Front. Struct. Civ. Eng. 2024, 18(11): 1698–1712
https://doi.org/10.1007/s11709-024-1114-y

RESEARCH ARTICLE

Interpretable gradient boosting based ensemble learning and African vultures optimization algorithm optimization for estimating deflection induced by excavation

Zenglong LIANG[a], Shan LIN[a], Miao DONG[a], Xitailang CAO[a], Hongwei GUO[a,b*], Hong ZHENG[a]

[a] *Key Laboratory of Urban Security and Disaster Engineering (Ministry of Education), Beijing University of Technology, Beijing 100124, China*

[b] *Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University (PolyU), Hong Kong 999077, China*

[*] *Corresponding author. E-mail: ghway0723@gmail.com*

ABSTRACT Intelligent construction has become an inevitable trend in the development of the construction industry. In the excavation project, using machine learning methods for early warning can improve construction efficiency and quality and reduce the chances of damage in the excavation process. An interpretable gradient boosting based ensemble learning framework enhanced by the African Vultures Optimization Algorithm (AVOA) was proposed and evaluated in estimating the diaphragm wall deflections induced by excavation. We investigated and compared the performance of machine learning models in predicting deflections induced by excavation based on a database generated by finite element simulations. First, we exploratively analyzed these data to discover the relationship between features. We used several state-of-the-art intelligent models based on gradient boosting and several simple models for model selection. The hyperparameters for all models in evaluation are optimized using AVOA, and then the optimized models are assembled into a unified framework for fairness assessment. The comprehensive evaluation results show that the AVOA-CatBoost built in this paper performs well ($RMSE = 1.84$, $MAE = 1.18$, $R^2 = 0.9993$) and cross-validation ($RMSE = 2.65 \pm 1.54$, $MAE = 1.17 \pm 0.23$, $R^2 = 0.998 \pm 0.002$). In the end, in order to improve the transparency and usefulness of the model, we constructed an interpretable model from both global and local perspectives.

KEYWORDS African vultures optimization algorithm, gradient boosting, ensemble learning, interpretable model, wall deflection prediction

1 Introduction

Currently, challenging construction projects, including super high-rise buildings, tunnels, underground superstores, and others, are developing rapidly [1–4]. In these projects, the investigation of excavation is inevitable. In the excavation process, we need to consider the damage to the surrounding buildings and their structural stability [5]. To solve these problems, it is often necessary to adopt various support structures, among which diaphragm walls are commonly used to support deep foundation pits in underground engineering [6].

Article history: Received Dec 27, 2023; Accepted Feb 15, 2024

The diaphragm wall cannot only support the surrounding soil and prevent soil landslides and the collapse of the surrounding buildings but also improve the stability of its structure. To ensure the reliability of a diaphragm wall, the deflection of the wall is often used to evaluate its stiffness and ability to resist deformation. Therefore, control and early warning of wall deflection is crucial [7].

In the excavation process, due to the complex and variable geological conditions around the foundation pit, there are phenomena such as uneven layering and significant changes in groundwater level, which will have an impact on the stability and lateral deflection of the foundation pit [8]. This makes it impossible to effectively control the lateral deflection of the foundation pit in the

excavation process, which may not only make the buildings and underground pipelines around the foundation pit affected, but may even lead to safety risks such as soil sliding, destruction of the supporting structure, and the construction risk is very high [9]. Therefore, in order to solve the problems of high construction risk, complex design, and construction of supporting structure caused by the difficulty of controlling lateral deflection of foundation pit, engineers need to adopt appropriate technology and methods, combined with the actual situation of the project, to accurately predict to ensure the safety and quality of foundation pit excavation [10].

In early studies, many researchers relied on past engineering examples and experience to analyze the deflection of diaphragm walls. In 1996, Masuda [11] studied the factors affecting the behavior of diaphragm walls by using 52 examples and proposed a procedure to predict their maximum lateral deflection. Moormann [12] analyzed the main factors affecting deep foundation excavation based on a database containing more than 530 cases of deep excavation, and his results showed that the maximum displacement varied between 0.5% and 1.0% of the excavation depth, with a mean value of 0.87%. Wang et al. [13] collected and analyzed 300 cases of lateral displacements of walls caused by deep excavation in soft soil in Shanghai, and their results showed that for diaphragm walls constructed from the bottom upward, for example, the maximum deflection ranged from 0.1%–1.0%, with a mean value of 0.4%. From these studies, it can be seen that the results obtained from examples and empirical analysis can be rough due to the differences in the specific engineering geological conditions.

With the development of computer simulation technology and the need for more accurate results, numerical simulations based on the finite element method have been widely used in the field. Kung et al. [14] used 33 cases and abundant calibrated finite element analysis results to develop the Kung–Juang–Hsiao–Hashash model, which showed excellent performance in predicting maximum deflection in soft to medium clay soils. Goh et al. [15] used a hardened soil model to study the effect of various factors on the maximum deflection of a supported excavated wall deflection using a hardened soil model and developed a simple wall deflection equation for estimating the maximum deflection of a wall with three-dimensional effects. Although this method can yield more accurate results, it tends to be more expensive, time-consuming, and difficult to apply widely.

Machine learning (ML) has recently been widely used as an efficient and highly generalizable modeling approach for various complex engineering problems [16]. In the excavation problem, many researchers have used ML models to predict wall deflection. Zhao et al. [17] used three ML algorithms, namely, back propagation neural network, long and short-term memory, and gated recurrent unit, to predict the deflection of a concrete diaphragm wall caused by excavation, and their results show that gated recurrent unit (GRU) performs the best in terms of efficiency and effectiveness. The prediction results of GRU can be helpful for the safety monitoring of the excavation site. van Nguyen et al. [5] used a Bayesian optimization of the XGBoost model to build a deep-supported excavation in the sand for the model for maximum wall deflection prediction, and their model results achieved a high degree of compliance with numerical simulation results. Zhang et al. [18] compared the performance of various ML algorithms such as XGBoost, Mars, and artificial neural network (ANN) on the problem of predicting the maximum deflection induced by excavation, and their results showed that these ML algorithms can learn the complex nonlinear relationships between features well enough to achieve the prediction.

However, ML models often consist of multiple parameters, so it becomes essential to find parameters that can obtain the globally optimal solution [19]. To fully utilize the performance of ML models and find the optimal parameters, a variety of metaheuristic algorithms have been developed and applied. Qi and Tang [20] used the Firefly algorithm to optimize the parameters of a variety of ML models, allowing these models to achieve excellent performance. Yong et al. [21] used the finite element method, multiple ML algorithms, and multiple meta-heuristic optimization algorithms to build MLP-HHO and MLP-WO intelligent models that can accurately predict the deflection of diaphragm walls. Grey wolf optimizer (GWO) was utilized by Shariati et al. [22] to optimize the parameters of the extreme learning machine (ELM). The results demonstrated that GWO can effectively improve the performance of ELM, and a comparative analysis of the built GWO-ELM revealed that it outperformed other models.

Even though ML models are very accurate and generalizable, their predictions are frequently hard to interpret because the models are constructed as black-box models [23]. Because of this, it is challenging to apply in practical engineering, necessitating an explanation of the models created [24]. Sensitivity analysis [25] is a method to study the effect of input parameters on the output results, which can be used in ML to check which parameters the model output is more sensitive to in order to explain the model. SHapley Additive exPlanations (SHAP) [26] is another method of model explanation. The SHAP value can be used to analyze the specificity of each person's output process, which helps us understand the model in addition to determining the contribution of each feature. ML models are increasingly being explained using these two approaches.

In summary, in order to better predict the wall deflection generated during excavation and to reduce

unnecessary damage and losses, combining the advantages of various methods, the main work of this paper can be divided into four parts: 1) exploratory analysis of the data; 2) modeling using the database obtained from finite element simulations, meta-heuristic optimization algorithms, and state-of-the-art ML algorithms; 3) comprehensive evaluation of the models in comparison; 4) explanation of the best-performing model. Figure 1 is the schematic graph of the whole data flow in predicting the diaphragm wall deflections induced by excavation.

2 Data visualization and analysis

In this chapter, we first describe the source of the data as well as the engineering background, and then, we perform Exploratory data analysis using visualization and other methods. Finally, we measure the correlation between the data features in this paper using the mutual information metric, taking into account the feasibility of ML modeling.

