University Library

MINERVA ACCESS

A gateway to Melbourne's research publications

Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:
Nguyen, T;Kashani, A;Ngo, T;Bordas, S

Title:
Deep neural network with high-order neuron for the prediction of foamed concrete strength

Date:
2019-04-01

Citation:
Nguyen, T., Kashani, A., Ngo, T. & Bordas, S. (2019). Deep neural network with high-order neuron for the prediction of foamed concrete strength. Computer Aided Civil and Infrastructure Engineering, 34 (4), pp.316-332. https://doi.org/10.1111/mice.12422.

Persistent Link:
https://hdl.handle.net/11343/284819

Deep neural network with high order neuron for the prediction of foamed concrete strength

Tuan Nguyen, Alireza Kashani, Tuan Ngo*

Department of Infrastructure Engineering, University of Melbourne, Melbourne, Australia

&

Stéphane Bordas

Institute of Computational Engineering, University of Luxembourg, Luxembourg

Author Manuscript

Abstract: *The paper presents a deep neural network model for the prediction of the compressive strength of foamed concrete. A new, high order neuron was developed for the deep neural network model to improve the performance of the model. Moreover, the cross-entropy cost function and rectified linear unit activation function were employed to enhance the performance of the model. The present model was then applied to predict the compressive strength of foamed concrete through a given dataset, and the obtained results were compared with other machine learning methods including conventional artificial neural network (C-ANN) and second order artificial neural network (SO-ANN). To further validate the proposed model, a new dataset from the laboratory and a given dataset of high performance concrete were used to obtain a higher degree of confidence in the prediction. It is shown that the proposed model obtained a better prediction, compared to other methods. In contrast to C-ANN and SO-ANN, the proposed model can genuinely improved its performance when training a deep neural network model with multiple hidden layers. A sensitivity analysis was conducted to investigate the effects of the input variables on the compressive strength. The results indicated that the compressive strength of foamed concrete is greatly affected by density, followed by the water-to-cement and sand-to-cement ratios. By providing a reliable prediction tool, the proposed model can aid researchers and engineers in mixture design optimisation of foamed concrete.*

1 INTRODUCTION

Foamed concrete is a type of lightweight concrete that offers many advantages such as reducing the dead load of structures, improving sound and thermal insulation properties, and reducing transportation and installation costs (Ngo et al., 2017; Hajimohammadi et al., 2017a; Nguyen et al., 2018). These advantages make foamed concrete suitable for a myriad of applications in infrastructure construction and the prefabricated industry such as prefabricated blocks or panels, insulated walls, aircraft runway arrestors, roof decks and geotechnical backfills. The use of foamed concrete poses a new challenge to perform its mixture design optimisation which requires an accurate and quick understanding of the relationship between its properties and mixture proportions. Among the properties of foamed concrete, compressive strength is the most important in terms of its mechanical performance. Numerous studies have revealed that the compressive strength of foamed concrete decreases when its density decreases (Hajimohammadi et al., 2017b,c). Several researches have conducted studies to predict the compressive strength of foamed concrete based on mixture proportions. Most studies are empirical, in which empirical equations were developed by calibrating experimental data as discussed in the following section. Predicting the compressive strength of foamed concrete using empirical equations is usually limited by a range of input conditions. Therefore, these predictions are generally extrapolative in practice (Nehdi et al., 2001). Moreover, such semi-analytical models required the determination of empirical constants which are not easy to obtain to describe such complex relationships between mixture proportions and the compressive strength (Chiew et al., 2017). Therefore, there is a significant need for the development of an advanced prediction tool.

This paper presents a deep neural network (DNN) model, which is a subset of Machine Learning (ML), for the pre-

This is the author manuscript accepted for publication and has undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process, which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1111/mice.12422

This article is protected by copyright. All rights reserved.

Author Manuscript

diction and understanding of foamed concrete strength. ML and data science has shown great potential for predicting, designing and discovering materials (Ley and Bordas, 2018). In civil engineering and construction, ML has been extensively used a variety of applications such as structural heal monitoring (Rafiei and Adeli, 2017, 2018; Xue and Li, 2018; Gao and Mosalam, 2018), reliability analysis (Dai and Cao, 2017; Nabian and Meidani, 2018; Grande et al., 2017), transportation (Dharia and Adeli, 2003; Zhang and Ge, 2013; García-Ródenas et al., 2017; Yu et al., 2018), prediction and estimation (Adeli and Wu, 1998; Zhao and Ren, 2002; Chou and Pham, 2013; Rafiei et al., 2017a). In concrete-related studies, DeRousseau et al. (2018) recently reviewed the application of ML to optimise mixture design of concrete. This study suggested that machine learning methods are powerful tools for mixture design optimisation because they are able to account for the complexity of mixtures and objective functions. Rafiei et al. (2016) comprehensively reviewed different ML approaches to estimate the properties of concrete. Using a deep restricted Boltzmann machine, Rafiei et al. (2017b) presented a mode to predict the compressive strength of concrete from 103 concrete test data retrieved from from the machine learning repository of the University of California, Irvine. The estimated results by the deep restricted Boltzmann machine were compared with ANN and support vector machine (SVM) models. Combining artificial firefly algorithm and SVM method, Chou and Pham (2015) proposed a framework for predicting the compressive strength of normal concrete and high strength concrete. Abd and Abd (2017) predicted the compressive strength of foamed concrete using a traditional multi-variable nonlinear regression and revolutionary SVM. The results show a good correlation between the actual compressive strength and predicted compressive strength. A model to predict the compressive strength of foamed concrete based on extreme learning machine (ELM) was proposed in Yaseen et al. (2018). The ELM model was subsequently validated in comparison with M5 Tree and SVM methods.

Recently, among different ML techniques, artificial neural network (ANN) has been widely used in civil engineering (Adeli and Panakkat, 2009; Panakkat and Adeli, 2009; Wang and Adeli, 2015). ANN was originally inspired by human brains in which information is transmitted and processed by biological neurons to build complicated concepts and ideas (Adeli, 2001). The architecture of conventional ANN consists of a input layer, a hidden layer and a output layer. The connection of neurons from one layer to others was modelled by weights and biases through the linear inner product and activation function (Bui et al., 2018). Sigmoid function is the most prevalent activation function in the conventional ANN model (C-ANN). To quantify the performance of the C-ANN, quadratic cost function is commonly used. Many advantages of the C-ANN were demonstrated for the prediction of concrete strength in the literature. For example, Naderpour et al. (2018) employed C-ANN with the sigmoid activation function and quadratic cost function to predict the compressive strength of recycled aggregate concrete. To do so, 139 concrete test data was retrieved from the literature to train and test the C-ANN model. Based on C-ANN approach, Sarıdemir (2009) developed a prediction tool for the compressive strength of concrete containing metakaolin and silica fume. 33 different mixtures with 195 specimens were collected from the literature to construct the model. The compressive strength of expanded polystyrene (EPS) lightweight concrete was predicted by ANN models in Sadrmomtazi et al. (2013). For foamed concrete strength, Nehdi et al. (2001) used the C-ANN model with eight hidden neurons (due to computational limitation) to forecast the strength of foamed concrete.

Despite many advantages, the C-ANN model suffers from several drawbacks such as learning slow-down problem due to the quadratic cost function and vanishing gradient problem due to the sigmoid activation function (Goodfellow et al., 2016; Nielsen, 2015). Moreover, it is known that the C-ANN only performs linear calculation between neurons in a layer through the inner product (Fan et al., 2018). This forms a hypothesis that the inner product (i.e. linear neuron) in the C-ANN can be replaced by a quadratic function (second order neuron) as presented in Fan et al. (2018). It is showed by Fan et al. (2018) that the new second order artificial neural network (SO-ANN) can successfully work for nonlinear function such as XOR gate and concentric rings which are unable to capture by the C-ANN with the linear neuron. Fan et al. (2018) concluded that the performance of the second order model would be better than the C-ANN with the linear neuron. In fact, Roberts and Attoh-Okine (1998) alternatively used a simple quadratic ANN model to predict the International Roughness Index (IRI) for pavement assessment. Even though the quadratic model in Roberts and Attoh-Okine (1998) is quite simple and different to the one in Fan et al. (2018), the quadratic model can capture the nonlinearity of this problem very well. It showed that the quadratic ANN model outperformed the linear ANN model in predicting the IRI.

Motivated by the literature, this study presents the deep neural network with high order neuron to predict the compressive strength of foamed concrete. Inspiring from the second order model presented in Fan et al. (2018), the paper proposes a general, high order neuron for the deep neural network model (HO-DNN). Moreover, the cross-entropy cost and rectified linear unit (ReLU)activation functions are utilised to address the learning slow-down and vanishing gradient problems. The paper is the first attempt to investigate the performance of HO-DNN model for real engineering applications and with deep neural network. It is noticeable that Fan et al. (2018) only validated their SO-ANN model with

This article is protected by copyright. All rights reserved.

Author Manuscript

mathematical functions (e.g. XOR, NAND and NOR gates) and a shallow neural network with one hidden layer. The developed model was validated by predicting foamed concrete strength through a given dataset. The obtained results were compared with other methods including C-ANN, SO-ANN. The model was further validated by a completely new dataset from the laboratory and another given dataset of high performance concrete to obtain a higher level of validation. Having a reliable model, a sensitivity analysis was then performed to determine the effects of the input variables on the compressive strength of foamed concrete.

