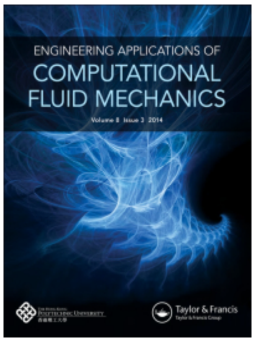


Taylor & Francis
Taylor & Francis Group

Engineering Applications of Computational Fluid Mechanics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcfm20

Modeling the infiltration rate of wastewater infiltration basins considering water quality parameters using different artificial neural network techniques

Ghada Abdalrahman, Sai Hin Lai, Pavitra Kumar, Ali Najah Ahmed, Mohsen Sherif, Ahmed Sefelnasr, Kwok Wing Chau & Ahmed Elshafie

To cite this article: Ghada Abdalrahman, Sai Hin Lai, Pavitra Kumar, Ali Najah Ahmed, Mohsen Sherif, Ahmed Sefelnasr, Kwok Wing Chau & Ahmed Elshafie (2022) Modeling the infiltration rate of wastewater infiltration basins considering water quality parameters using different artificial neural network techniques, Engineering Applications of Computational Fluid Mechanics, 16:1, 397-421, DOI: 10.1080/19942060.2021.2019126

To link to this article: https://doi.org/10.1080/19942060.2021.2019126

ENGINEERING APPLICATIONS OF COMPUTATIONAL FLUID MECHANICS
2021, VOL. 16, NO. 1, 397–421
https://doi.org/10.1080/19942060.2021.2019126

Taylor & Francis
Taylor & Francis Group

Modeling the infiltration rate of wastewater infiltration basins considering water quality parameters using different artificial neural network techniques

Ghada Abdalrahman[a], Sai Hin Lai[a], Pavitra Kumar[a], Ali Najah Ahmed [b], Mohsen Sherif [c,d], Ahmed Sefelnasr [c], Kwok Wing Chau [e] and Ahmed Elshafie [a,c]

[a]Department of Civil Engineering, Faculty of Engineering, University of Malaya (UM), Kuala Lumpur, Malaysia; [b]Institute of Energy Infrastructure (IEI), Department of Civil Engineering, College of Engineering, Universiti Tenaga Nasional (UNITEN), Selangor, Malaysia; [c]National Water and Energy Center, United Arab Emirates University, Al Ain, United Arab Emirates; [d]Civil and Environmental Engineering Department, College of Engineering, United Arab Emirates University, Al Ain, United Arab Emirates; [e]Department of Civil and Environmental Engineering, Hong Kong Polytechnic University, Kowloon, Hong Kong

ABSTRACT

Predicting the infiltration rate (IR) of treated wastewater (TWW) is essential in controlling clogging problems. Most researchers that predict the IR using neural network models considered the characteristics parameters of soil without considering those of TWW. Therefore, this study aims to develop a model for predicting the IR based on various combinations of TWW characteristics parameters (i.e. total suspended solids (TSS), biological oxygen demand (BOD), electric conductivity (EC), pH, total nitrogen (TN), total phosphorous (TP), and hydraulic loading rate (HLR)) as input parameters. Therefore, two different artificial neural network (ANN) architectures, multilayer perceptron model (MLP) and Elman neural network (ENN), were used to develop optimal model. The optimal model was selected through evaluating three stages: selecting the best division of data, selecting the best model, and deciding the best combination of input parameters based on several performance criteria. The study concluded that the first combination of inputs that include all the seven-parameter using MLP model associated with 90% division of data was the optimal model in predicting the IR depending on TWW characteristics parameters, achieving a promising result of 0.97 for the coefficient of determination, 0.97 for test regression, 0.012 for MSE with 32.4 of max relative percentage error.

Abbreviations: IR: Infiltration Rate; TWW: Treated Wastewater; TSS: Total Suspended Solids; BOD: Biological Oxygen Demand; EC: Electric Conductivity; HC: Hydraulic Conductivity; TN: Total Nitrogen; TP: Total Phosphorous; HLR: Hydraulic Loading Rate; ANN: Artificial Neural Network; MLP: Multilayer Perceptron Model; ENN: Elman Neural Network; FFANN: Feedforward Artificial Neural Networks; *R*: Regression Values; SAR: Sodium Adsorption Ratio; DOC: Dissolved Organic Carbon; ANAMMOX: Anaerobic Ammonium Oxidation; CEC: Cation Exchange Capacity; BPNN: Back Propagation Neural Network; GRNN: General Regression Neural Networks; ELM: Extreme Learning Machine Neural Networks; TDNN: Time Delay Neural Network; TLRN: Time Lag Recurrent Network; NGWTP: North Gaza Wastewater Treatment Plant; MASL: Meters Above Sea Level; DNC: Dynamic Node Creation; PWA: Palestinian Water Authority; RBF: Radial Basis Function; ANFIS: Adaptive Neuro Fuzzy Inference System; BD: Bulk Density; RMSE: Root Mean Square Error; MAE: Mean Absolute Error; MSE: Mean Square Error; R^2: Determination Coefficient; LLR: Local Linear Regression; DLLR: Dynamic Linear Regression; MNN: Modular Neural Networks; RNN: Recurrent Neural Network; NARX: Nonlinear Autoregressive with Exogenous input network; WNN: Wavelet Neural Networks

ARTICLE HISTORY
Received 3 March 2021
Accepted 10 December 2021

KEYWORDS
Infiltration rate; treated wastewater; artificial neural network; multilayer perceptron; Elman neural network

Introduction

The infiltration process of treated wastewater through the infiltration basins plays a fundamental role in the hydrologic cycle. It is known as one of the most effective and low-cost methods of achieving tertiary treating of the partial TWW and replenishing the deteriorated groundwater level. However, this process of recharging the treated wastewater results in a reduction in infiltration rate (IR) and surface clogging of soils of the basins (Abdalrahman et al., 2021). Surface spreading basins are the most widely used technique in the filtration of treated wastewater, as it is a simple and effective technique for

CONTACT Ahmed Elshafie elshafie@um.edu.my

primary and secondary TWW. A review by Abdalrahman et al. (2021) concluded that the influencing factors on the TWW infiltration rate and clogging of basins could be classified into soil-characteristics related factors, TWW characteristics factors, and hydraulic loading rate and operating procedure (drying and wetting periods). Soil characteristics play a leading role in the effects of TWW infiltrations which is represented by hydraulic conductivity; the bulk density, porosity, size of particles, and texture and structure of soil are represented. In contrast, the most controlling parameters of the TWW includes biological oxygen demand (BOD), total suspended solids (TSS), pH, electric conductivity (EC), total nitrogen (TN), total phosphorus (TP), and hydraulic loading rate (HLR) for the TWW.

Since the partial treatment of wastewater usually results in high concentrations of BOD, TSS, TN, TP, pH and high organic matter levels, and hence leading to biomass and organic matter accumulation on the surface of the infiltration basins, which decreases the infiltration rate and cause clogging to basins (Environmental Protection Agency, 2004). The negative effect of BOD on IR has been confirmed by many researchers (Aaltomaa & Joy, 2002; Jnad et al., 2001; Magesan, 2001; Vandevivere & Baveye, 1992). The TSS is another controlling factor in reducing IR due to suspended sediments' deposition(Horneck et al., 2007). Besides, Aboukarima et al. (2018) and Emdad et al. (2004) showed that as the EC value for TWW is lower, the IR is lower at the same value of SAR and vice versa on sandy loam land and clay loam sand.

Furthermore, Suarez and Gonzalez-Rubio (2017) stated that at the same SAR value, as pH increases with and without dissolved organic carbon (DOC), the IR decreases. Another polluted component in TWW is TN and TP, which can cause a reduction in the IR (Phong et al., 2013). The final parameter is the hydraulic loading rate (HLR); the review of Abdalrahman et al. (2021) shows that the infiltration rate is increased as the HLR decreases.

Almost all researchers who predicted the IR using artificial neural network models (ANN) or Multiple regression model depended on soil characteristics parameters only or combined with a few factors related to TWW characteristics. For example, Kashi et al. (2014) predicted the soil infiltration rate for two sites depended on some soil characteristics such as soil texture, lime percentage, sodium adsorption ratio (SAR), bulk density (BD), and electric conductivity (EC) on the estimation of infiltration rate and CEC. The efficiency of multiple regression and three architectures of ANN (MLP, RBF, and ANFIS) were compared to get the optimal model. The results showed that MLP model was the most effective model among the other models based on statistical parameters.

Previously, Sy (2006) used measurable data to model the infiltration process through a set of traditional models and MLP neural networks. The experimental input parameters were related to the slope of terrains and intensity of rainfall, and others related to soil characteristics parameters such as percentage of sand, percentage of clay, HC, BD, and moisture of the soil. The MLP model was also the best performance model compared to Philip and Green-Ampt, Kostiakov, and Horton models based on several statistical parameters.

Several researchers in the literature used predictive model in estimation of IR or Hydraulic conductivity depended on soil characteristics, bulk density, water content, etc. such as (Al-Janobi et al., 2010; Anari et al., 2011; Ekhmaj, 2010; Sihag, 2018; Sihag et al., 2017a, 2017b; Sy, 2006). Ekhmaj (2010), Al-Janobi et al. (2010), and Sarmadian and Taghizadeh-Mehrjardi (2014) used ANN with a backpropagation algorithm and strongly recommended the ANN in IR prediction compared over the multilinear regression models in predicting the infiltration rate and deep percolation.

While Anari et al. (2011) compared two neural network models which are ANN and ANFIS, to estimate the overall infiltration rate using the first and second 5 min time periods of infiltration rate, with the Local Linear Regression (LLR) and the Dynamic Linear Regression (DLLR), among other models, ANFIS shows the highest performance.

Since, Sihag et al. (2017a) started the estimation of IR using a novel nonlinear regression model that proved high performance over other traditional models. This study was followed by another laboratory research conducted on a different percentage of soil mixture of synthetic and compared the performance of three different models (support vector machine, Gaussian regression model) with the conventional models in predicting the IR. The study concluded that the Gaussian regression model had the optimal accuracy over the other models (Sihag et al., 2017b). Later, Sihag et al. (2019) used the ANFIZ model, which shows superior performance on similar laboratory data in predicting the IR.

Recently, Panahi et al. (2021) predicted the cumulative infiltration and infiltration rate for sixteen different sites in Iran. The prediction is performed through a convolutional neural network (CNN) that use gray wolf optimization (GWO), a genetic algorithm (GA), and an independent component analysis (ICA). The input data consist of 154 records that including the time of measuring; other soil characteristics parameters such as (sand, clay, and silt percent; bulk density; soil moisture percent). The results showed that CNN with GWO algorithm achieved the best performance of prediction both (cumulative infiltration and infiltration rate), followed by CNN with ICA, CNN with Genetic algorithm, and CNN standalone. In addition, this study stated that the most optimum combination of input parameters is the combination of all parameters.

According to the literature analysis, most researchers who use different multiple regression and neural network models to predict infiltration rate and especially for the infiltration of TWW through vadose zone depended on soil characteristics parameters only as input parameters, there are no researches focus on predicting the IR using the TWW characteristics parameters. Besides, most of the researchers confirm the limitation of the conventional multiple regression models to predict the IR either physically, conceptually, or empirically as each particular model can be more suitable for specific conditions (location, soil type, availability of data, etc.) in predicting the IR than the other regression models (Abdalrahman et al., 2021; Haghighi Fashi et al., 2010; Turner, 2006).

Moreover, the regression model is prepared based on fitting a linear equation to the observed data, so that it is inconvenient for transforming complex relations of different input variables to predict the target variable of IR to mathematical equations (Stangierski et al., 2019). Unlike, the ANN – the focus of this study – characterize with the high ability to learn complex nonlinear relations without the need to transform the relationship to a mathematical equation, the rapid and convenience of developing, the robustness to missing data, high ability of a large amount of data (Ebtehaj & Bonakdari, 2014; El-Shafie et al., 2011).

Consequently, most recent studies have been motivated to utilize the ANN modeling approach to model IR (Abdalrahman et al., 2021; Azimi et al., 2019; Ebtehaj & Bonakdari, 2014, 2016; El-Shafie et al., 2011). Nevertheless, ANN has some drawbacks regarding the overfitting of ANN that can be managed by stopping the training process whenever the testing performance begins to decrease as well as integrating the ANN model with advanced optimization models to improve the accuracy performance of predicting IR (Ebtehaj & Bonakdari, 2016; El-Shafie et al., 2011; Kumar, Lai, Mohd, et al., 2020).

Consequently, the primary objective of this study is to investigate and predict the IR based on TWW characteristics parameters, such as TSS, BOD, EC, pH, TN, TP, and HLR. While most of the research focused only on the effect of soil characteristics on the IR, current study investigates the effect of TWW characteristics parameters and the effect of different combinations of these parameters. Two different neural network architectures are used to obtain the most optimal IR predicting model: multilayer perceptron (MLP) model, and Elman neural network (ENN) model. Training and testing of these models are performed on the MATLAB platform.

Materials and methods

Study area

This study concerns the infiltration basins for the effluent of the north Gaza wastewater treatment plant (NGWTP), located in the north-east of Gaza city, Palestine, and covering a total area of around 80,000 m^2. The site is located on a slope ranges from 50–70 meters above sea level (masl) from the west part to the east part.