2.1 Excavation background

Zhang et al. [18] in order to investigate the effect of each parameter during the excavation process, PLAXIS 8.5 software was used to build the excavation model and database, which is shown in Fig. 2. For modeling, Hardening small-strain model was used considering small strain effect. The groundwater table was assumed to be the surface with hydrostatic pore pressure, and the soil was subjected to undrained shear during excavation. The studied parameters are excavation width B, the thickness of soft clay layer T, excavation depth H, relative shear strength ratio $\frac{C_u}{\sigma'_v}$, relative soil stiffness ratio $\frac{E_{50}}{C_u}$, system stiffness $\ln(EI/\gamma_w h_{avg}^4)$ and soil unit weight γ, respectively. In Table 1, the value ranges of these parameters are described in detail.

2.2 Exploratory data analysis (EDA)

EDA [27] is a method of exploring the characteristics, structure, and underlying patterns of existing data with as few a priori assumptions as possible through statistical and visualization methods in order to understand the data better. This approach emphasizes letting the data itself speak for itself, which can reveal patterns and trends in the data more realistically and lay the foundation for further statistical inference and modeling. Considering the discrete nature of the data features, as shown in Fig. 3, we use the box plot and the normal distribution of observation for our analysis, where the horizontal coordinate is the range of values of the feature and the vertical coordinate is the corresponding wall deflection

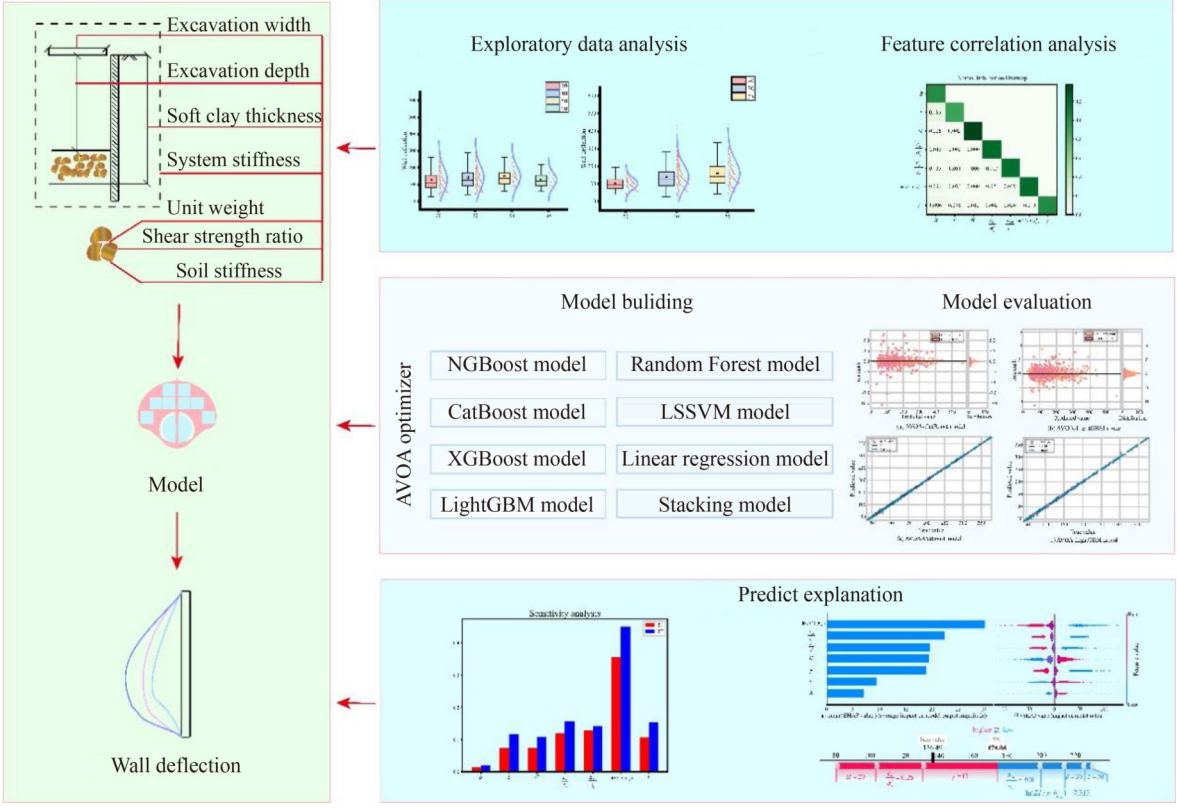

Fig. 1 Schematic graph of data flow in the whole ML model for excavation.

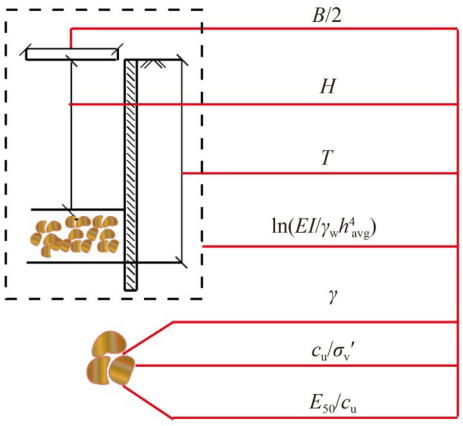

Fig. 2 Diaphragm wall deflection induced by braced excavations.

Table 1 Settings for each parameter

Parameter	Value
B (m)	30, 40, 50, 60
T (m)	20, 30, 35
H (m)	11, 14, 17, 20
$\ln(EI/\gamma_w h_{avg}^4)$	6.097, 7.313, 8.176, 8.846
γ (kN/m)	15, 17, 19
$\frac{E_{50}}{C_u}$	100, 200, 300, 400
$\frac{C_u}{\sigma_v'}$	0.21, 0.25, 0.29, 0.34

value.

As can be seen in Fig. 3, some of these parameters have a clear correlation with the predicted values. The thickness of the soft clay layer and excavation depth show a clear positive correlation with wall deflection. The relative soil stiffness ratio, system stiffness, and unit weight show a clear negative correlation with wall deflection. The correlation of excavation width and relative shear strength ratio with wall deflection is more complicated, with inflection points in both their multi-box plots, which means that after reaching a particular value, the feature's correlation with wall deflection changes. These rules in the data itself will not only provide a greater understanding of the effects of the individual parameters on wall deflection. Still, they will also be used in the model explanation section of the paper to corroborate the reliability of the model developed.

2.3 Feature correlation analysis

Mutual information [28] is a measure that evaluates the correlation between variables. Higher mutual information between variables implies the knowledge of which one of the variables can reduce the uncertainty about another random variable even more. Unlike the correlation coefficient, mutual information takes into account not only the degree of similarity between the values of the random variables, but also the degree of similarity between their joint distribution $p(X,Y)$ and the product of the marginal distributions of the decompositions, $p(X)\,p(Y)$. Thus, when the relationship between variables is not linear, mutual information can help to find any correlation between variables, including linear and nonlinear correlations.

Considering the nonlinearity of the features in this paper, we use mutual information to analyze their correlation. Figure 4 contains the mutual information of these features. As can be seen from the figure, the mutual information values of all these features are small, i.e., the correlation between them is low. In ML, the correlation between the features should be small. The data in this paper meets the requirements of ML modeling. In the next section, we will use this data to build a model to predict the wall deflection caused by excavation works.

3 Methodology

3.1 African vultures optimization algorithm

AVOA [29] is a metaheuristic algorithm inspired by the lifestyle of African vultures, which was proposed by Abdollahzadeh in 2021. This algorithm is designed to be an efficient global optimization algorithm with strong convergence and search capabilities that can help solve complex optimization problems. Compared with other optimization algorithms, AVOA, by combining vulture foraging behavior, roulette algorithm, and levy flight, is not only conducive to avoiding falling into local optimal solutions and having a higher global search capability but also can dynamically adjust the search strategy and parameters according to the results of the search process, to obtain the optimal solution more quickly. In complex ML algorithms such as ensemble learning, there is often the disadvantage of complex parameter tuning. Therefore, using the AVOA algorithm to optimize the parameters of the model can significantly reduce the time consumed by tuning under the premise of ensuring that the global optimal solution is found. Following the behavioral patterns of vultures, the primary process of the algorithm can be divided into three parts: initializing the vulture population and selecting the direction of update, calculating the starvation rate of the vultures, and determining whether the vultures enter the exploration or exploitation phase based on the starvation rate.

