number of trees), were considered. Three parameters are evaluated here for tuning since they are noted to affect the derived solutions strongly. Default values are set for all other hyper-parameters. According to Table 3, the corresponding grid is now constructed with the following parameters.

Table 3. Adjustment selection and search space of XGBoost model parameters for grid search.

XGBoost parameters	Lower limit	Upper limit	Scale	Number
Learning rate	0.01	0.15	0.01	15
Maximum depth	1	3	1	3
N estimators	50	250	1	200

Table 3 shows the following there are 18000 possible combinations of hyper-parameters in this configuration. After running the model 18000 times, it is optimized based on its minimum RMSE values to determine the optimal hyper-parameters. According to Table 4, those hyper-parameters are optimal for grid search.

Table 4. An overview of predictive performance and the model's optimal parameters with grid search.

Method	Search approach	Training data			Testing data			Parameters
		R^2	RMSE	MAE	R^2	RMSE	MAE	
XGB	Grid search	0.978	0.185	0.132	0.946	0.291	0.2	n_estimator = 173 maximum_depth = 1 learning_rate = 0.118

The best XGBoost models with the grid search approach are shown in Table 4. This optimized XGB model has shown high accuracy with an R^2 = 0.978 value. Figure 5 illustrates the scatter plot of the predicted and measured back-break with the trend line in the selected XGBoost model.

The testing dataset was used to assess and confirm the effectiveness of the XGBoost model. Figure 6 illustrates that the test sample points are predominantly situated in close proximity to the ideal fit line, indicating a strong correlation between predicted back-break values and actual back-break values.

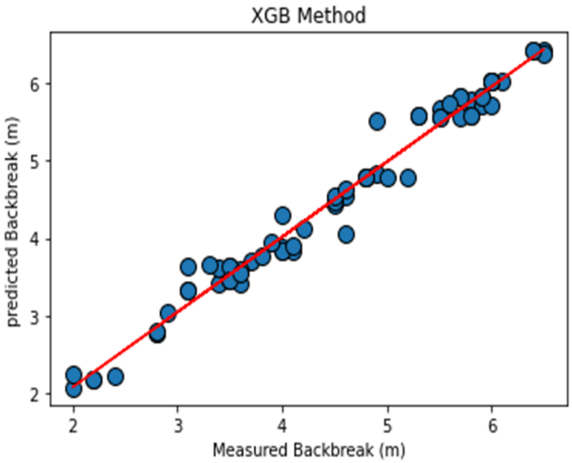


Figure 5. Predicted and measured values on the training dataset by selected XGBoost method.

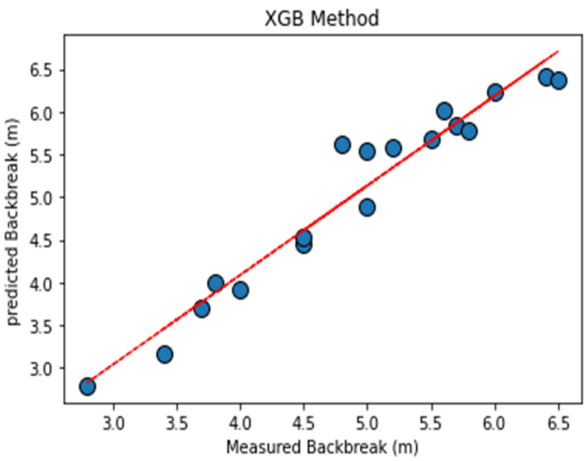


Figure 6. Predicted and measured values on the testing dataset by selected XGBoost method.

3.2. Metaheuristic result (Hybrid model)

Similarly, XGBoost method, three stopping criteria 1) maximum depth, 2) n estimators 3) learning rate were considered to avoid complexity. An assessment of significant values for each parameter can result in overfitting. Therefore, using gray wolf optimization (GWO) and particle swarm optimization (PSO), XGB parameters have been optimally determined. In Figure 2, a diagram illustrates the methods used to develop models based on the XGB data. XGB hybrid model parameters must now be set. An overview of the optimization algorithm's parameters can be found in Table 5. Furthermore, Table 5 displays the optimal parameter values for the model based on the optimization results.

Table 5. An overview of parameters of algorithms and the model's optimal parameters in the hybrid model.

Algorithm	parameters	Value	Optimal parameters
GWO	Convergence constant a linear decrease	[2,0]	n_estimator = 158 maximum_depth = 1 learning_rate = 0.129
PSO	Minimum inertia weight	0.2	learning_rate = 0.123
	Maximum inertia weight	0.9	
	Maximum velocity	6	maximum_depth = 1
	Cognitive coefficient 1	2	n_estimator = 163
	Cognitive coefficient 2	2	

In the following step, the two hybrid intelligent models based on XGB were trained using the trained data, and then prediction performance models were compared. According to Figure 7, the actual value of the training dataset is correlated with the predicted value. In intelligent models, the training sample points are primarily distributed near the closest fitting line, indicating a relatively favorable training effect. A pair of XGB-based optimization methods are proposed in this paper, with R^2 values of 0.99 demonstrating a high training effect.

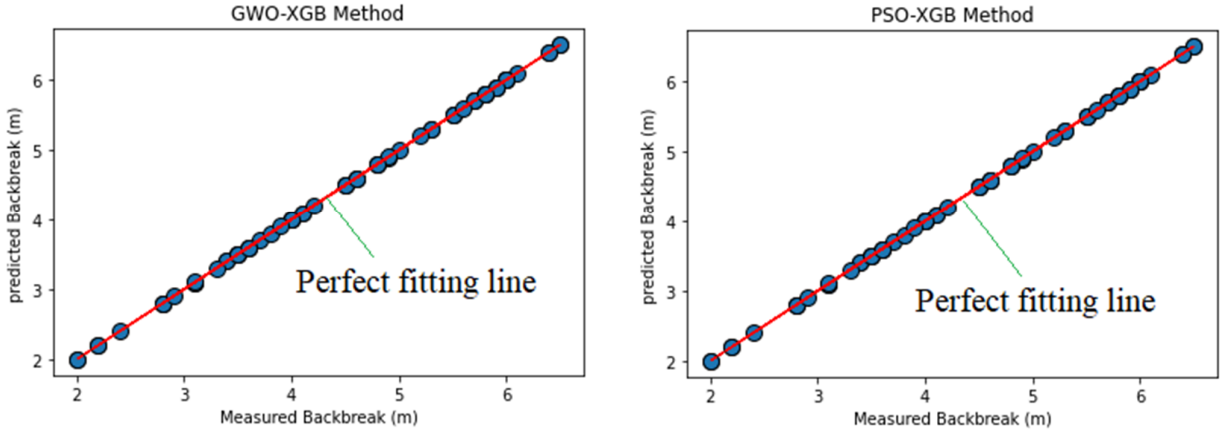


Figure 7. Comparison of correlation between predicted and actual values in the training dataset.

After model training, two hybrid intelligent models are evaluated and verified using the testing data set. In Figure 8, the points in the test sample are mainly located near the perfect fit line. Back-break values and predicted back-break values show a close correlation. Despite the high levels of prediction accuracy achieved by both hybrid models, the GWO-XGB model is superior.

