Journal of Soft Computing in Civil Engineering 7-1 (2023) 114-134

Contents lists available at **SCCE**

Journal of Soft Computing in Civil Engineering

Journal homepage: www.jsoftcivil.com

Deep Neural Network Regression with Advanced Training Algorithms for Estimating the Compressive Strength of Manufactured-Sand Concrete

Nhat-Duc Hoang[1,2*], Van-Duc Tran[2,3]

1. Lecturer, Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam
2. Lecturer, Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam
3. Lecturer, International School - Duy Tan University, Da Nang, 550000, Vietnam

Corresponding author: *hoangnhatduc@duytan.edu.vn*

https://doi.org/10.22115/SCCE.2022.349837.1485

ARTICLE INFO

Article history:
Received: 09 July 2022
Revised: 25 October 2022
Accepted: 26 December 2022

Keywords:
Compressive strength;
Manufactured-sand concrete;
Deep learning;
Neural network;
Advanced optimizers.

ABSTRACT

Manufactured sand has high potential for replacing natural sand and reducing the negative impact of the construction industry on the environment. This paper aims at developing a novel deep learning-based approach for estimating the compressive strength of manufactured-sand concrete. The deep neural networks are trained by the advanced optimizers of Root Mean Squared Propagation, Adaptive Moment Estimation, and Adaptive Moment Estimation with Nesterov momentum (Nadam). In addition, the activation functions of logistic sigmoid, hyperbolic tangent sigmoid, and rectified linear unit activation are employed. A dataset including 132 samples has been used to train and verify the deep neural networks. Stone powder content, sand ratio, quantity of cement, quantity of water, quantity of coarse aggregate, quantity of water-reducer, quantity of manufactured sand, concrete slump, unit weight of concrete, and curing age are utilized as predictor variables. Based on experiments, the Nadam-optimized model used with the sigmoid activation function has achieved the most desired performance with root mean square error (RMSE) = 1.95, mean absolute percentage error (MAPE) = 3.04%, and coefficient of determination (R^2) = 0.97. Thus, this neural computing model is recommended for practical purposes because it can help to mitigate the time and cost dedicated to laboratory work.

How to cite this article: Hoang ND, Tran VD. Deep neural network regression with advanced training algorithms for estimating the compressive strength of manufactured-sand concrete. J Soft Comput Civ Eng 2023;7(1):114–134. https://doi.org/10.22115/scce.2022.349837.1485

1. Introduction

Concrete has been extensively employed in the construction industry because it features many advantageous engineering properties. When combined with steel reinforcement, reinforced concrete achieves high strength and durability. In addition, concrete material offers good resistance to water and high temperature. Usual concrete mixtures include binding material (e.g. Portland cement), coarse aggregate, fine aggregate, and water. The components of a concrete mixture is typically not costly and can be easily assessed. The aforementioned features of concrete make this construction material highly suitable for a wide range of civil and infrastructure projects [1,2].

In Vietnam as well as in other countries around the globe, demand for sand rises at a fast pace due to the rapid infrastructure development. Therefore, natural sources of sand barely satisfy the domestic demand and sand dearth becomes apparent [3,4]. Accordingly, researchers and practicing engineers have resorted to using manufactured sand made from crushed rocks (e.g. granite, basalt, and other sand stones) as an alternative to natural sand [5,6].

Since concrete using manufactured sand is highly potential for solving the issue of sand dearth and mitigating the effect of the construction industry on the natural environment, various studies have dedicated to the investigation of this material's mechanical properties [7,8] . In concrete design, compressive strength (CS)is widely regarded as the most crucial index [9–11]. Other properties such as elastic modulus and water tightness can be inferred via their correlations with the CS [12]. Estimating the CS of a concrete mixture containing manufactured sand based on its components is particularly important for mixture design. It is because if this parameter is correctly predicted, time and cost dedicated to laboratory works can be reduced or even avoided [13,14].

Nevertheless, estimation of CS is a challenging task. The reason is that this mechanical property is dependent on various factors such as mix proportions and concrete age. Concrete is a highly nonhomogeneous material with a diverse set of constituents. Furthermore, the mapping function between the CS of a concrete mix and its components has been generally demonstrated to be complex and nonlinear [15]. Therefore, conventional regression analysis and equation-based models used for estimating the CS of concrete mixes often fall short of the industry's requirements [16–19].

In recent years, with the advancements of machine learning (ML) and computing power, researchers have increasingly relied on intelligent data-driven approach for predicting CS of concrete mixes based on their constituents and age [20,21]. ML-based models have demonstrated promising capabilities in capturing the nonlinear and multivariate relationships between concrete strength and its influencing factors. State-of-the-art regression analysis approaches such as artificial neural networks, fuzzy neural network, deep neural computing, boosting machines, ensembles of decision trees, etc. can not only learn these functional relationships with a high degree of precision in the learning phase but also perform well in the estimation of unseen data in the testing phase [22–31].

Artificial neural network (ANN) has been used in [32] to estimate the CS prediction of environmentally friendly concrete. ANN and adaptive fuzzy neural inference system (ANFIS) have been used in [33] to construct models for predicting the CS of regular and high-performance concretes. The authors compare different training schemes including the Grey Wolf Optimizer metaheuristic and the Levenberg-Marquardt (LM) algorithm. It is experimentally found that the ANN model trained with the LM algorithm achieves the most desired outcome. Czarnecki et al. [20] presents an integration of the self-organizing feature map and ANN to predict the CS of cementitious composites with ground granulated blast furnace slag.

Shahmansouri et al. [34] and Moradi et al. [35] both demonstrate the potentiality of ANN in modeling the CS of concrete mixes. The former work shows that ANN can achieve predictive performance that is better than that of gene expression programming. The latter once again confirms the finding of [33] which shows good outcomes obtained from the LM-based ANN model. Nevertheless, one notable disadvantage of the LM algorithm is that it requires the computation and storage of the Jacobian matrices. These matrices often become enormously large for big datasets and deep neural networks that involve multiple hidden layers. In addition, limitations of the conventional shallow backpropagation ANN in modeling complex engineering processes were also pointed out in [28–36].

Golafshani and Behnood [37] proposes a novel integration of ANN and multi-verse optimizer for predicting mechanical properties of sustainable concrete containing waste foundry sand. The capability of neural networks to model complex estimation tasks in civil engineering was demonstrated in [38]. Faraj et al. [39] constructs a data-driven approach for inferring the CS of eco-friendly self-compacting concrete incorporating ground granulated blast furnace; the ANN has been used as the function approximator and has achieved a good correlation of determination with $R^2 = 0.955$. Rezazadeh et al. [40] recently demonstrated the superiority of ANN over the Genetic Programming and the Combinatorial Group Method of Data Handling approaches.

Ahmed et al. [41] investigates the capability of an ANN model and a M5P-tree for predicting the CS of geopolymer concrete incorporated with nano-silica. Pan et al. [42] successfully integrates genetic algorithm (GA) and ANN to establish a hybrid intelligent model for estimating the CS of green concrete. Zhang et al. [43] develops a model for predicting the mechanical properties of manufactured-sand concrete using tree-based models. These tree-based models include regression tree, random forest, and gradient boosted regression tree. In addition, the Firefly algorithm (FA) has been integrated with the tree-based models to optimize their model selection phases. Although the hybrid GA-ANN and FA-tree models demonstrate good predictive outcomes, the main concern of the proposed framework is the high computational cost required for training or optimizing the prediction models. It is because both GA and FA are population-based metaheuristics. Therefore, a large number of function evaluations is required to adapt the ML-based models.

Recently, deep artificial neural network regression (DANNR) has gained the increasing attention of researchers in the field of modeling concrete strength. The basic idea of a deep ANN is to create a network with a deep hierarchical organization of hidden layers. Each layer can distill and generalize data from the previous layers to more informative signals that is transferred to the subsequent layers. Each hidden layer plays the role of a feature learning or engineering operator. In a normal shallow ANN, there is only one hidden layer that extracts and learns the feature from the input data. Meanwhile, in a DANNR, various feature learners can be stacked to generate increasingly informative signals used for function approximation. With such advantages, DANNR-based models are capable of coping with nonlinear and multivariate datasets [44].

