



*Research article*

## **Stability risk assessment of slopes using logistic model tree based on updated case histories**

**Feezan Ahmad<sup>1,\*</sup>, Xiao-Wei Tang<sup>1,\*</sup>, Mahmood Ahmad<sup>2,3</sup>, Roberto Alonso González-Lezcano<sup>4</sup>, Ali Majidi<sup>5</sup> and Mohamed Moafak Arbili<sup>6</sup>**

<sup>1</sup> State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian 116024, China

<sup>2</sup> Institute of Energy Infrastructure, University Tenaga Nasional, Kajang 43000, Malaysia

<sup>3</sup> Department of Civil Engineering, University of Engineering and Technology Peshawar (Bannu Campus), Bannu 28100, Pakistan

<sup>4</sup> Department of Architecture and Design, Escuela Politécnica Superior, Universidad San Pablo-CEU (Montepríncipe Campus), CEU Universities, Madrid 28668, Spain

<sup>5</sup> Department of Building and Construction Techniques Engineering, Al-Mustaqbal University, Hilla 51001, Iraq

<sup>6</sup> Department of Technical Civil Engineering, Erbil Technical Engineering College, Erbil Polytechnic University, Erbil 44001, Iraq

\* **Correspondence:** Email: [ahmadf@mail.dlut.edu.cn](mailto:ahmadf@mail.dlut.edu.cn), [tangxw@dlut.edu.cn](mailto:tangxw@dlut.edu.cn).

**Abstract:** A new logistic model tree (LMT) model is developed to predict slope stability status based on an updated database including 627 slope stability cases with input parameters of unit weight, cohesion, angle of internal friction, slope angle, slope height and pore pressure ratio. The performance of the LMT model was assessed using statistical metrics, including accuracy (Acc), Matthews correlation coefficient (Mcc), area under the receiver operating characteristic curve (AUC) and F-score. The analysis of the Acc together with Mcc, AUC and F-score values for the slope stability suggests that the proposed LMT achieved better prediction results (Acc = 85.6%, Mcc = 0.713, AUC = 0.907, F-score for stable state = 0.967 and F-score for failed state = 0.923) as compared to other methods previously employed in the literature. Two case studies with ten slope stability events were used to verify the proposed LMT. It was found that the prediction results are completely consistent with the actual situation at the site. Finally, risk analysis was carried out, and the result also agrees with the actual conditions. Such probability results can be incorporated into risk analysis with the

corresponding failure cost assessment later.

**Keywords:** logistic model tree; machine learning; slope stability; risk analysis; performance metrics

---

## 1. Introduction

Natural hazards such as landslides and subsidence have been recognized as important impediments to developing nations' sustainable development [1–4]. For example, on December 20, 2015, in Guangming, Shenzhen, China, a catastrophic landfill slope landslide claimed the lives of 69 individuals [5]. Natural hazard risk assessment and management will have short-term benefits in terms of severity reduction and long-term benefits in terms of achieving sustainable development goals [1].

Slope stability evaluation is essential for analyzing and mitigating natural hazards in mountainous environments. Many attempts have been made to evaluate slope stability [6–8]. Owing to the inherent complexity and uncertainty, assessing slope stability for circular mode failure, a common problem, remains a challenge for practitioners and researchers [9]. Several methods for evaluating slope stability have been presented, with the limit equilibrium approach and numerical simulation method being the two most commonly used [10]. Limited equilibrium methods, such as the simplified Bishop, Spencer and Morgenstern-Price methods, are frequently implemented in practice. In general, soil material properties (unit weight, cohesion and friction angle) and the pore pressure ratio are required for limited equilibrium methods [11,12]. Numerical methods (e.g., finite element methods) have been extensively used to analyze slope stability. However, the major drawback is that their input parameters need to be back analyzed using in-situ measurements, which is not available in many cases [13]. Both methods have pros and cons. Finding the critical slip surface using the limit equilibrium method is difficult due to the large number of potential slip surfaces [14]. The numerical simulation method's accuracy is greatly influenced by the choice of constitutive models, mechanical parameters and boundary conditions, and it is frequently necessary to have a great deal of engineering expertise and to conduct on-site back analysis in order to make a reasonable choice and obtain reasonable results [15,16]. Consequently, predicting slope stability still presents considerable challenges.

In recent years, machine learning (ML) models have gained attention for solving very complex, nonlinear and multivariable geotechnical problems [17–19]. Assessments of slope stability circular failure using methods based on soft computing are summarized in Table 1. Despite their reliable and precise outputs, however, most algorithms are not readily applicable in practice owing to their complicated training and modeling procedures and “black box” aspects, i.e., these models do not demonstrate a transparent and understandable relationship between inputs and output. Quinlan [20] developed the model tree algorithm to get over these limitations. It integrates principles from decision trees and linear regression. In addition to the widespread application of soft computing techniques, several studies have only been conducted on a limited amount of data, which might restrict the classifier's ability to generalize. In the current study, an updated database of 627 cases comprising unit weight, cohesion, internal friction angle, slope angle and height, pore pressure ratio and stability status of circular mode failure has been compiled. To predict slope stability, a new logistic model tree (LMT) model is developed. This algorithm is an intelligent choice for classification and decision-making since it can solve the classification problem by combining a tree model and a logistic regression (LR) technique. Adding LR to the leaves of the tree allows for a probabilistic interpretation of the model's output,

making it easier to explain, as they represent a series of if-then-else rules. The LMT has been employed in predicting pillar stability in geotechnical engineering [21], but it has not yet been used to predict slope stability.

**Table 1.** A summary of the ML-based circular mode failure assessment of slope stability.

Dataset (Stable/Failed)	Input parameters	Data Preprocessing	ML techniques	Reference
422 (226/196)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	MDMSE	Zhang et al. [22]
444 (224/220)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	AdaBoost, GBM, Bagging, XRT, RF, HGB, Voting, Stacked	Lin et al. [23]
19 (13/6)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	K-means cluster	Haghshenas et al. [24]
153 (83/70)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization and outlier removing	KNN, SVM, SGD, GP, QDA, GNB, DT, ANN, Bagging ensemble, Heterogeneous ensemble	Pham et al. [25]
257 (123/134)	$\gamma, c, \phi, \beta, H, r_u$	–	XGB, RF, LR, SVM, BC, LDA, KNN, DT, MLP, GNB, XRT, Stacked ensemble	Kardani et al. [26]
87 (42/45)	$\gamma, c, \phi, \beta, H, r_u$	–	J48	Amirkiyaei and Ghasemi [27]
221 (115/106)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	ANN, SVM, RF, GBM	Zhou et al. [28]
148 (78/70)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	LR, DT, RF, GBM, SVM, BP	Qi and Tang [29]
168 (84/84)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	GP, QDA, SVM, ADB-DT, ANN, KNN Classifier ensemble	Qi and Tang [15]
107 (48/59)	$\gamma, c, \phi, \beta, H, r_u$	–	RF, SVM, Bayes, GSA	Lin et al. [30]
82 (49/33)	$\gamma, c, \phi, \beta, H, r_u$	–	NB	Feng et al. [31]
168 (84/84)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	RBF, LSSVM, ELM	Hoang and Bui [32]
168 (84/84)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	LSSVM	Hoang and Pham [33]
46 (17/29)	$\gamma, c, \phi, \beta, H, r_u$	Data normalization	SVM	Xue et al. [34]
32 (14/18)	$\gamma, c, \phi, \beta, H, r_u$	–	ANN	Lu and Rosenbaum [35]
82 (38/44)	$\gamma, c, \phi, \beta, H, r_u$	–	BP	Feng [36]

Note: MDMSE-margin distance minimization selective ensemble, AdaBoost-adaptive boosting, GBM-gradient boosting machine, XRT-extremely randomized tree, RF-random forest, HGB-hist gradient boosting classifier, KNN-k-nearest neighbors, SVM-support vector machine, SGD-stochastic gradient descent, GP-Gaussian process, QDA-quadratic discriminant analysis, GNB-Gaussian naive Bayes, DT-decision tree, ANN-artificial neural network, XGB-extreme gradient boosting, BC-bagging classifier, LDA-linear discriminant analysis, MLP-multilayer perceptron, BP-back-propagation, QDA-quadratic discriminant analysis, ADB-DT-adaptive boosted decision tree, GSA-gravitational search algorithm, NB-naive Bayes, RBF-radial basis function, LSSVM-least squares support vector machine, ELM-extreme learning machine,  $\gamma$ -unit weight,  $c$ -cohesion,  $\phi$ -angle of internal friction,  $\beta$ -slope angle,  $H$ -slope height,  $r_u$ -pore pressure ratio.