1) Initialize the vulture population and select the direction of update: randomly generate n vultures within a defined range and calculate their fitness. Determine the optimal and sub-optimal solutions among the n vultures based on each vulture's fitness and the roulette rule (as Eq. (1)). Where the optimal solution is the first group of vultures, the sub-optimal solution is the second group of

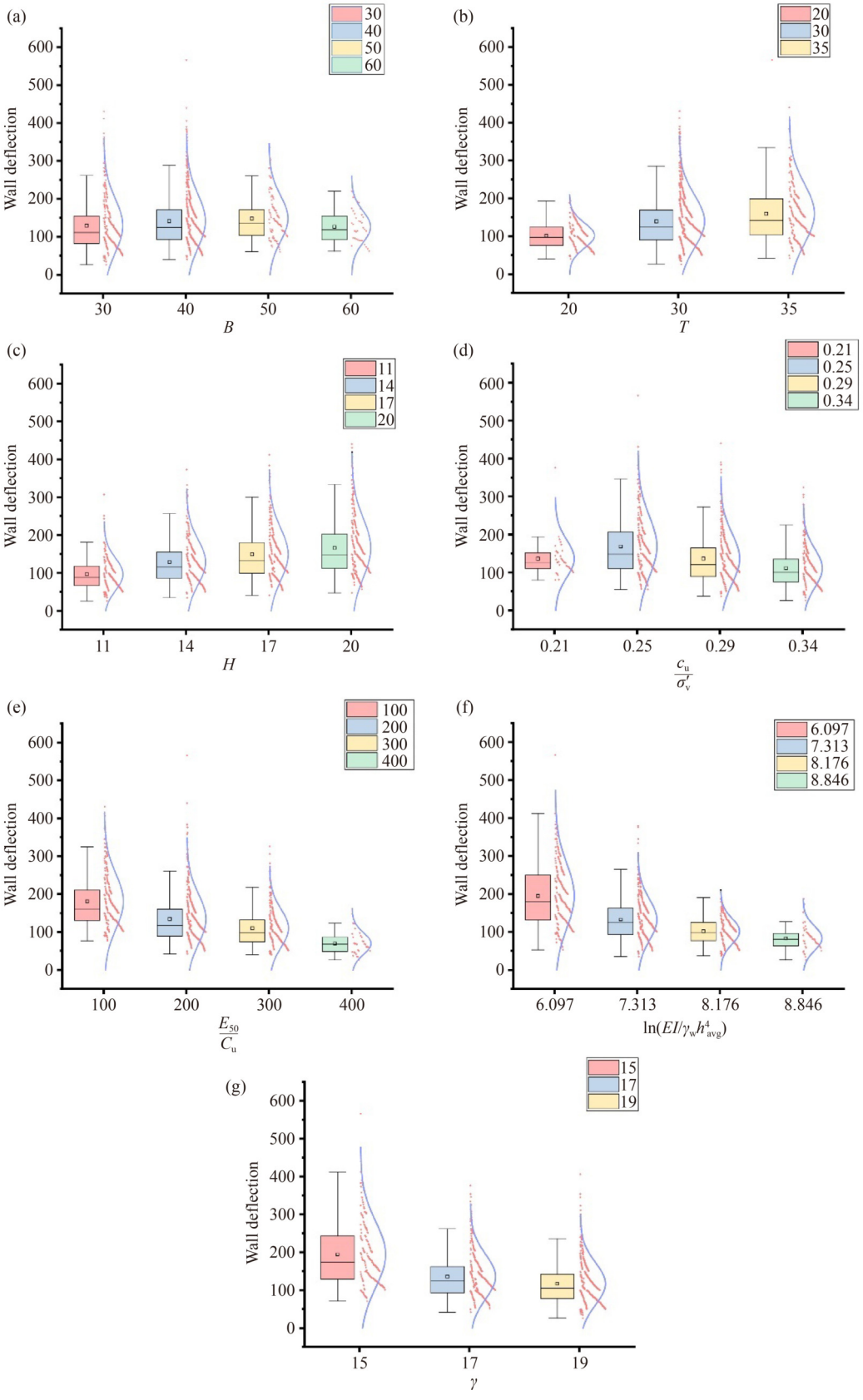

Fig. 3 The box plot and the normal distribution of observation,distinguished by features: (a) B; (b) T; (c) H; (d) $\frac{C_u}{\sigma'_v}$; (e) $\frac{E_{50}}{C_u}$; (f) $\ln(EI/\gamma_w h^4_{avg})$; (g) γ.

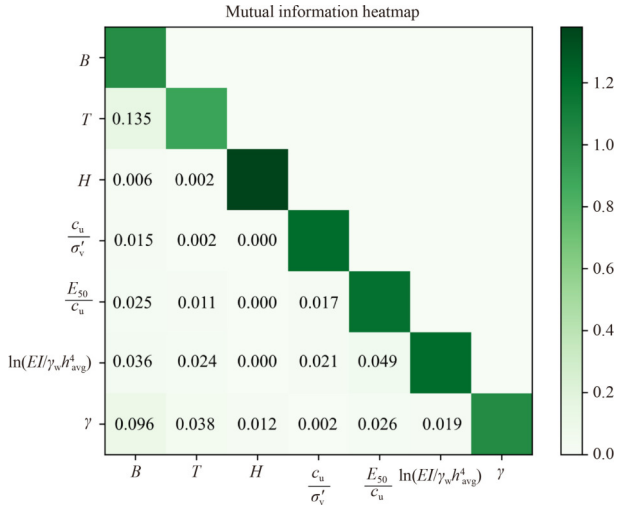

Fig. 4 Heatmap for mutual information.

vultures, and the others are the third group. The position update of the third group of vultures is realized by making the vultures of that group move closer to the first or second group of vultures using Eq. (2):

$$D(i)=\begin{cases}\text{Optimal solution}, & p_i=\alpha,\\ \text{Suboptimal solution}, & p_i=\beta,\end{cases} \tag{1}$$

$$p_t=\frac{F_t}{\sum_{t=1}^{n}F_i}. \tag{2}$$

In Eq. (1), i represents that the algorithm is currently at ith iteration, α and β represent the probability that the vultures in the third group choose the optimal and suboptimal solution as the way forward, respectively and $\alpha+\beta=1$. In Eq. (2), F_t is the fitness of the tth vulture, evaluated according to the problem being solved.

2) Calculating the starvation rate of the vultures: the search range of vultures is closely related to their hunger; when vultures are well fed, they have more energy, which allows them to go further into the field to feed. But if they are hungry, they will forage near the optimal solution. Based on this behavioral pattern, modeling was performed using Eq. (3):

$$E=(2\times R_1+1)\times z\times\left(1-\frac{i}{i_{\max}}\right)+u, \tag{3}$$

where E means the energy of the vulture, R_1 has a random value between 0 and 1, $i_{\max}$ is the number of iteration parameters we have set, i is the number of times the formula has been iterated so far, z is a random number between −1 and 1, u is a method that allows the optimization algorithm to escape from the local optimal point. When solving complex optimization problems, the optimization algorithm tends to converge prematurely at the local optimum point, and the method u has been used for this problem, the value of which can be calculated by Eq. (4):

$$u=j\times\left(\sin^w\left(\frac{\pi}{2}\times\frac{i}{i_{\max}}\right)+\cos\left(\frac{\pi}{2}\times\frac{i}{i_{\max}}\right)-1\right), \tag{4}$$

where j is a random number between −2 and 2, w is a fixed parameter, and its value is set prior to the optimization. The value of w is related to the final optimization stages. The larger the value of w, the higher the probability of entering the exploration phase in the final optimization stage.

3) Exploration or exploitation: behavioral patterns of vultures while foraging are correlated with the energy of the vulture, which gradually transitions from an exploratory to an exploitative phase as the vulture's energy decreases. Based on the vulture energy (E) calculated in Part 2, the transitions of vultures can be described as Eq. (5):

$$\begin{cases}\text{Exploration}, & |E|>1,\\ \text{Exploitation}\begin{cases}\text{Type 1}, & 0.5<|E|<1,\\ \text{Type 2}, & |E|<0.5.\end{cases}\end{cases} \tag{5}$$

a) Exploration: vultures usually examine their environment while foraging for food. They will fly high above the ground and look down on the surrounding terrain for possible prey or food sources. In AVOA, vultures choose two different inspection strategies through a fixed parameter P_1 set before the optimization's beginning, and a random parameter R_{P1}, which can be described as Eq. (6):

$$P(i+1)=\begin{cases}\text{Strategy 1}, & P_1\geqslant R_{P1},\\ \text{Strategy 2}, & P_1<R_{P1}.\end{cases} \tag{6}$$

In Eq. (6), the value of P_1 and R_{P1} is from 0 to 1, $P(i+1)$ refers to the position of the vulture at $(i+1)$th iteration. Strategy 1 is for the vultures to randomly search for food around the best solution chosen in Eq. (1), represented by Eq. (7):

$$\text{Strategy 1}: D(i)-H(i)\times E, \tag{7}$$

where E and $D(i)$ are calculated by Eqs. (3) and (1), respectively, $H(i)=|2\times r_1\times D(i)-P(i)|$, where r_1 is a random number changed in every iteration and its value between 0 and 1, $2\times r_1$ is used as a motion vector to increase the probability of vultures' random movement.

