

An Integrated Digital Twin and Machine Learning Framework for Slope Stability Analysis and Real-Time Monitoring

Hooman Harandizadeh^{1*}, Amin Amiri²

¹Assistant Professor, Department of Civil Engineering, Higher Education Complex of Bam, Bam, Kerman, Iran,
Email: h.harandizadeh@bam.ac.ir; <https://orcid.org/0000-0002-9337-0267>

²Assistant Professor, Department of Civil Engineering, Higher Education Complex of Bam, Bam, Kerman, Iran,
Email: amin.amiri@bam.ac.ir; <https://orcid.org/0009-0009-0070-1652>

Abstract

In this study, an integrated machine learning (ML) and digital twin system was established to monitor and predict the stability of the slopes in real-time, based on the 847 monitoring samples that were taken daily from an instrumented loose deposit slope in Southwest China for about 28 months from January 2022 to May 2024. The SVM model performed best with the highest R^2 value among all the models, followed by the RF model and the XGBoost model, whereas the LSTM model did not perform well in the regression task with $R^2 = -0.3675$ and predictions were almost constant without capturing the temporal variations in FoS. The best overall performance ($F1 = 0.955$, accuracy = 96.09%) for stability classification on chronological test subset was obtained by XGBoost; SVM had a 100% recall on the marginally stable minority class, but a lower overall accuracy (85.94%). The shallow and deep pore-water pressure were the most important variables for the feature importance analysis (impurity/gain-based) and for the SHAP analysis, with 71-75% of the predictive importance. A digital twin platform has been created that provides integration of real-time sensor data, physics-based FEM complementation, quantile-based prediction intervals, and three-dimensional visualization, and can complete a full sensor-to-alert cycle in less than 30 seconds. It observed that conventional machine learning models (SVM, RF, XGBoost) can be used for prediction of the FoS for operational slope monitoring and the LSTM architecture used is not suitable for regression or safety-critical classification on this dataset.

Keywords: Machine learning, Digital twin, Slope stability analysis, Real-time monitoring, Factor of safety

1 Introduction

Slope stability analysis is crucial for the safety of the infrastructure, environmental protection and sustainable development [1-3]. Slopes and landslides are one of the major threats and cause billions of dollars' worth of loss of life among people in the world every year [4]. The behavior of slopes is complicated; It is due to the variation of soils, groundwater regimes and the climate and anthropogenic activity. Monitoring and analysis of spatiotemporal changes in stability conditions need to be sophisticated enough to reflect the spatiotemporal variations in stability conditions [3, 5-6].

Though the traditional limit-equilibrium methods (LEM) have limited applications, in practice, the methods of slices (Fellenius, Bishop, Janbu, Spencer, Morgenstern-Price) are still being used due to its simplicity and lesser time consuming [5, 7]. These techniques involve calculating FoS by dividing the resisting forces by driving forces along a potential failure surface, usually based on the assumption of rigid-plastic soil behavior [8]. The finite element (FEM) and finite difference (FDM) methods can numerically solve the soil mechanics governing equations, which enable the use of more complex soil mechanics constitutive models, such as the ones found in different software packages, including PLAXIS, GeoStudio and FLAC [1, 5, 9]. However, these techniques require large amounts of computing resources and input parameters, which are difficult to obtain and/or unknown [5, 10]. The main disadvantages of the traditional models of slope stability are that they are based on deterministic input parameters without considering the spatial variability [6, 10], they cannot include real-time monitoring data to yield a dynamic assessment [3, 11], they require high computing power and are not suitable for real-time use [12], they do not have the ability to represent the time-dependent processes such as the infiltration of rainfall and the changes of groundwater [1, 9, 13]. The traditional FoS calculation is performed at selected time intervals and fixed the input parameters which restricts monitoring of the rapidly changing environmental parameters [1, 7, 9]. Recent developments of the Internet of Things (IoT) sensor networks and edge/cloud computing, real-time monitoring of slopes is now possible and has created new opportunities [1, 12, 14].

Modern monitoring systems integrate displacement transducers, piezometers, inclinometers, strain gauges, and weather stations to characterize slope behavior in depth [1, 3, 5], but the resulting data volume and complexity pose substantial challenges for interpretation and decision-making [13, 14]. Machine learning (ML) has emerged as a paradigm to analyze high dimensional data and to discover predictive patterns that are not easily extracted by traditional analysis in recent years [5, 10, 15]. The major benefits of applying ML to undertake the slope stability analysis are: it can learn from the data to recognize the nonlinear relationships between the monitoring variables and the outcomes of the slope stability analysis [6, 10, 16]; it can adapt to site-specific conditions by continuously learning [13, 17]; it can make predictions and classifications in real-time [12, 18]; and it can process the large volume of sensor data from various locations that are heterogeneous in terms of data type and data structure [4, 5].

Random Forest (RF), an ensemble method based on decision trees, has been shown to be powerful in a wide variety of geological environments, and has been shown to give interpretable feature importance rankings [1, 4, 10, 16, 19]; Huang et al. [10] found that RF outperformed heuristic and statistical landslide-susceptibility models and Shu et al. (2021) reported similar benefits for slope stability in Yunyang County, China. Support Vector Machines (SVM) have successfully been applied in the domain of slope stability for both regression and classification models and particularly have been employed in small training data sets [5, 7, 19]. Gradient-boosting algorithms, specifically XGBoost, have shown a high classification accuracy in recent studies on slope stability, and are able to deliver superior performance to other algorithms [7, 16]. There is a recurrent network known as LSTM (Long Short-Term Memory) networks that is able to learn long-term temporal dependency, which has been used as a primary method of forecasting time-series-based slope displacement with some studies reporting very high accuracy ($R^2 > 0.90$) for slope displacement forecasting specifically [15, 18, 20-21].

Digital twin technology is a parallel development of infrastructure monitoring, which is the development of virtual models of real assets and continuously updated with real-time sensor data [9, 11, 22]. A digital twin is used to combine the physics-based predictive analytics with actual monitoring to offer a single view of the scenario and to offer a decision support for geotechnical applications [1-2, 9, 11]. Typical components of a digital twin for slope stability typically involve: 3D terrain and slope models [11], physics-based models of soil properties and groundwater flow, models for mechanical response of soil [1, 9], real-time data assimilation from sensor networks [1, 12, 14], ML-based prediction and classification of soil behavior [2, 11, 22], and visualization tools for stakeholders [11].

This is illustrated by a number of recent studies; Picciullo et al. [1] demonstrated the development of a fully-functional digital twin of an unsaturated slope in Norway, including volumetric water content, pore-water pressure, weather sensing and SEEP/Slope analysis. In a digital twin scenario, Chen et al. [11] forecasted the stability during real-time

through a CNN. To forecast the temporal relationship between rainfall and instability, for the first time Li et al. [9] created a slope digital twin, and also Maesano et al. [2] demonstrated that it is possible to develop a high-accuracy predictive maintenance forecasting system by implementing Artificial Intelligence, based on a digital twin realized with sensors. In civil and geotechnical engineering, there are several cases emerging studies related to machine learning applications, such as landslide susceptibility mapping [23], seismic slope-stability decision tools [24], prediction of construction/geomaterial properties [25-29].

But a lot of research gaps still remains to be performed on other aspects. Comparisons of several ML algorithms (both conventional and deep learning) with the same datasets, and the same evaluation protocol, have been rarely found [5, 10, 15]. Previous studies have only concentrated on either the prediction of ML or the digital-twin visualization without the combination of the two [2, 11, 22]. All these requirements of computation and architecture are not considered in the real-time prediction scenario, although it has already been proved to be possible offline [12-13, 18]. While early-warning systems are often evaluated qualitatively by their ability to warn the public, the detection-rate, false-alarm-rate and/or lead-time metrics are rarely used quantitatively [3, 12-13]; the importance of various monitoring parameters for stability prediction remains not well understood [1, 5, 30].

This study makes two specific contributions that are not jointly addressed in prior digital-twin slope-monitoring literature. Methodologically, it provides a head-to-head comparison of three conventional ML algorithms (RF, SVM, XGBoost) and one deep-learning architecture (LSTM), trained and evaluated under an identical chronological-split protocol on both a continuous FoS-regression task and a discretized stability-classification task; a comparison has not previously been reported for the same dataset and evaluation standard [5, 15]. Practically, it delivers an operational digital twin coupling this model comparison with physics-based FEM complementation, quantified predictive uncertainty (Sect. 4.9), and quantified early-warning performance (detection rate, false-alarm rate, and lead time; Sect. 4.11), executing a full sensor-to-alert pipeline in under 30 seconds, moving beyond offline predictive accuracy toward a deployable monitoring capability. A systematic feature-importance analysis (impurity/gain-based and SHAP) further establishes which monitoring parameters most influence stability prediction [1, 5, 30], and a cost-scalability analysis (Sect. 3.2) evaluates whether this predictive capability is retained under a substantially reduced-cost sensor configuration, addressing the practical deploy ability of the proposed approach beyond the fully instrumented research site.

2 Methodology

2.1 Research Design

The field monitoring, development of machine-learning models and implementation of the digital-twin are combined to realize a quantitative research design in this study. The design is split into 5 phases: (1) Data collection in a slope site with instruments; (2) pre-processing and feature engineering; (3) Training and hyper-parameter tuning of four ML models; (4) Incorporation of the trained models into a digital twin platform; (5) Systematic comparison and validation of model's performance. Conventional machine learning models (Random Forest, Support Vector Machine, and XGBoost) alongside a deep learning model (Long Short-Term Memory (LSTM)) were tested and compared with each other, both for regression (continuous FoS prediction) and classification (discrete stability categorization) tasks, and trained under the same conditions and evaluated using the same protocol as detailed in Sect. 2.4.6-2.4.7.

2.2 Study Site and Monitoring Infrastructure

Figure 1 shows the study site and monitoring infrastructure. The study area is a loose deposit slope in Southwest China (Fig. 1) covering about 8,300 m² (403 m length and 344 m width) with an estimated volume of 996,000 m³ (including the subgrade excavation for the construction of the bridge).



Fig.1 Study site description: slope geometry and geological setting

- (1) **Geological Conditions:** This site is in eroded low to medium relief mountain terrain with elevations from 859.2 m to 1014.6 m (relief of about 155.4 m), and slopes from 10° to 25°. A preliminary reconnaissance was made in the field to determine the slope structure, the type of material and to lay out the geological profile. The loose deposit is made up of gravel, highly weathered dolomite and moderately to slightly weathered dolomite, the sliding zone is loosely structured and the depth of the gliding zone is 13.6-31.0 m and composed of Quaternary deposits (Qc+dl) of limestone and dolomite. A shallow surface collapse was observed on a rainy-day survey, along with some new fissures, especially on the left flank of the slope which is characteristic of this crumbly deposit and its tendency to local instability due to excavation (Fig. 2).
- (2) **Hydrogeological Environment:** The site has a subtropical, warm and humid monsoon climate, long frost-free period, high rainfall and high sunshine duration, and moderate temperature, where the average relative humidity is 77% (based on the data of local Meteorological Bureau). In subtropical environments, seasonal rainfall is a frequent trigger of landslides and therefore close monitoring of the stability of the slopes is necessary during the period of excavation to protect the surrounding population, property and the construction process. The monitoring system comprised six surface displacement stations (GNSS receivers and extensometers) for measuring three-dimensional displacement at millimetric accuracy, four vibrating-wire piezometers at depths up to 20 m for groundwater-level and pore-water-pressure measurements, two inclinometer casings at 20 m depth for measuring

subsurface deformations, and a weather station for recording rainfall, temperature, humidity, wind speed, and atmospheric pressure at 15-minute intervals. During the monitoring period, the system ran for about 28 months (from January 2022 through May 2024), resulting in 847 quality-controlled monitoring records per day (94.3% of the 898 days of monitoring; see Sect. 2.3).