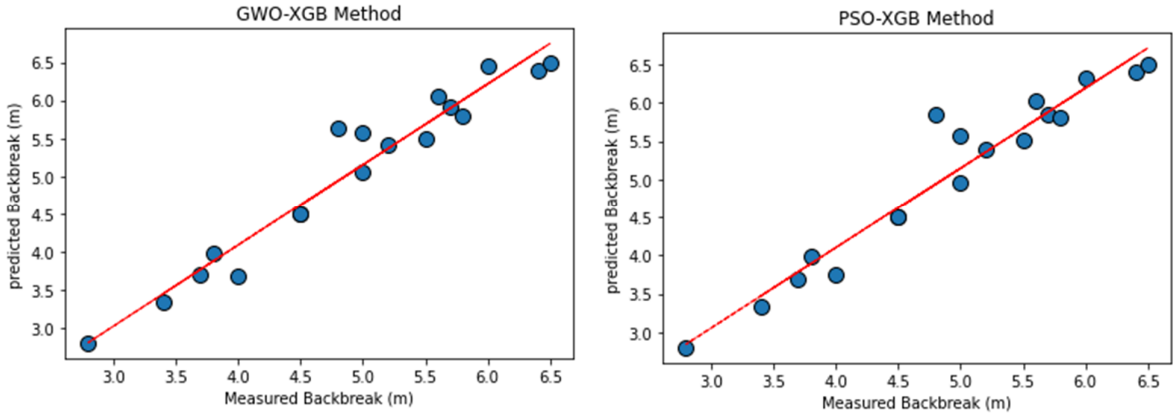


Figure 8. Comparison of correlation between predicted and actual values in the testing dataset.

As a result, the performance of the ordinary XGBoost optimized by grid search is close to the hybrid models. But using metaheuristic algorithms in hyper-parameters tuning has several benefits. The major benefit of using metaheuristic algorithms in hyper-parameter tuning over grid search is the reduction in time and cost. Metaheuristic algorithms can explore the hyper-parameter space more efficiently, reducing the number of evaluations needed to find a good solution. This leads to faster and more cost-effective hyper-parameter tuning compared to grid search, especially for large datasets or complex models. Additionally, metaheuristic algorithms are more likely to find near-optimal solutions, which can save time and resources in the long run. Overall, metaheuristic algorithms can offer a more efficient strategy for hyper-parameter tuning than grid search, leading to a better model performance and reduced computational cost.

3.3. LMR result

Multiple regression was performed on the parameters of back-break (BB) and burden(B), spacing (S), stemming (ST), diameter (D), hole depth (H), specific charge (SC), and the number of rows (NR). Below is the developed multiple regression model to predict back-break:

$$BB = 0.4D - 0.09B + 0.19S + 0.23ST + 0.26H + 2.96SC + 0.04NR - 3.72$$

Training data: $R^2 = 0.965$, RMSE = 0.235 , MAE = 0.181 (20)

Testing data: $R^2 = 0.937$, RMSE = 0.257 , MAE = 0.196

According to the explanation, MAE, RMSE, and R^2 were calculated to control the performance of the forecasting capacity of the predictive model in this study. Figure 9 shows the scatter plot of the predicted and measured back-break with the trend line.

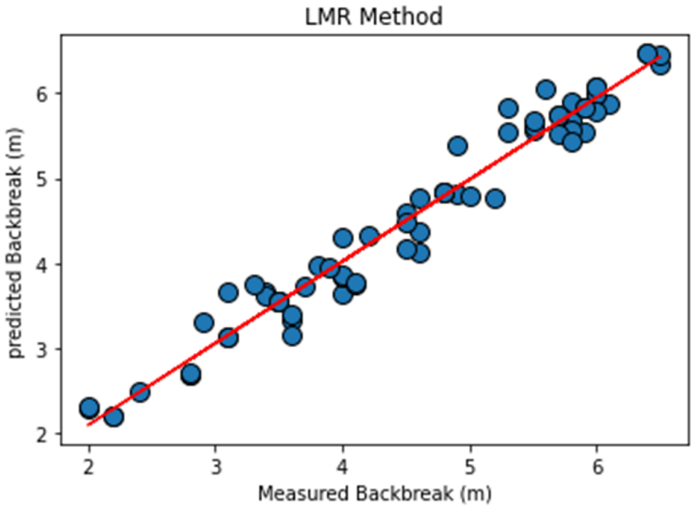


Figure 9. Predicted and measured values on the training dataset by LMR method.

3.4. GEP results

The gene expression programming (GEP) modeling process is presented in the flowchart shown in Figure 3. Similar to the previous modeling sections, the same training and testing datasets have been used to design the GEP. This research work used the GEP algorithm software (GeneXproTools 5.0) to analyze and obtain the final relationship between the initial data and Backbreak. Various mathematical operators (e.g., +, –, *, /, x2, inverse, Ln, and Exp) have been used to predict BB and develop a mathematical relationship using GEP.

One should consider the fitting parameters to build an efficient model. The execution time, vital in the model's performance, is controlled by the number of chromosomes. In addition, the number of genes and the head size must be involved to have

a suitable architecture. The head size and number of genes determine each component's complexity and the number of related equations, respectively. A trial-and-error mechanism was applied regarding the above elements to achieve the best GEP model (Table 6).

Table 6. Constructed 10 GEP models for predicting Backbreak.

GEP parameters	Value									
	GEP model number									
	1	2	3	4	5	6	7	8	9	10
Fitness function	RMSE	RMSE	RMSE	MAE	MSE	RMSE	RMSE	RMSE	RMSE	RMSE
No. chromosomes	30	40	40	40	40	40	50	60	40	40
Head size	8	8	8	8	8	8	8	8	10	10
No. genes	4	4	3	4	4	4	4	4	4	3
Linking function	+	+	+	+	+	×	+	+	+	+

According to other methods, using R^2, MAE, and RMSE indices, the performance prediction of GEP models was evaluated for training and testing datasets. Among the ten models stated in Table 6, which had the highest performance predicting back-break, five models were selected and shown in Table 7. As a result, as shown in Table 7, model No. 2 is the best model among all the models made by the GEP method. Figure 10 illustrates the scatter plot of the predicted and measured back-break with the trend line in the selected GEP model.

Table 7. GEP models with Performance indices

Method	Model No. Table (6)	Training			Testing			Rank
		RMSE	R2	MAE	RMSE	R2	MAE	
GEP	2	0.233	0.965	0.186	0.25	0.958	0.197	1
	3	0.237	0.963	0.186	0.257	0.959	0.205	2
	4	0.245	0.962	0.166	0.310	0.937	0.231	3
	5	0.252	0.959	0.208	0.266	0.947	0.216	5
	9	0.25	0.96	0.168	0.33	0.94	0.259	4

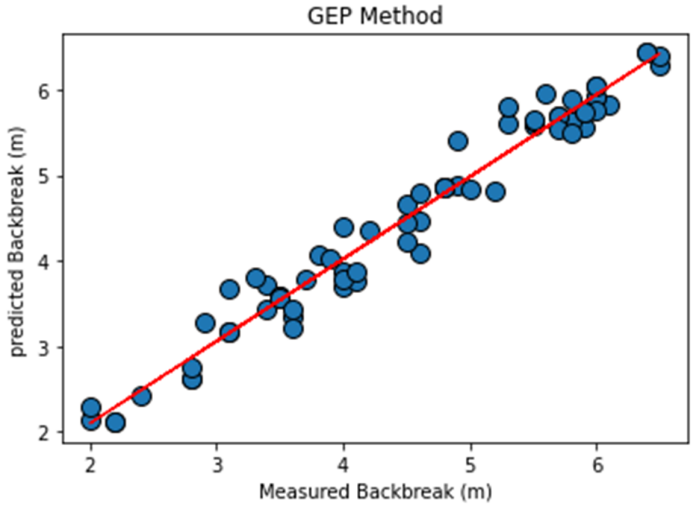


Figure 10. Predicted and measured values on the training dataset by selected GEP method.