2 EMPIRICAL EQUATIONS

This section briefly describes some empirical models reported in the literature. Interested readers are encouraged to refer to original publications for the detailed derivation of these models. Most empirical equations for the prediction of the compressive strength of foamed concrete are commonly based on three fundamental models, namely Balshin's, Feret's and Power's models (Neville, 2012). Balshin's model, which is known as the strength-porosity model (Kiani et al., 2016), assumes that the compressive strength of concrete is affected by the volume of air voids in concrete (interlayer pores/spaces, gel pores, capillary pores and entrapped air voids). Based on the strength-porosity model, Hoff (1972) firstly proposed an empirical equation as follows:

$$\sigma_y = \sigma_0 \left[\frac{d_c\,(1+0.2\rho_c)}{(1+k)\,\rho_c \gamma_w} \right]^b \tag{1}$$

in which σ_y is the compressive strength of foamed concrete; σ_0 is the theoretical compressive strength of cement paste at absolute zero porosity; d_c is the density of foamed concrete; k is the water-to-cement ratio (by weight); ρ_c is the specific gravity of ordinary Portland cement which is 3.15 as given in Hoff (1972) and γ_w is the unit weight of water. It is noted that σ_0 is impossible to obtain in practice, therefore σ_0 and b are empirical constants that are determined by calibrating the model with experimental data. Based on the same approach, Nambiar and Ramamurthy (2008) established an empirical equation which takes into account the presence of sand as follows:

$$\sigma_y = \sigma_0 \left[\frac{d_c\,(1+0.2\rho_c+s_c^v)}{(1+k)\,(1+s_c^w)\,\rho_c \gamma_w} \right]^b \tag{2}$$

in which s_c^w is the sand-to-cement ratio by weight and s_c^v is the sand-to-cement ratio by volume, which is calculated by $s_c^v = s_c^w \dfrac{\rho_c}{\rho_s}$; $\rho_c = 3.13$ and $\rho_s = 2.52$ are the specific gravity of cement and sand as given in Nambiar and Ramamurthy (2008). Similar to Hoff's model, σ_0 and b are empirical constants that require calibration.

Whilst Balshin's model is based on the porosity and the weights of the constituents, in Feret's model, the compressive strength of foamed concrete is governed by the absolute volume of its constituents as follows:

$$\sigma_y = K \left(\frac{c}{c+w+a} \right)^n \tag{3}$$

in which c, w and a are the absolute volumes of cement, water and air in the mixtures; k and n are empirical constants. Following Feret's model, Tam et al. (1987) calibrated Eq. (3) to predict the compressive strength of foamed concrete with $K = 5350$ and $n = 3.96$.

Power's models link the compressive strength of foamed concrete to the gel-space ratio as follows:

$$\sigma_y = k(g)^n \tag{4}$$

in which k is the intrinsic strength of the gel; n is an empirical constant and g is the gel-space ratio. In Nambiar and Ramamurthy (2008), the formulation of the gel-space ratio (g) was derived as follows:

$$g = \frac{2.06\alpha V_c}{1 - V_{fl} - V_c\,(1-\alpha)} \tag{5}$$

in which V_c is the volume of cement; V_{fl} is the volume of the fillers and α is the hydration parameter, which was assumed to be 0.8 (Nambiar and Ramamurthy, 2008). As porosity (or density) is the most important factor in foamed concrete strength, the strength-porosity models are prevalent in empirical equations for foamed concrete strength (Nguyen et al., 2017). Therefore, two empirical equations based this model (Hoff, 1972; Nambiar and Ramamurthy, 2008) are used in this paper.

3 DEEP NEURAL NETWORK WITH HIGH ORDER NEURON (HO-DNN)

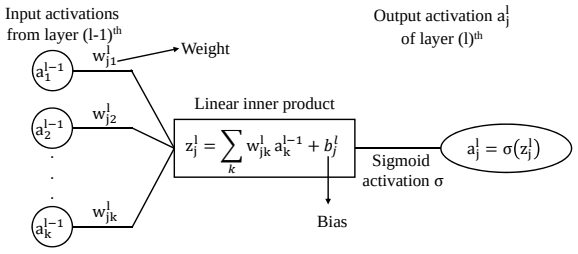
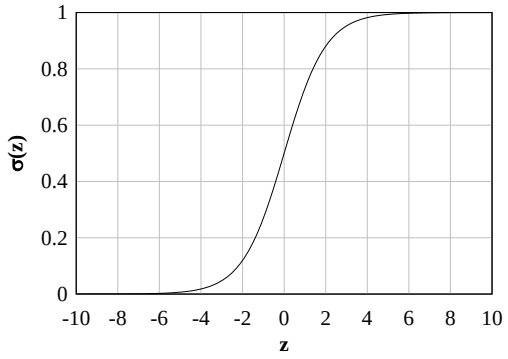


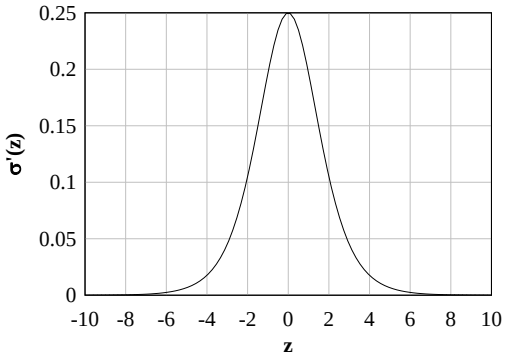
Figure 1: The structure of a linear neuron in C-ANN model.

This section presents the development of the deep neural network for the prediction of foamed concrete strength with the high order neuron, cross-entropy cost function, ReLU activation function. In this paper, the stochastic gradient decent-backpropagation (SGD-BP) algorithm is used to train the model because it is very effective for training neural network models. The SGD-BP algorithm only calculates and stores the gradient vector (first order) and does not require a Hessian matrix (second order). Therefore the algorithm and its

This article is protected by copyright. All rights reserved.



(a)



(b)

Figure 2: The sigmoid function (a) and its derivative (b).

variants can deal with any general engineering problems, particularly deep neural networks. The detailed description of the SGD-BP algorithm can be found in the literature (Goodfellow et al., 2016; Nielsen, 2015).

3.1 Conventional Artificial Neural Network (C-ANN)

Prior to the development of HO-DNN, the conventional artificial neural network (C-ANN) is quickly re-examined in this section. The structure of a neuron in C-ANN is depicted in Fig. 1. In C-ANN, input activations from the layer $(l-1)^{th}$ are transmitted to a neuron in the next layer by the linear inner product as depicted in Fig. 1. The weighted input z_j^l (Fig. 1) is then converted to the output activation a_j^l by the sigmoid activation function σ. To estimate the performance of C-ANN, the quadratic cost function $C_{quadratic}$ is commonly used in the literature as follows:

$$C_{quadratic} = \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{a}^L\right\|^2 \tag{6}$$

where $\boldsymbol{y}$ is the vector of actual outputs corresponding to an input x and $\boldsymbol{a}^L$ is the vector of activations of the last (output) layer L when x is input. Using the SGD-BP algorithm and the chain rule, the C-ANN model can be trained by propagating backward errors at each layer, which are defined as follows (Nielsen, 2015):

$$\delta_j^L = \frac{\partial C_{quadratic}}{\partial z_j^L} = \frac{\partial C_{quadratic}}{\partial a_j^L}\sigma'\left(z_j^L\right) = \left(y_j - a_j^L\right)\sigma'\left(z_j^L\right) \tag{7}$$

for the last layer L and

$$\delta_k^l = \frac{\partial C_{quadratic}}{\partial z_k^l} = \sum_j w_{jk}^{l+1}\delta_j^{l+1}\sigma'\left(z_k^l\right) \tag{8}$$

for layer l ($l = 1,2\ldots,(L-1)$). Following the SGD-BP algorithm, the weights and biases of C-ANN model (as illustrated in Fig. 1) are updated to improve the performance of C-ANN (Nielsen, 2015; Goodfellow et al., 2016).

However, Eq. (7) and Eq. (8) causes the learning slow-down and vanishing gradient problems of C-ANN as discussed in Section 1. In fact, due to its characteristic, the derivative of sigmoid function gets very small when the sigmoid function (i.e. output activation) is close to zero or one as depicted in Fig. 2. Therefore, the error δ_j^L in Eq. (7) might be very small even though C-ANN does not make a good prediction (i.e. $y_j - a_j^L$ is large). However, it is expected that a prediction tool is able to learn faster when it is decisively wrong. This phenomenon is referred to the learning slow-down problem of C-ANN model (Nielsen, 2015; Goodfellow et al., 2016). Moreover, the derivative of the sigmoid function varies in range of $[0, 0.25]$ as shown in Fig. 2b. Consequently, propagating backward the errors following Eq. (8) with the sigmoid function results in smaller and smaller errors in the earlier layers. This implies that weights and biases in the earlier layers are adjusted slower, which does not improve the learning process (Goodfellow et al., 2016; Nielsen, 2015). This phenomenon is known as the vanishing gradient problem, one of the main problems of training deep neural networks with the C-ANN (Schmidhuber, 2015).

3.2 The development of HO-DNN

The HO-DNN is developed in this section taking into account the limitation of C-ANN model discussed in the previous section. Firstly, the HO-DNN in this study is developed based on a new, high order neuron, which is motivated by the second order neuron presented in Fan et al. (2018). The structure of the high order neuron is presented in Fig. 3 in which the weighted input z_j^l is expressed as follows:

$$z_j^l = \sigma_1\left(z_j^{l,1}\right)\sigma_2\left(z_j^{l,2}\right) + \sigma_3\left(z_j^{l,3}\right) \tag{9}$$

where σ_1, σ_2 and σ_3 are the activation function 1, 2 and 3 in Fig. 3, which can be selected to account for nonlinear relationship between the components of the input activations, thereby building up a more complex and abstract model. Despite the fact that the HO-DNN model has triple the parameters as the C-ANN, this does not present a considerable computational difficulty. The new, high order neuron is more flexible and can cover the second order neuron proposed by Fan et al. (2018) as a special case with $\sigma_{1,2,3}$ is the linear function. Moreover, the new neuron can take into account higher nonlinear approximations by choosing $\sigma_{1,2,3}$ as any nonlinear functions. It is also worth noting that Fan et al.

Author Manuscript

This article is protected by copyright. All rights reserved.

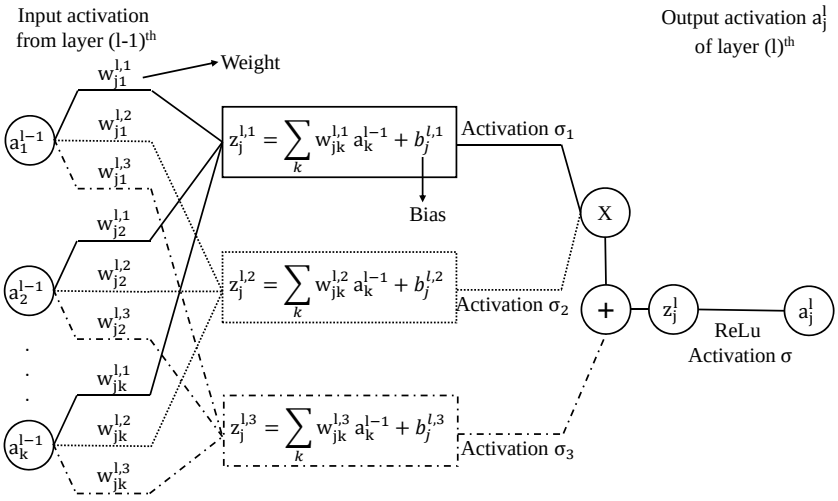


Figure 3: The structure of a neuron in the proposed HO-DNN model.

