applied sciences

MDPI

Article

Landslide Displacement Prediction during the Sliding Process Using XGBoost, SVR and RNNs

Jiancong Xu *, Yu Jiang and Chengbin Yang

Department of Geotechnical Engineering, Tongji University, Shanghai 200092, China; jiangyusure@163.com (Y.J.); 1932507@tongji.edu.cn (C.Y.)

* Correspondence: xjc1008@tongji.edu.cn

Abstract: In order to promptly evacuate personnel and property near the foot of the landslide and take emergency treatment measures in case of sudden danger, it is very necessary to select suitable forecasting methods for conduct short-term displacement predictions in the slope-sliding process. In this paper, we used Python to develop the landslide displacement-prediction method based on the eXtreme Gradient Boosting (XGBoost) algorithm, and optimized the hyperparameters through a genetic algorithm to solve the problem of insufficient short-term displacement-prediction accuracy for landslides. We compared the deviation, relative error (RE) and median of RE of predicted values obtained using XGBoost, SVR and RNNs, and the actual value of landslide displacement. The results show that the accuracies of slope displacement prediction using XGBoost and SVR are very high, and that using RNNs is very low during the sliding process. For large displacement values and small numbers of samples, the displacement-prediction effect of XGBoost algorithm is better than that of SVR and RNNs in the sliding process of landslide. There are generally only fewer data samples collected during the landslide sliding process, so RNNs is not suitable for displacement prediction in this scenario. If the number of data samples is large enough, using RNNs to predict the long-term displacement of the slope may also have a much higher accuracy.

Keywords: landslide; displacement prediction; XGBoost; SVR; RNNs

Citation: Xu, J.; Jiang, Y.; Yang, C. Landslide Displacement Prediction during the Sliding Process Using XGBoost, SVR and RNNs. *Appl. Sci.* **2022**, *12*, 6056. https://doi.org/10.3390/app12126056

Academic Editor: Alessandro Simoni

Received: 10 May 2022
Accepted: 13 June 2022
Published: 14 June 2022

1. Introduction

Accurately predicting the displacement of a landslide under critical sliding state and during the sliding process can mitigate its negative influence on both human lives and economic loss. It is also a major issue to be solved firstly to control economically and scientifically the disaster of residual soil slopes. However, it is still a difficult and important task.

Many scholars have carried out a lot of research on the prediction of landslide disasters and obtained many achievements [1–8]. The load–unload response ratio method was used to predict the failure time of landslides [1,2]. The temporal and spatial prediction of a shallow landslide induced by rainfall was carried out by using a combination of probabilistic and deterministic methods [3]. The occurrence times and displacement trends of landslides were predicted by using the cusp catastrophe theory, the polynomial model, the displacement critical-rate method and inverse velocity analysis [4–9]. Observed changes in rainfall amounts were used to predict changes in landslide displacement rates from [10], which made future landslide movements relate to expected rainfall scenarios [11]. Modern numerical methods are able to simulate large mass movements, and there is an opportunity to utilize such methods to evaluate the risks of catastrophic damage if a landslide occurs [12,13].

The coupling effect of periodic fluctuation of water level and rainfall, the displacement and monitoring-time curve of a landslide appears "step-like" [14–18]. Therefore, the cumulative displacement of a landslide should be decomposed into multiple displacement components with different scale characteristics of time series. Then, a polynomial regression model [19] and machine learning (backpropagation neural network [19], support-vector

regression machine [20], long short-term memory network [21,22] and kernel extreme learning machine [23]) are used to train and predict each displacement component, respectively. Finally, by superimposing the prediction results of each displacement component, the cumulative displacement prediction of the landslide is obtained. Coupled with the time series of landslide displacement, these methods can make clear the physical meaning of each part of the displacement and reflect the relationship between each displacement component and the triggering factors to improve the accuracy of displacement prediction [18].

With the improvement of computer calculating ability and the development of machine learning and artificial intelligence, the disaster-prediction ability of landslides has been greatly improved. At present, the commonly used models for landslide displacement prediction include the backpropagation neural network (BPNN) [14,15,23], support-vector regression (SVR) [18,24–28], extreme learning machine (ELM) [29–32], kernel extreme learning machine (KELM) [14,23,33], long short-term memory (LSTM) [25,34–37], decision tree [38], and so on. Moreover, many algorithms are used to optimize the parameters of the prediction models, including the genetic algorithm (GA) [28,39,40], particle swarm optimization (PSO) [16,26,28,29,41,42], fruit fly optimization algorithm (FOA) [18], grey wolf optimizer (GWO) [15], and so on. However, each algorithm has its limitations. Therefore, it is vital to select the appropriate machine-learning models for the accuracy of displacement prediction of landslides.

The disaster of granite residual landslides involves infiltration characteristics, the intensity and pattern of rainfall, the transport mode of gas and liquid in soil, the temperature of the earth surface, the development mode of soil creep and their coupling interaction. This type of landslide is often induced by heavy rainfall and experiences multiple processes of sliding, relative stability, re-sliding, then relative stability and re-sliding, which pose a great threat to buildings and structures at the foot of the slope. Moreover, it is difficult to judge the displacement trend of each new sliding by the displacement data collected in the past. Therefore, it is very important to select a method that can predict the slope displacement accurately during the sliding process.

In the paper, we used Python to develop the landslide displacement-prediction method based on the eXtreme Gradient Boosting (XGBoost) algorithm, and optimized the hyperparameters through a genetic algorithm to improve the prediction accuracy of landslide displacement. The proposed XGBoost algorithm may effectively solve the problem of insufficient prediction accuracy of small samples and meet the actual needs for short-term displacement prediction of landslides during the sliding process.

2. eXtreme Gradient Boosting (XGBoost)

XGBoost uses Newton's method to solve the extreme value of loss function, and adds a regularization term to the loss function, which is an improvement on the gradient boosting algorithm, developed by Chen and Guestrin [43]. The loss function of XGBoost is denoted by the second-order Taylor expansion. Thus, it can better capture nonlinear information. Moreover, XGBoost may reduce the overfitting effect caused by the gradient-boosting algorithm by introducing regular terms.

2.1. Methodology

Given a training sample data set $(x_1, y_1), (x_2, y_2), \cdots, (x_M, y_M)$, $x_i \in \mathbb{X} \in \mathbb{R}^m$, $\mathbb{X}$ is sample input space, $y_i \in \mathbb{Y} \in \mathbb{R}$, $\mathbb{Y}$ is output space, m is input-space dimension. If $\mathbb{X}$ is divided into nonintersecting regions $\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_J$, and c_j is the constant output determined in each region. The tree may then be expressed by Formula (1):

$$\mathcal{T}(x; \mathcal{Q}_k) = \sum_{j=1}^{J} c_j I(x \in \mathcal{R}_j) \tag{1}$$

where $\mathcal{Q} = \{(\mathcal{R}_1, c_1), (\mathcal{R}_2, c_2), \cdots, (\mathcal{R}_J, c_J)\}$ represents tree region division and the constants in each region; J is the number of leaf nodes.

The boosting-tree model of XGBoost may be denoted by Formula (2) as an additive model of decision trees:

$$f_D(x) = \sum_{d=1}^{D} \mathcal{T}(x; \mathcal{Q}_d) \tag{2}$$

where $\mathcal{T}(x; \mathcal{Q}_k)$ is decision tree; $\mathcal{Q}_d$ is decision-tree parameters; D is the number of trees.