## 2. Materials and methods

### 2.1. LMT

LR is a straightforward method with features such as stability, low variance and time-efficient training [37], but its prediction outputs are frequently biased. Decision trees are another ML technique

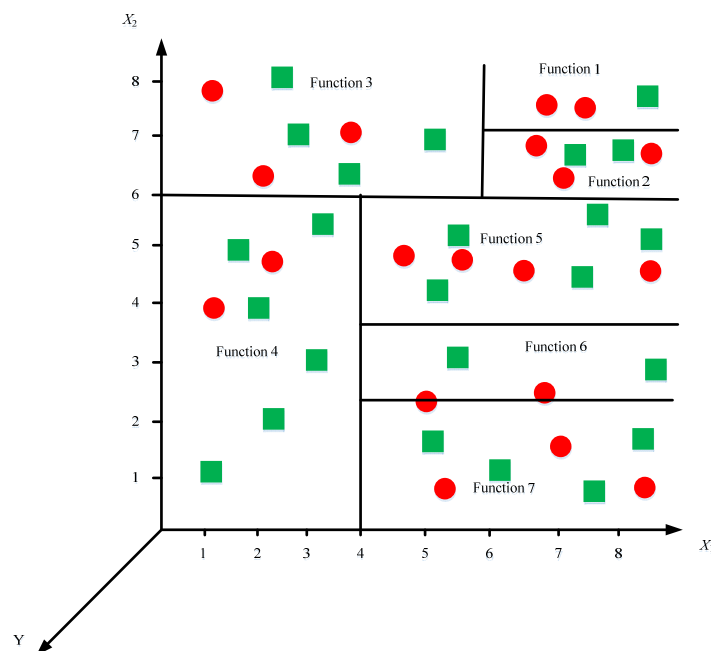
for searching a less confined space of candidate models and obtaining nonlinear patterns from a database; they have low bias but high variance and instability, making them susceptible to overfitting. Landwehr et al. [38] presented the LMT methodology as a result. It is based on Quinlan's model tree approach [20] for dealing with regression problems by combining linear regression with decision tree models, and it is extended to classification problems. This section provides a basic introduction to LMT, whereas the seminal work by Landwehr et al. [38] provides a more complete description.

## 2.2. Tree structure

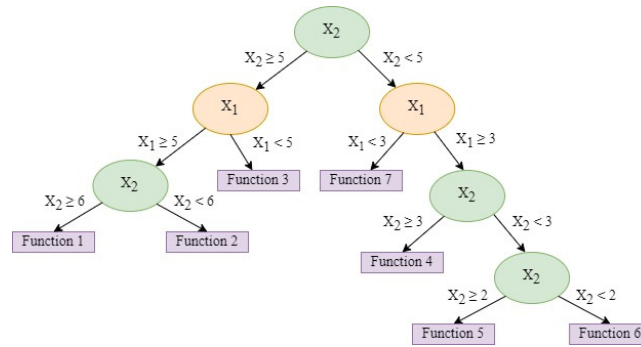
A LR tree with LR functions built at the leaves constitutes an LMT. It has a set of leaves or terminal nodes  $T$  and an inner or non-terminal node set  $N$ . Each leaf  $t$  of the LMT model has correlated LR functions instead of classification labels or linear regression functions. The output vector is  $Y$ , and the input vector  $X$  is  $(X_1, X_2)$ , for instance.  $S$  represents the complete instance space, which can be further partitioned into numerous subspaces  $S_t$ . Figure 1 displays a simple input space that has been partitioned into seven subspaces.

$$S = \bigcup_{t \in T} S_t, S_t \cap S_{t'} = \emptyset \quad (1)$$

The model determines LR functions for the seven subspaces represented in Figure 1. Figure 2 depicts the structure of the tree.



**Figure 1.** The input space is separated into seven subspaces (solid circles and square points represent different classes within the same case, and each subspace has its own function).



**Figure 2.** Example of a simplified LMT structure corresponding to the sample data shown in Figure 1 (adapted from Landwehr et al. [38]).

### 2.3. Logistic function

In contrast to standard forms of LR, the LogitBoost technique for fitting additive LR models proposed by Friedman et al. [39] is employed for model construction here. The prediction probability is given by Eq (2).

$$\Pr(G = j | X = x) = \frac{e^{F_j(x)}}{\sum_{k=1}^J e^{F_k(x)}} \quad (2)$$

where  $G$  denotes the output,  $J$  denotes the class labels,  $X$  denotes the inputs, and  $F_j(x)$  denotes the functions that the LMT will train in the tree's leaves, as follows:

$$F_j^t(x) = \sum_{m=1}^M f_{mj}^t(x) = \alpha_0^j + \sum_{S \in S_t} \alpha_S^j \cdot S \quad (3)$$

where  $m$  is the number of iterations,  $f_{mj}$  represents the functions of input variables,  $\alpha$  represents the intercepts and coefficients of the linear function, and  $S$  represents the variables of the subset  $S_t$  at the leaf  $t$ .

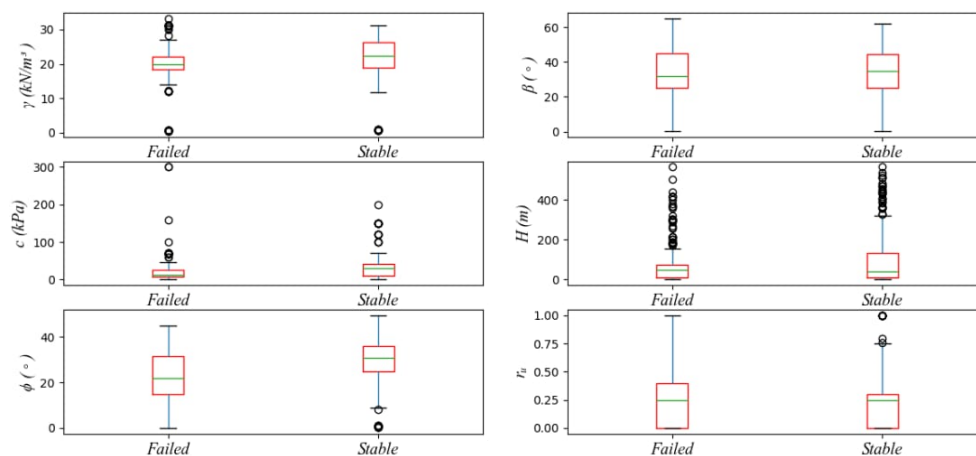
### 2.4. LMT training

A LMT can be established by the following steps: initial tree growth, tree splitting and stopping, and tree pruning. This section presents the basic idea; the reader is referred to Landwehr et al. [38] for additional detail. The M5P technique, which is commonly used for tree growth, can first construct a standard tree, after which a LR model can be established at each node [40,41]. This technique merely trains the model using case histories at each node in isolation, without taking the surrounding tree structure into account; therefore, in order for the LogitBoost algorithm to iteratively change  $F_j(x)$  to increase the fit in a natural way, another technique—one that can incrementally refine logistic model fit at high levels—is used [38]. The function  $f_{mj}$  is introduced to  $F_j$  by changing one of the function's coefficients or introducing another variable (see Eq (3)). As a result, in the initial growing process, a LR tree is formed in the root using proper iteration numbers. The tree then begins to grow by resembling specified subsets ( $t$ ) from the database ( $S$ ) to the child nodes, utilizing the C4.5 splitting law [20] to increase the accuracy of the classification variable. The LR functions are generated in the