(2018) used the sigmoid function as an activation (from the weighted input z_j^l to the output activation a_j^l, Fig. 3) and the quadratic cost function for the SO-ANN model , which indicates that the SO-ANN model still suffers from the learning slow-down and vanishing gradient problems.

Secondly, the cross-entropy cost function is used to address the learning slow-down problem in this study as follows:

$$C_{CE} = \sum_j y_j \ln a_j^L + (1-y_j)\ln\left(1-a_j^L\right) \tag{10}$$

Similar to Eq. (6), y_j is the actual output corresponding to an input x and a_j^L is the activation of the last (output) layer L when x is input. Although the HO-DNN is fairly complicated with the new neuron and cross-entropy cost function, it is straightforward to implement the SGD-BP for training the model. Indeed, following the SGD-BP algorithm and using the chain rule, the error for the last layer L in the HO-DNN is given as follows:

$$\delta_j^L = \frac{\partial C_{CE}}{\partial z_j^L} = \left(y_j - a_j^L\right) \tag{11}$$

Eq. (11) shows that the rate at which the HO-DNN learns (or updates its weights and biases) is solely governed by $\left(y_j - a_j^L\right)$ and the effect of derivative $\sigma'\left(z^L\right)$ is eliminated, compared to Eq. (7) of the C-ANN. Therefore, the HO-DNN directly learns from the error between its prediction and actual output. Hence, the cross-entropy cost function can accelerate the training process of the HO-DNN and help the HO-DNN achieve better performance.

The error for the layer l^{th} $(l = 1, 2\ldots,(L-1))$ is given as follows:

$$\delta_k^l = \sum_j \begin{bmatrix} w_{jk}^{l+1,1}\sigma'_1\left(z_j^{l+1,1}\right)\sigma_2\left(z_j^{l+1,2}\right) \\ +w_{jk}^{l+1,2}\sigma'_2\left(z_j^{l+1,2}\right)\sigma_1\left(z_j^{l+1,1}\right) \\ +2w_{jk}^{l+1,3}\sigma\left(z_k^l\right)\sigma'_3\left(z_j^{l+1,3}\right) \end{bmatrix} \delta_j^{l+1}\sigma'\left(z_k^l\right) \tag{12}$$

The detailed derivation of Eq. (12) is given in the Appendix A.

Lastly, to overcome the vanishing gradient problem, the rectified linear unit (ReLU) activation function is used in the HO-DNN as depicted in Fig. 3. The ReLU activation is given as follows:

$$\sigma(z) = \max(0, z) \tag{13}$$

Although the ReLU activation looks simple and linear, the ReLU is genuinely a nonlinear approximation. In fact, the ReLU activation is prevalent in deep learning researches due to the advantage of its derivative:

$$\sigma'(z) = \begin{cases} 0 & \text{if} \quad z \le 0 \\ 1 & \text{if} \quad z > 0 \end{cases} \tag{14}$$

As mentioned in Section 3.1, the SGD-BP algorithm works by propagating backward the errors from the last layer L to earlier layers $(L-1), \ldots, 2, 1$ and the range of $[0, 0.25]$ of the derivative of sigmoid activation creates the vanishing gradient problem. As shown in Eq. (14), the derivative of ReLU is either one or zeros, which alleviate the vanish gradient problem. It is worth mentioning that the ReLU still have its problem, a so-called dying ReLU problem. The dying ReLU problem occurs when a neuron is inactive and no longer useful during training process (Goodfellow et al., 2016). The dying ReLU activation can be solved by the leaky ReLU activation (Karpathy, 2018). However, it is found that the ReLU activation performs effectively in this study. Therefore, the ReLU is used in this paper for the development of HO-DNN. The activation σ_1, σ_2 and σ_3 (Fig. 3) are chosen as the linear function. Experimentally, it will be demonstrated that this provides the best prediction of the compressive strength of foamed concrete and high performance concrete in this study. However, it is believed that different applications might require different selection of activations in

Author Manuscript

This article is protected by copyright. All rights reserved.

the HO-DNN. Therefore, it is important to have the general framework to assure the versatility of the HO-DNN. It is important to note that the sigmoid function is used at the output layer because of an intrinsic issue of using the ReLU activation with the cross-entropy cost function (Nielsen, 2015). Following the SGD-BP algorithm (Nielsen, 2015), weights and biases of the HO-DNN model are updated as follows:

$$\begin{aligned} w_{jk}^{l,i} &\rightarrow w_{jk}^{l,i} - \eta \frac{\partial C_{CE}}{\partial w_{jk}^{l,i}} \quad \text{for} \quad i = 1,2,3 \\ b_{j}^{l,i} &\rightarrow b_{j}^{l,i} - \eta \frac{\partial C_{CE}}{\partial b_{j}^{l,i}} \quad \text{for} \quad i = 1,2,3 \end{aligned} \tag{15}$$

where η is a learning rate, which is determined in Section 5; $\frac{\partial C_{CE}}{\partial w_{jk}^{l,i}}$ and $\frac{\partial C_{CE}}{\partial b_{j}^{l,i}}$ are given as follows:

$$\frac{\partial C_{CE}}{\partial b_{j}^{l,1}} = \delta_j^l \sigma_1' \left(z_j^{l,1}\right) \sigma_2 \left(z_j^{l,2}\right) \tag{16a}$$

$$\frac{\partial C_{CE}}{\partial b_{j}^{l,2}} = \delta_j^l \sigma_1 \left(z_j^{l,1}\right) \sigma_2' \left(z_j^{l,2}\right) \tag{16b}$$

$$\frac{\partial C_{CE}}{\partial b_{j}^{l,3}} = \delta_j^l \sigma_3' \left(z_j^{l,3}\right) \tag{16c}$$

$$\frac{\partial C_{CE}}{\partial w_{jk}^{l,1}} = \delta_j^l \sigma_1' \left(z_j^{l,1}\right) \sigma_2 \left(z_j^{l,2}\right) a_k^{l-1} \tag{16d}$$

$$\frac{\partial C_{CE}}{\partial w_{jk}^{l,2}} = \delta_j^l \sigma_1 \left(z_j^{l,1}\right) \sigma_2' \left(z_j^{l,2}\right) a_k^{l-1} \tag{16e}$$

$$\frac{\partial C_{CE}}{\partial w_{jk}^{l,3}} = \delta_j^l \sigma_3' \left(z_j^{l,3}\right) \left(a_k^{l-1}\right)^2 \tag{16f}$$

The detailed derivation of Eq. (16) is given in the Appendix A.

The developed HO-DNN is implemented in the Python programming language and can be found in the Appendix B. The pseudo code of implementation is presented in Algorithm 1. It is worth noting that at the beginning of the algorithm, the weights and biases are initialised at each layer, normally by Gaussian distribution with a mean of zero and a standard deviation of one. However, this strategy of initialisation might lead to the saturation of neurons as discussed in Bengio (2012) and Nielsen (2015). Therefore, in this paper, the weights and biases of a layer are initialised by a Gaussian distribution with a mean of zero and a standard deviation of $\frac{1}{\sqrt{N}}$ (N is the number of neurons of the layer) to alleviate the saturation of neurons (Bengio, 2012). In Algorithm 1, when all the training data is used, it is said that the training process completes an epoch e and starts over with a new epoch. When the pre-defined *max_epochs* is reached, the training process of HO-DNN is completed.

4 DATASET AND PERFORMANCE INDICATOR

4.1 Data collection

The HO-DNN developed in the previous section is used to predict the compressive strength of foamed concrete through a given dataset. The dataset of foamed concrete (Dataset 1) consists of 177 testing results for different mixtures (density, water-to-cement ratio and sand-to-cement ratio). The samples of the Dataset 1 waere consistently made of ordinary Portland cement, water, sand and preformed foams, curing time of 28 days (Jones and McCarthy, 2005; Pan et al., 2007; Kiani et al., 2016; Abd and Abd, 2017; Asadzadeh and Khoshbayan, 2018). The range of density, w/c ratio and s/c ratio were $[430-2009]$ kg/m^3, $[0.26-0.83]$ and $[0-4.3]$, respectively. After developed and validated by the Dataset 1, the trained HO-DNN is supposed to be capable of predicting the foamed concrete strength correctly. Therefore, the trained HO-DNN was then exposed to a new dataset (Dataset 1A), which is completely different to the Dataset 1. The Dataset 1A consisting of 34 samples was prepared in the laboratory, having a density and w/c ratio in the range of $[444-2066]$ kg/m^3 and $[0.3-0.7]$, respectively. It is worth noting that the HO-DNN is trained in the range of $[430-2009]$ for density whilst the Dataset 1A has the density in the range of $[444-2066]$, outside of the training range. It is accepted that it is not recommended to use the HO-DNN for such extrapolation. However, for the Dataset 1A, it is a very near extrapolation (i.e. the Dataset 1A slightly exceeds the training region). Indeed, after militarisation by Eq. (17), the training region is in the range of $[0-1]$ for density and the Dataset 1A is in the range of $[0.009-1.036]$ for the density. Therefore, it is assume that the effect of such near extrapolation is negligible in this paper. For the Dataset 1A, general purpose ordinary Portland cement and a commercial foaming agent (Isochem S/X from Isoltech, Italy) are used to fabricate three foamed concrete groups: high, medium and low density. The cubic samples of 50x50x50mm in size are cured in sealed plastic bags at ambient temperature for 28 days. The uni-axial compressive strength of foamed concrete is measured using the INSTRON 5569A machine. The displacement-controlled test is conducted at a velocity of 0.03 mm/s to obtain the compressive strength. The purpose of the Dataset 1A is to show that the HO-DNN can obtain the reliable prediction of foamed concrete strength that the model has never been exposed, thereby helping the design of foamed concrete mixture.