The number of genes and head size in Model No. 2 is 4 and 8, respectively. The expression tree (sub-ETs of each gene) of the selected model is displayed in Figure 11. The expression trees were eventually linked with addition by linking the functions to create a large and complex ET. Equations (21) to (24) show the mathematical equations of each of these genes. Finally, Equation (25) shows the developed GEP equation for predicting setbacks.

$$Gene\ 1: \left[5.44 + \frac{1}{((NR - 7.44) * (H + D)) + (D * ST)}\right] \quad (21)$$

$$Gene\ 2: \left[\left(ST + \frac{0.59 - ST}{H - D}\right) + (SC + SC)^2\right] \quad (22)$$

$$Gene\ 3: \left[\frac{1}{S} * (-14.79 - Exp((ST * -4.46) + 7.76)\right] \quad (23)$$

$$Gene4: \left[-3.51 \times \left((Exp\left(\frac{ST}{B}\right) - 5.26) * SC\right)\right] \quad (24)$$

$$Backbreak = \left[5.44 + \frac{1}{((NR - 7.44) * (H + D)) + (D * ST)}\right] + \left[\left(ST + \frac{0.59 - ST}{H - D}\right) + (SC + SC)^2\right] + \left[\frac{1}{S} * (-14.79 - Exp((ST * -4.46) + 7.76)\right] + \left[-3.51 \times \left((Exp\left(\frac{ST}{B}\right) - 5.26) * SC\right)\right] \quad (25)$$

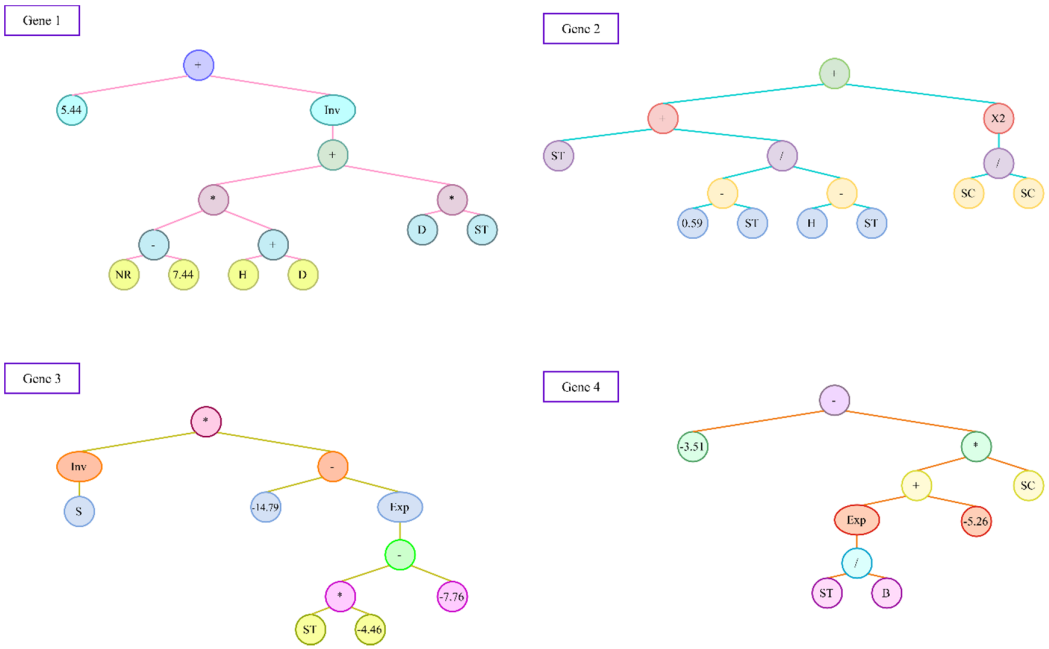


Figure 11. Sub-ETs of each gene for the best GEP model with addition as a linking function.

3.5. NLMR result

Using the same data for training and testing, multiple equations for back-break prediction were proposed utilizing NLMR. An NLMR equation was created using version 26 of the SPSS package. The model inputs were burden (B), spacing (S), stemming (ST), hole diameter (D), hole depth (H), specific charge (SC), and the number of rows (NR). Simple regression functions were employed to develop the NLMR equation. Equation 24 shows how the NLMR model predicts the back-break. For the training and testing datasets of the newly constructed NLMR equation, R^2 values of 0.966 and 0.959 were obtained. The scatter plot of the predicted and measured back-break with the trend line is shown in Figure 12.

$$BB = 2.021D^{0.23} + 3.9B^{-0.14} + 0.001S^{5.24} + 0.2ST + 0.989H^{63} + 36.65SC^{5.15} + 0.32NR^{1.07} - 7.37$$

Training data: $R^2 = 0.966$, RMSE = 0.229 , MAE = 0.177 (26)

Testing data: $R^2 = 0.959$, RMSE = 0.255 , MAE = 0.191

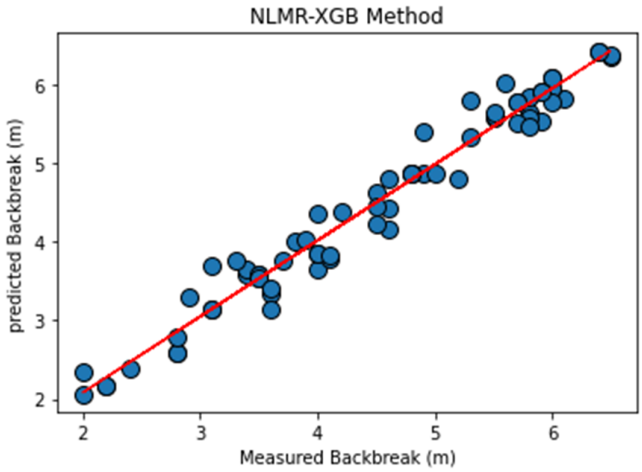


Figure 12. Predicted and measured values on the training dataset by NLMR method.

3.6. RF result

Based on the seven variables listed above, the RF model was developed. RF models were developed using train and test partitions applied to all models. This means that 80% of the data (n = 72) was used for the training phase, and 20% for the test phase (n = 18). The random forest (RF) method has been applied in several studies for backbreak [6, 30, 63]. Thus this model is used as a criterion to check the performance of developed models.

According to earlier discussion, RF builds an ensemble model that incorporates a wide variety of DTs. Based on the methodology of the CART, the RF is worth noting. Furthermore, several parameters were used for the development and fine-tuning of the RF model. A trial-and-error method was used by the authors to achieve the optimal combination of these parameters. RF models with 97.4% accuracy for training datasets and 94% accuracy for testing datasets had the highest accuracy. Figure 13 depicts the relationship between the measured and predicted back-break, which demonstrates a high interconnectivity.

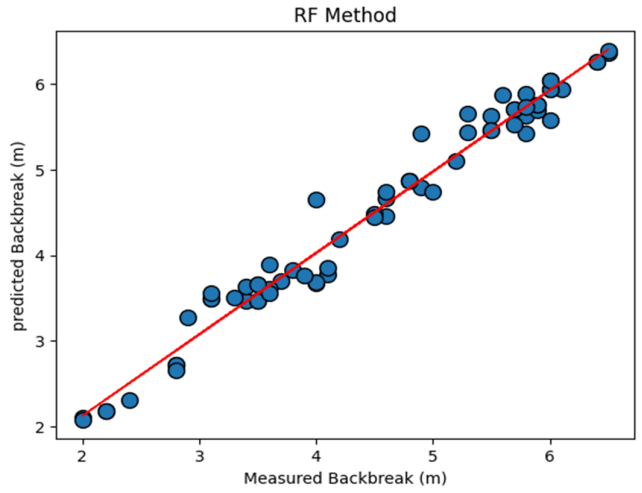


Figure 13. Predicted and measured values on the training dataset by RF method.

