Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.080077

ARTICLE

Novel Sine-Lucas Oscillation Particle Swarm Optimization XGBoost Method for Landslide Prediction

Qisen Jin[1,2], Xiaoping Wang[1], Feng Zhang[2], Yu Zeng[2], Jia Guo[3,4,5,*] and Jiacheng Li[6,*]

[1]Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, China
[2]China Communications Second Highway Consultants Co., Ltd., Wuhan, China
[3]Hubei Key Laboratory of Digital Finance Innovation, Hubei University of Economics, Wuhan, China
[4]Faculty of Information Science and Technology, Wenhua College, Wuhan, China
[5]Faculty of Computer and Information Science, Hosei University, Tokyo, Japan
[6]Department of Applied Mathematics and Systems, Faculty of Informatics, Kanagawa University, Yokohama, Japan

*Corresponding Authors: Jia Guo. Email: guojia314@gmail.com; Jiacheng Li. Email: lijiacheng@kanagawa-u.ac.jp

Received: 02 February 2026; Accepted: 18 June 2026; Published: 13 August 2026

ABSTRACT: This paper presents a Sine-Lucas Oscillation Particle Swarm Optimization (SLOPSO) XGBoost algorithm for landslide susceptibility prediction. SLOPSO extends canonical PSO by introducing an oscillation factor constructed from a sine function and the Lucas number sequence into the position update, which raises swarm diversity and improves the global search. SLOPSO is then applied to tune six XGBoost hyperparameters using K-fold cross-validation accuracy as the fitness function. In experimental evaluation, SLOPSO-XGBoost reaches a mean test Accuracy of 0.8113, F1 of 0.8182, and AUC of 0.8856 over 10 independent runs, outperforming standard PSO-XGBoost and the default Random Forest, SVM, and XGBoost baselines on every metric. The experimental results demonstrate that SLOPSO is capable of automatically tuning the hyperparameters of XGBoost, and that SLOPSO-XGBoost able to provide high precision solution for landslide event prediction.

KEYWORDS: Sine-Lucas oscillation; particle swarm optimization; XGBoost; landslide prediction

1 Introduction

Landslides [1–3] are common geological hazards that pose a significant threat to humans, infrastructure, and ecosystems. Over the past few decades, landslides have caused numerous casualties, permanently degraded vast areas of forest and farmland, and severely damaged regional ecosystem balance and biodiversity [4,5].

Landslide prediction [6–8] is a critical component of disaster risk reduction and brings important benefits to communities in landslide-prone regions. By identifying potential slope failures before they occur, prediction systems enable early mitigation, evacuation, and resource allocation [9,10], substantially reducing casualties, infrastructure damage, and economic losses. Prediction models also inform land-use planning, building code regulations, and infrastructure development in susceptible regions, supporting long-term community resilience [11,12]. Integrating landslide prediction into broader disaster management improves societal preparedness and helps protect communities that often inhabit high-risk terrain due to socioeconomic constraints [13,14]. High-precision landslide prediction therefore plays an important role in agriculture and land resource protection, ecosystem conservation and restoration, and urban planning and land-use management.

Effective landslide prediction [15–18] is essential for risk mitigation: it supports early warning, land-use planning, and resource allocation in disaster management. Landslides are typically triggered by the interaction of multiple factors, including sustained heavy rainfall, seismic activity, freeze-thaw cycles, volcanic eruptions, and anthropogenic activities such as slope-base excavation and crest overloading [19–21]. Several machine learning methods have been widely applied to this task, including Support Vector Machines (SVM) [22], Random Forest (RF) [23], and XGBoost [24].

More advanced learning strategies [25–27] can further improve prediction accuracy but often at the cost of longer computation time. Feature selection [28,29] is therefore commonly used to reduce data redundancy, lower the risk of overfitting, and improve efficiency. Metaheuristic algorithms are widely applied to optimization problems such as feature selection, image classification, and neural network hyperparameter tuning, owing to their strong search ability in high-dimensional spaces [30,31].

The main contributions of this work are summarized:

(1) A novel Sine-Lucas Oscillation Particle Swarm Optimization (SLOPSO) algorithm is proposed. By introducing an oscillation factor constructed from a sine function and the Lucas number sequence into the position update of canonical PSO, SLOPSO enhances global search ability in hyperparameter optimization tasks.

(2) A unified SLOPSO-XGBoost framework is developed for landslide susceptibility prediction. SLOPSO automatically optimizes six critical XGBoost hyperparameters using K-fold cross-validation accuracy as the fitness, eliminating manual tuning. The remainder of this paper is organized as follows. Section 2 introduces the proposed Sine-Lucas Oscillation Particle Swarm Optimization Algorithm. Section 3 presents the experiments and results. Section 4 discusses the limitation and future Work. Section 5 concludes the paper.

2 Sine-Lucas Oscillation Particle Swarm Optimization Algorithm for Landslide Prediction

2.1 The Proposed Sine-Lucas Oscillation Particle Swarm Optimization

The Sine-Lucas Oscillation Particle Swarm Optimization (SLOPSO) is proposed in this section. SLOPSO augments the canonical PSO with a Sin–Lucas adaptive oscillation factor that modulates the position update at each iteration, achieving a dynamic balance between broad exploration in the early iterations and fine-grained exploitation near convergence. At each iteration, the velocity and position of every particle are updated by Eq. (1).

$$
\begin{aligned}
v_i^{t+1} &= \omega v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g - x_i^t) \\
x_i^{t+1} &= x_i^t + W(t)\, v_i^{t+1} \\
W(t) &= 1 + \frac{\sin(t\pi/2)}{L(t)} \\
L_n &= L_{n-1} + L_{n-2}, \quad L_0 = 2,\ L_1 = 1
\end{aligned}
\tag{1}
$$

where x_i^t and v_i^{t+1} denote the position of particle i at iteration t and its updated velocity at iteration $t+1$, respectively; ω is the inertia weight; c_1 and c_2 are the cognitive and social acceleration coefficients; $r_1, r_2 \in U[0,1]$ are random numbers; p_i is the personal best position of particle i and g is the global best position; $W(t)$ is the Sin–Lucas oscillation factor that modulates the position step at iteration t; $L(t)$ is the t-th Lucas number; and $t \in \{0, 1, \ldots, T-1\}$ is the iteration index with T being the maximum number of iterations. When $W(t) \equiv 1$, Eq. (1) reduces to the canonical PSO.

