Front. Struct. Civ. Eng. 2021, 15(4): 821–833
https://doi.org/10.1007/s11709-021-0742-8

TRANSDISCIPLINARY INSIGHT

Evaluation and prediction of slope stability using machine learning approaches

Shan LIN[a], Hong ZHENG[a], Chao HAN[a], Bei HAN[a*], Wei LI[b*]

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

[b] *School of Civil Engineering and Architecture, Linyi University, Linyi 276000, China*

[*] *Corresponding author. E-mails: hanbxie@bjut.edu.cn; liwei9012@lyu.edu.cn*

ABSTRACT In this paper, the machine learning (ML) model is built for slope stability evaluation and meets the high precision and rapidity requirements in slope engineering. Different ML methods for the factor of safety (FOS) prediction are studied and compared hoping to make the best use of the large variety of existing statistical and ML regression methods collected. The data set of this study includes six characteristics, namely unit weight, cohesion, internal friction angle, slope angle, slope height, and pore water pressure ratio. The whole ML model is primarily divided into data preprocessing, outlier processing, and model evaluation. In the data preprocessing, the duplicated data are first removed, then the outliers are filtered by the LocalOutlierFactor method and finally, the data are standardized. 11 ML methods are evaluated for their ability to learn the FOS based on different input parameter combinations. By analyzing the evaluation indicators R^2, *MAE*, and *MSE* of these methods, SVM, GBR, and Bagging are considered to be the best regression methods. The performance and reliability of the nonlinear regression method are slightly better than that of the linear regression method. Also, the SVM-poly method is used to analyze the susceptibility of slope parameters.

KEYWORDS slope stability, factor of safety, regression, machine learning, repeated cross-validation

1 Introduction

The slope is the most common geomorphologic and geologic environment in human engineering and economic activities. Recently the slope instability (especially collapses, landslides, and mudslides) on account of the natural and artificial factors, caused heavy loss of lives, property and engineering economy. The accuracy of slope stability evaluation determines the success of slope engineering. Therefore, it has realistic engineering significance to predict slope stability accurately [1].

Slope engineering is a complex nonlinear dynamic open system containing many random, complex factors, and its stability and influencing factors have a highly nonlinear relationship. The traditional methods of slope stability analysis are the limit equilibrium method [2–4] and numerical analysis methods [5–8], such as numerical manifold method (NMM) [9], discontinuous deformation analysis (DDA) [10–12], phase-filed model (PFM) [13–15]. However, these traditional methods have encountered bottlenecks, such as "unclear mechanism" and "inaccurate models", and it is still difficult to accurately predict the stability of complex slopes. Also, these traditional methods are confronted with heavy computing burden. Particularly their stability calculating process is cumbersome, and is difficult to meet the requirements of speeding up slope designs. In most cases, engineering mega-projects involving a large number of slopes, such as hydroelectric engineering, require rapid stability assessment and rapid design at the preliminary design stage. However, data processing methods and data modeling algorithms based on machine learning (ML) [16–18] can productively extract data features and mine data value, providing a new way for intelligent research on slope stability [19–22].

Many scholars have introduced the theory of artificial intelligence into the field of Geology and put forward the

Article history: Received Jan 5, 2021; Accepted Apr 22, 2021

methods of slope stability evaluation based on ML [23]. Applying ML to slope stability analysis has produced positive outcomes [24–29]. The research result of Erzin and Cetin [25] shows that the artificial neural networks (ANN) models have better prediction performance than the multiple regression (MR) models. Wang and Sassa [30], Pradhan et al. [31], Melchiorre et al. [32] used ANN to estimate and predict the deformation of regional landslides. Based on the extreme learning neural network, Li et al. [33] have developed a productive tool that can fast-track assessing the stability of rock slopes. Pradhan [34] adopted decision tree (DT), support vector machine (SVM), and adaptive neuro-fuzzy inference system (ANFIS) to make spatial prediction of landslide susceptibility chart at Penang Hill, Malaysia. According to the Bayes discriminating analysis (BDA) theory and the engineering practice, some scholars [35–37] have established the BDA model for slope stability prediction. Li et al. [38] applied SVM model to identifying landslide stability in the Xiluodu reservoir areas, and obtained good results. Zhao et al. [39] examined the possible application of relevance vector machine (RVM) in slope stability analysis, and put forward the nonlinear relationship between slope stability and its influencing factors. Zhou et al. [40] proposed a method for slope stability prediction using the gradient boosting machine (GBM) method. Qi and Tang [41] demonstrated that integrated AI approaches had great potential to predict slope stability. Although ML has been broadly used in slope deformation prediction, various intelligent algorithms and technical means have perfected the slope deformation prediction model and brought the slope deformation prediction into a new era. However, it is still necessary to find a suitable slope deformation prediction method, because each of those above mentioned methods has disadvantages.

As mentioned above, this paper primarily aims to study the suitability of various ML methods for the evaluation of factor of safety (FOS) To achieve the objective, a survey methodology is developed to compare the performances of different ML methods, such as support vector regression (SVR), Bayesian ridge (BR), linear regression (LR), elastic net regression (ENR), K nearest neighbors (KNN), adaptive boosting regression (ABR), gradient boosting regression (GBR), Bagging, extra trees regression (ETR), DT, and random forest (RF). These methods have been chosen especially because they are broadly used in various projects, but rarely compared with each other, largely due to the availability of Python software. The following is the outline of this study. Section 2 briefly introduces the slope data set and ML algorithms, Section 3 presents these methods applied to slope stability evaluation, and Section 4 provides the results and discussions by performance criteria, as well as contributions and limitations of current research. The conclusions are drawn in Section 5.

2 Materials and algorithms

2.1 Data set and predictor variables

Many factors affect the stability of slopes, including the terrain landform, soil and rock physical mechanicals, and external triggering factors such as hydrogeological conditions. The FOS is a comprehensive index to evaluate slope stability. In the analysis of slope stability, FOS is essentially the ratio of slope sliding resistance and slope sliding force, and is directly related to the soil shear strength [42]. The unit weight γ, cohesion C, and angle of internal friction φ are the critical parameters to determine the soil shear strength. Some scholars use the strength reduction method [43] and the gravity increase method [44] to calculate FOS. Slope angle β and slope height H are geometric characteristics of a slope, often determining the slope failure conditions. As the slope height increases, the slope stability gradually decreases. The larger the slope angle, the lower the slope stability. Water infiltration increases the weight of the geotechnical, and the shear strength of the rock and soil is reduced due to softening. All these changes are unfavorable to slope stability. Therefore, each slope's parameters related to the geometry and geotechnical characteristics are selected. More specifically, lead factors affecting slope stability are as follows: unit weight γ (kN/m^3), cohesion C (kPa), angle of internal friction φ (°), slope angle β(°), slope height H (m), and pore water pressure ratio r_u. They are consistent with the parameters commonly available in the literature [19,20,40,45]. Theoretically, there are other indicators, but collecting these other indicators would be a considerable challenge before they can be practically applied. As seen in Fig. 1, the FOS value can be calculated from the mapping f which is established from the physical and mechanical characteristics of the slope.

$$FOS = f(\gamma, C, \varphi, \beta, H, r_u). \quad (1)$$

2.2 Data visualization

To evaluate the performance of the developed ML algorithms, the 349 slope cases used in this study are collected from more than 9 published articles (Refs.

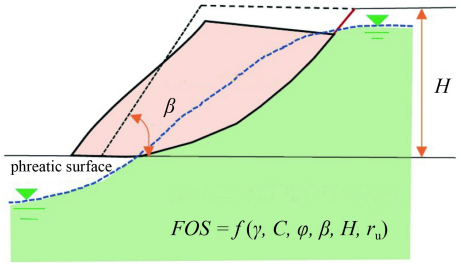

Fig. 1 Failure mechanism of the homogeneous finite slopes.