Furthermore, to demonstrate the efficacy and versatility of the proposed HO-DNN model for different engineering applications, the HO-DNN is validated through the other dataset of high performance concrete (Dataset 2). The Dataset 2 is retrieved from the machine learning benchmark repository at the University of California, Irvine and consists of 1,133 testing results of high performance con-

Author Manuscript

This article is protected by copyright. All rights reserved.

Algorithm 1 The pseudo code of HO-DNN

Begin
Input: Training dataset X
Initialise weights $w_{jk}^{l,i}$ and biases $b_j^{l,i}$ $(i = 1,2,3)$
while $e \leq max_epochs$ **do**
for each data x in X **do**
Calculate z_j^l ▷ Using Eq. (9)
Calculate $a_j^l = \sigma\left(z_j^l\right)$ ▷ Using ReLU activation Eq. (13)
Calculate the error δ_j^L of the last layer L ▷ Using Eq. (11)
Calculate the error δ_j^l of the earlier layers l $(l = 1,2\ldots,(L-1))$ ▷ Using Eq. (12)
Update the weights $w_{jk}^{l,i}$ and biases $b_j^{l,i}$ $(i = 1,2,3)$ ▷ Using Eq. (15) and Eq. (16)
end for
Evaluate the cross-entropy cost of HO-DNN at epoch e ▷ Using Eq. (10)
Evaluate the performance of HO-DNN at epoch e ▷ Using Eq. (18)
$e = e+1$
end while
Output: Trained HO-DNN model
End

crete with different mixtures of cement, slag, fly ash, water, super-plasticizer, fine and coarse aggregate and age of testing (Yeh, 1998). The Dataset 2 was originally provided by Yeh (1998) and subsequently contributed by other researchers (Yeh, 2006). The Dataset 2 was extensively used in the literature (Bui et al., 2018; Rafiei et al., 2017a,b; Chou et al., 2014; Chou and Pham, 2013; Mousavi et al., 2012) and the information of the Dataset 2 was detailed in these references. It is noted that the foamed concrete (and high performance concrete for the Dataset 2) might exhibits a variation of compressive strength due to the uncertainty of the preparation and testing of concrete material (Gribniak et al., 2015, 2016). However, the main objective of this study is to develop the new Ho-DNN and demonstrate its efficiency and versatility. Therefore, it is assumed that the uncertainty of data is neglected in this study. It is known that this assumption is commonly accepted in the literature (Rafiei et al., 2017a,b; Chou and Pham, 2013; Yaseen et al., 2018; Abd and Abd, 2017). Investigations on the influence of uncertainty on deep neural network models can be found in literature (Gal, 2016; Koziarski and Cyganek, 2017; Antoniades et al., 2018).

Finally, in order to avoid the magnitude difference between inputs (e.g. density: 430-2009 kg/m^3; w/c ratio: 0.26-0.83), all data was normalised to the range of $[0,1]$ using the following equation:

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{17}$$

This data normalisation is crucial to avoid numerical difficulties in the calculation because of the magnitude difference between input values (Chou and Pham, 2013). The Dataset 1, 1A and 2 are available publicly in the Appendix B.

4.2 Performance Indicators

In this study, the accuracy of the proposed HO-DNN was assessed by comparing the actual (y) and predicted (y^{pre}) compressive strength for the Dataset 1, 1A and 2. The correlation coefficient (R), root mean square error (RMSE), mean absolute error (MAE), relative root mean square error (RRMSE) and relative mean absolute error (RMAE) were used as performance indicators, as follows:

$$R = \frac{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)\left(y_i^{pre} - \bar{y}^{pre}\right)}{\sqrt{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2 \sum_{i=1}^{n}\left(y_i^{pre} - \bar{y}^{pre}\right)^2}} \tag{18a}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - y_i^{pre}\right)^2} \tag{18b}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - y_i^{pre}\right| \tag{18c}$$

$$RRMSE = \frac{RMSE}{\bar{y}} \times 100 \tag{18d}$$

$$RMAE = \frac{MAE}{\bar{y}} \times 100 \tag{18e}$$

where n is the number of data; and $\bar{y}$ and $\bar{y}^{pre}$ are the mean actual and predicted value, respectively.

The correlation coefficient R is used to quantify the degree of linear dependence between the actual value and the predicted value (Yaseen et al., 2018). When R is close to zero, it means that there is no evidence of any relationship between the actual and predicted value whilst when R is close to 1, we are close to a perfect fit between the actual and predicted value (Chou and Pham, 2013). However, the result of R is not sufficient to evaluate the accuracy of a model because it does not

Author Manuscript

This article is protected by copyright. All rights reserved.

Table 1: The description of the C-ANN, SO-ANN and DO-DNN model

Model	Number of hidden layers	Number of neurons per layer	Learning rate η	Type of neurons	Activation function	Cost function
C-ANN	1, 3, 6	30	0.01	Linear	Sigmoid	Quadratic
SO-ANN	1, 2	30	0.01	Second order	Sigmoid	Quadratic
HO-DNN	1, 2	30	0.01	High order	ReLU	Cross-entropy

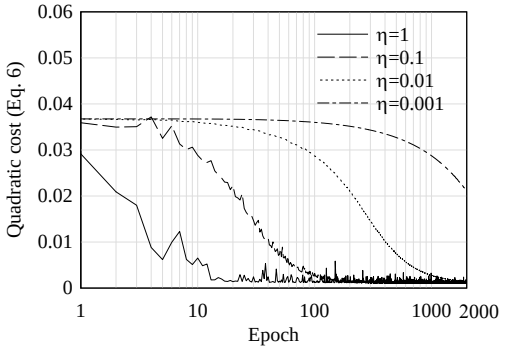
significantly change when y_i^{pre} is multiplied by a constant. Therefore, RMSE and MAE are the additional indicators to check the goodness-of-fit (in MPa units) of the ANN models. Moreover, RRMSE and RMAE are employed to investigate the percentage deviation of the actual and predicted data. In general, the higher value of R and the lower value of RMSE, MAE, RRMSE and RMAE result in a decrease of errors between the actual and predicted value, and thereby indicating the accuracy of models.

5 RESULTS AND DISCUSSION

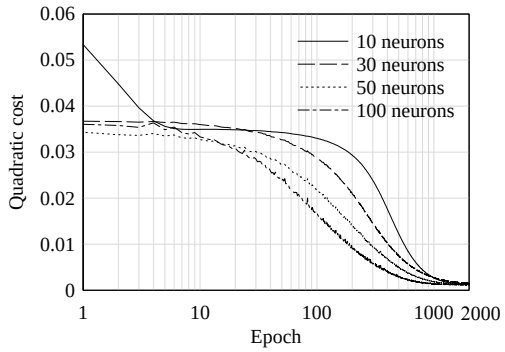
5.1 Comparative study

This section presents the application of the developed HO-DNN for the prediction of foamed concrete strength. To compare the performance of the HO-DNN, the C-ANN presented in Section 3.1 and the second order artificial neural network (SO-ANN), which was presented in Fan et al. (2018), are used. The description of the C-ANN, SO-ANN and HO-DNN is presented in Table 1. It is noted that the C-ANN with three and six hidden layers and the SO-ANN, HO-DNN with one and two hidden layers have the same number of weights and biases, respectively. To construct and validate the C-ANN, SO-ANN and HO-DNN, the data of the Dataset 1 is randomly split into training data (80%), validating data (10%) and testing data (10%). To avoid the effect of randomness of data, the same training, validating and testing data were used for the three models in this study. That means there models display the same level of randomness and uncertainty of data. Therefore, the comparison of there models is valid and informative. In this study, the training process is completed when the *max_epochs* is equal to 2000 (Algorithm 1).

The effects of learning rate η and the number of neurons of hidden layer are first investigated for the C-ANN model by the validating data of Dataset 1. Fig. 4a shows the variation of the quadratic cost (Eq. (6)) for validating data during the training process. The C-ANN model is used with one hidden layer, 30 neurons in the hidden layer and different learning rate $\eta = 1, 0.1, 0.01, 0.001$. Apparently, a large value of the learning rate ($\eta = 1, 0.1$) increases the resultant noise. This is attributed to the fact that the model learns too fast and may overshoot an expected minimum. On the contrary, this oscillation of the result can be alleviated by using a small learning rate ($\eta = 0.01, 0.001$). However, a very small learning rate ($\eta = 0.001$) slows down the learning process, increasing the training time until the solution process converges. Fig. 4a shows the learning rate of 0.01 is appropriate for the C-ANN in this study. The effect of the number of neurons of hidden layer is presented in Fig. 4b for the C-ANN with one hidden layer and the learning rate of 0.01. Fig. 4b shows that the number of neurons of hidden layer has negligible effect on the C-ANN, in which four cases are converged. It is also observed in Fig. 4b that the 50 and 100-neuron case lead to oscillating results. Therefore, the number of neurons of hidden layers is 30 for the C-ANN as presented in Table 1. Moreover, the variation of the quadratic cost (Fig. 4) for validating data during the training process shows that overfitting problem does not occur in this study. Because the cost of validating data decrease gradually during the training pro-



(a)



(b)

Figure 4: The effects of learning rate η (a) and the number of neurons of hidden layer (b) on the performance of C-ANN. The quadratic cost in evaluated using Eq. (6) for the validating data of the Dataset 1.

Author Manuscript

This article is protected by copyright. All rights reserved.

Table 2: The comparison of the C-ANN, SO-ANN and HO-DNN models with different number of hidden layers (N_layers) for the validating data of the Dataset 1.

Model	N_layers	R	RMSE (MPa)	MAE (MPa)	RMMSE (%)	RMAE (%)
C-ANN	1	0.9828	2.58	2.25	9.45	8.23
C-ANN	3	0.9639	4.52	3.83	16.55	14.04
		(-1.92)	(-75.08)	(-70.61)	(-75.13)	(-70.60)
C-ANN	6	0.7177	12.79	11.36	46.82	41.59
		(-26.97)	(-395.43)	(-405.39)	(-395.45)	(-405.35)
SO-ANN	1	0.9865	2.15	1.77	7.87	6.49
		(0.38)	(16.75)	(21.16)	(16.72)	(21.14)
SO-ANN	2	0.9854	2.18	1.88	7.98	6.87
		(0.26)	(15.59)	(16.48)	(15.56)	(16.52)
HO-DNN*	1	0.9877	2.15	1.67	7.87	6.11
		(0.50)	(16.76)	(25.80)	(16.72)	(25.76)
HO-DNN*	2	0.9921	1.65	1.17	6.06	4.29
		(0.95)	(35.91)	(47.89)	(35.87)	(47.87)

Values in parentheses indicate the improvement (%) compared to the C-ANN, N_layers= 1.
* indicates that the performances are genuinely improved for a deep model.

cess (Marsland, 2015). In the same way, the learning rate and number of neurons of the SO-ANN and HO-DNN are 0.01 and 30, respectively, as given in Table 1. It is noted that the same learning rate and number of neurons assure the fair comparison of C-ANN, SO-ANN and HO-DNN models.