The total wastewater flow to these basins is 35,713 M^3d^{-1} from NGWTP, which has been partially treated. In this partial treatment of wastewater, most contaminants were removed through primary and secondary treatment without passing into tertiary treated of wastewater which is mostly comprised of removing phosphates and nitrates from the water supply. Besides, some specific parameters may not be fully treated according to the allowed WHO or FAO standard for recharging the TWW and most Mediterranean region guidelines as stated in World Health Organization Regional Office for the Eastern Mediterranean (2006).

Figure 1 indicates the infiltration basins layout and NGWTP geographic location. There are 1 to 9 infiltration basins, and the flow of effluent is directed to basins by three groups. Group no.1: including the infiltration basins no. 1,2,3 at level 47 (masl), group no.2: including basins no. 4,5,7 at level 49 (masl), the group no. 3: including basins no. 6,8, 9 at level 52 (masl). The water is distributed to one to three basins in each group at the same time. The water is redirected to the next group after flooding for a certain time, from 0.5 to 2 days from the previous group. The basins are allowed to dry between flood cycles for a period of 1–4 days or longer (SWECO, 2003).

The arising problem of infiltration basins is the reduction of infiltration rate and clogging due to the partial treatment of wastewater, that may lead to accumulation of suspended solids and BOD loading as examples but not limited to that reasons only whereas the level of TWW can play a main role in clogging process so that the secondary effluent have high levels of SS comparing to tertiary effluents, so secondary effluent can result in more clogging (Abdalrahman et al., 2021; Gharaibeh et al., 2016; Riad et al., 2013).

Due to clogging problems, in 2017–2018, two meters of clogged layer soil under basins were replaced with

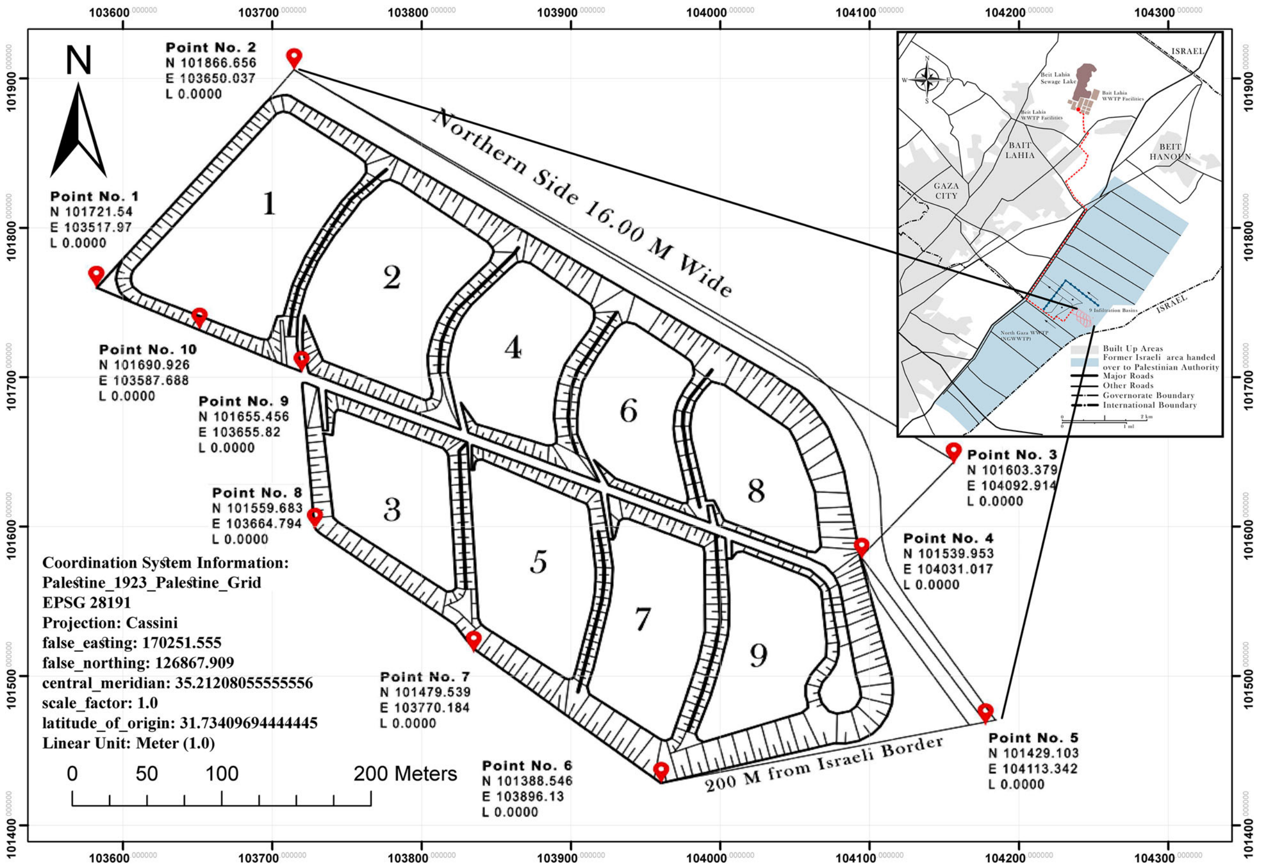


Figure 1. Layout and geographic location of infiltration basins.

new sandy soil instead by the Palestinian water authority (PWA). Within the course of the factors affecting the clogging and infiltration rate of basins, data from the Palestinian water authority is collected about the effluent wastewater quality and the corresponding infiltration rate during the period from January–July of the year 2020.

Artificial neural networks

Artificial neural network (ANN) models, a computational system, can connect inputs to the target output variables to form complete data-driven knowledge.

The ANN model is highly dependent on the amount of data since some valuable information may be lost in short-term data, which in turn contributes to poor prediction outcomes (Ebtehaj & Bonakdari, 2016; Liu et al., 2012). Furthermore, the correct division of data is essential in creating an efficient model and selecting a suitable training algorithm to calibrate the model parameters. The Levenberg-Marquardt algorithm and back-propagation algorithm are the most common algorithms used (Chen et al., 2020; Huo et al., 2013).

ANN models are considered tools for predicting problems in many nonlinear hydrologic processes such as rainfall-runoff, water quality and quantity simulation, groundwater management, and infiltration rate in the vadose zone, the subject of this study. Feedforward networks, recurrent networks, and hybrid models are the most frequently used architectures of ANN models in the hydrological processes in general and infiltration rate in particular.

Feedforward networks which are also known as multilayered network of neurons are the simplest structure of ANN and the information travel through forwarding paths only since the neuron connection exist from input's neuron to the hidden layer's neuron (single or multiple), or from the hidden layer's neuron to the output's neuron without feedback connections to the previous layers.

A multilayer perceptron (MLP) model is used in this study as an example of this type of network that most frequently used in hydrologic processes with two or three hidden(Chen et al., 2020; Sharaf El Din et al., 2017). The layers of the MLP model are connected to each other, while the neurons in the same layer are not connected. The neurons in layers have two different functions (combining functions and activation functions) since they combine the weighted sum and a differentiable non-linear activation function. These neurons sum the input signals and propagate the input through a non-linear function.

Generally, MLP characterize with adaptability for any applications of training process using the most common method of backpropagation algorithm; however, it has some limitations such as slow and non- stability of convergence, stucking in a local minimum. Therefore, a developed algorithm known as Levenberg-Marquard algorithm can overcome these limitations (Toha & Tokhi, 2008).

Recurrent neural network (RNN) uses sequential data or time-series data as training data to learn the model, but unlike the feedforward, it has a memory from previous inputs to influence the current input and the output, as well as the neurons in the same layer, are interconnected and allow feedback. RNN depends on the hidden state feature, which remembers the calculated information from previous elements of the sequence, and the same inputs can produce different outputs in a series. It has the advantage of reducing the complexity of other neural networks as it converts the independent activations into dependent activations by using the same parameters of weight and biases to all layers to produce output. Besides, RNN can take more than one input vector and bring out more than one output. On the other hand, RNN has some limitations in processing some activation functions in long sequences as well as it is a little bit more challenging to run than other neural networks.

Elman neural network (ENN), the other type of architecture used in this study, is considered as one of the most common recurrent dynamic neural network architecture models. It is characterized by an additional layer to store the internal state more than conventional neural networks called the context layer (Özcan et al., 2009). It is characterized by the nodes' self-connections that make the network sensitive to previous layer data by dynamic system modeling so that the feedback of errors is transferred back through the loops, which help in updating the weights of the corresponding inputs and produce short continuations of known sequences (Elman, 1990; Kumar, Lai, Mohd, et al., 2020). ENN can adjust to time-varying characteristics and has a strong computing power (Liu et al., 2012). On the other hand, Zhang et al. (2007) stated some limitations of Elman, such as sub-optimal solutions of problems, the inefficiency of getting proper weights for hidden layers due to the approximation of the error gradient, inefficiency memory capacity. Consequently, several researchers suggested some modifications to increase the capacity of memory to overcome the local minima problems.

To conclude, each MLP and ENN has strengths and weaknesses points so that there is no confirmation of the superiority of the model over the other. In this regard, both MLP and ENN performance were compared based on a specific statistical criterion.

Methodology

Data pre-processing

The collected data includes the frequent test analysis of the effluent TWW that directed to the operated basins, which consist of (BOD, TSS, HC, pH, TN, TP) as well as the hydraulic loading rate represented by total daily flow and the corresponding IR of those operated basins during the period of Jan–July (2020) for the nine basins of NGWTP.

A data set containing 150 analysis values of TWW effluent quality in parallel with the hydraulic loading rates of infiltration basins groups was used as input variables to evaluate ANN models' effectiveness to estimate the infiltration rate as the target variable. The target variable (IR) was measured on the same day of measuring the input variables as basin groups were operated from 2–4 days then left for drying 1–4 days while the TWW effluent was redirected to other groups. Regarding the used methods for each of the TWW characteristics parameters, the BOD (mgl^{-1}) parameter was measured using the OxiTop – Respirometers for the self-monitoring measurement of BOD in undiluted samples according to DIN EN 1899-2, the EC parameter (μScm^{-1}) was measured using conductivity meter AL20 Con, pH parameter was measured using pH meter AQUALYTIC AL20, hydraulic loading rate represented by total flow per day (m^3d^{-1}) was measured using Endress + Hauser-Electromagnetic Flow Measuring System. While for measuring the TSS (mgl^{-1}) parameter, 2540 SOLIDS, standard methods for the examination of water and wastewater was used ('2540 SOLIDS,' 2018), for the TN (mgl^{-1}) parameter, 4500-No_2, 4500-NO_3, 4500-N_{org} B/standard methods for the examination of water and wastewater ('4500-N NITROGEN,' 2018). Also, for measuring the TP (mgl^{-1}) parameter, 4500-P B.4. Sulfuric Acid-Nitric Acid Digestion /Standard Methods for the Examination of Water and Wastewater ('4500-P PHOSPHORUS,' 2018). Regarding the output parameter; IR (md^{-1}) was measured by calculating decreasing level in the SCADA trend during the time of filtration for each day (Palestinian Water Authority, 2020).

For the soil characteristics, the first two layers of the basin's soil consist of a mixture of Sand-SP, Sand SP-SM, Gravelly sand-SP, Gravelly sand-SP-SM. The average percentage of fines content (Silt and Clay) is 4.43 while the average hydraulic conductivity is equal (24.633 md^{-1})

and the average organic matter (ppm) is 863.33 ppm over the basins.

The focus of this study will be on the TWW characteristics parameters effect on predicting the IR such as hydraulic loading rate represented by total flow per day (m^3d^{-1}), and the TWW effluent quality comprising of BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), TN (mgl^{-1}), TP (mgl^{-1}) on the IR.

As the input variables' collected data have a few gaps around 3–4 records, these missing data were estimated using an interpolation process. Linear interpolation using the forecast function in excel was used as there are few missing data in input variables. Figure 2 shows the input data set after interpolation for total flow per day, and TWW effluent quality parameters (BOD, TSS, pH, EC, TN, TP) from January–July 2020, and Figure 3 shows the target infiltration rate in meter per day for the same period.

Regarding the statistical data in Table 1, mean, maximum, standard deviation, kurtosis, and skewness were calculated for all the input variables and the output variable. The average value of the target infiltration rate was 1.65 m/d with a maximum value of 3 m/d and a minimum value of 0.3 m/d. At the same time, the standard deviation, kurtosis, and skewness of infiltration rate were 0.61, −0.38, and 0.42, respectively.

The statistical data in Table 1 has been estimated for all average, maximum, standard deviation, kurtosis, and skewness for all input and output parameters. The average target infiltration rate was 1.65 m/d with the highest 3 m/d value, and the minimum infiltration rate was 0.3 m/d. In contrast, the standard deviation, kurtosis, and skewness of infiltration rate were 0.61, −0.38, and 0.42, respectively.