The forward step-by-step boosting tree algorithm in XGBoost is used as follows.

(1) The initial boosting tree is determined as $f_0(x) = 0$.

(2) The model at the d-th step is expressed by Formula (3):

$$f_d(x) = f_{d-1}(x) + \mathcal{T}(x; \mathcal{Q}_d),\ k = 1, 2, \cdots, K \tag{3}$$

where $f_{d-1}(x)$ is the current model; $\mathcal{Q}_d$ is decision-tree parameters determined by Formula (4):

$$\hat{\mathcal{Q}}_d = \underset{\mathcal{Q}_d}{\operatorname{argmin}} \sum_{i=1}^{D} \mathcal{L}(y_i, f_{d-1}(x_i) + \mathcal{T}(x_i; \mathcal{Q}_d)) \tag{4}$$

where $\hat{\mathcal{Q}}_d$ is the d-th tree parameters; $\mathcal{L}(y_i, f_{d-1}(x_i) + \mathcal{T}(x_i; \mathcal{Q}_d))$ is the loss value between observed value y and predicted value $f_d(x)$ for the i-th sample point.

The loss function can be denoted as $\mathcal{L}(y, f(x)) = (y - f(x))^2$ by the least-square-error criterion.

$$l(y, y') = \mathcal{L}(y, f_{d-1}(x) + \mathcal{T}(x; \mathcal{Q}_d)) = [y - f_{d-1}(x) - \mathcal{T}(x; \mathcal{Q}_d)]^2 = [r - \mathcal{T}(x; \mathcal{Q}_d)]^2 \tag{5}$$

where $r = y - f_{d-1}(x)$ is fitting residual of the current model; $f(x)$ is the prediction value obtained from the fitted model, $y' = f(x)$; $f_{d-1}(x)$ is the fitted model at the d-1-th step.

(3) The fitting residual is calculated by Formula (6).

$$r_{d_i} = y_i - f_{d-1}(x_i),\ i = 1, 2, \cdots, D \tag{6}$$

A regression tree is learned according to the fitting residual r_{d_i}, and $\mathcal{T}(x; \mathcal{Q}_d)$ is obtained.

(4) $f_d(x) = f_{d-1}(x) + \mathcal{T}(x; \mathcal{Q}_d)$ is updated.

(5) The boosting tree of regression problem $f_D(x) = \sum_{d=1}^{D} \mathcal{T}(x; \mathcal{Q}_d)$ is solved.

The loss function of XGBoost $\mathcal{L}(\phi)$ is denoted by Formula (7).

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, y_i') + \sum_{d} \mathcal{W}(f_d) = l(y, y') + \sum_{d} \mathcal{W}(f_d) = [r - \mathcal{T}(x; \mathcal{Q}_d)]^2 + \sum_{d} \mathcal{W}(f_d) \tag{7}$$

where n is the number of training samples; $\sum_{i=1}^{n} l(y_i, y_i')$ is loss value caused using the gradient-boosting algorithm, calculated by Formula (5), assuming it is a differentiable convex function; $\mathcal{W}(f_d)$ is regular terms; f_d is weak learner function.

The complexity of XGBoost model is denoted by Formula (8):

$$\mathcal{W}(f_k) = \gamma \mathcal{T}_{\ln} + \frac{1}{2} \lambda \|\boldsymbol{\omega}\|^2 = \gamma \mathcal{T}_{\ln} + \frac{\lambda}{2} \sum_{j=1}^{J} \omega_{tj}^2 \tag{8}$$

where γ and λ are two coefficients set artificially; $\boldsymbol{\omega}$ is the vector formed by the values of all leaf nodes of the decision tree; $\mathcal{T}_{\ln}$ is the number of leaf nodes; ω_{tj} is the weight of leaf j.

The final goal of XGBoost is to minimize $\mathcal{L}(\phi)$ in Formula (7). $\mathcal{L}(\phi)$ is approximately solved by the Newton's method, and can be expressed by the second-order Taylor expansion at the point $y'_{i,t-1}$ and Formula (9):

$$\begin{aligned}\mathcal{L}_t(\phi) \approx \sum_{i=1}^{n}\left[l\left(y_i, y'_{i,t-1}\right) + \frac{\partial \mathcal{L}(y_i,y'_i)}{\partial y'_i}\Big|_{y'_i=y'_{i,t-1}} f_t(x_i) + \frac{1}{2}\frac{\partial^2 \mathcal{L}(y_i,y'_i)}{\partial y'^2_i}\Big|_{y'_i=y'_{i,t-1}} f^2_{t-1}(x_i)\right] \\ +\gamma\mathcal{T}_{\ln} + \frac{\lambda}{2}\sum_{j=1}^{\mathcal{T}}\omega^2_{tj}\end{aligned} \tag{9}$$

where $f_t(\boldsymbol{x}_i)$ is the current weak learner; x_i is the i-th training sample.

2.2. XGBoost-Based Relationship between Displacement and Time

We used XGBoost to map the nonlinear relationship between time $\boldsymbol{T}$ and monitoring displacement $\boldsymbol{Y}$. The mathematical model of XGBoost, $XGBoost(\boldsymbol{T})$, is defined as:

$$\begin{cases} XGBoost(\boldsymbol{T}): \mathbb{R}^N \rightarrow \mathbb{R}^M \\ \boldsymbol{Y} = XGBoost(\boldsymbol{T}) \end{cases} \tag{10}$$

where $\boldsymbol{T} = (t_1, t_2, \cdots, t_N)$, t_i $(i = 1, 2, \cdots, N)$ is a vector of time; N is the number of monitoring time; $\boldsymbol{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_M)$ is the M dimensional vector of monitoring data, such as displacement.

In this paper, the genetic algorithm is used to tune its hyperparameters of the XGBoost model to improve the prediction accuracy of landslide displacement.

3. Recurrent Neural Network Prediction Algorithm

In addition to the interlayer connections, the RNNs allow for the addition of the intralayer connection that allows the RNNs to accumulate the information in the time domain. The RNNs allow the information in the previous steps to continue to affect subsequent steps, so the RNNs can effectively handle sequence-data problems and become more suitable for processing sequence-related machine-learning tasks.

The recurrent neural networks (RNNs) generally consist of an input layer, a hidden layer, and an output layer, used to model sequential data. In Figure 1, W is the weight matrix between the input of hidden layer at the current moment and the output of the hidden layer at the previous moment; U is the weight matrix between the input layer and hidden layer; V is the weight matrix between the hidden layer and output layer; x_t is the input at time t; O_t is the output at time t; S_t is the hidden state at time t.