Accordingly, ML-based models for estimating the CS of concrete incorporating waste marble powder have been recently put forward in [45]. The authors rely on a neural computing model with 3 hidden layers and this deep ANN model is trained by the Adaptive Moment Estimation (Adam). This deep ANN model demonstrates a competitive performance compared to the Extreme Gradient Boosting Machine (XGBoost). However, this study has not explored the potentiality of other state-of-the-art optimizers (e.g. Root Mean Squared Propagation or Adaptive Moment Estimation with Nesterov momentum) for training the deep ANN model. Haque et al. [46] relies on a DANNR for estimating the strength of fly ash-based magnesium phosphate cement mortar. This study shows the desired performance of a deep ANN with two hidden layers and the hyperbolic tangent sigmoid activation function. Nevertheless, the effectiveness of other activation functions has not been explored in this paper. Asghari et al. [44] proves the superiority of DANNR-based models in predicting the undrained shear strength of clays; the DANNR-based models have outperformed conventional regression and equation-based approaches.

According to the existing works, an increasing trend of utilizing sophisticated ML models and deep neural networks for predicting the CS of concrete can be observed. However, few studies have explored the potentiality of advanced gradient descent based-optimizers for training DANNR models. With such motivations, this study aims to compare the performances of DANNR models using different advanced optimizers in estimating the CS of concrete containing manufactured sand. The optimizers of the Adam, Root Mean Squared Propagation (RMSprop) and Adaptive Moment Estimation with Nesterov momentum (Nadam) are employed. Although deep learning has been used to estimate the CS of concrete, few studies have been dedicated to comparing the performance of different advanced optimizers used for training DANNR-based CS prediction models. Therefore, the current work is an attempt to fill this gap in the literature.

The subsequent sections of the study are presented as follows: The next section summarizes the research method that covers the DANNR, the used optimizers, and the employed datasets of concrete containing manufactured sand. The third part presents the findings of the current work. The conclusion is provided in the final section.

2. Research method

2.1. Deep artificial neural network regression (DANNR)

A deep neural network model generally comprises an input layer, a set of hidden layers, and an output layer [47]. The input layer is basically an external signal receiver and the output layer simply processes the results of the last hidden layer and yields the predicted dependent variable (e.g CS). In deep neural networks, there are multiple hidden layers containing neurons for processing a dataset and generalizing a mapping function between the input signal x (e.g concrete constituents and age) and the output y (e.g CS). Generally, the stacked hidden layers enhance the network's robustness in generalizing non-linear mapping functions; the suitable numbers of hidden layers and neurons in each hidden layer are data-dependent and should be determined experimentally [48]. A typical DANNR's structure is depicted in **Fig. 1**. Herein, there are D input variables (x_1,x_2,…,x_D) that represent the characteristics of a concrete mix.

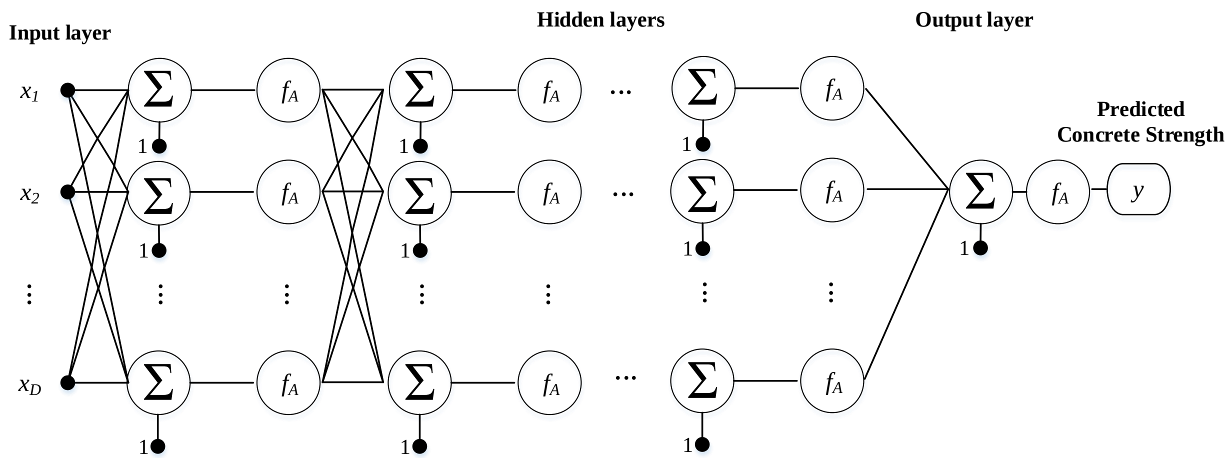


Fig. 1. A DANNR's structure.

Table 1
Activation functions used in a DANNR.

Activation function	Formula	Derivative
Logistic sigmoid	$f(x)=\dfrac{1}{1+\exp(-x)}$	$f'(x)=f(x)\times(1-f(x))$
Hyperbolic tangent sigmoid	$f(x)=\dfrac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$	$f'(x)=1-f^2(x)$
Rectified linear unit activation	$f(x)=\begin{cases} x, & if\ x>0 \\ 0, & if\ x\le 0\end{cases}$	$f'(x)=\begin{cases} 1, & if\ x>0 \\ 0, & if\ x\le 0\end{cases}$
Linear	$f(x)=x$	$f'(x)=1$

In Fig. 1, f_A denotes an activation function. In the input and hidden layers, nonlinear activation functions are often used to learn a nonlinear target function. The commonly used activation functions may include logistic sigmoid, hyperbolic tangent sigmoid, and rectified linear unit activation (ReLU) [47–49]. For a DANNR that performs function approximation tasks, the output layer simply employs a linear activation function. The used activation functions and their derivatives are summarized in Table 1.

To train a DANNR model used for estimating the CS of concrete mixes consisting of manufactured sand, the back-propagation and gradient-descent algorithms are used to adapt the connection weights between different layers. Herein, the connection weights in each layer is stored in the form of a matrix W. In the forward pass, the input layer obtains the signals representing the characteristics of a concrete mix (x) and transfers them through the hidden layers and the output layer. The output layer yields the estimated CS (y). To reduce the error (ε) between the observed CS (t) and the predicted one (y), the backward pass is performed. In the backward pass, ε is reversely transmitted to each precedent layer and the network's weights W are optimized via the gradient-descent algorithm.

The back-propagation algorithm requires the calculation of the gradient of a loss function which is used to determine the direction and the amount of the update for each weighting value [49]. For regression analysis, the commonly-used loss function is the Squared Error Loss (SEL) [35,36,45–50,37–44]. This loss function basically yields the squared difference between the actual and predicted CS. The SEL is given by:

$$SEL(t,y) = \frac{1}{2}(t-y)^2 \tag{1}$$

Additionally, a common problem faced during the training phase of a DANNR is how to alleviate overfitting. This phenomenon usually occurs when a deep learning model performs exceptionally well in the model construction phase but the model's estimations of unseen data are highly inaccurate. One effective method for mitigating overfitting is weight regularization. This approach prevents overfitting by constraining the magnitude of the network's weights. To do so, additional terms are included in the loss function [49–52]. Generally, there are two forms of weight regularization: L1 and L2. In the former case, the L1-norm of a network weight w is added to the loss function. In the latter case, the L2-norm of a network weight w is used. Accordingly, the modified loss functions are given by:

L1-norm: $$L(t,y) = SEL(t,y) + \lambda \|w\|_1 \tag{2}$$

L2-norm: $$L(t,y) = SEL(t,y) + \lambda \|w\|_2^2 \tag{3}$$

where λ is a hyper-parameter of the loss functions. A large λ significantly prohibits a large value of the network weight.