2.2 SLOPSO Optimized XGBoost

XGBoost (Extreme Gradient Boosting) is a high-performance, scalable machine learning algorithm based on gradient boosted decision trees that has gained prominence due to its exceptional predictive accuracy and computational efficiency. Details of XGBoost are shown in Eq. (2).

$$\mathrm{Obj}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k); \tag{2}$$

where $l(y_i, \hat{y}_i)$ is the loss function measuring prediction error, $\Omega(f_k)$ represents the regularization term, θ denotes the set of hyper-parameters, n is the number of training samples, and K is the number of trees in the model.

In the landslide prediction task, SLOPSO is used to optimize six XGBoost hyperparameters: *learning_rate*, the step size of gradient descent; *max_depth*, the maximum depth of each tree; *min_child_weight*, the minimum sum of instance weights required to split a node, which acts as a regularization term; *gamma*, the minimum loss reduction required for a further split; *subsample*, the fraction of training samples used to grow each tree; and *colsample_bytree*, the fraction of features sampled for each tree.

The fitness function is the K-fold stratified cross-validation accuracy on the training set ($K = 3$ in this work). For each candidate hyperparameter vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_6)$, an XGBoost classifier is trained on $K-1$ folds and evaluated on the held-out fold; the mean accuracy over the K folds gives the fitness of the particle. The search ranges and SLOPSO-specific parameters (swarm size and number of iterations) are listed in Table 1. The flowchart and pseudocode of SLOPSO-XGBoost are shown in Fig. 1 and Algorithm 1.

Table 1: Hyperparameter settings of the default XGBoost baseline, the PSO-XGBoost, and the SLOPSO-XGBoost optimizers.

Parameter	XGBoost (default)	PSO-XGBoost	SLOPSO-XGBoost
learning_rate	0.3	[0.01, 0.50]	[0.01, 0.50]
max_depth	6	[2, 15]	[2, 15]
min_child_weight	1	[1, 10]	[1, 10]
Gamma	0	[0, 1.0]	[0, 1.0]
Subsample	1.0	[0.5, 1.0]	[0.5, 1.0]
colsample_bytree	1.0	[0.5, 1.0]	[0.5, 1.0]