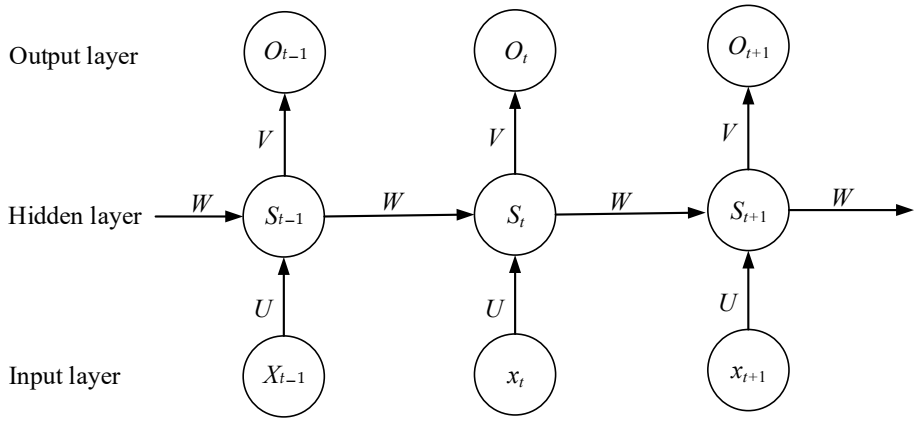

Figure 1. Unfolding action diagram of a RNN model.

3.1. BPTT Training Algorithm

The parameter training of the RNNs can be performed by the backpropagation-through-time (BPTT) algorithm proposed by Werbos [44]. In BPTT, the network is first trained with the training data, and the output error gradient is saved for each time step. Then, the network is unfolded in time, as shown in Figure 1, originating an equivalent feedforward network called an encoding network.

During the error propagation, the gradient of the sum of squared errors $err(t)$ of all the copies of the RNNs are stored and summed up to form one explicit error term, used to compute the appropriate ΔW and ΔV of the networks at the end of each h steps. The training cycle of the RNNs consists of forward propagation and error backpropagation [41].

The local gradient $\delta(l)$ is denoted by Formula (11):

$$\delta(l) = -\frac{\partial(err(l))}{\partial x(l-1)} \tag{11}$$

where $t - h < l < t$, and t denotes the time required to learn a temporal task starting from time t_0 all the way up to time t.

The actual updating of the weights is denoted by using Formula (12):

$$\begin{cases} W(h) = W(h-1) + \Delta W(h) \\ V(h) = V(h-1) + \Delta V(h) \end{cases} \tag{12}$$

where ΔW is the variations of input weights; ΔV is the variations of output weights.

3.2. Loss Function of RNNs

In this paper, a cross-entropy function $\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is selected as loss function, denoted by Formula (13):

$$\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{i=1}^{K} p(y_i) \log p(\hat{y}_i) \tag{13}$$

where $p(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{K} e^{y_j}}$; $p(\hat{y}_i) = \frac{e^{\hat{y}_i}}{\sum_{j=1}^{K} e^{\hat{y}_j}}$ is the output value of the last fully connected layer obtained by the *softmax* function; K is the dimension of the output value of the RNNs, K= 5 in this paper.

$\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is a function that measures the difference between the calculated value and the true value. Training the RNNs is the process of minimizing $\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}})$.

3.3. RNN-Based Relationship between Displacement and Time

We used *RNNs* to map the nonlinear relationship between time $\boldsymbol{T}$ and monitoring displacement $\boldsymbol{Y}$. The mathematical model of *RNNs*, $RNNs(\boldsymbol{T})$, is defined as:

$$\begin{cases} RNNs(\boldsymbol{T}): \mathbb{R}^N \rightarrow \mathbb{R}^M \\ \boldsymbol{Y} = RNNs(\boldsymbol{T}) \end{cases} \tag{14}$$

where $\boldsymbol{T} = (t_1, t_2, \cdots, t_N)$, t_i $(i = 1, 2, \cdots, N)$ is a vector of time; N is the number of monitoring time; $\boldsymbol{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_M)$ is the M dimensional vector of monitoring data, such as displacement.

4. Support-Vector Machine Regression Algorithm (SVR)

SVM is a machine-learning method used in the fields of classification, forecasting, and pattern identification [45]. The purpose of SVM is to find a plane that can divide training sets into two sets with the largest interval.

For the sample set D, SVR hopes to obtain a fitting relation such as Formula (15) so that the error between $f(\boldsymbol{x})$ and y satisfies a certain condition, where $f(\boldsymbol{x})$ is predicted value and y is sample value.

$$\min_{\omega,b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{m} l_\varepsilon(f(\boldsymbol{x}_i) - y_i) \tag{15}$$

where C is the penalty coefficient, termed as regularization constant; l_ε is insensitive loss function.

SVR gives certain tolerance ε for the error between $f(\boldsymbol{x})$ and y, as long as the error does not exceed ε; that is, the distance from sample point to $f(\boldsymbol{x}) = \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x} + b$ is less than ε, the regression model denoted by Formula (15) is considered correct.

The final form of the support-vector regression machine is denoted by Formula (16):

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i')\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) + b \tag{16}$$

where $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \varphi(\boldsymbol{x}_i)^{\mathrm{T}}\varphi(\boldsymbol{x}_j)$ represents a kernel function for a support=-vector machine. In the paper, $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{2\sigma^2}\right)$, and σ is the bandwidth of the Gaussian kernel, $\sigma > 0$. α_i, α_i' are obtained by Formula (17).

$$\begin{cases} \alpha_i(f(\boldsymbol{x}_i) - y_i - \varepsilon - \xi_i) = 0 \\ \alpha_i'(y_i - f(\boldsymbol{x}_i) - \varepsilon - \xi_i') = 0 \\ \alpha_i\alpha_i' = 0,\ \xi_i\xi_i' = 0 \\ (C - \alpha_i)\xi_i = 0,\ (C - \alpha_i')\xi_i' = 0 \end{cases} \tag{17}$$

where ξ_i and ξ_i' are slack-variable.

5. Description of the Investigated Slope

The Liangyeshan landslide is located in Wuping county, Fujian province, China. It is a typical drag medium-sized landslide. The landslide is composed of silty clay, residual sandy clay, fully weathered granite, and fragmented strong weathered granite (Figure 2).

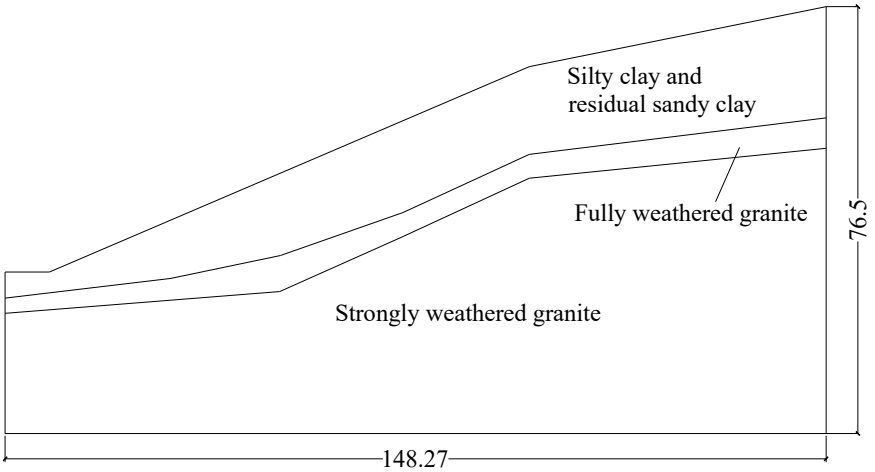

Figure 2. Engineering geological profile.

