Journal of Rock Mechanics and Geotechnical Engineering 16 (2024) 4034–4045

Contents lists available at ScienceDirect

Journal of Rock Mechanics and Geotechnical Engineering

journal homepage: www.jrmge.cn

Full Length Article

Reliability analysis of slope stability by neural network, principal component analysis, and transfer learning techniques

Sheng Zhang [a], Li Ding [a], Menglong Xie [b], Xuzhen He [c], Rui Yang [d], Chenxi Tong [a,*]

[a] School of Civil Engineering, Central South University, Changsha, 410075, China
[b] Changping District Commission of Housing and Urban-rural Development of Beijing Municipality, Beijing, 102200, China
[c] School of Civil and Environmental Engineering, University of Technology Sydney, Broadway, NSW 2007, Australia
[d] School of Civil Engineering, Beijing Jiaotong University, Beijing, 10044, China

ARTICLE INFO

Article history:
Received 15 February 2023
Received in revised form
25 May 2023
Accepted 15 June 2023
Available online 23 November 2023

Keywords:
Slope stability analysis
Monte Carlo simulation
Neural network (NN)
Transfer learning (TL)

ABSTRACT

The prediction of slope stability is considered as one of the critical concerns in geotechnical engineering. Conventional stochastic analysis with spatially variable slopes is time-consuming and highly computation-demanding. To assess the slope stability problems with a more desirable computational effort, many machine learning (ML) algorithms have been proposed. However, most ML-based techniques require that the training data must be in the same feature space and have the same distribution, and the model may need to be rebuilt when the spatial distribution changes. This paper presents a new ML-based algorithm, which combines the principal component analysis (PCA)-based neural network (NN) and transfer learning (TL) techniques (i.e. PCA–NN–TL) to conduct the stability analysis of slopes with different spatial distributions. The Monte Carlo coupled with finite element simulation is first conducted for data acquisition considering the spatial variability of cohesive strength or friction angle of soils from eight slopes with the same geometry. The PCA method is incorporated into the neural network algorithm (i.e. PCA-NN) to increase the computational efficiency by reducing the input variables. It is found that the PCA-NN algorithm performs well in improving the prediction of slope stability for a given slope in terms of the computational accuracy and computational effort when compared with the other two algorithms (i.e. NN and decision trees, DT). Furthermore, the PCA–NN–TL algorithm shows great potential in assessing the stability of slope even with fewer training data.

1. Introduction

Geological hazards triggered by slope instability, such as landslides and failure of dams and open-pit mines, are reported worldwide and will result in huge social and economic losses (Ma et al., 2018; Wang et al., 2021). It is therefore essential to analyze and predict the slope stability for geotechnical engineers in practice. Generally, the stability of slope is evaluated primarily on the basis of the well-known factor of safety (FOS), which is defined as the ratio of the shear strength (or, alternatively, an equivalent measure of shear resistance or capacity) to the shear stress (or other equivalent measure required for equilibrium). In the last few decades, numerous attempts have been made to conduct slope stability analysis based on either the deterministic methods, such as limit equilibrium method (Duncan, 1996), finite elements limit analysis (Sloan, 1988), and strength reduction method (Dawson et al., 1999), or the probabilistic methods, such as first-order second-moment method (Christian et al., 1994), first-order reliability method (Low and Tang, 1997), and random finite-element method (Griffiths and Fenton, 2007). The former approach is simpler, and requires less computing resources, but fails to consider the uncertainty of soil properties (i.e. shear strength and cohesion); while the latter approach is capable of capturing the influence of uncertainties in soil properties, which makes it more reasonable to estimate the slope stability.

The inherent spatial variability of soil properties, which is attributed to different deposition conditions due to different loading histories, has been considered one of the main sources of geotechnical uncertainty. Such variability is typically modeled with random fields and slope stability is estimated using the Monte Carlo sampling approach or Monte Carlo coupled with conventional

* Corresponding author.
E-mail address: cxtong@csu.edu.cn (C. Tong).
Peer review under responsibility of Institute of Rock and Soil Mechanics, Chinese Academy of Sciences.

https://doi.org/10.1016/j.jrmge.2023.06.023

deterministic methods, where the soil properties are commonly assumed to follow different types of autocorrelation functions (ACFs) (Jiang et al., 2014; Li et al., 2015). In a Monte Carlo-based simulation, numerous simulations with varying soil properties is required to obtain a good estimation of FOS, which, however, is time-consuming and highly computation-demanding.

In recent years, with the development of artificial intelligence technology and computer resources, machine learning (ML) has yielded revolutionary results across diverse disciplines, and has also been used as surrogate models for the slope stability analyses, such as support vector machine (SVM) (Zhao, 2008; Kang et al., 2016), artificial neural network (ANN) (Wang et al., 2005; He et al., 2020, 2021a; Meng et al., 2021), decision tree (DT) (Hwang et al., 2009), multivariate adaptive regression splines (MARS) (Liu et al., 2019), and to name a few. The core idea of these ML-based approaches is trying to use a small amount of data, which is commonly generated by finite element simulations, as training dataset to make predictions of FOS for a given slope. Most recently, Qi and Tang (2018) proposed a hybrid ensemble method for the improved prediction of slope stability, which combines with six ML algorithms. He et al. (2020) conducted stochastic reliability analyses with promising accuracy based on two ML algorithms, i.e. neural network (NN) and SVM with only a small size of training data. Meng et al. (2021) adopted ANNs to predict three-dimensional (3D) slope stability. He et al. (2021b) trained deep-learning models with big data for the bearing capacity of strip footing, which covers all the soil properties, loading conditions, and spatial variability commonly encountered in practice, such that numerical simulations and training are not necessary anymore for this problem.

This study presents a new ML-based method for conducting stability analysis of slopes with different spatial distributions, which combines the principal component analysis (PCA)-based neural network and transfer learning (TL) techniques (i.e. PCA–NN–TL). The Monte Carlo coupled with finite element simulation is first conducted for data acquisition, where the ACF is also adopted for describing the spatial variability of cohesive strength or friction angle of soils. Then, the PCA approach is employed to reduce the number of input variables (i.e. cohesive strength or friction angle of each element in FE simulations) in the NN. Thus, the training parameters used in PCA-based NN (i.e. PCA-NN) with a given slope are transferred to other slopes with the same geometry but different spatial distributions. The proposed TL techniques exhibit satisfactory performance even with a small size of the training dataset.

2. Data acquisition considering spatially variable soil strength parameters

In the present study, eight slopes A-H with the same geometry but different spatial distributions are considered. The slope stability analysis is conducted, and the FOS is estimated by the strength reduction method using the commercial finite element software OptumG2. The saturated bulk unit weight is assumed as $\gamma = 19\text{kN/m}^3$. The upper loading is fixed to zero, and the boundary of slopes are standard boundary conditions. The constitutive relationship of soils is the Mohr-Coulomb model.

In a finite element simulation, the grid elements and nodes will be automatically divided according to the shape of slope as shown in Fig. 1. In slopes A-D, the drained friction angle of all nodes is assumed as $\varphi = 19°$. The cohesive strength of soils at various positions on the slope is assumed to satisfy logarithmic normal distribution, which is expressed as (Charlton et al., 2018)

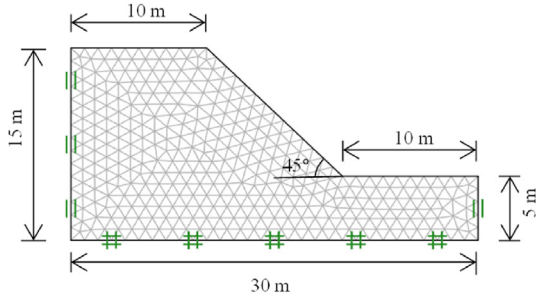

Fig. 1. Schematic diagram of slope.

$$\ln C \sim N\left(\mu, \sigma^2\right) \tag{1}$$

where C denotes the set of cohesion strength, μ is the mean value of $\ln C$, and σ is the standard deviation of $\ln C$. In the present study, the values of μ and σ are shown in Table 1.