Strategy 2 is to allow vultures to randomly search for food over a wider area, which can be described by Eq. (8):

$$\text{Strategy 2}: D(i)-E+r_2\times((ub-lb)\times r_3+lb), \tag{8}$$

where r_2 and r_3 are a random number between 0 and 1, r_3 is used to increase the randomness of vulture locations. ub and lb are the upper and lower bounds of the solution

space, respectively.

b) Exploitation: From Eq. (5), it can be seen that the exploitation stage can be categorized into two types depending on the magnitude of E. In type 1, as determined similarly in Eq. (6), vultures can adopt siege-fight (Strategy 3) or rotating flight (Strategy 4). The adoption mechanism is shown in Eq. (9):

$$P(i+1)=\begin{cases}\text{Strategy 3}, & P_2 \geqslant R_{\mathrm{P2}},\\ \text{Strategy 4}, & P_2 < R_{\mathrm{P2}}.\end{cases} \tag{9}$$

Strategy 3 refers to the fact that when vultures in this state congregate on a single food source, it may cause severe conflict over food. Equations (10) and (11) are used to model this behavior:

$$\text{Strategy 3}: H(i)\times(E+r_4)-d(t), \tag{10}$$

$$d(t)=D(i)-P(i). \tag{11}$$

In Eqs. (10) and (11), $H(i)$ and $D(i)$ are obtained through Eqs. (7), (3), and (1), respectively. Equation (11) is to calculate the distance between the current vulture and $D(i)$.

Strategy 4 represents the rotational flight of a vulture, using Eq. (12) to model the behavior. S_1 and S_2 can be calculated by Eqs. (13) and (14):

$$\text{Strategy 4}: D(i)-(S_1+S_2), \tag{12}$$

$$S_1=D(i)\times\left(\frac{r_5\times P(i)}{2\pi}\right)\times\cos(P(i)), \tag{13}$$

$$S_2=D(i)\times\left(\frac{r_6\times P(i)}{2\pi}\right)\times\sin(P(i)). \tag{14}$$

In type 2, as determined similarly in Eq. (9), vultures can adopt accumulation (Strategy 5) or aggressive siege-fight (Strategy 6). The adoption mechanism is shown in Eq. (15):

$$P(i+1)=\begin{cases}\text{Strategy 5}, & P_3 \geqslant R_{\mathrm{P3}},\\ \text{Strategy 6}, & P_3 < R_{\mathrm{P3}}.\end{cases} \tag{15}$$

Strategy 5 indicates that at this phase, several species of vultures congregate on a single food source, and this behavior is modeled using Eqs. (16)–(18):

$$A_1=\text{Optimal solution}(i)-\frac{\text{Optimal solution}(i)\times P(i)}{\text{Optimal solution}(i)-P(i)^2}\times E, \tag{16}$$

$$A_2=\text{Suboptimal solution}(i)-\frac{\text{Suboptimal solution}(i)\times P(i)}{\text{Suboptimal solution}(i)-P(i)^2}\times E, \tag{17}$$

$$\text{Strategy 5}: \frac{A_1+A_2}{2}. \tag{18}$$

Strategy 6 refers to the aggressive food competition behavior of vultures and can be described by Eqs. (19)–(21):

$$\text{Strategy 6}: D(i)-|d(t)|\times E\times LF(dim), \tag{19}$$

$$LF(x)=0.01\times\frac{\mu\times\sigma}{|v|^{\beta^{-1}}}, \tag{20}$$

$$\sigma=\left(\frac{\Gamma(1+\beta)\times\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)\times\beta\times2^{\frac{\beta-1}{2}}}\right)^{\beta^{-1}}. \tag{21}$$

In Eq. (19), $d(t)$, $D(i)$, and E are obtained through Eqs. (11), (1), and (3), respectively. $LF(dim)$ is the search path of Levy flight, where dim is the dimension of solution space. Equations (20) and (21) is the step length formula of Levy flight, where v are random values obeying a standard normal distribution, β is a fixed parameter whose default setting is 1.5.

Above is the introduction of the overall process of the AVOA algorithm. In this paper, we use the algorithm to optimize the parameter models of the model. The solution space varies according to the parameters often debugged in each ML model, mainly including the number of iterations, learning rate, etc. The fitness, i.e., the $RMSE$ predicted by the ML model in the validation set, where $RMSE$ can be calculated as in Eq. (22):

$$RMSE=\sqrt{\frac{\sum_{c=1}^{N}(Y_c-f(x_c))^2}{N}}, \tag{22}$$

where N is the total number of samples in the validation set, c is the cth sample in the validation set, Y_c and $f(x_c)$ are the actual value and predicted value by the model of the cth sample, respectively.

3.2 Ensemble machine learning-based gradient boosting algorithm

The Gradient Boosting algorithm (GBA) [30] is an ensemble learning method that iteratively trains a series of weak regression trees and then weights them to combine them into a strong regression tree. In each iteration, a new regression tree is obtained by fitting the residuals between the prediction results of the previous round and the actual labels, and the final prediction result is the weighted sum of the results of all regression trees. The core idea of the algorithm is to gradually improve the predictive ability of the model by continuously reducing

the residuals, and it shows high accuracy and robustness in dealing with regression problems. Figure 5 is the schematic diagram of the GBA algorithm flow.

3.2.1 Light gradient boosting algorithm (LightGBM)

LightGBM [31] is an efficient ensemble learning model for implementing the GBA algorithm, which has faster training speed, lower memory consumption and better accuracy by improving XGBoost, which solves the problem of excessive time consumption when facing massive engineering data. To reduce its time consumption, the improvement can be divided into three main parts: 1) reduce the use of samples; 2) reduce the use of features; 3) using the Histogram Algorithm.

1) To ensure the accuracy of the model, LightGBM uses Gradient-based one-sided sampling, which significantly reduces the size of the training data. The principle of the method is as follows. a) Samples are sorted using global gradient information and then divided according to the magnitude of the gradient. The samples with larger absolute values of the gradient are divided into subset A, while the samples with smaller absolute values of the gradient are divided into another subset B. b) Obtain the subset C from B by random downsample and train using subset D, where $D = A \bigcup C$.

2) To process high-latitude data more efficiently, LightGBM uses Exclusive Feature Bundling (EFB) for data features.EFB consists of Greedy Bundling (GB) and Merge Exclusive Features (MEF), where GB solves the problem of how to discover mutually exclusive feature pairs and MEF solves the problem of how to bundle mutually exclusive feature pairs.

3) After using these two means of processing the data, LightGBM also uses a histogram algorithm to find the optimal segmentation point of feature f at point p ($\tilde{V}_{\mathrm{f}}(p)$). Setting the global gradient obtained above as g_i, and the numbers of data in the partitioned subset A, C as a, c, then $\tilde{V}_{\mathrm{f}}(p)$ can be calculated by Eq. (23).

$$\tilde{V}_{\mathrm{f}}(p) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \in A_{\mathrm{l}}} g_i + \frac{1-a}{c}\sum_{x_i \in C_{\mathrm{l}}} g_i\right)^2}{n_i^{\mathrm{f}}(p)} + \frac{\left(\sum_{x_i \in A_{\mathrm{r}}} g_i + \frac{1-a}{c}\sum_{x_i \in C_{\mathrm{r}}} g_i\right)^2}{n_{\mathrm{r}}^{\mathrm{f}}(p)}\right). \tag{23}$$

3.2.2 CatBoost

CatBoost [32] is another engineering implementation of the GBA, which is widely used for its ability to handle

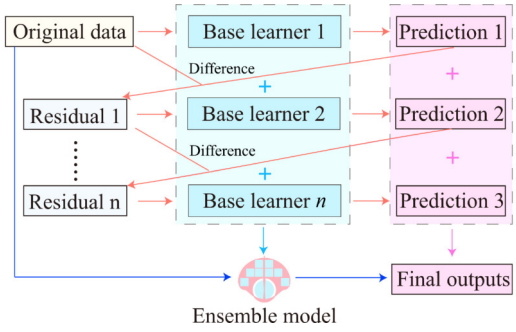

Fig. 5 Schematic diagram of GBA algorithm flow.

category-based features in data efficiently. Compared to LightGBM, CatBoost adds two methods, ordered target statistics (TS) and ordered boosting, which allows CatBoost to not only deal with categorical features but also effectively solve the gradient bias problem. In this section, we focus on these two algorithms: Ordered TS and Ordered boosting.