3.7. Validation performance and comparison of models

This section is to evaluate the efficiency of the developed models. This study used MAE, R^2, and RMSE (Equations 15 to 17) to measure the performance of the selected prediction models. In the first step, optimal models were chosen for each approach. Then according to Table 8, the above statistical criteria for all models were calculated for the training and test datasets. According to these results, the performance and accuracy level of the GWO-XGBoost technique is better than the PSO-XGB, XGBoost, GEP, RF, LMR, and NLMR models. The performance of the models in predicting the back-break caused by the blasting in the testing data set is shown in Figure 14.

Table 8. Selected predictive models with statistical values.

Model	Training data					Testing data				
	R^2	RMSE	MAE	VAF	a-20	R^2	RMSE	MAE	VAF	a-20
GWO-XGB	0.99	0.01	0.001	0.99	0.98	0.946	0.281	0.186	0.912	0.91
PSO-XGB	0.99	0.01	0.001	0.99	0.98	0.934	0.36	0.205	0.901	0.88
XGBoost	0.978	0.185	0.132	0.98	0.95	0.946	0.291	0.2	0.92	0.98
GEP	0.965	0.233	0.186	0.967	0.935	0.96	0.25	0.197	0.94	0.96
RF	0.974	0.210	0.156	0.97	0.94	0.94	0.295	0.215	0.915	0.92
LMR	0.965	0.235	0.181	0.964	0.92	0.962	0.257	0.196	0.937	0.96
NLMR	0.966	0.229	0.177	0.968	0.93	0.959	0.255	0.191	0.938	0.93

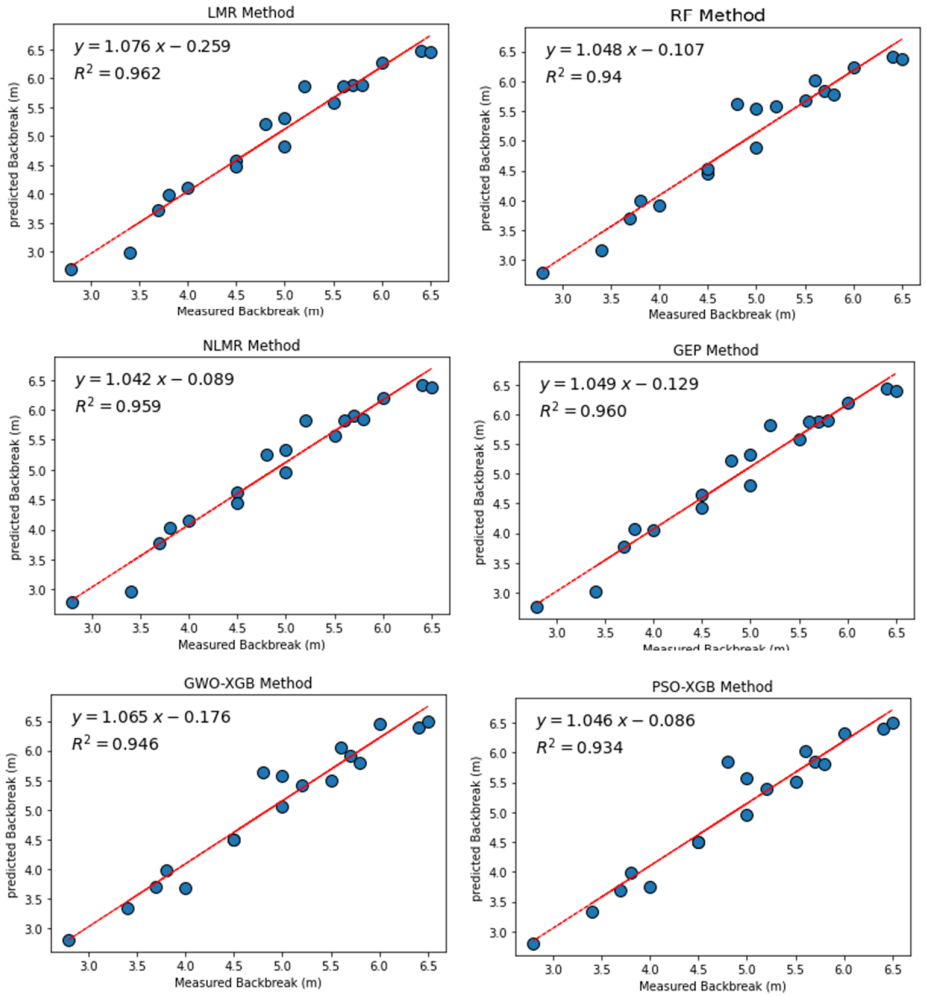


Figure 14. Predicted and measured values on the training dataset by selected XGBoost method.

In the next step, the selected BB prediction models' accuracy is compared, as shown in Figure 15. According to Figure 15, the hybrid XGBoost technique gives the most reliable and consistent results in BB prediction among the developed models.

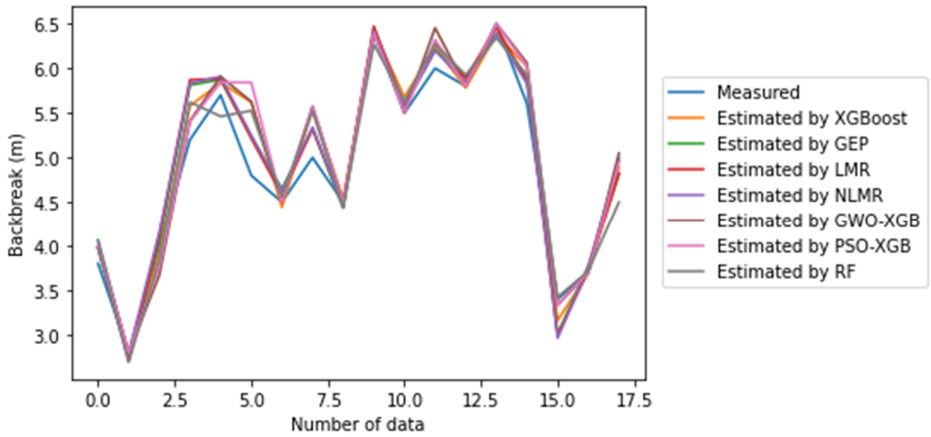


Figure 15. Selected predictive models with prediction values on testing datasets.

In this subsection, we present the Taylor diagram that illustrates the performance of optimal predictive models at various testing stages. Taylor diagrams briefly assess a model's accuracy in two dimensions [64]. The ratio of standard deviation, RMSE, and R indicate the relationship between the actual and estimated observations. The Taylor diagram represents each model as a point. In an ideal model, the point's position would be close to the reference point. Figures 16 and 17 illustrate the Taylor diagram of this study's models generated for testing and training datasets. According to Figure 17, the GWO-XGB hybrid model is more accurate in predicting back-break than other predictive models.