[46–55]) which are referred to in the appendix.

The Violin plots of the slope data set are shown in Fig. 2. The violin plots show the maximum values, median values, and minimum values of each input parameter, and the distribution of its density. In Fig. 2, the white dot in the center of the violin plots represents the median, and the top and the bottom edges of the black bar represent the upper and lower quartiles, respectively. The thin black line extended by the thick black bar represents the 95% confidence interval. The data density at any position can be seen from the shape of the violin plots. The greater the width, the higher the data density. The distribution of these variables and the relationship between FOS and other input variables are shown in the correlation matrix diagram of Fig. 3. From the figure, it can be seen the pairwise relationship between the parameters with corresponding correlation coefficients and the marginal frequency distribution of each parameter. The upper triangle is the correlation coefficient, and the histogram on the diagonal depicts the numerical distribution of each feature. The curve drawn in each scatter plot of the lower triangle is called local smoothing, which represents the general relationship between the x-axis and y-axis variables. As can be seen from Fig. 3, FOS has a positive correlation with C cohesion, FOS has a negative correlation with β, and β has a significant correlation with φ. Meanwhile, it demonstrates that different variables have different effects on FOS. Therefore, a series of comparative experiments are conducted with varying combinations of input variables to assess the role

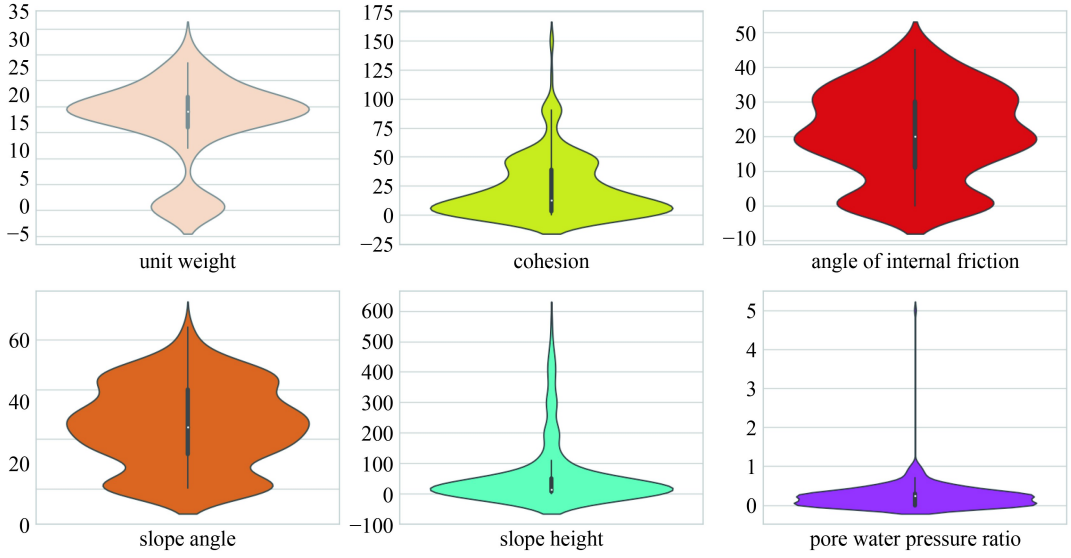

Fig. 2 Violin plots showing the distribution of observed slope cases.

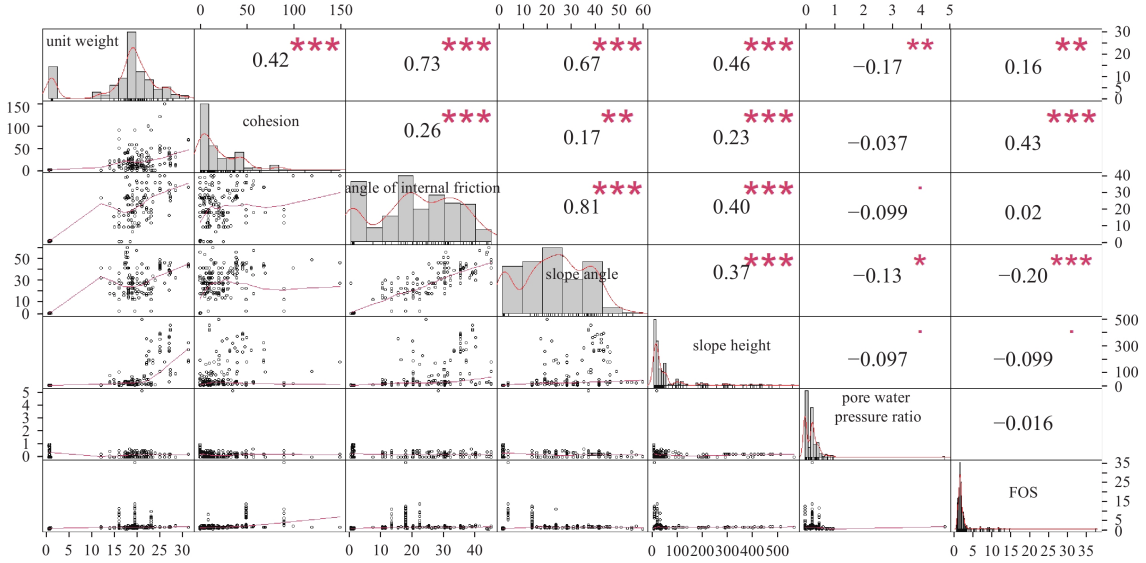

Fig. 3 Correlation matrix of variables for all slope cases.

of different variables in this study, as shown in Table 1.

2.3 Data preprocessing

Data preprocessing is a data processing method with quality requirements and calculation requirements as the primary motivation. The influence of redundant variables, missing values, and unbalanced data on subsequent research can be reduced by using data processing. It includes the processing of data values and variables. This paper uses the Isolation Forest algorithm to identify outliers. Then duplicate data are deleted, saving the time of the training model and avoiding the overfitting training model. Data has varying scales and some ML algorithms make assumptions about your data having a Gaussian distribution. To make the feature extraction more accurate, we use the Power Transformer in the Scikit-learn library to standardize the data.

2.4 ML techniques

In this paper, 11 regression techniques of ML are studied. These algorithms are increasingly used for slope stability analysis. Some of them have been used in slope stability evaluation, and have established more complex nonlinear relationships between input variables and outputs, and have been proven to have good prediction performance. They have an effective implementation, and the resulting model allows for fast calculation of FOS. The following slope stability prediction effects of different methods are compared. In this article, the number of samples is N (349), the feature vector $\boldsymbol{X}_i$ consists of six slope performance modifiers $\{x_1, x_2, x_3, x_4, x_5, x_6\}$, corresponds to the variables discussed in Section 2.2. It is worth reminding that, several articles on regression methods have been published over the years but the subject will always need updating [20,56–58]. And these articles only provide a brief introduction to each regression algorithm.