child nodes by running the LogitBoost algorithm with the logistic model, weights and probability estimate from the previous iteration at the parent node. The splitting process is then repeated. When it comes to model fitting, the tree stops splitting when a node has less than 15 cases. After the tree is constructed, the tree pruning approach is used to trade off tree size and model complexity while maintaining predictive accuracy. After experimenting with several pruning initiatives, Landwehr et al. [38] employed the classification and regression trees pruning approach [42] to make pruning decisions while taking training error and model complexity into account. These three processes can be used to create a LMT.

## 2.5. Data catalog

An updated database with 627 instances was obtained from previous studies [22,23,25,29,33,43] and can be found in Table S1 of the supplementary information file. The database includes the unit weight, cohesion, angle of internal friction, slope angle and height, pore pressure ratio and slope stability status. There are 311 positive (stable) and 316 negative (failed) samples. The statistics of the input features are summarized in Table 2. The data normalization is not carried out owing to the fact that tree-based methods are insensitive to feature scaling; they make decisions based on relative feature values and splits [44]. The database box plot is shown in Figure 3, where solid black spots represent “outliers”, and the statistical features are shown in Table 2. The bottom and top quartiles are shown by horizontal lines, while the median values are represented by bold lines inside the boxes. Slopes with “failed” and “stable” instances are also demonstrated separately.



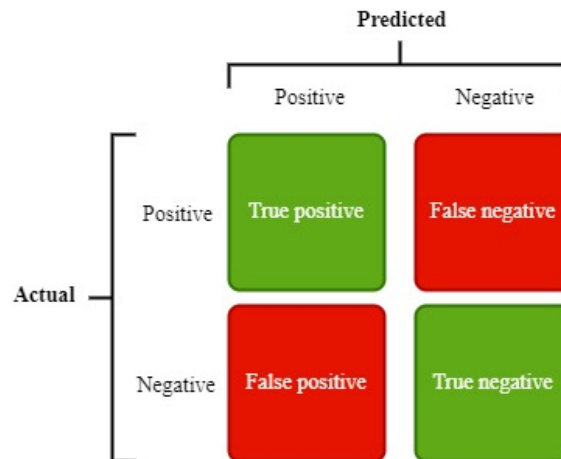
**Figure 3.** Box plots show the variation among data points in the database.

**Table 2.** Statistical values for features in the database.

Input parameter	Unit	Min.	Max.	Mean	Std. Dev.
Unit weight ( $\gamma$ )	$\text{kN/m}^3$	0.492	33.16	20.185	7.044
Cohesion ( $c$ )	kPa	0	300	25.6	31.036
Angle of internal friction ( $\phi$ )	–	0	49.5	25.308	12.331
Slope angle ( $\beta$ )	–	0.302	65	32.605	13.711
Slope height ( $H$ )	m	0.018	565	90.289	120.14
Pore pressure ratio ( $r_u$ )	–	0	1	0.254	0.26

## 2.6. Performance metrics

The classification metrics include accuracy evaluation indices (accuracy (Acc), Matthews correlation coefficient (Mcc), precision (Prec), recall (Rec) and F-score) that are calculated from the confusion matrix (see Figure 4).



**Figure 4.** Confusion matrix ( $2 \times 2$ ) for classification problem.

Each row of the matrix represents the instances in an actual class, while each column represents the instances in a predicted class [45]. In predictive analytics, a confusion matrix is described as a table with two rows and two columns that provides the numbers of true positives (TPs), true negatives (TNs), false positives (FPs) and false negatives (FNs). The classification evaluation metrics derived from the confusion matrix results were used to compare the prediction performances of the models [46–48]:

$$Acc = \frac{TN + TP}{TN + FP + FN + TP} \quad (4)$$

$$Prec = \frac{TN}{FN + TN} \text{ or } \frac{TP}{TP + FP} \quad (5)$$

$$Rec = \frac{TN}{TN + FP} \text{ or } \frac{TP}{TP + FN} \quad (6)$$

$$F\text{-score} = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (7)$$

$$Mcc = \frac{TN \times TP - FP \times FN}{\sqrt{(FN + TP)(FP + TN)(FP + TP)(FN + TN)}} \quad (8)$$

Acc is the sum of TP and TN and represents the percentage of correctly classified instances in the data as a whole. This metric measures the model's total prediction accuracy. If the data set is

unbalanced, that is, the numbers of observations in different classes vary substantially, accuracy will be deceiving [49]. As a result, further assessment metrics such as precision, recall,  $F$ -score and Mcc were utilized to examine the model's performance further. Precision is known as positive predicted value, and recall is known as true positive rate (TPR).  $F$ -score is a generalized index that evaluates the performance of both recall and precision and ranges from 0 (worst value) to 1 (best value). Mcc denotes the degree of agreement between observed and predicted classes of failed and stable instances [48]. It is a standard metric used by statisticians that accepts values ranging from  $-1$  to  $1$ . Mcc value of  $-1$  indicates complete disagreement (strong negative association), a value of  $1$  indicates complete agreement (strong positive association), and a value of  $0$  indicates that the prediction was unrelated to the ground truth (very weak or no correlation between dependent and independent variables). Additionally, another succinct metric, i.e., area under the receiver operating characteristic curve (AUC), was employed for effective evaluation. This curve is expressed in a graphical plot that shows a binary classification system's diagnostic capacity as its discrimination threshold changes. The AUC is plotted by comparing the TPR vs. the false positive rate (FPR) at various threshold levels. An acceptable classification model should have an AUC near to  $1$ . Table 3 shows the main rule for defining discrimination based on AUC value [25].

**Table 3.** Rule for classifying discrimination based on AUC value.

Discrimination category	AUC value
No discrimination	$\text{AUC} = 0.5$
Acceptable	$0.7 \leq \text{AUC} < 0.8$
Excellent	$0.8 \leq \text{AUC} < 0.9$
Outstanding	$0.9 \leq \text{AUC}$

### 3. Results and discussion

#### 3.1. Development of LMT models for slope stability prediction

Waikato Environment for Knowledge Analysis (WEKA) software was used for developing a model based on the acquired data set. WEKA is a collection of ML algorithms that supports data mining tasks by providing a wide range of tools that can be used for data pre-processing, classification, clustering, regression, association and visualization [50]. The database was divided into two parts: training (80%) and testing (20%). The training set contained 249 stable instances and 253 failed cases, while the test set contained 125 instances, 62 of which were stable and 63 of which were failed instances. The stable-to-failed instance ratios in the training and testing sets were close to one, indicating that the distribution of these two instances does not necessitate a cost-sensitive technique to address problem due to imbalance [51,52].

