Proc. of the 8th International Symposium on Geotechnical Safety and Risk (ISGSR)
Edited by Jinsong Huang, D.V. Griffiths, Shui-Hua Jiang, Anna Giacomini and Richard Kelly
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-18-5182-7_03-003-cd

ISGSR 2022

Locally Connected Neural Networks as Surrogate Models for Stochastic Analysis with Spatial Variability

Xuzhen He[1], Fang Wang[2], and Daichao Sheng[3]

[1]University of Technology Sydney, NSW, 2007, Australia
E-mail: xuzhen.he@uts.edu.au
[2]University of Sydney, NSW, 2006, Australia
E-mail: fang.wang@sydney.edu.au
[3]University of Technology Sydney, NSW, 2007, Australia
E-mail: daichao.sheng@uts.edu.au

Abstract: Using machine-learning models as surrogate models is a popular technique to increase the computational efficiency of stochastic analysis. In this technique, a smaller number of numerical simulations are conducted for a case, and obtained results are used to train machine-learning surrogate models specific for this case. This study presents a new framework using deep learning, where models are trained with a big dataset covering any soil properties, spatial variabilities, or load conditions encountered in practice. These models are very accurate for new data without re-training. So, the small number of numerical simulations and training process are not needed anymore, which further increases efficiency. The prediction of bearing capacity of shallow strip footings is taken as an example. More than 12000 data are used in training. It is shown that one-hidden-layer fully connected networks are ineffective for complex problems, where deep neural networks show a competitive edge, and a deep-learning model achieves a very high accuracy (the root-mean-square relative error is 3.1% for unseen data).

Keywords: bearing capacity, locally connected neural networks, surrogate models, stochastic analysis

1 Introduction

Due to uncertainties in soil properties, stochastic analysis has been increasingly employed in geotechnical risk assessment. Soil properties often vary spatially due to its sediment history and environmental evolution. Random field theory is used to account for both the uncertainty and spatial variability of soil properties (Griffiths and Fenton, 2001).

Brute-force stochastic analysis is often challenging when the sampling demand is very high, because querying the structural response via enormous numerical simulations is computational demanding. To overcome this challenge, one solution is to reduce sampling demand by using the subset simulation method (Beck and Au, 2002), which is unfortunately still expensive for most geotechnical applications (Huang *et al.*, 2017). Another popular solution is to replace time-consuming numerical simulations with surrogate models (Jiang and Huang, 2016), which are cheap to evaluate.

Training machine-learning models as surrogate models starts to gain popularity recently. Previous studies are mostly about using simple machine-learning methods like the supporting vector regression in simple problems involving only random variables (Kang, Xu and Li, 2016). When spatial variability is considered, the input size to machine-learning algorithms increases exponentially, and simple machine-learning tools will not be able to cope. The authors have used tools like the artificial neural networks in slope stability analysis with spatial variability (He *et al.*, 2020). In the machine-learning-aided stochastic analysis, a small number of numerical simulations are conducted. The random field samples and calculated outcomes of structural response (safety factor in slope stability analysis), respectively, are treated as input and output data, and are fed into machine-learning algorithms. The obtained machine-learning surrogate models are then used to evaluate structural response for a large number of random field samples. In this framework, a small number of numerical simulations are still required for each specific analysis.

The last decade has seen explosion in machine learning research and applications, because of two fundamental reasons: (i) availability of big data; (ii) deep neural networks and availability of powerful hardware to train these networks. Deep neural networks can perform very complex tasks. For example, convolutional neural networks (CNNs) have been successfully used in image classification, image and video recognition, mastering the game GO (AlphaGO) (Silver *et al.*, 2016), and predicting the stiffness of composite materials (Chen and Gu, 2020) etc. Recurrent neural networks are a powerful tool for time-series predictions, and have been successfully used in natural language processing, speech recognition, machine translation (Google Translate), and time-series predictions (Gao *et al.*, 2019), etc. In this paper, we present a new framework (deep-learning-aided stochastic analysis), in which deep-learning models are used as surrogate models. These deep-learning models are trained with a big dataset covering any soil properties, spatial variabilities, or load conditions encountered in practice. For a new dataset, the model is very accurate without re-training. Therefore, the small number of numerical simulations and training process are not needed anymore in stochastic analysis, which further increases computational efficiency.