Data division

Data division can be carried out using random data division, contiguous blocks division, interleaved selection division, and index data division in geotechnical applications. The input data is divided into three data sets: training, validation, and testing set (Kumar, Lai, Mohd, et al., 2020). The training set is a sample of data used to train the model with the appropriate weights and biases of connections between neurons in ANN. In contrast, the validation set is a development set that is used as an unbiased evaluation of the model to adopt the training dataset and to adjust the model hyperparameters. Finally, the testing dataset is used to assess the model's performance characteristics such as accuracy, sensitivity, specificity, and F-measure. To ensure the minimum overfitting, the model must fit the test dataset as well as fit to the training model.

In this study, the division by index method is used for MLP and ENN model in each of four combination groups which divide the input data into three different divisions in training the models which are (70% for training, 15% for validation, 15% for testing), (80% for training, 10% for validation, 10% for testing), and (90% for training, 5% for validation, 5% for testing). While in the four combination groups of parameters used in ENN, the divisions were (70% for training, 30% for testing), (80% for training, 20% for testing), and (90% for training, 10% for testing).

In this method, the data is divided into 3 sets of training, validation, and testing in the MLP network using index function so that each set has approximately the same statistical characteristics. These indices are selected so that the three sets have close mean values to each other in addition to that the training data should have the maximum and minimum values of the target to enable the network to train different patterns of data (Lagos-Avid & Bonilla, 2017; Lu et al., 2019).

Furthermore, for the Elman neural network, the same index data division method is used for dividing the data set into two groups of training and testing as the Elman does not need separate data for validation.

Model training and parameter selection

The training of the neural network was made using Matlab platform to estimate the infiltration rate based on the input variables of TWW quality parameters and hydraulic loading rate as most of the previous studies depend on the soil characteristics and other factors rather than the TWW characteristics, which is the focus of this study. Two models were trained for four different combination groups of input parameters which are MLP, ENN. In addition, for each combination group of parameters in MLP, five different divisions of data were used in training the models (50%, 60%, 70%, 80%, 90%) divisions as explained in Table 2. In the first combination group, all of the seven inputs parameters of TWW characteristics were trained together which include the total flow (m^3d^{-1}), BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), TN (mgl^{-1}), TP (mgl^{-1}). The second combination group consists of six input parameters which are total flow (m^3d^{-1}), BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), TN (mgl^{-1}) while excluding the last parameter TP (mgl^{-1}) from this group. The third combination group consists of five input parameters which are total flow (m^3d^{-1}), BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), while excluding TN, and TP. The fourth combination group consists of four parameters which are the total flow (m^3d^{-1}), BOD (mgl^{-1}), TSS (mgl^{-1}), HC (μScm^{-1}) excluding pH, TN, TP. These combination groups were chosen based on the most effective parameters by reviewing previous studies mentioned in

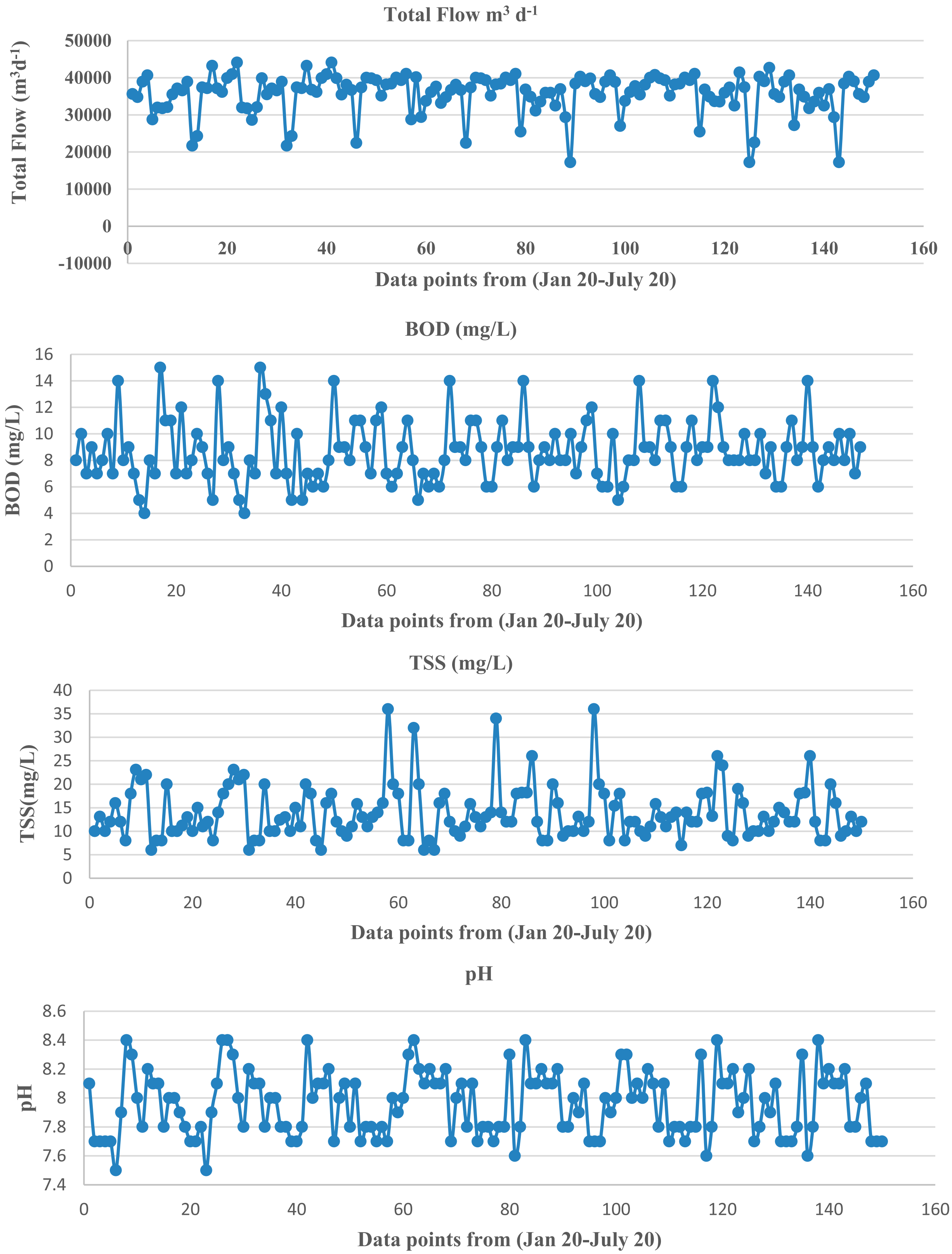


Figure 2. The input parameters.

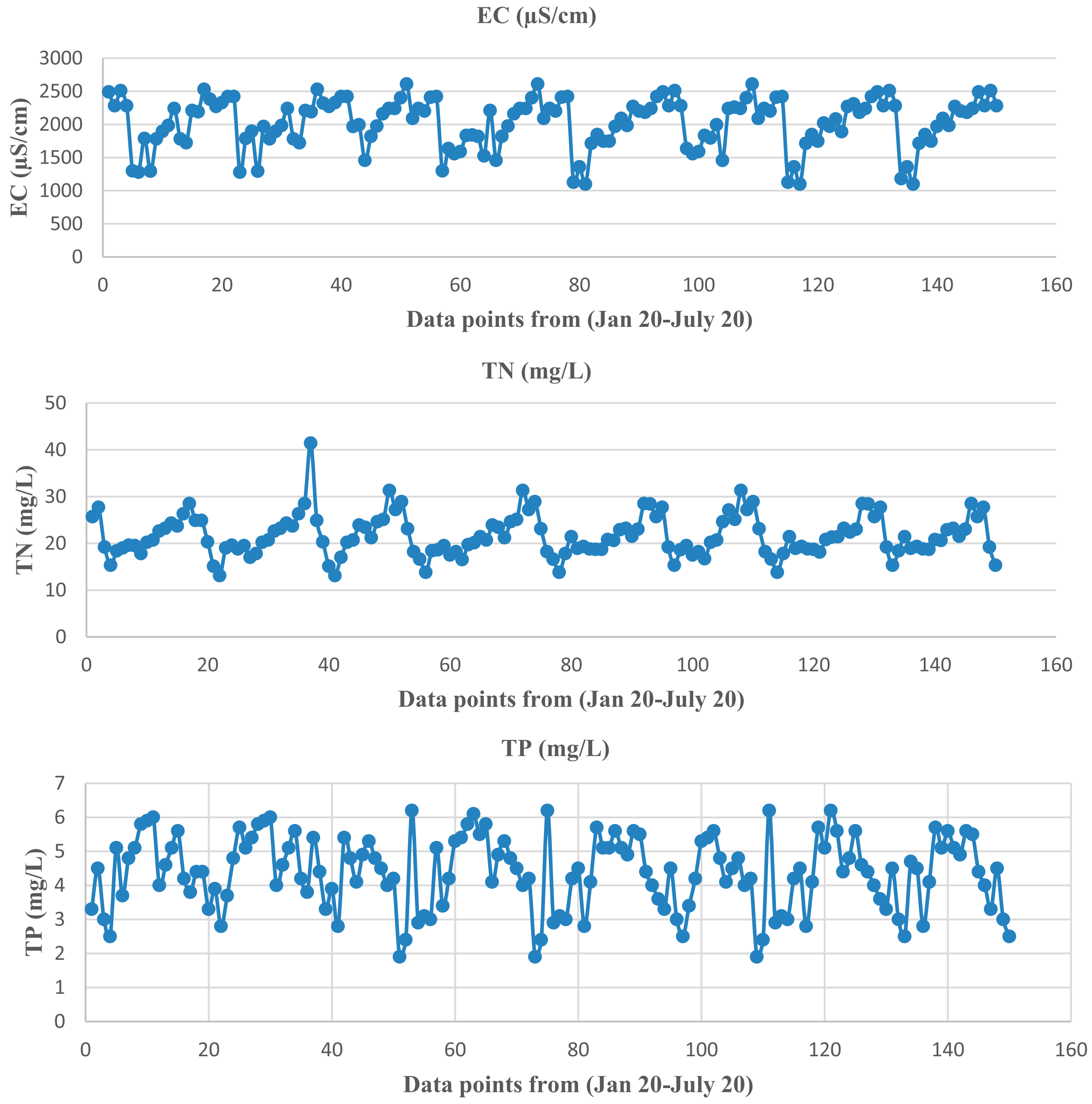


Figure 2. Continued.

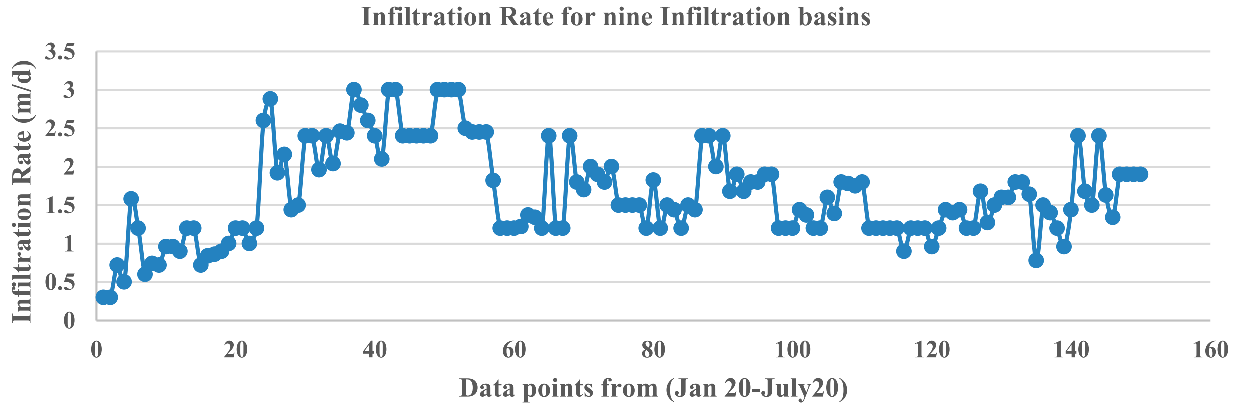


Figure 3. The target parameter (infiltration rate).

Table 1. Statistical analysis for both input and target variables.

	Input variables							Target
	Total Flow / Day	EC µScm^{-1}	pH	TSS mgl^{-1}	BOD mgl^{-1}	TN mgl^{-1}	TP mgl^{-1}	IR md^{-1}
Minimum	17,240	1098	7.5	6	4	13.10	1.90	0.30
Max	44,090	2610	8.40	36	15	41.40	6.20	3.00
Mean	35,713.10	2021.55	7.96	13.89	8.67	21.60	4.37	1.65
Standard deviation	5345.57	379.95	0.22	5.72	2.34	4.38	1.07	0.61
kurtosis	2.38	−0.25	−0.90	2.95	0.29	1.81	−0.74	−0.38
skewness	−1.52	−0.73	0.23	1.46	0.66	0.83	−0.33	0.42

Table 2. Different combination groups of inputs and internal network properties for MLP and ENN models.