2.4.1 Linear regression (LR)

LR [59] is one of the fundamental problems in regression analysis. For the sample $(\boldsymbol{X}, \boldsymbol{y})$, $\boldsymbol{y} = (y_1, y_2, ..., y_N)^{\mathrm{T}}$, the input data has p features, $\boldsymbol{X}_i = (x_{1i}, x_{2i}, ..., x_{Np})^{\mathrm{T}}$, the coefficient is $\boldsymbol{w} = (w_1, ..., w_p)$. Therefore, the corresponding multivariate linear regression equations are established:

$$f(\boldsymbol{X}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{X} + b, \tag{2}$$

where b is the intercept. As an optimization problem, usually, the least-squares form is used to minimize the following regularized cost function:

$$\arg\min_{w,b} L(\boldsymbol{w}, b) = \min_{w,b} \|f(\boldsymbol{X}) - \boldsymbol{y}\|_2^2. \tag{3}$$

Table 1 Different models for FOS prediction with different input parameters

model	γ	C	φ	β	H	r_u
A	√	√	√	√	√	√
B	√	√	√	√	√	–
C	√	√	–	√	√	√
D	√	√	√	–	√	√
E	√	√	√	√	–	√
F	–	√	√	√	√	√
G	√	–	√	√	√	√

2.4.2 Bayesian ridge (BR)

Ridge regression increases the norm of weight coefficient based on LR to overcome the collinearity problem. If there are any missing or contradictory ill-conditioned data, BR can be considered. It is an algorithm to estimate the regression problem by using a probability model. After $\boldsymbol{X}$ is regularized, the class is represented as $(\boldsymbol{X}, f(\boldsymbol{X})) = (\boldsymbol{X}, \boldsymbol{w}^{\mathrm{T}}\boldsymbol{X})$. If the complexity of the class representation is defined as $\|w\|^2$, and the class consistency criterion and Occam's razor criterion are used at the same time, the following questions can be considered:

$$\min_{w} \left\|\boldsymbol{y} - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}\right\|_2^2 + \lambda\|w\|_2^2, \tag{4}$$

where $\lambda \geqslant 0$ represents a regularization parameter. The larger λ, the greater the degree of contraction; when $\lambda = 0$, BR degenerates to LR automatically. BR is very robust to ill-conditioned data, and cross-validation (CV) is not required to select hyper-parameters, but the inference process of maximizing the likelihood function is relatively time-consuming.

2.4.3 Elastic net regression (ENR)

ENR [60] is a mixture of lasso regression and ridge regression. ENR is a linear regression model that uses L1 and L2 priors as a regularization matrix. ENR has the advantages of both lasso regression and ridge regression, and can achieve the purpose of variable selection and a good grouping effect. The goal function of the model is

$$\min_{w} \frac{1}{2N}\left\|\boldsymbol{y} - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}\right\|_2^2 + \alpha\|w\|_1 + \lambda\|w\|_2^2, \tag{5}$$

when $\alpha = 0$, ENR regression is ridge regression; $\lambda = 0$, ENR regression is lasso regression.

2.4.4 K nearest neighbors (KNN)

KNN [61] predict the expectant outputs based on

distance. In KNN, when the training set, distance measure, k value, and classification decision rules (such as majority voting) are determined, the classes of inputs are uniquely determined. The distance between the two samples in the feature space reflects the similarity of the two samples. Generally, the feature space of KNN is an n-dimensional vector space over the field of real number R^n. Euclidean distance or other distances can be used, such as L_P distance or Minkowski distance. The choice of k value will have a significant impact on the output results. In the practical application, CV is usually used to determine the optimal k value. KNN and its improved algorithm have high time complexity, and cannot give the specific objective function expression. However, it can establish different objective function approximations for different cases to be tested, so it has high prediction accuracy.

2.4.5 Support vector machine (SVM)

SVM [62] is a learning process that transforms the input space into high-dimensional space through nonlinear transformation defined by the inner product function and solves the regression function in the high-dimensional space. SVM can solve the problems of small samples, high dimensionality, and nonlinearity. SVR is characterized by the use of kernel functions, sparse solutions, and dimension control of the number of edges and support vectors. SVR has good generalization ability and reliable prediction accuracy, being described as an optimization problem. First, a convex ε loss function is defined to control the fitting accuracy. By minimizing it, the flattest tube containing most of the training samples is found. Figure 4 shows the schematic of the linear model $f(\boldsymbol{X})$ work principle. For nonlinear regression problems, the final decision function is as follows:

$$f(\boldsymbol{X}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^*)\kappa(\boldsymbol{X}, \boldsymbol{X}_i) + b, \tag{6}$$

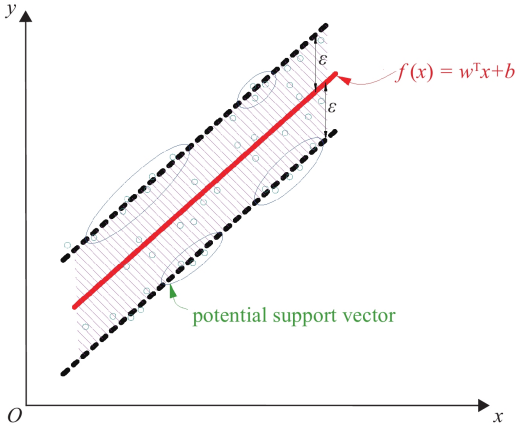

Fig. 4 The schematic diagram of SVR.

where α_i, α_i^* is the Lagrange multiplier, $\kappa(\boldsymbol{X}_i, \boldsymbol{X}_j) = \phi(\boldsymbol{X}_i)^{\mathrm{T}} \phi(\boldsymbol{X}_j)$ are the kernel parameters. There are three kinds of kernel types used in this paper: linear kernel (SVR-linear), polynomial kernel (SVR-poly), and RBF kernel (SVR-rbf, often called Gaussian kernel).

2.4.6 Decision tree (DT)

A DT [63,64] is a tree data structure, composed of nodes and directed edges. Nodes are divided into two types: branch nodes and leaf nodes. The branches of non-leaf nodes represent the tests of data features, and each branch represents the output of the data feature in a certain value range. Admittedly, the process of building a DT is a process of summarizing a set of classification rules from training samples. Each time the classification feature is selected, the local optimal feature under the current conditions is selected to generate the classification rule. Local optimization means that the data set will be more ordered once the features have been selected for data classification. The order degree of the data set is usually measured by entropy, Gini, variance. DT has the advantages of being simple and easy to learn, being intuitive and easy to master, high efficiency, and high analytical accuracy.

2.4.7 Random forest (RF)

RF [65] is an integrated algorithm based on a DT. In the process of constructing the regression tree, RF will use the Bootstrap re-sampling method to randomly select values from y, and randomly select a specified number of variables from the independent variables to determine the classification tree nodes. The method based on ensemble learning takes the mean value of each DT as the regression prediction value.

$$\bar{h}(\boldsymbol{X}) = \frac{1}{T} \sum_{t=1}^{T} \{h(\boldsymbol{X}, \theta_t)\}, \tag{7}$$

where θ_t is assumed to be an independent and identically distributed random variable; T represents the number of DT; $h(X, \theta_t)$ stands for the outputs. RF randomly selects features based on the local region, and significantly improves operating efficiency. Also, the contribution of features can be obtained, but it is easy to overlearn.

2.4.8 Adaptive boosting machine

Adaptive boosting (Ada boost) is a common and extensively used iterative algorithm with high efficiency, simple implementation, and flexibility. This algorithm is implemented by re-adjusting the data distribution. It calculates the weight of each sample based on the

correctness of each sample classification and the accuracy of the previous overall classification. First, the new data sets are sent to the classifier for training, and then the trained classifiers are integrated to form the final decision classifier. The loss function of the Ada boost algorithm is

$$G_t(\alpha) = \sum_{n=1}^{N} \boldsymbol{w}^{(t)} \exp\left(-\boldsymbol{y}_n \alpha h_t(\boldsymbol{X}_n)\right)^2, \tag{8}$$

where α is the multiplier, w is the weight. Ada boost algorithm can combine multiple single models, but it is inefficient in large-scale data.

2.4.9 Gradient boosting machine (GBM)

GBM [66] refers to the gradient descent method. It is an integrated model by constructing multiple classification regression DT and combining them to form a strong learner. GBR is one of the boosting algorithms. Its basic idea is to use the value of the negative gradient of the loss function in the current model as an approximation of the residual error of the training result, and take the value as the target of the next training. Just like the other boosting algorithms, GBR adopts the forward phase method to establish the additive model [67]. Therefore, the prediction value of GBR is relatively stable, and it can achieve high precision even in the case the amount of data are small. At the same time, the use of strong pre-pruning makes it develop the advantages of small depth, fewer memory resources, and fast prediction speed.