Using the backpropagation framework, the partial derivatives of the loss function with respect to each connection weight must be specified. The readers are guided to the previous works of [53] and [47] to acquire the equations employed for adapting a model's weights. Herein, we focus on the equation used to update the connection weights in the output layer. These connection weights are directly associated with the derivative of the loss function $L()$ with respect to the predicted output variable y. In detail, if the standard loss function $L(\varepsilon)$ = SEL(ε) is utilized, the partial derivative of $L(\varepsilon)$ with respect to the i^{th} connection weight in the output layer w_i^O is presented as follows:

$$\frac{\partial L}{\partial w_i^O} = \frac{\partial L}{\partial y} \times \frac{\partial y}{\partial w_i^O} = -(t-y) \times \frac{\partial y}{\partial w_i^O} \tag{4}$$

2.2. Advanced optimizers for training deep neural networks

2.2.1. Root mean squared propagation (RMSprop)

RMSprop, described in [54], is an improved gradient descent algorithm with the use of adaptive learning rate. Herein, the gradient at time step *t* ($g_t = \partial L / \partial w$) is divided by a running average of its magnitude [49]. This running average at time step *t* is given by:

$$v(t) = \gamma \times v(t-1) + (1-\gamma) g_{t-1}^2 \tag{5}$$

where $\gamma \in (0,1)$ and g_{t-1}^2 denotes the element-wise square of the gradient g_t.

The equation used to revise the model's weights is given by:

$$w(t+1) = w(t) - \alpha \times \frac{g_t}{\sqrt{|v_t|} + \zeta} \tag{6}$$

where α is the learning rate; $\zeta = 1e-8$ is a small number to guarantee the numerical stability of the calculation process.

2.2.2. Adaptive moment estimation (Adam)

The Adam optimizer, presented in [55], utilizes the estimation of the first and second moments of the gradient via exponential moving averages and bias corrections. This algorithm also employs an exponentially decaying average of past gradients [56]. To update the network's weights, it is first required to compute the 1st biased moment estimation as follows:

$$m_t = \beta_1 \times m_{t-1} + (1-\beta_1) \times g_t \tag{7}$$

where $\beta_1 = 0.9$.

The 2nd biased moment estimation is given by:

$$v_t = \beta_2 \times v_{t-1} + (1-\beta_2) \times g_t^2 \tag{8}$$

where $\beta_2 = 0.9999$ is a hyper-parameter of the algorithm.

The bias-corrected 1st moment estimate is revised in the following manner:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{9}$$

The bias-corrected 2nd moment estimate is obtained via:

$$\hat{v}_t = \frac{v}{1-\beta_2^t} \tag{10}$$

Accordingly, the optimized parameters of a neural computing model are adapted via:

$$w_t = w_{t-1} - \alpha \times \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \zeta} \tag{11}$$

2.2.3. Nesterov-accelerated adaptive moment estimation (Nadam)

The Nadam [57] combines Nesterov accelerated gradient and the Adam optimizer. Herein, Nesterov momentum is used to consider the gradient at the projected future position [56]. Therefore, the Nadam optimizer can be effective to perform the searching process in regions of the loss function where the gradient is flat. The equation used to update the network's weight according to the Nadam algorithm is given by:

$$w_t = w_{t-1} - \alpha \times \frac{\hat{m}_t}{\sqrt{\hat{n}_t} + \zeta} \quad (12)$$

where α is also the learning rate parameter; $\zeta = 1e-8$.

The revised 1st moment estimate $\hat{m}_t$ is given by:

$$\hat{m}_t = (\mu \times m_t / (1 - \prod_{i=1}^{t+1} \mu_i)) + ((1-\mu) \times g_t / (1 - \prod_{i=1}^{t+1} \mu_i)) \quad (13)$$

where $m_t = (1-\mu) \times g_t + \mu \times m_{t-1}$ and $\mu = 0.975$ denotes a hyper-parameter.

The 2nd biased moment estimation n_t and corrected moment estimation $\hat{n}_t$ are given by:

$$n_t = \upsilon \times n_{t-1} + (1-\upsilon) \times g_t^2 \quad (14)$$

$$\hat{n}_t = \frac{\upsilon \times n_t}{1-\upsilon^t} \quad (15)$$

where $\upsilon = 0.999$ is a hyper-parameter of the Nadam algorithm.

2.3. The collected dataset

The dataset consisting of testing results of manufactured-sand concrete samples has been collected and compiled in the previous works of [58] and [59]. The aforementioned works carried out experimental studies on the development of CS of concrete containing manufactured sand. There are 132 testing records that provide the concrete mixes' constituents and the CS corresponding to different curing ages. The input factors of stone powder content, sand ratio, quantity of cement, quantity of water, quantity of coarse aggregate, quantity of water-reducer, quantity of manufactured sand, concrete slump, unit weight of concrete, and curing age are used as independent variables to estimate the CS as a dependen variable. Herein, the manufactured sand is obtained from crushed limestone with the particle size of 0–4.75 mm. Table 2 provides the detailed information on the CS and its predictor variables.

Table 2
Statistical description of the collected dataset.

Variables	Note	Min	Average	Std.	Skewness	Max
X_1	Stone powder content (%)	5.000	9.000	3.278	0.000	13.000
X_2	Sand ratio (%)	34.000	37.636	3.999	0.883	44.000
X_3	Quantity of cement (kg/m^3)	321.000	397.182	52.638	-0.324	462.000
X_4	Quantity of water (kg/m^3)	180.000	181.364	2.235	1.032	185.000
X_5	Quantity of coarse aggregate (kg/m^3)	1091.000	1166.182	46.306	-1.019	1197.000
X_6	Quantity of water-reducer (kg/m^3)	2.247	2.997	0.358	-0.122	3.696
X_7	Quantity of manufactured sand (kg/m^3)	613.000	707.091	96.040	0.799	858.000
X_8	Concrete slump (mm)	30.000	75.909	42.506	1.035	160.000
X_9	Unit weight of concrete (kg/m^3)	2410.000	2443.758	16.647	-0.889	2463.000
X_{10}	Curing age (day)	3.000	132.159	120.968	0.703	388.000
Y	Compressive strength (CS) (MPa)	28.500	55.840	11.793	-0.250	78.200

2.4. The metrics used for performance measurement

To evaluate of the performance of the deep learning models used in this paper, a set of three indicators are considered; they include coefficient of determination (R^2), root mean square error (RMSE), and mean absolute percentage error (MAPE). These indicators are widely used for assessing the predictive capability of regression models [45,46,55–60,47–54]. The equations used to compute these three indicators are presented in Table 3. It is worth noticing that that the closer the R^2 to 1, the better the prediction outcome. In addition, small values of RMSE and MAPE reflect low prediction errors. R^2 and MAPE are unitless. Meanwhile, the unit of the RMSE is MPa.

Table 3
The employed performance indicators.

Indices	Equation	Range	Ideal value
R^2	$R^2 = 1 - \frac{\sum_{i=1}^{N}(t_i - y_i)^2}{\sum_{i=1}^{N}(t_i - \bar{t})^2}$	(0,1)	1
RMSE	$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - t_i)^2}$	(0,+∞)	0
MAPE	$MAPE = \frac{100}{N} \times \sum_{i=1}^{N}\frac{\lvert y_i - t_i \rvert}{y_i}$	(0,+∞)	0

Note: t_i and y_i are the observed and predicted CS values of the i^{th} data instance. N denotes the number of data records.

3. Experimental results and discussion

This part of the paper presents the experimental outcomes of the DANNR used for predicting the CS of manufactured-sand concrete. The deep learning models use three activation functions in the hidden layers: the logistic sigmoid (Sigmoid), the hyperbolic tangent sigmoid (Tanh), and rectified linear unit activation (ReLU). The state-of-the-art RMSprop, Adam, and Nadam optimizers are used to adapt the deep networks with respect to the collected dataset. Hence, there are nine DANNR models (RMSprop-Sigmoid-DANNR, RMSprop-Tanh-DANNR, RMSprop-Tanh-ReLU, Adam-Sigmoid-DANNR, Adam-Tanh-DANNR, Adam-Tanh-ReLU, Nadam-Sigmoid-DANNR, Nadam-Tanh-DANNR, and Nadam-Tanh-ReLU) are constructed and used for result comparison. It is noted that the employed deep learning models have been developed in MATLAB programming environment. In addition, the computational experiments in this study are performed with a desktop computer using the Intel(R) Core(TM) i7-10700F CPU @ 2.90GHz and 16GB RAM.