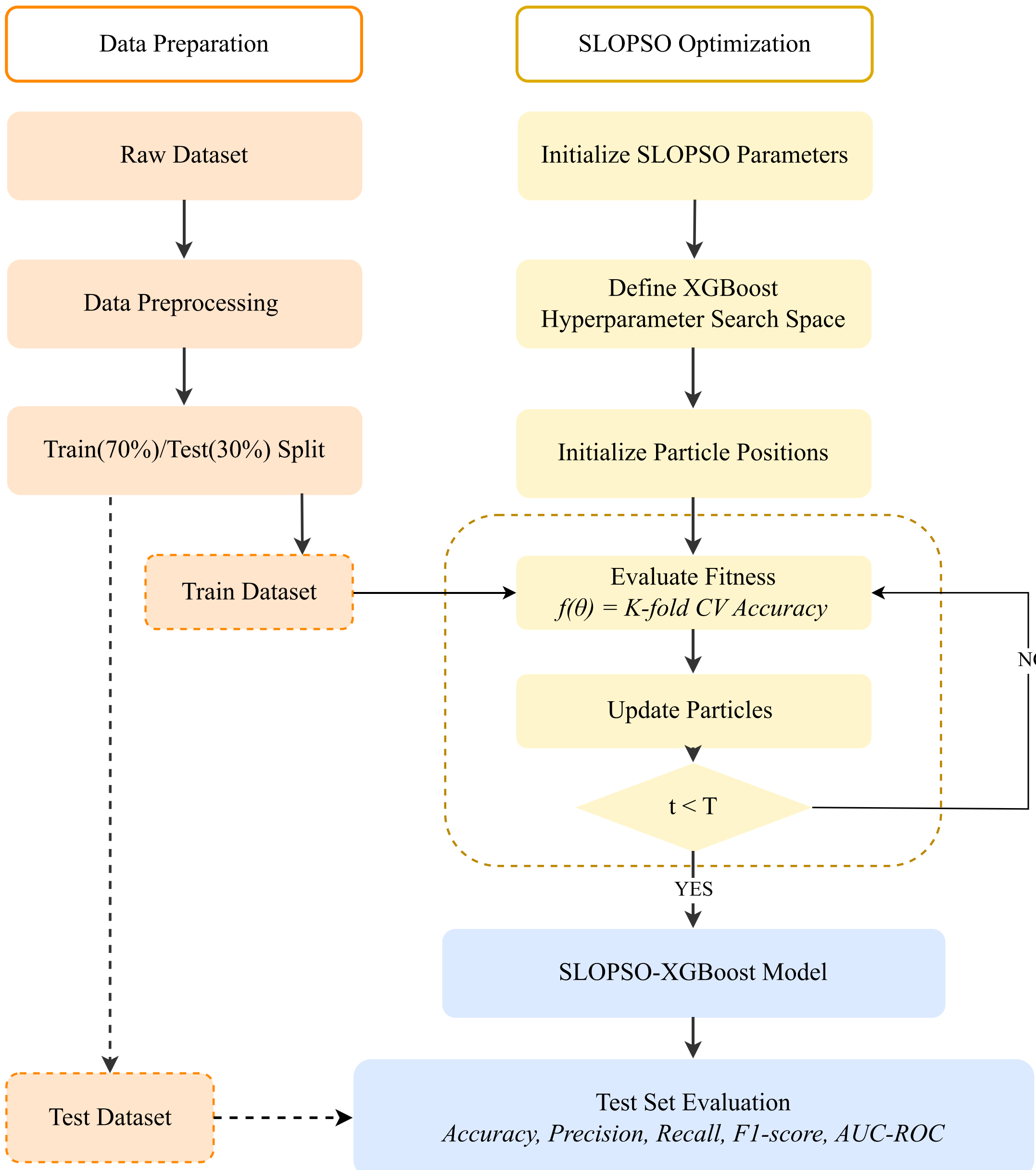

Figure 1: Flowchart of SLOPSO-XGBoost.

Algorithm 1: Sine-Lucas oscillation particle swarm optimization (SLOPSO).

Require: Objective function $f(\mathbf{x})$, dimension D, swarm size N, maximum iterations T, bounds $[lb, ub]$, inertia weight ω, acceleration coefficients c_1, c_2

Ensure: Global best solution $\mathbf{x}^*$ and fitness f^*

1: Initialize Lucas sequence: $L_0 \leftarrow 2,\ L_1 \leftarrow 1,\ L_n \leftarrow L_{n-1} + L_{n-2}$ for $n \geq 2$

2: **for** $i = 1$ **to** N **do**

3: $\quad \mathbf{x}_i \leftarrow$ random vector uniformly drawn from $[lb, ub]^D$

4: $\quad \mathbf{v}_i \leftarrow$ random vector uniformly drawn from $[-\frac{1}{2}(ub - lb),\ \frac{1}{2}(ub - lb)]^D$

(Continued)

Algorithm 1 (continued)

    $\mathbf{p}_i \leftarrow \mathbf{x}_i, f_{p_i} \leftarrow f(\mathbf{x}_i)$
end for
$\mathbf{g} \leftarrow \arg\min_i f_{p_i}, f_g \leftarrow \min_i f_{p_i}$
for $t = 1$ **to** T **do**
    Compute oscillation factor: $W(t) \leftarrow 1 + \sin(t\pi/2)/L_t$
    for $i = 1$ **to** N **do**
        $\mathbf{v}_i \leftarrow \omega \mathbf{v}_i + c_1 \mathbf{r}_1 \odot (\mathbf{p}_i - \mathbf{x}_i) + c_2 \mathbf{r}_2 \odot (\mathbf{g} - \mathbf{x}_i)$
        $\mathbf{x}_i \leftarrow \mathbf{x}_i + W(t)\,\mathbf{v}_i$ ▷ SLOPSO position update
        Clip $\mathbf{x}_i$ to $[lb, ub]$
        $f_i \leftarrow f(\mathbf{x}_i)$
        if $f_i < f_{p_i}$ **then**
            $\mathbf{p}_i \leftarrow \mathbf{x}_i, f_{p_i} \leftarrow f_i$
        end if
        if $f_{p_i} < f_g$ **then**
            $\mathbf{g} \leftarrow \mathbf{p}_i, f_g \leftarrow f_{p_i}$
        end if
    end for
end for
return $\mathbf{x}^* \leftarrow \mathbf{g}, f^* \leftarrow f_g$

3 Experiments and Results

3.1 Datasets and Parameters

The experimental datasets comprises 1212 records (606 landslide points and 606 non-landslide points). To ensure experimental fairness and training effectiveness, the dataset was randomly partitioned into a training set (848 samples) and a test set (364 samples) following a 70/30 stratified split, with a balanced class distribution of 50% positive and 50% negative samples in each subset [26,27]. The data contain binary target variable Landslide and 12 conditioning factors:

- Aspect: slope orientation
- Curvature: terrain curvature
- Earthquake: seismic hazard zone
- Elevation: height above sea level
- Flow: flow accumulation
- Lithology: bedrock composition
- NDVI: Normalized Difference Vegetation Index
- NDWI: Normalized Difference Water Index
- Plan: planar curvature
- Precipitation: mean annual rainfall
- Profile: profile curvature
- Slope: terrain steepness

The correlation of different features is shown in Fig. 2. For PSO and SLOPSO, the maximum number of iterations is 120, the population size is 20, and the number of independent runs is 10.