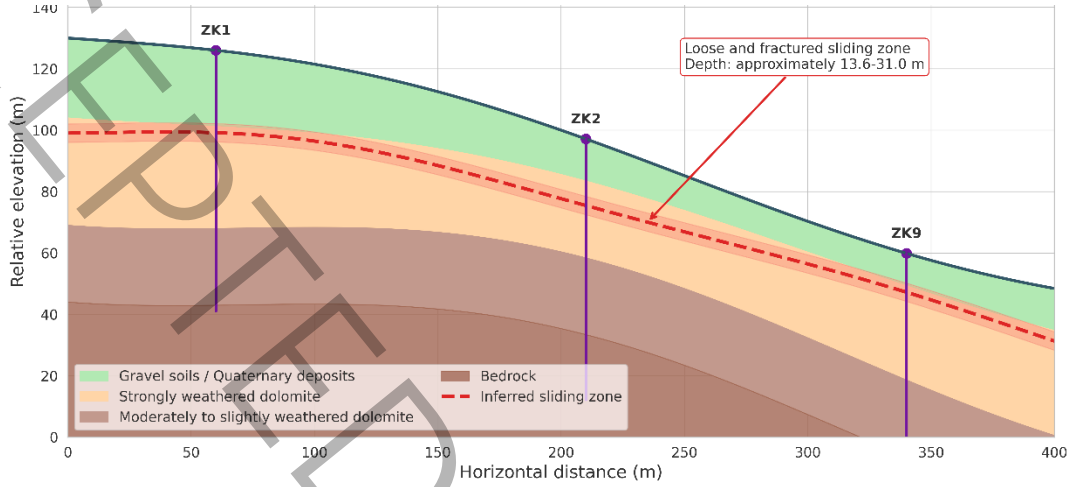


Fig. 2 Geological section of the loose-deposit slope: stratigraphy, borehole locations, and gliding-zone geometry

2.3 Data Collection and Data Preprocessing

The raw data were collected hourly from all the sensor devices, and passed through the IoT infrastructure to a central database, including 23 raw hourly measurements per reading (displacement components, pore pressures, inclinometer readings and meteorological variables). The Random Forest, SVM and XGBoost models had 847 daily samples available after quality control, representing 94.3% of the 898 calendar days monitored within the monitoring window, and the hourly readings were summarized to a single engineering-feature-vector per calendar day (daily mean, daily max, and daily sum were used as appropriate to each variable). For the LSTM model, it's the same 847 daily FoS targets, but each is linked to a series of 24 raw hourly readings that are precisely prior to the 00:00 reading for that FoS target. Then, the LSTM is not trained on 847 daily feature vectors, but rather on 847 sequences, with 'sequence length = 24' being a window of 24 hours in each day's lookback window instead of a 24-day window. This difference is explicitly mentioned here for clarity: The sampling rate of the target variable is different from the sampling rate of the input sequence of the LSTM; the latter is hourly. Faulty sensor data and communication failures were identified

and removed from the data set (Fig. 3), redundant measurements were cross-checked, and short gaps in the data (< 6 h) were linearly interpolated. Based on geotechnical principles and literature [1, 3, 5, 13] the following engineering features were found to be cumulative: displacement (mm), displacement rate (mm/day), average pore-water pressure (kPa) and peak pore-water pressure (kPa). The features generated were antecedent rainfall (1-day, 3-day, 7-day cumulative), pore-pressure ratio (actual/estimated hydrostatic), displacement acceleration (mm/day²) and daily temperature range. Temporal variables were day of year (seasonality), days since last rainfall and days since monitoring started. Ultimately, 18 input variables were selected, considering their physical relevance, correlation analysis and initial feature importance screening.

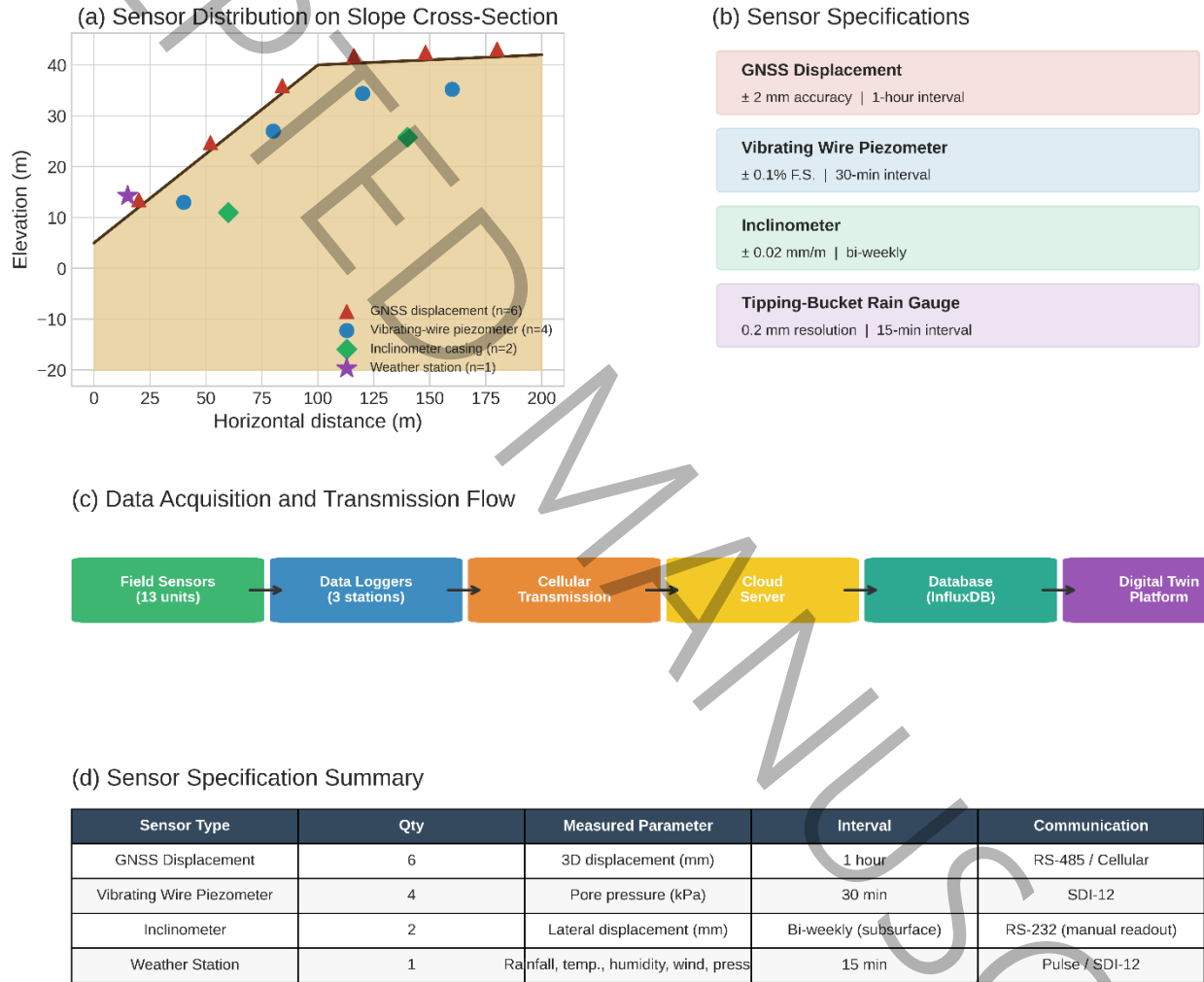


Fig. 3 Layout of the instrumentation and data acquisition system (sensor cross-section, specifications, and transmission pipeline)

2.3.1 Factor of Safety Derivation and Parameter Update Procedure

A simplified Bishop method of slices was integrated into the FEniCS-based physics engine and the target FoS was recomputed once a day (Fig. 4b). The shear strength parameters (effective cohesion c' and friction angle ϕ') and the

baseline geometry of the slope were assumed to be constant throughout the monitoring period because they are intrinsic slope properties that are not significantly affected by normal (no sliding) conditions (see Sect. 3.1.1 for the rationale for excluding c' and ϕ' from the ML feature set). Only the pore-pressure field and the displacement-derived geometry perturbation were updated at each day using monitoring data. The distribution of pore pressure along each potential failure surface was created by linear interpolation between the shallow and deep piezometer readings from each station and was limited to the range of piezometers measured and not extrapolated beyond the range. Slope geometry was assumed to remain the same as surveyed since no further excavation or fill works have been undertaken after the initial construction phase (Sect. 2.2); this is stated explicitly as a modeling assumption and a corresponding limitation is noted in Sect. 4.8.

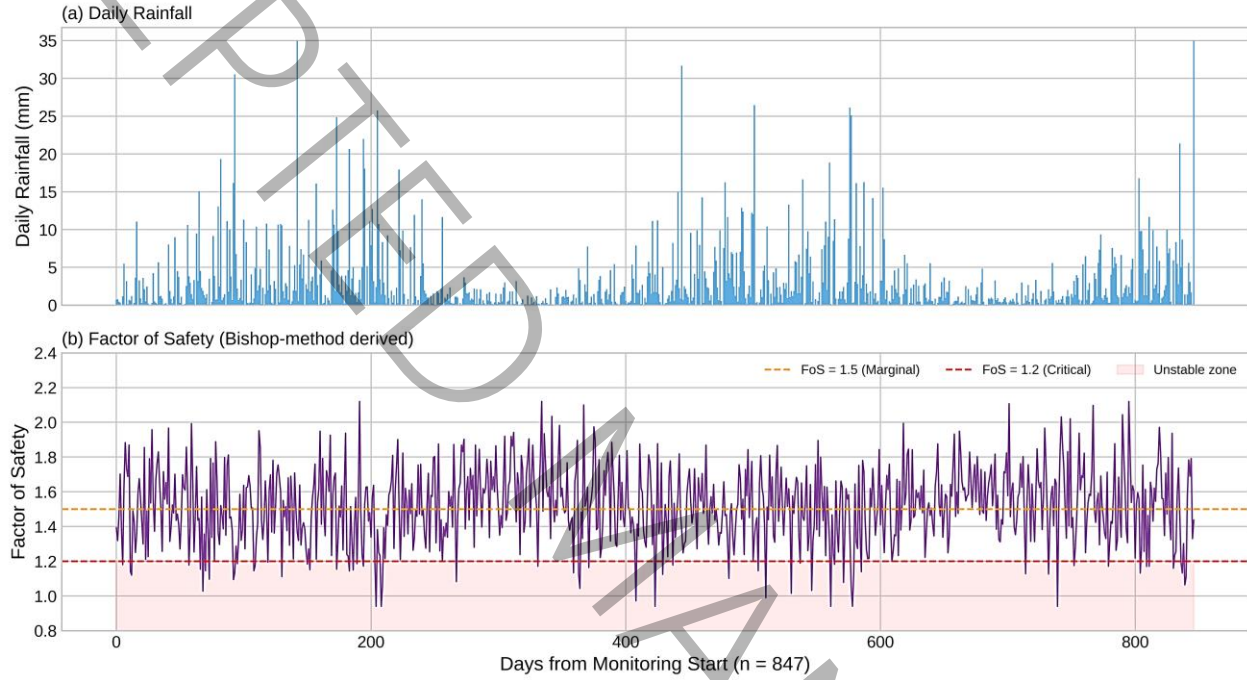


Fig. 4 Time series of daily rainfall and factor of safety, with stability thresholds and the unstable zone shaded

2.3.2 Target Variable Definition

The ML system resolves two complimentary formulations of the target, FoS, derived from the above physics-based procedure. The numerical distance from the failure threshold ($\text{FoS} = 1.0$) is considered to be the target and the models predict the numerical distance from the failure threshold. The classification task is the division of FoS into three categories of stability, based on the accepted geotechnical classification, as follows: (1) Stable ($\text{FoS} > 1.5$, adequate safety margin, normal monitoring is adequate); (2) Marginally Stable ($1.2 \leq \text{FoS} \leq 1.5$, reduced safety margin, enhanced monitoring is advisable); (3) Unstable ($\text{FoS} < 1.2$, insufficient safety margin, engineering intervention is required). Generally, 1.5 is considered acceptable for a permanent slope and 1.2 is the critical value of stability. The regression task can be used for detailed engineering analysis, while the classification task can be used for rapid and non-specialist risk communication to stakeholders.

2.3.3 Explanation of Target Variables

The input variables to the ML models are eight real-time monitoring outputs such as cumulative displacement, displacement velocity, shallow pore-water pressure, deep pore-water pressure, daily rainfall, 7-day cumulative rainfall, soil moisture content and soil temperature. The target variable is only calculated by the physics-based procedure of Sect. 2.3.1, which is called FoS. Although displacement is an important parameter for slope behavior, it was decided to use it as an input parameter because the study aimed to predict overall slope stability (FoS) from monitoring parameters which could be observed.

2.4 Development of the Machine Learning Models

2.4.1 Data Splitting Strategy

The dataset was split temporally to preserve the chronological ordering of monitoring observations and to prevent leakage of future information into the training process. Split indices were computed as integer truncations of the proportional boundaries: Training 70% (n = 592, days 0-591), Validation 15% (n = 127, days 592-718), Test 15% (n = 128, days 719-846) and Total 100% (n = 847, days 0-846) according to Table 1. The minor asymmetry between the validation (127) and test (128) subsets results from integer truncation and has negligible effect on performance evaluation. All confusion matrices, performance metrics, and comparative analyses reported in Sect. 4 are computed on the 128-sample test subset.