The cohesion strength of the ith position Q_i in the random field can be expressed as

$$c(x_i, y_i) = \exp[\mu + \sigma G(x_i, y_i)] \tag{2}$$

where $Q_i = (x_i, y_i)$ is the coordinates of the random field position Q_i, and G is a standard normal distribution random field.

In order to take account of the inherent spatial variability of soil properties, the following *ACF* is adopted for correlating the cohesive strength of soils at any two positions on the slope (Wu et al., 2012; Li et al., 2015).

$$ACF\left[c(x_i, y_i), c\left(x_j, y_j\right)\right] = \frac{COV\left[c(x_i, y_i), c\left(x_j, y_j\right)\right]}{\sqrt{Var[c(x_i, y_i)]}\sqrt{Var\left[c\left(x_j, y_j\right)\right]}} \tag{3}$$

where *COV* is the covariance function and *Var* is the variance function. The *ACF* satisfies the following form:

$$ACF\left[c(x_i, y_i), c\left(x_j, y_j\right)\right] = \exp\left[-2\left(\frac{\tau_x}{\delta_h} + \frac{\tau_y}{\delta_v}\right)\right] \tag{4}$$

$$\tau_x = |x_i - x_j| \tag{5}$$

$$\tau_y = \left|y_i - y_j\right| \tag{6}$$

where τ_x is the absolute difference in the horizontal distance between two nodes, τ_y is the absolute difference in the vertical distance between two nodes, $\delta_h = 40$ m is the horizontal autocorrelation distance, and $\delta_v = 1$ m is the vertical autocorrelation distance as suggested by Li et al. (2015).

In slopes *E*-H, the cohesive strength is assumed as $C = 10$ kPa. The friction angle of soils at various positions on the slope is again assumed to satisfy logarithmic normal distribution. The values of μ

Table 1
Relevant soil strength parameters of slopes A-D.

Dataset	μ (ln kPa)	σ	φ (°)	Number of nodes
A	2.2595	0.2936	19	386
B	2.665	0.2936	19	386
C	2.9526	0.2936	19	386
D	3.1758	0.2936	19	386

and σ are shown in Table 2. And the values of δ_h and δ_v are 40 m and 1 m, respectively.

3. Stability assessment for a given slope using PCA-based neural networks (PCA-NNs)

3.1. PCA

PCA is an efficient multivariate method for reducing the dimensionality of datasets, increasing interpretability but at the same time minimizing information loss, and has been successfully used in many applications for several decades (Hasan and Abdulazeez, 2021). The mathematical essence of the PCA method is the rotation and transformation of coordinates. The original n variables are linearly combined to generate n new variables in the new coordinate system, while they are not related to each other (Ture et al., 2007). For the sake of simplicity, take the slope with only two nodes in the finite element simulations as an example, the main steps of the PCA method are summarized as follows (Smith, 2002).

(1) Centralize the cohesion matrix

Define C_{ij} that represents the cohesion value of the ith node ($i = 1, 2$) for the jth simulation ($1 \le j \le n$). The standardization of $\tilde{C}_{ij}$ can be calculated as $\tilde{C}_{ij} = C_{ij} - \frac{1}{n}\sum_{j=1}^{n} C_{ij}$ after data centralization (see Fig. 2b), and the standardization of the cohesion matrix can be expressed as

$$\boldsymbol{A} = \begin{bmatrix} \tilde{C}_{11} & \tilde{C}_{12} & \cdots & \tilde{C}_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \end{bmatrix}^{\mathrm{T}} \tag{7}$$

(2) Calculate the covariance matrix

In order to understand the relationship between the variables of the input dataset, and therefore keep as much information of the original data as possible, the following covariance matrix is then computed as expressed by Shlens (2014):

$$\boldsymbol{B} = \frac{1}{n-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} = \frac{1}{n-1}\begin{bmatrix} \sum_{j=1}^{n} \tilde{C}_{1j}{}^{2} & \sum_{j=1}^{n} \tilde{C}_{1j}\tilde{C}_{2j} \\ \sum_{j=1}^{n} \tilde{C}_{2j}\tilde{C}_{1j} & \sum_{j=1}^{n} \tilde{C}_{2j}{}^{2} \end{bmatrix} \tag{8}$$

(3) Compute the eigenvalues and eigenvectors for the covariance matrix

Rank the eigenvectors in order of their eigenvalues, highest to lowest. Define the contribution rate that represents how much information the corresponding component under the new coordinate contains.

(4) Choose a benchmark for dimensionality reduction

Define the sum of the contribution rate as the cumulative contribution rate, which indicates how much information the new coordinate contains (Pu et al., 2019).

$$\beta = \frac{\sum_{i=1}^{k}\delta_i}{\sum_{i=1}^{n}\delta_i} \tag{9}$$

where β is the cumulative contribution rate, δ_i is the eigenvalues of the covariance matrix, and k is the size of the reduced dimension, indicating that the new projection coordinates are composed of k variables.

(5) Calculate the dataset after dimensionality reduction

The original data (see Fig. 2a) are projected onto the projection coordinates determined by step (4) (see Fig. 2c), and the reduced dimensionality data are obtained.

3.2. PCA-NN for predicting FOS

ANN is a computing model inspired by the biological neural networks that constitute animal brains and has been widely used in the field of geotechnical engineering (Anderson, 1995). Fig. 3a shows a typical structure of NN, which includes one input layer with n variables, one output layer with m output variables, and one hidden layer. As shown in Fig. 3a, each layer includes neurons that are not connected within the layer but are fully connected with the neurons from the neighboring layers. The neurons' weight is adjusted to improve the accuracy of the model during the learning process (Cheng et al., 2007). Fig. 3b shows a typical structure of PCA-NN algorithm with the input size being reduced to k based on the abovementioned steps of PCA. As indicated in Fig. 3b, the only difference between NN and PCA-NN algorithms is the input variable. PCA is used to remove new variables that have little impact on the results, so as to achieve the purpose of data dimensionality reduction. Therefore, the PCA-NN algorithm aims to reduce the data dimensionality in order to find the optimal dimension, which can both ensure a good fit and reduce the computation time.

To determine the optimal size of the input layer in a PCA-NN algorithm, the datasets of slopes A and E are selected as examples, for which the finite element model involves 386 nodes with the cohesive strength or the friction angle being spatially variable. Therefore, the input size of the conventional NN algorithm is 386, and the output size is 1 (i.e. FOS). The input size is reduced using the PCA method. The original data compose a 5000 × 386 matrix, which is centralized and then calculated the covariance matrix of the sample matrix. After that, the eigenvalues and eigenvectors of the covariance matrix are found, and the summed eigenvectors are combined according to the size of the eigenvalues to form a mapping matrix. Finally, the first n columns of the mapping matrix are set as the final mapping matrix according to the specified number of features retained by PCA to achieve the purpose of data dimensionality reduction. In this study, in order to find the optimal dimension, a reduction of n from 386 to 10 was tested sequentially. It is suggested that the performance of the NN algorithm will be improved, and then approaches a stable level with increasing hidden layers and number of neurons per layer for most cases. In the present study, the architecture of the NN and PCA-NN algorithms is fixed to one hidden layer with 30 neurons, while the

Table 2
Relevant soil strength parameters of slopes *E*-H.

Dataset	μ (ln kPa)	σ	C (kPa)	Number of nodes
E	2.9248	0.198	10	386
F	3.0249	0.198	10	386
G	3.1149	0.198	10	386
H	3.1993	0.198	10	386

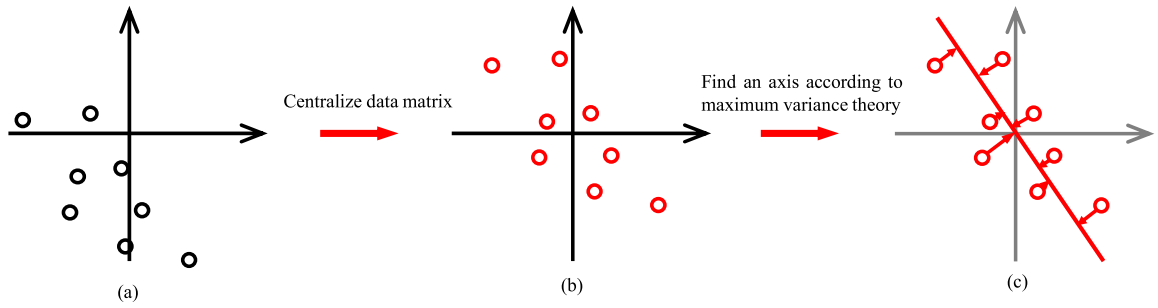

Fig. 2. Process of PCA: (a) Original data, (b) original centralization, and (c) original projection.