1) Ordered TS: TS refers to statistical data for a specific goal or indicator, and using TS to process categorical feature data, which transforms raw features into some relevant statistic, is often effective and the loss of information is slight. Ordered TS is a more effective TS that avoids the conditional shift of Greedy TS, which is inspired by online learning and relies on the ordering principle. In Ordered TS, the kth feature of the ith data x_i^k can be replaced by Eq. (24).

$$\hat{x}_i^k = \frac{\sum_{x_j \in D_k} 1_{[x_j^j = x_k^j]} \cdot y_j + \alpha p}{\sum_{x_j \in D_k} 1_{[x_j^j = x_k^j]} + \alpha}. \tag{24}$$

In Eq. (24), $1_{[A=\mathrm{B}]}$ is an indicator function, $1_{[A=\mathrm{B}]} = 1$ when $A = B$, otherwise, $1_{[A=\mathrm{B}]} = 0$. p is a prior value and α is the set weight parameter. Since this paper solves a regression problem, the value of p is equal to the mean of the label values in the training set.

2) Ordered boosting: to tackle gradient bias, CatBoost uses this method. The core of the method is that during the modeling process, the model M_i is constructed through the first i data only. Moreover, in each model, the residual value of the jth sample is obtained through the model M_{j-1}.

3.3 Model building

The modeling process can be divided into three main parts. 1) Split the data set into training, validation, and test data sets according to 6:2:2. 2) For parameter optimization, use the AVOA algorithm as well as the training and validation sets, where the training set is used for model training, and the validation set is used for the fitness assessment of the AVOA algorithm. 3) Use the training and validation sets as the final training set and the best parameters for the AVOA algorithm. The best

parameters found are used as model parameters to build the model.

In addition to the two models presented above, we have built NGBoost [33], XGBoost [34], Random Forests [35], Least Squares Support Vector Machine (LSSVM) [36], Linear Regression (LR), and stacking model [37] and optimized their parameters using AVOA optimization. In the stacking process, CatBoost, XGBoost, and LightGBM are base classifiers, and LR is the meta-classifier. In the next chapter, we will evaluate these models.

4 Model evaluation

4.1 Evaluation in test data set

Following the 6:2:2 data set division pattern described above, we first tested the model's performance on the test data set. Table 2 shows the performance of all the models, and the AVOA-CatBoost model performs best. From the model perspective, complex models such as LSSVM perform far better than the simpler baseline model LR. From the perspective of ensemble models, the gradient boosting-based models detailed in this paper perform much better than the RF models based on the bagging integration method.

Fig. 6 is the residual plot of some of the top-performing models as well as the baseline model AVOA-LR. The plot consists of two parts: the scatterplot on the left side shows in detail the prediction biases of the model in the training set as well as the test data set, and the histogram on the right side shows the distribution of these biases. As can be seen from the plot, the AVOA-CatBoost, AVOA-LightGBM, and AVOA-SRR models share a similar lower bound of −10 on the lower bias, but on the upper bias, AVOA-CatBoost has a lower upper bound of only a few bias points around 5. In addition to this, it can be seen from this figure that the distribution of AVOA-CatBoost deviations is closer to the 0 line, which means that its predictions are more stable.

In addition, Fig. 7 is the prediction error plot of the four models above, except for AVOA-LR, which is poorly fitted. The fitting effect of the other models is perfect, and their identity and best-fit line overlap and R^2 reaches

Table 2 Performance of the predictive models on the test data set

Model	*RMSE*	*MAE*	R^2
AVOA-NGBoost	4.92	3.40	0.9953
AVOA-CatBoost	1.84	1.18	0.9993
AVOA-XGBoost	4.92	3.44	0.9953
AVOA-LightGBM	2.43	1.68	0.9989
AVOA-RF	15.17	11.88	0.9551
AVOA-LSSVM	3.97	2.80	0.9969
AVOA-LR	27.56	20.83	0.8520
AVOA-Stacking	2.11	1.39	0.9991

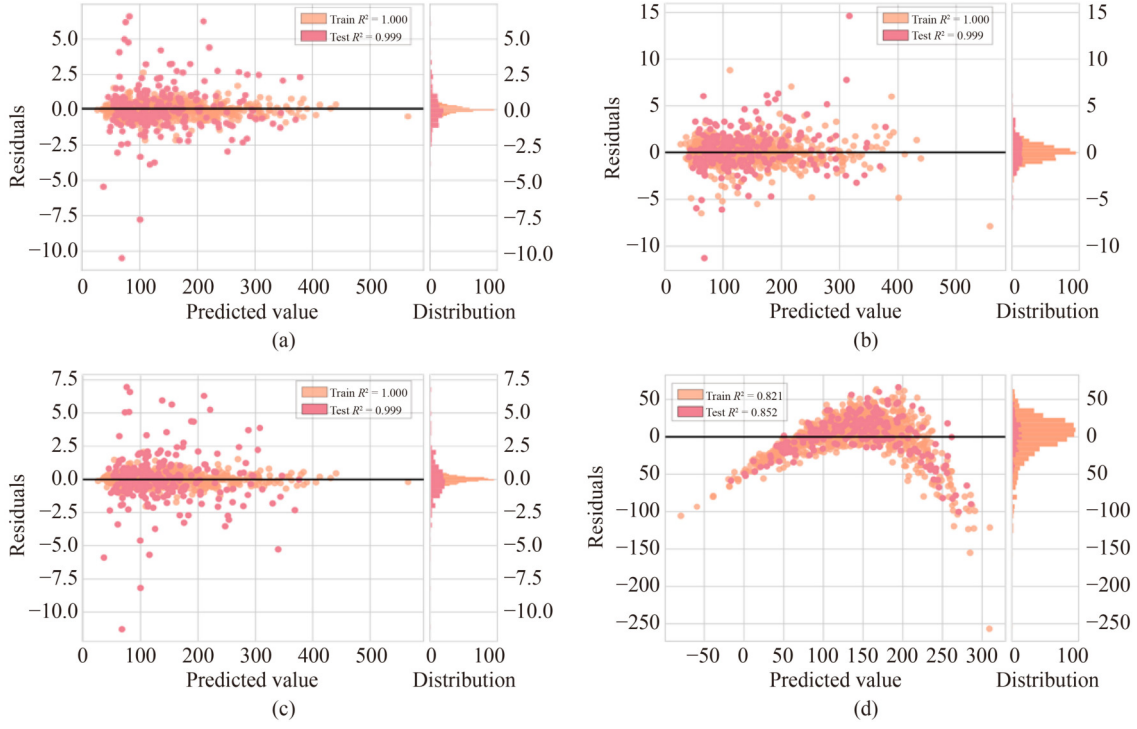

Fig. 6 Residuals plots for predictive models with AVOA optimization. Differentiation by model: (a) AVOA-CatBoost model; (b) AVOA-LightGBM model; (c) AVOA-SRR model; (d) AVOR-LR model.