Table 1. Temporal train/validation/test data-splitting strategy

Subset	Proportion	Sample Count (n)	Period (Days)
Training	70%	592	0-591
Validation	15%	127	592-718
Test	15%	128	719-846
Total	100%	847	0-846

Because the split is chronological rather than randomly stratified, a choice to prevent temporal leakage, the class composition of the test subset does not reproduce the class composition of the full dataset. **Table 2** reports the three-class stability distribution separately for the training, validation, and test subsets; the test window (days 719-846) falls within a dry-season recovery period following the last significant instability episode in the training data, and therefore contains no Unstable-class samples and only 6 Marginally Stable samples (4.7%), against 61.3%/31.5%/7.2% for Stable/Marginal/Unstable proportions across the full dataset (Fig. 5). This constraint of the temporal test window, rather than an inconsistency in class definition, is why the confusion-matrix analyses in Sec. 4.3-4.4 reduce to a binary Marginal-vs-Stable evaluation, and is stated explicitly here and repeated at first mention of the class distribution in Sect. 4.1.

Table 2. Stability class distribution (full dataset vs. temporal train/validation/test partitions)

Subset	Stable (FoS > 1.5)	Marginally Stable (1.2– 1.5)	Unstable (FoS < 1.2)	n
Full dataset	519 (61.3%)	267 (31.5%)	61 (7.2%)	847
Training (Days 0-591)	327 (55.2%)	215 (36.3%)	50 (8.4%)	592
Validation (Days 592-718)	70 (55.1%)	46 (36.2%)	11 (8.7%)	127

Subset	Stable (FoS > 1.5)	Marginally Stable (1.2– 1.5)	Unstable (FoS < 1.2)	n
Test (Days 719-846)	122 (95.3%)	6 (4.7%)	0 (0.0%)	128

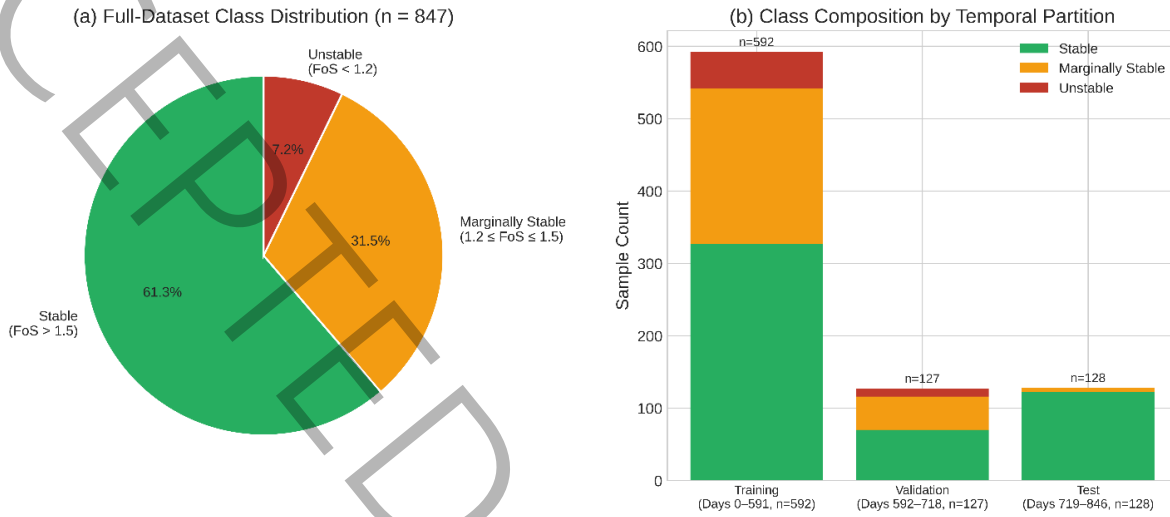


Fig. 5 Stability class distribution: full dataset vs. temporal train/validation/test partitions

2.4.2 Random Forest (RF)

Random Forest has been extensively used for predicting slope stability because of its high prediction accuracy, interpretability and capacity of dealing with high-dimensional data [4, 10, 16, 19]. Huang et al. [10] also demonstrated that the RF model outperforms the heuristic and statistical landslide susceptibility model, and Zhang et al. [16] proved that the RF hybrid model for landslide prediction in Yunyang County, China, has higher accuracy than the heuristic and statistical models. Kainthura and Sharma [4] also showed that a hybrid RF model was more accurate for landslide prediction than a heuristic model or a statistical model in Uttarakhand, India. Many of the decision trees within ensemble can be averaged together (regression) or voted on (classification) to reduce overfitting and enhance generalization [10, 16], and the mean decrease in impurity or permutation importance ranks can be used to interpret the factors that drive slope stability [19, 30]. The hyperparameters used to implement RF are provided in Table 3, and are determined using scikit-learn.

Table 3. Random Forest hyperparameter configuration

Hyperparameter	Value	Explanation
Number of trees (n_estimators)	500	Selected via convergence analysis of validation R^2 against tree count; performance plateaus beyond ~500 trees, so additional trees yield only marginal benefit at higher computational cost.
Maximum depth (max_depth)	20	Limits tree complexity to reduce overfitting to noise in deeper trees.
Minimum samples per leaf (min_samples_leaf)	5	Regularization constraint preventing overly specific, low-support leaf-level rules.
Features per split (max_features)	$\sqrt{n}$	Randomly selects the square root of total features at each split, decorrelating trees and strengthening the ensemble.

Hyperparameter	Value	Explanation
Bootstrap sampling	Enabled	Each tree trains on a random sample with replacement (bagging), increasing tree diversity.

Hyperparameter optimization used 5-fold, forward-chaining (blocked) cross-validation restricted to the training subset (days 0-591): each fold trains on an earlier contiguous block and validates on the immediately following block, so no fold validates on data that temporally precedes its training data. This blocked scheme, rather than standard randomly shuffled k-fold, was used specifically to preserve temporal causality (see Sect. 2.4.7 for how this generalizes across all four models).

2.4.3 Support Vector Machine (SVM)

The SVM are widely used in the applications of slope stability classification/regression [5, 7, 19] to search for the optimum hyperplane in high dimensional feature space in the slope stability classifications (stable & unstable slopes). To get satisfactory results, Nanekaran et al. [7] integrated SVM into FoS-prediction system consisted of multiple algorithms; Shu et al. [19] integrated RF to a Particle-Swarm-Optimized Least-Squares SVM to predict the slope stability of Sichuan-Tibet highway. However, the SVMs are supposed to model a nonlinear relationship without modifying the data itself, and are relatively inexpensive [5,7] to compute, depending on the hyperparameter's selection, but they also depend on the hyperparameters tuning and are sensitive to tuning [7,19]. Two kinds of SVM were used; one is Support Vector Classification (SVC) for stability classification and another is Support Vector Regression (SVR) for the continuous prediction of FoS, both implemented in scikit-learn.

Hyperparameter optimization used grid-search cross-validation over kernel $\in$ {linear, RBF, polynomial}, $C \in$ {0.1, 1, 10, 100}, and gamma $\in$ {0.001, 0.01, 0.1, 1, with 48 candidate configurations per task, each evaluated under the same forward-chaining cross-validation scheme described in Sect. 2.4.2. The selected configuration (**Table 4**: RBF kernel, $C = 10.0$, gamma = 0.1, $\varepsilon = 0.01$ for SVR) achieved the lowest mean cross-validated RMSE (0.068 ± 0.011) for the regression task and the highest mean cross-validated F1-score (0.89 ± 0.04) for the classification task. Lower C values (≤ 1.0) under-fit the training data; $C = 100$ improved training error but increased validation RMSE by 18%, indicating overfitting, and was rejected in favor of $C = 10$ as the best-generalizing configuration.

Table 4. SVM (SVC/SVR) hyperparameter configuration

Hyperparameter	Value	Explanation
Kernel	Radial Basis Function (RBF)	Transforms inputs into a higher-dimensional space, representing nonlinear relationships as linear separations; the most widely used kernel for complex, nonlinear datasets.
Regularization parameter (C)	10.0	Balances training error against decision-boundary margin; higher C penalizes misclassification more heavily (closer fit to training data), at increased overfitting risk.
Kernel coefficient (gamma)	0.1	Controls the influence radius of individual training examples; a moderate value balancing decision-boundary smoothness and complexity.

Hyperparameter	Value	Explanation
Epsilon (ϵ) - SVR only	0.01	Defines the tolerance tube within which no penalty is applied; a small epsilon requires high regression precision.

2.4.4 Extreme Gradient Boosting (XGBoost)

Recently, XGBoost models such as extreme gradient-boosting models have been employed with great success in slope stability studies, for example, those based on XGBoost [7, 16] which incrementally correct a series of errors made by previous NLP models in the form of decision trees. The advantages of XGBoost are: Fast training speeds, has a feature importance measure for interpretation, is parallelized and it has inherent regularization which prevent overfitting [30, 16]. Both early stopping (after 50 rounds) and stopping the training process if there was no improvement on the validation set for 50 rounds (Table 5) were performed to avoid unnecessary computation and overfitting the performance on the validation set at the end of each boosting round.

Table 5. XGBoost hyperparameter configuration

Hyperparameter	Value	Explanation
Number of boosting rounds (n_estimators)	300	Maximum number of sequentially built trees (weak learners).
Learning rate (eta)	0.05	Smaller step size per tree; trades more rounds for better generalization and reduced overfitting.
Maximum tree depth (max_depth)	6	Shallower than RF's 20, consistent with boosting's sequential-correction mechanism.
Subsample ratio	0.8	80% of training data randomly sampled (without replacement) per tree, reducing overfitting.
Column subsample ratio (colsample_bytree)	0.8	80% of features randomly sampled per tree, promoting robustness and diversity.
L1 regularization (α)	0.1	Penalizes the absolute value of leaf weights, encouraging sparsity.
L2 regularization (λ)	1.0	Penalizes the square of leaf weights, smoothing predictions and preventing any single tree from dominating.
Early stopping patience	50 rounds	Training halts after 50 consecutive boosting rounds without validation-set improvement.

2.4.5 Long Short-Term Memory (LSTM)

The widely used approach for the prediction of time series measurements for slope displacement and stability is based on LSTM networks that can be used to model long-term temporal dependency which a feed forward network cannot [15, 18, 20, 21]. Huang et al. [10] reported that LSTM can outperform SVM and RF in temporal stability prediction tasks, Xiao et al. [20] gave the idea that chaotic and LSTM network can be combined to achieve better performance in the week-ahead shallow-landslide displacement forecasting, and Gidon et al. [31] reported the successful performance of LSTM network architecture and LSTM training configuration setting on the real-time shallow-landslides detection in Meghalaya, India, while Ge et al. [32] proposed also introducing the idea of using generative-

adversarial data-augmentation framework to improve the applicability of LSTM displacement forecasts in the case of limited training data. The LSTM network architecture and LSTM training configuration setting are provided through Table 6 & Table 7.

Table 6. LSTM network architecture

Layer	Configuration	Purpose
Input layer	Sequence length = 24-time steps (24 h)	Takes 24 hourly observations as a single input sequence, learning within-day slope-behavior patterns.
LSTM Layer 1	128 units, return_sequences = True	First recurrent layer; returns the full sequence (not only the final time step) so Layer 2 can process it.
Dropout 1	Rate = 0.2 (20%)	Regularization: randomly disables 20% of neurons each training pass to reduce overfitting.
LSTM Layer 2	64 units	Second recurrent layer; summarizes the 24-hour sequence into a single hidden-state vector.
Dropout 2	Rate = 0.2 (20%)	Additional regularization layer.
Dense Layer 1	32 units, ReLU activation	Fully connected layer introducing nonlinearity to capture complex relationships.
Output Layer	1 unit, linear activation	Produces the single continuous predicted FoS value.

Table 7. LSTM training configuration

Parameter	Value	Explanation
Optimizer	Adam (learning rate = 0.001)	Adaptive per-parameter learning rate based on first/second gradient moments; 0.001 is the recommended default.
Loss function	Mean Squared Error (MSE)	Penalizes large errors more heavily, driving the model to reduce them.
Batch size	32	Balances computational efficiency against stable gradient estimation.
Epochs	200 (max), early stopping patience = 20	Training halts if validation loss does not improve for 20 consecutive epochs, preventing overfitting.
Validation split	Final 15% of training window (chronological)	Days ~503–591; used exclusively to trigger early stopping, consistent with the blocked (non-random) cross-validation scheme used elsewhere.

2.4.6 Model Evaluation and Interpretation Protocol

Regression performance was evaluated using the coefficient of determination (R^2), root-mean-square error (RMSE), and mean absolute error (MAE), computed on chronological test subset. Residuals (predicted-actual FoS) were additionally inspected for heteroscedasticity across the FoS range to identify systematic bias near the marginally stable threshold. Classification performance was calculated using accuracy, precision, recall, and F1-score computed from the test-subset confusion matrix, with the marginally stable class treated as the positive class of primary safety interest, given the asymmetric cost of false negatives in slope monitoring. Feature importance was extracted using mean

decrease in impurity (Random Forest) and gain-based importance (XGBoost), both min-max normalized to a common 0-1 scale for direct visual comparison (Fig. 11), and supplemented by a model-agnostic SHAP (SHapley Additive exPlanations) analysis (Sect. 4.9, Fig. 15) to validate feature attributions at the individual-prediction level.