Every terminal node (or leaf) of the tree model was trained and updated using LR models during training (see Section 2). The minimal number of instances for the LMT model was determined to be 15 based on predictive performance, readily applicable tree structure and total training data. The size of the tree was 45, and it had 23 leaves. LogitBoost had a weight trimming value of 0.2 and a number of iterations of  $-1$ . Figure 5 depicts a representation of a tree generated by LMT. The LMT model has 23 logistic functions (LMs), and their detailed expressions are shown in Table 4. It is important to notice that some of the functions in Table 4 do not incorporate all of the parameters that have been chosen.



The LM1 function for stable slopes in Table 4, for example, does not account for unit weight ( $\gamma$ ). The simple logistic technique is used in the LMT training phase [38]. The goal of the simple logistic method is to control the parameter numbers as simply and straightforwardly as possible. New parameters are gradually introduced during training in order to improve the performance of each function at each node of the tree (see Section 2). This can also help to avoid the issue of model significance in LR, especially when using multiple logistic functions to build a full logistic model with all parameters. However, just a few of the functions had fewer parameters than those selected, showing that most of the factors we chose have an influence on predictive performance.

**Table 4.** Logistic functions in the LMT model.

No.	Regression model for the slope stability condition	No.	Regression model for the slope stability condition
LM1	$F_S = 0.3 + 14.53c + 4.62\phi - 5.83\beta - 9.17H - 0.69r_u$ $F_F = -0.3 - 14.53c - 4.62\phi + 5.83\beta + 9.17H + 0.69r_u$	LM13	$F_S = -1.87 - 0.01c - 0.03\phi + 0.04\beta + 0H + 1.3r_u$ $F_F = 1.87 + 0.01c + 0.03\phi - 0.04\beta - 0H - 1.3r_u$
LM2	$F_S = 0.36 - 0.04\gamma + 0.02c - 0.04\phi + 0.13\beta - 0.05H$ $F_F = -0.36 + 0.04\gamma - 0.02c + 0.04\phi - 0.13\beta + 0.05H$	LM14	$F_S = -6.73 + 0.22\gamma - 0.01c - 0.03\phi + 0.05\beta - 0H + 1.3r_u$ $F_F = 6.73 - 0.22\gamma + 0.01c + 0.03\phi - 0.05\beta + 0H - 1.3r_u$
LM3	$F_S = 0.73 - 0.07\gamma - 0.02c - 0.13\phi + 0.11\beta + 0.08H + 3.03r_u$ $F_F = -0.73 + 0.07\gamma + 0.02c + 0.13\phi - 0.11\beta - 0.08H - 3.03r_u$	LM15	$F_S = -10.69 - 0.01c - 0.09\phi + 0.28\beta + 0H + 2.97r_u$ $F_F = 10.69 + 0.01c + 0.09\phi - 0.28\beta - 0H - 2.97r_u$
LM4	$F_S = -5.08 + 0.02\gamma - 0.02c - 0.1\phi + 0.28\beta + 0H - 8.66r_u$ $F_F = 5.08 - 0.02\gamma + 0.02c + 0.1\phi - 0.28\beta - 0H + 8.66r_u$	LM16	$F_S = -8.31 - 0.01c + 0.02\phi + 0.02\beta + 0H - 1.59r_u$ $F_F = 8.31 + 0.01c - 0.02\phi - 0.02\beta - 0H + 1.59r_u$
LM5	$F_S = 7.18 + 0.01\gamma - 0.01c - 0.04\phi + 0\beta$ $F_F = -7.18 - 0.01\gamma + 0.01c + 0.04\phi - 0\beta$	LM17	$F_S = -2.05 - 0.01c + 0.02\phi + 0.02\beta + 0H - 1.67r_u$ $F_F = 2.05 + 0.01c - 0.02\phi - 0.02\beta - 0H + 1.67r_u$
LM6	$F_S = -4.4 - 0.51\gamma + 0.03\phi + 0.09\beta + 0.01H$ $F_F = 4.4 + 0.51\gamma - 0.03\phi - 0.09\beta - 0.01H$	LM18	$F_S = 7.6 + 0.2\gamma - 0.04c - 0.1\phi - 0.01\beta - 0.08H - 1.42r_u$ $F_F = -7.6 - 0.2\gamma + 0.04c + 0.1\phi + 0.01\beta + 0.08H + 1.42r_u$
LM7	$F_S = 27.08 - 1.49\gamma - 0.09c - 0.17\phi - 0.22\beta - 0.02H + 23.63r_u$ $F_F = -27.08 + 1.49\gamma + 0.09c + 0.17\phi + 0.22\beta + 0.02H - 23.63r_u$	LM19	$F_S = -4.4 + 0.58\gamma - 0.04c - 0.07\phi - 0.01\beta + 0.01H - 1.72r_u$ $F_F = 4.4 - 0.58\gamma + 0.04c + 0.07\phi + 0.01\beta - 0.01H + 1.72r_u$
LM8	$F_S = -9.31 + 0.08\gamma - 0.03c + 0.01\phi + 0.03\beta - 0H + 0.31r_u$ $F_F = 9.31 - 0.08\gamma + 0.03c - 0.01\phi - 0.03\beta + 0H - 0.31r_u$	LM20	$F_S = 5.13 + 0.25\gamma - 0.11c - 0.05\phi + 0.08\beta - 0.01H + 5.74r_u$ $F_F = -5.13 - 0.25\gamma + 0.11c - 0.05\phi - 0.08\beta + 0.01H - 5.74r_u$
LM9	$F_S = 11.33 + 0.03\gamma + 0.02c - 0.04\phi - 0.31\beta + 0.02H - 23.05r_u$ $F_F = -11.33 - 0.03\gamma - 0.02c + 0.04\phi + 0.31\beta - 0.02H + 23.05r_u$	LM21	$F_S = -10.75 + 0.74\gamma - 0c - 0.16\phi + 0.09\beta - 0H - 2.68r_u$ $F_F = 10.75 - 0.74\gamma + 0c + 0.16\phi - 0.09\beta + 0H + 2.68r_u$
LM10	$F_S = 4.75 - 0.21\gamma - 0.01c - 0.06\phi + 0.07\beta + 0H - 2.67r_u$ $F_F = -4.75 + 0.21\gamma + 0.01c + 0.06\phi - 0.07\beta - 0H + 2.67r_u$	LM22	$F_S = -90.05 + 1.93\gamma - 0.01c + 0.41\phi + 0.48\beta + 0H + 0.58r_u$ $F_F = 90.05 - 1.93\gamma + 0.01c - 0.41\phi - 0.48\beta - 0H - 0.58r_u$
LM11	$F_S = 0.17 - 0.01\gamma + 0c + 0.02\phi + 0\beta - 1.67r_u$ $F_F = -0.17 + 0.01\gamma - 0c - 0.02\phi - 0\beta + 1.67r_u$	LM23	$F_S = -98.1 + 2.52\gamma + 0.86c - 1.19\phi - 0\beta + 0.01H + 3.15r_u$ $F_F = 98.1 - 2.52\gamma - 0.86c + 1.19\phi + 0\beta - 0.01H - 3.15r_u$
LM12	$F_S = -43.04 - 0.01\gamma + 0c + 0.73\phi - 0.2\beta + 43.7r_u$ $F_F = 43.04 + 0.01\gamma - 0c - 0.73\phi + 0.2\beta - 43.7r_u$		

Note: Angle of internal friction ( $\phi$ ) and slope angle ( $\beta$ ) are in degrees.