As mentioned earlier, the dataset includes 132 records and ten predictor variables. These predictor variables provide information on the concrete mix and curing age with respect to the output variable of the CS. In this study, to standardize the ranges of the predictor and predicted variables, the Z-score normalization equation is used. Thus, the orginal variables are normalized as follows:

$$X_Z = \frac{X_O - \mu_X}{\sigma_X} \tag{16}$$

where X_Z and X_O denote the standardized and the original variables, respectively. μ_X and σ_X are the mean and standard deviation of the original variable.

The aforementioned deep learning models are trained by the stochastic gradient descent method with the three optimizers (RMSprop, Adam, and Nadam). The batch-size used in the stochastic gradient descent method is 16. In addition, the deep learning models have been trained during 500 epochs. Furthermore, the DANNR models require the setting of their hyper-parameters including the learning rate, the regularization type (L1 or L2), the regularization parameter, the number of hidden layers as well as the number of neurons in each hidden layer. This study has carried out a five-fold cross validation process [61] to identify suitable settings of the DANNR models. Based on this cross validation process, the suitable learning rate and regularization parameter are 0.01 and 0.001, respectively. In addition, the configurations of the DANNR models are summarized in Table 4.

Table 4
Configurations of the DANNR models.

Models	Regularization type	Number of hidden layers	Number of neurons in each hidden layer
RMSprop-Sigmoid-DANNR	L2	3	12
RMSprop-Tanh-DANNR	L1	2	10
RMSprop- ReLU-DANNR	L2	2	10
Adam-Sigmoid-DANNR	L2	4	12
Adam-Tanh-DANNR	L2	2	8
Adam-ReLU-DANNR	L1	2	8
Nadam-Sigmoid-DANNR	L2	3	12
Nadam-Tanh-DANNR	L1	2	10
Nadam-ReLU-DANNR	L1	2	12

Table 5
Prediction results of the DANNR models.

Optimizer	Phase	Indices	Sigmoid-DANNR		Tanh-DANNR		ReLU-DANNR	
			Mean	Std	Mean	Std	Mean	Std
RMSprop	Training	RMSE	1.360	0.115	1.427	0.105	1.677	0.262
		MAPE (%)	2.009	0.195	2.077	0.184	2.409	0.363
		R^2	0.986	0.002	0.985	0.002	0.979	0.007
	Testing	RMSE	2.399	0.571	3.230	1.502	2.944	0.764
		MAPE (%)	3.720	1.056	4.060	1.605	4.580	1.658
		R^2	0.955	0.019	0.903	0.090	0.929	0.037
Adam	Training	RMSE	1.255	0.098	1.331	0.149	2.444	0.726
		MAPE (%)	1.863	0.149	1.951	0.234	3.527	1.052
		R^2	0.989	0.002	0.987	0.003	0.953	0.028
	Testing	RMSE	2.098	0.540	2.881	1.167	3.203	1.035
		MAPE (%)	3.105	0.763	4.193	1.685	4.798	1.999
		R^2	0.958	0.032	0.921	0.085	0.912	0.045
Nadam	Training	RMSE	1.364	0.075	1.452	0.130	2.320	0.393
		MAPE (%)	2.011	0.140	2.152	0.227	3.382	0.573
		R^2	0.986	0.001	0.985	0.003	0.960	0.013
	Testing	RMSE	1.952	0.683	2.698	0.773	3.847	1.128
		MAPE (%)	3.043	1.261	3.878	1.178	5.825	1.826
		R^2	0.970	0.018	0.922	0.050	0.861	0.095

Using the configurations identified by the cross validation processes, a repeated sampling of the collected data in which 90% of the dataset is used for model training and 10% of the dataset is used for model testing is carried out 20 times. This repeated sampling process aims at mitigating the bias in model evaluation due to the randomness in data selection. The prediction results of the nine DANNR models used for predicting the CS of manufactured-sand concrete are reported in **Table 5**. Overall, all deep learning models yield accurate and reliable predictions of the CS as shown by low values of RMSE and MAPE as well as high values of R^2. Nevertheless, the DANNR using the logistic sigmoid function and trained by the Nadam optimizer outperforms other benchmark approaches. The Nadam-Sigmoid-DANNR yields the highest predictive accuracy with the average RMSE = 1.952, MAPE = 3.043%, and R^2 = 0.97. Notably, R^2 is the proportion of the variation in the CS that can be estimated from the DANNR that uses the set of the ten predictor variables. This means that 97% of the total variation in the CS of manufactured-sand concrete can be explained by the deep learning model.

The Adam-Sigmoid-DANNR (with RMSE = 2.098, MAPE = 3.105%, R^2 = 0.958) and RMSprop-Sigmoid-DANNR (with RMSE = 2.399, MAPE = 3.725%, R^2 = 0.955) are the second and third best models, respectively. This fact point outs that DANNR used with the logistic sigmoid activation function is highly suitable with the dataset at hand. The DANNR with the ReLU activation function (RMSE = 2.944) is slightly better than the one with the Tanh function (RMSE = 3.230) when the RMSprop is used. However, when the DANNR models are trained by the Adam and Nadam algorithms, the models using the Tanh function always excel the ones using the ReLU function.

Table 6
The computational (com.) time of the DANNR models.

DANNR models	RMSprop -Sigmoid	RMSprop-Tanh	RMSprop-ReLU	Adam-Sigmoid	Adam-Tanh	Adam-ReLU	Nadam-Sigmoid	Nadam-Tanh	Nadam-ReLU
Average com. time (s)	2.19	1.60	1.59	2.51	1.55	1.58	2.05	1.62	1.66

Moreover, the average computational time of the DANNR models is reported in Table 6. In general, the computational times of the DANNR models using the Sigmoid function are higher than those of other models. The training progresses of the deep learning models are demonstrated in Fig. 2. Apparently, the convergence rates of the deep learning models using Sigmoid and Tanh functions are faster than those of the models employing the ReLU function. Therefore, it can be observed that the Sigmoid and Tanh functions are more suitable for modeling the current dataset than the ReLU function. Compared to the ReLU function, the Sigmoid and Tanh activation functions can help attain better prediction accuracy with a slight increase in computational expense.

The boxplots illustrating the prediction performances of the DANNR models after 20 independent runs are shown in Fig. 3. Based on the boxplots, the RMSprop-Tanh-DANNR, Adam-ReLU-DANNR, and Nadam-ReLU-DANNR demonstrate relatively unstable performances; their ranges between the minimum RMSE and maximum RMSE are considerably wider compared to those of other models. Additionally, the median (shown as a red line) of the Nadam-Sigmoid-DANNR is the lowest among all of the employed models.