2 Problem definition

We take the the prediction of bearing capacity of shallow strip footings as an example. For a strip footing with width B, its bearing capacity is the maximum load that can be placed on it before a bearing-capacity failure occurs (Figure 1). The bearing capacity ($\boldsymbol{q_u}$) is related to the soil strength (the cohesion $\boldsymbol{c}$ and friction angle $\boldsymbol{\phi}$ if modelled by the Mohr–Coulomb model), the unit weight of soil ($\boldsymbol{\gamma}$), and the overburden load ($\boldsymbol{q_0}$). Because these soil properties vary spatially, we can write a conceptual expression as

$$q_u = f[B, c(\boldsymbol{x}), \phi(\boldsymbol{x}), \gamma(\boldsymbol{x}), q_0] \tag{1}$$

where x = (x, y) s the position vector to indicate spatial dependence.

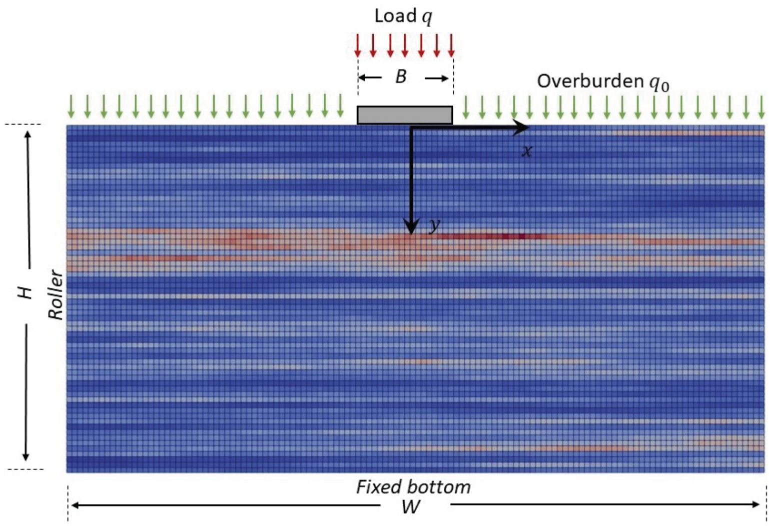


Figure 1. Illustration of a strip footing and the bearing capacity

By normalising all the quantities against a reference strength $\boldsymbol{c_r}$ a dimensionless equation is obtained. The mean of the cohesion field is a good choice for the reference strength.

$$q'_u = f[c'(\boldsymbol{x}'), \phi(\boldsymbol{x}'), \gamma'(\boldsymbol{x}'), q'_0] \tag{2}$$

where $\boldsymbol{q'_u}$ is the dimensionless bearing capacity. The dimensionless unit weight ($\boldsymbol{\gamma'}$) measures the relative significance of gravity and strength, the dimensionless overburden load ($\boldsymbol{q'_0}$) measures the relative significance of overburden and strength. $\boldsymbol{c'}$ is the dimensionless cohesion and $\boldsymbol{x'}$ is the dimensionless position vector.

The bearing capacity is mostly determined by the soil properties near the footing, soil properties far away are often negligible. Therefore, we limit the spatial dependence to a finite region (-6.4< x <6.4 0.0< y <6.4) in this study. It will be illustrated later that this finite-region dependence will give accurate results when $\boldsymbol{\phi}$ <25º and give conservative results (preferred in practice) when $\boldsymbol{\phi}$ > 25º.