### 3.2. Model performance evaluation

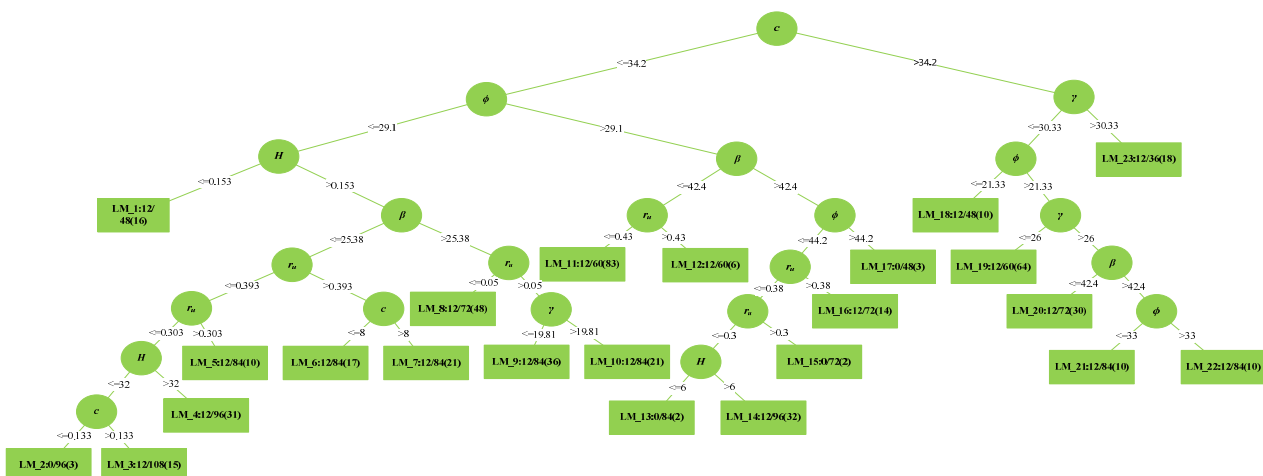
The proposed LMT model was quantified using several performance metrics based on confusion matrices. The confusion matrices of the model in training and test sets were then obtained, as shown in Table 5.

**Table 5.** Confusion matrices and performance metric results for training and testing datasets.

Dataset	Actual	Predicted		Acc (%)	Mcc	AUC	Prec	Rec	F-score
		Stable	Failed						
Training	Stable	237	12	92.23	0.846	0.974	0.898	0.952	0.924
	Failed	27	226						
Testing	Stable	51	11	85.60	0.713	0.907	0.879	0.823	0.850
	Failed	7	56						

The proposed LMT model predicts two classes (stable and failed). Table 5 displays the confusion matrices that illustrate training and prediction results. The number of cases that could have been correctly predicted is indicated by the values along the major diagonal. The results of Table 5 show that the LMT correctly classified the majority of the cases.

The prediction outcomes of classification problems can be assessed using a variety of metrics, such as Acc, Mcc, AUC, recall, precision and F-score. The LMT model performed well in both the training set (Acc = 92.23%, Mcc = 0.846, AUC = 0.974, F-score for stable state = 0.924 and F-score for failed state = 0.921) and testing set (Acc = 85.60%, Mcc = 0.713, AUC = 0.907, F-score for stable state = 0.850 and F-score for failed state = 0.862). The employed metrics show that the prediction's results are accurate and acceptable.

**Figure 5.** A LMT's structure.

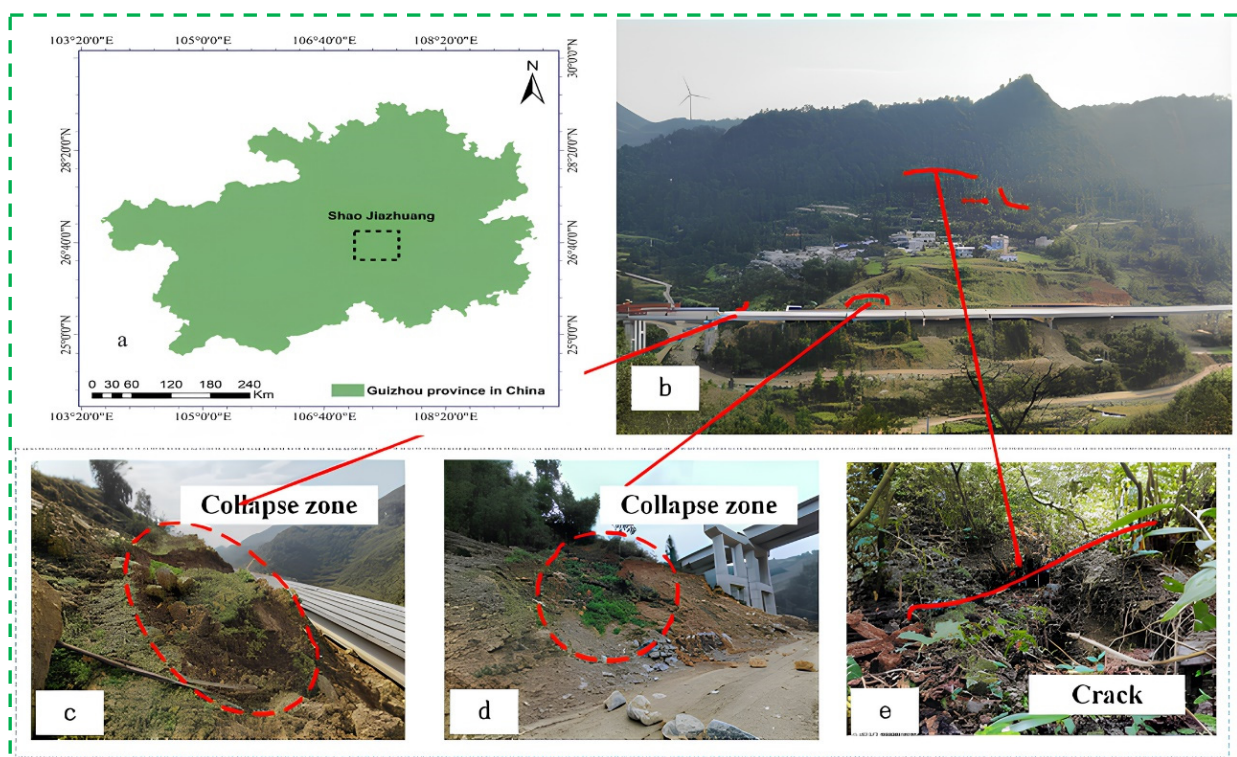
The LMT model achieved better prediction performance (Acc = 0.856 and AUC = 0.907) in comparison to the SVM model (Acc = 0.812 and AUC = 0.796), SGD model (Acc = 0.640 and AUC = 0.688), QDA model (Acc = 0.788 and AUC = 0.817), GNB model (Acc = 0.812 and AUC = 0.775), DT model (Acc = 0.788 and AUC = 0.829) and RT model (Acc = 0.788 and AUC = 0.904), reported by Pham et al. [25], respectively, for the test data. When using ML techniques to predict slope stability, Acc generally ranges from 0.640 to 0.812, according to the results of previous study by Pham et al. [25]. Meanwhile, in the present study, it is 0.856 for the test data set. However, due to the use of different data sets, a comparison between these results is unwarranted. A project that uses different data sets is needed to give a generalized model to predict slope stability. Additionally, the results for relatively small sample sizes (less than 100) are not presented or compared. These comparative results

demonstrate conclusively that the LMT model is capable of improved generalization performance compared to other models in the literature. The potential reasons underlying this observation are that the LMT model can capture interactions between features effectively, the branching structure of the decision tree can identify feature interactions, and LR at the leaves can model the relationships between these interactions and the target variable.