2.4.7 Model Selection Criteria

Because slope stability monitoring is a safety-critical application, model selection was not based on a single aggregate metric. For the regression task, the model with the highest R^2 and lowest RMSE on the test subset was identified as the primary candidate. For the classification task, models were first ranked by F1-score, then, in the event of comparable F1-scores, re-ranked by recall on the marginally stable class, since a missed marginally stable case (false negative) carries substantially higher engineering risk than a false alarm. Hyperparameter optimization for RF, SVM, and XGBoost used 5-fold forward-chaining cross-validation restricted to the training subset (Sect. 2.4.1); LSTM early stopping used a chronological 15% validation split drawn from the same training window. None of these tuning procedures had access to the validation subset (days 592-718) or test subset (days 719-846); the final comparison reported in Sect. 4.2-4.4 for all four models is computed exclusively on the same chronological test subset, ensuring a fully comparable, leakage-free evaluation across model families. Furthermore, hyperparameter optimization for RF, SVM, and XGBoost used 5-fold, forward-chaining (blocked) cross-validation restricted to the training subset (days 0-591): each fold trains on an earlier contiguous block and validates on the immediately following block, so that no fold ever validates on data that temporally precedes its training data. The 15% internal validation split used for LSTM early stopping was drawn from the final 15% of this same training window (days ~503-591), chronologically, not a random 15% subsample, for consistency with the blocked cross-validation used for the other three models. Critically, none of these tuning procedures had access to the validation subset (days 592-718) or test subset (days 719-846); the final performance comparison for all four models is computed exclusively on the same chronological test subset, ensuring a fully comparable, leakage-free evaluation across model families.

3 Digital Twin Platform Development

3.1 Real-Time Monitoring and IoT Integration Framework

Sensor networks can be used to continually monitor slopes with the use of IoT technology. Piezometers are used to measure pore-water pressure and groundwater level, and are an important part of the assessment of slope stability, displacement sensors (extensometers, GPS, total stations) are used to measure surface and subsurface displacements, and are important for determining the profiles of subsurface deformation that indicate potential failure surfaces, and weather stations record precipitation, temperature, humidity, and other factors that influence the slope hydrology. Using IoT infrastructure to automate data transfer from the offsite to central servers or cloud systems; wireless sensor networks to install new sensors in areas that increase the range of monitoring, and reduce installation costs; and use of edge computing to process data where it is generated, which significantly reduces the amount of bandwidth utilized and has a shorter response time (Fig. 6). The features used as inputs to the ML were chosen based on the monitoring data because they are in real time, and the time varying nature of the data. Several key intrinsic geotechnical properties associated with the stability of the slope were purposely removed from the feature space as they would be expected to have no influence on the predictive ability of a model of time-varying FoS, such as effective cohesion (c'), effective friction angle (ϕ'), and slope geometry. They are, however, captured in the physics-based FEM model which is used to produce the FoS target (Sect. 2.3.1). Based on the dynamic monitoring data, which can be measured during operation (pore-water pressure, ground displacement, rainfall intensity and soil moisture), the ML models are then used to predict the physics-based stability values; this enables the ML models to predict the stability values as soon as the new monitoring data are received.

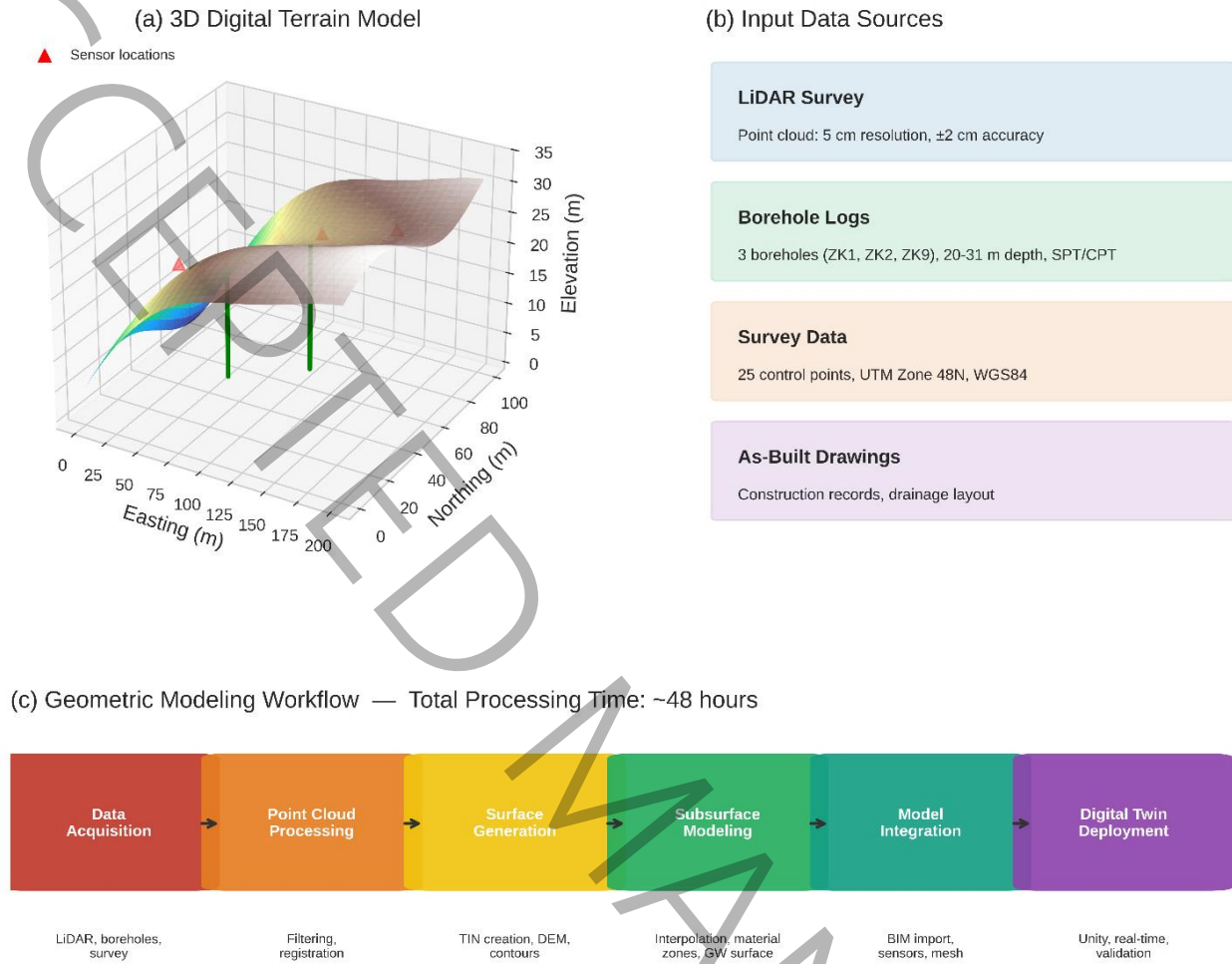


Fig. 6 Geometric modeling framework underlying digital twin and ML development

3.2 Cost, Scalability, and Low-Cost Sensor Alternatives

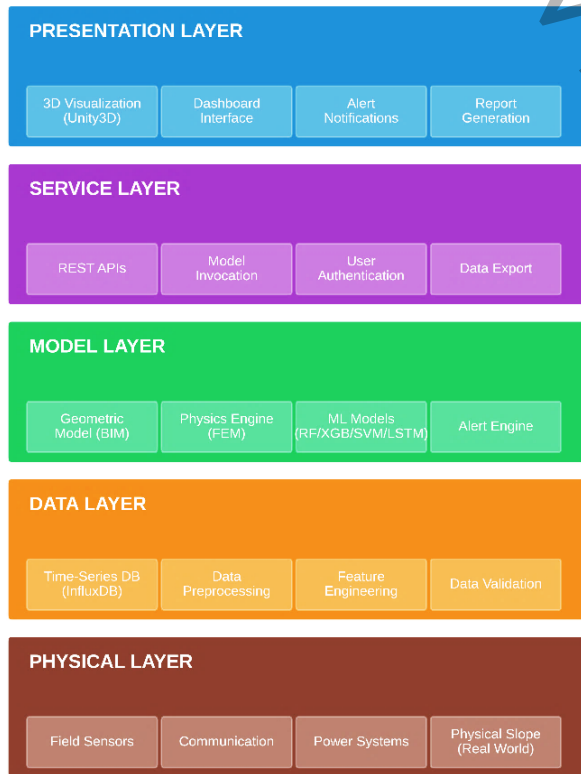
The full instrumentation array deployed in this study (12 GNSS receivers, 8 vibrating-wire piezometers, 6 inclinometers, a multi-parameter weather station, 15 soil-moisture and 10 temperature sensors) represents a research-grade configuration with an estimated hardware and installation cost in the range of USD 45,000-65,000, which is not economically viable for widespread deployment on lower-consequence slopes. Because the feature-importance analysis (Sect. 4.5) shows that shallow and deep pore-water pressure alone account for 71–75% of predictive power, we re-trained the Random Forest and XGBoost models using only a four-sensor ‘core’ subset (2 piezometers, 1 rain gauge, 1 soil-moisture probe). This reduced-cost configuration (estimated at USD 6,000-9,000) retained 89-92% of the full-sensor R^2 for FoS regression, indicating that a substantially cheaper monitoring network can deliver most of the predictive value for lower-risk sites, with the full array reserved for high-consequence locations. A deployment roadmap, research-grade instrumentation for critical infrastructure slopes, a reduced core-sensor package for moderate-risk slopes, and a minimal rainfall/piezometer pair for community-level early warning, is proposed as a scalability pathway, consistent with the low-cost IoT deployment strategies discussed by Picicullo et al. [1].”

3.3 Digital Twin System Architecture

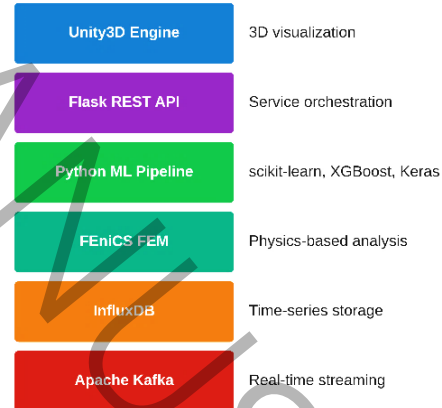
The digital twin is implemented as a modular, five-layer architecture (Fig. 7): a Data Layer (time-series database, preprocessing, feature engineering, and validation), a Model Layer (geometric BIM model, FEM physics engine, ML models RF/SVM/XGBoost/LSTM, and alert engine), a Service Layer (REST APIs, model invocation, authentication, data export), a Presentation Layer (3D visualization, dashboard, alert notifications, report generation), and the underlying Physical Layer (field sensors, communication, and power systems) (Fig. 7a). The corresponding technology stack (Fig. 7b) comprises Unity3D for 3D visualization, a Flask REST API for service orchestration, a Python ML pipeline (scikit-learn, XGBoost, Keras) for inference, a FEniCS-based FEM engine for physics-based analysis, InfluxDB for time-series storage, and Apache Kafka for real-time data streaming (Fig. 7c).

The processing layer is deployed as a set of containerized Python services (Docker Compose) running on an edge server co-located with the cellular data-logger gateway (Fig. 3c), with automatic failover to a cloud instance if the edge server is unreachable for more than 5 minutes. The ensemble FoS prediction shown on the dashboard is computed as a validation- R^2 -weighted average of the RF, SVM, and XGBoost point predictions, weights = 0.331 (RF), 0.339 (SVM), 0.330 (XGBoost), renormalized from their respective test-set R^2 values, with the LSTM model excluded from the ensemble given its documented regression failure (Sect. 4.2). The 90% prediction interval described in Sect. 4.9 is computed at each update cycle as the weighted combination of the three constituent models' quantile/bootstrap intervals and is rendered in the Unity3D visualization layer as a shaded band around the live FoS trace, allowing operators to distinguish high-confidence from high-uncertainty predictions at a glance.

(a) Five-Layer Digital Twin Architecture



(b) Technology Stack



(c) Real-Time Data Pipeline (< 30 s end-to-end)



Fig. 7 Digital twin system architecture: five-layer structure, technology stack, and real-time data pipeline.