Table 2 shows the comparison of the C-ANN, SO-ANN and HO-DNN models with different number of hidden layers (N_layers) for the validating data of the Dataset 1. As aforementioned, the C-ANN (N_layers= $3,6$), the SO-ANN (N_layers= $1,2$) and the HO-DNN (N_layers= $1,2$) have the same number of weights of biases, respectively. It is shown that the HO-DNN (N_layers= $1,2$) outperforms the SO-ANN due to the advantages of cross-entropy cost and ReLU activation functions. Moreover, the HO-DNN (N_layers= 2) achieves the lowest RMSE and MAE (1.65 and 1.17 MPa, respectively), which exhibits approximately 36% and 48% improvement compared to the C-ANN (N_layers= 1). Table 2 also demonstrates the vanishing gradient problem of the C-ANN and SO-ANN. In fact, adding more hidden layers does not improve, but genuinely degrade the performances of C-ANN model as presented for the cases of N_layers= $3,6$ in Table 2. The same phenomenon can be observed for the SO-ANN in Table 2, in which the performances of SO-ANN with two hidden layers are worst than those of one hidden layer. On the contrary, the proposed HO-DNN genuinely improves its performances when adding more hidden layers as shown in Table 2. That is why the present approach is named as high order **"deep"** neural network while the other models are conventional and second order artificial neural network. It is remarkable that the HO-DNN with two hidden layers can break the 0.9900 barrier of the correlation coefficient R which is not possible with the C-ANN and SO-ANN.

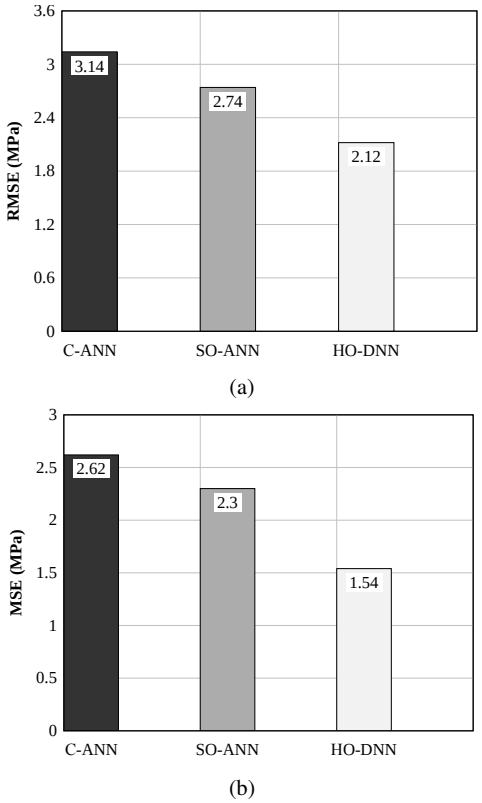
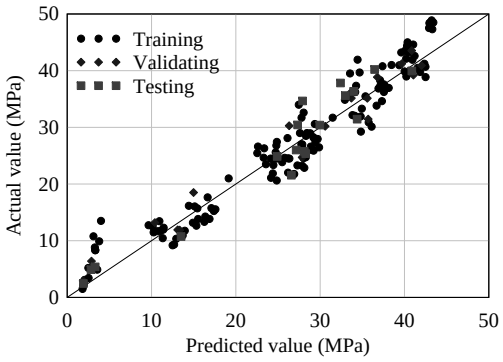


Figure 5: The comparison of the C-ANN, SO-ANN and HO-DNN for the testing data of the Dataset 1: (a) root mean square error (RMSE) Eq. (18b) and (b) mean absolute error (MAE) Eq. (18c).

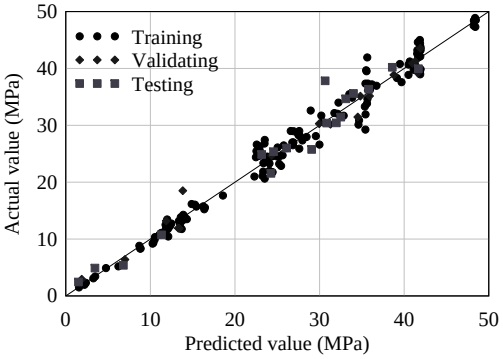
After the architecture (e.g. number of hidden layer, number of neurons, learning rate) of the models is determined using the validating data, the C-ANN (N_layers= 1), the SO-ANN (N_layers= 1) and the HO-DNN (N_layers= 2) are validated

Author Manuscript

This article is protected by copyright. All rights reserved.

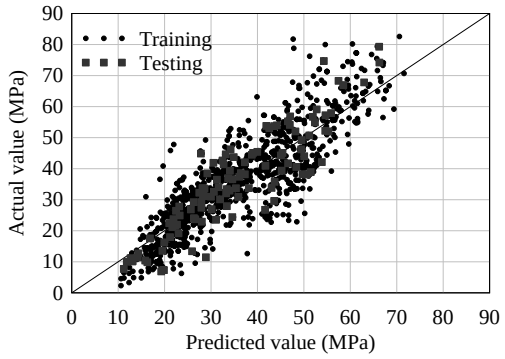


(a) C-ANN

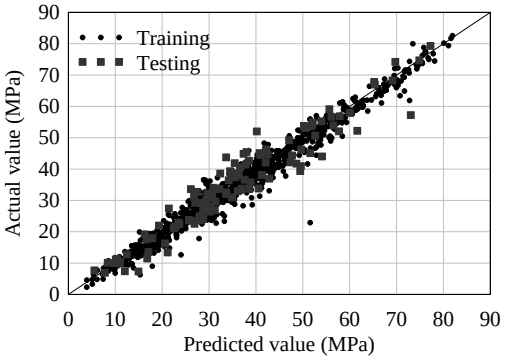


(b) HO-DNN

Figure 6: The correlation between the actual and predicted foamed concrete strength of the Dataset 1 using the C-ANN and HO-DNN model.



(a) C-ANN



(b) HO-DNN

Figure 7: The correlation between the actual and predicted compressive strength of the Dataset 2 using the C-ANN and HO-DNN model.

for the testing data of the Dataset 1 in Fig. 5. It is shown that the HO-DNN is better than the C-ANN and SO-ANN in the prediction of foamed concrete strength for the testing data of Dataset 1. In fact, the HO-DNN produces the smallest RMSE and MAE (2.12 and 1.54 MPa, respectively). On the other hand, the C-ANN provides the prediction at a lowest level of accuracy with the highest RMSE and MAE (3.14 and 2.62 MPa, respectively). Fig. 6 shows the correlation between the actual and predicted compressive strength of foamed concrete for the training, validating and testing data of the Dataset 1 using the C-ANN (N_layers= 1) and HO-DNN (N_layers= 2). It is obvious that the HO-DNN achieves the most reliable prediction as shown in Fig. 6 as well as Table 2 and Fig. 5. This can be attributed to the use of high order neurons in the HO-DNN, incorporating with the ReLU and cross-entropy cost functions. In fact, it is known that the compressive strength of foamed concrete is a highly nonlinear function of its density and constituents (Ngo et al., 2017; Nguyen et al., 2017, 2018). The high order neurons enable the HO-DNN to account for the higher order relationship between neurons rather than the linear inner product of the C-ANN whilst the ReLU and cross-entropy cost function improve the training process and enable a genuinely deep neural network model. As a result, the HO-DNN model outperforms the C-ANN and SO-ANN models in predicting the foamed concrete strength of the Dataset 1.

To demonstrate the efficacy and versatility of the proposed HO-DNN method for different engineering applications, the HO-DNN, SO-ANN and C-ANN are tested through the Dataset 2 of the compressive strength of high performance concrete. The architecture of HO-DNN, SO-ANN and C-ANN is identical with the previous example of the Dataset 1. The Dataset 2 is randomly split into training data (90%) and testing data (10%) (Bui et al., 2018; Chou and Pham, 2013). Table 3 shows the performance of the proposed HO-DNN in the prediction of the Dataset 2. The performance of the HO-DNN is compared with the C-ANN, SO-ANN and other ML methods reported in the literature including Gene Expression Programming (GEP) (Mousavi et al., 2012), Multi-Gene Genetic Programming (M-GGP) (Gandomi et al., 2013), Ensemble Model Artificial Neural Network-Support Vector Regression (ANN-SVR) (Chou and Pham, 2013), Smart Firefly Algorithm–based Least Squares SVR (SFA-LSSVR) (Chou et al., 2016) and Modified Firefly Algorithm-based Artificial Neural Network (MFA-ANN) (Bui et al., 2018). Table 3 shows that the proposed HO-DNN obtains the best prediction for the Dataset 2 with the correlation coefficient R of 0.97, the root mean square error RMSE of 4.05 MPa and the mean absolute error MAE of 2.85 MPa, compared to other methods. In general, the RMSE and MAE of the HO-DNN

Author Manuscript

This article is protected by copyright. All rights reserved.

Table 3: The comparison of performance and improvement rates of the HO-DNN for the testing data of the Dataset 2.

Model	Performance			Improved by HO-DNN (%)		
	R	RMSE (MPa)	MAE (MPa)	R	RMSE	MAE
GEP^{+}	0.91	-	5.20	6.59*	-	45.19*
M-GGP^{+}	0.90	7.31	5.48	7.78*	44.60*	47.99*
ANN-SVR^{+}	0.94	6.17	4.24	3.19	34.36*	32.78*
SFA-LSSVR^{+}	0.94	5.62	3.86	3.19	27.94	26.17
MFA-ANN^{+}	0.95	4.85	3.41	2.11	16.49	16.42
C-ANN (present)	0.90	7.17	5.59	7.78*	43.51*	49.02*
SO-ANN (present)	0.93	5.82	4.50	4.30	30.41*	36.67*
HO-DNN (present)	0.97	4.05	2.85	-	-	-

$^{+}$ indicates the results are obtained from the literature.