2.4.10 Bagging

The general idea of Bagging technology is to use a weak learning algorithm and a training set to make the learning algorithm repeat training. In each round, N training samples are randomly selected from the initial training set by using the repeatable sampling method. After training, a sequence of prediction functions is obtained. For regression problems, the simple average method is often used to distinguish new cases. As shown in Fig. 5, the selection of Bagging's training set is random, and the training sets of each round are independent of each other. Each prediction function of Bagging has no weight and can be generated in parallel. It reduces the generalization error by reducing the variance.

2.4.11 Extra trees

Extra trees [68] were proposed by Pierre Geurts in 2006. It is a ML algorithm based on Bagging and a variant of the random forest algorithm. It uses classification and regression tree (CART) as the basic weak learner model, and each DT is independent of the other. In the process of combining the final models, for regression problems, the arithmetic average of all DT models is taken as the final result. Each DT of the extra trees selects multiple features randomly, and each classification threshold is randomly generated, and can effectively prevent overfitting, improve the generalization ability of the model, and make the model have good performance in predicting untrained data.

Bagging and DT can enhance the generalization performance of regression, and have high stability, and are not prone to overfitting.

3 Slope stability assessment with the ML model

3.1 Training and testing data sets

This study implements regression analysis in Python 3.8. During the implementation of different ML methods, 9/10 slope cases are used to train the model, and the rests are used to test the generalization ability of the models.

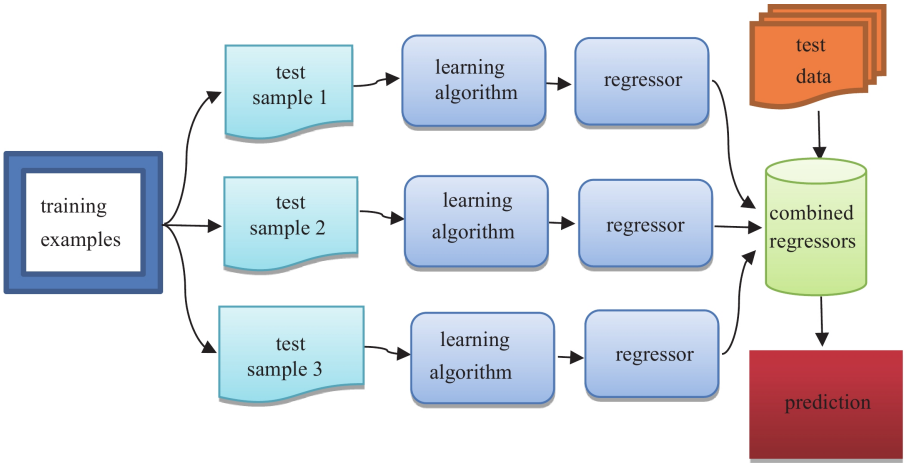

Fig. 5 Bagging algorithm.

Numerical results may vary given the stochastic nature of the algorithm or evaluation procedure or differences in numerical precision. Consider taking the average value of the calculated results for comparison.

3.2 Evaluation of the regression model's performance

Mean absolute error (*MAE*), mean square error (*MSE*), and coefficient of determination (R^2) are selected to evaluate the prediction effect of the regression model. *MAE* represents the average value between the predicted value and the actual value. The closer the *MAE* is to 0, the effect of the adjustment is better, meaning that the prediction model more precisely describes the set of training samples. *MSE* represents the mean of the squares between the predicted value and the actual value. The following are the calculation formulas of *MAE* and *MSE*.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|f(\boldsymbol{X}_i) - y_i|, \tag{9}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(f(\boldsymbol{X}_i) - y_i)^2, \tag{10}$$

where n is the number of training data sets, y_i is the actual value of the test data set, $f(\boldsymbol{X}_i)$ is the predicted value of the test data set.

R^2 is the coefficient of determination indicating the degree of matching between the regression model and the target variable.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(f(\boldsymbol{X}_i) - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{11}$$

where $\bar{y}$ is the mean of the test data set. Theoretically, the value range R^2 is $(-\infty, 1]$. Generally speaking, the closer the value of R^2 is to 1, the imitation effect of the regression model is better on the target variables. By experience, we can know that R^2 is greater than 0.4, indicating that the model has a good effect on data fitting and can be used to predict the target variable [69]. What we need to know is that R^2 is affected by the number of samples and the number of features. Therefore, the results of different data sets trained by the same model will be different.

3.3 Verification technique of the proposed models

CV methods are generally used to evaluate the generalization performance of different regression algorithms on the same data set. Besides, the training process of the ML is enhanced by the repeated CV used for the model evaluation. To obtain a reliable performance evaluation, a 10-fold CV is often performed in multiple runs to avoid deviations caused by different sample divisions. In this paper, the 10-fold CV is used in the inner loop and 50 times in the external cycle, and the average value of repeated CV *MAE*, *MSE*, and R^2 is taken as the final evaluation value.

3.4 ML method development and parameter optimization

We established seven models to study the qualitative sensitivity of the parameters in Eq. (1). The input parameters involved in each model are shown in Table 1. 11 kinds of ML regression algorithm comparison schemes are designed. This paper uses the 10-fold CV to optimize the parameters of each algorithm. The parameter settings of each scheme are shown in Table 2.

4 Regression results and discussions

4.1 Regression results achieved by regression models

The performance comparison of all ML regression methods is shown in Fig. 6. The boxplots in Fig. 7 show the results of 50 repeated CVs for different models. For model A, the histogram in Fig. 8 shows that the overall performance of nonlinear SVR is the best, both SVR-poly and SVR-rbf are better than other algorithms. The average R^2 of SVR-poly is 0.8640, the highest among all the algorithms, and its *MAE* and *MSE* is 0.4208 and 1.5531 respectively, the lowest among all algorithms; The average values of R^2, *MAE*, and *MSE* of SVR-rbf are 0.8461, 0.4288, and 2.0153, respectively, followed by GBR and Bagging, with the average values of R^2 is 0.7949 and 0.7715 respectively. LR has the lowest

Table 2 Tuning parameters of different regression models

method	tuning parameters
BR	–
LR	–
ENR	{alphas = [0.0001, 0.0005, 0.001, 0.01, 0.1, 1, 10],l1_ratio = [.01, 0.1, 0.5, 0.9, 0.99], max_iter = 5000}
KNR	{n_neighbors = 5, p = 5}
SVR	{kernel = [rbf, poly, linear], c = [43,10, 100], degree = [, 3,]epsilon = [.1,.1,], coef0 = [, 2,]}
RFR	{n_estimators = 100, random_state = 100}
ABR	{n_estimators = 200, random_state = 100}
GBR	{n_estimators = 30}
Bagging	{n_estimators = 100}
ETR	{n_estimators = 500, random_state = 1}
DTR	{max_depth = 4, random_state = 0}