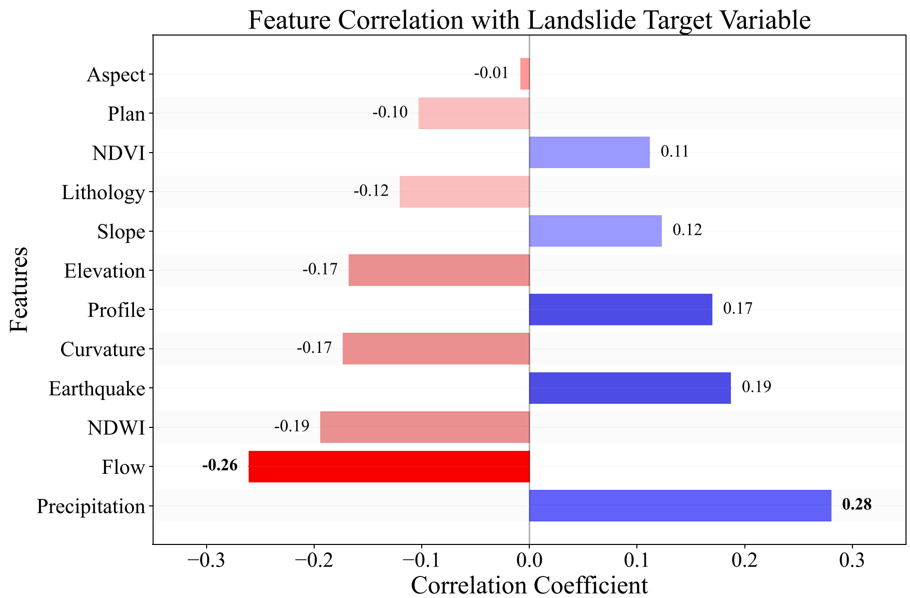

Figure 2: Correlation of different features.

3.2 Experimental Results and Discussion

Table 2 reports the test-set performance of the five methods on Accuracy, Precision, Recall, F1-score, and AUC-ROC. Among the three baselines using default hyperparameters, SVM ranks first (Accuracy 0.8022, AUC-ROC 0.8792), followed by Random Forest (0.7940, 0.8712); the default XGBoost trails behind on every metric (0.7720, 0.8602). The relatively poor performance of untuned XGBoost is consistent with the well-known sensitivity of gradient boosting to its learning rate, tree depth, and regularization terms.

Table 2: Performance comparison of Random Forest, SVM, XGBoost, PSO-XGBoost, and SLOPSO-XGBoost. The best value of each column is in **bold**.

Method	Accuracy	Precision	Recall	F1 Score	AUC-ROC
Random Forest	0.7940	0.7831	0.8132	0.7978	0.8712
SVM	0.8022	0.7835	0.8352	0.8085	0.8792
XGBoost	0.7720	0.7538	0.8077	0.7798	0.8602
PSO-XGBoost	0.8099	0.7881	0.8478	0.8168	0.8824
SLOPSO-XGBoost	**0.8113**	**0.7893**	**0.8495**	**0.8182**	**0.8856**

Hyperparameter optimization with canonical PSO substantially reshapes this ranking. Compared with the default XGBoost, PSO-XGBoost improves Accuracy by 3.79 percentage points (0.7720 → 0.8099), F1-score by 3.70 (0.7798 → 0.8168), and AUC-ROC by 2.22 (0.8602 → 0.8824). The gain is largest in Recall, which rises by 4.01 percentage points (0.8077 → 0.8478)—a desirable property for landslide prediction, where a missed positive corresponds to an undetected hazardous zone. After tuning, XGBoost surpasses both Random Forest and SVM on all five metrics.

SLOPSO-XGBoost further improves over PSO-XGBoost on every metric, although the margins are smaller than the gains from tuning itself: Accuracy rises from 0.8099 to 0.8113 (+0.14 pp), Precision from 0.7881 to 0.7893 (+0.12 pp), Recall from 0.8478 to 0.8495 (+0.17 pp), F1-score from 0.8168 to 0.8182 (+0.14 pp), and AUC-ROC from 0.8824 to 0.8856 (+0.32 pp). The improvement is consistent across all five metrics, indicating that the Sin-Lucas oscillation factor helps the swarm locate slightly better regions of the hyperparameter space than canonical PSO.

Relative to the traditional baselines, SLOPSO-XGBoost shows clear advantages. Against the strongest baseline (SVM), the gains are 0.91 percentage points in Accuracy, 0.58 in Precision, 1.43 in Recall, 0.97 in F1-score, and 0.64 in AUC-ROC. Against Random Forest, the improvements widen to 1.73, 0.62, 3.63, 2.04, and 1.44 percentage points in the same metrics, respectively. All five methods exhibit Recall greater than Precision, and SLOPSO-XGBoost shows the largest Recall–Precision gap (0.8495 vs. 0.7893, about 6 percentage points), reflecting the optimizer's preference for high-sensitivity configurations—an alignment with the asymmetric cost of false negatives in disaster prevention. Details of the performance comparison are shown in Fig. 3.