* indicates significant improvements by HO-DNN ($\geq 5\%$ for R, $\geq 30\%$ for RMSE and MAE).

Table 4: The comparison of C-ANN, SO-ANN, HO-DNN and two empirical equations in predicting the compressive strength of the Dataset 1A.

Model	R	RMSE (MPa)	MAE (MPa)	RMMSE (%)	RMAE (%)
Hoff	0.9911	4.89	2.559	31.58	16.48
Nambiar	0.9913	3.17	1.75	20.46	11.31
C-ANN	0.9937	2.11	1.39	13.66	8.96
SO-ANN	0.9951	1.90	1.29	12.26	8.35
HO-DNN	0.9978	1.30	0.91	8.38	5.89

are about 17%-45% and 16%-49%, respectively, better than those of other methods. Whilst the results of SO-ANN are comparable with those of GEP (Mousavi et al., 2012) and M-GGP (Gandomi et al., 2013) methods. Fig. 7 depicts the correlation between the actual and predicted strength of the Dataset 2 using the HO-DNN and C-ANN models. It is observed that by using the HO-DNN, the predicted strengths are closer to the actual strength, compared to the C-ANN model. The obtained results for the Dataset 1 and Dataset 2 show that the proposed HO-DNN provides a reliable and versatile prediction tool for engineering applications.

5.2 Accuracy of the proposed HO-DNN on the new Dataset 1A

In the previous section, the proposed HO-DNN was validated through the Dataset 1 and Dataset 2 from the literature. It was shown that the HO-DNN can deliver the best performance in predicting the foamed concrete strength of the Dataset 1. In order to achieve a higher degree of confidence and demonstrate that the proposed HO-DNN is genuinely capable of aiding researchers and engineers in the practical design of foamed concrete, the trained HO-DNN model was tested with the Dataset 1A for foamed concrete in this section. The obtained results are compared with two empirical equations presented in Section 2, the Hoff's equation Eq. (1) and Nambiar's equation Eq. (2). It is accepted that the Hoff's and Nambiar's equations are commonly used to predict the compressive strength of foamed concrete in the mixture design of foamed concrete. The Hoff's equation Eq. (1) is used with $\sigma_0 = 115$ MPa, $b = 2.7$ as given by Hoff (1972) and the Nambiar's equation is used with $\sigma_0 = 105.14$ MPa, $b = 2.68$ as given by Nambiar and Ramamurthy (2008). Table 4 presents the comparison of the trained C-ANN, trained SO-ANN, trained HO-DNN models and two empirical equations in the prediction of the Dataset 1A. The correlation between the actual and predicted strength of the Dataset 1A is depicted in Fig. 8 using the C-ANN, HO-DNN and Hoff's and Nambia's equations. Due to the small number of data in the Dataset 1A (34 data), it is easy for all methods (Hoff's, Nambiar's, C-ANN, SO-ANN and HO-DNN) to obtain a high value for the correlation coefficient R (Table 4). However, the relative errors (RRMSE and RMAE) of Hoff's, Nambiar's, C-ANN and SO-ANN methods are higher than 10%, indicating a low level of prediction accuracy. It can be observed in Fig. 8 that Hoff and Nambiar (Fig. 8a and Fig. 8b) equations can provide the good prediction of low strength values group but not high strength values group. On the contrary, the HO-DNN model is able to predict accurately the compressive strength of the Dataset 1A with an acceptable relative error (less than 8.38% in Table 4). Based on the examples of the Dataset 1 and Dataset 1A, one can conclude that the proposed HO-DNN provides a reliable and accurate tool to predict the compressive strength of foamed concrete. Achieving the RRMSE of 8.38% and the RMAE of 5.89% for the Dataset 1A, the proposed HO-DNN model can realis-

Author Manuscript

This article is protected by copyright. All rights reserved.

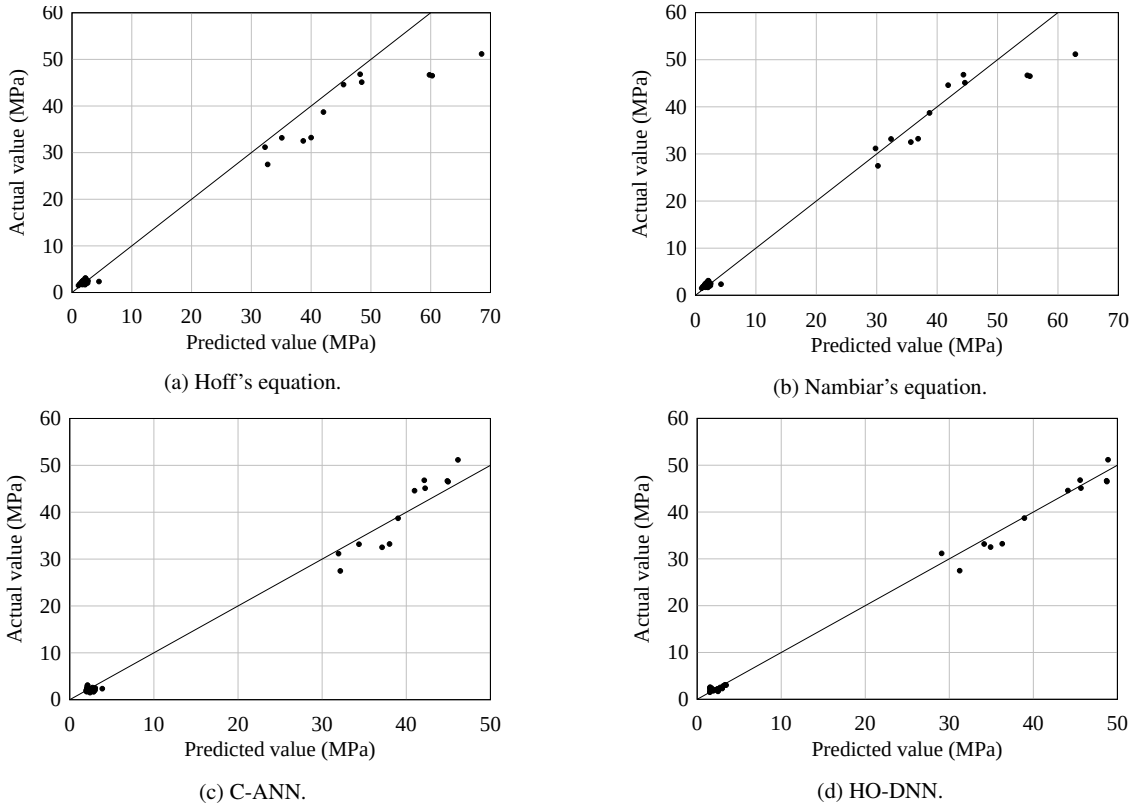


Figure 8: The correlation between the actual and predicted compressive strength of the Dataset 1A.

tically benefit the design of foamed concrete by reducing the laboratory work required.

5.3 Sensitivity analysis

With the reliable HO-DNN model presented in the previous sections, a sensitivity analysis (SA) can be conducted to examine the effect of density, w/c ratio and s/c ratio on the compressive strength of foamed concrete. In order to conduct the sensitivity study, the compressive strength of foamed concrete was determined by changing one input variable, whilst other variables are kept constant at the average values (Kiani et al., 2016). For instance, to examine the effect of density, density is varied from 430 kg/m^3 to 2009 kg/m^3 and the w/c and s/c ratios were held constant at 0.405 and 1.167, respectively. Once the data for the SA study was created, the data was fed into the trained HO-DNN model to obtain the compressive strength. For each input variable, a corresponding SA parameter can be simply evaluated as follows (Gandomi et al., 2011):

$$I_i = f_{\max}(x_i) - f_{\min}(x_i) \tag{19a}$$

$$SA_i = \frac{I_i}{\sum_i I_i} \times 100 \tag{19b}$$

in which $f_{\max}(x_i)$ and $f_{\min}(x_i)$ are the maximum and minimum predicted strength corresponding to the input variable x_i whilst other input variables are kept constants at their average values. Fig. 9 shows the SA results of the compressive strength of foamed concrete with respect to density, w/c ratio and s/c ratio. As shown in Fig. 9, the compressive strength of foamed concrete is strongly influenced by its density which has the SA parameter of about 44%. This can be attributed to the fact that air voids in foamed concrete, which reduce density, collapse and coalesce into macro-cracks inside materials under mechanical load, leading to a decrease in the compressive strength (Hajimohammadi et al., 2018; Nguyen et al., 2018). Fig. 9 also shows that the w/c ratio is the second important parameter with a SA parameter of about 34.5%, followed by the s/c ratio with a SA parameter of about 21.5%. In fact, the similar observation can be also found in the literature (Kiani et al., 2016; Ramamurthy et al., 2009)

6 CONCLUSIONS

In this paper, a deep, high order neural network (HO-DNN) was presented for predicting the compressive strength of foamed concrete (Dataset 1, Dataset 1A) and high performance concrete (Dataset 2). Particularly, this study proposed a new, high order neuron for the deep neural network model to account for nonlinear relationship between input activations at layers. Furthermore, the cross-entropy cost and ReLU activation functions were used to develop the HO-

This article is protected by copyright. All rights reserved.

Author Manuscript

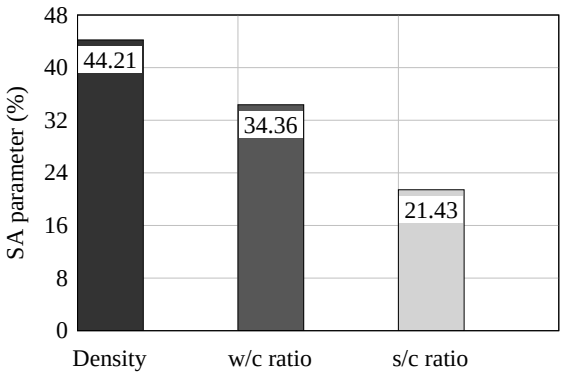


Figure 9: The sensitivity analysis for the compressive strength of foamed concrete.