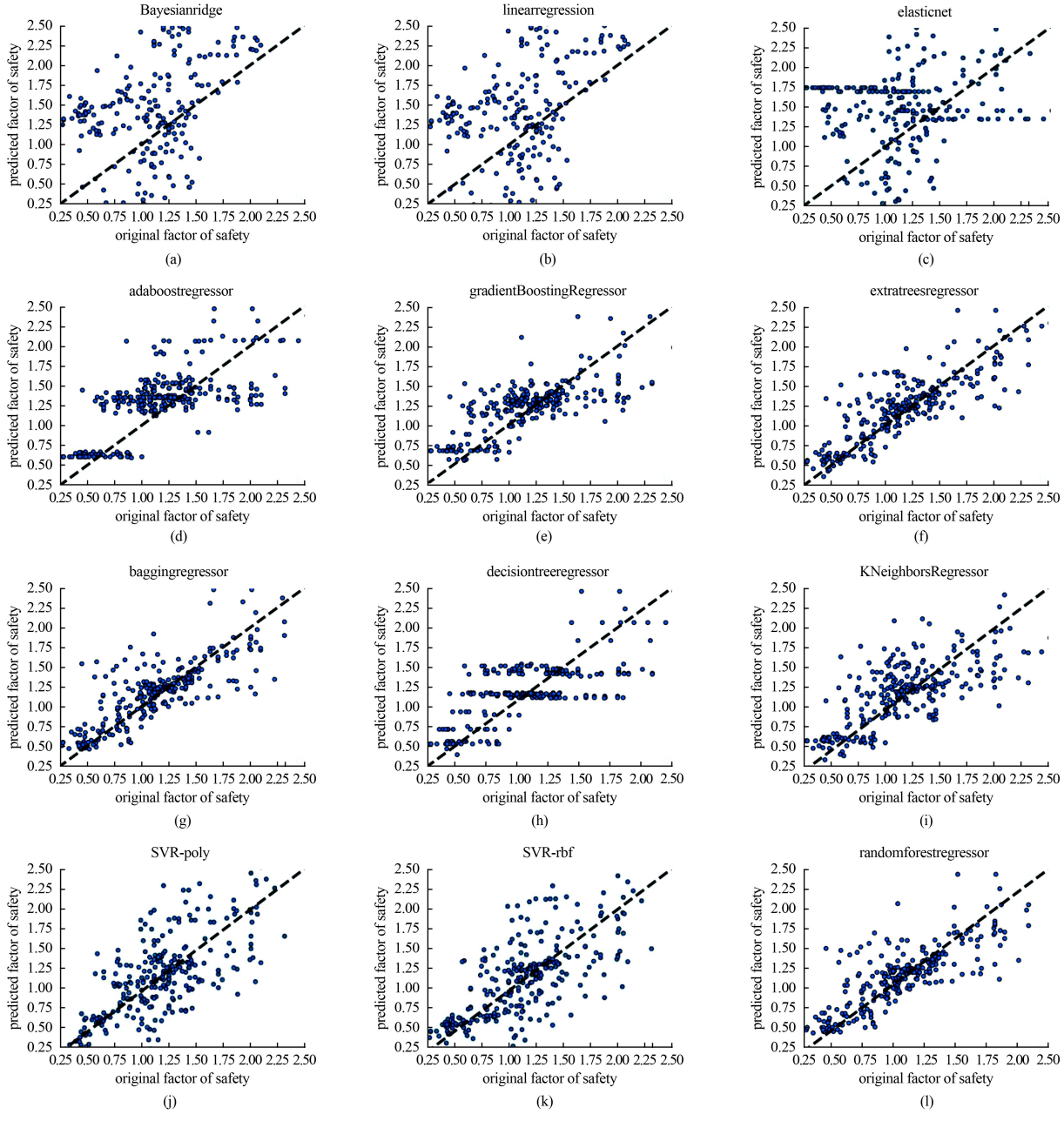

Fig. 6 Comparison of all ML regression methods for slope stability analysis. (a) BR; (b) LR; (c) ENR; (d) ABR; (e) GBR; (f) ETR; (g) Bagging; (h) DTR; (i) KNR; (j) SVR-poly; (k) SVR-rbf; (l) RFR.

relative performance, the average R^2 value of SVR-linear is 0.2392, the lowest among all algorithms, and the average R^2 value of EN is 0.2584; the average values of R^2, *MAE*, and *MSE* of DTR are 0.2886, 0.6374, and 4.0272, respectively. For model B, the average R^2 of SVR-rbf is 0.7966, followed by Bagging, SVR-poly, and KNR, and their average R^2 is 0.7837, 0.7528, and 0.7444, respectively. For model C, the higher R^2 obtained by ETR is 0.8085, followed by RFR, SVR-rbf, SVR-poly, and Bagging, and their R^2 is 0.7934, 0.7486, 0.7250, and 0.7290, respectively. Not surprisingly, the linear methods, such as SVR-linear, BR, LR, and EN, have not performed well here. For model D, the average value of R^2 obtained by SVR-rbf, SVR-poly, and GBR is relatively high, 0.4879, 0.4738, and 0.4478, respectively. The average R^2 of ETR is 0.1141, the relatively lowest among all algorithms. In a similar vein, the results of models E and F are shown in Fig. 7. We can observe that the average value of R^2 obtained by SVR-poly and SVR-rbf is relatively high. We can get the following facts: for ML techniques mentioned in this paper, the performance (R^2) of the training data set is between 0.0780 and 0.8640, the performance (*MAE*) is between 0.3613 and 1.1137, and the performance (*MSE*) is between 0.9859 and 6.8485.

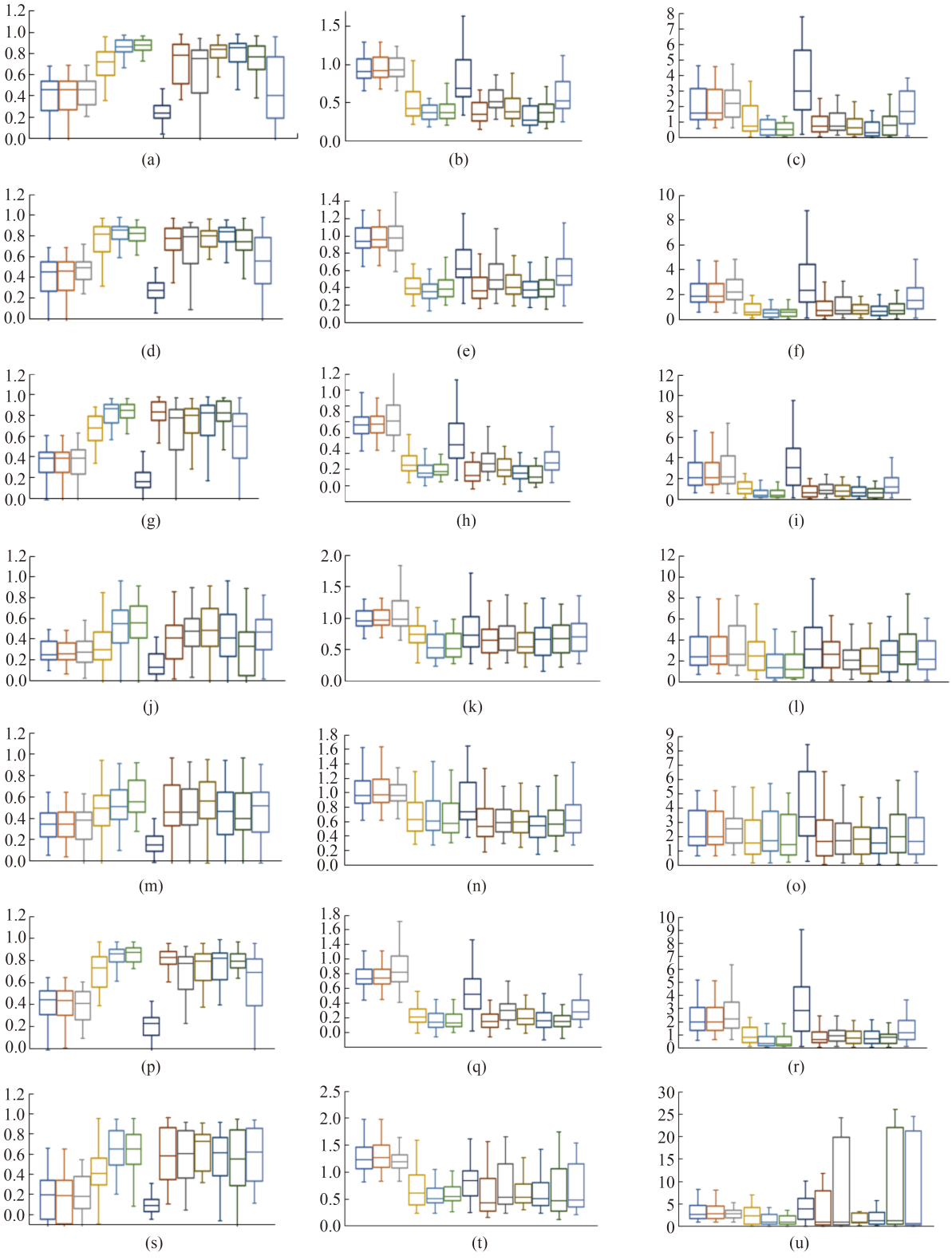

Fig. 7 Boxplot distributions of training data sets for different models—resulting from repeated CV. (a) R^2 of model A; (b) *MAE* of model A; (c) *MSE* of model A; (d) R^2 of model B; (e) *MAE* of model B; (f) *MSE* of model B; (g) R^2 of model C; (h) *MAE* of model C; (i) *MSE* of model C; (j) R^2 of model D; (k) *MAE* of model D; (l) *MSE* of model D; (m) R^2 of model E; (n) *MAE* of model E; (o) *MSE* of model E; (p) R^2 of model F; (q) *MAE* of model F; (r) *MSE* of model F; (s) R^2 of model G; (t) *MAE* of model G; (u) *MSE* of model G.