The natural grade of the slope varies greatly, which is about 5~25°, and the vegetation on the slope is developed. The sliding body is in the shape of a chair and is located on the right side of the main line of the expressway. The width of the landslide is about 60 m. The elevation of the trailing edge of the slope is 352.85 m, the elevation difference is about 36.6 m, and the azimuth of the main slide direction is about 337°. There are the typical characteristics of landslide accumulation, with thin leading and trailing edges, and thick

middle. The initial sliding surface can be defined as the interface between residual sandy clay and fully weathered granite.

There are buildings of the highway service area near the foot of Liangyeshan landslide (Figure 3). In order to evacuate personnel and property in time and take emergency treatment measures in case of sudden danger, it is necessary to predict the short-term displacement during the slope-sliding process.

Figure 3. Study area background.

Before the slope slides, we arranged ten displacement-monitoring points, D1–D10, in the direction of the main sliding line, as shown in Figure 4. Slope displacement is automatically collected by the displacement-monitoring system installed on the slope, and then transmitted remotely to the database of the monitoring platform through the Internet of Things (Figure 5). We also buried soil-moisture sensors in the slope, so as to monitor the moisture content of soil.

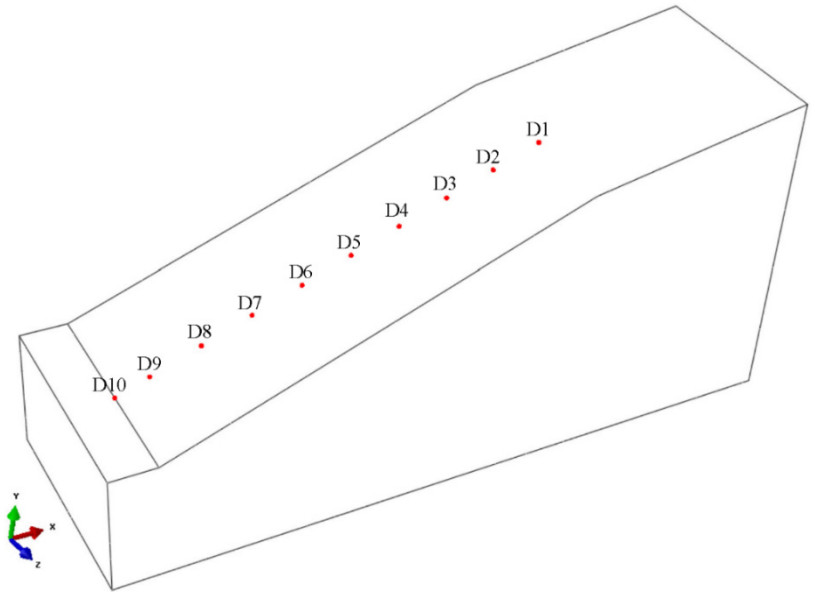

Figure 4. Monitoring-point layout.

Figure 5. Automatic data collection and monitoring system.

We started monitoring the slope on 1 June 2013. We recorded monitoring data of precipitation, soil moisture content and slope displacement during rainfall and after the rain through the Internet of Things. The slope had been monitored for two years. After it rained for six days from 2 June, the slope soil reached a saturated state, and the slope started to slide on 8 June. In this paper, we selected 151 sets of slope displacement and time collected for 20 h during the sliding process as the training dataset.

For the purpose of mitigating the disaster caused by the landslide, we used XGBoost, RNNs and SVR to predict the displacement corresponding to the predetermined sliding time of the slope after the slope reached the saturation state. Furthermore, we investigated the variation law of slope displacement with time during the slope-sliding process. We collected 151 sets of data of the resultant displacement of the slope after the soil was saturated to carry out the displacement-prediction research by using XGBoost, RNNs and SVR.

6. Results

6.1. Deviation between Predicted Value and Actual Value

For the monitoring points D5 and D7, the deviation between the actual values and the predicted values obtained using SVR is the smallest among the ones using three landslide displacement-prediction methods, and the one using RNNs is the largest (see Figures 6 and 7).

For the monitoring point D6, the deviation between the actual values and the predicted values obtained using XGBoost is the smallest among the ones using three landslide displacement-prediction methods (see Figure 8), and the prediction values using the other two prediction methods deviate greatly from actual values. From Figure 8, we note that as the curvature of the curve changes from large to small, the deviation of predicted values obtained using RNNs from real values becomes larger and larger.

For the monitoring points D8, D9 and D10, the deviation between the actual values and the predicted values obtained using SVR and XGBoost is smaller than that using RNNs, and the predicted values using RNNs are the largest among that using these three landslide displacement-prediction methods (see Figures 9–11). However, the prediction values using these three prediction methods are not very different from actual values when the curvature of the curve is very small (see Figures 9–11).

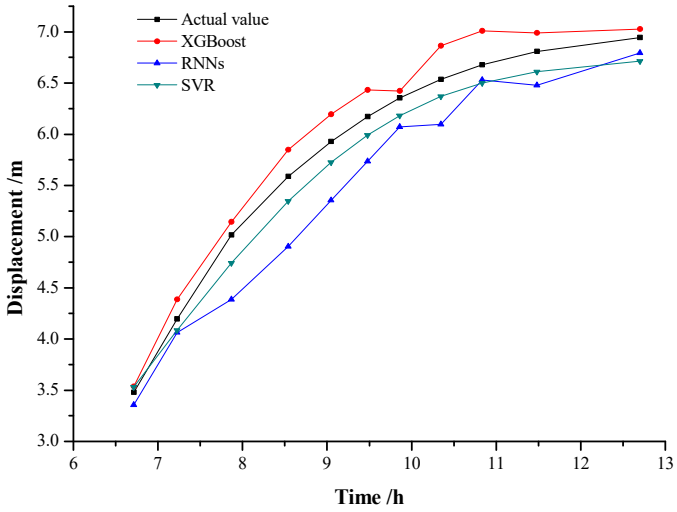

Figure 6. Variation of displacement predicted values of D5 with time.

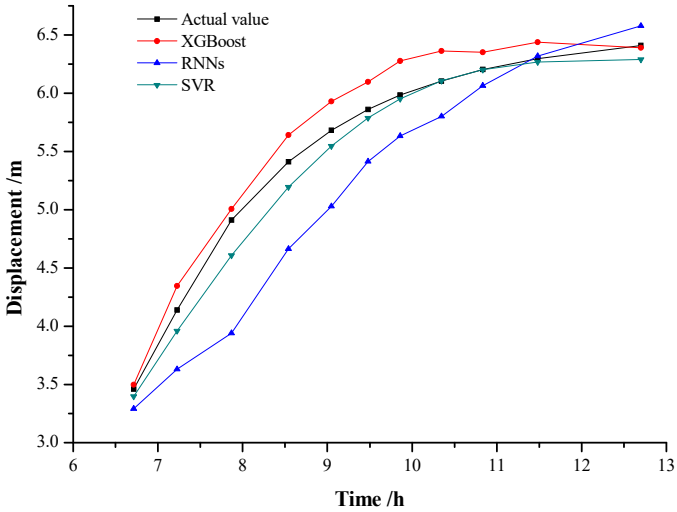

Figure 7. Variation of displacement predicted values of D7 with time.