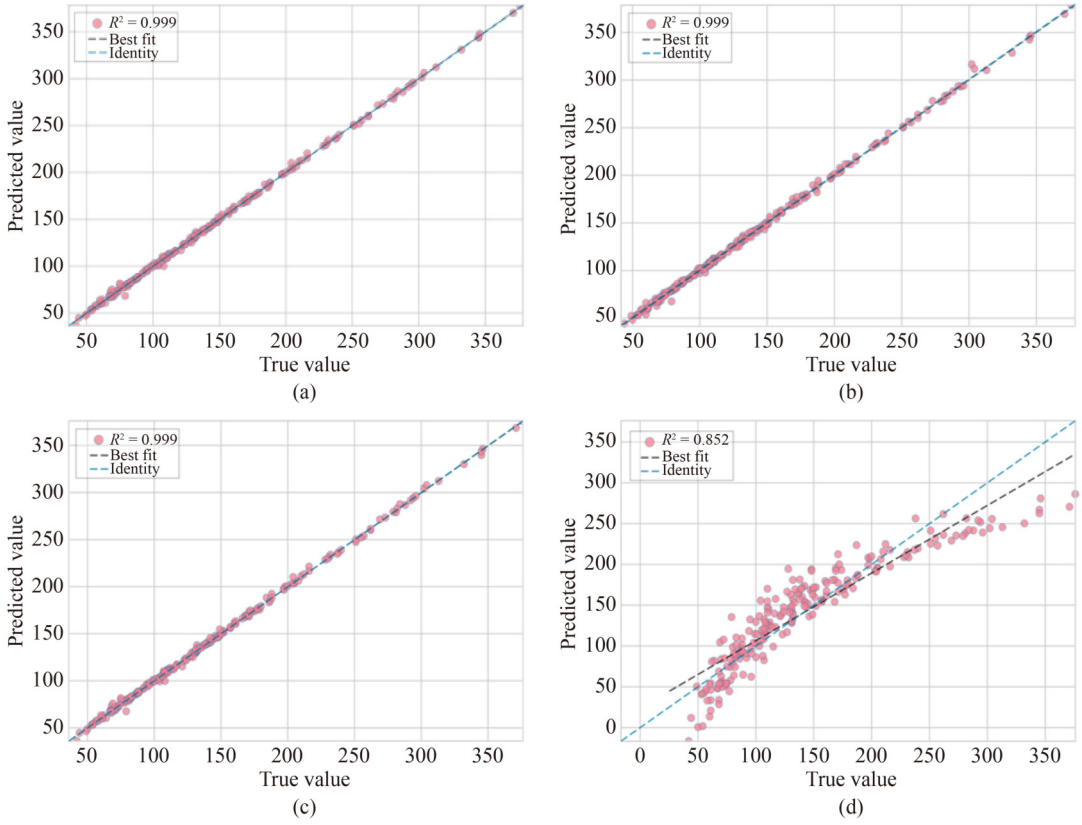

Fig. 7 Regression plots for predictive models with AVOA optimization. Differentiation by model: (a) AVOA-CatBoost model; (b) AVOA-LightGBM model; (c) AVOA-SRR model; (d) AVOR-LR model.

0.999. Therefore, in terms of performance on the test data set, the AVOA-CatBoost built in this paper performs the best.

In addition to the pairs of local models, we also compared them with the models of others using the same data set, and Table 3 records the model performance yielded by others on the same data set. From the perspective of the optimization algorithm, the AVOA-XGBoost constructed in this paper performs much better than the previously constructed XGBoost model. From the perspective of overall model performance, the performance of AVOA-CatBoost, AVOA-LightGBM, AVOA-LSSVM, and AVOA-Stacking in this paper is much better than that of previous studies.

Table 3 Model performance yielded from past literature

Model	*RMSE*	*MAE*	R^2	source
HHO-MLP	4.51	3.23	0.996	[21]
WO-MLP	4.74	3.44	0.996	
XGBoost	7.90	None	0.990	
MARS	11.10	None	0.970	[18]
ANN	11.73	None	0.970	
SVR	17.40	None	0.940	

4.2 *N*-repeated *K*-fold cross-validation

K-fold cross-validation [38,39] is a statistical technique for evaluating the performance of a model by dividing the data set into K folds, where $(K-1)$ folds are used as the training set and the remaining one is used as the test data set, and repeating the training and testing of the model several times to get a more accurate assessment of the model performance. *N*-repeated cross-validation [40] is based on cross-validation with N repetitions, where the data set is re-randomized and cross-validated for each repetition. This can reduce the impact of randomness on model performance evaluation and get more stable and reliable evaluation results. In this paper, we use repeated 10-fold cross-validation as the evaluation method and *RMSE*, *MAE*, and R^2 as the evaluation criteria. Table 4 shows the scoring results for all models of each repetition.

5 Model explanation

In this chapter, we use two methods to explain the built AVOA-CatBoost model. Sensitivity analysis not only allows us to understand which changes in features have a more significant impact on the model output, but also allows us to understand whether these features

Table 4 10-repeated 10-fold cross-validation (*RMSE*)

Model	10-repeated 10-fold cross-validation (*RMSE*)										Mean and variance		
	1	2	3	4	5	6	7	8	9	10	*RMSE*	*MAE*	R^2
AVOA-NGBoost	5.11	5.35	5.36	5.38	5.32	5.39	5.56	4.88	5.31	5.07	5.27 ± 1.52	3.10 ± 0.39	0.994 ± 0.003
AVOA-CatBoost	2.61	2.66	2.76	2.58	2.58	2.72	2.68	2.59	2.73	2.57	2.65 ± 1.54	1.17 ± 0.23	0.998 ± 0.002
AVOA-XGBoost	5.30	5.00	4.96	5.16	5.38	4.98	5.41	4.71	5.69	4.63	5.12 ± 2.17	2.80 ± 0.41	0.994 ± 0.006
AVOA-LightGBM	3.19	3.17	3.20	3.14	3.28	3.14	3.34	2.98	3.02	3.14	3.16 ± 1.31	1.68 ± 0.25	0.998 ± 0.002
AVOA-RF	15.86	15.59	15.89	15.58	15.76	15.46	16.11	15.53	15.65	15.63	15.71 ± 2.19	11.31 ± 1.05	0.948 ± 0.012
AVOA-LSSVM	4.53	4.56	4.57	4.49	4.55	4.48	4.57	4.49	4.50	4.50	4.52 ± 1.60	2.77 ± 0.37	0.995 ± 0.003
AVOA-LR	28.78	28.78	28.82	28.77	28.85	28.75	28.82	28.73	28.74	28.71	28.77 ± 1.97	20.62 ± 1.45	0.827 ± 0.027
AVOA-Stacking	2.88	2.92	3.00	2.82	2.98	2.87	2.99	2.76	3.09	2.74	2.90 ± 1.73	1.31 ± 0.26	0.998 ± 0.003

individually have an impact on the model output or whether their interactions affect the model output. However, this method does not capture the correlation between features and model output. To further explain the model, we also used the Tree SHAP analysis method to explain the model from multiple perspectives.

5.1 Sensitivity analysis

In ML, sensitivity analysis [41] is a method for assessing the sensitivity of ML model output results to changes in model input parameters. The method can be applied not only to feature engineering to identify the most critical parameters affecting the model but also to evaluate the reliability of the model by the consistency of the model's sensitivity with that of the actual project. In this section, the Sobol sensitivity analysis method is used, by which the first-order ($S1$) and total-order (ST) variance contributions of the parameters to the output are obtained. $S1$ indicates the extent to which an input variable contributes to changes in the system output, reflecting the sensitivity of the system output to a single input variable [42]. ST indicates the extent to which the interaction of an input variable with other input variables contributes to the system output, reflecting the extent to which the interaction of the input variables influences the system output. $S1$ and ST can be calculated by Eqs. (25) and (26):

$$S1_j = \frac{Var_{X_j}\left(E_{X_{\sim j}}\left(Y\middle|X_j\right)\right)}{Var(Y)}, \tag{25}$$

$$ST_j = \frac{E\left(Var_{X_j}\left(E_{X_{\sim j}}(Y|X_j)\right)\right)}{Var(Y)}, \tag{26}$$

where j is the jth feature, $\sim j$ is all the other characteristics except j.

Based on the performance of all models in the model evaluation, we performed sensitivity analysis on the best-performing AVOA-CatBoost model, which is illustrated in Fig. 8 with $S1$ and ST for each feature. As can be seen in Fig. 8, $\ln(EI/\gamma_w h_{avg}^4)$ has the highest $S1$ and ST, which means that its changes have the most significant impact on the model output, and the rest of the features, except for B, have ST above 0.01, which means that they have more inter-feature interactions in addition to a single impact on the model output. This is one of the reasons why the ensemble learning model developed in this paper is better, which tends to learn the nonlinear relationship between features better.

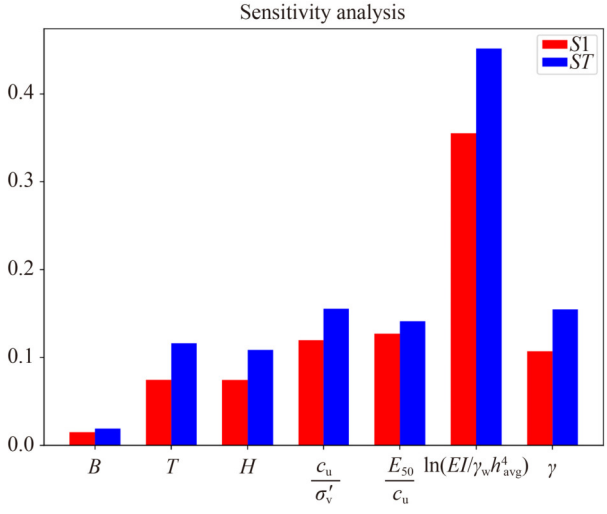

Fig. 8 $S1$ and ST of the AVOA-CatBoost model.