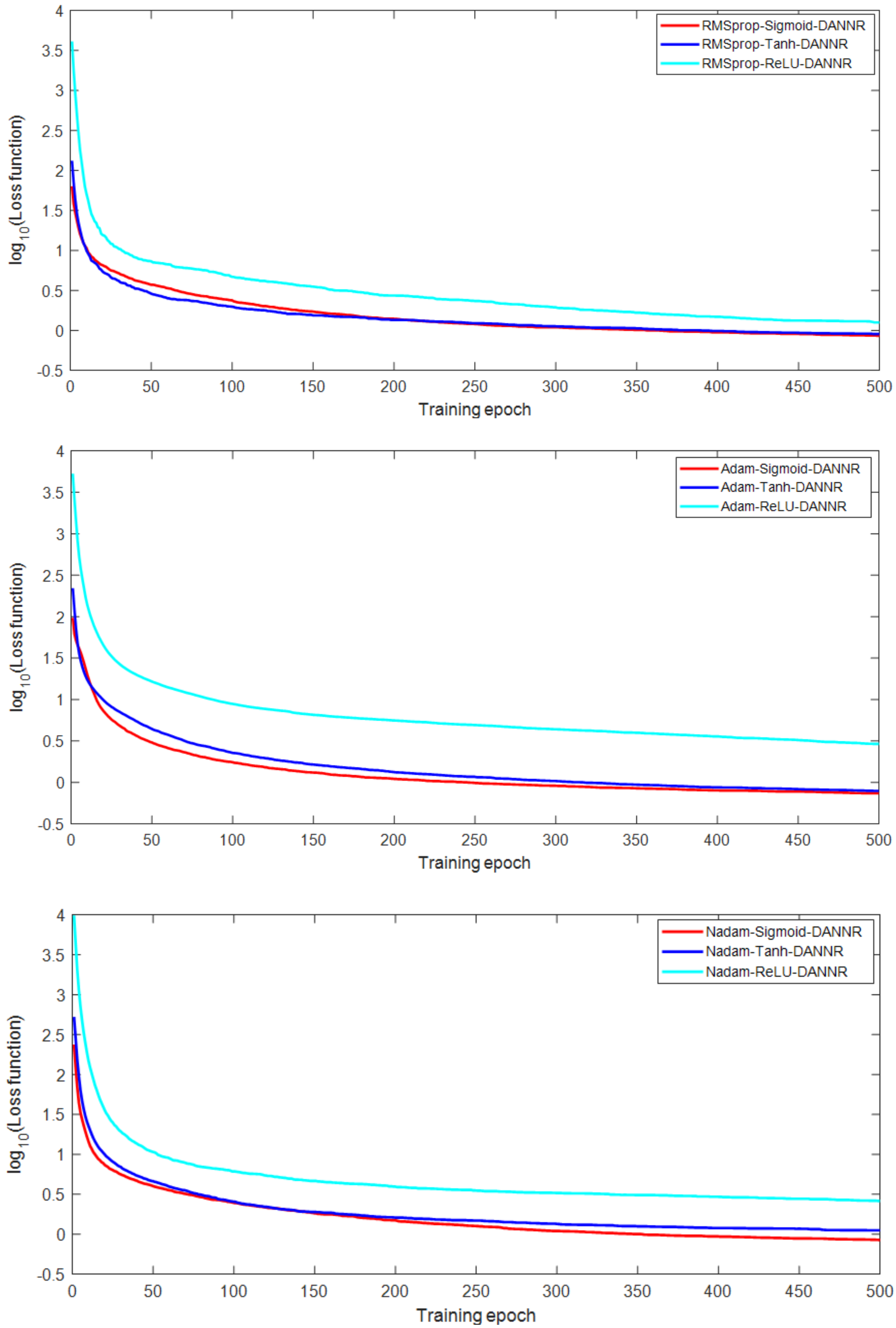
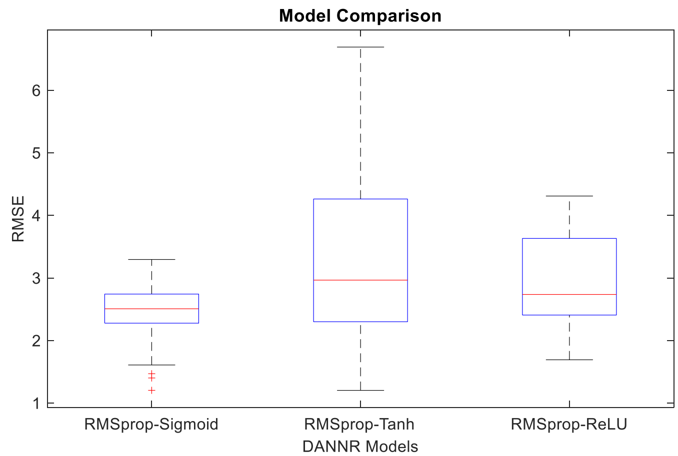
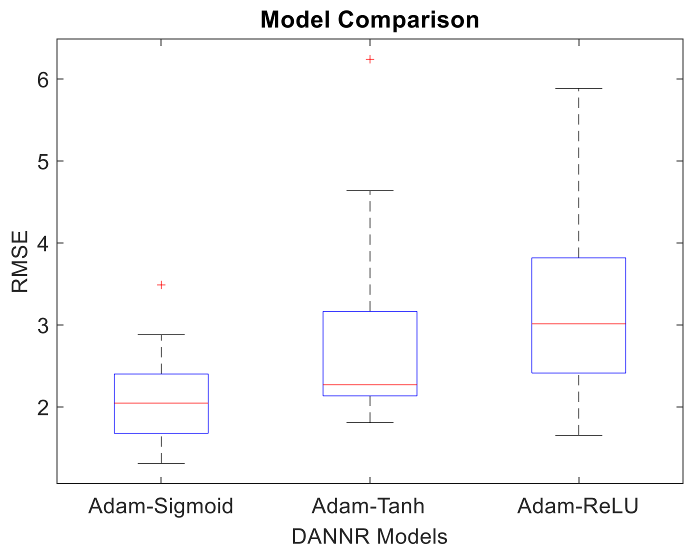


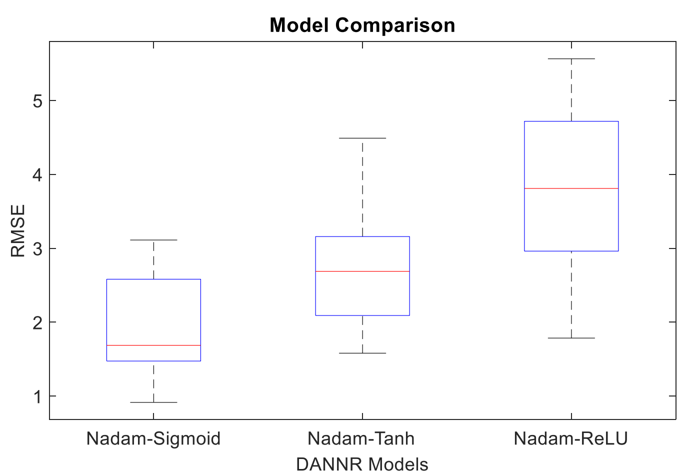
Fig. 2. Training progresses of the DANNR models.



(a)



(b)



(c)

Fig. 3. Boxplots of the models' performance obtained from 20 independent runs: (a) RMSprop optimizer, (b) Adam optimizer, and (c) Nadam optimizer.

As mentioned earlier, the Nadam-Sigmoid-DANNR has obtained a good prediction performance with RMSE = 1.952, MAPE = 3.043%, and R^2 = 0.97. This result can be benchmarked with machine learning models previously used for estimating the CS of manufactured sand concrete. In [19], Adaptive Neuro Fuzzy Inference System (ANFIS) and feedforward Artificial Neural Network (ANN) have been employed. ANFIS and ANN attain the RMSE = 6.46 and 7.67, respectively. In addition, Ly et al. [19] also enhances the ANFIS model by using the Teaching-Learning-Based Optimization (TLBO); the hybrid ANFIS-TLBO yields a better data fitting with RMSE = 4.93. The model proposed by Zhang et al. [43] combines gradient boosted regression tree (GBRT) and Firefly algorithm (FA); The latter algorithm is employed to optimize the tuning-parameters of the former algorithm. The model has yielded the RMSE = 3.346 [43]. Based on the results reported in the previous studies, it can be seen that the Nadam-Sigmoid-DANNR proposed in this study has provided a promising prediction performance in estimating the CS of manufactured sand concrete.

Fig. 4 and Fig. 5 demonstrate the goodness of fit obtained by the Nadam-Sigmoid-DANNR model. Although the proposed method has attained a high degree of fit, its results show certain deviations from the actual CS. The absolute deviations (or residuals) of the proposed deep learning model is demonstrated in Fig. 6. The histogram of the model's residuals is presented in Fig. 7. The maximum, minimum, and average residual are 8.4540 MPa, 0.001 MPa, and 1.4321 MPa, respectively. This discrepancy between the estimated and observed CS is understandable. It is because the prediction of CS is highly complex due to the nonlinear and multivariate nature of the estimation task [15]. Moreover, a certain degree of uncertainty always exists in the experimental and testing processes used to measure the CS value of a concrete mix.