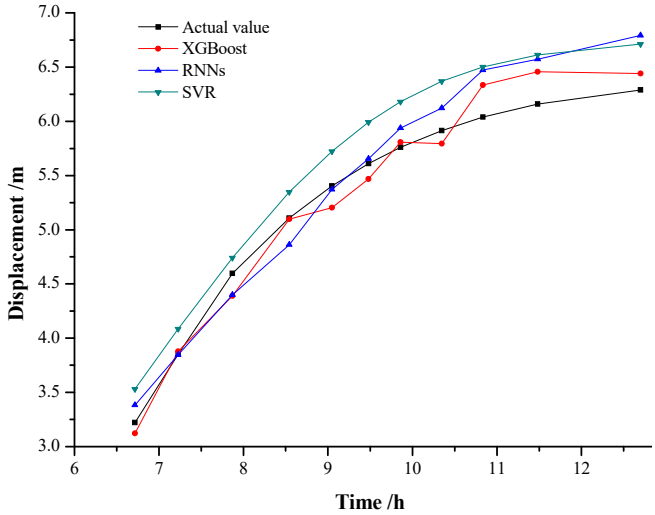

Figure 8. Variation of displacement predicted values of D6 with time.

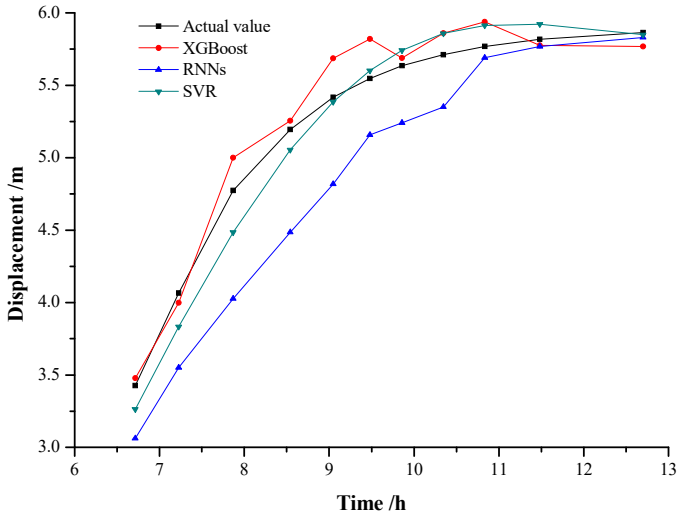

Figure 9. Variation of displacement predicted values of D8 with time.

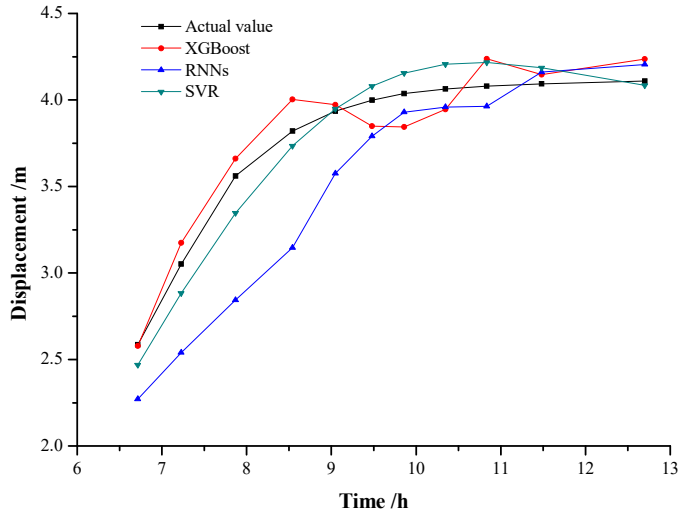

Figure 10. Variation of displacement predicted values of D9 with time.

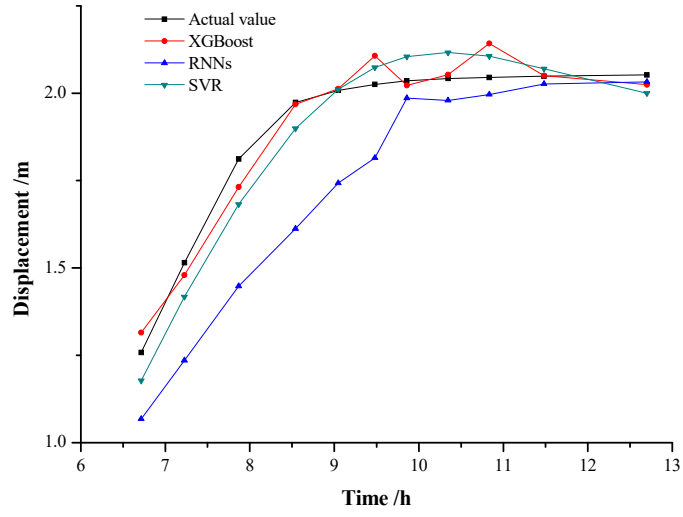

Figure 11. Variation of displacement predicted values of D10 with time.

6.2. Relative Error (RE) between Predicted Value and Actual Value

The maximum RE values using XGBoost, SVR and XGBoost are 5.0%, 9.55% and 20.14% (see Table 1), respectively. From Figures 12–17, we note that the REs of predicted values obtained using the XGBoost method are less than 5.0% for all monitoring points. From Figures 12–16, we note that the REs using the SVR method also are mostly less than 5.0%. The REs using the RNNs method are mostly more than 6.0% for all monitoring points, vary greatly, and their data are very discrete.

Table 1. Median of relative error and upper limit (%).

Monitoring Point Number	Median			Upper Limit		
	XGBoost	RNNs	SVR	XGBoost	RNNs	SVR
D5	4.21	4.86	2.89	5.0	12.53	5.45
D6	2.52	4.33	6.79	4.89	8.0	9.55
D7	4.04	5.88	1.83	4.93	19.76	6.16
D8	1.66	7.01	2.52	4.96	15.63	6.05
D9	3.08	5.2	2.93	4.79	20.14	6.0
D10	1.37	10.37	3.41	4.76	20.12	7.15

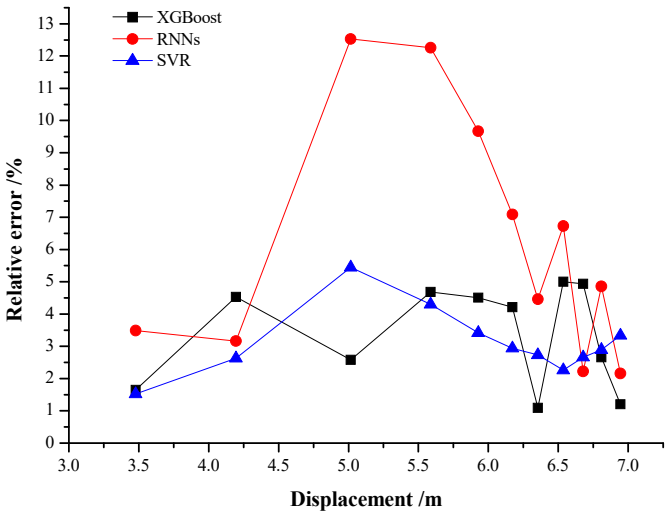

Figure 12. Variation of RE with slope displacement of D5.