Neural network architectures	Combination of inputs	Network properties	Data division
Feed Forward backpropagation neural network: Multilayer Perceptron Neural network (MLP)	**Combination inputs 1** (Total flow, BOD, TSS, pH, HC, TN, TP(	**Hidden Layers:** 1, 2, 3 **Nodes:** 2–10 **Epochs:** 100–1000	**First Division:** 50% for Training 25% for validation 25% for Testing
	Combination inputs 2 (Total flow, BOD, TSS, pH, HC, TN)		**Second Division:** 60% for Training 20% for validation 20% for Testing
	Combination inputs 3 (Total flow, BOD, TSS, pH, EC)		**Third Division:** 70% for Training 15% for validation 15% for Testing
	Combination inputs 4 (Total flow, BOD, TSS, EC)		**Fourth Division:** 80% for Training 10% for validation 10% for Testing
		Fifth Division: 90% for Training 5% for validation 5% for Testing	
Recurrent Neural Network: Elman Neural Network	**Combination inputs 1** (Total flow, BOD, TSS, pH, HC, TN, TP(	**Hidden Layers:** 1, 2, 3 **Nodes:** 2–10 **Epochs:** 100–1000 **Goal:** (0.01, 0.02, 0.03, 0.04, 0.05)	**First Division:** 50% for Training 25% for validation 25% for Testing
	Combination inputs 2 (Total flow, BOD, TSS, pH, HC, TN)		**Second Division:** 60% for Training 20% for validation 20% for Testing
	Combination inputs 3 (Total flow, BOD, TSS, pH, EC)		**Third Division:** 70% for Training 15% for validation 15% for Testing
	Combination inputs 4 (Total flow, BOD, TSS, EC)		**Fourth Division:** 80% for Training 10% for validation 10% for Testing
		Fifth Division: 90% for Training 5% for validation 5% for Testing	

(section1). Several trials of models were developed for each group's training with 1, 2, 3 hidden layers, and each hidden layer has nodes from 2 to 10. These models were trained with a wide range of numbers of epochs (100 to 1000) using codes that allow training a huge number of trials with different combinations of input parameters, as shown in Table 2.

The method of determining the possible number of hidden layers and neurons for the network was by trial and error and roughly estimated using some rules of thumb then evaluate the output of the model with the actual target to find the optimum number of hidden layers and neurons as there is not a specified guideline to select the correct number directly. The rule of thumb method to estimate the starting point to the range of neurons is 1. The number of neurons should be between the number of inputs and outputs layers and less than the double number of inputs, 2. The number of hidden layers should be two-thirds the number of inputs plus outputs. Through the training and testing of these different combination groups, it can be determined the most optimum model based on the regression values as well as it can be defined as the most prominent input parameters that affect the target IR based on the most accurate model. There are two other approaches to finding the perfect network to learn the mapping and enough to generalize the network. The first approach is by using a more extensive network than needed topology for training until the mapping is found. In this case, some elements of the network are eliminated if it is not active. The second approach starts with a small network and increases the nodes and weights until an optimum network is found (Ash, 1989). However, these approaches have a shortcoming in consuming a long time to find the optimal network. A developed approach is introduced called dynamic node creation (DNC), which automatically develops backpropagation networks by adding neurons to the hidden layer until the required accuracy is achieved. This accuracy of getting the closest output to the desired target can be achieved through the sum of squared differences.

Modeling performance criteria

For ANN multilayer perceptron, the most crucial step is to specify the optimal number of the hidden layers and neurons of the neural network to produce a model with high accuracy depending on performance criteria such as the regression values (R) of analysis for training, validation, and testing as a preliminary indication of how the data fit to the best fit line (Equation (1)). In this study, the accuracy of ANN will be evaluated based on nine other performance criteria, which are: (1) mean square error (MSE) in which the errors are squared before they are averaged and assigns a higher weight to larger errors (Equation (2)). (2) The plot of the observed and the predicted values. (3) Plot of relative error percentage values (Equation (3)). (4) Nash-Sutcliffe Efficiency (NSE) that indicate how the plot of observed values versus predicted values fits the 1:1 line (If NSE = 1, that means a perfect match of predicted values to the measured values, if NSE = 0, that means the predicted values are as accurate as the mean of observed value if 0 < NSE < ∞, that means the mean of observed values is a better predictor than the model) so the model have values close to 1 is considered the best prediction model (Equation (4)) (Bonakdari et al., 2019; Ebtehaj & Bonakdari, 2014, 2016; El-Shafie et al., 2011; Kumar, Lai, Wong et al., 2020; Moeeni et al., 2017; Zhang et al., 2007). Additional two criteria were used also to investigate the ability of models to accurately predict the peak and low values for the target by measuring the error of peak and low values. These two criteria are: (5) the peak flow criteria (PFC) (Equation (5)) and (6) low flow criteria (LFC) (Equation (6)). The values of PFC and LFC that are close to zero means the more accurate model (Coulibaly et al., 2001). The equations of these modeling performance criteria are shown as follows:

$$\text{Regression Value, } R = \frac{n(\sum^{x} y) - (\sum^{x})(\sum^{y})}{\sqrt{\left[n\sum^{x^2} - (\sum^{x})^2\right]\left[n\sum^{y^2} - (\sum^{y})^2\right]}} \tag{1}$$

$$\text{Mean square error, MSE} = \frac{1}{n}\sum_{i=1}^{n}(x-y)^2 \tag{2}$$

$$\text{Relative Error Percentage, REP} = \frac{|x-y|}{x} * 100 \tag{3}$$

$$\text{Nash - Sutcliffe Efficiency (NSE)} = 1 - \frac{\sum^{(Y-X)^2}}{\sum^{(Y-\bar{X})^2}} \tag{4}$$

$$\text{Peak Flow Criteria (PFC)} = \frac{\sum_{1}^{T_p}(x-y)^2 \times X^2)^{0.25}}{(\sum_{1}^{T_p} X^2)^{0.5}} \tag{5}$$

$$\text{Low Flow Criteria (LFC)} = \frac{\sum_{1}^{T_L}(x-y)^2 \times X^2)^{0.25}}{(\sum_{1}^{T_L} X^2)^{0.5}} \tag{6}$$

where n is the number of data points, x is the observed data points, y is the predicted data points. T_P = the number of peak flow greater than one-third of observed mean peak flow, and T_L = the number of low flows lower than one-third of observed mean low flow.

Results

Two different ANN architectures (MLP, ENN) were developed to simulate and predict infiltration rate using TWW characteristics parameters and hydraulic loading rate as inputs. Regarding the soil texture parameters, the first two meters of basin layers are from the same type of sandy soil, so the effect of soil is considered homogeneous under all the basins, and the focus of the study will be on the effect of hydraulic loading rate parameter represented by total flow per day (m^3d^{-1}), and the TWW effluent quality parameters comprising of BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), TN (mgl^{-1}), TP (mgl^{-1}) on the IR.

The models have been trained thousands of times with different internal parameters of models such as five different data divisions and four different combination groups of input parameters that were selected based on an in-depth review of the affecting factors of TWW characteristics on IR. The training has been done using a range of hidden layers (1–3) and a range of neurons (2–10). Analysis of each model was performed based on the performance criteria to select the optimum model for predicting IR. Thousands of lower accuracy models were filtered out based on their test regression value. The remaining models were analyzed using other performance criteria such as: MSE, MAE, max relative error.

The purpose of training many models is to get the best model for each combination group of input parameters in each division set of data. Hence, a comparison between models was conducted based on the higher performance criteria to select the most optimal data division and most optimal combination inputs in that division in the prediction of IR. Table 3 shows the two models' comparison results for four inputs combination groups with five data divisions. The table also shows the best trial's internal parameters for each model, including the number of hidden layers and neurons, training and testing regression, MSE, and max relative percentage error.

As it is clear from Table 3, the 90% data division models for all four combination groups shows the efficiency of getting the highest values for regression, and coefficient of determination and the lowest values for MSE, MAE, and max relative error percentage over the other two divisions (70%, 80%). For the 90% division in the MLP model, the test regression was the highest as it ranges from (0.94–0.97) while the 70% and 80% recorded less values (0.74–0.79), (0.74–0.88), respectively. For MSE and MAE, the values for 90% division were (0.012–0.078), (0.067–0.219), respectively, which are much smaller than the values for 50%, 60%, 70% and 80% division. Regarding the most control factor, the max relative error in 90% division was the lowest value as it ranges from (32.4–81.2) while in the 50%, 60%, 70% and 80% division, the values were relatively high and were in the range of (102.53–133.6), (80.47–121.28), (65.385–117.366), and (73.164–131.97) respectively.

The 90% division in the ENN also confirms the high efficiency of getting more accurate results over the other four divisions. The MSE and MAE recorded values within the range of (0.065–0.074), (0.192–0.207) respectively, which are lower than the other divisions.

Furthermore, the max relative percentage error was (44.55–56.58), which are the lowest values in comparison to the other four divisions. However, the values of test regression values were very close to each other; hence, the other performance criteria, such as MSE, max relative error, PFC, and LFC were opted to select the optimum model. MSE, max relative error, PFC, and LFC criteria state that the 90% data division models have more accurate performance.

Regarding the NSE factor, as it is noticeable that most of the values especially for the three divisions (70%, 80%, 90%) of the two models close to one while for (50%, 60%) were not close to 1, so both models are efficient in predicting the target output (IR). For more precise, in MLP model, the values for 90% division were the highest value and very close to one as it ranges from (0.717–0.965) which means the highest efficiency in predicting the output. In ENN, the values of NSE for the 90% division ranges from (0.578–0.839), which are lower than the MLP.

Additionally, by investigating the ability of models to predict the peak and low values of target, it is found in MLP model that the values of PFC for 90% division have the lowest values and close to zero as it ranges from (0.07–0.12) while the values for the other divisions were (0.11–0.125). In the same way, the values for LFC for 90% were also the lowest and more close to zero than the other divisions as it ranges from (0.24–0.7) while the values for other divisions ranges from (0.3–0.93) but it is noticed that values of LFC are higher than the PFC for same division and combination of inputs. This can be interpreted that the number of low values that are less than one-third the mean of low values is very small which in turn cause less accurate prediction of low values more than peak values however the values of LFC is acceptable for 90% division.

Similarly, in ENN, the 90% division has the lowest values for PFC and LFC as it ranges from (0.11–0.13) and (0.28–0.55) respectively which are lower than the other divisions.

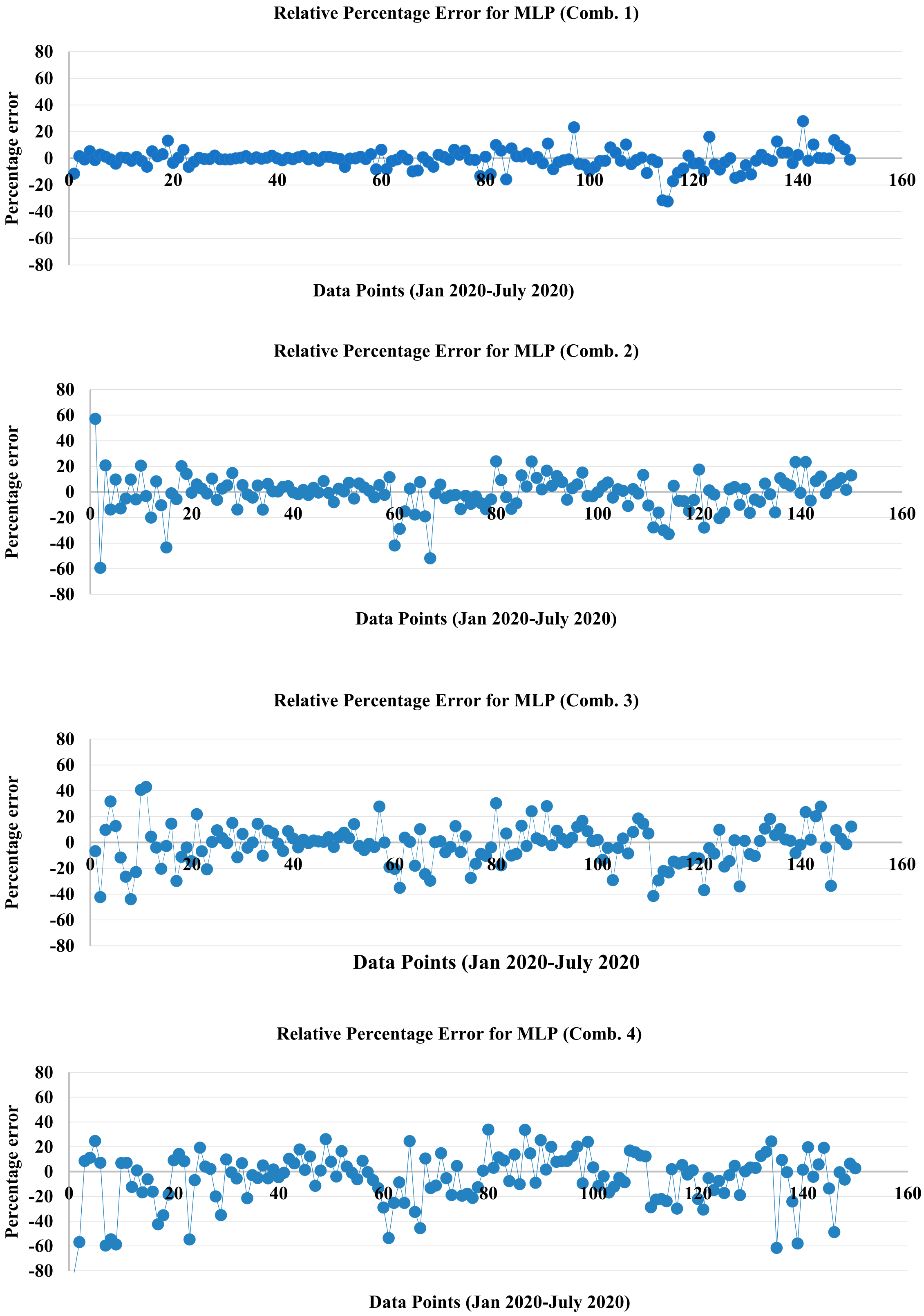
The study's objective was to obtain the optimum IR prediction model, be it 50%, 60%, 70%, 80%, or 90% data division model or any of the four-input combination models. As shown in the result section, all the five data divisions' comparison states that the models with 90% data division exhibit greater performance. Figures 4 and 5 present the relative percentage error plot for the models having 90% data division for all the four input combinations for MLP and ENN models.