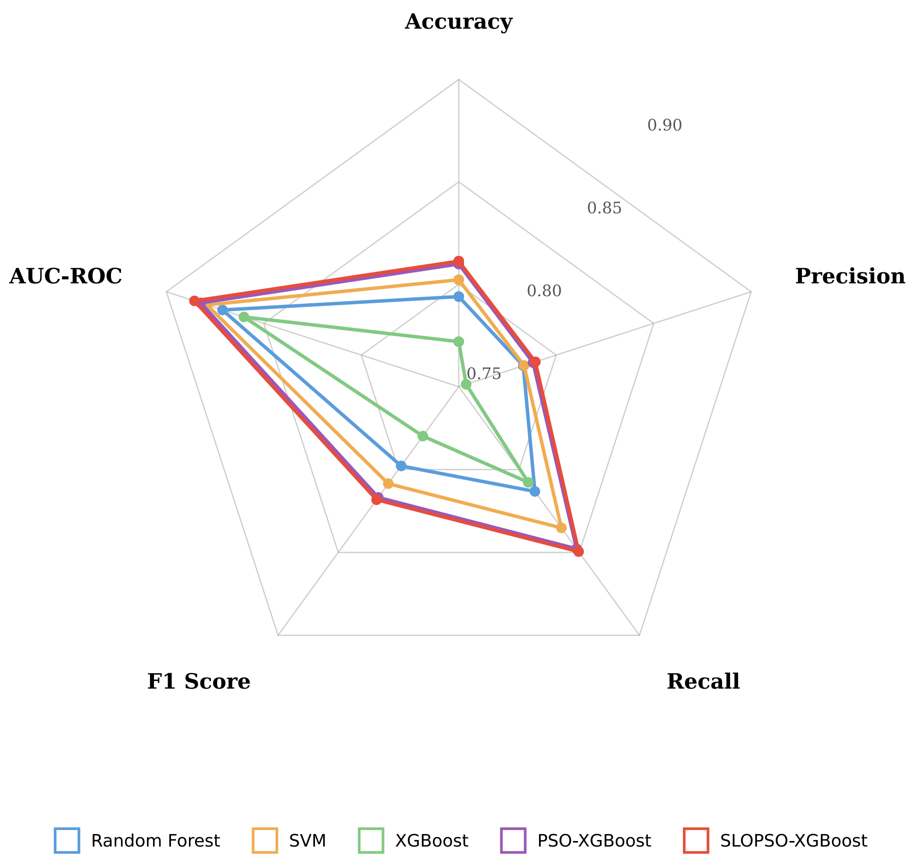

Figure 3: Performance comparison of Random Forest, SVM, XGBoost, PSO-XGBoost, and SLOPSO-XGBoost on the five test-set metrics.

Fig. 4 plots the convergence curves of PSO and SLOPSO during training. Table 3 reports the mean and standard deviation of the six evaluation metrics over 10 independent runs of PSO and SLOPSO. SLOPSO obtains a higher mean than PSO on every metric, with the largest absolute gain in AUC-ROC, followed by Recall, F1-score and Test Accuracy, Precision, and CV Accuracy. The improvement in CV Accuracy is of the same order as that in Test Accuracy, indicating that the additional optimization gain is reflected both at the cross-validation stage (used as the fitness function) and at the final test-set evaluation.

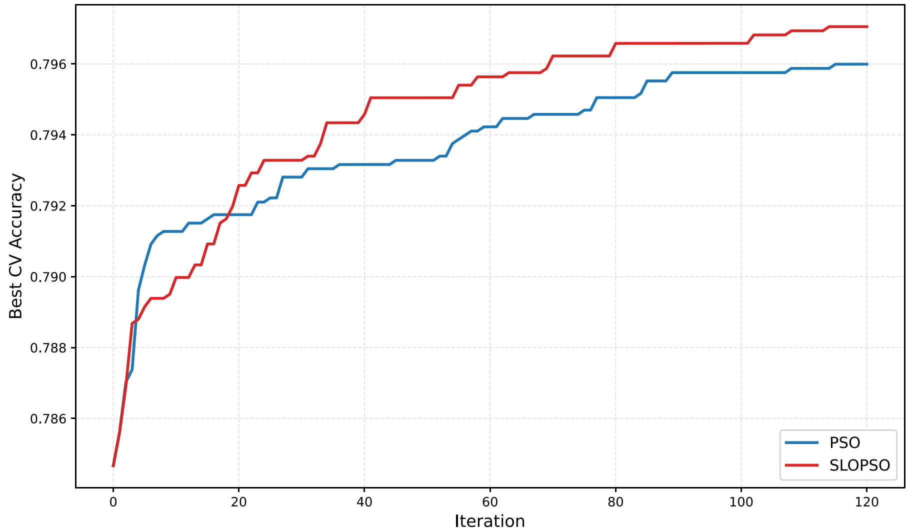

Figure 4: Convergence comparison of PSO and SLOPSO.

Table 3: Performance comparison of PSO-XGBoost and SLOPSO-XGBoost for the landslide prediction task. The better mean value of each row is in **bold**.

Metric	PSO		SLOPSO	
	Mean	**Std**	**Mean**	**Std**
CV Accuracy	0.7960	0.0025	**0.7971**	0.0021
Test Accuracy	0.8099	0.0124	**0.8113**	0.0119
Precision	0.7881	0.0118	**0.7893**	0.0138
Recall	0.8478	0.0154	**0.8495**	0.0102
F1 Score	0.8168	0.0121	**0.8182**	0.0107
AUC-ROC	0.8824	0.0057	**0.8856**	0.0055

3.3 SHAP-Based Feature Contribution Analysis

The interpretability of a learning model is essential in geohazard assessment, since end users need to understand which factors drive a prediction. To interpret the SLOPSO-XGBoost model, we apply SHAP (SHapley Additive exPlanations), which decomposes each prediction into per-feature contributions derived from Shapley values in cooperative game theory. The SHAP value of a feature for a given sample quantifies how much that feature pushes the model output above or below the average prediction, allowing us to identify which conditioning factors drive landslide predictions and in which direction.

Fig. 5 shows the SHAP beeswarm plot for the SLOPSO-XGBoost model on the test set. Features are ordered top-down by mean absolute SHAP value, and the three most influential factors are Precipitation, Flow, and Elevation. Each point corresponds to a test sample; its horizontal position is the SHAP value (positive means a higher predicted landslide probability) and its colour encodes the feature value (red for high, blue for low). Precipitation exhibits a clear monotonic effect: high values (red) lie on the positive SHAP side while low values (blue) lie on the negative side, indicating that heavier rainfall increases the predicted landslide probability, consistent with rainfall as a primary triggering factor. For Flow, the pattern is reversed: high flow accumulation (red) is associated with strongly negative SHAP values, whereas low flow (blue) shifts the prediction toward landslide. This is consistent with the geomorphology of the study area, where cells with large upstream contributing areas correspond to drainage channels and depositional zones, while landslide initiation typically occurs on upper slopes with small contributing areas. Elevation shows a similar inverted pattern, with low-elevation samples contributing positively to the landslide prediction; this matches the descriptive statistics of the dataset, in which landslide samples have a lower mean elevation than non-landslide samples. Slope and Profile curvature contribute positively at high values, in line with the geomorphological view that steeper slopes and convex profiles are less stable. Aspect ranks fifth in importance with a wide SHAP distribution despite a near-zero linear correlation with the label, demonstrating that the non-linear tree model captures directional effects of slope orientation that simple linear correlation cannot. The remaining factors (Lithology, NDVI, NDWI, Curvature, Plan, and Earthquake) have SHAP distributions concentrated near zero and contribute less to the predictions of this model.