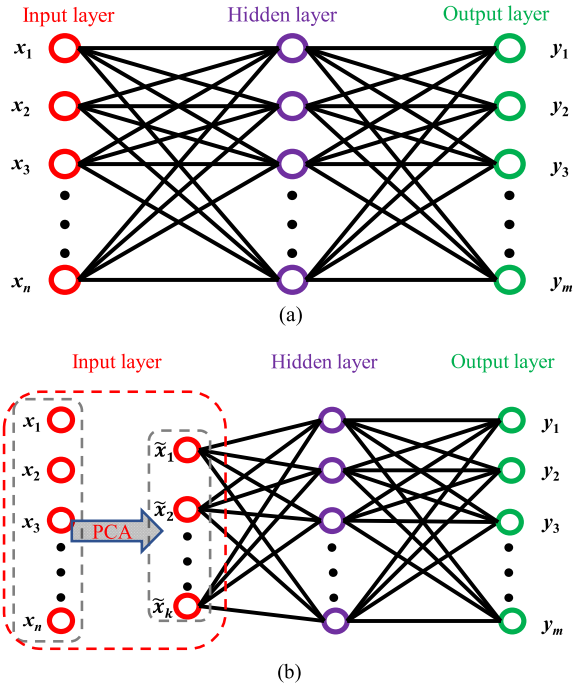

Fig. 3. Schematic diagram of (a) typical NN and (b) PCA-NN.

learning rate and the number of iterations are fixed to 0.001 and 10,000 by trial and error, respectively. After initial tests, it is determined the root mean square error (*RMSE*) is 1×10^{-6}. All the hyperparameters adopted in this paper are summarized in Table 3. The nonlinear activation function, i.e. the sigmoid function is adopted in this paper for enhancing the network representation and learning ability, and is computed as

$$F(T) = \frac{1}{1 + \exp(-T)} \tag{10}$$

where T is the linear weighted sum of inputs at the input layer plus the bias term.

All simulations are conducted on a Dell laptop with Intel(R) Core(TM) i5-8250U CPU @ 1.60 GHz 1.80 GHz. The coefficient of determination (R^2) is employed as the measurement of goodness-of-fit, which is defined as (Crewson, 2014)

Table 3
Parameters used in the NN/PCA-NN algorithm.

Parameter	Value
Hidden layer	1
Neuron	30
Learning rate	0.001
Epoch	10,000
RMSE	1×10^{-6}

$$R^2 = \frac{\left(n\sum_{i=1}^{n}\hat{t}_i t_i - \sum_{i=1}^{n}\hat{t}_i \sum_{i=1}^{n} t_i\right)^2}{\left[n\sum_{i=1}^{n}\hat{t}_i^2 - \left(\sum_{i=1}^{n}\hat{t}_i\right)^2\right]\left[n\sum_{i=1}^{n} t_i^2 - \left(\sum_{i=1}^{n} t_i\right)^2\right]} \tag{11}$$

where, $\hat{t}_i$ is the predicted FOS and t_i is the calculated FOS. An R^2 value closer to 1 indicates a better performance of the model. A dataset for the NN/PCA-NN algorithm is required to be randomly split into two independent subsets, i.e. the training dataset and the validation dataset. In the present study, the total number of available data is 5000, while the number of the training dataset is ranging from 1000 to 4000.

Fig. 4a shows the performance of the PCA-NN algorithm with varying number of input variable and training dataset in the case of slope A. It is worth noting that even if the number of input variable is reduced from 386 to 50, the performance of the PCA-NN algorithm is better maintained or even better, while the performance of the PCA-NN algorithm deteriorates when the number of input size is further reduced, as indicated in Fig. 4a. Such behaviors seem reasonable because those key input variables, such as the strength of the soil represented by the nodes near the sliding surface, can be selected using the PCA method when conducting slope stability analysis, while the others with less useful information will be discarded, resulting in a simpler structure of NN and a better performance. It is also found that the larger the number of training dataset, the higher the accuracy of the PCA-NN algorithm for a given number of input size. Fig. 4a also shows the corresponding computation time for all the cases of slope A. As expected, a larger number of input variable and training dataset will both lead to a higher computation time for the PCA-NN algorithm. This finding is also applicable to the slope with different soil properties, for example, slope E as shown in Fig. 4b.

In this study, the size of the input layer is fixed to 150 for the PCA-NN algorithm considering both the computational accuracy (i.e. coefficient of determination) and computational effort (i.e. computation time). Furthermore, the cumulative distribution function (CDF) of FOS is used for further evaluating the effectiveness of the proposed PCA-NN algorithm (Huang et al., 2007). Fig. 5

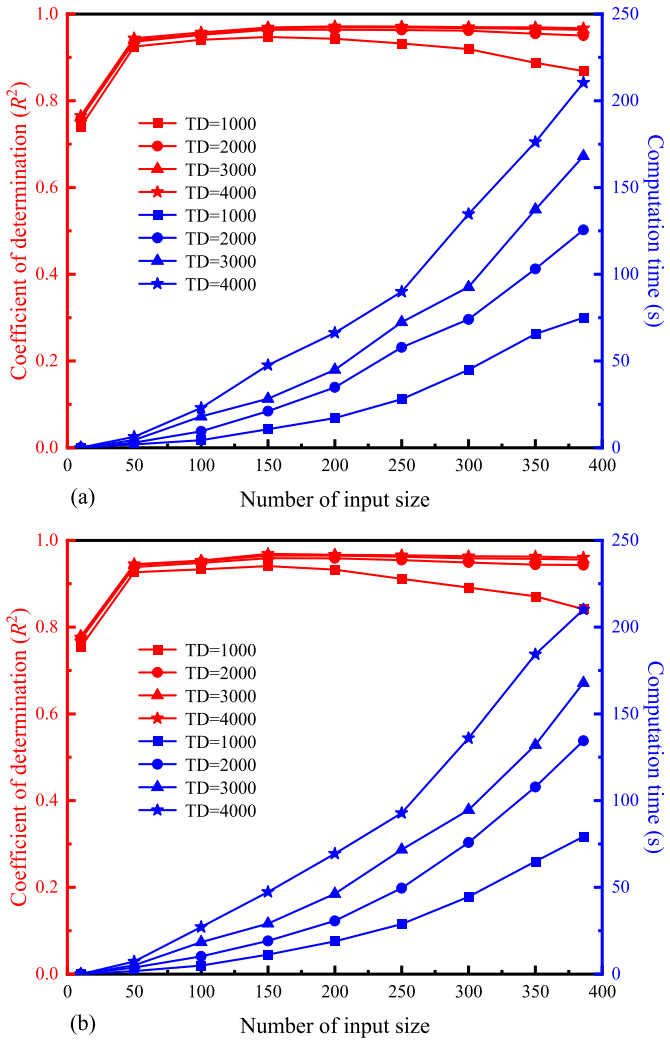

Fig. 4. The performance of the PCA-NN algorithm with varying input variables and training datasets: (a) slope A and (b) slope E. TD is short for the training dataset in the figure.

presents the comparisons between the PCA-NN algorithm at different training datasets and the finite element method (FEM) for predicting the CDF of FOS of slopes A and E. For each training dataset, the CDF of FOS derived by PCA-NN algorithm and the results derived by FEM are very close, indicating that the PCA-NN method can well capture the global distribution of finite element prediction. As shown in Fig. 5, it is clear that the PCA-NN algorithm proposed in this paper shows better performance compared with FEM for a given slope in terms of computational efficiency as it only needs several seconds, which is extremely fewer than that of FEM even though the CDF of FOS is almost the same. In order to verify the proposed PCA-NN algorithm, other two ML-based algorithms are also adopted for comparisons, i.e. NN and DT in terms of computational accuracy and computational effort. All the algorithms are conducted with the same computer resource. Table 4 shows the coefficient of determination (R^2) and computation time (t) for all the three algorithms with varying number of training dataset. A larger number of training dataset will generally lead to more accurate predictions and larger computation time of the three ML algorithms. However, when the training dataset is sufficiently large (i.e. 4000), these models have almost no significant improvement. In general, the proposed PCA-NN algorithm is superior to the NN and DT algorithms due to its better performance (i.e. higher coefficient of determination) and shorter training time.