This selection process corresponds to the first stage of the selection process, followed by the second stage, which includes selecting the most optimal model between MLP and ENN by comparing their performance criteria of 90% data division models. According to Table 3 and Figure 6, the max relative error for the MLP model in the four combination groups of parameters was (32.411–81.233) and that for ENN, it varies in the range of (44.55–56.58). MSE values for MLP for all the four-input combinations vary in the range of (0.01–0.08), and that of ENN it varies around 0.07. As stated previously, the MLP model with the first set of input combinations has the least relative percentage error (32.41%) and least MSE value (0.01).

Besides, referring to the NSE efficiency factor for MLP and ENN, the values for 90% division for the four

Table 3. Performance comparison between MLP, ENN for four combination groups of parameters and three divisions.

Neural networks model	Statistical criteria	Best Model Flow chart for 50% Data division				Best Model Flow chart for 60% Data division				Best Model Flow chart for 70% Data division				Best Model Flow chart for 80% Data division				Best Model Flow chart for 90% Data division			
		Comb 1	Comb 2	Comb3	Comb 4	Comb 1	Comb 2	Comb3	Comb 4	Comb 1	Comb 2	Comb3	Comb 4	Comb 1	Comb 2	Comb3	Comb 4	Comb 1	Comb 2	Comb3	Comb 4
MLP	Number of hidden Layers	2	2	3	3	2	2	2	2	2	2	2	2	2	2	3	2	3	2	2	2
	Number of Neurons in 1st layer	10	7	7	9	8	8	7	9	6	9	7	10	7	7	8	6	9	8	8	9
	Overall Regression	0.91	0.69	0.86	0.84	0.81	0.82	0.85	0.82	0.895	0.884	0.921	0.866	0.882	0.954	0.927	0.863	0.983	0.956	0.932	0.891
	Training Regression	0.95	0.77	0.97	0.91	0.85	0.85	0.90	0.87	0.953	0.915	0.975	0.919	0.897	0.990	0.964	0.988	0.993	0.988	0.927	0.898
	Validation Regression	0.86	0.52	0.74	0.78	0.71	0.74	0.75	0.66	0.793	0.785	0.787	0.739	0.848	0.733	0.605	0.770	0.637	0.804	0.970	0.835
	Testing Regression Value	0.72	0.71	0.77	0.76	0.78	0.84	0.81	0.83	0.790	0.741	0.752	0.738	0.812	0.875	0.884	0.739	0.966	0.935	0.969	0.940
	MSE	0.06	0.22	0.10	0.11	0.13	0.13	0.11	0.13	0.080	0.082	0.058	0.094	0.086	0.034	0.053	0.113	0.012	0.032	0.049	0.078
	MAE	0.18	0.38	0.21	0.26	0.29	0.27	0.24	0.28	0.203	0.218	0.160	0.236	0.225	0.103	0.156	0.160	0.067	0.136	0.168	0.219
	Max Relative Error Percentage	102.53	119.51	121.93	133.66	80.47	122.24	91.45	121.28	78.390	79.115	65.385	117.366	73.164	75.773	87.702	131.970	32.411	59.359	43.989	81.233
	Coefficient of determination, R^2	0.80	0.79	0.69	0.70	0.79	0.67	0.71	0.67	0.800	0.781	0.848	0.749	0.778	0.957	0.860	0.744	0.967	0.914	0.869	0.795
	Nash-Sutcliffe Efficiency (NSE)	0.81	0.63	0.53	0.63	0.48	0.61	0.64	0.62	0.833	0.729	0.723	0.675	0.839	0.800	0.738	0.693	0.965	0.905	0.843	0.717
	PFC	0.11	0.17	0.14	0.13	0.14	0.13	0.13	0.13	0.126	0.123	0.112	0.124	0.139	0.113	0.107	0.125	0.073	0.092	0.104	0.119
	LFC	0.38	0.78	0.52	0.82	0.60	0.81	0.31	0.31	0.296	0.659	0.303	0.701	0.484	0.928	0.747	0.592	0.242	0.642	0.463	0.704
ENN	Number of hidden Layers	3	3	2	3	2	2	2	2	3	2	2	2	3	2	3	2	2	3	3	2
	Number of Neurons	7	8	9	9	10	10	8	9	6	9	8	8	7	9	7	7	10	9	9	10
	Goal	0.04	0.04	0.04	0.04	0.04	0.040	0.03	0.03	0.05	0.02	0.04	0.01	0.03	0.04	0.03	0.03	0.01	0.02	0.02	0.05
	Overall Regression	0.844	0.863	0.843	0.853	0.900	0.854	0.857	0.826	0.906	0.897	0.894	0.873	0.892	0.901	0.877	0.880	0.909	0.900	0.895	0.903
	Train Regression	0.940	0.948	0.910	0.906	0.916	0.921	0.895	0.933	0.912	0.909	0.917	0.882	0.912	0.917	0.883	0.888	0.915	0.907	0.899	0.905
	Test Regression	0.757	1	0.794	0.775	0.800	0.755	0.819	0.733	0.877	0.889	0.851	0.860	0.800	0.839	0.876	0.849	0.873	0.865	0.864	0.884
	MSE	0.112	0.10	0.121	0.103	0.072	0.106	0.103	0.120	0.097	0.075	0.077	0.089	0.076	0.071	0.087	0.085	0.065	0.071	0.074	0.069
	MAE	0.225	0.229	0.251	0.255	0.189	0.237	0.236	0.255	0.242	0.202	0.199	0.230	0.205	0.197	0.224	0.225	0.192	0.198	0.200	0.207
	Max Relative Error Percentage	63.7	84.7	151.4	92.864	71.278	71.300	83.6	133.2	55.8	56.6	74.8	60.2	94.5	64.0	60.2	66.0	44.6	51.2	52.8	56.6
	Coefficient of determination, R^2	0.713	0.746	0.743	0.728	0.810	0.810	0.743	0.683	0.821	0.805	0.799	0.763	0.796	0.812	0.761	0.774	0.826	0.765	0.802	0.815
	Nash-Sutcliffe Efficiency (NSE)	0.673	0.693	0.669	0.661	0.780	0.780	0.669	0.578	0.833	0.729	0.723	0.675	0.839	0.800	0.738	0.693	0.965	0.905	0.843	0.717
	Peak Flow Criteria (PFC)	0.148	0.148	0.134	0.129	0.117	0.117	0.134	0.136	0.126	0.123	0.112	0.124	0.139	0.113	0.107	0.125	0.073	0.092	0.104	0.119
	Low Flow Criteria (LFC)	0.127	0.386	0.446	0.588	0.715	0.318	0.446	0.740	0.296	0.659	0.303	0.701	0.484	0.928	0.747	0.592	0.242	0.642	0.463	0.704


Relative Percentage Error for MLP (Comb. 1)

Relative Percentage Error for MLP (Comb. 2)

Relative Percentage Error for MLP (Comb. 3)

Relative Percentage Error for MLP (Comb. 4)

Figure 4. Relative percentage error plot for the four combinations of MLP model.

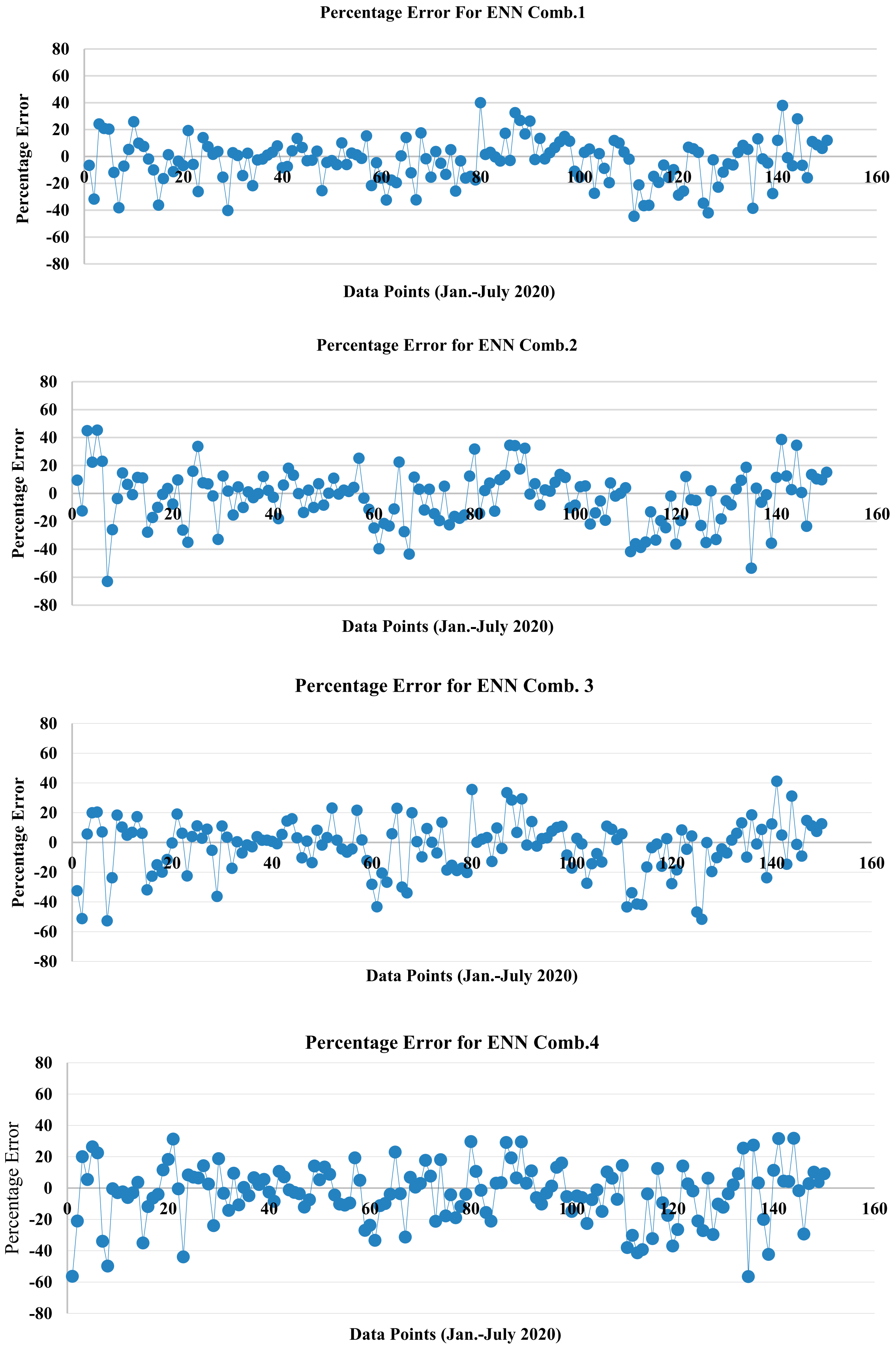


Figure 5. Relative percentage error plot for the four combinations of the ENN model.