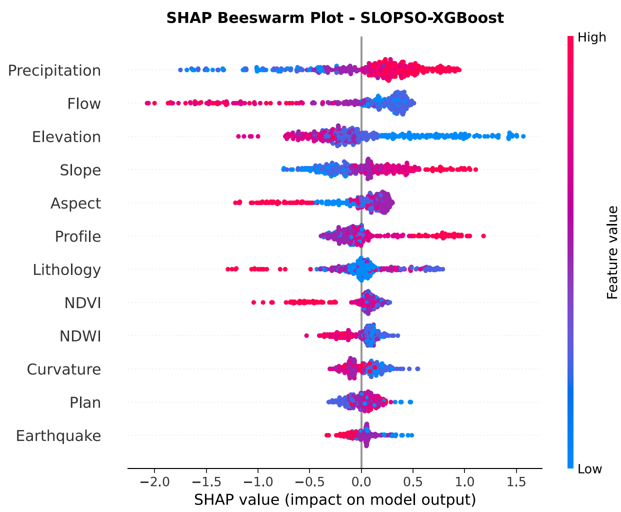

Figure 5: SHAP beeswarm plot of the SLOPSO-XGBoost model.

Fig. 6 shows the SHAP dependence plots for the four most influential features. In each panel, the horizontal axis is the feature value, the vertical axis is the SHAP value contributed by that feature, and the colour encodes a second feature that interacts most strongly with the target feature. Precipitation and Slope exhibit monotonic positive effects, indicating that heavier rainfall and steeper terrain consistently increase landslide susceptibility. In contrast, Elevation shows a monotonic negative trend, with low-elevation cells contributing most strongly to landslide probability. Flow exhibits a non-monotonic pattern, where low and moderate flow accumulation is associated with positive contributions while high flow accumulation produces strongly negative SHAP values, reflecting the distinction between upper hillslopes where landslides initiate and low-lying drainage channels where deposition occurs.

4 Limitations and Future Work

4.1 Limitations

The study has three limitations. First, SLOPSO is used here to tune hyperparameters within a fixed XGBoost architecture; the model structure itself is not part of the search. Coupling SLOPSO with Neural Architecture Search (NAS) would extend the framework to jointly discover the model architecture and its hyperparameters, which is particularly relevant when the underlying model is a deep neural network rather than a tree ensemble. Second, each fitness evaluation in SLOPSO requires training an entire XGBoost model with K-fold cross-validation; the total training time scales with both the dataset size and the model complexity, and can become prohibitive on large-scale problems. Third, the proposed SLOPSO-XGBoost pipeline has not been deployed for real-time prediction. Integrating it with online data streams such as live rainfall and seismic feeds is required before the method can be used operationally for in-situ landslide warning.

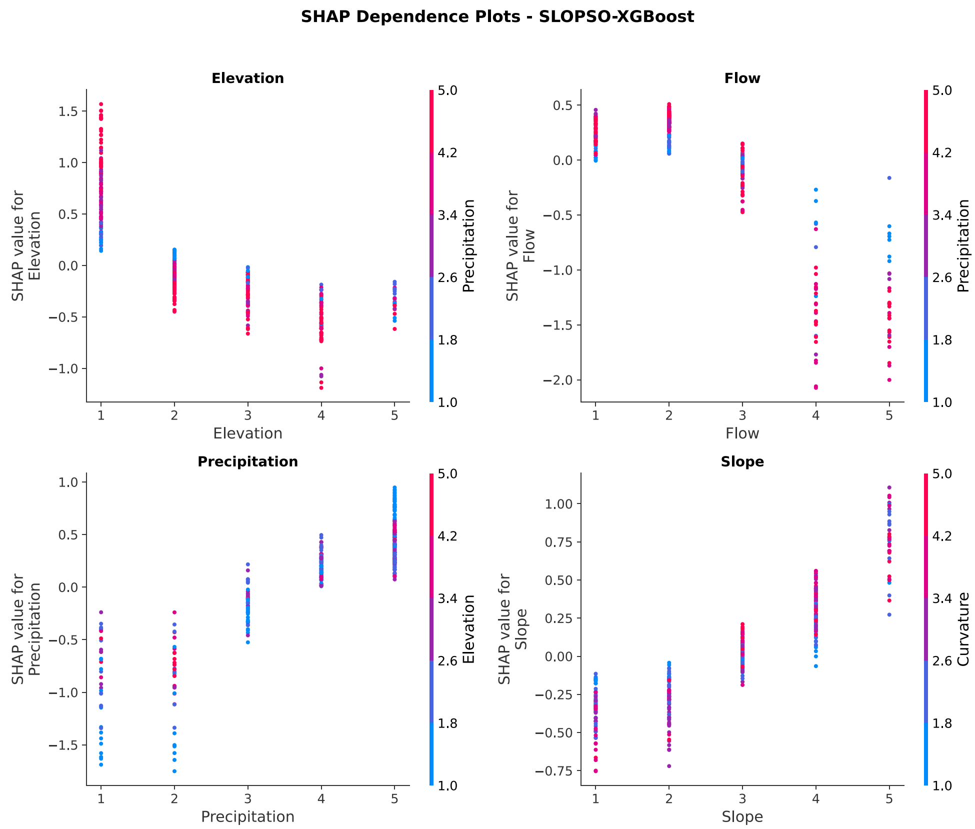

Figure 6: SHAP dependence plot of the SLOPSO-XGBoost model.