The neural network with reduced inputs in PCA-NN performs essentially the same as pure NN with full inputs. More complex models (more inputs) usually require more data to train, and PCA can remove the less influential variables and reduce the data dimension, which can maintain better performance and greatly reduce training time. The use of PCA is like the use of filters in deep learning, and the selection of principal component is like abstractions in deep learning (He et al., 2021b), so it helps to obtain a better model.

It should be noted that the slope cases considered in this study are simple, there is a reasonable prospect that such a PCA-NN algorithm shows great potential in greatly improving the computation effort but with a high standard of computation accuracy for predicting the FOS of slopes with more complex conditions.

4. Slope stability assessment for slopes with different spatial distributions by TL

4.1. TL method

For most ML algorithms, the same feature space and the same distribution are needed for the training and validation datasets. However, many ML models need to be rebuilt and the training dataset needs to be recollected when the task changes. Therefore, TL is proposed to establish the connection between the source task domain and the target task domain (Gao and Mosalam, 2018). Specifically, the objective of TL is to help improve the prediction function in learning target task using the knowledge from source domain with source target, more details can be found from the work of Pan and Yang (2009) and Tan et al. (2018).

For example, as indicated in Fig. 6a, conventional ML-based stochastic reliability analysis uses soil strength parameters to predict FOS, and slopes of the same shape with different spatial distributions may use different sets of hyperparameters, which cannot establish the connection between models and models, resulting in a significant increase in workload. The TL-based stochastic reliability analysis attempts to apply the basic structure, hyperparameters, connection weights, and other knowledge of the existing algorithm model (i.e. the source domain) directly to the new model (i.e. the target domain), as shown in Fig. 6b. This method can greatly reduce the sample data required for the training of the new model, and at the same time achieve a better fitting effect, which makes the slope stability analysis easier.

In this paper, a parameter-transfer approach is proposed to predict the FOS of slopes B-D and F—H by transferring the hyperparameters of the PCA-NN algorithm adopted by slopes A and E (i.e. the architecture of NN, the weights between nodes, the learning rate, and the number of iterations), respectively. As mentioned earlier, the input layer size of slopes A and E is reduced to 150 b y the PCA method with the FOS being predicted using the NN algorithm. It is worth noting that slopes A and E are already trained algorithm models and can achieve good fitting effects. Moreover, slope A is related to slopes B-D, and slope E is also related to slopes F—H. Therefore, slope A and slope E will be used as the source domain to perform TL on the target slopes. Building the slope algorithm model from scratch is complex and time-consuming, and TL can improve the computational efficiency and effect. Therefore, in the following study, the input size used by the PCA-NN algorithm for slopes B-D and F—H is also reduced to 150 to ensure that the number of input variables is the same for all slopes, which is also a prerequisite for utilizing transfer learning idea. Because all connection weights of the trained PCA-NN algorithm model (i.e. slope A) can be migrated to the target slope (i.e. slope B) only if the

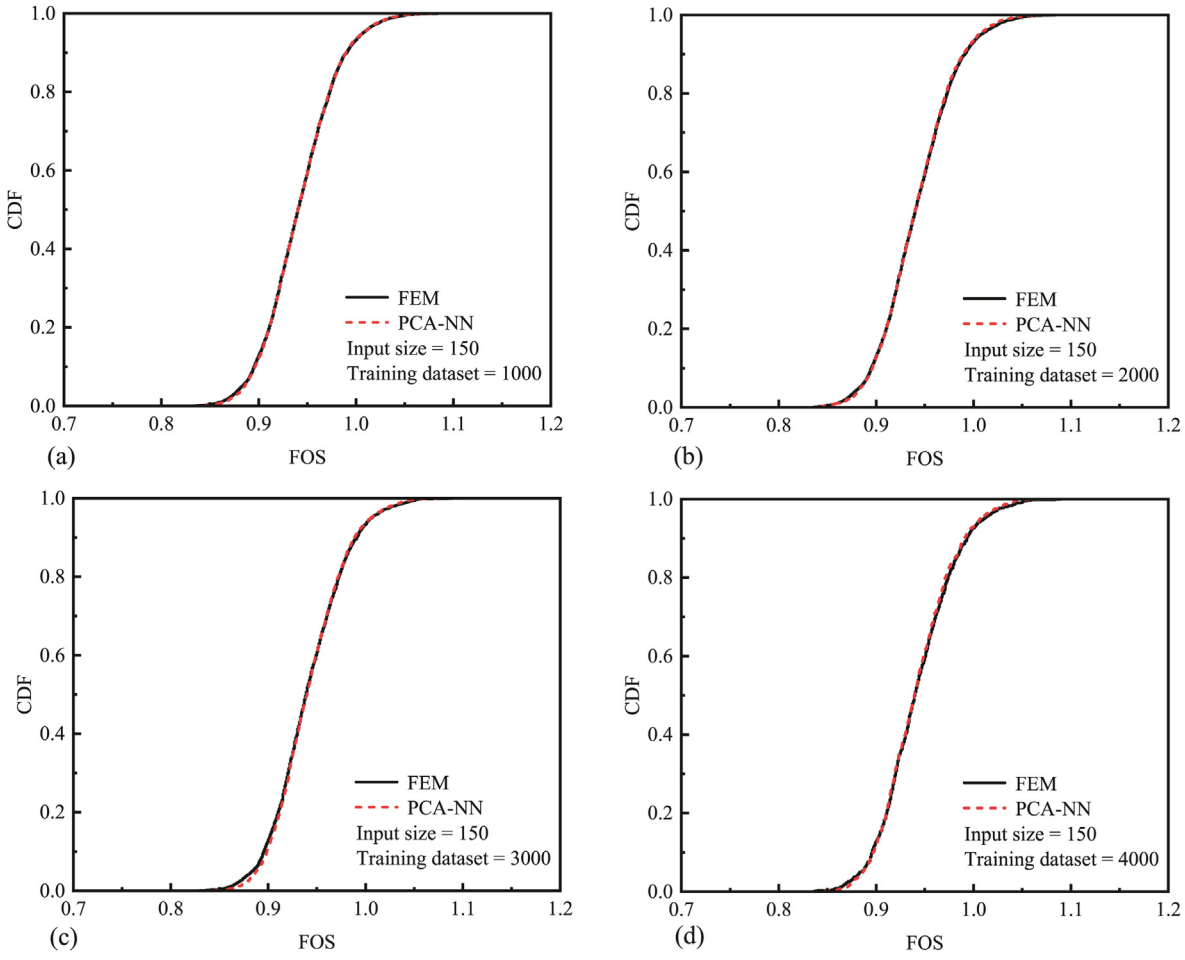

Fig. 5. Cumulative probability distribution of FEM and PCA-NN for FOS on slopes A and E for four training datasets: (a)–(d) slope A; and (e)–(h) slope E.

input size of all slopes is consistent. If the input size is inconsistent, the connection weights cannot be migrated, so the connection weights of the target slope need to be initialized, which will greatly reduce the computational effect of the model in the case of small training datasets.

We apply the architectural forms and hyperparameters (i.e. number of hidden layers, number of hidden layer neurons, learning rate, number of iterations, connection weights) of slopes A and E directly to slopes B-D and E-H, respectively. The hyperparameters for slopes B-D and E-H are also set as shown in Table 3. This is a set of parameters that we have determined after many trials. Thus, we migrate directly to slopes B-D and E-H. In NN algorithm, each neuron is connected to all neurons in the next layer, and each connection has a weight parameter that determines how much the output of the current neuron affects the next layer. In this paper, the number of input layer neurons is 150, the number of hidden layers is 1, the number of hidden layer neurons is 30, and the number of input layer neurons is 1. Therefore, there are 4530 ($=150 \times 30 + 30 \times 1$) connection weights.