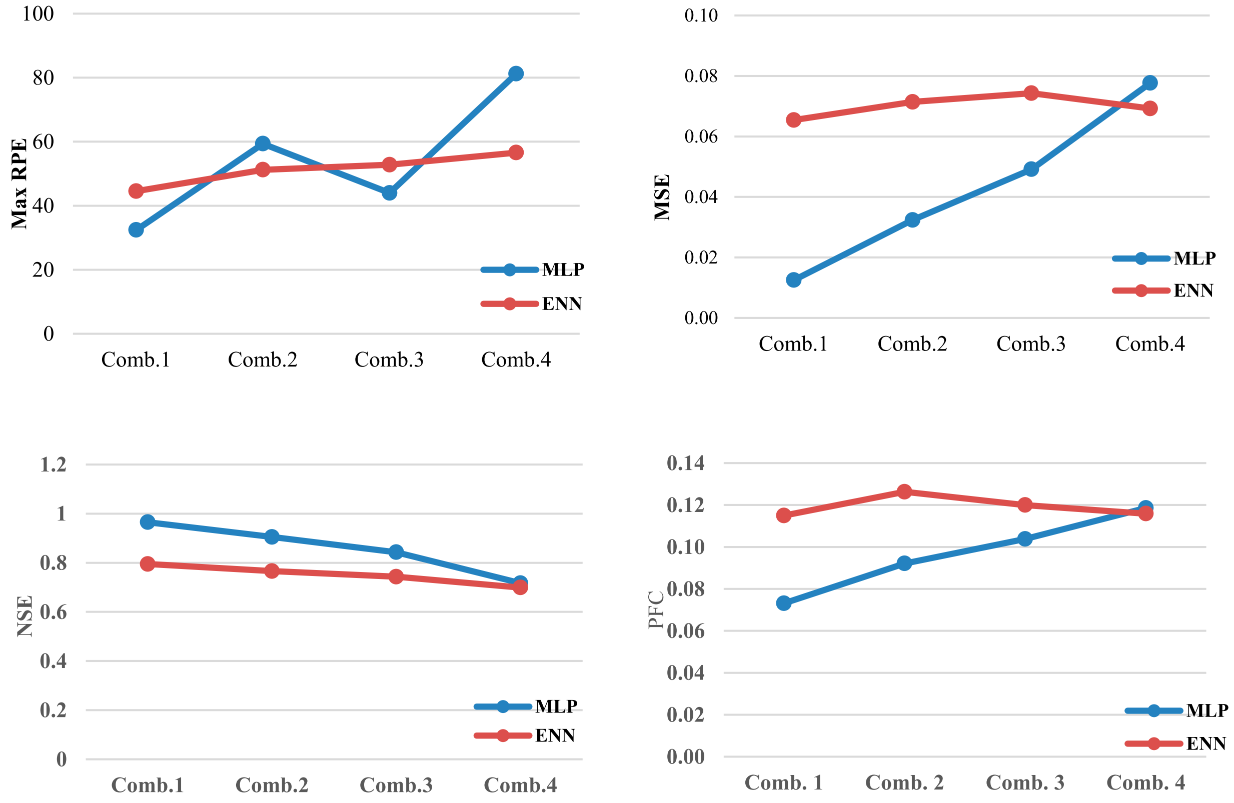


Figure 6. Comparison of max relative percentage error, MSE, NSE, PFC between MLP and ENN.

combinations were higher and very close to 1 for MLP than ENN. Since the value for the first combination of inputs is 0.965, which is much higher than for ENN with 0.795 for NSE, MLP with the first combination of inputs proved more efficient in predicting the IR. In addition to that, the PFC values for MLP were less than ENN and the lowest value was 0.073 for the first combination of inputs in MLP and the closest to zero. Therefore, the first combinations of MLP model show more simplicity.

Hence, the second selection stage provides the optimum model (multilayer perception model) which has the first set of input combination (Total flow (m^3d^{-1}), BOD (mgl^{-1}), TSS (mgl^{-1}), pH, HC (μScm^{-1}), TN (mgl^{-1}), TP (mgl^{-1})), and has the 90% data division which yielded the accuracy of the model as: the training regression: 0.993, the testing regression: 0.966, MSE: 0.012, and max relative percentage error: 32.41%. The NSE was 0.965 which is very close to 1 and means that it proves the highest efficiency. For the ability of model to predict the peak and low flow values, the PFC and LFC values were 0.073 and 0.242 respectively which are the lowest in comparison to other combination of inputs. Consequently, the first combination of inputs for MLP model fullfill all accuracy performance criteria, and efficiency so it is considered as the most optimal model.

This indicates that the selected seven inputs have a combined effect on the performance of models while by removing one or two or three (TP, TN, pH) parameters affect negatively on the overall performance even that these three parameters are not considered as the most prominent parameters that affect the infiltration rate directly. The fourth combination results by removing these three parameters resulted in the worst performance of the highest MSE, max relative error, and the least values for test regression and coefficient of determination. Besides, less values for the efficiency factor NSE.

Discussions

There is no evidence that a specific architecture model has the unique ability to apply for any locations, conditions, and patterns of data set in predicting the IR as it is clear from the literature reviews. Most of the literature that studied the prediction of IR using the multiple regression models or empirically on modeling the IR found that these models were applicable to a specific conditions (e.g. location, soil type, availability of data, etc.). Consequently, determining the best optimal multiple regression model needs comprehensive comparison for each field. In addition to that, it has been reported that the linear regression model is inconvinent for modeling infiltrations (Abdalrahman et al., 2021; Kashi et al., 2014; Pachepsky et al., 1996; Sy, 2006). While, most of the recent studies recommended using the ANN models in predicting IR, since ANN has the ability to detect the majority of previous studies that used multiple regression models either physically, conceptually, or empirically on modeling the infiltration rates stated that the applicability of these models are limited to some specific conditions (e.g. location, soil type, availability of data, etc.). Such studies required comprehensive comparison analysis and tests for each field to define the best model. Besides, it has been reported that the linear regression is inconvenient for modeling infiltration. Therefore, recent studies have been motivated to utilize ANN modeling approach to model IR, since ANN has the ability to detect the complex relations between the inputs and outputs, without being the relation explicitly transformed into mathematical equations. However, there are limitations and uncertainties associated with ANN modeling. The limitation accounts for overfitting of the ANN model which can be controlled by stopping the training when testing performance starts to decrease. In addition, integrating the ANN model with advanced optimization models such as the nature-inspired meta-heuristic algorithms to achieve the optimal ANN architecture, and hence, it is expected to achieve better IR prediction accuracy.

Besides, the literature reported both advantages and disadvantages for the different architectures of NN (MLP and ENN). Consequently, investigating the accuracy of different ANN architectures is required to determine the most optimal model. Hence, two different networks architectures (MLP as feedforward architecture and ENN as recurrent architecture) commonly used and proved high efficiency in predicting the IR were trained. In addition, a different division of data was used (90%, 80%, 70%, 60%, 50%) to find out the most optimal division as well as different combinations of inputs were trained to determine the best optimal combination and also to investigate the combined effect of input parameters and the effect of removing of each parameter on the overall performance of model.

By analysing the results of the first stage of modeling using MLP and ENN models, it is found that the MLP model has improved the accuracy and performance over the ENN model as shown in Table 4. The percentage of improvement in the best combination (i.e. first combination of inputs) of MLP over ENN was 81.5% for MSE, 27.25% for Max RPE, 21.38% for NSE, 36.52% for PFC and 39.95% for LFC. The superiority performance of MLP model complies with the studies of Alam et al. (2019), Sy (2006), Parchami-Araghi et al. (2013), Ekhmaj (2010), Al-Janobi et al. (2010), and Sarmadian and Taghizadeh-Mehrjardi (2014) whereas the MLP prove the superiority in predicting the IR over other models based on different input parameters as shown in Table 5.

Table 4. Percentage of performance improvement for MLP over ENN.

	Percentage of improvement of MLP over ENN			
Performance criteria	Comb.1	Comb. 2	Comb. 3	Comb.4
MSE	81.54	54.93	33.78	13.04
Max RPE	27.25	15.87	16.67	43.57
NSE	21.38	18.15	13.46	2.58
PFC	36.52	26.98	13.33	2.59
LFC	39.95	128.47	15.97	10

For analyzing the sufficiency of data records, 150 records were used of the TWW quality parameters and HLR as inputs in parallel with the IR as the target output in the period of January 2020–July 2020. From the literature, as shown in Table 5, the amount of data set ranging from 32–230 records that were used in training different models and get high performance accuracy results. Consequently, 150 records were considered enough to get accurate models.

Since almost all recent studies have stated that the method of data division has a significant impact on the results of models, so the used method of data division was the index data division in which each set has the same statistical characteristics such as the same mean value. Besides, the training dataset has the maximum and minimum values of the target to enable the model to train different patterns of data and ensure that optimal performance is achieved.

From the literature, Anari et al. (2011), Jain and Kumar (2006), Kashi et al. (2014), Parchami-Araghi et al. (2013), Sihag et al. (2017b) have used the default division of data (70% training, 15%validation, 15% testing). While, Sy (2006) and Ekhmaj (2010) have used other division of data (60% training, 20% validation, 20% testing). Alam et al. (2019) has used (85% Training, 15% Testing) as shown in Table 5. In these studies, they depend only on the default division or one other division without investigating the effect of different percentages of division.

While in this study, five different data divisions; (90%,80%,70%, 60%, 50%) for training, (5%, 10%, 15%, 20%, 25%) for validation, (5%,10%, 15%, 20%, 25%) for testing were used by index method followed by their comparison to get the best division that get the highest performance model. Contrary to the previous studies, 90% division of data was the optimal division of data whereas the model was trained on the highest amount of data which allowed the model to learn different patterns and general trends of data so that the ability of generalization improves accordingly.

By comparing the current study with the previous research that focus on predicting the infiltration rate, it is clear that all of the previous research used different soil parameters such as EC, soil texture, lime percentage, sodium adsorption ratio, soil moisture content and bulk density (Alam et al., 2019; Kashi et al., 2014; Sy, 2006), first and second 5 min time period of infiltration rate (Anari et al., 2011), rainfall and runoff quantity (Jain & Kumar, 2006), different categories of mass fractions of particles (Pachepsky et al., 1996), and time, % Clay,

Table 5. Comparison between the results of previous studies.

#	Authors	Data set	Input variables	Prediction variable	Percentage of data division	Performance criteria	ANN models	Best models
–	Current study	150	BOD TSS HLR EC pH TN TP	IR	(70, 15,15) (80, 10,10) (90, 5,5)	MSE MAE Max RPE R2 NSE PFC LFC	MLP ENN ACO-ENN	ACO-ENN (90%, 5%, 5%)
1	Anari et al. (2011)	32	Time (1st and 2nd 5 min)	Total IR	(70,15,15)	NSE RMSE R^2	LLR DLLR MLP RNN ANFIS	ANFIS ($E = 0.94$, RMSE $= 0.51$, $R^2 = 0.94$) followed by MLP and RNN ($E = 0.93$, RMSE $= 0.59$, $R^2 = 0.93$)
2	Kashi et al. (2014)	200	EC Soil texture % lime SAR BD	CEC and IR	(70,15,15)	RMSE MAE ME R^2	MR MLP RBF ANFIS MR	MLP model followed by ANFIS, then RBF (RMSE $= 0.63$, MAE $= 0.59$, ME $= -0.18$, $R^2 = 0.97$)
3	Pachepsky et al. (1996)	230	Different Mass fractions of particles Density.	Soil Water Retention	(70, 15,15)	- MSE - Regression - R^2	MLP and Regression models	MLP
4	Sy (2006)	80	% Sand % Silt % Clay HC Slope BD SM	Infiltration Rate	(60,20,20)	- MSE - R^2	MLP, MR	MLP (MSE $= 0.0027$, $R^2 = 0.95$)
5	Ekhmaj (2010)	159	% Sand % Silt % Clay BD Saturated HC WC	Steady IR	(60,19,21)	MAE RMSE R D (Agreement index)	MLP and MR	ANN (MLP) (MAE $= 1.19$, RMSE $= 1.82$, $R = 0.95$)
6	Sihag et al. (2017b)	138	Time % Clay % Sand % Fly Ash Density	IR	(70,30)	RMSE R NSE	ANN Gaussian process, SVM Random forest M5P model tree, SCS Kostiakov	SVM with RBF kernel (CC $= 0.9133$, RMSE $= 0.0911$, and NSE $= 08302$) followed by GP, random forest, ANN, and M5p Model Tree

(*continued*)

Table 5. Continued.

#	Authors	Data set	Input variables	Prediction variable	Percentage of data division	Performance criteria	ANN models	Best models
7	Parchami-Araghi et al. (2013)	210	% sand %silt Clay SM Particle size distribution BD Particle Density Organic Carbon CaCO3 Gravel Contents Permanent wilting point	Cumulative Infiltration rate at 5, 10, 15, 20, 30, 45, 60, 90, 120, 150, 180, 210, 240, and 270 min after the start of infiltration process	(70,30)	R RMSE ME CRM MD	**Regression Models:** (Green and Ampt, Philip, Kostiakov, Horton, Kostiakov–Lewis, USDA-NRCS) **MLP**	MLP (RMSE = 2.8390)
8	Alam et al. (2019)	600	% Sand % Silt % Clay BD Particle Density Porosity Moisture Content HC (Topsoil and subsoil)	IR	(85,15)	Regression Plots	MLP	MLP (Test Regression = 0.997, Training Regression = 0.89, Validation = 0.90)

Notes: Local Linear Regression (LLR), Dynamic Local Linear Regression (DLLR), Adaptive Neuro Fuzzy Inference System (ANFIS), Runoff Optimization Model (ROM), support vector machine (SVM), Soil moisture (SM), Agreement index (D), the potential infiltration rate (f_t), the constant steady-state IR after sufficient time has elapsed (mm/h) (f_c), the initial infiltration rate at the beginning of a rainfall event (mm/h) (f_0), an exponential decay constant (h^{-1}) (K).

%Sand, %Fly Ash, %Density (Sihag et al., 2017b). Therefore, investigating the IR using different ANN models depending on the TWW characteristics parameters represent the novelty of the current study as the used TWW quality parameters were (BOD, TSS, EC, pH, TN, TP) and HLR.

From the analysis of the results of different combinations of input parameters for MLP model, it is observed that the first combination which consists of all the seven parameters has the highest performance over the other combinations which means that there is an integrated effect of parameters on the accurate prediction of the target IR. Therefore, removing one, two, or three parameters (TP, TN, pH) negatively effect on the overall performance even that these three parameters are not commonly mentioned in the literature reviews as the most prominent parameters that affect the infiltration rate directly. Therefore, the fourth combination results by removing these three parameters resulted in the worst performance of the highest MSE, max relative error, and the least values for test regression and the efficiency factor NSE. These trend of results whereas the combined effect resulted in the best accurate models compiled with the results of the study of Kashi et al. (2014) and Sy (2006) in which they found that the combination of all parameters resulted in the best accurate model.