#### 4. Case studies

Two case studies are analyzed using our proposed LMT-based slope stability prediction model to determine its efficacy and applicability in engineering practice. The LMT model predicted ten slope stability status events in two different projects. The field data was obtained from the available literature, which includes the Shao Jiazhuang slope failure [53] and the Daguangbao landslide [54].

The slope at Shao Jiazhuang village in Guizhou province, China (see Figure 6 (a)) [53], is utilized as a real-world example to analyze slope stability with the proposed LMT model and validate its efficacy and practicality in engineering practice application. As depicted in Figure 6(c)–(e), during the survey, there were localized shallow surface collapses and new cracks on the slope's left side [53].



**Figure 6.** Slope failure area at Shao Jiazhuang, China: (a) location map, (b) slope collapse, (c), (d) collapse zone, and (e) crack.

The input data shown in Table 6 were used by the LMT to predict the stability of the case slope based on the material parameters and geometry feature parameters of the case slope. The slope is failed, according to the prediction's outcome. This study demonstrates that the LMT model is reliable for predicting slope stability and that it can be applied to a variety of geotechnical applications.

**Table 6.** Application of the proposed LMT model in real-world slope stability prediction.

S. No.	$\gamma$ (kN/m <sup>3</sup> )	$c$ (kPa)	$\phi$ (°)	$\beta$ (°)	$H$ (m)	$r_u$	Actual status	LMT predicted result
1	20	22.4	28	39.47	14	0	Failed	Failed

The Daguangbao landslide, is one of the few extremely big landslides known to exist worldwide, with a size of over 1 billion m<sup>3</sup> [55]. It is also the most extensive and largest landslide ever recorded in Chinese historical records. Because of its massive volume, unusual genetic mechanism and complex movement process [56–58], the Daguangbao landslide has drawn a lot of attention and interest.

The stability evaluation is carried out on 9 slopes, approved by specialists, from the Daguangbao landslide, of which 5 are stable, and 4 failed [54]. The index values of the samples and predicted results of the LMT model are compared to the fuzzy discriminant method and the unascertained measure method in Table 7.

**Table 7.** Predicted results using LMT, fuzzy discriminant and unascertained measure methods.

S. No.	$\gamma$ (kN/m <sup>3</sup> )	$c$ (kPa)	$\phi$ (°)	$\beta$ (°)	$H$ (m)	$r_u$	Actual status	LMT proposed method	Fuzzy discriminant method [54]	Unascertained measure method [54]
1	27	32	33	42.4	289	0.25	Stable	Stable	Stable	Stable
2	20.41	33	11	16	46	0.2	Failed	Failed	Stable*	Failed
3	21.43	0	20	20	61	0.5	Failed	Failed	Failed	Failed
4	19.63	11.97	20	22	21.19	0.405	Failed	Failed	Failed	Failed
5	18.68	26.34	15	35	8.23	0.25	Failed	Failed	Failed	Stable*
6	27	50	40	42	407	0.25	Stable	Stable	Stable	Stable
7	27.3	14	31	50	92	0.25	Stable	Stable	Stable	Failed*
8	21.4	10	30.34	30	20	0.25	Stable	Stable	Stable	Stable
9	25	46	35	46	393	0.25	Stable	Stable	Stable	Stable

Note: (\*) wrongly predicted.

The prediction outcomes in Table 7 indicate that the slope stability was predicted correctly for all cases. The theory of unascertained measure and its mathematical processing were first put forward by Wang [59] in 1990. On this basis, Liu et al. [60,61] established the unascertained mathematical theory and applied it to decision-making problems. Unascertained information is a new kind of uncertainty information which is different from fuzzy information, random information and grey information. Compared with other evaluation methods, the unascertained measure method has the advantages of non-negativity, normalization and additivity, which also ensures the order of evaluation space. A fuzzy discriminant method [62] is a decision classification to construct a numerical tabular knowledge base from historical cases, and it derives inferences from particular case histories using discrimination and connectivity analyses which are based on a theory of fuzzy relations. For further details regarding these methods, readers may refer to [54,59–62].

## 5. Risk analysis

To demonstrate a use of the proposed LMT model for risk analysis, we use the case history data

in Table 7. The following lists typical input data for the Shao Jiazhuang, China, slope failure area:  $\gamma = 20 \text{ kN/m}^3$ ,  $c = 22.4 \text{ kPa}$ ,  $\phi = 28^\circ$ ,  $\beta = 39.47^\circ$ ,  $H = 14 \text{ m}$ , and  $r_u = 0$ . Then, using the tree structure from Figure 5, we use the appropriate functions to compute the probability of failure ( $PoF$ ). For example, to make a prediction for a new input instance, start at the root node of the LMT model (e.g., the value of  $c$  in this case is  $22.4 \text{ kPa}$ ) and follow the path through the tree according to the splitting rules. Then, proceed to the end function at a leaf node (for this instance, LM8). The function values ( $F_S$  and  $F_F$ ) are first calculated using the LM8 equations in Table 4. Then, using Eq (2), the probability of slope stability is given by Eq (11):

$$F_S = -9.31 + 0.08\gamma - 0.03c + 0.01\phi + 0.03\beta - 0H + 0.31r_u \quad (9)$$

$$F_F = 9.31 - 0.08\gamma + 0.03c - 0.01\phi - 0.03\beta + 0H - 0.31r_u \quad (10)$$

$$P_S = \frac{e^{F_S}}{e^{F_S} + e^{F_F}}, P_F = \frac{e^{F_F}}{e^{F_S} + e^{F_F}} \quad (11)$$

Finally, we can get the results:  $P_S = 1\%$ , and  $P_F = 99\%$ , implying that this instance has a  $PoF$  equal to  $99\%$ . This also accords with the actual facts. Such probability results can then be incorporated into risk analysis with the associated failure cost estimate.

## 6. Conclusions

For the assessment of slope stability, a novel application of LMT is proposed. The tree structure and corresponding functions are used to assess slope stability, given information on several features, such as slope height ( $H$ ), slope angle ( $\beta$ ), cohesion ( $c$ ), internal friction angle ( $\phi$ ), unit weight ( $\gamma$ ) and pore pressure ratio ( $r_u$ ). LogitBoost learns the LMT by utilizing a larger database obtained from the literature. The results show that the LMT model can effectively predict slope stability. A testing set was used to validate the trained LMT model. Furthermore, real-world application to new cases—which had not previously been used for training—was used to validate the proposed LMT model. Comparative study and engineering application results show that the LMT model has the best prediction effect and is the best, optimal model. Furthermore, the results indicate that the LMT technique can provide useful information regarding the probability of slope failure, allowing it to be used for risk analysis of slope stability. Finally, the main advantages of LMT over other soft computing models commonly used for slope stability prediction are that it can be trained easily (even with more input parameters) and that its tree structure, with an LR function in each leaf, explicitly demonstrates the relationship between the inputs and predictive output. Furthermore, the LMT has the potential to be used to solve other geotechnical problems in the future due to its intuitive features and ease of implementation. It is recommended for future work that considering tangents (i.e.,  $\tan \phi$  and  $\tan \beta$ ) could improve performance.

### Use of AI tools declaration

The authors declare they have not used artificial intelligence (AI) tools in the creation of this article.

## Acknowledgments

This work was supported by the National Key Research and Development Plan of China under Grant No. 2021YFB2600703. The authors also wish to thank CEU San Pablo University Foundation for the funds dedicated to the ARIE Research Group, through Project Ref. EC01/0720-MGI23RGL, provided by the CEU San Pablo University.

## Conflict of interest

The authors declare there is no conflict of interest.