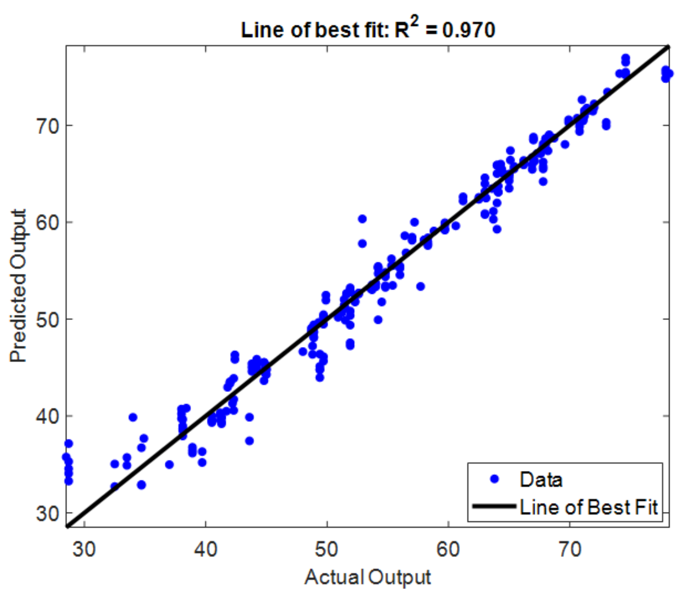


Fig. 4. Correlation between actual output and predicted output obtained by the Nadam-Sigmoid-DANNR model.

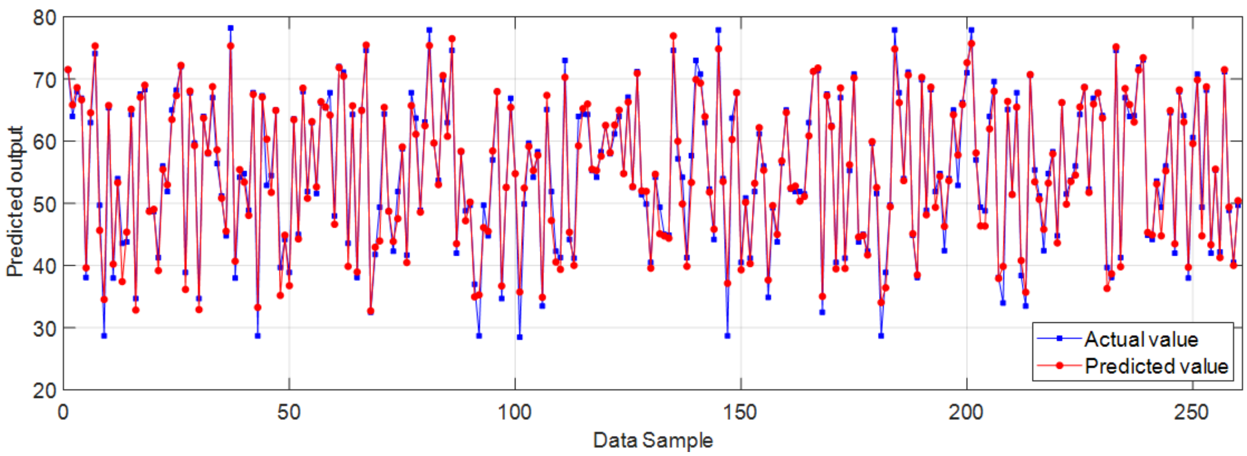


Fig. 5. Actual output vs. predicted output obtained by the Nadam-Sigmoid-DANNR.

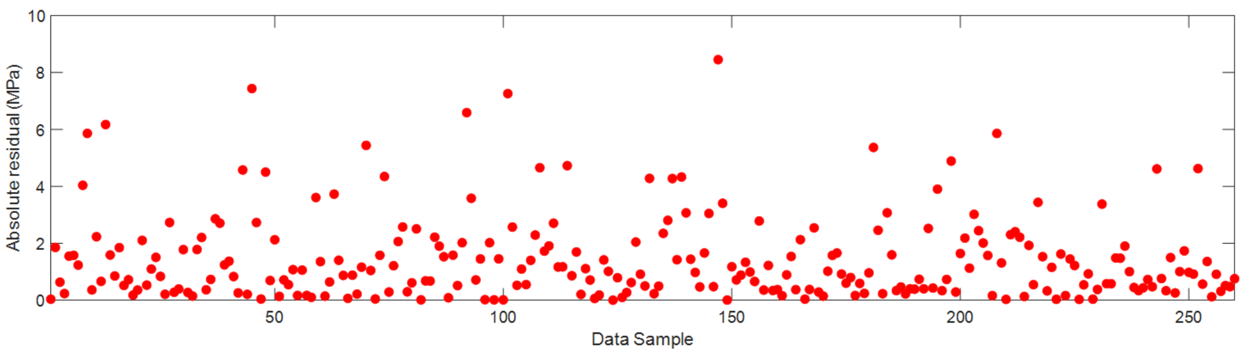


Fig. 6. Absolute residuals of the Nadam-Sigmoid-DANNR model.

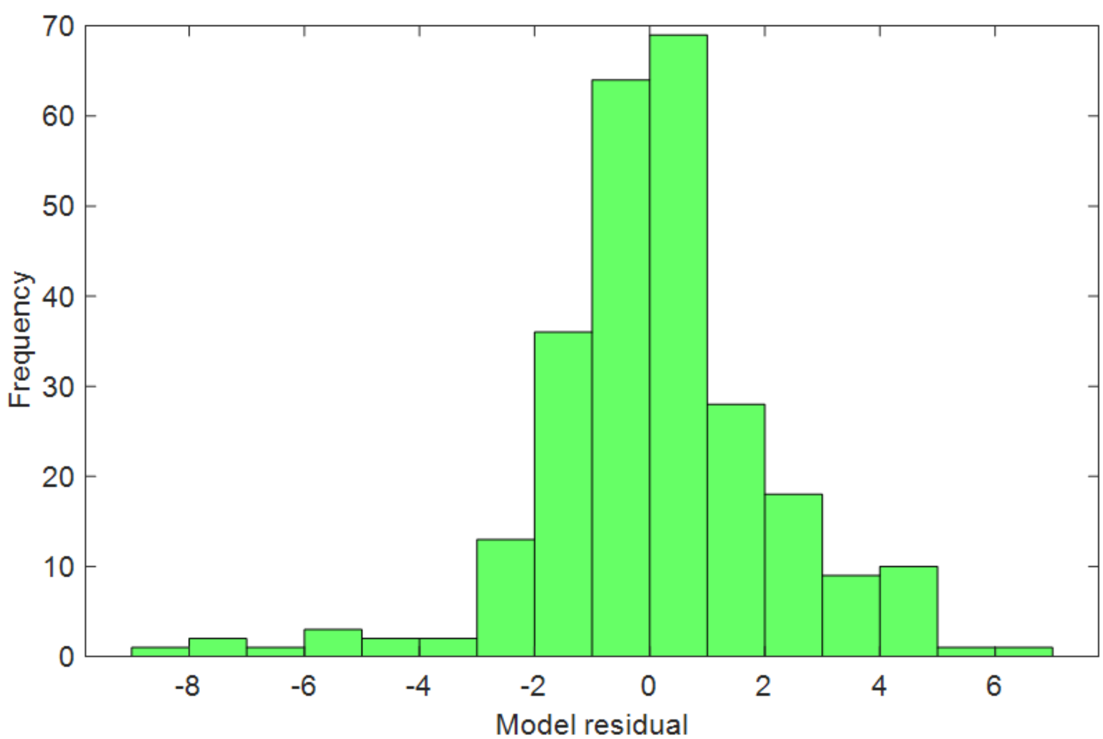


Fig. 7. Histogram of the Nadam-Sigmoid-DANNR model.

4. Conclusion

Prediction of the CS of manufactured-sand concrete based on its constituents and curing age is crucial for concrete mix design. In addition, the development of the CS at various ages is a complex phenomenon that involves the interplay of multiple predictor variables. Although various machine learning methods have been put forward to construct data-driven tools for concrete strength estimation, few studies have investigated the capabilities of deep neural network regression models in the task of interest.