DNN to improve the performances and alleviate the vanishing gradient problem when training a deep neural network model. The accuracy of the proposed HO-DNN in predicting the compressive strength of the Dataset 1 and Dataset 2 is obtained and compared with a conventional artificial neural network (C-ANN), a second order artificial neural network (SO-ANN) proposed by Fan et al. (2018) and other methods from previous studies (for Dataset 2).

The higher order neuron, cross-entropy cost and ReLU activation functions significantly improved the performances of the HO-DNN, compared to the C-ANN and SO-ANN. The cross-entropy cost function helped the proposed HO-DNN to learn quickly and more accurately. The ReLU activation function alleviates the vanishing gradient problem and genuinely improved the performance of the HO-DNN when training multiple hidden layers. For the Dataset 1, the HO-DNN outperformed the C-ANN and SO-ANN in the prediction of foamed concrete strength with the remarkable improvements of 36% (RMSE) and 48% (MAE) compared to the C-ANN model. Notably, the HO-DNN model was able to break the 0.99 barrier of the correlation coefficient R which was not possible with the C-ANN and SO-ANN. For the Dataset 2, the developed HO-DNN was better than other methods presented in the previous studies. The improvement rates of RMSE and MAE were approximately 50% for Gene Expression Programming method, Multi-Gene Generic Programming method and C-ANN method.

To further demonstrate the accuracy of the HO-DNN in the prediction of foamed concrete strength, the trained HO-DNN model (by the Dataset 1), was validated through the Dataset 1A obtained from the laboratory. The comparison between the trained HO-DNN with two empirical equations showed that the HO-DNN can predict the foamed concrete strength with a high level of accuracy. Having the reliable HO-DNN model, a sensitivity analysis (SA) was conducted to investigate the effects of the input variables on the compressive strength of foamed concrete. It was shown that density is the most important factor affecting the compressive strength of foamed concrete with a SA parameter of about 44.2%, followed by w/c ratio (34.4%) and s/c ratio (21.4%).

Whilst the proposed HO-DNN model was used to predict the compressive strength of concrete material in this study, it can be used to estimate other properties of concrete such as thermal conductivity, flexural strength. Therefore, the proposed framework can support researcher and engineers in mixture design optimisation of concrete. Moreover, it is desirable to investigate the application of the HO-DNN for different engineering problems in future studies.

ACKNOWLEDGMENTS

The authors would like to thank the Editor and the five anonymous reviewers for their constructive comments and valuable suggestions to improve the quality of the article. The first author would like to thank the University of Melbourne for offering the Melbourne Research Scholarship. The authors would like to acknowledge the support of the ARC Centre for Advanced Manufacturing of Prefabricated Housing (ARC IC150100023).

A Appendix A

The detailed derivation of the HO-DNN model are given as follows:

$$\delta_k^l = \frac{\partial C_{CE}}{\partial z_k^l} = \sum_j \frac{\partial C_{CE}}{\partial z_j^{l+1}} \frac{\partial z_j^{l+1}}{\partial z_k^l} = \sum_j \delta_j^{l+1} \left[\begin{array}{l} \dfrac{\partial \sigma_1\left(z_j^{l+1,1}\right)}{\partial z_k^l} \sigma_2\left(z_j^{l+1,2}\right) \\ + \dfrac{\partial \sigma_2\left(z_j^{l+1,2}\right)}{\partial z_k^l} \sigma_1\left(z_j^{l+1,1}\right) + \dfrac{\partial \sigma_3\left(z_j^{l+1,3}\right)}{\partial z_k^l} \end{array} \right] \tag{A.1a}$$

$$\frac{\partial \sigma_1\left(z_j^{l+1,1}\right)}{\partial z_k^l} = \sum_m \frac{\partial \sigma_1\left(z_j^{l+1,1}\right)}{\partial a_m^l} \frac{\partial a_m^l}{\partial z_k^l} = \frac{\partial \sigma_1\left(z_j^{l+1,1}\right)}{\partial a_k^l} \sigma'\left(z_k^l\right) = \sum_m \frac{\partial \sigma_1\left(z_j^{l+1,1}\right)}{\partial z_m^{l+1,1}} \frac{\partial z_m^{l+1,1}}{\partial a_k^l} \sigma'\left(z_k^l\right) = w_{jk}^{l+1,1} \sigma'_1\left(z_j^{l+1,1}\right) \sigma'\left(z_k^l\right) \tag{A.1b}$$

$$\frac{\partial \sigma_2\left(z_j^{l+1,2}\right)}{\partial z_k^l} = w_{jk}^{l+1,2} \sigma'_2\left(z_j^{l+1,2}\right) \sigma'\left(z_k^l\right) \tag{A.1c}$$

$$\frac{\partial a_j^{l+1,3}}{\partial z_k^l} = 2 w_{jk}^{l+1,3} \sigma'_3\left(z_j^{l+1,3}\right) \sigma\left(z_k^l\right) \sigma'\left(z_k^l\right) \tag{A.1d}$$

Author Manuscript

This article is protected by copyright. All rights reserved.

$$\frac{\partial C_{CE}}{\partial b_j^{l,1}} = \sum_k \frac{\partial C_{CE}}{\partial z_k^l}\frac{\partial z_k^l}{\partial b_j^{l,1}} = \sum_k \delta_k^l \frac{\partial \sigma\left(z_k^{l,1}\right)}{\partial b_j^{l,1}}\sigma_2\left(z_k^{l,2}\right)$$
$$= \delta_j^l \frac{\partial \sigma_1\left(z_j^{l,1}\right)}{\partial b_j^{l,1}}\sigma_2\left(z_j^{l,2}\right) =$$
$$\delta_j^l \sigma_2\left(z_j^{l,2}\right)\sum_k \frac{\partial \sigma\left(z_j^{l,1}\right)}{\partial z_k^{l,1}}\frac{\partial z_k^{l,1}}{\partial b_j^{l,1}} = \delta_j^l \sigma'_1\left(z_j^{l,1}\right)\sigma_2\left(z_j^{l,2}\right) \tag{A.2}$$

$$\frac{\partial C}{\partial w_{jk}^{l,1}} = \sum_m \frac{\partial C}{\partial z_m^l}\frac{\partial z_m^l}{\partial w_{jk}^{l,1}} = \sum_m \delta_m^l \frac{\partial \sigma_1\left(z_m^{l,1}\right)}{\partial w_{jk}^{l,1}}\sigma_2\left(z_m^{l,2}\right) =$$
$$\delta_j^l \frac{\partial \sigma_1\left(z_j^{l,1}\right)}{\partial w_{jk}^{l,1}}\sigma_2\left(z_j^{l,2}\right) = \delta_j^l \sigma_2\left(z_j^{l,2}\right)\sum_m \frac{\partial \sigma_1\left(z_j^{l,1}\right)}{\partial z_m^{l,1}}\frac{\partial z_m^{l,1}}{\partial w_{jk}^{l,1}}$$
$$= \delta_j^l \sigma'_1\left(z_j^{l,1}\right)\sigma_2\left(z_j^{l,2}\right)a_k^{l-1} \tag{A.3}$$

B Appendix B

The Dataset 1, Dataset 1A, Dataset 2 and the developed HO-DNN model can be downloaded from `https://figshare.com/s/6424e3931fd5a82bbd3a`

References

Abd, A. M. and Abd, S. M. (2017). Modelling the strength of lightweight foamed concrete using support vector machine (SVM). *Case Studies in Construction Materials*, 6:8–15.

Adeli, H. (2001). Neural Networks in Civil Engineering: 1989–2000. *Computer-Aided Civil and Infrastructure Engineering*, 16(2):126–142.

Adeli, H. and Panakkat, A. (2009). A probabilistic neural network for earthquake magnitude prediction. *Neural networks*, 22(7):1018–1024.

Adeli, H. and Wu, M. (1998). Regularization neural network for construction cost estimation. *Journal of construction engineering and management*, 124(1):18–24.

Antoniades, A., Spyrou, L., Martin-Lopez, D., Valentin, A., Alarcon, G., Sanei, S., and Took, C. C. (2018). Deep neural architectures for mapping scalp to intracranial eeg. *International journal of neural systems*, page 1850009.

Asadzadeh, S. and Khoshbayan, S. (2018). Multi-objective optimization of influential factors on production process of foamed concrete using Box-Behnken approach. *Construction and Building Materials*, 170:101–110.

Bengio, Y. (2012). Practical Recommendations for Gradient-Based Training of Deep Architectures. *Neural Networks: Tricks of the Trade*, pages 437–478.

Bui, D.-K., Nguyen, T., Chou, J.-S., Nguyen-Xuan, H., and Ngo, T. D. (2018). A modified firefly algorithm-artificial neural network expert system for predicting compressive and tensile strength of high-performance concrete. *Construction and Building Materials*, 180:320 – 333.

Chiew, F. H., Ng, C. K., Chai, K. C., and Tay, K. M. (2017). A Fuzzy Adaptive Resonance Theory-Based Model for Mix Proportion Estimation of High-Performance Concrete: A fuzzy adaptive resonance theory. *Computer-Aided Civil and Infrastructure Engineering*, 32(9):772–786.

Chou, J.-S., Chong, W. K., and Bui, D.-K. (2016). Nature-inspired metaheuristic regression system: programming and implementation for civil engineering applications. *Journal of Computing in Civil Engineering*, 30(5):04016007.

Chou, J.-S. and Pham, A.-D. (2013). Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength. *Construction and Building Materials*, 49:554–563.

Chou, J.-S. and Pham, A.-D. (2015). Smart Artificial Firefly Colony Algorithm-Based Support Vector Regression for Enhanced Forecasting in Civil Engineering: Smart artificial firefly colony-based support vector regression. *Computer-Aided Civil and Infrastructure Engineering*, 30(9):715–732.

Chou, J.-S., Tsai, C.-F., Pham, A.-D., and Lu, Y.-H. (2014). Machine learning in concrete strength simulations: Multi-nation data analytics. *Construction and Building Materials*, 73:771–780.

Dai, H. and Cao, Z. (2017). A wavelet support vector machine-based neural network metamodel for structural reliability assessment. *Computer-Aided Civil and Infrastructure Engineering*, 32(4):344–357.

DeRousseau, M. A., Kasprzyk, J. R., and Srubar, W. V. (2018). Computational design optimization of concrete mixtures: A review. *Cement and Concrete Research*, 109:42–53.