## References

1. K. Méheux, D. Dominey-Howes, K. Lloyd, Natural hazard impacts in small island developing states: A review of current knowledge and future research needs, *Nat. Hazards*, **40** (2007), 429–446. <https://doi.org/10.1007/s11069-006-9001-5>
2. S. Iai, *Geotechnics and Earthquake Geotechnics Towards Global Sustainability*, Springer, Dordrecht, 2011. <https://doi.org/10.1007/978-94-007-0470-1>
3. J. Ma, X. Liu, X. Niu, Y. Wang, T. Wen, J. Zhang, et al., Forecasting of landslide displacement using a probability-scheme combination ensemble prediction technique, *Int. J. Environ. Res. Public Health*, **17** (2020), 4788. <https://doi.org/10.3390/ijerph17134788>
4. X. Niu, J. Ma, Y. Wang, J. Zhang, H. Chen, H. Tang, A novel decomposition-ensemble learning model based on ensemble empirical mode decomposition and recurrent neural network for landslide displacement prediction, *Appl. Sci.*, **11** (2021), 4684. <https://doi.org/10.3390/app11104684>
5. C. Ouyang, K. Zhou, Q. Xu, J. Yin, D. Peng, D. Wang, et al., Dynamic analysis and numerical modeling of the 2015 catastrophic landslide of the construction waste landfill at Guangming, Shenzhen, China, *Landslides*, **14** (2017), 705–718. <https://doi.org/10.1007/s10346-016-0764-9>
6. J. M. Duncan, S. G. Wright, The accuracy of equilibrium methods of slope stability analysis, *Eng. Geol.*, **16** (1980), 5–17. [https://doi.org/10.1016/0013-7952\(80\)90003-4](https://doi.org/10.1016/0013-7952(80)90003-4)
7. D. Y. Zhu, C. F. Lee, H. D. Jiang, Generalised framework of limit equilibrium methods for slope stability analysis, *Géotechnique*, **53** (2003), 377–395. <https://doi.org/10.1680/geot.2003.53.4.377>
8. S. Y. Liu, L. T. Shao, H. J. Li, Slope stability analysis using the limit equilibrium method and two finite element methods, *Comput. Geotech.*, **63** (2015), 291–298. <https://doi.org/10.1016/j.compgeo.2014.10.008>
9. A. J. Li, R. S. Merifield, A. V. Lyamin, Limit analysis solutions for three dimensional undrained slopes, *Comput. Geotech.*, **36** (2009), 1330–1351. <https://doi.org/10.1016/j.compgeo.2009.06.002>
10. Y. Yang, W. Wu, H. Zheng, Stability analysis of slopes using the vector sum numerical manifold method, *Bull. Eng. Geol. Environ.*, **80** (2021), 345–352. <https://doi.org/10.1007/s10064-020-01903-x>
11. H. B. Wang, W. Y. Xu, R. C. Xu, Slope stability evaluation using back propagation neural networks, *Eng. Geol.*, **80** (2005), 302–315. <https://doi.org/10.1016/j.enggeo.2005.06.005>
12. L. Wang, Z. Chen, N. Wang, P. Sun, S. Yu, S. Li, et al., Modeling lateral enlargement in dam breaches using slope stability analysis based on circular slip mode, *Eng. Geol.*, **209** (2016), 70–81. <https://doi.org/10.1016/j.enggeo.2016.04.027>

13. C. Qi, A. Fourie, G. Ma, X. Tang, X. Du, Comparative study of hybrid artificial intelligence approaches for predicting hangingwall stability, *J. Comput. Civ. Eng.*, **32** (2018). [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000737](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000737)
14. Y. Yang, Y. Sun, G. Sun, H. Zheng, Sequential excavation analysis of soil-rock-mixture slopes using an improved numerical manifold method with multiple layers of mathematical cover systems, *Eng. Geol.*, **261** (2019), 105278. <https://doi.org/10.1016/j.enggeo.2019.105278>
15. C. Qi, X. Tang, A hybrid ensemble method for improved prediction of slope stability, *Int. J. Numer. Anal. Methods Geomech.*, **42** (2018), 1823–1839. <https://doi.org/10.1002/nag.2834>
16. A. Ray, V. Kumar, A. Kumar, R. Rai, M. Khandelwal, T. N. Singh, Stability prediction of Himalayan residual soil slope using artificial neural network, *Nat. Hazards*, **103** (2020), 3523–3540. <https://doi.org/10.1007/s11069-020-04141-2>
17. W. Zhang, H. Li, L. Tang, X. Gu, L. Wang, L. Wang, Displacement prediction of Jiuxianping landslide using gated recurrent unit (GRU) networks, *Acta Geotech.*, **17** (2022), 1367–1382. <https://doi.org/10.1007/s11440-022-01495-8>
18. W. Zhang, H. Li, L. Han, L. Chen, L. Wang, Slope stability prediction using ensemble learning techniques: A case study in Yunyang County, Chongqing, China, *J. Rock Mech. Geotech. Eng.*, **14** (2022), 1089–1099. <https://doi.org/10.1016/j.jrmge.2021.12.011>
19. L. Wang, C. Wu, X. Gu, H. Liu, G. Mei, W. Zhang, Probabilistic stability analysis of earth dam slope under transient seepage using multivariate adaptive regression splines, *Bull. Eng. Geol. Environ.*, **79** (2020), 2763–2775. <https://doi.org/10.1007/s10064-020-01730-0>
20. J. R. Quinlan, Learning with continuous classes, in *Proceedings of Australian Joint Conference on Artificial Intelligence*, (1992), 343–348.
21. Li, N., et al., Stability risk assessment of underground rock pillars using logistic model trees, *Int. J. Environ. Res. Public Health*, **19** (2022), 2136. <https://doi.org/10.3390/ijerph19042136>
22. H. Zhang, S. Wu, X. Zhang, L. Han, Z. Zhang, Slope stability prediction method based on the margin distance minimization selective ensemble, *CATENA*, **212** (2022), 106055. <https://doi.org/10.1016/j.catena.2022.106055>
23. S. Lin, H. Zheng, B. Han, Y. Li, C. Han, W. Li, Comparative performance of eight ensemble learning approaches for the development of models of slope stability prediction, *Acta Geotech.*, **17** (2022), 1477–1502. <https://doi.org/10.1007/s11440-021-01440-1>
24. S. S. Haghshenas, S. S. Haghshenas, Z. W. Geem, T. Kim, R. Mikaeil, L. Pugliese, et al., Application of harmony search algorithm to slope stability analysis, *Land*, **10** (2021), 1250. <https://doi.org/10.3390/land10111250>
25. K. Pham, D. Kim, S. Park, H. Choi, Ensemble learning-based classification models for slope stability analysis, *CATENA*, **196** (2021), 104886. <https://doi.org/10.1016/j.catena.2020.104886>
26. N. Kardani, A. Zhou, M. Nazem, S. Shen, Improved prediction of slope stability using a hybrid stacking ensemble method based on finite element analysis and field data, *J. Rock Mech. Geotech. Eng.*, **13** (2021), 188–201. <https://doi.org/10.1016/j.jrmge.2020.05.011>
27. V. Amirkiyaei, E. Ghasemi, Stability assessment of slopes subjected to circular-type failure using tree-based models, *Int. J. Geotech. Eng.*, **16** (2022), 301–311. <https://doi.org/10.1080/19386362.2020.1862538>
28. J. Zhou, E. Li, S. Yang, M. Wang, X. Shi, S. Yao, et al., Slope stability prediction for circular mode failure using gradient boosting machine approach based on an updated database of case histories, *Saf. Sci.*, **118** (2019), 505–518. <https://doi.org/10.1016/j.ssci.2019.05.046>