From these results, we can see that SVR nonlinearity can achieve relatively higher precision for all regressions despite the seriously unbalanced data set, while the results for linear algorithms are generally lower and this

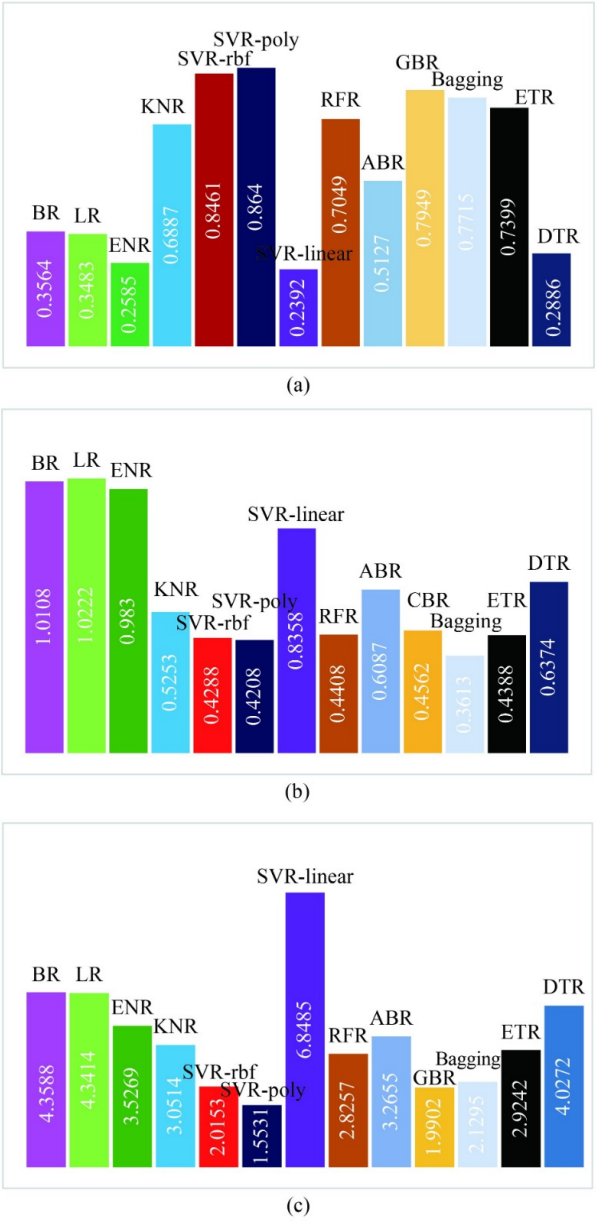

Fig. 8 The calculation results of model A. (a) The value of R^2; (b) the value of *MAE*; (c) the value of *MSE*.

is probably because algorithms fail to handle the nonlinear boundary class.

In addition to the prediction performance, the efficiency of different ML algorithms in slope evaluation varies significantly. By using a personal computer with Intel CPU i5-9400 to realize the modeling, the average value of the time spent in repeated CV was used as the final evaluation value. Taking model A as an example, BR generates results in 3.3162 s; LR generates results in 3.2106 s; EN generates results in 9.9647 s; KNR generates results in 3.5616 s; SVR-rbf generates results in 7.2451 s; SVR-poly generates results in 31.8426 s; SVR-linear generates results in 77.8207 s; RF generates results in 39.3332 s; AdaBoost generates results in 11.9616 s; GBR generates results in 6.1160 s; Bagging generates results in 44.6803 s; ET generates results in 130.2820 s; DT generates results in 3.3344 s. The above data show that ET and SVR-linear are the most computationally intensive techniques and take the longest amount of time to train. Although linear (BR, LR, and EN) algorithms and DT are relatively fast in evaluating the FOS values of these cases, their prediction performance is worse than others.

4.2 Comparison of ML regression techniques

Among the seven models in Table 1, model A using six parameters $(\gamma,C,\varphi,\beta,H,r_u)$ to train SVM have the best generalization performance. It is clear that SVR's polynomial algorithm produced relatively good results in terms of R^2, *MAE*, and *MSE* of all model training data sets, and the average value of R^2 is 0.8640. Therefore, this paper adopts the SVR-poly algorithm for parameter sensitivity analysis. During the research, the parameters of the SVR-poly algorithm are the same as those in the previous section. Here, the relative sensitivity of a parameter is assessed by the degree of performance degradation of the model (excluding the parameters analyzed) compared with model A (including all six parameters). As can be seen from Fig. 9, the vertical coordinate is R^2, the polynomial algorithm in SVR is used for regression fitting of all input parameters, and model A $(\gamma,C,\varphi,\beta,H,r_u)$ provides the best result. Comparing the performance of model A and model B $(\gamma,C,\varphi,\beta,H)$, we can see from Fig. 9 that r_u is a little sensitive in analyzing the FOS. Comparisons of the results of model A and model C (γ,C,φ,H,r_u) show that φ is sensitive in FOS. The comparisons between model A and model D (γ,C,φ,H,r_u) reveals that β is more sensitive. By analyzing the performance of model A and model E (γ,C,φ,H,r_u), it is found that H is also more sensitive. Comparing model A with model F (C,φ,β,H,r_u) implies that γ is a bit sensitive. It can be seen from the last row of Figs. 7 and 9 that the number of negative values in the R^2 calculation results are significantly increased due to

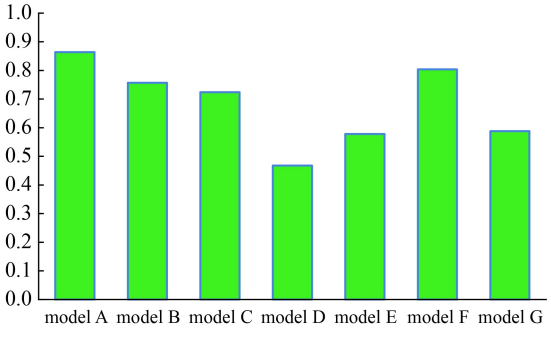

Fig. 9 Sensitivity analysis of SVR-poly input parameters.

removing the input parameter C in model G, and such a removal demonstrates that C is more sensitive.

Through the above analysis, it can be concluded that all the parameters contained in Table 1 are sensitive to FOS. Nevertheless, according to the performance of these models given in Fig. 9, the relative sensitivity of the parameter in descending order is C, β, H, φ, r_u, γ. β, H are the geometric parameters of the slope and play a vital role in FOS assessment. For this reason, in engineering practices of slope design and protection, cutting-slope unloading is usually carried out to ensure slope stability.