This paper has proposed and verified a deep learning-based solution for achieving accurate estimations of the CS of manufactured-sand concrete. The DANNR models are trained with the advanced RMSprop, Adam, and Nadam optimizers. The research findings show that the Nadam-optimized DANNR with the Sigmoid activation function can help achieve the most accurate predictions of the CS with RMSE = 1.952, MAPE = 3.043%, and R^2 = 0.97. Therefore, the Nadam-Sigmoid-DANNR model is recommended for practical purposes because it can help to mitigate the time and cost dedicated to laboratory work.

Funding

This research received no external funding.

Conflicts of interest

The authors declare no conflict of interest.

Authors contribution statement

NHD: Conceptualization; Methodology; Software; Writing – original draft; Writing – review & editing. VDT: Data processing; Supervision ; Writing – review & editing.

References

[1] Nguyen H, Vu T, Vo TP, Thai H-T. Efficient machine learning models for prediction of concrete strengths. Constr Build Mater 2021;266:120950. https://doi.org/10.1016/j.conbuildmat.2020.120950.

[2] Mohe NS, Shewalul YW, Agon EC. Experimental investigation on mechanical properties of concrete using different sources of water for mixing and curing concrete. Case Stud Constr Mater 2022;16:e00959. https://doi.org/10.1016/j.cscm.2022.e00959.

[3] Kavya A, Venkateshwara Rao A. Experimental investigation on mechanical properties of concrete with M-sand. Mater Today Proc 2020;33:663–7. https://doi.org/10.1016/j.matpr.2020.05.774.

[4] T. V. Vietnam's Mekong Delta province faces serious sand dearth, Tuoi Tre News n.d.

[5] Needhidasan S, Ramesh B, Joshua Richard Prabu S. Experimental study on use of E-waste plastics as coarse aggregate in concrete with manufactured sand. Mater Today Proc 2020;22:715–21. https://doi.org/10.1016/j.matpr.2019.10.006.

[6] Karthik S, Rao PRM, Awoyera PO. Strength properties of bamboo and steel reinforced concrete containing manufactured sand and mineral admixtures. J King Saud Univ - Eng Sci 2017;29:400–6. https://doi.org/10.1016/j.jksues.2016.12.003.

[7] Nanthagopalan P, Santhanam M. Fresh and hardened properties of self-compacting concrete produced with manufactured sand. Cem Concr Compos 2011;33:353–8. https://doi.org/10.1016/j.cemconcomp.2010.11.005.

[8] Li H, Huang F, Cheng G, Xie Y, Tan Y, Li L, et al. Effect of granite dust on mechanical and some durability properties of manufactured sand concrete. Constr Build Mater 2016;109:41–6. https://doi.org/10.1016/j.conbuildmat.2016.01.034.

[9] Akin OO, Abejide OS. Modelling of concrete compressive strength admixed with GGBFS using gene expression programming. J Soft Comput Civ Eng 2019;3:43–53. https://doi.org/10.22115/scce.2019.178214.1103.

[10] Hoang N-D. Machine Learning-Based Estimation of the Compressive Strength of Self-Compacting Concrete: A Multi-Dataset Study. Mathematics 2022;10:3771. https://doi.org/10.3390/math10203771.

[11] Ahmadi-Nedushan B. An optimized instance based learning algorithm for estimation of compressive strength of concrete. Eng Appl Artif Intell 2012;25:1073–81. https://doi.org/10.1016/j.engappai.2012.01.012.

[12] Pham A-D, Hoang N-D, Nguyen Q-T. Predicting Compressive Strength of High-Performance Concrete Using Metaheuristic-Optimized Least Squares Support Vector Regression. J Comput Civ Eng 2016;30. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000506.

[13] Latif SD. Concrete compressive strength prediction modeling utilizing deep learning long short-term memory algorithm for a sustainable environment. Environ Sci Pollut Res 2021;28:30294–302. https://doi.org/10.1007/s11356-021-12877-y.

[14] Zhao Y, Hu H, Song C, Wang Z. Predicting compressive strength of manufactured-sand concrete using conventional and metaheuristic-tuned artificial neural network. Measurement 2022;194:110993. https://doi.org/10.1016/j.measurement.2022.110993.

[15] Bui D-K, Nguyen T, Chou J-S, Nguyen-Xuan H, Ngo TD. A modified firefly algorithm-artificial neural network expert system for predicting compressive and tensile strength of high-performance concrete. Constr Build Mater 2018;180:320–33. https://doi.org/10.1016/j.conbuildmat.2018.05.201.

[16] Chou J-S, Chiu C-K, Farfoura M, Al-Taharwa I. Optimizing the Prediction Accuracy of Concrete Compressive Strength Based on a Comparison of Data-Mining Techniques. J Comput Civ Eng 2011;25:242–53. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000088.

[17] Khademi F, Akbari M, Jamal SM, Nikoo M. Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete. Front Struct Civ Eng 2017;11:90–9. https://doi.org/10.1007/s11709-016-0363-9.

[18] Deshpande N, Londhe S, Kulkarni S. Modeling compressive strength of recycled aggregate concrete by Artificial Neural Network, Model Tree and Non-linear Regression. Int J Sustain Built Environ 2014;3:187–98. https://doi.org/10.1016/j.ijsbe.2014.12.002.

[19] Ly, Pham, Dao, Le, Le, Le. Improvement of ANFIS Model for Prediction of Compressive Strength of Manufactured Sand Concrete. Appl Sci 2019;9:3841. https://doi.org/10.3390/app9183841.

[20] Czarnecki S, Shariq M, Nikoo M, Sadowski Ł. An intelligent model for the prediction of the compressive strength of cementitious composites with ground granulated blast furnace slag based on ultrasonic pulse velocity measurements. Measurement 2021;172:108951. https://doi.org/10.1016/j.measurement.2020.108951.

[21] Sharafati A, Naderpour H, Salih SQ, Onyari E, Yaseen ZM. Simulation of foamed concrete compressive strength prediction using adaptive neuro-fuzzy inference system optimized by nature-inspired algorithms. Front Struct Civ Eng 2021;15:61–79. https://doi.org/10.1007/s11709-020-0684-6.

[22] Young BA, Hall A, Pilon L, Gupta P, Sant G. Can the compressive strength of concrete be estimated from knowledge of the mixture proportions?: New insights from statistical analysis and machine learning methods. Cem Concr Res 2019;115:379–88. https://doi.org/10.1016/j.cemconres.2018.09.006.

[23] Song H, Ahmad A, Farooq F, Ostrowski KA, Maślak M, Czarnecki S, et al. Predicting the compressive strength of concrete with fly ash admixture using machine learning algorithms. Constr Build Mater 2021;308:125021. https://doi.org/10.1016/j.conbuildmat.2021.125021.

[24] Asteris PG, Skentou AD, Bardhan A, Samui P, Pilakoutas K. Predicting concrete compressive strength using hybrid ensembling of surrogate machine learning models. Cem Concr Res 2021;145:106449. https://doi.org/10.1016/j.cemconres.2021.106449.

[25] Cook R, Lapeyre J, Ma H, Kumar A. Prediction of Compressive Strength of Concrete: Critical Comparison of Performance of a Hybrid Machine Learning Model with Standalone Models. J Mater Civ Eng 2019;31. https://doi.org/10.1061/(ASCE)MT.1943-5533.0002902.

[26] Chopra P, Sharma RK, Kumar M, Chopra T. Comparison of Machine Learning Techniques for the Prediction of Compressive Strength of Concrete. Adv Civ Eng 2018;2018:1–9. https://doi.org/10.1155/2018/5481705.

[27] Thai H-T. Machine learning for structural engineering: A state-of-the-art review. Structures 2022;38:448–91. https://doi.org/10.1016/j.istruc.2022.02.003.