Taking the transfer learning of slope A to slope B as an example, the hyperparameters of slope B (e.g. the number of hidden layers and neurons per layer, learning rate, number of iterations, *RMSE*) directly use the parameters determined in slope A, which is the first step of TL. The second step of TL is to transfer the connection weights between the input layer and the hidden layer and the connection weights between the hidden layer and the output layer of slope A. The slope A model has been trained, and the connection weights between its layers have been optimized and adjusted to minimize losses. Therefore, we assign 4530 ($=150 \times 30 + 30 \times 1$) weight parameters of slope A directly to slope B, i.e. the initial weight parameters of slope B are not randomly initialized, but have been optimized and adjusted, so as to quickly achieve better prediction performance under a small amount of training data. The structure of the proposed network-based parameter passing algorithm is shown in Fig. 7, where ω_i represents the weight between the input layer and the hidden layer, and ω_j represents the weight between the hidden layer and the output layer. Thus, all 4530 weight parameters and other training parameters between nodes are transferred from the source slopes A and E to the target slopes B-D and F–H in the initial state, respectively.

4.2. Application examples

In this study, the slope geometry is fixed, but soil properties are allowed to vary spatially according to random field simulations. To illustrate that the proposed method can perform TL on the same shape slope with different spatial distributions, the prediction results of FOS using PCA–NN–TL algorithm are compared with the coefficient of determination (R^2) and mean absolute error (*MAE*) of the pure PCA-NN algorithm under the different numbers of training datasets, as shown in Fig. 8. *MAE* is defined as (Willmott and Matsuura, 2005)

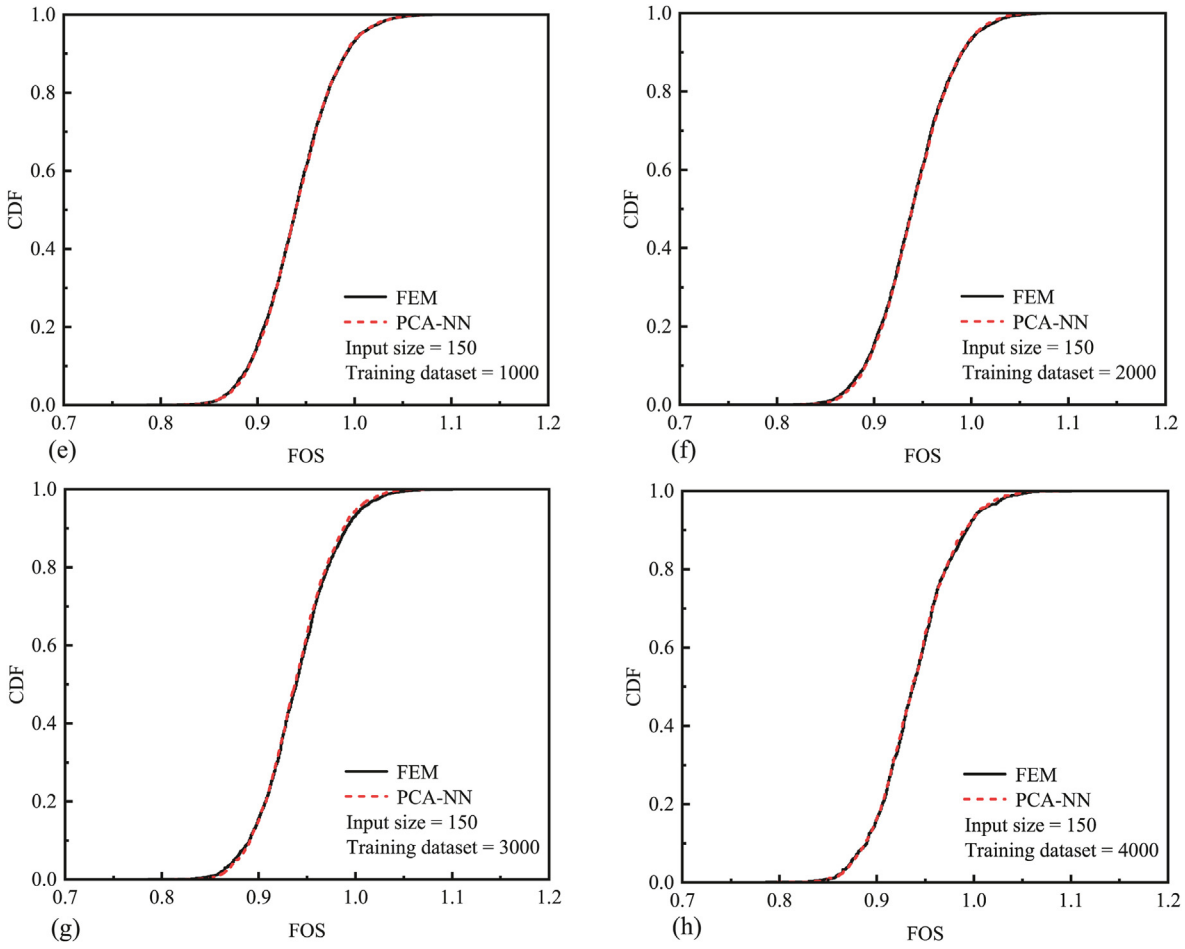

Fig. 5. (*continued*).

$$MAE = \frac{\sum_{i=1}^{n} |\widehat{t_i} - t_i|}{n} \tag{12}$$

A *MAE* value closer to 0 indicates a better performance of the model.

Fig. 8 shows the comparisons between PCA–NN–TL algorithm and PCA-NN algorithm for predicting FOS in terms of R^2 and *MAE* of different slopes. In Fig. 8b, d, f, h, j, and l, each algorithm of each subplot contains 4900 scatters that correspond to 4900 Monte Carlo samples. It can be seen from Fig. 8 that the performance of PCA–NN–TL algorithm is much better than that of PCA-NN algorithm, especially when the number of training datasets is small, indicating that all the hyperparameters of the proposed PCN-NN-TL algorithm can reach the optimal values in a short training time. As expected, the performances of both approaches are improved with increasing of number of training dataset. However, the differences of R^2 and *MAE* between these two approaches decrease when the number of training dataset increases. For example, as indicated in Fig. 8, the value of R^2 for the method of PCA–NN–TL is more than 0.8 for all the slopes, while the value of R^2 for the pure PCA-NN algorithm is almost 0 when the number of training data is 100. It should be noted that the value of R^2 for both approaches will be the same when the number of training dataset is sufficiently large because all the parameters will be the optimal values when the models have been trained sufficiently, which is not presented in this paper. The abovementioned findings highlight the conclusion that the PCA–NN–TL algorithm shows great potential in assessing the stability of slope when it has fewer training data, while the conventional NN algorithm needs a larger amount of training data to reach the same level of performance.

5. Discussion and limitations of the proposed method

The previous sections have shown the effectiveness of the proposed method in analyzing slope stability in terms of both computational time and accuracy when compared with other methods in the literature. The data for the training dataset are generated from a simple case with a fixed slope geometry, while the soil properties are allowed to change spatially according to random field simulations. The PCA method incorporated into the NN algorithm not only increases the computational efficiency by reducing the input variables but ensures the same number of input variables for the slope with different features, which is a prerequisite for the parameter-transfer approach. Therefore, the proposed PCA–NN–TL algorithm provides new ideas with a broad application prospect in the field of slope stability analysis even though the training data are not in the same feature space, for example, the slope geometry is changed, or much more complex geological uncertainties are expected to be considered.

It should be noted that the proposed method for the slope stability analysis problems still suffers from limitations in that the input variables can only be a set of one-dimensional (1D) data (i.e. the cohesive strength or friction angle) since the NN algorithm

Table 4
Coefficient of determination (R^2) and computation time (t) of different algorithms of slopes A and E.