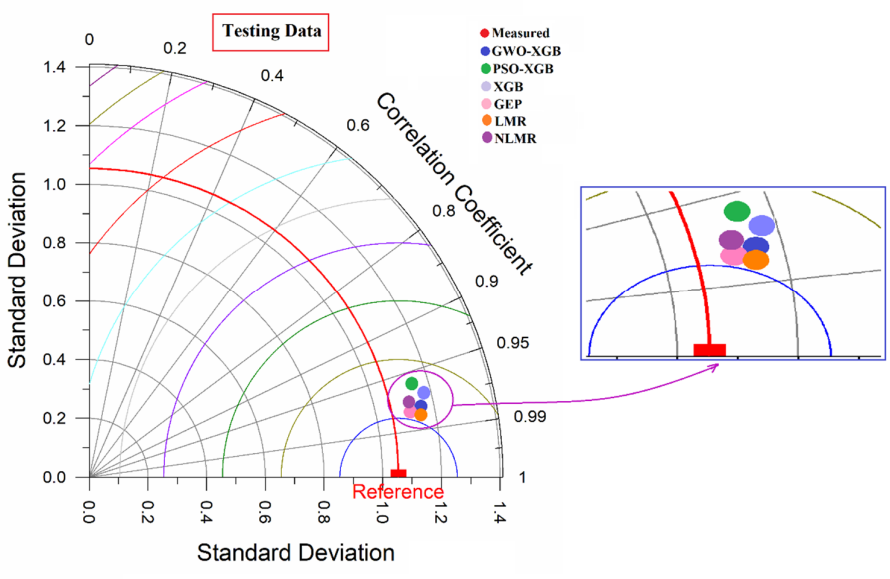


Figure 16. Comparative analysis of developed models with testing datasets in Taylor diagrams.

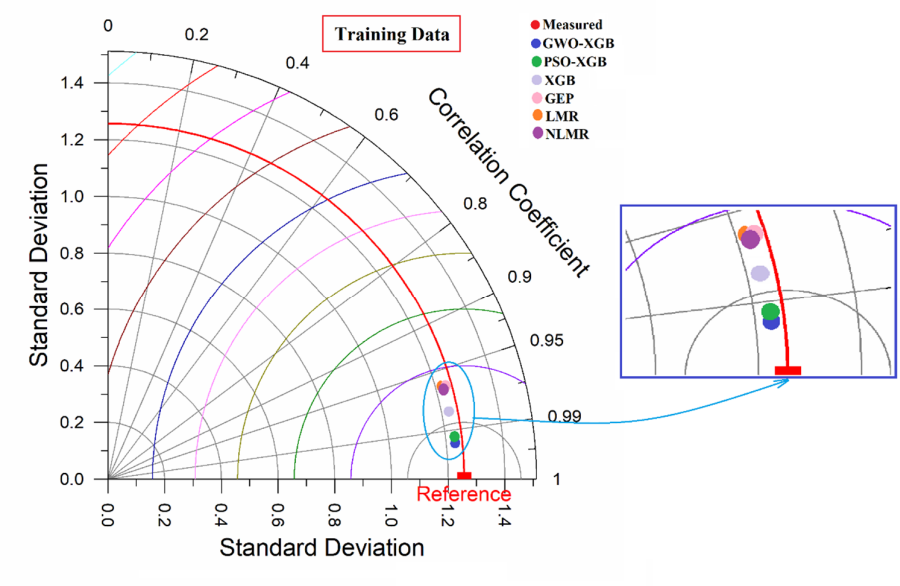
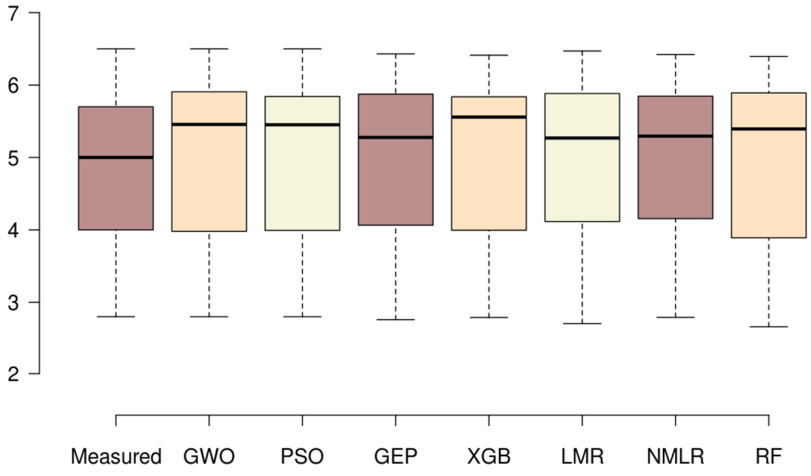


Figure 17. Comparative analysis of developed models with training datasets in Taylor diagrams.

The distribution of predicted values is one way to evaluate predictive models. Box plots in Figure 18 illustrate the distribution functions for the measured and predicted back-break values in testing data. Due to their similar probability distributions with observational results, hybrid XGBoost approaches resulted in the most promising performance compared to the other models.



Quartiles	Measured	GWO	PSO	GEP	XGB	LMR	NMLR	RF
Upper whisker	6.50	6.50	6.50	6.43	6.41	6.47	6.42	6.39
3rd quartile	5.70	5.91	5.84	5.87	5.84	5.88	5.85	5.89
Median	5.00	5.46	5.45	5.28	5.56	5.27	5.29	5.39
1st quartile	4.00	3.98	3.99	4.06	3.99	4.12	4.16	3.89
Lower whisker	2.80	2.80	2.80	2.76	2.79	2.71	2.79	2.66

Figure 18. Box plot of all predictive models for testing data.

4. Sensitivity Analysis

During the last modeling stage, the model's output is evaluated in relation to the input parameters. A sensitivity analysis can assess the relative effects of model input parameters on model output (objective function). The cosine amplitude method (CAM) is one of the methods for determining sensitivity analysis [5, 65]. The CAM

method considers a space (n-dimensional), where n is the number of input parameters (Equation (27)).

$$X = \{X_1, X_2, X_3 \dots, X_n\} \quad (27)$$

Each member of this input parameter such as X is connected to the objective function by a length vector. That means:

$$X_i = \{X_{i1}, X_{i2}, X_{i3} \dots, X_{in}\} \quad (28)$$

The effect of each input parameter X on the objective function can be obtained using Equation (29).

$$R_{ij} = \frac{\sum_{k=1}^{n} X_{ik} X_{jk}}{\sqrt{\sum_{k=1}^{m} X_{ik}^2 \sum_{k=1}^{m} X_{jk}^2}} \quad (29)$$

As R_{ij} approaches the value of one, it indicates more influence of the input parameters on the outputs. An output parameter with a value above 0.9 significantly impacts the input parameter [5]. Figure 19 shows the results of the sensitivity analysis of the regression prediction parameters. According to Figure 19, hole depth and stemming had the most significant influential effect among the input elements on the back-break. Also the number of rows showed the lowest impact on the back-break.

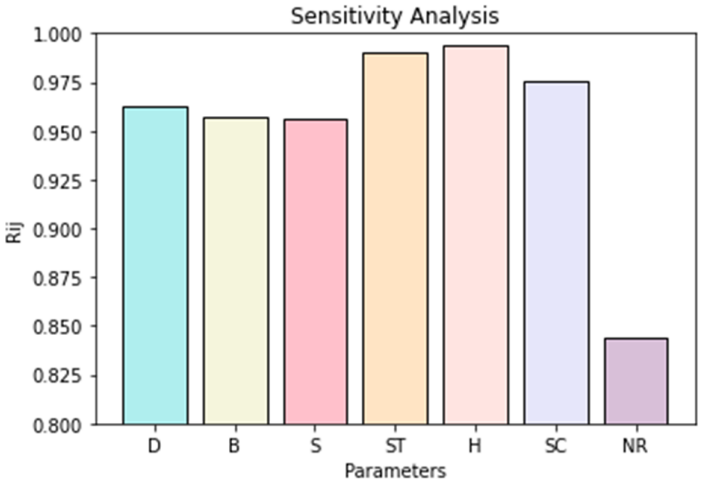


Figure 17. Effect and sensitivity analysis of input data on back-break.