29. C. Qi, X. Tang, Slope stability prediction using integrated metaheuristic and machine learning approaches: a comparative study, *Comput. Ind. Eng.*, **118** (2018), 112–122. <https://doi.org/10.1016/j.cie.2018.02.028>
30. Y. Lin, K. Zhou, J. Li, Prediction of slope stability using four supervised learning methods, *IEEE Access*, **6** (2018), 31169–31179. <https://doi.org/10.1109/ACCESS.2018.2843787>
31. X. Feng, S. Li, C. Yuan, P. Zeng, Y. Sun, Prediction of slope stability using naive Bayes classifier, *KSCE J. Civ. Eng.*, **22** (2018), 941–950. <https://doi.org/10.1007/s12205-018-1337-3>
32. N. Hoang, D.T. Bui, Slope stability evaluation using radial basis function neural network, least squares support vector machines, and extreme learning machine, in *Handbook of Neural Computation*, Elsevier, (2017), 333–344. <https://doi.org/10.1016/B978-0-12-811318-9.00018-1>
33. N. Hoang, A. Pham, Hybrid artificial intelligence approach based on metaheuristic and machine learning for slope stability assessment: A multinational data analysis, *Expert Syst. Appl.*, **46** (2016) 60–68. <https://doi.org/10.1016/j.eswa.2015.10.020>
34. X. Xue, X. Yang, X. Chen, Application of a support vector machine for prediction of slope stability, *Sci. China Technol. Sci.*, **57** (2014), 2379–2386. <https://doi.org/10.1007/s11431-014-5699-6>
35. P. Lu, M. S. Rosenbaum, Artificial neural networks and grey systems for the prediction of slope stability, *Natural Hazards*, **30** (2003), 383–398. <https://doi.org/10.1023/B:NHAZ.0000007168.00673.27>
36. X. Feng, *Introduction of Intelligent Rock Mechanics*, Science Press, Beijing, 2000.
37. D. W. Hosmer Jr., S. Lemeshow, R. X. Sturdivant, *Applied Logistic Regression*, Wiley, New Jersey, 2013. <https://doi.org/10.1002/9781118548387>
38. N. Landwehr, M. Hall, E. Frank, Logistic model trees, *Mach. Learn.*, **59** (2005), 161–205. <https://doi.org/10.1007/s10994-005-0466-3>
39. J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: A statistical view of boosting, *Ann. Statist.*, **28** (2000), 337–407. <http://doi.org/10.1214/aos/1016218223>
40. E. Ghasemi, H. Kalhori, R. Bagherpour, S. Yagiz, Model tree approach for predicting uniaxial compressive strength and Young's modulus of carbonate rocks, *Bull. Eng. Geol. Environ.*, **77** (2018), 331–343. <https://doi.org/10.1007/s10064-016-0931-1>
41. Y. Wang, I. Witten, Inducing Model trees for continuous classes, in *Proceedings of the Ninth European Conference on Machine Learning*, (1997).
42. L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone, Classification and regression trees, *Biometrics*, **40** (1984), 874. <https://doi.org/10.2307/2530946>
43. M. G. Sakellariou, M. D. Ferentinou, A study of slope stability prediction using neural networks, *Geotech. Geol. Eng.*, **23** (2005), 419–445. <https://doi.org/10.1007/s10706-004-8680-5>
44. V. Chavan, How to choose the right machine learning algorithm, 2022. Available from: <https://medium.com/@vishakhachavan/how-to-choose-the-right-machine-learning-algorithm-8fc615de6869#:~:text=How%20to%20Choose%20The%20Right%20Machine%20Learning%20Algorithm,Parameters%20...%208%20Linear%20or%20not%20...%20%E6%9B%B4%E5%A4%9A%E9%A1%B9%E7%9B%AE>.
45. A. C. Müller, S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly Medi, California, 2016.
46. M. Sokolova, N. Japkowicz, S. Szpakowicz, Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation, in *Australasian Joint Conference on Artificial Intelligence*, **4304** (2006), 1015–1021. [https://doi.org/10.1007/11941439\\_114](https://doi.org/10.1007/11941439_114)
47. D. M. W. Powers, Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation, *J. Mach. Learn. Technol.*, **2** (2011), 37–63.



48. B. W. Matthews, Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochim. Biophys. Acta*, **405** (1975), 442–451. [https://doi.org/10.1016/0005-2795\(75\)90109-9](https://doi.org/10.1016/0005-2795(75)90109-9)
49. C. C. Aggarwal, *Neural Networks and Deep Learning*, Springer, Cham, 2018. <https://doi.org/10.1007/978-3-319-94463-0>
50. Machine Learning at Waikato University, Weka 3-Data Mining with Open Source Machine Learning Software in Java, 2023. Available from: <https://www.cs.waikato.ac.nz/ml/weka/>.
51. C. Elkan, The foundations of cost-sensitive learning, in *Proceedings of the 17th international joint conference on Artificial intelligence*, **2** (2001), 973–978.
52. G. Zazzaro, P. Mercogliano, F. M. Pisano, Data mining to classify fog events by applying cost-sensitive classifier, in *Proceedings of the 2010 International Conference on Complex, Intelligent and Software Intensive Systems*, (2010), 568–573.
53. H. Jia, S. Zhang, C. Wang, X. Wang, Z. Ma, Y. Tan, MSC-1DCNN based homogeneous slope stability state prediction method integrated with empirical information, **118** (2023), 729–753. <https://doi.org/10.1007/s11069-023-06026-6>
54. P. Jing, X. Zhang, W. Gong, L. Ma, Y. Xu, H. Yang, Study on the initiation mechanism and motion characteristics of the Daguangbao landslide and the slope stability evaluation method, *Res. Square*, (2023). <https://doi.org/10.21203/rs.3.rs-2704990/v1>
55. L. Zhu, X. Pei, S. Cui, S. Wang, X. Zhang, Y. Liang, On the initiation mechanism of the Daguangbao landslide triggered by the 2008 Wenchuan (Ms 7.9) earthquake, *Soil Dyn. Earthquake Eng.*, **137** (2020), 106272. <https://doi.org/10.1016/j.soildyn.2020.106272>
56. Y. Song, D. Huang, D. Cen, Numerical modelling of the 2008 Wenchuan earthquake-triggered Daguangbao landslide using a velocity and displacement dependent friction law, *Eng. Geol.*, **215** (2016), 50–68. <https://doi.org/10.1016/j.enggeo.2016.11.003>
57. S. Cui, Q. Yang, X. Pei, R. Huang, B. Guo, W. Zhang, Geological and morphological study of the Daguangbao landslide triggered by the Ms. 8.0 Wenchuan earthquake, China, *Geomorphology*, **370** (2020), 107394. <https://doi.org/10.1016/j.geomorph.2020.107394>
58. X. Li, X. Tang, S. Zhao, Q. Yan, Y. Wu, MPM evaluation of the dynamic runout process of the giant Daguangbao landslide, *Landslides*, **18** (2021), 1509–1518.
59. G. Wang, Unascertained information and its mathematical treatment, *J. Harbin Univ. Civ. Eng. Archit.*, (1990), 1–9.
60. K. Liu, H. Wu, N. Wang, H. Li, S. Liu, *Unascertained Mathematics*, Huazhong University of Science and Technology Press, Wuhan, 1997.
61. K. Liu, *Mathematical Processing and Application of Uncertainty Information*, Science Press, Beijing, 1999.
62. D. Norris, B. W. Pilsworth, J. F. Baldwin, Medical diagnosis from patient records-A method using fuzzy discrimination and connectivity analyses, *Fuzzy Sets Syst.*, **23** (1987), 73–87.



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)