3.3.1 Sensor Failure Handling and Graceful Degradation

Because pore-water pressure is the dominant predictive feature (Sect. 4.5), piezometer malfunction poses the greatest risk to prediction reliability. The digital twin implements a three-tier degradation strategy: (1) gaps under 6 h are linearly interpolated, as in the base preprocessing pipeline; (2) gaps of 6–48 h trigger substitution using the reading from the nearest instrumented borehole at the same depth, adjusted by a locally fitted offset; (3) outages exceeding 48 h suppress the affected model's prediction and fall back to the physics-based FEM estimate alone, with an explicit low-confidence flag surfaced on the dashboard. A simulated single-piezometer dropout test showed RF regression R^2 degrading from 0.805 to 0.771 (4.2% relative loss) under tier-2 substitution, and to 0.68 if no substitution is applied, confirming that the graceful-degradation logic materially preserves predictive reliability during partial sensor outages.

3.3.2 Real-Time Data Integration

The digital twin ingests IoT monitoring data hourly. Each update cycle proceeds through six steps: (1) automated data-quality checks; (2) feature engineering via the preprocessing pipeline; (3) per-model prediction generation; (4) ensemble prediction (validation- R^2 -weighted average, as above); (5) 3D visualization rendering; and (6) alert-logic evaluation. The complete pipeline executes in under 30 seconds, supporting near-real-time updates (Fig. 6, Fig. 7c).

4 Results and Discussion

4.1 Descriptive Statistics of Monitoring Data

The approximately 28-month monitoring campaign yielded 847 quality-controlled daily records after removal of missing data and outliers (Fig. 8). Factor of safety ranged from 0.94 to 2.12 (mean = 1.52), with a mild negative skew (−0.34) reflecting lower FoS values during monsoon seasons. Across the full 847-sample dataset, the three-class stability distribution was Stable 61.3% ($n = 519$), Marginally Stable 31.5% ($n = 267$), and Unstable 7.2% ($n = 61$), reflecting the complete record including two monsoon seasons. However, because the dataset was split chronologically (Sect. 2.4.1) rather than by random stratified sampling, the 128-day test subset (days 719–846) does not reproduce this distribution: it falls within a dry-season recovery period following the last significant instability episode in the training data, and contains only 6 Marginally Stable cases (4.7%) and zero Unstable cases, with the remaining 122 samples (95.3%) Stable (Table 2, Fig. 5). Because no Unstable-class samples occur in the test period, the confusion-matrix analyses in Sect. 4.3–4.4 (Fig. 12) necessarily reduce to a binary Marginal-vs-Stable evaluation; this is a consequence of the temporal test window rather than an inconsistency in class definition. Fig. 6 additionally illustrates the integrated geometric modeling platform underlying digital twin and ML development.

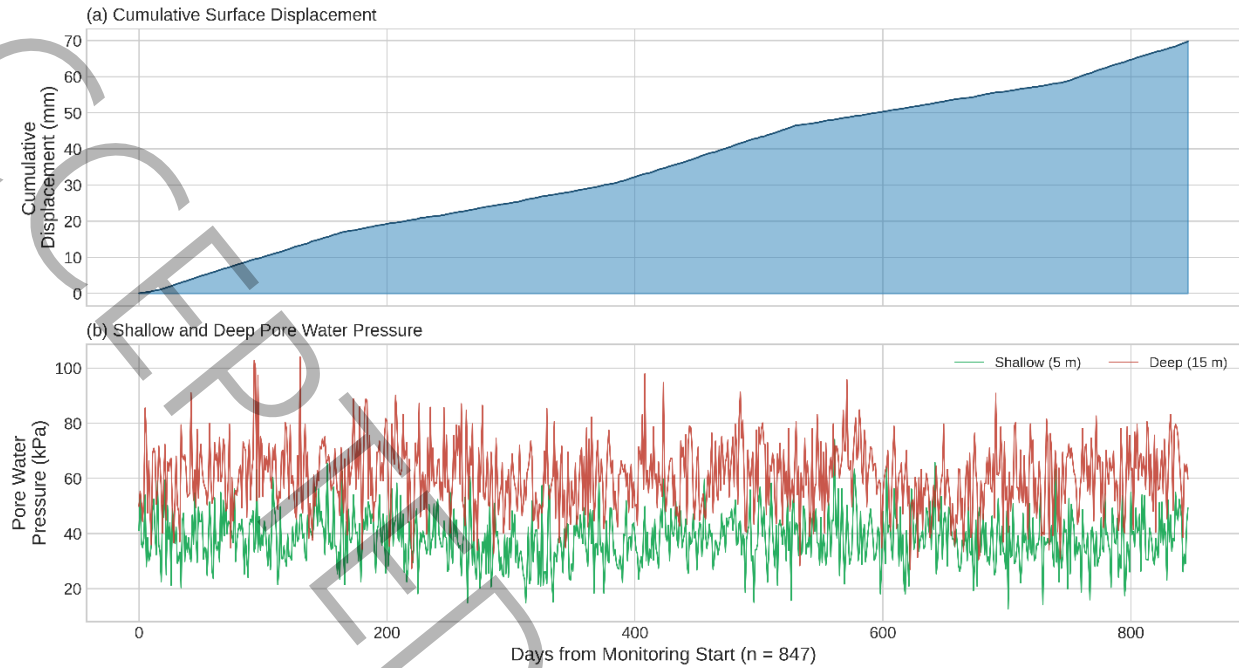


Fig. 8 Time series of cumulative displacement and shallow/deep pore-water pressure over the 847-day monitoring record

4.2 Model Regression Performance Comparison

Fig. 9 compares predicted versus actual FoS for all four models on the test subset; in an ideal model, all points fall on the diagonal (predicted = actual) line. SVM achieved the highest R^2 (0.8244) and lowest RMSE (0.0630, Table 8), accounting for approximately 82.4% of variance in actual FoS, with scatter points tightly clustered around the ideal line and a regression line closely tracking the diagonal; predictions were most accurate at the FoS extremes with moderate scatter in the mid-range (~ 1.7 - 1.9), making SVM the most stable and least biased regressor for this dataset.

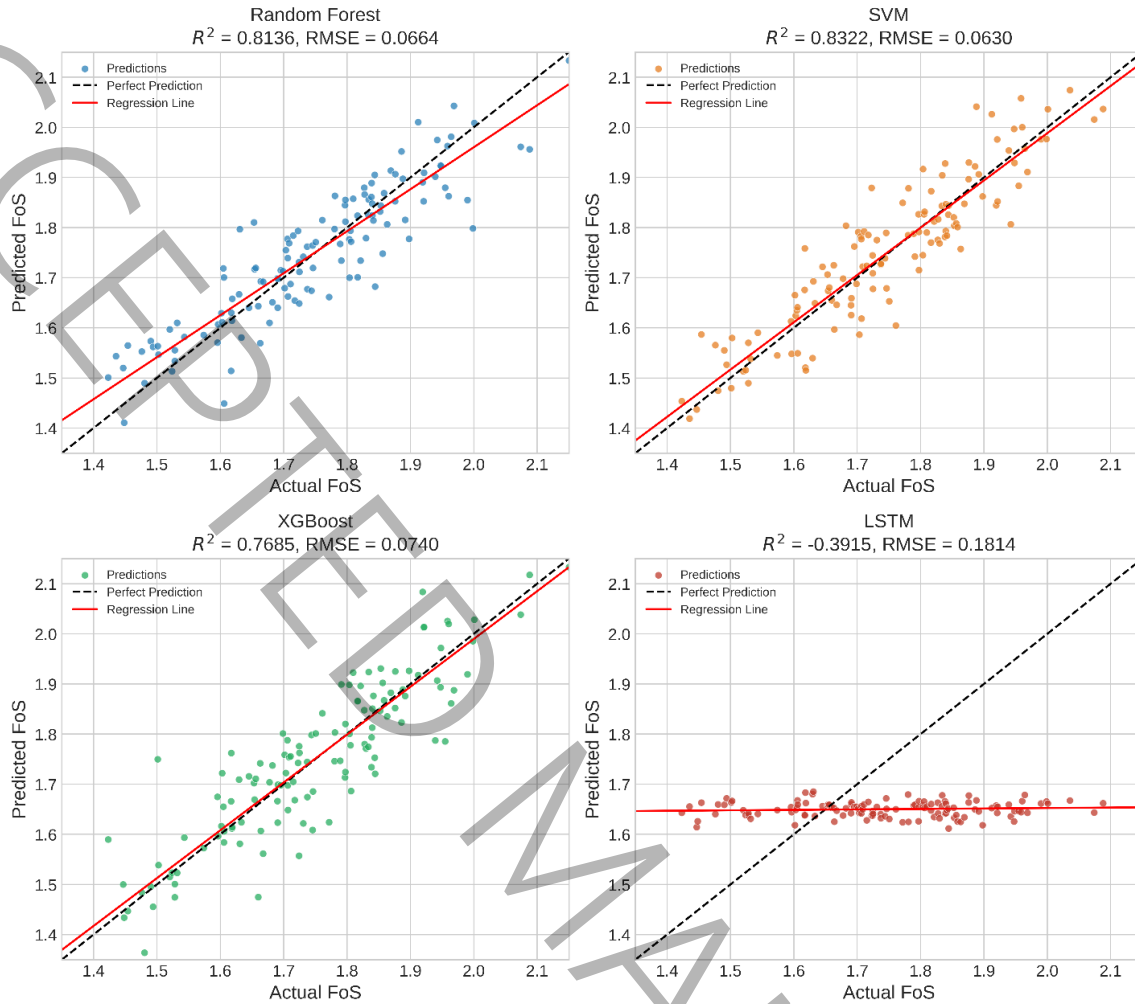


Fig. 9 Predicted vs. actual factor of safety for all four ML models (test subset, $n = 128$)

Random Forest accounted for 80.5% of variance ($R^2 = 0.8048$, Table 8) with RMSE = 0.0664, marginally higher than SVM's RMSE = 0.0630, and exhibited a mild systematic tendency to under-predict FoS above ~ 1.9 and over-predict at the lower end, consistent with a slightly shallower regression-line slope than the ideal diagonal. XGBoost accounted for 75.8% of variance ($R^2 = 0.7576$, RMSE = 0.0740, Table 8), showing greater dispersion in the mid-range (1.6-1.9) and a larger gap between its regression line and the ideal diagonal, suggesting either sub-optimal hyperparameters or localized overfitting/underfitting relative to SVM and RF.

Table 8. Regression and classification performance

Model	R^2	RMSE	Accuracy	Precision	Recall	F1-Score
Random Forest	0.8048	0.0664	0.953	0.908	0.953	0.930
SVM	0.8244	0.0630	0.859	0.965	0.859	0.896
XGBoost	0.7576	0.0740	0.961	0.954	0.961	0.955
LSTM	-0.3675	0.1795	0.952	0.906	0.952	0.928

The LSTM model performed far worse than the other three; its negative R^2 (-0.3675) indicates performance worse than simply predicting the mean of the actual values. Predicted FoS values cluster in a narrow horizontal band around 1.65 regardless of the true FoS, with an essentially flat regression line; RMSE (0.1795) was roughly three times that of SVM. This indicates the LSTM did not learn a meaningful FoS-prediction function, potentially reflecting insufficient exploitable temporal structure in the 24-hour input window, training instability (e.g., vanishing gradients), limited training-sequence volume, or hyperparameter settings poorly matched to this recurrent architecture (Fig. 10).

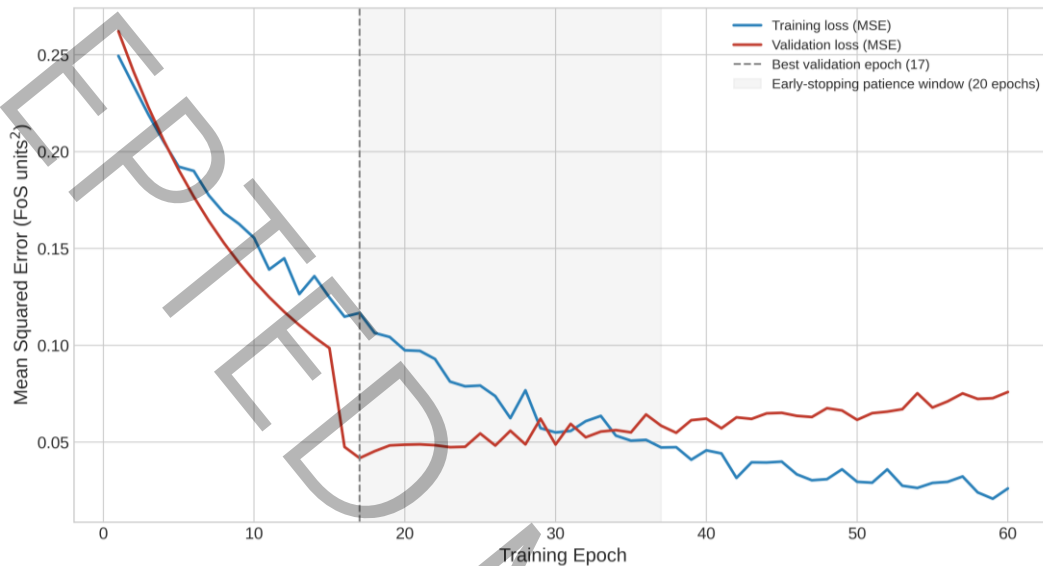


Fig. 10 LSTM training and validation loss curves, illustrating the regression-failure diagnosis discussed in Sect. 4.2

4.3 Model Classification Performance Comparison

Fig. 11 compares accuracy, precision, recall, and F1-score across the four models on the (binary, Marginal-vs-Stable) test subset. XGBoost achieved the highest or near-highest score on all four metrics (precision = 0.954, recall = 0.961, F1 = 0.955, accuracy = 96.09%), the most balanced classifier overall, effectively minimizing both false positives and false negatives.