5. Conclusions

Back-break is one of the unintended consequences of blasting. This consequence causes the reduction of drilling efficiency, the instability of walls, the fall of machines, and finally, the increase in the cost of mining. Hence, the back-break must be accurately predicted, and the subsequent explosion planned accordingly. This plan will improve safety and reduce the production cost of mines. This study attempts to predict the back-break of the Chadormalu mine blasting operation using the hybrid XGBoost machine, which is optimized by gray wolf optimization (GWO) and particle swarm optimization (PSO). Furthermore, XGBoost (optimized with grid search), GEP, RF, LMR, and NLMR were used to measure the validity of the hybrid model. The effectiveness of the models was evaluated by measuring their performance using MAE, RMSE, and R2 metrics. Finally, the cosine amplitude method was utilized to determine the importance of each input variable.

The parameters include burden (B), spacing (S), stemming (ST), hole diameter (D), hole depth (H), specific charge (SC), and the number of rows (NR) involved in developing predictive models. In summary, this study suggests that the hybrid XGB models have potential in predicting back-break, and can effectively assist XGB in adjusting hyper-parameters. The performance of the predictive model for the training data dataset is ranked from high to low as follows: GWO-XGB (R2 = 99, RMSE = 0.01, MAE = 0.001, VAF = 0.99, a-20 = 0.98), and PSO-XGB (99, 0.01, 0.001, 0.99, 0.98) is better than XGBoost (97, 0.185, 0.132, 0.98, 95), GEP (96, 0.233, 0.186, 0.967, 0.935), RF (97, 0.210, 0.156, 0.97, 0.94), LMR (96, 0.235, 0.181, 0.964, 0.92), and NLMR (96, 0.229, 0.177, 0.968, 0.93). Notably, the GWO-XGB hybrid model has superior overall performance as compared to the other models. The according to the sensitivity

analysis results, among the input parameters, hole depth and stemming had the most significant effect on the prediction of back-break. The proposed GWO-XGB model proved robust and applicable to predicting back-break in this study.

In conclusion, the hybrid model developed in this study showed promising results in predicting backbreak in mining operations. However, there are several limitations that should be acknowledged. Firstly, the model was trained and tested on a single dataset, and its performance on other datasets remains to be evaluated. Secondly, the model relies on various input parameters, some of which may not be readily available in real-world scenarios. Finally, the model's accuracy may be affected by other factors not considered in this study.

In light of these limitations, future research can explore methods to improve the model's robustness and generalizability. Additionally, investigations can be carried out to assess the model's applicability to other mining operations and identify ways to streamline data collection and input parameter selection.

Abbreviation

Symbol	Explanation	Symbol	Explanation
UCS	Uniaxial compressive strength	D	Diameter
W	Water content	B	Burden
S	Spacing	ST	Stemming length
HD	Hole diameter	BH	Bench height
PF	Powder factor	N	No. of rows
K	Stiffness ratio	HL	Hole length
CPD	Charge per delay	LCT	Last row charge to total charge ratio
JC	Joint condition	ANN	Artificial neural network
PSO	Particle swarm optimization	RT	Regression tree
GP	Genetic programming	ANFIS	Adaptive neuro-fuzzy inference system
RFA	Random forest algorithm	MFO	Moth flame optimization
ELM	Extreme learning machine	FOA	Fruit fly optimization
WOA	Whale optimization algorithm	LOA	Lion swarm optimization
SOA	Seagull optimization algorithm	SSA	Sparrow search algorithm

Reference

[1]. Jahed Armaghani, D., Tonnizam Mohamad, E., Hajihassani, M., Alavi Nezhad Khalil Abad, S.V., Marto, A., and Moghaddam, M.R. (2016). Evaluation and prediction of fly-rock resulting from blasting operations using empirical and computational methods. Engineering with Computers, 32 (1): 109-121.

[2]. Agrawal, H. and Mishra, A.K. (2018). Evaluation of initiating system by measurement of seismic energy dissipation in surface blasting. Arabian Journal of Geosciences, 11 (13): 1-12.

[3]. Dai, Y., Khandelwal, M., Qiu, Y., Zhou, J., Monjezi, M., and Yang, P. (2022). A hybrid metaheuristic approach using random forest and particle swarm optimization to study and evaluate backbreak in open-pit blasting. Neural Computing and Applications, 34 (8): 6273-6288.

[4]. Ramesh Murlidhar, B., Yazdani Bejarbaneh, B., Jahed Armaghani, D., Mohammed, A.S., and Tonnizam Mohamad, E. (2021). Application of tree-based predictive models to forecast air overpressure induced by mine blasting. Natural Resources Research, 30 (2): 1865-1887.

[5]. Sirjani, A.K., Sereshki, F., Ataei, M., and Hosseini, M.A. (2022). Prediction of Backbreak in the Blasting Operations using Artificial Neural Network (ANN) Model and Statistical Models (Case study: Gol-e-Gohar Iron Ore Mine No. 1). Archives of Mining Sciences, 107-121.

[6]. Sharma, M., Choudhary, B.S., and Agrawal, H. (2021). Prediction and assessment of back break by multivariate regression analysis, and random forest algorithm in hot strata/fiery seam of open-pit coal mine.

[7]. Gates, W.C., Ortiz, L.T., and Florez, R. M. (2005, June). Analysis of rockfall and blasting backbreak problems, US 550, Molas Pass, CO. In Alaska Rocks 2005, The 40th US Symposium on Rock Mechanics (USRMS). OnePetro.

[8]. Mohammadnejad, M., Gholami, R., Sereshki, F., and Jamshidi, A. (2013). A new methodology to predict backbreak in blasting operation. International journal of rock mechanics and mining sciences, 60, 75-81.

[9]. Sari, M., Ghasemi, E., and Ataei, M. (2014). Stochastic modeling approach for the evaluation of backbreak due to blasting operations in open pit mines. Rock mechanics and rock engineering, 47 (2): 771-783.

[10]. Esmaeili, M., Osanloo, M., Rashidinejad, F., Aghajani Bazzazi, A., and Taji, M. (2014). Multiple regression, ANN and ANFIS models for prediction of backbreak in the open pit blasting. Engineering with computers, 30 (4): 549-558.

[11]. Konya, C.J. and Walter, E. J. (1991). Rock blasting and overbreak control (No. FHWA-HI-92-001; NHI-13211). United States. Federal Highway Administration.

[12]. Monjezi, M., Amini Khoshalan, H., and Yazdian Varjani, A. (2012). Prediction of flyrock and backbreak in open pit blasting operation: a neuro-genetic approach. Arabian Journal of Geosciences, 5 (3): 441-448.

[13]. Roy, M.P., Mishra, A.K., Agrawal, H., and Singh, P.K. (2020). Blast vibration dependence on total explosives weight in open-pit blasting. Arabian Journal of Geosciences, 13 (13): 1-8.

[14]. Singh, C.P., Agrawal, H.E.M.A.N.T., Mishra, A.K., and Singh, P.K. (2019). Reducing environmental hazards of blasting using electronic detonators in a large opencast coal project-a case study. J Mines Met Fuels, 67(7): 345-350.