4.2 Future Work

Several directions can address the limitations above. To overcome the training-time bottleneck, surrogate-based fitness approximation or asynchronous parallel evaluation can reduce the number of full training runs. To enable real-time inference, the optimized model can be embedded in a streaming pipeline coupled with live rainfall and seismic feeds. Beyond addressing limitations, the framework can also be applied to additional landslide datasets in different geological contexts, and the oscillation factor admits generalisations such as Fibonacci or higher-order Lucas-type sequences and alternative trigonometric carriers (cosine, sigmoid-based) that may yield different exploration–exploitation balances.

5 Conclusion

This paper introduces SLOPSO, a particle swarm optimizer in which the position update is modulated by a Sin-Lucas oscillation factor built from a sine function and the Lucas number sequence. The factor enhances swarm diversity in the early iterations and gradually relaxes to the standard PSO update. We apply SLOPSO to tune six XGBoost hyperparameters using K-fold cross-validation accuracy as the fitness function, while keeping the test set held out from the optimization process.

Across 10 independent runs on the landslide dataset (1212 grid samples, 12 conditioning factors), SLOPSO-XGBoost reaches a mean test Accuracy of 0.8113, F1-score of 0.8182, and AUC-ROC of 0.8856, ahead of canonical PSO-XGBoost and the default Random Forest, SVM, and XGBoost baselines on every metric. SHAP analysis identifies Precipitation, Flow, Elevation, and Slope as the dominant predictors, with directions matching the geomorphology of the study area. Overall, the Sin-Lucas oscillation factor yields a small but stable improvement over canonical PSO for landslide susceptibility modeling.

Future work will extend the SLOPSO-XGBoost framework to full-coverage raster prediction, enabling the generation of spatially continuous landslide susceptibility zonation maps classified into meaningful hazard levels, with systematic validation against historical landslide inventories across diverse geological settings.

Acknowledgement: Not applicable.

Funding Statement: The research was conducted under the auspices of the Hosei International Fund (HIF) Foreign Scholars Fellowship.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Qisen Jin and Xiaoping Wang; methodology, Qisen Jin, Feng Zhang and Yu Zeng; software, Qisen Jin and Yu Zeng; validation, Jia Guo and Jiacheng Li; formal analysis, Feng Zhang and Yu Zeng; investigation, Jia Guo and Jiacheng Li; resources, Qisen Jin and Yu Zeng; data curation, Qisen Jin and Xiaoping Wang; writing—original draft preparation, Qisen Jin and Jia Guo; writing—review and editing, Jia Guo and Jiacheng Li; visualization, Qisen Jin and Xiaoping Wang; supervision, Jia Guo and Jiacheng Li; project administration, Jia Guo and Jiacheng Li; funding acquisition, Jia Guo. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the authors.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Huang F, Xiong H, Jiang SH, Yao C, Fan X, Catani F, et al. Modelling landslide susceptibility prediction: a review and construction of semi-supervised imbalanced theory. Earth Sci Rev. 2024;250:104700. doi:10.1016/j.earscirev.2024.104700.
2. He L, Zhou Y, Liu L, Zhang Y, Ma J. Application of the YOLOv11-seg algorithm for AI-based landslide detection and recognition. Sci Rep. 2025;15(1):12421. doi:10.1038/s41598-025-95959-y.
3. Huang F, Yang Y, Jiang B, Chang Z, Zhou C, Jiang SH, et al. Effects of different division methods of landslide susceptibility levels on regional landslide susceptibility mapping. Bull Eng Geol Environ. 2025;84(6):276. doi:10.1007/s10064-025-04281-4.
4. Duan G, Su Y, Fu J. Landslide displacement prediction based on multivariate LSTM model. Int J Environ Res Public Health. 2023;20(2):1167. doi:10.3390/ijerph20021167.
5. Ma Z, Mei G. Forecasting landslide deformation by integrating domain knowledge into interpretable deep learning considering spatiotemporal correlations. J Rock Mech Geotech Eng. 2025;17(2):960–82. doi:10.1016/j.jrmge.2024.02.034.
6. Gómez D, García EF, Aristizábal E. Spatial and temporal landslide distributions using global and open landslide databases. Nat Hazards. 2023;117(1):25–55. doi:10.1007/s11069-023-05848-8.
7. Jia W, Wen T, Li D, Guo W, Quan Z, Wang Y, et al. Landslide displacement prediction of Shuping landslide combining PSO and LSSVM model. Water. 2023;15(4):612. doi:10.3390/w15040612.
8. Li M, Tian H. Insights from optimized non-landslide sampling and SHAP explainability for landslide susceptibility prediction. Appl Sci. 2025;15(3):1163. doi:10.3390/app15031163.