Dharia, A. and Adeli, H. (2003). Neural network model for rapid forecasting of freeway link travel time. *Engineering Applications of Artificial Intelligence*, 16(7-8):607–613.

Fan, F., Cong, W., and Wang, G. (2018). A new type of neurons for machine learning. *International Journal for Numerical Methods in Biomedical Engineering*, 34(2):e2920.

Gal, Y. (2016). Uncertainty in deep learning. *University of Cambridge*.

Gandomi, A. H., Alavi, A. H., Shadmehri, D. M., and Sahab, M. (2013). An empirical model for shear capacity of rc deep beams using genetic-simulated annealing. *Archives of Civil and Mechanical Engineering*, 13(3):354–369.

Gandomi, A. H., Tabatabaei, S. M., Moradian, M. H., Radfar, A., and Alavi, A. H. (2011). A new prediction model for the load capacity of castellated steel beams. *Journal of Constructional Steel Research*, 67(7):1096 – 1105.

Gao, Y. and Mosalam, K. M. (2018). Deep Transfer Learning for Image-Based Structural Damage Recognition. *Computer-Aided Civil and Infrastructure Engineer-*

Author Manuscript

This article is protected by copyright. All rights reserved.

ing, 0(0).

García-Ródenas, R., López-García, M. L., and Sánchez-Rico, M. T. (2017). An approach to dynamical classification of daily traffic patterns. *Computer-Aided Civil and Infrastructure Engineering*, 32(3):191–212.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Grande, Z., Castillo, E., Mora, E., and Lo, H. K. (2017). Highway and road probabilistic safety assessment based on bayesian network models. *Computer-Aided Civil and Infrastructure Engineering*, 32(5):379–396.

Gribniak, V., Mang, H. A., Kupliauskas, R., and Kaklauskas, G. (2015). Stochastic tension-stiffening approach for the solution of serviceability problems in reinforced concrete: Constitutive modeling. *Computer-Aided Civil and Infrastructure Engineering*, 30(9):684–702.

Gribniak, V., Mang, H. A., Kupliauskas, R., Kaklauskas, G., and Juozapaitis, A. (2016). Stochastic tension-stiffening approach for the solution of serviceability problems in reinforced concrete: Exploration of predictive capacity. *Computer-Aided Civil and Infrastructure Engineering*, 31(6):416–431.

Hajimohammadi, A., Ngo, T., and Mendis, P. (2017a). How does aluminium foaming agent impact the geopolymer formation mechanism? *Cement and Concrete Composites*, 80(Supplement C):277–286.

Hajimohammadi, A., Ngo, T., and Mendis, P. (2018). Enhancing the strength of pre-made foams for foam concrete applications. *Cement and Concrete Composites*.

Hajimohammadi, A., Ngo, T., Mendis, P., Nguyen, T., Kashani, A., and van Deventer, J. S. J. (2017b). Pore characteristics in one-part mix geopolymers foamed by H2o2: The impact of mix design. *Materials & Design*, 130(Supplement C):381–391.

Hajimohammadi, A., Ngo, T., Mendis, P., and Sanjayan, J. (2017c). Regulating the chemical foaming reaction to control the porosity of geopolymer foams. *Materials & Design*, 120:255–265.

Hoff, G. C. (1972). Porosity-strength considerations for cellular concrete. *Cement and Concrete Research*, 2(1):91–100.

Jones, M. and McCarthy, A. (2005). Preliminary views on the potential of foamed concrete as a structural material. *Magazine of concrete research*, 57(1):21–31.

Karpathy, A. (2018). Stanford University CS231n: Convolutional Neural Networks for Visual Recognition.

Kiani, B., Gandomi, A. H., Sajedi, S., and Liang, R. Y. (2016). New Formulation of Compressive Strength of Preformed-Foam Cellular Concrete: An Evolutionary Approach. *Journal of Materials in Civil Engineering*, 28(10):04016092.

Koziarski, M. and Cyganek, B. (2017). Image recognition with deep neural networks in presence of noise–dealing with and taking advantage of distortions. *Integrated Computer-Aided Engineering*, 24(4):337–349.

Ley, C. and Bordas, S. P. A. (2018). What makes data science different? a discussion involving statistics2.0 and computational sciences. *International Journal of Data Science and Analytics*.

Marsland, S. (2015). *Machine learning: an algorithmic perspective*. CRC press.

Mousavi, S. M., Aminian, P., Gandomi, A. H., Alavi, A. H., and Bolandi, H. (2012). A new predictive model for compressive strength of hpc using gene expression programming. *Advances in Engineering Software*, 45(1):105–114.

Nabian, M. A. and Meidani, H. (2018). Deep Learning for Accelerated Seismic Reliability Analysis of Transportation Networks. *Computer-Aided Civil and Infrastructure Engineering*, 33(6):443–458.

Naderpour, H., Rafiean, A. H., and Fakharian, P. (2018). Compressive strength prediction of environmentally friendly concrete using artificial neural networks. *Journal of Building Engineering*, 16:213–219.

Nambiar, E. K. K. and Ramamurthy, K. (2008). Models for strength prediction of foam concrete. *Materials and Structures*, 41(2):247.

Nehdi, M., Djebbar, Y., and Khan, A. (2001). Neural network model for preformed-foam cellular concrete. *Materials Journal*, 98(5):402–409.

Neville, A. M. (2012). *Properties of Concrete*. Prentice Hall.

Ngo, T., Hajimohammadi, A., Sanjayan, J., and Mendis, P. (2017). Characterisation tests and design of foam concrete for prefabricated modular construction. *Concrete in Australia*, 43(3).

Nguyen, T., Ghazlan, A., Kashani, A., Bordas, S., and Ngo, T. (2018). 3d meso-scale modelling of foamed concrete based on x-ray computed tomography. *Construction and Building Materials*, 188:583–598.

Nguyen, T. T., Bui, H. H., Ngo, T. D., and Nguyen, G. D. (2017). Experimental and numerical investigation of influence of air-voids on the compressive behaviour of foamed concrete. *Materials & Design*, 130(Supplement C):103–119. ++.

Nielsen, I. A. (2015). *Neural Networks and Deep Learning*. Determination Press.

Pan, Z., Hiromi, F., and Wee, T. (2007). Preparation of high performance foamed concrete from cement, sand and mineral admixtures. *Journal of Wuhan University of Technology-Mater. Sci. Ed.*, 22(2):295–298.

Panakkat, A. and Adeli, H. (2009). Recurrent neural network for approximate earthquake time and location prediction using multiple seismicity indicators. *Computer-Aided Civil and Infrastructure Engineering*, 24(4):280–292.

Rafiei, M. H. and Adeli, H. (2017). A novel machine learning-based algorithm to detect damage in high-rise

Author Manuscript

This article is protected by copyright. All rights reserved.

building structures. *The Structural Design of Tall and Special Buildings*, 26(18).

Rafiei, M. H. and Adeli, H. (2018). A novel unsupervised deep learning model for global and local health condition assessment of structures. *Engineering Structures*, 156:598–607.

Rafiei, M. H., Khushefati, W. H., Demirboga, R., and Adeli, H. (2016). Neural network, machine learning, and evolutionary approaches for concrete material characterization. *ACI Materials Journal*, 113(6).

Rafiei, M. H., Khushefati, W. H., Demirboga, R., and Adeli, H. (2017a). Novel approach for concrete mixture design using neural dynamics model and virtual lab concept. *ACI Materials Journal*, 114(1).

Rafiei, M. H., Khushefati, W. H., Demirboga, R., and Adeli, H. (2017b). Supervised deep restricted boltzmann machine for estimation of concrete. *ACI Materials Journal*, 114(2).

Ramamurthy, K., Kunhanandan Nambiar, E., and Indu Siva Ranjani, G. (2009). A classification of studies on properties of foam concrete. *Cement and Concrete Composites*, 31(6):388–396.

Roberts, C. A. and Attoh-Okine, N. O. (1998). A Comparative Analysis of Two Artificial Neural Networks Using Pavement Performance Prediction. *Computer-Aided Civil and Infrastructure Engineering*, 13(5):339–348.

Sadrmomtazi, A., Sobhani, J., and Mirgozar, M. (2013). Modeling compressive strength of EPS lightweight concrete using regression, neural network and ANFIS. *Construction and Building Materials*, 42:205–216.

Sarıdemir, M. (2009). Prediction of compressive strength of concretes containing metakaolin and silica fume by artificial neural networks. *Advances in Engineering Software*, 40(5):350 – 355.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117.

Tam, C. T., Lim, T. Y., Sri Ravindrarajah, R., and Lee, S. L. (1987). Relationship between strength and volumetric composition of moist-cured cellular concrete. *Magazine of Concrete Research*, 39(138):12–18.

Wang, N. and Adeli, H. (2015). Self-constructing wavelet neural network algorithm for nonlinear control of large structures. *Engineering Applications of Artificial Intelligence*, 41:249–258.

Xue, Y. and Li, Y. (2018). A Fast Detection Method via Region-Based Fully Convolutional Neural Networks for Shield Tunnel Lining Defects. *Computer-Aided Civil and Infrastructure Engineering*, 0(0).

Yaseen, Z. M., Deo, R. C., Hilal, A., Abd, A. M., Bueno, L. C., Salcedo-Sanz, S., and Nehdi, M. L. (2018). Predicting compressive strength of lightweight foamed concrete using extreme learning machine model. *Advances in Engineering Software*, 115:112–125.

Yeh, I.-C. (1998). Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete Research*, 28(12):1797 – 1808.

Yeh, I.-C. (2006). Analysis of strength of concrete using design of experiments and neural networks. *Journal of Materials in Civil Engineering*, 18(4):597–604.

Yu, B., Wang, H., Shan, W., and Yao, B. (2018). Prediction of bus travel time using random forests based on near neighbors. *Computer-Aided Civil and Infrastructure Engineering*, 33(4):333–350.

Zhang, Y. and Ge, H. (2013). Freeway travel time prediction using takagi–sugeno–kang fuzzy neural network. *Computer-Aided Civil and Infrastructure Engineering*, 28(8):594–603.

Zhao, Z. and Ren, L. (2002). Failure Criterion of Concrete under Triaxial Stresses Using Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 17(1):68–73.

Author Manuscript

This article is protected by copyright. All rights reserved.