Dataset	Number of TD	PCA-NN		NN		DT	
		R^2	t (s)	R^2	t (s)	R^2	t (s)
A	1000	0.947	10.7	0.9429	125.4	0.0746	0.14
	2000	0.9639	21.1	0.9724	194.5	0.0813	0.24
	3000	0.9675	28.2	0.9814	275.3	0.0763	0.4
	4000	0.9688	47.6	0.9848	375.8	0.0927	0.56
E	1000	0.9405	11.2	0.9366	100.6	0.3519	0.16
	2000	0.959	19.1	0.9691	158.9	0.4136	0.28
	3000	0.9649	29.1	0.9788	230.3	0.4103	0.37
	4000	0.969	47.3	0.9816	302.6	0.4319	0.5

Note: TD is short for the training dataset.

adopted in this study. In that case, the cross-correlation between cohesive strength and friction angle of soil cannot be well considered. Such a limitation can be alleviated using a more advanced NN algorithm such as the convolutional NN algorithm, which can fully consider the couple relationship between soil input parameters with multi-dimensional data. Furthermore, the present study also suffers the same limitation as other studies in the sense that the choice of hyperparameters is more or less empirical.

6. Summary and conclusion

This paper presented a new technique to build a surrogate model for predicting the FOS slope with different spatial distributions of soil properties (i.e. PCA–NN–TL). The PCA-NN algorithm improves the computational efforts at a small cost of computational accuracy compared with the conventional NN algorithm, and the TL algorithm is adopted for improving the computational accuracy of unknown target slopes with limited training data. The main conclusions are as follows:

(1) Eight slopes with different spatial distributions of cohesive strength or friction angle are considered in this paper. The FOS is estimated by the Monte Carlo coupled with finite element simulations, where the inherent spatial variability of cohesive strength or friction angle is considered via the ACF.

(2) NN algorithm establishes the relationship between the input variables (i.e. samples of random fields) and the output variable (i.e. FOS) for a given slope, which has been proven its satisfactory computational time compared with conventional Monte Carlo coupled with finite element analysis, although it still needs a long training time because of the numerous trainable parameters for those complex slopes. This paper further proposes a new PCA-NN algorithm with the aim of improving computational time at a small cost of

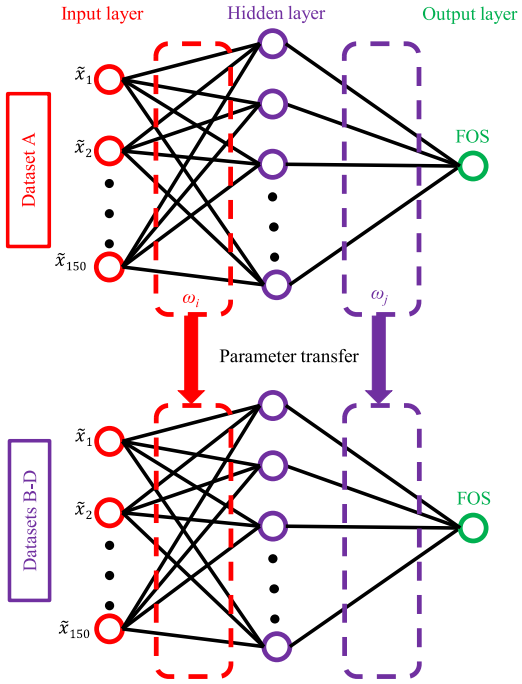

Fig. 7. Structure of the network-based parameter-transfer algorithm.

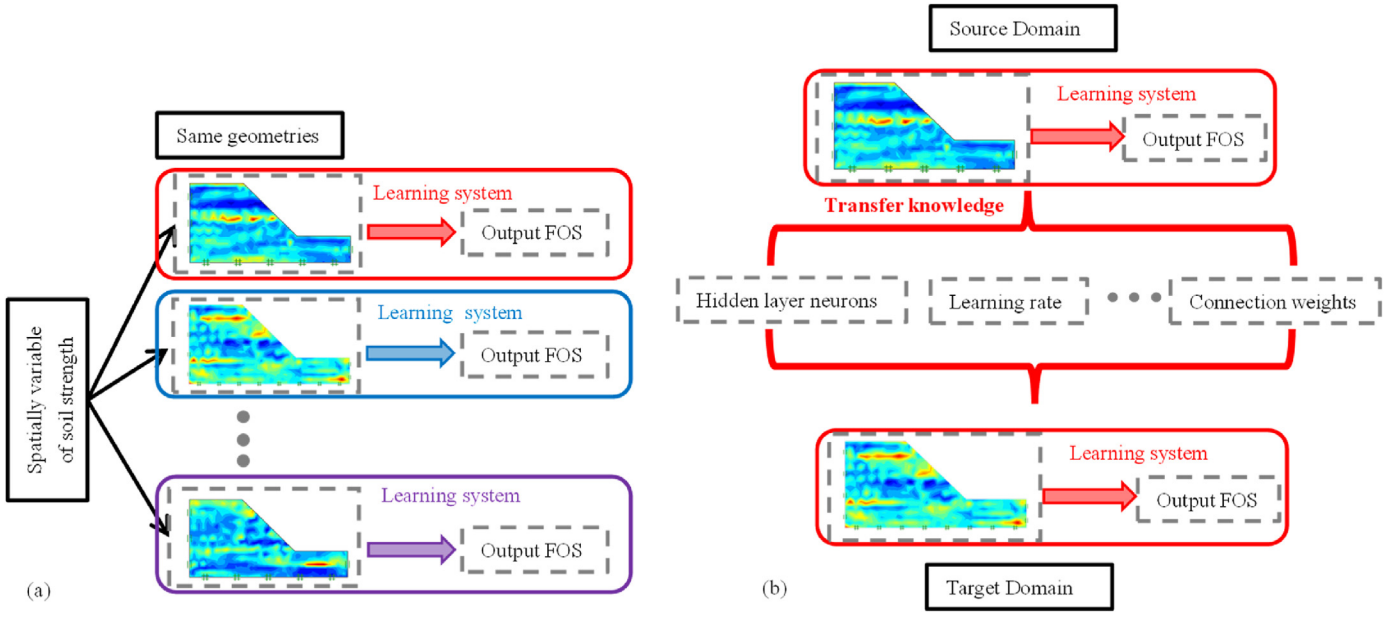

Fig. 6. Learning process of (a) conventional ML-based stochastic reliability analysis and (b) TL-based stochastic reliability analysis.