For investigating the relations between the model performance represented by MSE with the number of neurons, and combination of inputs, it is observed within this research that as the number of neurons increased, the MSE decreased as it is clear on the plot of first combination of inputs using MLP and ENN models in Figure 7, and Figure 8. This can be explicated that by increasing the number of neurons, the complexity of the network increase which help the model to learn different patterns in the target data. However, increasing the number of hidden layers and neurons to a certain limit can lead to overcomplex structure of model and affect inversely on the performance of models. While for the hidden layer's numbers, it is observed that both two and three hidden layers were efficient to get the lowest MSE for models. On the other hand, Kashi et al. (2014) stated in their study that there is an irregular variation of the RMSE for the different neurons and concluded that determining the optimal number of neurons is impossible.

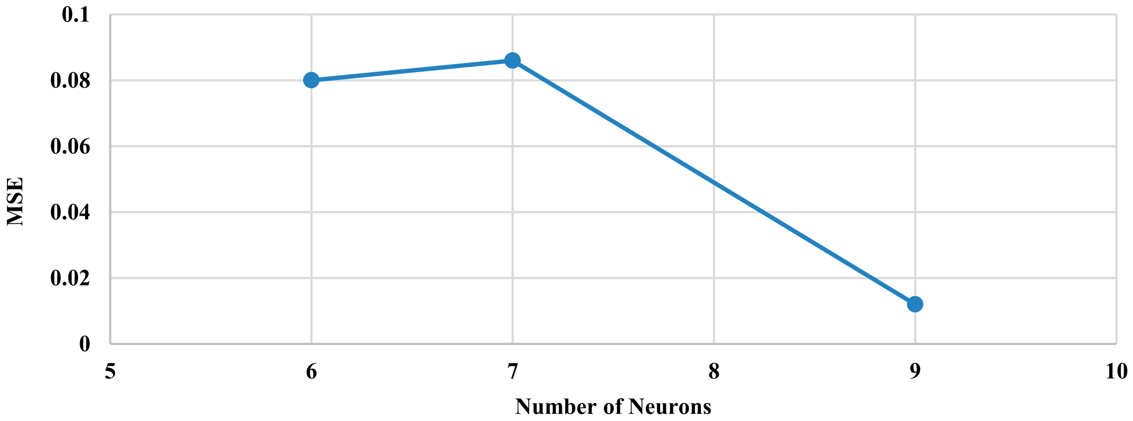


Figure 7. Plot of MSE vs. number of neurons for MLP.

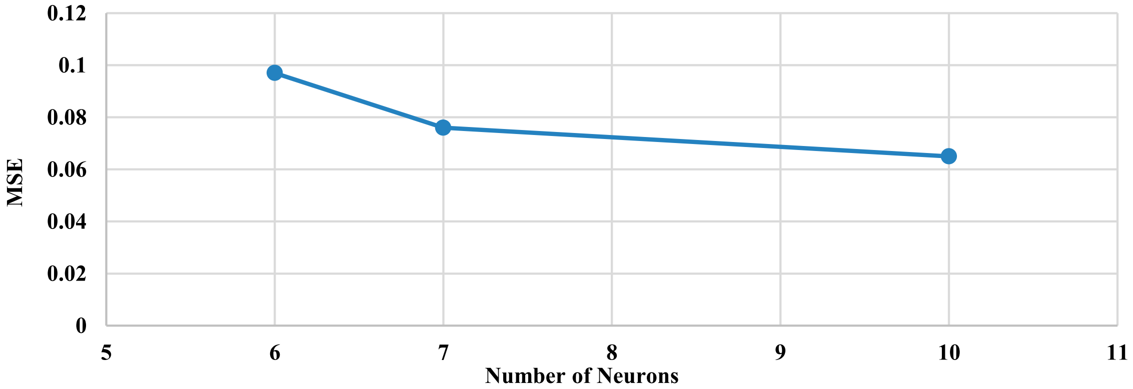


Figure 8. Plot of MSE vs. number of neurons for ENN.

By comparison of the used performance criteria in this study with the literature research in same field, it is observed that it is sufficient and cover almost all of the used criteria as shown in Table 5. Besides, it is observed that the performance criteria were consistent to each other except the LFC criteria which give high values. This can be justified that the collected observed data has a small number of low values less than one-third of mean value which in turn effect on the ability of LFC to predict low flow values.

Since in MLP and ENN some of the performance criteria have close values between different combinations so the comparison process depends prominently on test regression, max RPE, NSE, PFC.

In general, models are exposed to several sources of uncertainties that may have resulted from the input data, the architecture of model, weights, and biases (Chitsazan et al., 2015). In this study, there are very few missing in input data that did not exceed (3–4%) of total data that is filled using interpolation functions; thus, it can form one of the uncertainty sources of model input. Consequently, the average uncertainty of the optimum model was calculated using the following equation (Zeleňáková et al., 2013): $\sigma = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{|x-y|}{x}\right) \times 100$ where σ is the average of uncertainty percentage, n is the number of input data, x is the observed value, y is the predicted value. The calculated average uncertainty percentage for the selected optimum model (the first combination of 90% division in MLP model) was 4.66% which is considered as acceptable low uncertainty percentage. Generally, for better accurate prediction of models it should minimize the sources of error and uncertainty such as checking outlier values and precise pre-processing of the input data. In addition, it should be pointed out that this study is conducted on a case study of specific infiltration basins that have homogeneous coarse sand (Kurkar) for a depth of two meters under the bottom of basins. Therefore, it is expected that the performance of models may be different or not applicable for the other types of soils.

In this study, the sensitivity analysis was conducted using the backward stepwise method through training all the seven parameters of TWW quality parameters then omitting one by one parameter to test the sensitivity of this parameter on predicting the target of IR. The results of the training were analyzed based on the MSE performance criteria such that the increase in MSE values means that the corresponding parameter is highly sensitive and has a high effect if it is omitted.

Table 6. The results of backward stepwise method of sensitivity analysis.

	Layers	Neurons	Goal	Test regression	Train regression	MSE	MAE	MAX_RE
All-Total flow	2	10	0.05	0.924	0.906	0.057	0.191	64.068
All-BOD	2	10	0.05	0.885	0.893	0.063	0.215	70.557
All-TSS	2	10	0.04	0.886	0.908	0.074	0.183	58.046
All-pH	2	10	0.05	0.865	0.908	0.114	0.234	67.739
All-EC	2	10	0.05	0.884	0.927	0.114	0.267	85.218
All-TP	2	10	0.05	0.709	0.843	0.172	0.330	84.868
All-TN	2	10	0.01	0.748	0.904	0.230	0.393	82.672

As shown in Table 6, when TN, TP parameters were omitted then high MSE value were recorded (i.e. 0.23, 0.17), hence, they were considered as the most sensitive parameters, followed by EC, pH parameters with the same MSE value of 0.114, followed by TSS, BOD, and Total flow with MSE values of 0.074, 0.063, 0.057 respectively.

The results of sensitivity analysis confirm and agree with the results of the comparison between the combinations of parameters as the first combination is the most optimum model in predicting the IR which means that all parameters have a combined effect on the IR. Therefore, when the TP and TN were omitted in the second and third combinations (comb.2, comb.3), the results become worse. While in the fourth combination (comb.4) when the TP, TN, pH was omitted, the results become the worst. Consequently, for getting best prediction of the IR, all parameters should be involved as input parameters. From the literature review, there is not a focus studies on the TP, and TN as major factors affecting the infiltration rate, but it has a high sensitivity and omitting them can reduce the accuracy of predicting the IR. Therefore, it is highly recommended to conduct more research on these two factors.

It is worth mentioning that the explication of the high sensitivity of TN is justified by the fact that the existence of TN components helps in growing plants and algae as well as it is highly related to aerobic and anoxic redox conditions which can facilitate nitrogen consumption through firstly the nitrification process that transforms NH_4^+ to NO_3 followed by the denitrification process in which NO_3-can be transformed into nitrogen and nitrous gases that can be emitted to the atmosphere. Therefore, a short cycle of drying and wetting process is recommended instead of long cycles to facilitate the oxidize of all nitrogen to nitrate (Barry et al., 2017; Bouwer, 2002).

For the high sensitivity of TP, as it is mentioned in the literature review, phosphorus components can be removed from the infiltrated water through fast sorption and/or slow precipitation reactions to amorphous or crystalline forms. Besides, it has insolubility nature and can precipitate with iron and aluminum in acidic condition, while in alkaline condition it can precipitate with calcium and magnesium to form calcium phosphate and magnesium phosphate which in turn cause a calcification and gypsification in soil. Consequently, TP components affect on the infiltration rate and clogging of soils (Phong et al., 2013).

By comparison, the results of this sensitivity analysis using the same approach of backward stepwise with the study of Kashi et al. (2014) and Sihag et al. (2017b) as they concluded also that the combined effects of all parameters have the best performance but different parameters that related to soil characteristics were used.

Conclusion

This study was conducted to select the optimal model in predicting the infiltration rate of TWW through infiltration basins using a different combination of input parameters related to TWW quality characteristics and the hydraulic loading rate. According to results, the model with 90% division of inputs was the best performance model over the other divisions of inputs depending on MLP and ENN's statistical criteria. These results indicate that the higher percentage of data included in the training process (90%) will improve models' performance as it can train the model for a large amount of data with different patterns. While the comparison between the two models with the 90% division of data resulted in the superiority of MLP over ENN.

Furthermore, the first combination group of inputs which include all the seven inputs of parameters shows the best performance over the other combinations for the same ENN model with the 90% division as it has the lowest MSE, max RPE, PFC, and LFC as well as the highest values for test regression, and NSE value. This result was compiled with the result of sensitivity analysis as the first combination with all the seven parameters and when the TP and TN were omitted in the second and third combinations (comb.2, comb.3), the results became worse. While in the fourth combination (comb.4) when the TP, TN, pH was omitted, the results became the worst. Based on the sensitivity analysis by using back stepwise method, the results confirm and agree with the results of the comparison between the combinations of parameters. Omitting one of the most sensitive parameters (TP, TN, and pH) affects the accuracy of IR prediction that interprets the lower accuracy of the second, third, and fourth combination of inputs than the first combination group. Finally, it is worth mentioning that this study is conducted on a case study of specific infiltration basins that have homogeneous coarse sand (Kurkar) for a depth of two meters on the bottom of basins. Therefore, it is expected that the performance of models may be different or not applicable for the other types of soils.

Acknowledgements

The authors would like to acknowledge the access of data from the Palestinian Water Authority (PWA). Conceptualization done by Ghada Abdalrahman, Ahmed El-shafie and Mohsen Sherif; Data curation done by Ghada Abdalrahman, Sai Hin Lai and Ali Najah Ahmed; Methodology done by Ghada Abdalrahman and Ahmed Sefelnasr; Software developed by Ghada Abdalrahman and Pavitra Kumar; Supervision done by Sai Hin Lai and Ahmed El-shafie; Visualization done by Ghada Abdalrahman; Writing – original draft done by Ghada Abdalrahman and Ahmed El-shafie; Writing – review & editing done by Ali Najah Ahmed, Mohsen Sherif, Ahmed Sefelnasr and Kwok wing Chau.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the Institut Pengurusan dan Pemantauan Penyelidikan, Universiti Malaya, Malaysia [grant number RP025A-18SUS]; United Arab Emirate University (UAEU) within the initiatives of the Asian Universities Alliance (AUA) collaboration [IF059-2021].

ORCID

Ali Najah Ahmed http://orcid.org/0000-0002-5618-6663
Mohsen Sherif http://orcid.org/0000-0002-6368-8143
Ahmed Sefelnasr http://orcid.org/0000-0002-0281-6455
Kwok Wing Chau http://orcid.org/0000-0001-6457-161X
Ahmed Elshafie http://orcid.org/0000-0001-5018-8505

References

2540 SOLIDS. (2018). *Standard methods for the examination of water and wastewater*. American Public Health Association. https://doi.org/10.2105/SMWW.2882.030

4500-N NITROGEN. (2018). *Standard methods for the examination of water and wastewater* (23rd ed.). https://doi.org/10.2105/smww.2882.086

4500-P PHOSPHORUS. (2018). *Standard methods for the examination of water and wastewater*. American Public Health Association. https://doi.org/10.2105/SMWW.2882.093

Aaltomaa, T., & Joy, D. M. (2002). Field testing of absorption bed clogging. In R. G. Zytner (Ed.), *An international perspective on environmental engineering*. Canadian Society for Civil Engineering. 14 p.

Abdalrahman, G. A. M., Lai, S. H., Snounu, I., Kumar, P., Sefelnasr, A., Sherif, M., & El-shafie, A. (2021). Review on wastewater treatment ponds clogging under artificial recharge: Impacting factors and future modelling. *Journal of Water Process Engineering*, *40*, Article 101848. https://doi.org/10.1016/j.jwpe.2020.101848

Aboukarima, A., Al-Sulaiman, M., & Elmarazky, M. S. (2018). Effect of sodium adsorption ratio and electric conductivity of the applied water on infiltration in a sandy-loam soil. *Water SA*, *44*, 105. https://doi.org/10.4314/wsa.v44i1.12

Alam, T., Barua, S., & Anisa, H. (2019). Presentation on soil properties using evaluation of infiltration rate artificial neural network.