Although the bearing capacity depends on continuous fields, in stochastic analysis, it is often calculated by a numerical model with a finite resolution. Generally, the continuous fields are mapped into finite mesh/grids/point-clouds with either a local average method or a midpoint method. We also only consider a finite resolution in this study, which is fine enough because the scale of fluctuation for the spatial variability of soil is greater than 0.2 m from Phoon and Kulhawy(1999), and so is greater than 0.2B (B= 0.2–1.0 m explained below). We thus have the following equation.

$$q'_u = f[\mathbf{C}', \boldsymbol{\Phi}, \boldsymbol{\Gamma}', q'_0] \tag{3}$$

where $\mathbf{C}'$, $\boldsymbol{\Phi}$, and $\boldsymbol{\Gamma}'$ are finite fields of the soil properties and are represented by matrices of size 64 × 128, which indicates the finite-region and finite-resolution dependence adopted in this study.

The goal of the present paper is to build a deep-learning model to approximate Eq. 3, which can be effectively used in stochastic analysis. We only need to cover soil properties and loads commonly encountered in practice to avoid building an over-complex model. Typical soil parameters are summarised by Phoon and Kulhawy(1999) and dimensionless parameters are shown in Table 1 (He *et al.*, 2021). The training data are with independent random fields.

Table 1. Dimensionless parameters to be covered by the deep-learning model

Parameters	Mean μ	COV v	l'_y	l'_x
c'	1	0–0.55		
ϕ	0–40º	0–0.15	0.2–∞	23–∞
γ'	0–2	0–0.1		
q'_0	0–6			

3 Data acquisition

Deep neural networks can perform very complex tasks but must be trained with big data. Otherwise, a complex model with very few data will only lead to "overfitting". The quality of data is also important and data with less noise will give better models. In this study, for each input, we will estimate the output by three different numerical methods for cross validation and for improved data quality.

Finite-difference quasi-static (FDQS) method: This is conducted in the commercial software FLAC3D, which uses a 3D explicit finite-difference scheme. To provide quasi-static solutions, the equations of motion are damped by a combined damping method with a damping constant of 0.95.

Finite-element static (FES) method: This is conducted in the commercial software ABAQUS. Elements with linear shape functions and reduced integration are used. The total stiffness approach is used to suppress the hourglass modes with default hourglass stiffness values. General static analysis is conducted. Since the bearing-capacity problem is unstable, especially when the friction angle is large, viscous forces are added to the global equilibrium equations for automatic stabilisation, and the damping factor is calculated based on a dissipated energy fraction (0.0002).

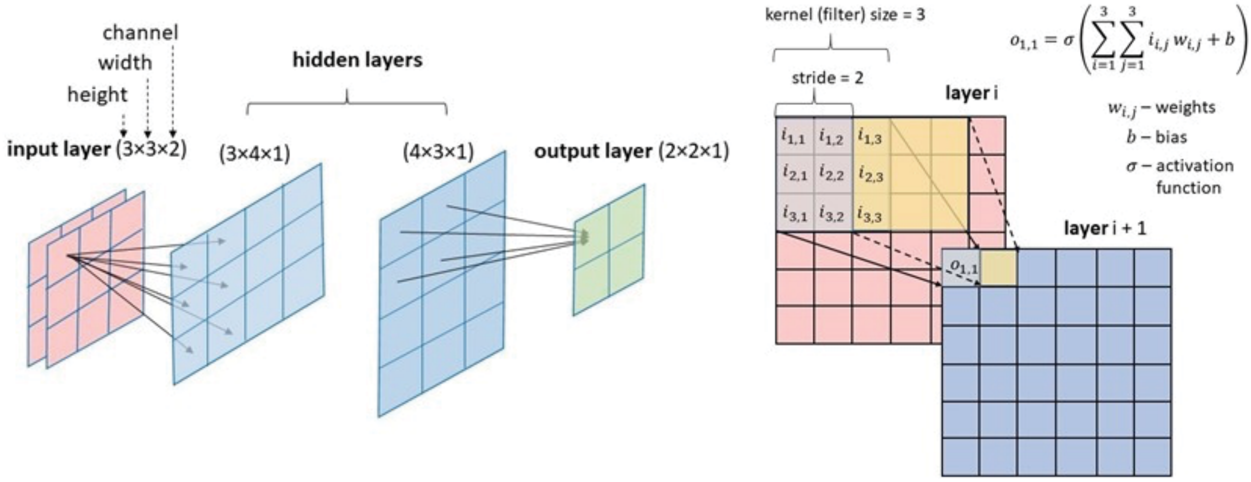
Finite-element quasi-static (FEQS) method: This is also conducted in ABAQUS but solved by an implicit dynamic scheme for quasi-static problems.