5.2 Tree SHapley Additive exPlanations analysis

SHAP [43] is an explanation method based on game theory, which in the field of ML can explain the prediction results of the model in an intuitive way to help us understand the decision-making process of the model. This method can be used to quantitatively measure the role of each feature on the model output by calculating the additivity contribution of each feature to the model output, in order to explain the model. Define C_i as the contribution value of the ith feature, and its value can be calculated by Eq. (27):

$$C_i = \sum_{N \subseteq M \setminus \{i\}} \frac{|D_N|(D_M - D_N - 1)!}{D_M!} [Y(N \cup \{i\}) - Y(N)], \quad (27)$$

where $M \setminus \{i\}$ is the set of all the remaining features in the training set except for feature i, N is a subset of it, D_N is the dimension of N, and $Y(N \cup \{i\})$ is the prediction result predicted by the model based on the features in N and feature i.

Tree SHAP is one branch of SHAP methods fitted to the tree-based ML model, which can take advantage of the structural characteristics of the tree model to give stable and reliable explanation results efficiently. The AVOA-CatBoost model developed in this paper is essentially a gradient-boosting tree model, so in this section, we will use Tree SHAP to explain the model.

Figure 9 demonstrates the global explanation plot of Tree SHAP for the AVOA-CatBoost model, which has two parts: the left side (Fig. 9) is the sorting plot according to the sum of SHAP values of each feature, and the right side (Fig. 9) is the scatter SHAP plot of these data points. From Fig. 9, we can see that the results are the same as those in the sensitivity analysis. The highest ranking is also $\ln(EI/\gamma_w h_{avg}^4)$, and the lowest is also B. Besides, the scatter plot on the right side also indicates conducive information about the correlation between the features and the output results. From Fig. 9, we can see that the larger these values of $\ln(EI/\gamma_w h_{avg}^4)$, $\frac{E_{50}}{c_u}$, $\frac{c_u}{\sigma_v'}$, and γ are, the smaller their SHAP values are, which present a significant negative correlation to the model output; H, T, and B, on the contrary, the larger their values are, the smaller their SHAP values are, which present a significant negative correlation to the model output.

In addition to this, Tree SHAP can explain specifically for any of the samples. Figure 10 shows the local explanation analysis for the 468th sample, which was test data set data at the time of segmentation. From this figure, it can be seen that each feature makes a certain contribution and the specific details can be seen in Table 5. The output process of the model can be explained as follows: Based on the training set data, the base value = 136.49 is obtained, which is then added with the contribution brought by each feature to get the final output value of 174.06.

In summary, the combination of the two explanation methods, sensitivity analysis and Tree SHAP, has led to a

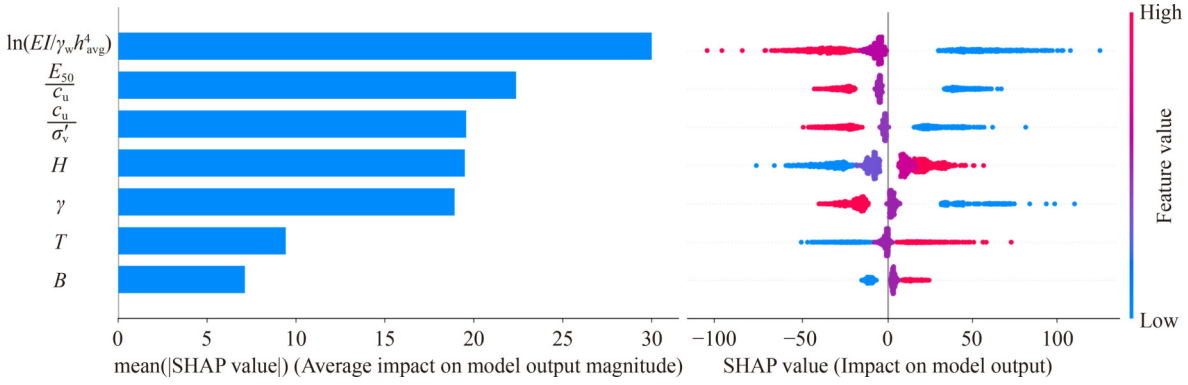

Fig. 9 Global Tree SHAP value of AVOA-CatBoost.

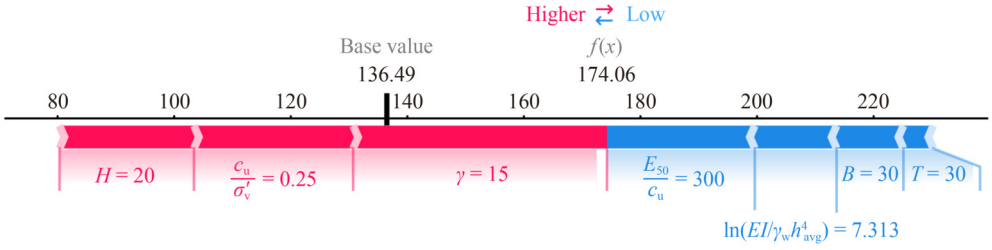

Fig. 10 Local Tree SHAP Value of AVOA-CatBoost in 468th sample.

Table 5 Details about the 468th sample and Tree SHAP explanation results

Location of the data in the dataset	Features							Wall deflection
	B	T	H	$\frac{c_u}{\sigma_v'}$	$\frac{E_{50}}{c_u}$	$\ln(EI/\gamma_w h_{avg}^4)$	γ	
468	30	30	20	0.25	300	7.313	15	174
Tree SHAP values	−11.36	−5.08	22.85	27.06	−25.14	−13.89	43.13	174.06

further understanding of the model. The results of these explanations are consistent with the underlying physical laws and previous engineering examples, which indicates that the model makes predictions in a factually correct manner [44]. The global and local explanations give us a better understanding of the model's decision-making process and deepen our understanding and trust in the model.

6 Conclusions

The main contribution of this study is to combine the finite element method, meta-heuristic optimization algorithms, ML algorithms, and explainable methods and to obtain an AVOA-CatBoost model with excellent performance through comparison. Specifically, we first used a database obtained from simulations based on the finite element method and performed EDA on this data to understand the patterns of the data itself. This database is then modeled using a state-of-the-art ensemble ML algorithm based on gradient boosting. To fully exploit the performance of the model, we used the AVOA algorithm to optimize the parameters of the model. To fully express the advantages of the models developed in this paper, not only were a variety of ML models selected but they were also compared with the models of other researchers. In the selection of the comparison method, in addition to the performance on the test data set such as *RMSE*, *MAE*, and R^2, We still used 10 repetitions and 10-fold cross-validation and conducted a total of 100 repetitive experiments to reflect the model's generalization performance. Finally, in order to apply the model better in the actual engineering field, we still used sensitivity analysis as well as the SHAP method to explain AVOA-CatBoost comprehensively. From the above study, the following conclusions can be drawn.

1) The AVOA-CatBoost established in this paper has the best performance on the problem of predicting diaphragm wall deflection during excavation (*RMSE* = 1.84, *MAE* = 1.18, R^2 = 0.9993) and cross-validation (*RMSE* = 2.65 ± 1.54, *MAE* = 1.17 ± 0.23, R^2 = 0.998 ± 0.002).

2) A comprehensive explanation of the model was carried out using sensitivity analysis with the SHAP method, and the results of the explanation were consistent with the EDA as well as the actual engineering situation, which improved the transparency and practicability of the model.

3) The explainable analysis revealed that the most crucial parameter is the system stiffness, and that the remaining features tend not to have a single effect on the model output, but rather a complex interaction between multiple features affects the model output.

In actual projects, engineers can rely on the proposed AVOA-CatBoost model to predict the lateral deflection of diaphragm walls by measuring the actual structural parameters of the footings and soil geologic parameters required by the model. In addition, the most critical factors affecting the current maximum lateral deflection of diaphragm walls can be found in the output of the model and the interpretable analysis results, which provide a basis for engineers to control the maximum lateral deflection of diaphragm walls.

Undoubtedly, the model developed in this paper shows an extremely high performance. However, due to the limitation of the data set, there still exists a gap between the current study and the complex practical engineering. Future research can improve these limitations and further improve the performance and applicability of the model by collecting more features and engineering excavation examples. We will also establish a graphics user interface system that interfaces with the actual construction management system to realize complete automation of surveillance and prediction.

Acknowledgements This study was supported by the National Natural Science Foundation of China (Grant Nos. 42107214 and 52130905).

Competing interests The authors declare that they have no competing interests.