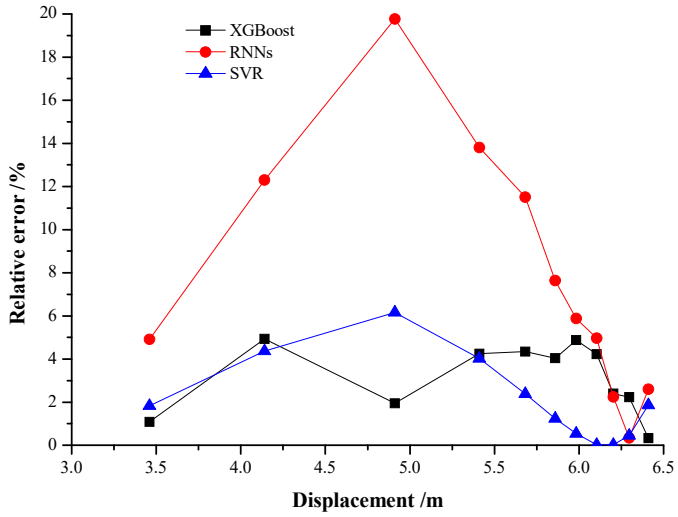

Figure 13. Variation of RE with slope displacement of D7.

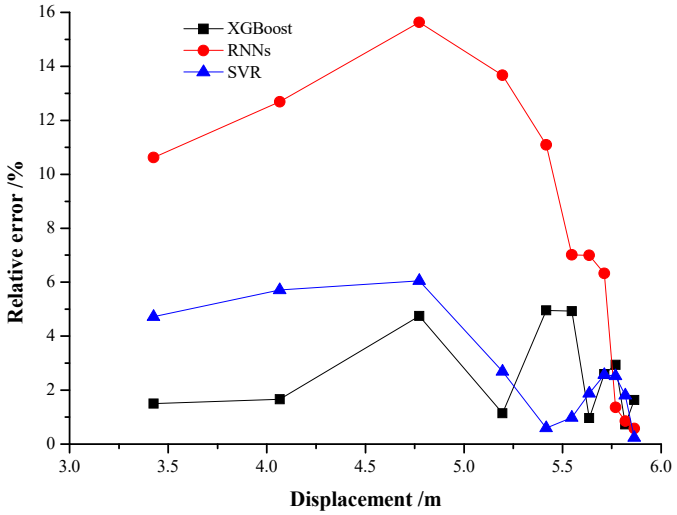

Figure 14. Variation of RE with slope displacement of D8.

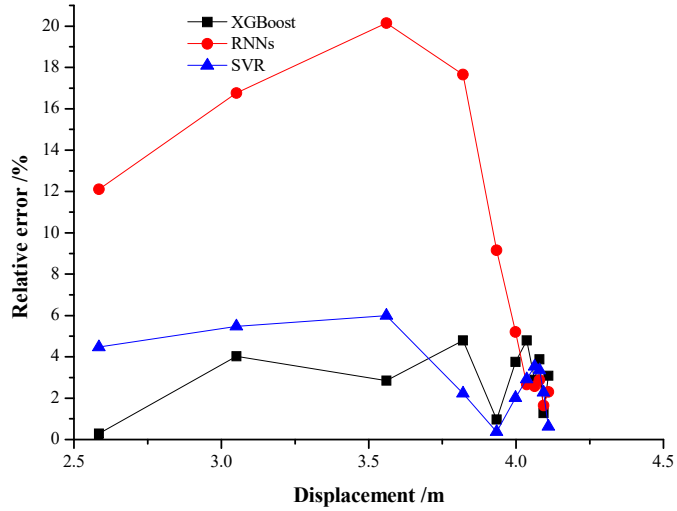

Figure 15. Variation of RE with slope displacement of D9.

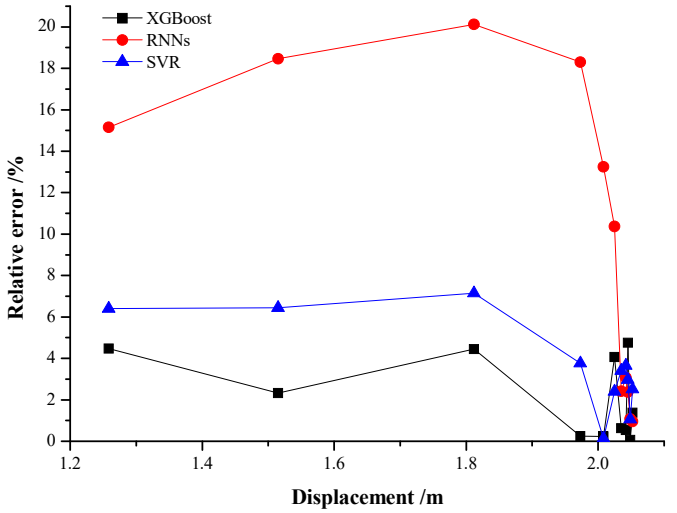

Figure 16. Variation of RE with slope displacement of D10.

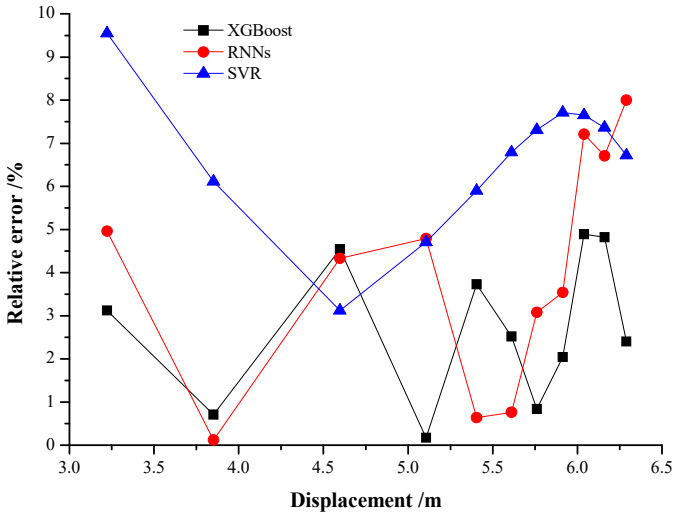

Figure 17. Variation of RE with slope displacement of D6.

From Figures 12–16 and 18, we note that after the curvature variation of the curve of displacement with time is stabilized, the RE values using XGBoost, SVR and RNNs are all less than 5.0%. However, from Figures 17 and 18, we also note that the RE values using SVR and RNNs fluctuate greatly with the displacement when the curvature variation of the curve of displacement changes greatly with time.

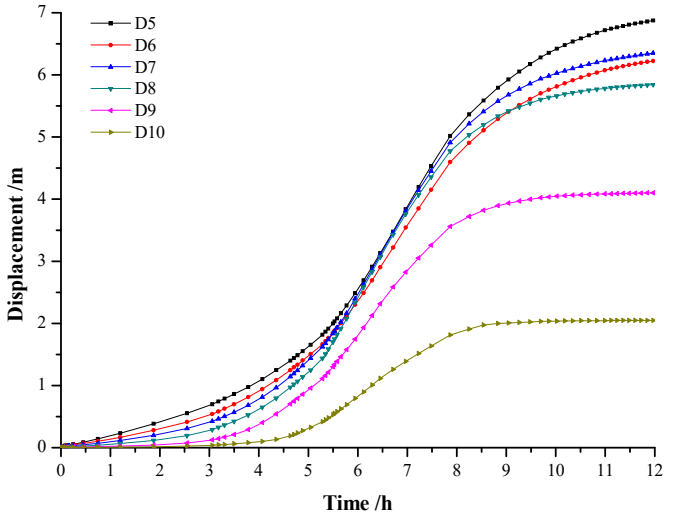

Figure 18. Variation of slope displacement with time.