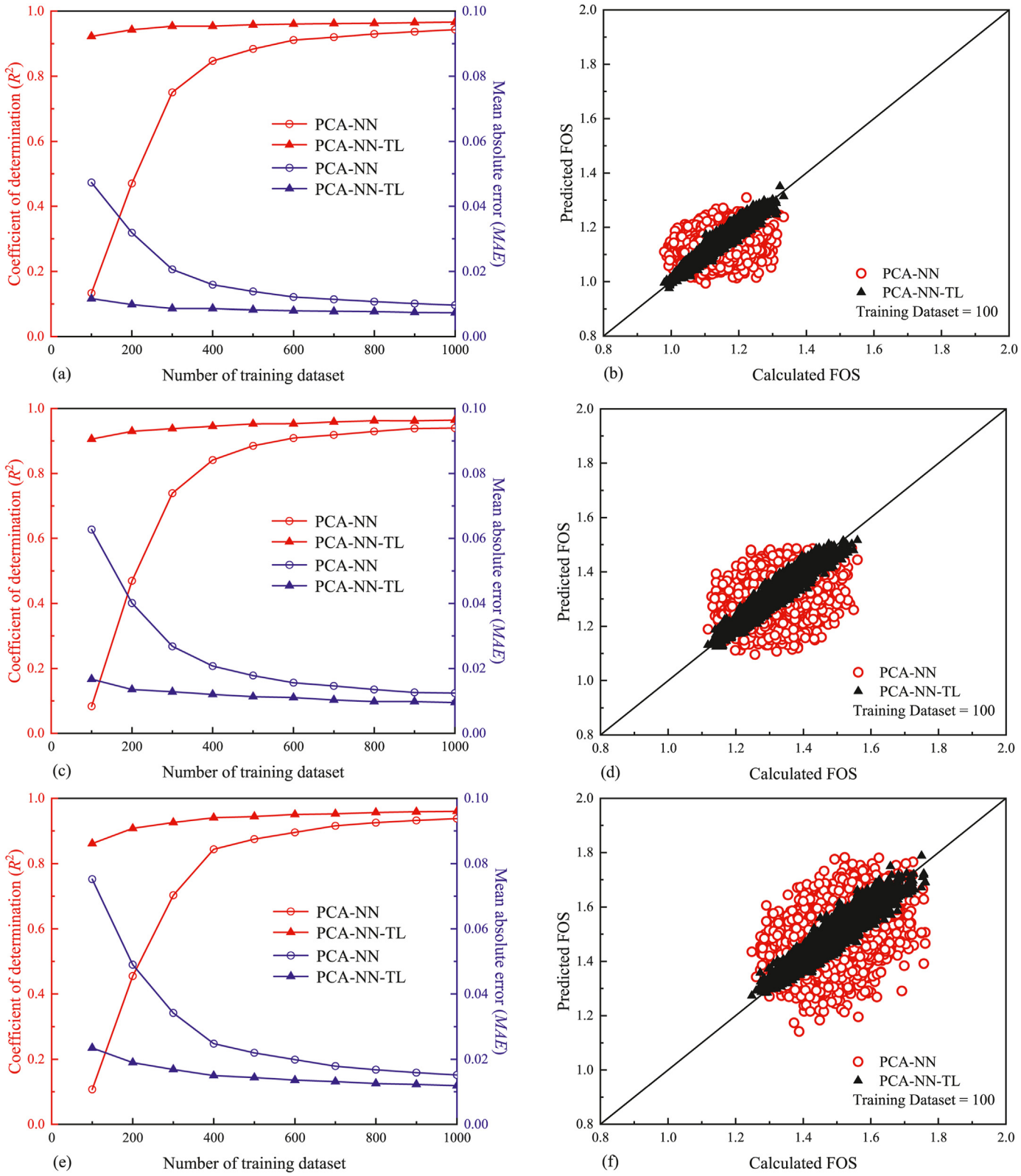

Fig. 8. Comparisons between PCA–NN–TL algorithm and PCA-NN algorithm for predicting FOS in terms of R^2 and *MAE* of different slopes: (a)–(b) slope B; (c)–(d) slope C; (e)–(f) slope D; (g)–(h) slope F; (i)–(j) slope G; and (k)–(l) slope H.

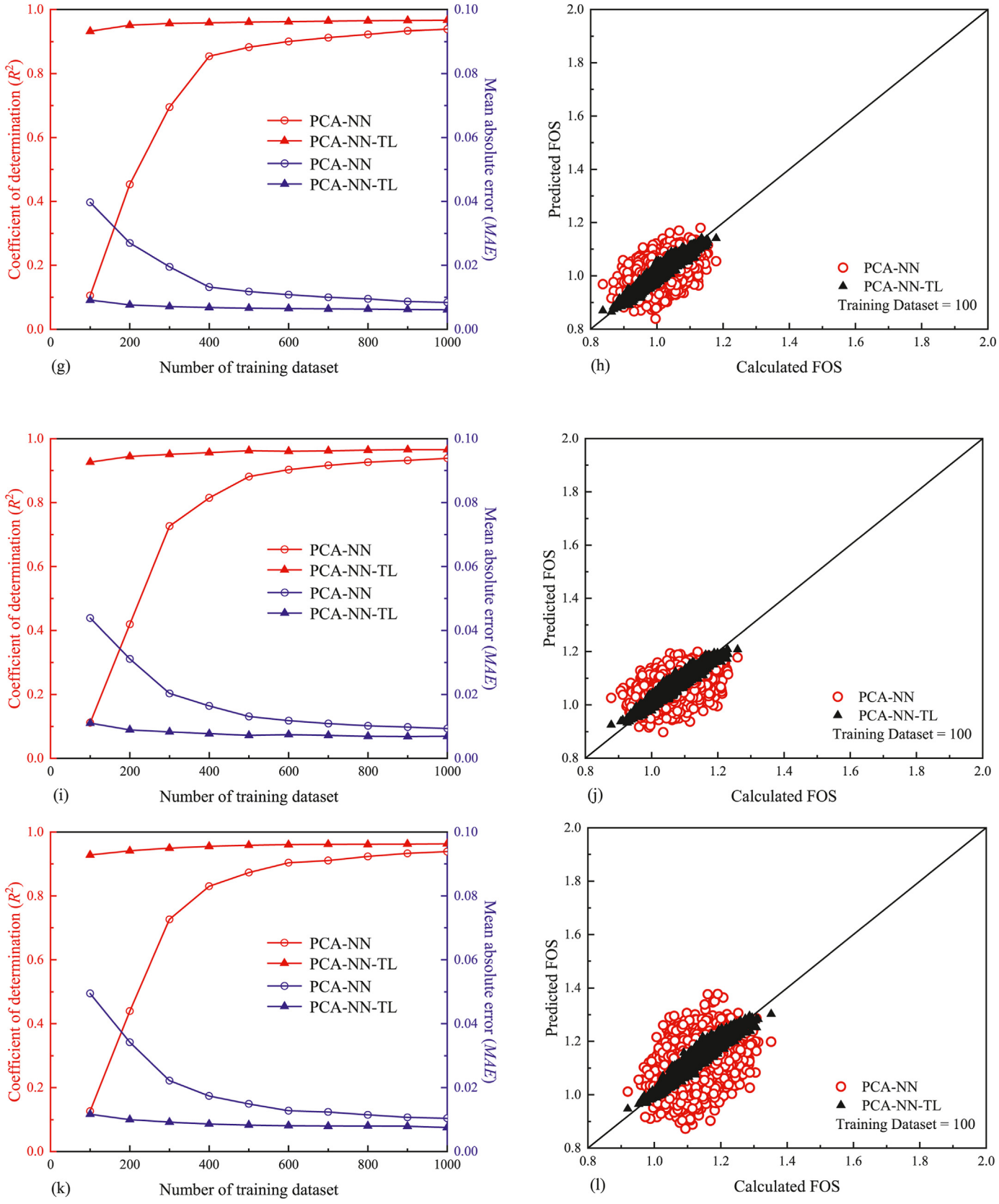

Fig. 8. (*continued*).

computational accuracy. The PCA method is employed to reduce the number of input variables of the NN algorithm. It has been demonstrated that the computational time of the PCA-NN algorithm is only 1/10 of that of the NN algorithm, and the computational accuracy (i.e. R^2) of the PCA-NN algorithm can be basically the same as the NN algorithm when the dimensionality is reduced to 150.

(3) For the slopes with the same geometry but different spatial distributions of soil properties, the PCA-NN models commonly need to be rebuilt and the training dataset needs to be recollected for each slope. TL algorithm is therefore introduced to transfer the hyperparameters from the previous slope to target slopes with limited training data. It is found that the TL algorithm greatly improves the performance of the PCA-NN models, especially when the number of training data is limited. The PCA–NN–TL algorithm provides new ideas with a broad application prospect in the field of slope stability analysis in terms of both computational time and computational accuracy even when the field-measured training data are limited.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant No. 52008402) and the Central South University autonomous exploration project (Grant No. 2021zzts0790).