In all three types of numerical models, the footing width is 1 m, the simulation domain is 6.4 m by 12.8 m, which is discretised into 64 × 128 quadrilateral/hexahedron elements (Figure 1). The left and right boundaries are smooth, while the bottom boundary is fixed. The soil elements are modelled by the Mohr–Coulomb model. In each simulation, the footing is incrementally displaced vertically into the soil. A relatively large elastic modulus is used, so most simulations will reach bearing-capacity failure when the displacement is between 0.004B and 0.08B. Each simulation will stop after the maximum load does not change for a further displacement of 0.01B. The numerical models are validated against analytical solutions, and all three numerical methods give accurate results (He *et al.*, 2021).

4 Deep neural networks

Artificial neural networks are a computing model inspired by the biological neural networks that constitute animal brains (Anderson, 1995). Their structure is a collection of connected nodes (the squares in Figure 2a), which are often called artificial neurons or simply neurons. These artificial neurons retain the biological concept of neurons, which receives a signal, then processes it and signals the neurons connected to it. The connections between neurons are typically called "edges" (arrows in Figure 2a). A given neuron can have multiple input and output connections. "Signal"s are real numbers, which are transmitted from the input layer, through several hidden layers and reach the output layer (Figure 2a). A deep neural network is a network with multiple hidden layers. Because the inputs for the bearing-capacity problem are mostly matrices with identical sizes (64 × 128), each layer in this study is arranged to have neurons in 3 dimensions (height, width, and channels) like that in convolutional neural networks (CNNs). CNNs are mostly applied to analysing visual imagery, so the height and width are the image pixel sizes, and the channel corresponds the colour channels such as the RGB values. The size of each layer is represented by a tuple (height × width × channels).

To transmit signals, each edge is assigned a weight that represents its relative importance (w in Figure 2b). To obtain the signal on a neuron, a weighted sum of signals from its predecessor neurons is firstly calculated and a bias (b in Figure 2b) is added to the sum, which is then combined with this neuron' internal state by using an activation function (σ in Figure 2b). Neural networks use non-linear activation functions to learn complex data, compute and learn almost any function, and provide accurate predictions. The rectified linear unit (ReLU) is exclusively used as activation functions in this study. Machine learning algorithms must have associated learning strategies to automatically adjust ("train") the parameters (weights and bias) to improve prediction accuracy. Stochastic gradient descent is a popular method and is used in this study.



(a) (b)

Figure 2. Illustration of deep neural networks

Different types of layers have been created to accommodate different need and data structure:

Input layer: The very first layer, characterised by a size tuple (height × width × channels) in this study.

Output layer: The output of neural networks. The bearing capacity in this study – a scalar or size = (1 × 1 × 1).

Fully connected (FC) layer: Each neuron is connected with all neurons in the previous layer, and this layer type is characterised by a size tuple (height × width × channels). Due to the dense connectivity, this layer type involves many trainable parameters even for a small layer size.

Locally connected (LC) layer: As the name suggests, each neuron is locally connected to a small region (a cuboid) in the previous layer; the cuboid fully spans the channel direction. These connections are often called kernels or filters, which mimic the neurons in visual cortexes – each neuron responds to only a small region of visual field. Because padding is not used in this study, a LC layer is characterised by three parameters: the filter size, the stride, and the number of channels. The filter size is the size of corresponding neurons in the previous layer (Figure 2b). The stride is the sliding distance in the previous layer when neurons slide by 1 in the next layer (Figure 2b). Both the filter size and stride have two dimensions and are characterised by a size tuple (height × width). The height and width of the next layer is totally determined by the height and width of the previous layer, filter size, and stride. Due to the local connectivity, this layer type reduces the number of trainable parameters compared with fully connected layers.