4.3 Superiority and limitations

This article systematically evaluates the FOS regression utilizing 11 ML methods. Although these studies reveal some significant findings, there are also limitations. First of all, we only predicted the FOS and did not explore the failure mechanism of the slope, such as circular sliding, plane failure, toppling failure, and wedge-shaped failure. And we did not touch upon the reasons for these failures. Secondly, rainfall, earthquake, human activities, and other external or trigger factors have a significant impact on the slope stability, but these factors are ignored in this study due to challenges in acquiring their data. Thirdly, other variables related to slope stability should have been collected to improve the model's prediction accuracy. Finally, other new development regression methods may be used to improve comparison of ML methods. If so, their outcome could have been compared with the methods studied in this paper. Also, the processing of outliers and sampling deviation are significant components of model uncertainty. They have a vital influence on the regression model of predicting slope stability, and need addressing in future research.

5 Summary and conclusions

Intelligent optimization algorithms have been widely used in slope deformation prediction. The integration of various intelligent algorithms and technical means has improved the slope deformation prediction models and brought their capacity up to a new era. The proposed models are constructed based on a data set of the 349 slope cases collected from published research works in recent years. This paper is primarily divided into data preprocessing, outlier processing, and comparison of various models. In the data preprocessing, the duplicate data are first removed, then the outliers are filtered by the Local Outlier Factor method, and finally, the data are standardized. In comparison and selection of various methods, BR method, LR method, EN method, KNR method, SVR method, RFR method, ABR method, GBR method, Bagging method, ETR method, DTR method are used for comparative experiments. By analyzing the evaluation indexes R^2, MAE, and MSE of these models, we can draw following conclusions. First, the ML regression method cannot be used blindly, because no method is considered to be a fully automatic regression method. For slope data sets, it is necessary to use repeated CV. Among 11 different ML algorithms, GBR, SVM, and Bagging are considered to be the best regression methods. The performance and reliability of the nonlinear regression method are slightly better than that of the linear regression method. The comparison results show that model A composed of 6 input variables $(\gamma, C, \varphi, \beta, H, r_u)$ is more reliable when combined with SVM, GBR, and Bagging methods. The cohesion of materials and the geometric design parameters (β, H) of slope have a vital influence on the stability of a slope. This study can be used as a benchmark for applying the artificial intelligence method to slope stability prediction and has a significant reference value for slope designs. The impact of data imbalance on slope stability prediction will be discussed in future research.

Acknowledgements Supported by the National Natural Science Foundation of China (Grant Nos. 11972043 and 11902134), Open Research Fund of the State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, and Chinese Academy of Sciences (Z019008), China Postdoctoral Science Foundation funded project (No. 2020M670077).

References

1. He X, Li S J, Liu Y X, Zhou Y P. Analyzing method of rock slope stability based on artificial neural network. Rock and Soil Mechanics, 2003, 24: 73−76 (in Chinese)
2. Nawari O, Hartmann R, Lackner R. Stability analysis of rock slopes with the direct sliding blocks method. International Journal of Rock Mechanics and Mining Sciences, 1997, 34(3–4): 220
3. Thiebes B, Bell R, Glade T, Jäger S, Mayer J, Anderson M, Holcombe L. Integration of a limit-equilibrium model into a landslide early warning system. Landslides, 2014, 11(5): 859−875
4. Johari A, Mousavi S. An analytical probabilistic analysis of slopes based on limit equilibrium methods. Bulletin of Engineering Geology and the Environment, 2019, 78(6): 4333−4347
5. Luan M T, Li Y, Yang Q. Discontinuous deformation computational mechanics model and its application to stability analysis of rock slope. Chinese Journal of Rock Mechanics and Engineering, 2000, 19(3): 289−294 (in Chinese)
6. Gitirana G Jr, Santos M A, Fredlund M D. Three-dimensional slope stability model using finite element stress analysis. In: Proceedings of GeoCongress 2008. New Orleans: ASCE, 2008, 191–198
7. Sun G, Lin S, Zheng H, Tan Y, Sui T. The virtual element method strength reduction technique for the stability analysis of stony soil slopes. Computers and Geotechnics, 2020, 119: 103349
8. Trivedi R, Vishal V, Pradhan S P, Singh T N, Jhanwar J C. Slope

stability analysis in limestone mines. International Journal of Earth Sciences and Engineering, 2012, 5(4): 759−766
9. Guo H, Zheng H, Zhuang X. Numerical manifold method for vibration analysis of Kirchhoff's plates of arbitrary geometry. Applied Mathematical Modelling, 2019, 66: 695−727
10. Zheng F, Leung Y F, Zhu J B, Jiao Y Y. Modified predictor-corrector solution approach for efficient discontinuous deformation analysis of jointed rock masses. International Journal for Numerical and Analytical Methods in Geomechanics, 2019, 43(2): 599−624
11. Zheng F, Zhuang X, Zheng H, Jiao Y Y, Rabczuk T. Kinetic analysis of polyhedral block system using an improved potential-based penalty function approach for explicit discontinuous deformation analysis. Applied Mathematical Modelling, 2020, 82: 314−335
12. Zhuang X, Zheng F, Zheng H, Jiao Y Y, Rabczuk T, Wriggers P. A cover-based contact detection approach for irregular convex polygons in discontinuous deformation analysis. International Journal for Numerical and Analytical Methods in Geomechanics, 2021, 45(2): 208−233
13. Zhou S, Rabczuk T, Zhuang X. Phase-field modeling of quasi-static and dynamic crack propagation: COMSOL implementation and case studies. Advances in Engineering Software, 2018, 122: 31−49
14. Zhou S, Zhuang X, Rabczuk T. A phase-field modeling approach of fracture propagation in poroelastic media. Engineering Geology, 2018, 240: 189−203
15. Zhou S, Zhuang X, Zhu H, Rabczuk T. Phase field modeling of crack propagation, branching and coalescence in rocks. Theoretical and Applied Fracture Mechanics, 2018, 96: 174−192
16. Fielding A H. Machine Learning Methods for Ecological Applications. Boston: Springer, 1999, 1–35
17. Moayedi H, Hayati S. Modelling and optimization of ultimate bearing capacity of strip footing near a slope by soft computing methods. Applied Soft Computing, 2018, 66: 208−219
18. Anitescu C, Atroshchenko E, Alajlan N, Rabczuk T. Artificial neural network methods for the solution of second-order boundary value problems. Computers, Materials and Continua, 2019, 59(1): 345−359
19. Hoang N D, Pham A D. Hybrid artificial intelligence approach based on metaheuristic and machine learning for slope stability assessment: A multinational data analysis. Expert Systems with Applications, 2016, 46: 60−68
20. Liu Z, Shao J, Xu W, Chen H, Zhang Y. An extreme learning machine approach for slope stability evaluation and prediction. Natural Hazards, 2014, 73(2): 787−804
21. Samui P. Slope stability analysis: A support vector machine approach. Environmental Geology, 2008, 56(2): 255−267
22. Kothari U C, Momayez M. Machine learning: A novel approach to predicting slope instabilities. International Journal of Geophysics, 2018: 1−9
23. Eberhart R. Neural Network PC Tools: A Practical Guide. San Diego: Academic Press, 1990
24. Das S K, Biswal R K, Sivakugan N, Das B. Classification of slopes and prediction of factor of safety using differential evolution neural networks. Environmental Earth Sciences, 2011, 64(1): 201−210
25. Erzin Y, Cetin T. The prediction of the critical factor of safety of homogeneous finite slopes using neural networks and multiple regressions. Computers & Geosciences, 2013, 51: 305−313
26. Gelisli K, Kaya T, Babacan A E. Assessing the factor of safety using an artificial neural network: Case studies on landslides in Giresun, Turkey. Environmental Earth Sciences, 2015, 73(12): 1−8
27. Wu C I, Kung H Y, Chen C H, Kuo L C. An intelligent slope disaster prediction and monitoring system based on WSN and ANP. Expert Systems with Applications, 2014, 41(10): 4554−4562
28. Hoang N D, Bui D T. Slope Stability Evaluation Using Radial Basis Function Neural Network, Least Squares Support Vector Machines, and Extreme Learning Machine. Handbook of Neural Computation, 2017: 333−344
29. Koopialipoor M, Armaghani D J, Hedayat A, Marto A, Gordan B. Applying various hybrid intelligent systems to evaluate and predict slope stability under static and dynamic conditions. Soft Computing, 2019, 23(14): 5913−5929
30. Wang H B, Sassa K. Rainfall-induced landslide hazard assessment using artificial neural networks. Earth Surface Processes and Landforms, 2006, 31(2): 235−247
31. Pradhan B, Lee S, Buchroithner M F. A GIS-based back-propagation neural network model and its cross-application and validation for landslide susceptibility analyses. Computers, Environment and Urban Systems, 2010, 34(3): 216−235
32. Melchiorre C, Matteucci M, Azzoni A, Zanchi A. Artificial neural networks and cluster analysis in landslide susceptibility zonation. Geomorphology, 2008, 94(3–4): 379−400
33. Li A J, Khoo S, Lyamin A V, Wang Y. Rock slope stability analyses using extreme learning neural network and terminal steepest descent algorithm. Automation in Construction, 2016, 65(5): 42−50
34. Pradhan B. A comparative study on the predictive ability of the decision tree, support vector machine and neuro-fuzzy models in landslide susceptibility mapping using GIS. Computers & Geosciences, 2013, 51: 350−365
35. Shi X Z, Zhou J, Zheng W, Hu H Y, Wang H Y. Bayes discriminant analysis method and its application for prediction of slope stability. Journal of Sichuan University: Engineering Science Edition, 2010, 42(3): 63−68 (in Chinese)
36. Yan X M, Li X B. Bayes discriminant analysis method for predicting the stability of open pit slope. In: International Conference on Electric Technology & Civil Engineering. Lushan: IEEE, 2011
37. Cheng M Y, Hoang N D. Slope collapse prediction using Bayesian framework with k-nearest neighbor density estimation: Case-study in Taiwan. Journal of Computing in Civil Engineering, 2016, 30(1): 04014116
38. Li X Z, Kong J M, Wang C H. Application of multi-classification support vector machine in the identifying of landslide stability. Journal of Jilin University, 2010, 40(3): 631−637 (in Chinese)
39. Zhao H, Yin S, Ru Z. Relevance vector machine applied to slope stability analysis. International Journal for Numerical and Analytical Methods in Geomechanics, 2012, 36(5): 643−652
40. Zhou J, Li E, Yang S, Wang M, Shi X, Yao S, Mitri H S. Slope stability prediction for circular mode failure using gradient boosting machine approach based on an updated database of case