List of symbols

γ	saturated bulk unit weight
φ	drained friction angle
C	set of cohesion strength
μ	mean value of cohesion
σ	standard deviation of cohesion
Q_i	ith position in random field
G	standard normal distribution random field
τ_x	absolute difference in the horizontal distance between two nodes
τ_y	absolute difference in the vertical distance between two nodes
δ_h	horizontal autocorrelation distance
δ_v	vertical autocorrelation distance
A	standardization of the cohesion matrix
C_{ij}	cohesion value of the ith node for the jth simulation
B	covariance matrix
β	cumulative contribution rate
δ	eigenvalues of the covariance matrix
k	size of reduced dimension
T	the linear weighted sum of inputs at the input layer plus the bias term
R^2	coefficient of determination
t	computation time
$\widehat{t_i}$	the predicted FOS
t_i	the calculated FOS
ω_i	the weight between the input layer and hidden layer
ω_j	the weight between the hidden layer and output layer

References

Anderson, J.A., 1995. An Introduction to Neural Networks. MIT Press, Cambridge, MA, USA.
Charlton, T.S., Rouainia, M., Dawson, R.J., 2018. Control variate approach for efficient stochastic finite-element analysis of geotechnical problems. ASCE-ASME J. Risk. Uncertain. Eng. Syst. Part A.-Civ. Eng. 4 (3), 04018031.
Cheng, J., Zhang, J., Cai, C.S., Xiao, R.C., 2007. A new approach for solving inverse reliability problems with implicit response functions. Eng. Struct. 29 (1), 71–79.
Christian, J.T., Ladd, C.C., Baecher, G.B., 1994. Reliability applied to slope stability analysis. J. Geotech. Eng. 120 (12), 2180–2207.
Crewson, P., 2014. Applied Statistics. AcaStat Software: Winter Garden, FL, USA.
Dawson, E.M., Roth, W.H., Drescher, A., 1999. Slope stability analysis by strength reduction. Geotechnique 49 (6), 835–840.
Duncan, J.M., 1996. State of the art: limit equilibrium and finite-element analysis of slopes. J. Geotech. Eng. 122 (7), 577–596.
Gao, Y., Mosalam, K.M., 2018. Deep transfer learning for image-based structural damage recognition. Comput. Civ. Infrastruct. Eng. 33 (9), 748–768.
Griffiths, D.V., Fenton, G.A., 2007. The random finite element method (RFEM) in slope stability analysis. In: Probabilistic Methods in Geotechnical Engineering. Springer, Vienna, Austria, pp. 317–346.
Hasan, B.M.S., Abdulazeez, A.M., 2021. A review of principal component analysis algorithm for dimensionality reduction. J. Soft Comput. Data Min. 2 (1), 20–30.
He, X., Xu, H., Sabetamal, H., Sheng, D., 2020. Machine learning aided stochastic reliability analysis of spatially variable slopes. Comput. Geotech. 126, 103711.
He, X., Wang, F., Li, W., Sheng, D., 2021a. Efficient reliability analysis considering uncertainty in random field parameters: trained neural networks as surrogate models. Comput. Geotech. 136, 104212.
He, X., Wang, F., Li, W., Sheng, D., 2021b. Deep learning for efficient stochastic analysis with spatial variability. Acta Geotech 1–21.
Huang, S., Mahadevan, S., Rebba, R., 2007. Collocation-based stochastic finite element analysis for random field problems. Probab. Eng. Eng. Mech. 22 (2), 194–205.
Hwang, S., Guevarra, I.F., Yu, B., 2009. Slope failure prediction using a decision tree: a case of engineered slopes in South Korea. Eng. Geol. 104 (1–2), 126–134.
Jiang, S.H., Li, D.Q., Zhang, L.M., Zhou, C.B., 2014. Slope reliability analysis considering spatially variable shear strength parameters using a non-intrusive stochastic finite element method. Eng. Geol. 168, 120–128.
Kang, F., Xu, Q., Li, J., 2016. Slope reliability analysis using surrogate models via new support vector machines with swarm intelligence. Appl. Math. Model. 40 (11–12), 6105–6120.
Li, D.Q., Jiang, S.H., Cao, Z.J., Zhou, W., Zhou, C.B., Zhang, L.M., 2015. A multiple response-surface method for slope reliability analysis considering spatial variability of soil properties. Eng. Geol. 187, 60–72.
Liu, L., Zhang, S., Cheng, Y.M., Liang, L., 2019. Advanced reliability analysis of slopes in spatially variable soils using multivariate adaptive regression splines. Geosci. Front. 10 (2), 671–682.
Low, B.K., Tang, W.H., 1997. Reliability analysis of reinforced embankments on soft ground. Can. Geotech. J. 34 (5), 672–685.
Ma, G., Hu, X., Yin, Y., Luo, G., Pan, Y., 2018. Failure mechanisms and development of catastrophic rockslides triggered by precipitation and open-pit mining in Emei, Sichuan, China. Landslides 15 (7), 1401–1414.
Meng, J., Mattsson, H., Laue, J., 2021. Three-dimensional slope stability predictions using artificial neural networks. Int. J. Numer. Anal. Methods GeoMech. 45 (13), 1988–2000.
Pan, S.J., Yang, Q., 2009. A survey on transfer learning. IEEE Trans. Knowl. Data Eng. 22 (10), 1345–1359.
Pu, H., Gao, P., Liu, Y., Rong, J., Shi, F., Lu, H., 2019. Principal component analysis based dynamic cone beam x-ray luminescence computed tomography: a feasibility study. IEEE Trans. Med. Imag. 38 (12), 2891–2902.
Qi, C., Tang, X., 2018. A hybrid ensemble method for improved prediction of slope stability. Int. J. Numer. Anal. Methods GeoMech. 42 (15), 1823–1839.
Shlens, J., 2014. A tutorial on principal component analysis. arXiv preprint arXiv 1404, 1100.
Sloan, S.W., 1988. Lower bound limit analysis using finite elements and linear programming. Int. J. Numer. Anal. Methods GeoMech. 12 (1), 61–77.
Smith, L.I., 2002. A Tutorial on Principal Components Analysis. Technical Report OUCS-2002-12. University of Otago, New Zealand.
Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A survey on deep transfer learning. In: Proceedings of the International Conference on Artificial Neural Networks. Springer, Cham, pp. 270–279.
Ture, M., Kurt, I., Akturk, Z., 2007. Comparison of dimension reduction methods using patient satisfaction data. Expert Syst. Appl. 32 (2), 422–426.
Wang, H.B., Xu, W.Y., Xu, R.C., 2005. Slope stability evaluation using back propagation neural networks. Eng. Geol. 80 (3–4), 302–315.
Wang, L., Zhang, X., Tinti, S., 2021. Large deformation dynamic analysis of progressive failure in layered clayey slopes under seismic loading using the particle finite element method. Acta Geotech 1–14.
Willmott, C.J., Matsuura, K., 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Clim. Res. 30 (1), 79–82.
Wu, S.H., Ou, C.Y., Ching, J., Hsein Juang, C., 2012. Reliability-based design for basal heave stability of deep excavations in spatially varying soils. J. Geotech. Geoenviron. Eng. 138 (5), 594–603.

Zhao, H.B., 2008. Slope reliability analysis using a support vector machine. Comput. Geotech. 35 (3), 459–467.

Professor Sheng Zhang is the Deputy Head of School of Civil Engineering at Central South University in China. He received a B.Eng degree and a M.Eng degree in Civil Engineering from Hunan University in China, and a PhD in geotechnical Engineering from Nagoya Institute of Technology in Japan. Prof Zhang's research interests include unsaturated soils, thermo-hydro-mechanical coupling for geomaterials, and cold region geotechnics. He has published more than 100 peer-reviewed papers in top journals in these areas. He currently serves as an editorial board member for *Canadian Geotechnical Journal*, *Journal of Architecture and Civil Engineering*, and *Journal of Central South University*. He also serves as a committee member on the Chinese Institution of Soil Mechanics and Geotechnical Engineering (CISMGE), the International Society for Soil Mechanics and Geotechnical Engineering (ISSMGE), and the Japanese Geotechnical Society (JGS). Prof. Zhang has received the First prize of Hunan Technological Invention, the Best Paper Award of the International Association for Computer Methods and Advances in Geomechanics (IACMAG), Zhan Tianyou Railway Science and Technology Youth Award, National Outstanding Youth Funding, Hunan Outstanding Youth Funding, and Hunan Talents Gathering Project.

Dr. Chenxi Tong is an associate professor of Geotechnical Engineering of School of Civil Engineering at Central South University. He received a B.Eng degree and a M.Eng degree in Civil Engineering from Central South University, and a PhD in Civil Engineering from University of Technology Sydney in Australia. The major research interests of Dr. Chenxi Tong include constitutive modeling and advanced laboratory testing of granular soils. He has published more than 20 peer-reviewed papers in top journals in geotechnical engineering and also served as a reviewer for several journals.