7. Discussion

The deviation between the actual values and the predicted values obtained using SVR and XGBoost is much smaller than that using RNNs. However, the prediction values using these three prediction methods are not very different from actual values when the curvature of the curve is very small.

The RE median of the predicted values obtained using XGBoost is very small and less than 5.0% for every monitoring point. The REs of predicted values obtained using the XGBoost method are less than 5.0% for all monitoring points, and the REs using the SVR method also are less than 5.0%, except when the curvature variation of the curve of displacement with time changes greatly. Among using XGBoost, SVR and RNNs, the RE variation of RNNs is the biggest, and the RE variation of XGBoost is mostly the smallest. It may be related to the strong ability of XGBoost to better capture nonlinear information.

For large displacement values and small numbers of samples, the displacement-prediction effect of the XGBoost algorithm is better than that of the other two prediction methods in the sliding process of landslides. It is consistent with existing research results [46,47].

Both XGBoost and SVR can predict the displacement of landslides accurately in the sliding process, and RNNs is not suitable in this condition. In further research work, we will investigate the comparison of the predictive abilities of XGBoost and SVR with other deep-learning algorithms.

If these displacement-prediction methods are used for other landslides, the accuracy of the automatic collection of displacement data should be guaranteed. XGBoost, SVR and RNNs should be selected in order, according to the number of data samples from small to large and the nonlinear degree from large to small. In order to improve the early warning level in the critical slip state, it is recommended to evaluate the reliability of the prediction scheme by combining the three-dimensional numerical simulation test of nonstationary rheological parameters and the multivariate Bayesian update inverse analysis theory.

Author Contributions: J.X.: conceptualization and algorithm development; Y.J.: data curation and processing, investigation; C.Y.: software; writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China-Yalong River Joint Fund, grant number U1765110, National Natural Science Foundation of China, grant number 40872179.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within this article.

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

References

1. He, K.; Wang, R.; Li, X.; Yu, G.; Sun, L.; Wang, S. Load-unload dynamic law of groundwater level and dynamic displacement prediction of debris landslide—A case study of Bazimen landslide in Three Gorges Reservoir. *Chin. J. Rock Mech. Eng.* **2018**, *27*, 1644–1651.
2. He, K.; Yang, Q.; Chen, W.; Zhao, M. A new dynamic prediction method and its application in prediction of the colluvial landslides induced by rainfall. *Appl. Mech. Mater.* **2012**, *170–173*, 3–7. [CrossRef]
3. Grelle, G.; Soriano, M.; Revellino, P.; Guerriero, L.; Anderson, M.G.; Diambra, A.; Fiorillo, F.; Esposito, L.; Diodato, N.; Guadagno, F.M. Space-time prediction of rainfall-induced shallow landslides through a combined probabilistic/deterministic approach, optimized for initial water table conditions. *Bull. Eng. Geol. Environ.* **2014**, *73*, 877–890. [CrossRef]
4. Xu, J.; Shang, Y.; Zheng, S.; Zhang, T. Cusp-catastrophic model of shallow landslide under intensive rainfall. *J. Zhejiang Univ. (Eng. Ed.)* **2005**, *39*, 1675–1679.
5. Xu, D.; Chen, C.; Liu, X.; Shi, Y. On the prediction of landslide of rock slopes. *Chin. J. Rock Mech. Eng.* **1999**, *18*, 369–372.
6. Nie, L.; Li, Z.; Lv, Y.; Wang, H. A new prediction model for rock slope failure time: A case study in west open-pit mine, Fushun, China. *Bull. Eng. Geol. Environ.* **2017**, *76*, 975–988. [CrossRef]
7. Sunwoo, C.; Jung, Y.B.; Park, H.D.; Choi, Y. Failure prediction for weak rock slopes in a large open-pit mine by GPS measurements and assessment of landslide susceptibility. *J. Eng. Geol.* **2010**, *20*, 243–255.
8. Manconi, A.; Giordan, D. Landslide early warning based on failure forecast models: The example of the Mt. de La Saxe rockslide, northern Italy. *Nat. Hazard Earth Sys.* **2015**, *15*, 1639–1644. [CrossRef]
9. Segalini, A.; Valletta, A.; Carri, A. Landslide time-of-failure forecast and alert threshold assessment: A generalized criterion. *Eng. Geol.* **2018**, *245*, 72–80. [CrossRef]
10. Bernardie, S.; Desramaut, N.; Malet, J.P.; Gourlay, M.; Grandjean, G. Prediction of changes in landslide rates induced by rainfall. *Landslides* **2015**, *12*, 481–494. [CrossRef]
11. Troncone, A.; Pugliese, L.; Lamanna, G.; Conte, E. Prediction of rainfall-induced landslide movements in the presence of stabilizing piles. *Eng. Geol.* **2021**, *288*, 106143. [CrossRef]
12. Soga, K.; Alonso, E.; Yerro, A.; Kumar, K.; Bandara, S. Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method. *Geotechnique* **2016**, *66*, 248–273. [CrossRef]
13. Troncone, A.; Pugliese, L.; Conte, E. Analysis of an excavation-induced landslide in stiff clay using the material point method. *Eng. Geol.* **2022**, *296*, 106479. [CrossRef]