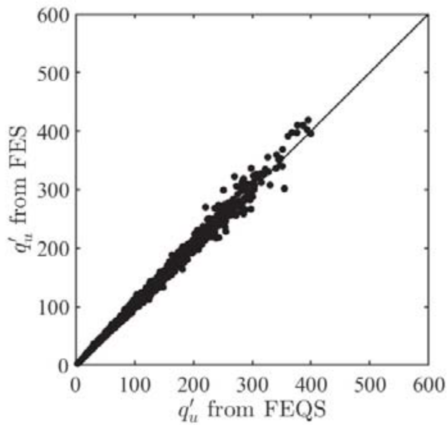
Convolutional (Conv) layer: This layer type further reduces the number of trainable parameters by assuming that if one filter is useful at some spatial position, then it should also be useful at a different position. So, all filters in the same channel should share the same trainable parameters (parameter sharing). The forward pass is then computed as a convolution of the filter's weights with the input cuboid. This is where the name Conv layer comes from.

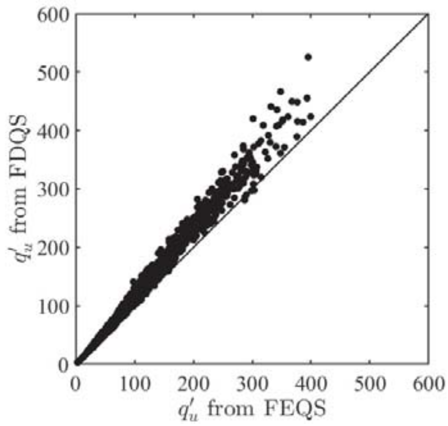
Pooling layers: Pooling layers are often inserted after Conv layers to progressively reduce the spatial size, to reduce the number of parameters and computation, and hence to control overfitting. They often have the stride equal to the filter size and denoted as a window size in this study. The most common form is with a filter size of 2 × 2 and a stride of 2 × 2, which will discard 75% of the information from the previous layer. Pooling layers do not compute like Figure 2b with weights or bias but undertake a maximum operation over the input cuboid (MaxPool) or an average operation (AvgPool).

Dropout layer: Deep neural networks are very likely to overfit with a small training dataset. The dropout layers randomly set signals to 0 from the previous layer with a frequency of dropout rate. This will make the training process noisy, forcing neurons within a layer to probabilistically take on more responsibility for the inputs.

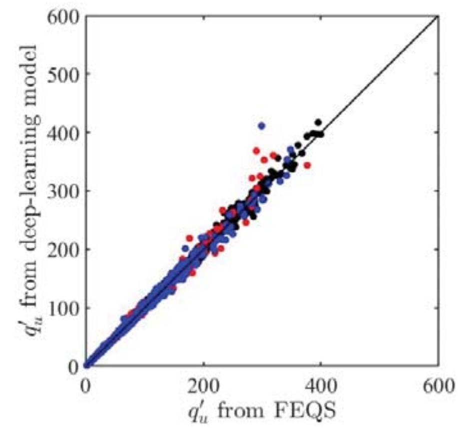
Deep neural networks with stacks of these layer types can perform very complex tasks. For example, the GoogLeNet (Szegedy *et al.*, 2015) is a powerful image classification model consisting of 27 layers (mainly Conv and MaxPool layers). AlphaGo (Silver *et al.*, 2016) is a model that winning human experts in the game GO, mainly consisting of Conv layers. DeepFace (Taigman *et al.*, 2014) is a face recognition model that outperforms human experts, which consists of 2 Conv, 1 MaxPool, 1 FC, and three LC layers.

5 Results and conclusions





(a) (b)



(c) with the final locally connected neural network (black = 8621 training data; red = 1847 validation data, blue = 1847 test data)

Figure 3. Bearing capacity of the full problem predicted by different methods (12 326 data)

The full inputs are three matrices and a scalar. This problem is characterised by 13 parameters: the mean, COV, l'_y, and l'_y, for each random field, and the dimensionless load. 12 326 data are prepared for training. The full problem is relatively complex, and the dimensionless bearing capacity can reach up to 400. Similarly, The FES and FEQS give consistent results with RMSRE(FES→FEQS) = 2.84% (Fig.3a). Due to the different damping methods used, the FDQS gives slightly larger bearing capacity and RMSRE(FDQS→FEQS) = 9.44% (Fig.3b).