9. Casagli N, Intrieri E, Tofani V, Gigli G, Raspini F. Landslide detection, monitoring and prediction with remote-sensing techniques. Nat Rev Earth Environ. 2023;4(1):51–64. doi:10.1038/s43017-022-00373-x.
10. Zai D, Pang R, Xu B, Liu J. A novel data-driven hybrid intelligent prediction model for reservoir landslide displacement. Bull Eng Geol Environ. 2024;83(12):493. doi:10.1007/s10064-024-03987-1.
11. Chang Z, Huang F, Huang J, Jiang SH, Liu Y, Meena SR, et al. An updating of landslide susceptibility prediction from the perspective of space and time. Geosci Front. 2023;14(5):101619. doi:10.1016/j.gsf.2023.101619.
12. Liu F, Zhang T, Deng Y, Qian F, Yang N. Landslide susceptibility prediction based on landform predisposing indexes-an example from the Beiluo River Basin. Adv Space Res. 2024;74(11):5348–70. doi:10.1016/j.asr.2024.08.003.
13. Yang C, Liu LL, Huang F, Huang L, Wang XM. Machine learning-based landslide susceptibility assessment with optimized ratio of landslide to non-landslide samples. Gondwana Res. 2023;123(11):198–216. doi:10.1016/j.gr.2022.05.012.
14. Wang H, Zhang L, Yin K, Luo H, Li J. Landslide identification using machine learning. Geosci Front. 2021;12(1):351–64. doi:10.1016/j.gsf.2020.02.012.
15. Ehsan M, Anees MT, Bakar AFBA, Ahmed A. A review of geological and triggering factors influencing landslide susceptibility: artificial intelligence-based trends in mapping and prediction. Int J Environ Sci Technol. 2025;22(16):17347–82. doi:10.1007/s13762-025-06741-6.
16. Zhou C, Ye M, Xia Z, Wang W, Luo C, Muller JP. An interpretable attention-based deep learning method for landslide prediction based on multi-temporal InSAR time series: a case study of Xinpu landslide in the TGRA. Remote Sens Environ. 2025;318(10):114580. doi:10.1016/j.rse.2024.114580.
17. Cui HZ, Tong B, Wang T, Dou J, Ji J. A hybrid data-driven approach for rainfall-induced landslide susceptibility mapping: physically-based probabilistic model with convolutional neural network. J Rock Mech Geotech Eng. 2025;17(8):4933–51. doi:10.1016/j.jrmge.2024.08.005.
18. Li J, Zhang J, Fu Y. CTHNet: a CNN–transformer hybrid network for landslide identification in Loess Plateau regions using high-resolution remote sensing images. Sensors. 2025;25(1):273. doi:10.3390/s25010273.
19. Ma T, Wu L, Zhou J, Zhang H, Xiao H. An interpretable hybrid model for predicting step-like landslide displacement: a case study in the Three Gorges Reservoir. Nat Hazards. 2025;121(18):21441–58. doi:10.1007/s11069-025-07638-w.
20. Zheng D, Li Y, Yan C, Wu H, Yamashiki YA, Gao B, et al. Landslide susceptibility assessment using AutoML-SHAP method in the southern foothills of Changbai Mountain, China. Landslides. 2025;22(6):1855–75. doi:10.1007/s10346-025-02462-6.
21. Lv J, Zhang R, Shama A, Hong R, He X, Wu R, et al. Exploring the spatial patterns of landslide susceptibility assessment using interpretable Shapley method: mechanisms of landslide formation in the Sichuan-Tibet region. J Environ Manag. 2024;366:121921. doi:10.1016/j.jenvman.2024.121921.
22. Wang Z, Brenning A. Active-learning approaches for landslide mapping using support vector machines. Remote Sens. 2021;13(13):2588. doi:10.3390/rs13132588.
23. Kanwar M, Pokharel B, Lim S. A new random forest method for landslide susceptibility mapping using hyper-parameter optimization and grid search techniques. Int J Environ Sci Technol. 2025;22(11):10635–50. doi:10.1007/s13762-024-06310-3.
24. Sutou A, Wang J. Influence-balanced XGBoost: improving XGBoost for imbalanced data using influence functions. IEEE Access. 2024;12:193473–86. doi:10.1109/access.2024.3520159.
25. Ouyang S, Chen W, Liu H, Li Y, Xu Z. A novel landslide susceptibility prediction framework based on contrastive loss. Giscience Remote Sens. 2024;61(1):2306740. doi:10.1080/15481603.2024.2306740.
26. Khalil U, Imtiaz I, Aslam B, Ullah I, Tariq A, Qin S. Comparative analysis of machine learning and multi-criteria decision making techniques for landslide susceptibility mapping of Muzaffarabad district. Front Environ Sci. 2022;10:1028373. doi:10.3389/fenvs.2022.1028373.
27. Aslam B, Zafar A, Khalil U. Development of integrated deep learning and machine learning algorithm for the assessment of landslide hazard potential. Soft Comput. 2021;25(21):13493–512. doi:10.1007/s00500-021-06105-5.

28. Ge Q, Wang J, Liu C, Wang X, Deng Y, Li J. Integrating feature selection with machine learning for accurate reservoir landslide displacement prediction. Water. 2024;16(15):2152. doi:10.3390/w16152152.
29. Guo J, Ye W, Wang D, He Z, Yan Z, Sato M, et al. A novel snow leopard optimization for high-dimensional feature selection problems. Sensors. 2024;24(22):7161. doi:10.3390/s24227161.
30. Zhou G, Du J, Guo J, Li G. A novel hippo swarm optimization: for solving high-dimensional problems and engineering design problems. J Comput Des Eng. 2024;11(3):12–42. doi:10.1093/jcde/qwae035.
31. Guo J, Zhou G, Yan K, Di Y, Sato Y, He Z, et al. A deep backtracking bare-bones particle swarm optimisation algorithm for high-dimensional nonlinear functions. CAAI Trans Intel Tech. 2025;10(5):1501–20. doi:10.1049/cit2.70028.