14. Li, L.; Wu, Y.; Miao, F.; Zhang, L.; Xue, Y.; Zhang, L.; Liao, K.; Teng, W.; Shi, H. Displacement interval prediction method for step-like landslides considering deformation state dynamic switching. *Chin. J. Rock Mech. Eng.* **2019**, *38*, 2272–2287.
15. Guo, Z.Z.; Chen, L.X.; Gui, L.; Du, J.; Yin, K.L.; Do, H.M. Landslide displacement prediction based on variational mode decomposition and WA-GWO-BP model. *Landslides* **2020**, *17*, 567–583. [CrossRef]
16. Du, H.; Song, D.Q.; Chen, Z.; Shu, H.P.; Guo, Z.Z. Prediction model oriented for landslide displacement with step-like curve by applying ensemble empirical mode decomposition and the PSO-ELM method. *J. Clean. Prod.* **2020**, *270*, 122248. [CrossRef]
17. Liu, Q.; Lu, G.Y.; Dong, J. Prediction of landslide displacement with step-like curve using variational mode decomposition and periodic neural network. *Bull. Eng. Geol. Environ.* **2021**, *80*, 3783–3799. [CrossRef]
18. Lu, X.S.; Miao, F.S.; Xie, X.X.; Li, D.Y.; Xie, Y.H. A new method for displacement prediction of "step-like" landslides based on VMD-FOA-SVR model. *Environ. Earth Sci.* **2021**, *80*, 542. [CrossRef]
19. Tang, L. *Study on Stability and Prediction of Accumulation Landslide in the Three Gorges Reservoir Area*; China University of Geosciences: Wuhan, China, 2013.
20. Zhou, C.; Yin, K.; Cao, Y.; Ahmed, B. Application of time series analysis and PSO-SVM model in predicting the Bazimen Landslide in the Three Gorges Reservoir, China. *Eng. Geol.* **2016**, *204*, 108–120. [CrossRef]
21. Yang, B.; Yin, K.; Du, J. A model for predicting landslide displacement based on time series and long and short term memory neural network. *Chin. J. Rock Mech. Eng.* **2018**, *37*, 2334–2343.
22. Xie, P.H.; Zhou, A.G.; ChaI, B. The application of long short-term memory(LSTM) method on displacement prediction of multifactor-induced landslides. *IEEE Access* **2019**, *7*, 54305–54311. [CrossRef]
23. Li, L.; Wu, Y.; Miao, F.; Zhang, L.; Xue, Y. Landslide displacement interval prediction based on different Bootstrap methods and KELM-BPNN model. *Chin. J. Rock Mech. Eng.* **2019**, *38*, 912–926.
24. Zhang, J.R.; Tang, H.M.; Wen, T.; Ma, J.W.; Tan, Q.W.; Xia, D.; Liu, X.; Zhang, Y.Q. A Hybrid Landslide Displacement Prediction Method Based on CEEMD and DTW-ACO-SVR-Cases Studied in the Three Gorges Reservoir area. *Sensors* **2020**, *20*, 4287. [CrossRef] [PubMed]
25. Jiang, H.W.; Li, Y.Y.; Zhou, C.; Hong, H.Y.; Glade, T.; Yin, K.L. Landslide Displacement Prediction Combining LSTM and SVR Algorithms: A Case Study of Shengjibao Landslide from the Three Gorges Reservoir area. *Appl. Sci.* **2020**, *10*, 7830. [CrossRef]
26. Han, H.M.; Shi, B.; Zhang, L. Prediction of landslide sharp increase displacement by SVM with considering hysteresis of groundwater change. *Eng. Geol.* **2021**, *280*, 105876. [CrossRef]
27. Li, X.Z.; Li, S.W. Large-scale landslide displacement rate prediction based on multi-factor support vector regression machine. *Appl. Sci.* **2021**, *11*, 1381. [CrossRef]
28. Zhang, J.R.; Tang, H.M.; Tannant, D.D.; Lin, C.Y.; Xia, D.; Liu, X.; Zhang, Y.Q.; Ma, J.W. Combined forecasting model with CEEMD-LCSS reconstruction and the ABC-SVR method for landslide displacement prediction. *J. Clean. Prod.* **2021**, *293*, 126205. [CrossRef]
29. Wang, Y.K.; Tang, H.M.; Wen, T.; Ma, J.W. A hybrid intelligent approach for constructing landslide displacement prediction intervals. *Appl. Soft Comput.* **2019**, *81*, 105506. [CrossRef]
30. Li, D.Y.; Sun, Y.Q.; Yin, K.L.; Miao, F.S.; Glade, T.; Leo, C. Displacement characteristics and prediction of Baishuihe landslide in the Three Gorges Reservoir. *J. Mt. Sci.* **2019**, *16*, 2203–2214. [CrossRef]
31. Zhang, L.G.; Chen, X.Q.; Zhang, Y.G.; Wu, F.W.; Chen, F.; Wang, W.T.; Guo, F. Application of GWO-ELM model to prediction of Caojiatuo landslide displacement in the Three Gorge Reservoir area. *Water* **2020**, *12*, 1860. [CrossRef]
32. Zhang, Y.G.; Chen, X.Q.; Liao, R.P.; Wan, J.L.; He, Z.Y.; Zhao, Z.X.; Zhang, Y.; Su, Z.Y. Research on displacement prediction of step-type landslide under the influence of various environmental factors based on intelligent WCA-ELM in the Three Gorges Reservoir area. *Nat. Hazards* **2021**, *107*, 1709–1729. [CrossRef]
33. Ling, Q.; Zhang, Q.; Zhang, J.; Kong, L.J.; Zhang, W.Q.; Zhu, L. Prediction of landslide displacement using multi-kernel extreme learning machine and maximum information coefficient based on variational mode decomposition: A case study in Shaanxi, China. *Nat. Hazards* **2021**, *108*, 925–946. [CrossRef]
34. Xing, Y.; Yue, J.P.; Chen, C. Interval estimation of landslide displacement prediction based on time series decomposition and long short-term memory network. *IEEE Access* **2020**, *8*, 3187–3196. [CrossRef]
35. Liu, Z.Q.; Guo, D.; Lacasse, S.; Li, J.H.; Yang, B.B.; Choi, J.C. Algorithms for intelligent prediction of landslide displacements. *J. Zhejiang Univ.-Sci. A* **2020**, *21*, 412–429. [CrossRef]
36. Li, J.Y.; Wang, W.D.; Han, Z. A variable weight combination model for prediction on landslide displacement using AR model, LSTM model, and SVM model: A case study of the Xinming landslide in China. *Environ. Earth Sci.* **2021**, *80*, 386. [CrossRef]
37. Zeng, T.R.; Jiang, H.W.; Liu, Q.L.; Yin, K.L. Landslide displacement prediction based on variational mode decomposition and MIC-GWO-LSTM model. *Stoch. Environ. Res. Risk Assess.* **2022**, *36*, 1353–1372.
38. Ma, J.W.; Tang, H.M.; Liu, X.; Hu, X.L.; Sun, M.J.; Song, Y.J. Establishment of a deformation forecasting model for a step-like landslide based on decision tree C5.0 and two-step cluster algorithms: A case study in the Three Gorges Reservoir area, China. *Landslides* **2017**, *14*, 1275–1281. [CrossRef]
39. Wang, Y.K.; Tang, H.M.; Wen, T.; Ma, J.W.; Zou, Z.X.; Xiong, C.R. Point and Interval Predictions for Tanjiahe Landslide Displacement in the Three Gorges Reservoir Area, China. *Geofluids* **2019**, *2019*, 8985325. [CrossRef]
40. Wang, C.H.; Zhao, Y.J.; Bai, L.B.; Guo, W.; Meng, Q.J. Landslide Displacement Prediction Method Based on GA-Elman Model. *Appl. Sci.* **2021**, *11*, 11030. [CrossRef]

41. Zhang, L.; Shi, B.; Zhu, H.H.; Yu, X.B.; Han, H.M.; Fan, X.D. PSO-SVM-based deep displacement prediction of Majiagou landslide considering the deformation hysteresis effect. *Landslides* **2020**, *18*, 179–193. [CrossRef]
42. Lian, C.; Zeng, Z.G.; Wang, X.P.; Yao, W.; Su, Y.X.; Tang, H.M. Landslide displacement interval prediction using lower upper bound estimation method with pre-trained random vector functional link network initialization. *Neural Netw.* **2020**, *130*, 286–296. [CrossRef] [PubMed]
43. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016.
44. Mazumdar, J.; Harley Ronald, G. Recurrent neural networks trained with backpropagation through time algorithm to estimate nonlinear load harmonic currents. *IEEE Trans. Ind. Electron.* **2008**, *55*, 3484–3491. [CrossRef]
45. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
46. Tan, C.D.; Yang, J.Z.; Cui, M.Y.; Wu, H.; Wang, C.Q.; Deng, H.W.; Song, W.R. Research article fracturing productivity prediction model and optimization of the operation parameters of shale gas well based on machine learning. *Lithosphere* **2021**, *2021*, 2884679. [CrossRef]
47. Wang, Y. *Research on Prediction Model and Optimization Method of Ground Settlement of Shield Tunnel Based on Statistical Machine Learning*; Beijing Jiaotong University: Beijing, China, 2021.