histories. Safety Science, 2019, 118: 505−518
41. Qi C, Tang X. Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. Computers & Industrial Engineering, 2018, 118: 112−122
42. Duncan J M. State of the art: Limit equilibrium and finite-element analysis of slopes. Journal of Geotechnical Engineering, 1996, 122(7): 577−596
43. Swan C C, Seo Y K. Limit state analysis of earthen slopes using dual continuum/FEM approaches. International Journal for Numerical and Analytical Methods in Geomechanics, 1999, 23(12): 1359−1371
44. Seo Y K. Computational methods for elastoplastic slope stability analysis with seepage. Dissertation for the Doctoral Degree. Iowa City: University of Iowa, 1998
45. Sah N K, Sheorey P R, Upadhyaya L N. Maximum likelihood estimation of slope stability. International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts, 1994, 31(1): 47−53
46. Fattahi H. Prediction of slope stability using adaptive neuro-fuzzy inference system based on clustering methods. Journal of Mining and Environment, 2017, 8(2): 163−177
47. Wang H B, Xu W Y, Xu R C. Slope stability evaluation using backpropagation neural networks. Engineering Geology, 2005, 80(3–4): 302−315
48. Manouchehrian A, Gholamnejad J, Sharifzadeh M. Development of a model for analysis of slope stability for circular mode failure using genetic algorithm. Environmental Earth Sciences, 2014, 71(3): 1267−1277
49. Lu P, Rosenbaum M S. Artificial neural networks and grey systems for the prediction of slope stability. Natural Hazards, 2003, 30(3): 383−398
50. Li J, Wang F. Study on the forecasting models of slope stability under data mining. In: The 12th Biennial International Conference on Engineering, Science, Construction, and Operations in Challenging Environments. Honolulu: ASCE, 2010, 765–776
51. Gelisli K, Kaya T, Babacan A E. Assessing the factor of safety using an artificial neural network: case studies on landslides in Giresun, Turkey. Environmental Earth Sciences, 2015, 73(12): 1−8
52. Chakraborty A, Goswami D. Prediction of slope stability using multiple linear regression (MLR) and artificial neural network (ANN). Arabian Journal of Geosciences, 2017, 10(17): 1−11
53. Feng X T, Wang Y J, Lu S Z. Neural network estimation of slope stability. Journal of Engineering Geology, 1995, 3(4): 54−61
54. Kostić S, Vasović N, Todorović K, Samčović A. Application of artificial neural networks for slope stability analysis in geotechnical practice. In: 2016 13th Symposium on Neural Networks and Applications (NEUREL). Belgrade: IEEE, 2016, 1–6
55. Fattahi H. Prediction of slope stability using adaptive neuro-fuzzy inference system based on clustering methods. Journal of Mining and Environment, 2017, 8(2): 163−177
56. Zhang Z, Liu Z, Zheng L, Zhang Y. Development of an adaptive relevance vector machine approach for slope stability inference. Neural Computing & Applications, 2014, 25(7−8): 2025−2035
57. Band S S, Janizadeh S, Saha S, Mukherjee K, Bozchaloei S K, Cerdà A, Shokri M, Mosavi A. Evaluating the efficiency of different regression, decision tree, and bayesian machine learning algorithms in spatial piping erosion susceptibility using ALOS/PALSAR data. Land (Basel), 2020, 9(10): 346−368
58. Tang Z H, Maclennan J. Data Mining with SQL Server 2005. Indianapolis: Wiley Publishing, Inc., 2005
59. Akgun A. A comparison of landslide susceptibility maps produced by logistic regression, multi-criteria decision, and likelihood ratio methods: A case study at İzmir, Turkey. Landslides, 2012, 9(1): 93−106
60. Ogutu J O, Schulz-Streeck T, Piepho H P. Genomic selection using regularized linear regression models: Ridge regression, lasso, elastic net and their extensions. BMC Proceedings, 2012, 6(2): 1−6
61. Kramer O. K-Nearest Neighbors. Berlin: Springer, 2013
62. Boser B E, Guyon I M, Vapnik V N. A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. NewYork: Association for Computing Machinery, 1992, 144–152
63. Myles A J, Feudale R N, Liu Y, Woody N A, Brown S D. An introduction to decision tree modeling. Journal of Chemometrics: A Journal of the Chemometrics Society, 2004, 18(6): 275−285
64. Murthy S K. Automatic construction of decision trees from data: A multi-disciplinary survey. Data Mining and Knowledge Discovery, 1998, 2(4): 345−389
65. Breiman L. Random forests. Machine Learning, 2001, 45(1): 5−32
66. Friedman J H. Greedy function approximation: A gradient boosting machine. Annals of Statistics, 2001, 29(5): 1189−1232
67. Li X Y, Li W D, Yan X. Human age prediction based on DNA methylation using a gradient boosting regressor. Genes, 2018, 9(9): 424−439
68. Simm J, Abril I M. ExtraTrees: Extremely Randomized Trees (Extra Trees) Method for Classification and Regression. R Package Version 1.0. 5. 2014
69. Landis J R, Koch G G. The measurement of observer agreement for categorical data. Biometrics, 1977, 33(1): 159−174