[15]. Enayatollahi, I. and Aghajani-Bazzazi, A. (2009, September). Evaluation of salt-ANFO mixture in back break reduction by data envelopment analysis. In Proceedings of the 9th international symposium on rock fragmentation by blasting, Granada, Spain (pp. 127-133).

[16]. Enayatollahi, I. and Aghajani-Bazzazi, A. (2009). Evaluation of salt-ANFO mixture in back break reduction by data envelopment analysis. In Proceedings of the 9th international symposium on rock fragmentation by blasting, Granada, Spain (pp. 127-133).

[17]. Faradonbeh, R.S., Armaghani, D.J., Monjezi, M., and Mohamad, E.T. (2016). Genetic programming and gene expression programming for fly-rock assessment due to mine blasting. International Journal of Rock Mechanics and Mining Sciences, 88, 254-264.

[18]. Lawal, A.I., Kwon, S., Hammed, O.S., and Idris, M.A. (2021). Blast-induced ground vibration prediction in granite quarries: An application of gene expression programming, ANFIS, and sine cosine algorithm optimized ANN. International Journal of Mining Science and Technology, 31 (2): 265-277.

[19]. Mahdiyar, A., Jahed Armaghani, D., Koopialipoor, M., Hedayat, A., Abdullah, A., and Yahya, K. (2020). Practical risk assessment of ground vibrations resulting from blasting, using gene expression programming and Monte Carlo simulation techniques. Applied Sciences, 10 (2): 472.

[20]. Nguyen, H., Bui, X.N., Bui, H.B., and Cuong, D.T. (2019). Developing an XGBoost model to predict blast-induced peak particle velocity in an open-pit mine: a case study. Acta Geophysica, 67 (2): 477-490.

[21]. Zhang, X., Nguyen, H., Bui, X.N., Tran, Q.H., Nguyen, D. A., Bui, D.T., and Moayedi, H. (2020). Novel soft computing model for predicting blast-induced ground vibration in open-pit mines based on particle swarm optimization and XGBoost. Natural Resources Research, 29 (2): 711-721.

[22]. Chandrahas, N., Choudhary, B.S., Teja, M.V., Venkataramayya, M.S., and Prasad, N.S.R. (2022). XG Boost Algorithm to Simultaneous Prediction of Rock Fragmentation and Induced Ground Vibration using Unique Blast Data. Applied Sciences, 12 (10): 5269.

[23]. Monjezi, M., Rezaei, M., and Yazdian Varjani, A. (2010). Prediction of backbreak in open-pit blasting using fuzzy set theory. Expert Systems with Applications, 37 (3): 2637-2643.

[24]. Monjezi, M., Ahmadi, Z., Varjani, A. Y., and Khandelwal, M. (2013). Backbreak prediction in the Chadormalu iron mine using artificial neural network. Neural Computing and Applications, 23 (3): 1101-1107.

[25]. Ghasemi, E. (2017). Particle swarm optimization approach for forecasting backbreak induced by bench blasting. Neural Computing and Applications, 28 (7): 1855-1862.

[26]. Saghatforoush, A., Monjezi, M., Shirani Faradonbeh, R., and Jahed Armaghani, D. (2016). Combination of neural network and ant colony optimization algorithms for prediction and optimization of fly-rock and back-break induced by blasting. Engineering with Computers, 32 (2): 255-266.

[27]. Ghasemi, E., Amnieh, H. B., and Bagherpour, R. (2016). Assessment of backbreak due to blasting operation in open pit mines: a case study. Environmental Earth Sciences, 75 (7): 1-11.

[28]. Shirani Faradonbeh, R., Monjezi, M., and Jahed Armaghani, D. (2016). Genetic programing and nonlinear multiple regression techniques to predict backbreak in blasting operation. Engineering with computers, 32(1): 123-133.

[29]. Hasanipanah, M., Shahnazar, A., Arab, H., Golzar, S.B., and Amiri, M. (2017). Developing a new hybrid-AI model to predict blast-induced backbreak. Engineering with Computers, 33 (3): 349-359.

[30]. Yu, Q., Monjezi, M., Mohammed, A.S., Dehghani, H., Armaghani, D.J., and Ulrikh, D.V. (2021). Optimized support vector machines combined with evolutionary random forest for prediction of back-break caused by blasting operation. Sustainability, 13 (22): 12797.

[31]. Kumar, S., Mishra, A.K., and Choudhary, B.S. (2022). Prediction of back break in blasting using random decision trees. Engineering with Computers, 38 (2): 1185-1191.

[32]. Li, C., Zhou, J., Khandelwal, M., Zhang, X., Monjezi, M., and Qiu, Y. (2022). Six Novel Hybrid Extreme Learning Machine–Swarm Intelligence

Optimization (ELM–SIO) Models for Predicting Backbreak in Open-Pit Blasting. Natural Resources Research, 1-23.

[33]. Fan, J., Wu, L., Ma, X., Zhou, H., and Zhang, F. (2020). Hybrid support vector machines with heuristic algorithms for prediction of daily diffuse solar radiation in air-polluted regions. Renewable Energy, 145, 2034-2045.

[34]. Ghorbani, M.A., Deo, R.C., Yaseen, Z.M., H Kashani, M., and Mohammadi, B. (2018). Pan evaporation prediction using a hybrid multilayer perceptron-firefly algorithm (MLP-FFA) model: case study in North Iran. Theoretical and applied climatology, 133 (3): 1119-1131.

[35]. Friedman, J.H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.

[36]. Friedman, J.H. (2002). Stochastic gradient boosting. Computational statistics & data analysis, 38(4): 367-378.

[37]. Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). The annals of statistics, 28 (2): 337-407.

[38]. Hastie, T., Tibshirani, R., and Friedman, J. (2001). The elements of statistical learning. Springer series in statistics. New York, NY, USA.

[39]. Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., and Chen, K. (2015). Xgboost: extreme gradient boosting. R package version 0.4-2, 1(4): 1-4.

[40]. Chen, T. and Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[41]. Zhou, J., Li, E., Wang, M., Chen, X., Shi, X., and Jiang, L. (2019). Feasibility of stochastic gradient boosting approach for evaluating seismic liquefaction potential based on SPT and CPT case histories. Journal of Performance of Constructed Facilities, 33 (3): 04019024.

[42]. Gao, W., Wang, W., Dimitrov, D., and Wang, Y. (2018a). Nano-properties analysis via fourth multiplicative ABC indicator calculating. Arabian journal of chemistry, 11 (6): 793-801.

[43]. Kennedy, J. and Eberhart, R. (1995, November). Particle swarm optimization. In Proceedings of ICNN'95-international conference on neural networks (Vol. 4, pp. 1942-1948). IEEE.

[44]. Cao, Y., Zhang, H., Li, W., Zhou, M., Zhang, Y., and Chaovalitwongse, W.A. (2018). Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions. IEEE Transactions on Evolutionary Computation, 23(4): 718-731.

[45]. Mirjalili, S., Mirjalili, S.M., and Lewis, A. (2014). Grey wolf optimizer. Advances in engineering software, 69, 46-61.

[46]. Emary, E., Yamany, W., Hassanien, A. E., and Snasel, V. (2015). Multi-objective gray-wolf optimization for attribute reduction. Procedia Computer Science, 65, 623-632.

[47]. Song, X., Tang, L., Zhao, S., Zhang, X., Li, L., Huang, J., and Cai, W. (2015). Grey wolf optimizer for parameter estimation in surface waves. Soil Dynamics and Earthquake Engineering, 75, 147-157.