References

1. Fang L. Environmental impact assessment in the whole process of super high-rise building construction. Fresenius Environmental Bulletin, 2021, 30(6B): 7923–7932
2. Wu J, Cai J, Liu Z, Yuan S, Bai Y, Zhou R. BI-IEnKF coupling model for effective source term estimation of natural gas leakage in urban utility tunnels. Tunnelling and Underground Space Technology, 2023, 136: 136
3. Guo, J. and G. Liu, Experimental study on the soil–structure responses induced by tunnelling in limited space. Applied Sciences, 2023, 13(12): 7000
4. Zhang C, Zhao Z, Guo D, Gong D, Chen Y. Optimization of spatial layouts for deep underground infrastructure in central

business districts based on a multi-agent system model. Tunnelling and Underground Space Technology, 2023, 135: 135
5. van Nguyen D, Kim D, Choo Y. Optimized extreme gradient boosting machine learning for estimating diaphragm wall deflection of 3D deep braced excavation in sand. Structures, 2022, 45: 1936–1948
6. Roy A F V, Cheng M Y, Wu Y W. Time dependent evolutionary fuzzy support vector machine inference model for predicting diaphragm wall deflection. Neural Network World, 2014, 24(2): 193–210
7. Demeijer O, Chen J J, Li M G, Wang J H, Xu C J. Influence of passively loaded piles on excavation-induced diaphragm wall displacements and ground settlements. International Journal of Geomechanics, 2018, 18(6): 04018052
8. Sabzi Z, Fakher A. The performance of buildings adjacent to excavation supported by inclined struts. International Journal of Civil Engineering, 2015, 13(1B): 1–13
9. Xiao H J, Zhou S H, Sun Y Y. Wall deflection and ground surface settlement due to excavation width and foundation pit classification. KSCE Journal of Civil Engineering, 2019, 23(4): 1537–1547
10. Liu S, Song Z, Zhang Y, Guo D, Sun Y, Zeng T, Xie J. Risk assessment of deep excavation construction based on combined weighting and nonlinear FAHP. Frontiers in Earth Science, 2023, 11: 1204721
11. Masuda T. A study of empirical correlation for lateral deflections of diaphragm walls in deep excavations. In: Proceedings of International Symposium on Geotechnical Aspects of Underground Construction in Soft Ground. London: A.A. Balkema, 1996, 167–172
12. Moormann C. Analysis of wall and ground movements due to deep excavations in soft soil based on a new worldwide database. Soil and Foundation, 2004, 44(1): 87–98
13. Wang J H, Xu Z H, Wang W D. Wall and ground movements due to deep excavations in shanghai soft soils. Journal of Geotechnical and Geoenvironmental Engineering, 2010, 136(7): 985–994
14. Kung G T C, Juang C H, Hsiao E C, Hashash Y M. Simplified model for wall deflection and ground-surface settlement caused by braced excavation in clays. Journal of Geotechnical and Geoenvironmental Engineering, 2007, 133(6): 731–747
15. Goh A T C, Zhang F, Zhang W, Zhang Y, Liu H. A simple estimation model for 3D braced excavation wall deflection. Computers and Geotechnics, 2017, 83: 106–113
16. Liu B K, Wang Y, Rabczuk T, Olofsson T, Lu W. Multi-scale modeling in thermal conductivity of polyurethane incorporated with phase change materials using physics-informed neural networks. Renewable Energy, 2024, 220: 220
17. Zhao H, Liu W, Guan H, Fu C. Analysis of diaphragm wall deflection induced by excavation based on machine learning. Mathematical Problems in Engineering, 2021, 2021(1): 6664409
18. Zhang W G, Zhang R, Wu C, Goh A T C, Lacasse S, Liu Z, Liu H. State-of-the-art review of soft computing applications in underground excavations. Geoscience Frontiers, 2020, 11(4): 1095–1106
19. Liu B K, Vu-Bac N, Zhuang X, Fu X, Rabczuk T. Stochastic integrated machine learning based multiscale approach for the prediction of the thermal conductivity in carbon nanotube reinforced polymeric composites. Composites Science and Technology, 2022, 224: 224
20. Qi C C, Tang X L. Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. Computers & Industrial Engineering, 2018, 118: 112–122
21. Yong W X, Zhang W, Nguyen H, Bui X N, Choi Y, Nguyen-Thoi T, Zhou J, Tran T T. Analysis and prediction of diaphragm wall deflection induced by deep braced excavations using finite element method and artificial neural network optimized by metaheuristic algorithms. Reliability Engineering & System Safety, 2022, 221: 221
22. Shariati M, Mafipour M S, Ghahremani B, Azarhomayun F, Ahmadi M, Trung N T, Shariati A. A novel hybrid extreme learning machine-grey wolf optimizer (ELM-GWO) model to predict compressive strength of concrete with partial replacements for cement. Engineering with Computers, 2022, 38(1): 757–779
23. Liu B, Lu W, Olofsson T, Zhuang X, Rabczuk T. Stochastic interpretable machine learning based multiscale modeling in thermal conductivity of Polymeric graphene-enhanced composites. Composite Structures, 2024, 327: 327
24. Hassija V, Chamola V, Mahapatra A. Interpreting black-box models: A review on explainable artificial intelligence. Cognitive Computation, 2024, 16(1): 45–74
25. Tang Y, Reed P M, Wagener T, van Werkhoven K. Comparison of parameter sensitivity analysis methods for lumped watershed model. In: Proceedings of World Environmental and Water Resources Congress 2008. Honolulu: ASCE, 2008, 1–8
26. Lundberg S M, Lee S I. A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 2017, 30
27. Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding. Science, 2000, 290(5500): 2323–2326
28. Peng H C, Long F H, Ding C. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(8): 1226–1238
29. Abdollahzadeh B, Gharehchopogh F S, Mirjalili S. African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. Computers & Industrial Engineering, 2021, 158: 158
30. Friedman J H. Greedy function approximation: A gradient boosting machine. Annals of Statistics, 2001, 29(5): 1189–1232
31. Ke G, Meng Q, Finley T. Lightgbm: A highly efficient gradient boosting decision tree. In: Proceedings of Advances in Neural Information Processing Systems. Long Beach, CA: Curran Associates, Inc., 2017, 30
32. Prokhorenkova L, Gusev G, Vorobev A. CatBoost: Unbiased boosting with categorical features. In: Proceedings of Advances in Neural Information Processing Systems. Montreal: Curran Associates, Inc., 2018, 31
33. Duan T, Anand A, Ding D Y. Ngboost: Natural gradient boosting for probabilistic prediction. In: Proceedings of International Conference on Machine Learning. Auckland: PMLR, 2020, 2690–2700
34. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In:

Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining. San Francisco, CA: Association for Computing Machinery, 2016, 785–794

35. Breiman L. Random forests. Machine Learning, 2001, 45(1): 5–32
36. Suykens J A K, Vandewalle J. Least squares support vector machine classifiers. Neural Processing Letters, 1999, 9(3): 293–300
37. Breiman L. Stacked regressions. Machine Learning, 1996, 24(1): 49–64
38. Cawley G C, Talbot N L C. On over-fitting in model selection and subsequent selection bias in performance evaluation. Journal of Machine Learning Research, 2010, 11: 2079–2107
39. Liu B, Vu-Bac N, Zhuang X, Lu W, Fu X, Rabczuk T. Al-DeMat: A web-based expert system platform for computationally expensive models in materials design. Advances in Engineering Software, 2023, 176: 103398
40. Merghadi A, Abderrahmane B, Bui D T. Landslide susceptibility assessment at mila basin (Algeria): A comparative assessment of prediction capability of advanced machine learning methods. ISPRS International Journal of Geo-Information, 2018, 7(7): 268
41. Lucay F A. Accelerating global sensitivity analysis via supervised machine learning tools: Case studies for mineral processing models. Minerals, 2022, 12(6): 750
42. Liu B, Penaka S R, Lu W, Feng K, Rebbling A, Olofsson T. Data-driven quantitative analysis of an integrated open digital ecosystems platform for user-centric energy retrofits: A case study in northern Sweden. Technology in Society, 2023, 75: 75
43. Hickey J M, Di Stefano P G, Vasileiou V. Fairness by explicability and adversarial SHAP learning. In: Proceedings of Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020. Ghent: Springer International Publishing, 2021, 174–190
44. Kung G T C, Hsiao E C L, Schuster M, Juang C H. A neural network approach to estimating deflection of diaphragm walls caused by excavation in clays. Computers and Geotechnics, 2007, 34(5): 385–396