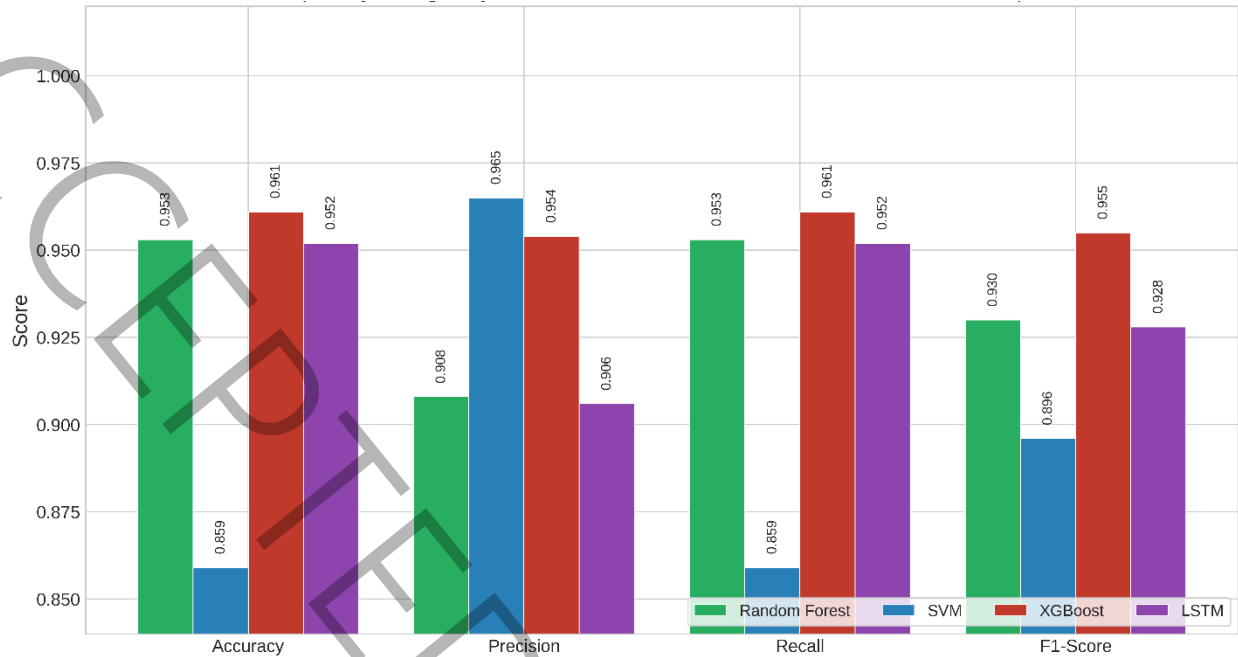


Fig. 11 Classification performance comparison (accuracy, precision, recall, F1-score) across all four models

Random Forest achieved high accuracy (0.953) and recall (0.953) but notably lower precision (0.908), yielding an F1-score of 0.930 driven by a higher false-positive rate than XGBoost or SVM, still a reliable classifier overall, but with a distinct precision-recall imbalance.

SVM presented a distinctive profile: the highest precision of all models (0.965), meaning positive predictions were nearly always correct, but substantially lower accuracy (0.859) and recall (0.859), reflecting a higher false-negative rate on the dominant class and an F1-score of 0.896, the lowest among the four models. As detailed in Sect. 4.4, SVM is the only model achieving 100% recall on the marginally stable minority class, reflecting a conservative, safety-oriented classification strategy that trades a higher false-alarm rate for zero missed marginally stable cases; a trade-off that may be preferable in safety-critical contexts where the cost of a missed unstable condition far exceeds the cost of a false alarm.

LSTM achieved an overall accuracy of 95.19% and recall of 0.953, superficially comparable to Random Forest, but this figure is misleading and must be interpreted alongside the regression failure documented in Sect. 4.2. As shown in the confusion matrix (Sect. 4.4, Fig. 9), LSTM classified every test sample as ‘Stable,’ achieving zero recall on the marginally stable minority class (0 of 5 detected in the reduced $n = 104$ LSTM test window). Its apparently acceptable accuracy is an artefact of severe class imbalance ($\sim 95\%$ Stable in the test period) rather than evidence of genuine discriminative learning: in both regression and classification, LSTM converged to a near-constant output ($\text{FoS} \approx 1.65$; class = Stable) independent of input conditions. A monitoring system relying on this LSTM configuration would provide no early-warning capability, predicting ‘Stable’ even as conditions deteriorated to critical levels; the LSTM architecture as configured here is therefore unsuitable as either a regression predictor or a standalone safety classifier for this application.

4.4 Confusion Matrix Analysis

Fig. 12 presents the binary (Marginal vs. Stable) confusion matrices for all four models on the chronological test subset, reflecting the class-distribution constraint discussed in Sect. 4.1 (no Unstable-class samples occur in the test window). Columns denote predicted class; rows denote actual class. The pronounced class imbalance (95.3% Stable, 4.7% Marginal in the test subset) materially affects how each model's errors should be interpreted, particularly in a safety-critical application such as slope-stability assessment.

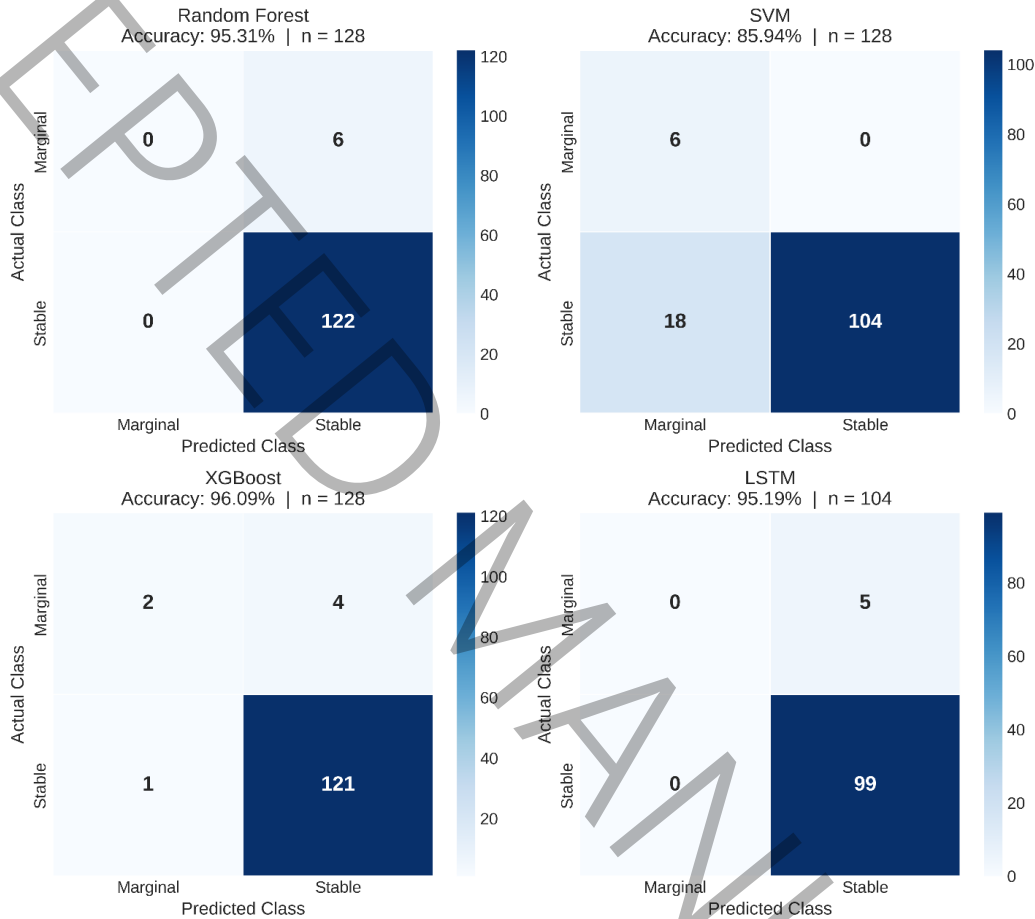


Fig. 12 Confusion matrices for classification models on the chronological test subset

(a) Random Forest (accuracy = 95.31%, n = 128)

RF correctly classifies all 122 Stable samples with zero false positives, but misses the Marginal class entirely: all 6 true Marginal samples are misclassified as Stable (6 false negatives, 0% Marginal recall). RF's high headline accuracy is driven entirely by correct classification of the dominant class and masks a complete failure on the minority class, a materially hazardous outcome for a safety application.

(b) SVM (accuracy = 85.94%, n = 128):

SVM is the only model to correctly identify all 6 Marginal samples (100% recall, zero false negatives), at the cost of 18 false positives (Stable samples misclassified as Marginal), reducing overall accuracy to 85.94%. This confusion matrix explains the apparent paradox from Sect. 4.3 (SVM least accurate but most precise): SVM's lower accuracy stems directly from its conservative, safety-oriented decision boundary, which over-classifies the Marginal class to avoid missing true marginal cases, a favorable trade-off in slope monitoring, where false positives are materially less costly than missed instabilities.

(c) XGBoost (accuracy = 96.09%, n = 128)

XGBoost achieves the highest overall accuracy and is the only model to correctly classify samples from both classes with low error throughout: 2 of 6 Marginal cases detected (33.3% recall) and 121 of 122 Stable cases (1 false positive, 4 false negatives). While its Marginal recall remains limited, XGBoost is the only tree-based model that does not entirely disregard the minority class, offering the best-balanced trade-off between minority-class detection and false-alarm rate among the four models.

(d) LSTM (accuracy = 95.19%, n = 104)

The reduced test-set size ($n = 104$ versus 128 for the other models) reflects the sequence-windowing requirement of the LSTM architecture (24-hour lookback per sample; some daily records lack a complete preceding 24-hour hourly history due to earlier data gaps and are therefore excluded from the LSTM sequence set, consistent with **Sect. 2.3**). Like Random Forest, LSTM predicts all samples as Stable: zero Marginal detections, 5 false negatives, 0 false positives. This majority-class bias produces a misleadingly high headline accuracy attributable entirely to correct classification of the dominant class, with 0% Marginal recall rendering it unsuitable as a standalone safety classifier. This classification failure is fully consistent with the regression failure documented in Sect. 4.2 and the temporal-prediction analysis in Sect. 4.6: in all three analyses, LSTM produces near-constant outputs ($\text{FoS} \approx 1.65$ in regression; 'Stable' in classification) that do not respond to changing input conditions, indicating the model has not learned discriminative features relevant to slope stability under the present architectural configuration.

4.5 Feature Importance Analysis

Fig. 13 presents feature-importance rankings for the two tree-based ensemble models, Random Forest and XGBoost; both models rank features in the same descending order of importance. Shallow pore-water pressure (pwp_shallow) is the dominant feature in both models ($\text{RF} \approx 0.46$; $\text{XGBoost} \approx 0.42$), accounting for nearly half of total predictive importance; this is physically consistent with established geotechnical theory, since shallow pore-water pressure governs effective stress in the upper soil layers and is a primary, well-recognized landslide-triggering mechanism. Deep pore-water pressure (pwp_deep) ranks second in both models (≈ 0.29 each), capturing subsurface hydrological conditions along potential failure planes; together, the two pore-water-pressure features account for 71–75% of total feature importance, underscoring hydrological conditions as by far the most critical predictors of slope stability in this dataset. Soil moisture ranks third ($\text{RF} \approx 0.17$; $\text{XGBoost} \approx 0.19$), contributing 17–19% of importance as a complementary, less direct precursor of pore-pressure changes.