A fully connected network achieves an average error of 21.3%. With such a large error, this model is unacceptable to be used as a surrogate model. A locally connected neural network is built with the architecture as explained in He *et al.* (2021), which achieves a RMSRE of only 3.3% (Fig.3c).. Therefore, deep neural networks can be used for very complex tasks and have a competitive edge over the one-layer fully connected networks in these tasks. As for the bearing-capacity problem, we train a very powerful model, which can then be applied to any soil properties, spatial variabilities, or load conditions encountered in practice and is particularly very accurate (3.3% error) – making it an ideal surrogate model for stochastic analysis.

The use of the deep-learning model in stochastic analysis is tested in some examples (He *et al.*, 2021).. This model is proven very accurate if the parameters of specific problem is well in the defined limits. Large errors can be observed when outside the limits due to extrapolation. It is also shown that because deep learning is a datadriven tool, its capability can be extended by simply generating more representative data outside the current limits and retrain the models.

References

Anderson, J. A. (1995) An Introduction To Neural Networks. *MIT Press*.

Beck, J. L. and Au, S.-K. (2002) 'Bayesian Updating of Structural Models and Reliability using Markov Chain Monte Carlo Simulation', *Journal of Engineering Mechanics*, 128(4), pp. 380–391. doi: 10.1061/(asce)0733-9399(2002)128:4(380).

Chen, C. T. and Gu, G. X. (2020) 'Generative Deep Neural Networks for Inverse Materials Design Using Backpropagation and Active Learning Support', *Advanced Science*, 7(5). doi: 10.1002/advs.201902607.

Gao, X. *et al.* (2019) 'Recurrent neural networks for real-time prediction of TBM operating parameters', *Automation in Construction*, 98(October 2018), pp. 225–235. doi: 10.1016/j.autcon.2018.11.013.

Griffiths, D. V. and Fenton, G. A. (2001) 'Bearing capacity of spatially random soil: the undrained clay Prandtl problem revisited', *Géotechnique*, 51(4), pp. 351–359. doi: 10.1680/geot.51.4.351.39396.

He, X. *et al.* (2020) 'Machine learning aided stochastic reliability analysis of spatially variable slopes', *Computers and Geotechnics*, 126(June), p. 103711. doi: 10.1016/j.compgeo.2020.103711.

He, X. *et al.* (2021) 'Deep learning for efficient stochastic analysis with spatial variability', *Acta Geotechnica*, 1. doi: 10.1007/s11440-021-01335-1.

Huang, J. *et al.* (2017) 'On the efficient estimation of small failure probability in slopes', *Landslides*, 14(2), pp. 491–498. doi: 10.1007/s10346-016-0726-2.

Jiang, S. H. and Huang, J. S. (2016) 'Efficient slope reliability analysis at low-probability levels in spatially variable soils', *Computers and Geotechnics*, 75, pp. 18–27. doi: 10.1016/j.compgeo.2016.01.016.

Kang, F., Xu, Q. and Li, J. (2016) 'Slope reliability analysis using surrogate models via new support vector machines with swarm intelligence', *Applied Mathematical Modelling*, 40(11–12), pp. 6105–6120. doi: 10.1016/j.apm.2016.01.050.

Phoon, K.-K. and Kulhawy, F. H. (1999) 'Characterization of geotechnical variability', *Canadian Geotechnical Journal*, 36(4), pp. 612–624. doi: 10.1139/t99-038.

Silver, D. *et al.* (2016) 'Mastering the game of Go with deep neural networks and tree search', *Nature*, 529(7585), pp. 484–489. doi: 10.1038/nature16961.

Szegedy, C. *et al.* (2015) 'Going deeper with convolutions', *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

Taigman, Y. *et al.* (2014) 'DeepFace: Closing the Gap to Human-Level Performance in Face Verification', in *2014 IEEE Conference on Computer Vision and Pattern Recognition. IEEE*, pp. 1701–1708. doi: 10.1109/CVPR.2014.220.