[48]. El-Kenawy, E.S.M., Eid, M.M., Saber, M., and Ibrahim, A. (2020). MbGWO-SFS: Modified binary grey wolf optimizer based on stochastic fractal search for feature selection. IEEE Access, 8, 107635-107649.

[49]. Ferreira, C. (2001). Gene expression programming: a new adaptive algorithm for solving problems. arXiv preprint cs/0102027.

[50]. Faradonbeh, R.S., Armaghani, D.J., Amnieh, H.B., and Mohamad, E.T. (2018). Prediction and minimization of blast-induced fly-rock using gene expression programming and firefly algorithm. Neural Computing and Applications, 29 (6): 269-281.

[51]. Zhou, J., Li, C., Koopialipoor, M., Jahed Armaghani, D., and Thai Pham, B. (2021). Development of a new methodology for estimating the amount of PPV in surface mines based on prediction and probabilistic models (GEP-MC). International Journal of Mining, Reclamation and Environment, 35 (1): 48-68.

[52]. Faradonbeh, R. S., Hasanipanah, M., Amnieh, H. B., Armaghani, D. J., and Monjezi, M. (2018b). Development of GP and GEP models to estimate an environmental issue induced by blasting operation. Environmental monitoring and assessment, 190 (6): 1-15.

[53]. Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

[54]. Vigneau, E., Courcoux, P., Symoneaux, R., Guérin, L., and Villière, A. (2018). Random forests: A machine learning methodology to highlight the volatile organic compounds involved in olfactory perception. Food Quality and Preference, 68, 135-145.

[55]. Gao, W., Guirao, J. L., Basavanagoud, B., and Wu, J. (2018). Partial multi-dividing ontology learning algorithm. Information Sciences, 467, 35-58.

[56]. Dou, J., Yunus, A.P., Bui, D.T., Merghadi, A., Sahana, M., Zhu, Z., and Pham, B.T. (2019). Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan. Science of the total environment, 662, 332-346.

[57]. Himanshu, V.K., Roy, M.P., Mishra, A.K., Paswan, R.K., Panda, D., and Singh, P.K. (2018). Multivariate statistical analysis approach for prediction of blast-induced ground vibration. Arabian Journal of Geosciences, 11 (16): 1-11.

[58]. Shakeri, J., Shokri, B.J., and Dehghani, H. (2020). Prediction of blast-induced ground vibration using gene expression programming (GEP): artificial neural networks (ANNS), and linear multivariate regression (LMR). Archives of Mining Sciences, 65(2).

[59]. Shokri, B.J., Dehghani, H., and Shamsi, R. (2020). Predicting silver price by applying a coupled multiple linear regression (MLR) and imperialist competitive algorithm (ICA). Metaheuristic Computing and Applications, 1 (1): 1.

[60]. Bhatawdekar, R.M., Kumar, R., Sabri Sabri, M.M., Roy, B., Mohamad, E.T., Kumar, D., and Kwon, S. (2023). Estimating Fly-rock Distance Induced due to Mine Blasting by Extreme Learning Machine Coupled with an Equilibrium Optimizer. Sustainability, 15 (4): 3265.

[61]. James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

[62]. Chakraborty, S. and Bhattacharya, S. (2021). Application of XGBoost algorithm as a predictive tool in a CNC turning process. Reports in Mechanical Engineering, 2 (1): 190-201.

[63]. Sharma, M., Agrawal, H., and Choudhary, B. S. (2022). Multivariate regression and genetic programming for prediction of backbreak in open-pit blasting. Neural Computing and Applications, 34 (3): 2103-2114.

[64]. Taylor, K.E. (2001). Summarizing multiple aspects of model performance in a single diagram. Journal of Geophysical Research: Atmospheres, 106 (D7): 7183-7192.

[65]. Nikafshan Rad, H., Bakhshayeshi, I., Wan Jusoh, W.A., Tahir, M.M., and Foong, L.K. (2020). Prediction of flyrock in mine blasting: a new computational intelligence approach. Natural Resources Research, 29 (2): 609-623

یک مدل ترکیبی برای پیش‌بینی عقب‌زدگی با استفاده از الگوریتم گرادیان تقویت شده و الگوریتم‌های فراابتکاری در معدن سنگ آهن چادرملو

زهره نبوی[1]، محمد میرزهی کلاته کاظمی[1]، حسام دهقانی[2*] و پدرام اشتری[2]

1- بخش مهندسی معدن، دانشگاه تربیت مدرس تهران، تهران، ایران

2- دانشکده مهندسی معدن، دانشگاه صنعتی همدان، همدان، ایران

ارسال2023/03/09، پذیرش 2023/05/14

* نویسنده مسئول مکاتبات: maebrahimi@uk.ac.ir

چکیده:

عقب‌زدگی یکی از اثرات نامطلوب انفجار است که منجر به ناپایداری دیواره‌های معدن، رقت بالا، سقوط ماشین‌آلات و خردایش نامناسب می‌شود. بنابراین، منافع اقتصادی معدنکاری کاهش یافته و ایمنی به شدت تحت تأثیر قرار می‌گیرد. بروز عقب‌زدگی تحت تأثیر پارامترهای مختلفی مانند خواص توده سنگ، پرامترهای هندسی الگوی انفجار و خواص مواد انفجاری قرار داد. بنابراین در طی فرآیند انفجار و سایر فعالیت‌های تولیدی به منظور جلوگیری و کاهش اثرات نامطلوب عقب‌زدگی، باید به طور دقیق این پیامد پیش‌بینی شود. در این راستا، یک مدل ترکیبی از الگوریتم گرادیان تقویت شده (XGB) برای پیش‌بینی عقب‌زدگی با استفاده از الگوریتم‌های فراابتکاری بهینه‌سازی گرگ خاکستری (GWO) و بهینه‌سازی ازدحام ذرات (PSO) پیشنهاد شده است. علاوه بر این، اعتبارسنجی مدل ترکیبی با استفاده از روش‌های معمولی XGBoost، برنامه‌ریزی بیان ژن (GEP)، جنگل تصادفی (RF)، رگرسیون چندگانه خطی (LMR) و رگرسیون چندگانه غیرخطی (NLMR) انجام می‌شود. برای این منظور داده‌های به دست آمده از 90 عملیات انفجار در معدن سنگ آهن چادرملو با در نظر گرفتن پارامترهای طراحی الگوی انفجار جمع‌آوری شده است. با توجه به نتایج به‌دست‌آمده، عملکرد و سطح دقت مدل‌های ترکیبی شامل GWO-XGB (R2 = 99, RMSE = 0.01, MAE = 0.001, VAF = 0.99, a-20 = 0.98) و PSO-XGB (99, 0,01, 0,001, 0,99, 0,98) بهتر از XGBoost (97, 0.185, 0.132, 0.98, 0.95)، GEP (96, 0.233, 0.0.186, 0.967, 0.93)، RF (97, 0.21, 0.156, 0.97, 0.95)، LMR (96, 0,235, 0,181, 0,964, 0,92)، و NLMR (96, 0,229, 0,177, 0,968, 0,93). قابل ذکر است که مدل ترکیبی GWO-XGB عملکرد کلی بالاتری نسبت به مدل PSO-XGB دارد. بر اساس نتایج تجزیه و تحلیل حساسیت، ارتفاع چال و طول گلگذاری پارامترهای با بیشترین تاثیر در پیشبینی عقب‌زدگی هستند.

کلمات کلیدی: عقب‌زدگی، الگوریتم گرادیان تقویت شده، بهینه‌سازی ازدحام ذرات، بهینه‌سازی گرگ خاکستری، معدن سنگ آهن چادرملو.