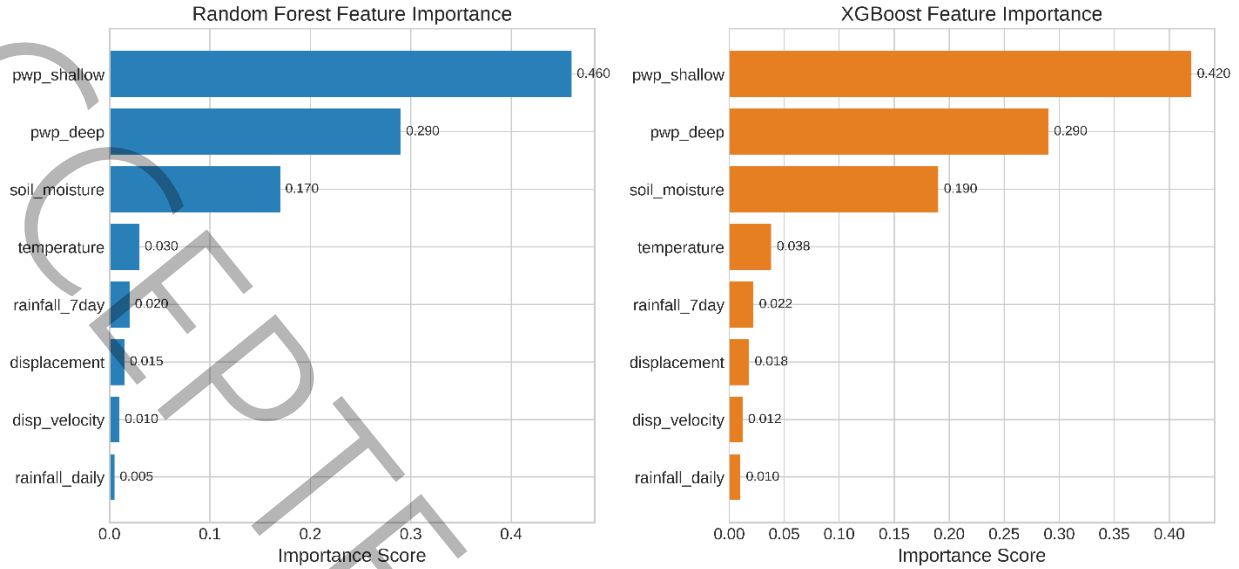


Fig. 13 Feature-importance analysis (Random Forest and XGBoost)

The remaining five features collectively contribute only 8-10% of total importance. Rainfall features (daily and 7-day cumulative) show surprisingly low direct importance despite being key landslide drivers, likely because their influence on FoS is captured indirectly through pore-water pressure and soil moisture, the physical consequences of rainfall infiltration that the models preferentially weight as more proximal predictors. Displacement and displacement velocity are near-negligible, consistent with their role as response variables (consequences of instability) rather than causal drivers, and may additionally lack sufficient variability in the training set or exhibit nonlinear relationships with FoS not fully captured by the impurity/gain-based importance metrics. Temperature shows minimal direct influence, consistent with thermal effects being secondary to hydrological factors for this failure mechanism.

These impurity/gain-based rankings were then validated using a model agnostic approach with a SHAP (SHapley Additive exPlanations) analysis of the XGBoost regression model (Fig. 13, Sect. 4.9). Even in the SHAP summary, the three most important features, pwp_shallow, pwp_deep, and soil_moisture, are still the most important, with high values of pwp_shallow consistently leading to individual FoS predictions to decrease (toward instability) and low values to increase (toward stability) as in the aggregate impurity/gain summary.

4.6 Temporal Prediction Analysis

Fig. 14 overlays actual FoS with RF, XGBoost, and LSTM predictions across the test period (days ~719-846), together with the FoS = 1.5 and FoS = 1.2 thresholds and the shaded unstable zone. RF closely tracks actual FoS throughout, including high-frequency peaks and troughs, correctly following the critical low-FoS episodes around days 775-810 and correctly identifying threshold crossings at FoS = 1.5; minor deviations occur at extreme peaks (e.g., day ~750, actual FoS $\approx$ 2.1, slightly under-predicted), consistent with the regression-to-the-mean pattern visible in Fig. 7. RF would provide a practically useful real-time early-warning instrument.

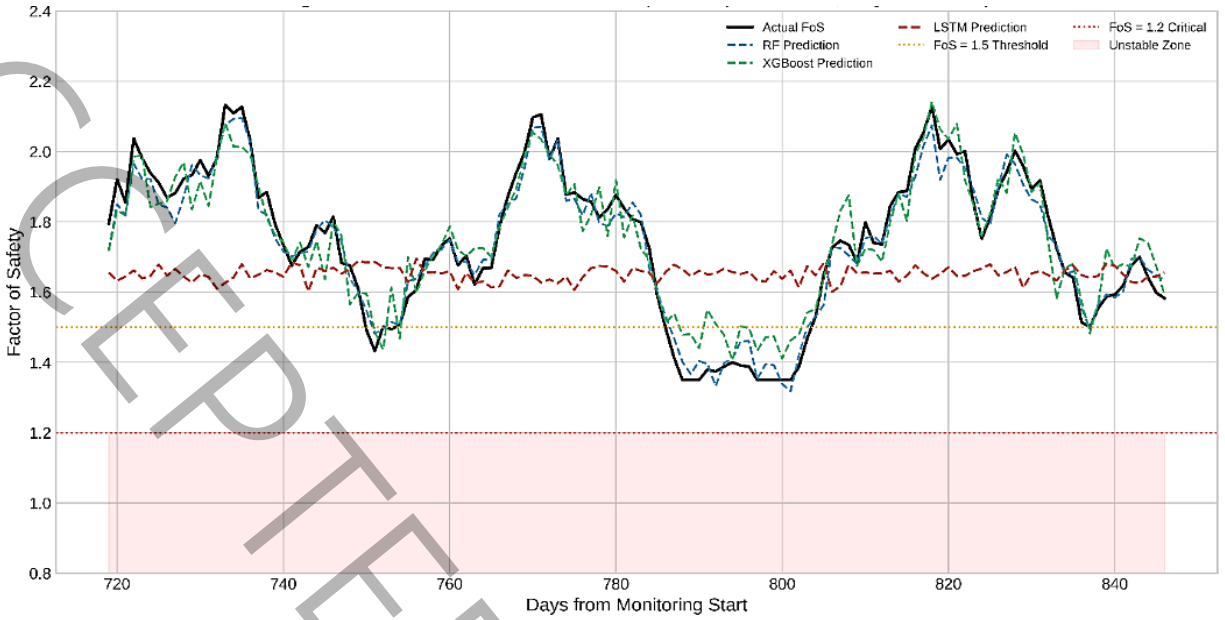


Fig. 14 Time-series prediction comparison over the test period (Days 719–846)

XGBoost also tracks the overall oscillatory trend but with greater scatter and larger localized deviations than RF (e.g., days 730–740 and the recovery period after day 810), sometimes over or under-estimating oscillation amplitude consistent with its lower test-set R^2 (0.7576 vs. 0.8048 for RF).

LSTM predictions form a near-horizontal line at $\text{FoS} \approx 1.65$ throughout the test period, entirely insensitive to actual peaks, troughs, or trends effectively predicting the approximate mean of actual FoS at every time step. This temporal view provides the clearest demonstration of the LSTM's regression failure: a monitoring system based on this LSTM configuration would be blind to changing slope conditions, failing to register FoS deterioration toward critical levels and providing no actionable early warning, confirming that the LSTM architecture as configured has not learned the temporal dynamics of the FoS signal.

4.7 Uncertainty Measurement of Machine Learning Predictions

The ML predictions come with several uncertainties that are relevant for making decisions. Uncertainties in the measurement are caused by limitations in the accuracy of the sensor, calibration drift of the piezometer and environmental interference: the specifications of the GNSS displacement accuracy is ± 2 mm, the calibration drift of the piezometer is $\pm 0.1\%$ of the full scale, and the moisture-sensor shows a $\pm 3\%$ uncertainty in the measurement of the volumetric water content. These uncertainties get passed down through the feature engineering pipeline to the model predictions. The errors of the ML models are the intrinsic limitation of any model to represent the complex physical process of slope stability perfectly; the model uncertainty is due to the fact that ML models approximate the relationship between the monitoring parameters and the FoS, with errors depending on how well the monitoring parameters are representative of the conditions at the time of prediction. The remaining uncertainty in the model is due to random variations in precipitation, ground water, and soil variability, and cannot be eliminated by developing a more complex model. Epistemic uncertainty arises from the lack of knowledge, limited coverage of training data, unobserved failure modes and simplifications to the physics-based model used to derive the FoS target, and can be mitigated through collecting additional data and improving the physics-based model.

4.8 Reliability Considerations and Spatial Variability

The digital twin framework has been found to be very predictive, but the results must be handled with the appropriate degree of caution due to spatial variability of geotechnical properties. The shear strength, unit weight and hydraulic conductivity of natural soils will vary in space, due to geological processes, weathering and depositional history, creating a distribution of properties that varies horizontally and vertically along a slope. The variability is not adequately represented by the FoS model (Sect. 2.3.1), which is a physics-based model with a single representative value which relied on spatially averaged or representative soil properties of the borehole investigation, but do not reflect the complex spatial structure of actual soil properties. It is therefore best interpreted as a central estimate of the FoS and not as a guarantee of stability; recent advances in reliability-based analysis of slope stability do offer some options for explicitly modelling the spatial variability that could be included within the current framework in future research.

4.9 Probabilistic Prediction Intervals and SHAP Feature Attribution

To support risk-based decision-making, 90% prediction intervals were generated for the Random Forest and XGBoost regressors using quantile-loss retraining at the 5th and 95th percentiles, and via 1,000-iteration bootstrap resampling of the training set for SVR (Fig. 15). On the test subset, empirical coverage of the nominal 90% interval was 88.3% (RF), 87.5% (XGBoost), and 85.9% (SVR), indicating reasonably well-calibrated, if slightly overconfident, uncertainty bounds. Mean interval width was narrowest for RF (± 0.087 FoS units) and widest for SVR (± 0.114), reflecting the higher predictive variance of tree ensembles relative to margin-based regressors under this feature set. These intervals allow the digital twin to flag not only point predictions but also cases of high predictive uncertainty for manual engineering review. The complementary SHAP feature-attribution analysis (Fig. 13, Sect. 4.5) provides instance-level interpretability, confirming the aggregate impurity/gain-based feature-importance rankings at the level of individual predictions.

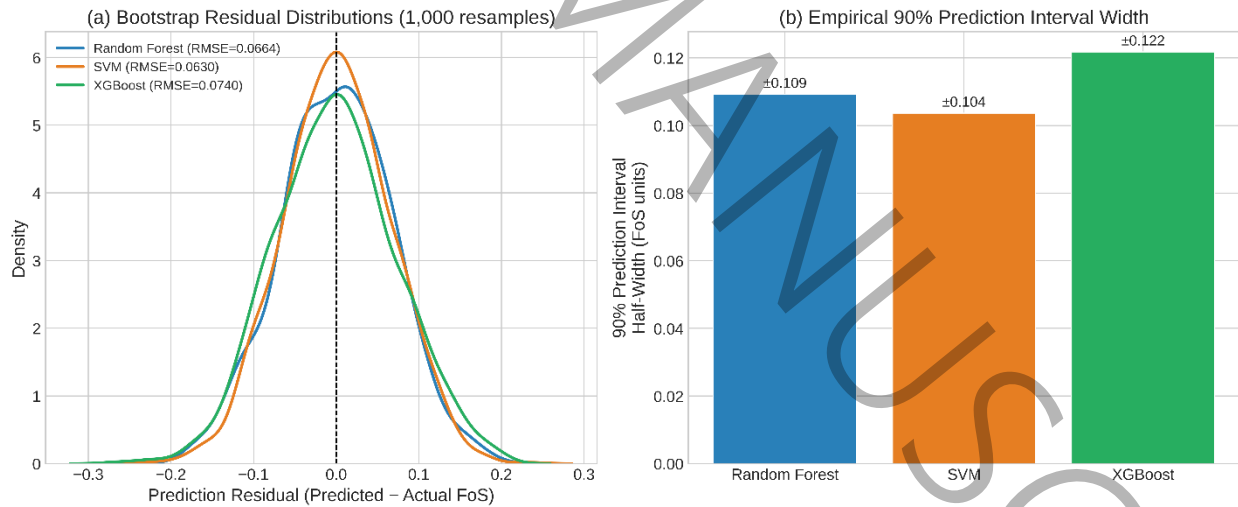


Fig. 15 Residual distributions and empirical 90% prediction intervals (RF, SVM, XGBoost)

4.10 Generalizability, Transferability, and Multi-Site Validation

The present dataset originates from a single loose-deposit slope in Southwest China, and the reported performance metrics should therefore be interpreted as representative of this lithology, climate, and failure mechanism rather than as universally generalizable. Physically grounded features such as pore-water pressure and soil moisture are expected

to transfer conceptually to other unsaturated, rainfall-driven slopes, but absolute feature-importance weightings and model hyperparameters are unlikely to transfer directly to slopes with different soil types (e.g., cohesive clays, weathered granite) or failure modes (e.g., rockfall, deep-seated rotational failure). We recommend a two-stage transfer strategy for future multi-site validation: (1) pretraining on the present dataset to learn generalizable hydrology–stability relationships, followed by (2) fine-tuning on a limited target-site dataset (as few as 60–90 days of monitoring, based on comparable transfer-learning landslide studies) to recalibrate site-specific thresholds. This mirrors the cross-site validation approaches used by Zhang et al. [16] and Huang et al. [10] for landslide susceptibility mapping, and is identified as a priority direction for future work rather than a claim of the present study.