[28] Hoang N-D, Tran X-L, Huynh T-C. Prediction of Pile Bearing Capacity Using Opposition-Based Differential Flower Pollination-Optimized Least Squares Support Vector Regression (ODFP-LSSVR). Adv Civ Eng 2022;2022:1–25. https://doi.org/10.1155/2022/7183700.

[29] Nguyen H, Nguyen N-M, Cao M-T, Hoang N-D, Tran X-L. Prediction of long-term deflections of reinforced-concrete members using a novel swarm optimized extreme gradient boosting machine. Eng Comput 2022;38:1255–67. https://doi.org/10.1007/s00366-020-01260-z.

[30] Chalabi M, Naderpour H, Mirrashid M. Seismic resilience index for RC moment frames of school buildings using neuro-fuzzy approach. Nat Hazards 2022;114:1–26. https://doi.org/10.1007/s11069-022-05377-w.

[31] Garg C, Singhal A, Singh P, Namdeo A, Rai JK. Soft computing based formulations for prediction of compressive strength of sustainable concrete: a comprehensive review. Innov Infrastruct Solut 2022;7:156. https://doi.org/10.1007/s41062-022-00754-7.

[32] Naderpour H, Rafiean AH, Fakharian P. Compressive strength prediction of environmentally friendly concrete using artificial neural networks. J Build Eng 2018;16:213–9. https://doi.org/10.1016/j.jobe.2018.01.007.

[33] Golafshani EM, Behnood A, Arashpour M. Predicting the compressive strength of normal and High-Performance Concretes using ANN and ANFIS hybridized with Grey Wolf Optimizer. Constr Build Mater 2020;232:117266. https://doi.org/10.1016/j.conbuildmat.2019.117266.

[34] Shahmansouri AA, Yazdani M, Ghanbari S, Akbarzadeh Bengar H, Jafari A, Farrokh Ghatte H. Artificial neural network model to predict the compressive strength of eco-friendly geopolymer concrete incorporating silica fume and natural zeolite. J Clean Prod 2021;279:123697. https://doi.org/10.1016/j.jclepro.2020.123697.

[35] Moradi MJ, Khaleghi M, Salimi J, Farhangi V, Ramezanianpour AM. Predicting the compressive strength of concrete containing metakaolin with different properties using ANN. Measurement 2021;183:109790. https://doi.org/10.1016/j.measurement.2021.109790.

[36] Naderpour H, Rezazadeh Eidgahee D, Fakharian P, Rafiean AH, Kalantari SM. A new proposed approach for moment capacity estimation of ferrocement members using Group Method of Data Handling. Eng Sci Technol an Int J 2020;23:382–91. https://doi.org/10.1016/j.jestch.2019.05.013.

[37] Golafshani EM, Behnood A. Predicting the mechanical properties of sustainable concrete containing waste foundry sand using multi-objective ANN approach. Constr Build Mater 2021;291:123314. https://doi.org/10.1016/j.conbuildmat.2021.123314.

[38] Karami H, Ghazvinian H, Dehghanipour M, Ferdosian M. Investigating the Performance of Neural Network Based Group Method of Data Handling to Pan's Daily Evaporation Estimation (Case Study: Garmsar City). J Soft Comput Civ Eng 2021;5:1–18. https://doi.org/10.22115/scce.2021.274484.1282.

[39] Faraj RH, Mohammed AA, Omer KM. Modeling the compressive strength of eco-friendly self-compacting concrete incorporating ground granulated blast furnace slag using soft computing techniques. Environ Sci Pollut Res 2022;29:71338–57. https://doi.org/10.1007/s11356-022-20889-5.

[40] Rezazadeh Eidgahee D, Jahangir H, Solatifar N, Fakharian P, Rezaeemanesh M. Data-driven estimation models of asphalt mixtures dynamic modulus using ANN, GP and combinatorial GMDH approaches. Neural Comput Appl 2022. https://doi.org/10.1007/s00521-022-07382-3.

[41] Ahmed HU, Mohammed AS, Mohammed AA. Proposing several model techniques including ANN and M5P-tree to predict the compressive strength of geopolymer concretes incorporated with nano-silica. Environ Sci Pollut Res 2022;29:71232–56. https://doi.org/10.1007/s11356-022-20863-1.

[42] Pan L, Wang Y, Li K, Guo X. Predicting compressive strength of green concrete using hybrid artificial neural network with genetic algorithm. Struct Concr 2022. https://doi.org/10.1002/suco.202200034.

[43] Zhang J, Li D, Wang Y. Toward intelligent construction: Prediction of mechanical properties of manufactured-sand concrete using tree-based models. J Clean Prod 2020;258:120665. https://doi.org/10.1016/j.jclepro.2020.120665.

[44] Asghari V, Leung YF, Hsu S-C. Deep neural network based framework for complex correlations in engineering metrics. Adv Eng Informatics 2020;44:101058. https://doi.org/10.1016/j.aei.2020.101058.

[45] Asadi Shamsabadi E, Roshan N, Hadigheh SA, Nehdi ML, Khodabakhshian A, Ghalehnovi M. Machine learning-based compressive strength modelling of concrete incorporating waste marble powder. Constr Build Mater 2022;324:126592. https://doi.org/10.1016/j.conbuildmat.2022.126592.

[46] Haque MA, Chen B, Javed MF, Jalal FE. Evaluating the mechanical strength prediction performances of fly ash-based MPC mortar with artificial intelligence approaches. J Clean Prod 2022;355:131815. https://doi.org/10.1016/j.jclepro.2022.131815.

[47] Kim P. MATLAB Deep Learning. Berkeley, CA: Apress; 2017. https://doi.org/10.1007/978-1-4842-2845-6.

[48] Tavana Amlashi A, Mohammadi Golafshani E, Ebrahimi SA, Behnood A. Estimation of the compressive strength of green concretes containing rice husk ash: a comparison of different machine learning approaches. Eur J Environ Civ Eng 2022:1–23. https://doi.org/10.1080/19648189.2022.2068657.

[49] Calin O. Deep Learning Architectures. Cham: Springer International Publishing; 2020. https://doi.org/10.1007/978-3-030-36721-3.

[50] Mendenhall W, Sincich T, Boudreau NS. A second course in statistics: regression analysis. vol. 6. Prentice Hall New York; 2003.

[51] Aggarwal CC. Neural Networks and Deep Learning. Cham: Springer International Publishing; 2018. https://doi.org/10.1007/978-3-319-94463-0.

[52] Goodfellow I, Bengio Y, Courville A. Deep learning (adaptive computation and machine learning series). Cambridge Massachusetts 2017:321–59.

[53] Haykin S. Neural networks and learning machines. Pearson Education India; 2009.

[54] Hinton G, Srivastava N, Swersky K, Tieleman T, Mohamed A. Coursera: Neural networks for machine learning. Lect 9c Using Noise as a Regul 2012.

[55] Kingma Diederik P, Adam JB. A method for stochastic optimization. ArXiv Prepr ArXiv14126980 2014.

[56] Kochenderfer MJ, Wheeler TA. Algorithms for optimization. Mit Press; 2019.

[57] Dozat T. Incorporating nesterov momentum into adam 2016.

[58] Ding X, Li C, Xu Y, Li F, Zhao S. Dataset of long-term compressive strength of concrete with manufactured sand. Data Br 2016;6:959–64. https://doi.org/10.1016/j.dib.2016.01.065.

[59] Ding X, Li C, Xu Y, Li F, Zhao S. Experimental study on long-term compressive strength of concrete with manufactured sand. Constr Build Mater 2016;108:67–73. https://doi.org/10.1016/j.conbuildmat.2016.01.028.

[60] Ghazvinian H, Karami H, Farzin S, Mousavi S-F. Introducing affordable and accessible physical covers to reduce evaporation from agricultural water reservoirs and pools (field study, statistics, and intelligent methods). Arab J Geosci 2021;14:2543. https://doi.org/10.1007/s12517-021-08735-3.

[61] Wong T-T, Yeh P-Y. Reliable Accuracy Estimates from k -Fold Cross Validation. IEEE Trans Knowl Data Eng 2020;32:1586–94. https://doi.org/10.1109/TKDE.2019.2912815.