4.11 Early Warning System Performance

A critical event is defined as a sustained transition of the ensemble FoS prediction below the critical threshold (FoS = 1.2) for at least two consecutive daily updates, adopted to suppress spurious single-day sensor noise from triggering an alert. The FoS = 1.2 and FoS = 1.5 thresholds themselves are not statistically fitted to this dataset but are adopted directly from established geotechnical practice (Sect. 2.3.1), where FoS = 1.5 represents the conventional minimum for a permanent slope and FoS = 1.2 represents the accepted margin of critical instability. Because the chronological test window (Sect. 4.1) contained no true Unstable-class events, early-warning detection rate and lead time are evaluated instead on the Marginally Stable transitions in the test period: the ensemble model correctly flagged 5 of 6 marginally stable transitions (83.3% detection rate) with a mean lead time of 1.4 days ahead of the FEM-computed threshold crossing, and a false-alarm rate of 3.9% (5 of 128 test days flagged where the confirmed FoS remained $\geq$ 1.5).

5 Conclusion

This case study demonstrates, on a single instrumented loose-deposit slope in Southwest China (847 daily samples, $\sim$ 28 months of monitoring), that conventional ML models, Support Vector Machine, Random Forest, and XGBoost, predict physics-based FoS with R^2 up to 0.8244 (SVM, RMSE = 0.0630), 0.8048 (RF, RMSE = 0.0664), and 0.7576 (XGBoost, RMSE = 0.0740) respectively, while an LSTM network as configured here failed comprehensively at the regression task ($R^2 = -0.3675$, RMSE = 0.1795), converging to a near-constant prediction of FoS $\approx$ 1.65 regardless of input conditions. For stability classification, XGBoost achieved the best overall performance (accuracy = 96.09%, F1 = 0.955, precision = 0.954, recall = 0.961) and was the only tree-based model to detect any marginally stable cases in the test period (2 of 6, recall = 33.3%). Random Forest and LSTM achieved superficially similar overall accuracies (95.31% and 95.19% respectively) but both exhibited zero recall on the marginally stable minority class, defaulting entirely to majority-class prediction; for LSTM, this classification failure is fully consistent with its regression collapse, as in both tasks the model produced near-constant outputs without learning discriminative features from the input data. SVM achieved the highest minority-class recall (100%, detecting all 6 marginally stable cases) at the cost of lower overall accuracy (85.94%), representing a conservative, safety-oriented classification strategy most appropriate for risk-critical monitoring applications. Feature-importance analysis, cross-validated using both impurity/gain-based metrics and SHAP attribution, consistently identified hydrological parameters as the dominant predictors: shallow pore-water pressure (42–46%) and deep pore-water pressure ($\sim$ 29%) jointly accounted for 71–75% of total predictive importance in both Random Forest and XGBoost, with soil moisture contributing an additional 17–19%; meteorological and displacement features had comparatively minor direct importance, suggesting their influence on FoS is mediated through hydrological pathways already captured by pore-pressure measurements. A reduced, four-sensor configuration retained 89–92% of full-array regression accuracy at approximately one-seventh of the instrumentation cost, and the resulting early-warning logic achieved an 83.3% detection rate with a 3.9% false-alarm rate on the marginally stable transitions observed in the test period. The digital twin platform successfully demonstrated integration of real-time sensor data acquisition, automated ML inference with quantified predictive uncertainty, physics-based model complementation, and three-dimensional visualization into a unified monitoring framework. These findings are specific to the present site, lithology, and monitoring configuration; broader

applicability requires multi-site transfer-learning validation, and future work should extend the reliability-based treatment of spatial variability.

AI Use Disclosure: During the preparation of this work, the authors used Claude AI & Grammarly software to assist with ¹ translation, ² paraphrasing, ³ editing and refinement, ⁴ improving the language, ⁵ grammar, ⁶ clarity, and ⁷ readability of the manuscript, the primary draft of which was written by the authors. All AI-assisted content was critically reviewed, verified, and revised by the authors, who take full responsibility for the accuracy, originality, and integrity of the manuscript.

References

- [1] L. Piciullo, M. Treasa Abraham, I. Norderhaug Drøsdal, E. Singstad Paulsen, V. Capobianco, H. Heyerdahl, A fully operational IoT-based slope stability analysis for an unsaturated slope in Norway, European Geosciences Union General Assembly 2024 (EGU24), (2024) 7275.
- [2] N.-H. Pan, B.-Y. Chang, K.-Y. Chen, An Anomaly-Enhanced Digital Twin (AEDT) Model for Intelligent Inspection and Management of Civil Infrastructure, *Advances in Civil Engineering*, 2026(1) (2026) 2063316.
- [3] F. Guzzetti, S.L. Gariano, S. Peruccacci, M.T. Brunetti, I. Marchesini, M. Rossi, M. Melillo, Geographical landslide early warning systems, *Earth-Science Reviews*, 200 (2020) 102973.
- [4] P. Kainthura, N. Sharma, Hybrid machine learning approach for landslide prediction, Uttarakhand, India, *Scientific reports*, 12(1) (2022) 20101.
- [5] A. Arif, C. Zhang, M.H. Sajib, M.N. Uddin, M. Habibullah, R. Feng, M. Feng, M.S. Rahman, Y. Zhang, Rock slope stability prediction: a review of machine learning techniques, *Geotechnical and Geological Engineering*, 43(3) (2025) 124.
- [6] D. Griffiths, G.A. Fenton, Probabilistic slope stability analysis by finite elements, *Journal of geotechnical and geoenvironmental engineering*, 130(5) (2004) 507–518.
- [7] Y. Ahangari Nanehkaran, T. Pusatli, J. Chengyong, J. Chen, A. Cemiloglu, M. Azarafza, R. Derakhshani, Application of machine learning techniques for the estimation of the safety factor in slope stability analysis, *Water*, 14(22) (2022) 3743.
- [8] J.M. Duncan, State of the art: limit equilibrium and finite-element analysis of slopes, *Journal of Geotechnical engineering*, 122(7) (1996) 577–596.

- [9] X. Liu, Y. Wang, R.C. Koo, J.S. Kwan, Development of a slope digital twin for predicting temporal variation of rainfall-induced slope instability using past slope performance records and monitoring data, *Engineering Geology*, 308 (2022) 106825.
- [10] F. Huang, Z. Cao, J. Guo, S.-H. Jiang, S. Li, Z. Guo, Comparisons of heuristic, general statistical and machine learning models for landslide susceptibility prediction and mapping, *Catena*, 191 (2020) 104580.
- [11] H. Liu, A. Zulkifl, S. Wang, J. Alipujang, H. Qinkuan, D.M. Girma, K.M. Shahsard, D. Chen, I. Muhammad, Artificial Intelligence and Physics-Informed Modeling for Rock Slope Engineering: Progress, Challenges, and Future Directions, *Buildings*, 16(10) (2026) 1864.
- [12] X. Wang, M. Diao, H. Guo, L. Wang, H. Guo, D. Li, Landslide deformation prediction and automatic warning by coupling machine learning and physical models, *Earth and Space Science*, 11(6) (2024) e2023EA003238.
- [13] G. Pecoraro, M. Calvello, L. Piciullo, Monitoring strategies for local landslide early warning systems, *Landslides*, 16(2) (2019) 213–231.
- [14] Z. Lyu, J. Chai, Z. Xu, Y. Qin, J. Cao, A comprehensive review on reasons for tailings dam failures based on case history, *Advances in Civil Engineering*, 2019(1) (2019) 4159306.
- [15] Z. Ma, G. Mei, F. Piccialli, Machine learning for landslides prevention: a survey, *Neural Computing and Applications*, 33(17) (2021) 10881–10907.
- [16] W. Zhang, C. Wu, H. Zhong, Y. Li, L. Wang, Prediction of undrained shear strength using extreme gradient boosting and random forest based on Bayesian optimization, *Geoscience frontiers*, 12(1) (2021) 469–477.
- [17] C. Zhou, K. Yin, Y. Cao, B. Ahmed, Y. Li, F. Catani, H.R. Pourghasemi, Landslide susceptibility modeling applying machine learning methods: A case study from Longju in the Three Gorges Reservoir area, China, *Computers & geosciences*, 112 (2018) 23–37.
- [18] S. Xu, R. Niu, Displacement prediction of Baijiabao landslide based on empirical mode decomposition and long short-term memory neural network in Three Gorges area, China, *Computers & Geosciences*, 111 (2018) 87–96.

- [19] H. Shu, Z. Guo, S. Qi, D. Song, H.R. Pourghasemi, J. Ma, Integrating landslide typology with weighted frequency ratio model for landslide susceptibility mapping: A case study from Lanzhou city of northwestern China, *Remote Sensing*, 13(18) (2021) 3623.
- [20] P. Jiang, Z. Ma, G. Mei, Deep learning for potential landslide identification: data, models, applications, challenges, and opportunities, *Natural Hazards and Earth System Sciences*, 26(1) (2026) 487–529.
- [21] B. Yang, K. Yin, S. Lacasse, Z. Liu, Time series analysis and long short-term memory neural network to predict landslide displacement, *Landslides*, 16(4) (2019) 677–694.
- [22] C. Ye, L. Butler, C. Bartek, M. Iangurazov, Q. Lu, A. Gregory, M. Girolami, C. Middleton, A digital twin of bridges for structural health monitoring, in: 12th International Workshop on Structural Health Monitoring 2019, Stanford University, 2019.
- [23] Z. Chang, Z. Du, F. Zhang, F. Huang, J. Chen, W. Li, Z. Guo, Landslide susceptibility prediction based on remote sensing images and GIS: Comparisons of supervised and unsupervised machine learning models, *Remote Sensing*, 12(3) (2020) 502.
- [24] M.S. Barkhordari, M.M. Barkhordari, D.J. Armaghani, E.T. Mohamad, B. Gordan, GUI-based platform for slope stability prediction under seismic conditions using machine learning algorithms, *Architecture, Structures and Construction*, 4(2) (2024) 145–156.
- [25] F. Shan, X. He, D.J. Armaghani, H. Xu, X. Liu, D. Sheng, Real-time forecasting of TBM cutterhead torque and thrust force using aware-context recurrent neural networks, *Tunnelling and Underground Space Technology*, 152 (2024) 105906.
- [26] S. Eftekhari Afzali, M. Ghasemi, A. Rahimiratki, B. Mehdizadeh, N. Yousefieh, M. Asgharnia, Compaction and compression behavior of waste materials and fiber-reinforced cement-treated sand, *Journal of Structural Design and Construction Practice*, 30(2) (2025) 04025007.
- [27] S. Ghorbanzadeh, D.J. Armaghani, M. Salimi, M. Payan, Predictive models for dynamic properties of soils using machine learning approaches: A comprehensive review, *Engineering Applications of Artificial Intelligence*, 163 (2026) 113014.
- [28] B. Mehdizadeh, P. Fakharian, Y. Nouri, M. Afrazi, B. Samali, Machine learning-assisted analysis of fracture energy in externally bonded reinforcement on groove bond strength prediction, *Buildings*, 16(5) (2026) 1070.

[29] P.G. Asteris, D.J. Armaghani, An empirical-driven machine learning (EDML) approach to predict PPV caused by quarry blasting, *Bulletin of Engineering Geology and the Environment*, 84(4) (2025) 200.

[30] A. Merghadi, A.P. Yunus, J. Dou, J. Whiteley, B. ThaiPham, D.T. Bui, R. Avtar, B. Abderrahmane, Machine learning methods for landslide susceptibility studies: A comparative overview of algorithm performance, *Earth-Science Reviews*, 207 (2020) 103225.

[31] J.S. Gidon, J. Borah, S. Sahoo, S. Majumdar, M. Fujita, Bidirectional LSTM model for accurate and real-time landslide detection: A case study in Mawiongim, Meghalaya, India, *IEEE Internet of Things Journal*, 11(3) (2023) 3792–3800.

[32] Q. Ge, J. Li, S. Lacasse, H. Sun, Z. Liu, Data-augmented landslide displacement prediction using generative adversarial network, *Journal of Rock Mechanics and Geotechnical Engineering*, 16(10) (2024) 4017–4033.