Al-Janobi, A., Aboukarima, A., & Ahmed, K. (2010). Modeling water infiltration rate under conventional tillage systems on a clay soil using artificial neural networks. *Australian Journal of Basic and Applied Sciences*, *4*(8), 3869–3879. http://www.insipub.com/ajbas/2010/386.

Anari, P., Darani, H., & Nafarzadegan, A. R. (2011). Application of ANN and ANFIS models for estimating total infiltration rate in an arid rangeland ecosystem. *Research Journal of Environmental Sciences*, *5*(3), 236–247. https://doi.org/10.3923/rjes.2011.236.247

Ash, T. (1989). Dynamic node creation in backpropagation networks. *Connection Science*, *1*(4), 365–375. https://doi.org/10.1080/09540098908915647

Azimi, H., Bonakdari, H., & Ebtehaj, I. (2019). Design of radial basis function-based support vector regression in predicting the discharge coefficient of a side weir in a trapezoidal channel. *Applied Water Science*, *9*(4), 78. https://doi.org/10.1007/s13201-019-0961-5

Barry, K., Vanderzalm, J., Miotliński, K., & Dillon, P. (2017). Assessing the impact of recycled water quality and clogging on infiltration rates at a pioneering soil aquifer treatment (SAT) site in alice springs, Northern Territory (NT), Australia. *Water*, *9*(3), 179. https://doi.org/10.3390/w9030179

Bonakdari, H., Moeeni, H., Ebtehaj, I., Zeynoddin, M., Mahoammadian, A., & Gharabaghi, B. (2019). New insights into soil temperature time series modeling: Linear or nonlinear? *Theoretical and Applied Climatology*, *135*(3), 1157–1177. https://doi.org/10.1007/s00704-018-2436-2

Bouwer, H. (2002). Artificial recharge of groundwater: Hydrogeology and engineering. *Hydrogeology Journal*, *10*(1), 121–142. https://doi.org/10.1007/s10040-001-0182-4

Chen, Y., Song, L., Liu, Y., Yang, L., & Li, D. (2020). A review of the artificial neural network models for water quality prediction. *Applied Sciences*, *10*(17), 5776. https://doi.org/10.3390/app10175776

Chitsazan, N., Nadiri, A. A., & Tsai, F. T. C. (2015). Prediction and structural uncertainty analyses of artificial neural networks using hierarchical Bayesian model averaging. *Journal of Hydrology*, *528*, 52–62. https://doi.org/10.1016/j.jhydrol.2015.06.007

Coulibaly, P., Anctil, F., & Bobée, B. (2001). Multivariate reservoir inflow forecasting using temporal neural networks. *Journal of Hydrologic Engineering*, *6*(5), 367–376. https://doi.org/10.1061/(ASCE)1084-0699(2001)6:5(367

Ebtehaj, I., & Bonakdari, H. (2014). Comparison of genetic algorithm and imperialist competitive algorithms in predicting bed load transport in clean pipe. *Water Science and Technology*, *70*(10), 1695–1701. https://doi.org/10.2166/wst.2014.434

Ebtehaj, I., & Bonakdari, H. (2016). Assessment of evolutionary algorithms in predicting non-deposition sediment transport. *Urban Water Journal*, *13*(5), 499–510. https://doi.org/10.1080/1573062X.2014.994003

Ekhmaj, A. I. (2010, September 10–12). Predicting soil infiltration rate using artificial neural network. *2010 International Conference on Environmental Engineering and Applications*. IEEE. https://doi.org/10.1109/ICEEA.2010.5596107

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1

El-Shafie, A., Mazoghi, H., AbouKheira, A., & Taha, M. (2011). Artificial neural network technique for rainfall forecasting applied to Alexandria, Egypt. *International Journal of the Physical Sciences*, *6*(6), 1306–1316. https://doi.org/10.5897/IJPS11.143

Emdad, M., Raine, S., Smith, R., & Fardad, H. (2004). Effect of water quality on soil structure and infiltration under furrow irrigation. *Irrigation Science*, *23*(2), 55–60. https://doi.org/10.1007/s00271-004-0093-y

Environmental Protection Agency. (2004). *Guidelines for water reuse*. https://cfpub.epa.gov/si/si_public_record_report.cfm?Lab = NRMRL&dirEntryId = 129543

Gharaibeh, M. A., Ghezzehei, T. A., Albalasmeh, A. A., & Alghzawi, M. I. Z. (2016). Alteration of physical and chemical characteristics of clayey soils by irrigation with treated waste water. *Geoderma*, *276*, 33–40. https://doi.org/10.1016/j.geoderma.2016.04.011

Haghighi Fashi, F., Gorji, M., Shorafa, M., Sarmadian, F., & Mohammadi, M. (2010). Evaluation of some infiltration models and hydraulic parameters. *Spanish Journal of Agricultural Research*, *8*(1), 210–217. https://doi.org/10.5424/sjar/2010081-1160

Horneck, D., Ellsworth, J., Hopkins, B., Sullivan, D., & Stevens, R. (2007). Managing salt-affected soils for crop production.

Huo, S., He, Z., Su, J., Xi, B., & Zhu, C. (2013). Using artificial neural network models for eutrophication prediction. *Procedia Environmental Sciences*, *18*, 310–316. https://doi.org/10.1016/j.proenv.2013.04.040

Jain, A., & Kumar, A. (2006). An evaluation of artificial neural network technique for the determination of infiltration model parameters. *Applied Soft Computing*, *6*(3), 272–282. https://doi.org/10.1016/j.asoc.2004.12.007

Jnad, I., Lesikar, B., Kenimer, A., & Sabbagh, G. (2001). Subsurface drip dispersal of residential effluent: II. Soil hydraulic characteristics. *Transactions of the ASAE*, *44*. https://doi.org/10.13031/2013.6443

Kashi, H., Emamgholizadeh, S., & Ghorbani, H. (2014). Estimation of soil infiltration and cation exchange capacity based on multiple regression, ANN (RBF, MLP), and ANFIS models. *Communications in Soil Science and Plant Analysis*, *45*(9), 1195–1213. https://doi.org/10.1080/00103624.2013.874029

Kumar, P., Lai, S. H., Mohd, N. S., Kamal, M. R., Afan, H. A., Ahmed, A. N., Sherif, M., Sefelnasr, A., & El-shafie, A. (2020). Optimised neural network model for river-nitrogen prediction utilizing a new training approach. *PLOS ONE*, *15*(9), e0239509. https://doi.org/10.1371/journal.pone.0239509

Kumar, P., Lai, S. H., Wong, J. K., Mohd, N. S., Kamal, M. R., Afan, H. A., Ahmed, A. N., Sherif, M. S., & Efelnasr, A. E.-S. (2020). A review of nitrogen compounds prediction in water bodies using Artificial Neural Networks and other models. *Sustainability*, *12*(11), 4359. https://doi.org/10.3390/su12114359

Lagos-Avid, M. P., & Bonilla, C. A. (2017). Predicting the particle size distribution of eroded sediment using artificial neural networks. *Science of The Total Environment*, *581–582*, 833–839. https://doi.org/10.1016/j.scitotenv.2017.01.020

Liu, S., Yan, M., Tai, H., Xu, L., & Li, D. (2012). Prediction of dissolved oxygen content in aquaculture of hyriopsis cumingii using Elman neural network. In D. Li & Y. Chen (Eds.), *Computer and computing technologies in agriculture V* (pp. 505–518). Springer.

Lu, H., Li, H., Liu, T., Fan, Y., Yuan, Y., Xie, M., & Qian, X. (2019). Simulating heavy metal concentrations in an aquatic environment using artificial intelligence models and physic-chemical indexes. *Science of the Total Environment*, *694*, Article 133591. https://doi.org/10.1016/j.scitotenv.2019.133591

Magesan, G. N. (2001). Changes in soil physical properties after irrigation of two forested soils with municipal wastewater. *New Zealand Journal of Forestry Science*, *31*(2), 188–195.

Moeeni, H., Bonakdari, H., & Ebtehaj, I. (2017). Integrated SARIMA with neuro-fuzzy systems and neural networks for monthly inflow prediction. *Water Resources Management*, *31*(7), 2141–2156. https://doi.org/10.1007/s11269-017-1632-7

Özcan, F., Atiş, C. D., Karahan, O., Uncuoğlu, E., & Tanyildizi, H. (2009). Comparison of artificial neural network and fuzzy logic models for prediction of long-term compressive strength of silica fume concrete. *Advances in Engineering Software*, *40*(9), 856–863. https://doi.org/10.1016/j.advengsoft.2009.01.005

Pachepsky, Y. A., Timlin, D., & Varallyay, G. (1996). Artificial neural networks to estimate soil water retention from easily measurable data. *Soil Science Society of America Journal*, *60*(3), 727–733. https://doi.org/10.2136/sssaj1996.03615995006000030007x

Palestinian Water Authority. (2020). *North gaza wastewater treatment plant* (No. Monthly report Nr 10).

Panahi, M., Khosravi, K., Ahmad, S., Panahi, S., Heddam, S., Melesse, A. M., Omidvar, E., & Lee, C.-W. (2021). Cumulative infiltration and infiltration rate prediction using optimized deep learning algorithms: A study in western Iran. *Journal of Hydrology: Regional Studies*, *35*, Article 100825. https://doi.org/10.1016/j.ejrh.2021.100825

Parchami-Araghi, F., Mirlatifi, S. M., Ghorbani Dashtaki, S., & Mahdian, M. H. (2013). Point estimation of soil water infiltration process using artificial neural networks for some calcareous soils. *Journal of Hydrology*, *481*, 35–47. https://doi.org/10.1016/j.jhydrol.2012.12.007

Phong, T., Phuc, D. H., Phi, T., & Hiramatsu, K. (2013). Measuring load of phosphate in the environment by passive sampling techniques – an introduction. *Journal of the Faculty of Agriculture Kyushu University*, *58*(1), 153–157. https://doi.org/10.5109/26175

Riad, P., Billib, M., Boochs, P., Hassan, A., Eldin, M., & Omar, M. (2013). Vadose zone wells and surface spreading basins modeling to augment the groundwater resources in a semi arid area in Egypt. https://doi.org/10.13140/RG.2.2.21141.17126

Sarmadian, F., & Taghizadeh-Mehrjardi, R. (2014). Estimation of infiltration rate and deep percolation water using feed-forward neural networks in Gorgan province. *Eurasian*

Journal of Soil Science (EJSS), *3*, 1–6. https://doi.org/10.18393/ejss.03148

Sharaf El Din, E., Zhang, Y., & Suliman, A. (2017). Mapping concentrations of surface water quality parameters using a novel remote sensing and artificial intelligence framework. *International Journal of Remote Sensing*, *38*(4), 1023–1042. https://doi.org/10.1080/01431161.2016.1275056

Sihag, P. (2018). Prediction of unsaturated hydraulic conductivity using fuzzy logic and artificial neural network. *Modeling Earth Systems and Environment*, *4*(1), 189–198. https://doi.org/10.1007/s40808-018-0434-0

Sihag, P., Tiwari, N. K., & Ranjan, S. (2017a). Estimation and inter-comparison of infiltration models. *Water Science*, *31*(1), 34–43. https://doi.org/10.1016/j.wsj.2017.03.001

Sihag, P., Tiwari, N. K., & Ranjan, S. (2017b). Modelling of infiltration of sandy soil using Gaussian process regression. *Modeling Earth Systems and Environment*, *3*(3), 1091–1100. https://doi.org/10.1007/s40808-017-0357-1

Sihag, P., Tiwari, N. K., & Ranjan, S. (2019). Prediction of unsaturated hydraulic conductivity using adaptive neuro- fuzzy inference system (ANFIS). *ISH Journal of Hydraulic Engineering*, *25*(2), 132–142. https://doi.org/10.1080/09715010.2017.1381861

Stangierski, J., Weiss, D., & Kaczmarek, A. (2019). Multiple regression models and artificial neural network (ANN) as prediction tools of changes in overall quality during the storage of spreadable processed gouda cheese. *European Food Research and Technology*, *245*(11), 2539–2547. https://doi.org/10.1007/s00217-019-03369-y

Suarez, D. L., & Gonzalez-Rubio, A. (2017). Effects of the dissolved organic carbon of treated municipal wastewater on soil infiltration as related to sodium adsorption ratio and pH. *Soil Science Society of America Journal*, *81*(3), 602–611. https://doi.org/10.2136/sssaj2016.09.0310

SWECO. (2003). *Northern Gaza WWTP infiltration system*.

Sy, N. (2006). Modelling the infiltration process with a multilayer perceptron artificial neural network. *Hydrological Sciences Journal*, *51*(1), 3–20. https://doi.org/10.1623/hysj.51.1.3

Toha, S. F., & Tokhi, M. O. (2008, September 9–10). MLP and Elman recurrent neural network modelling for the TRMS. *2008 7th IEEE International Conference on Cybernetic Intelligent Systems*. IEEE. https://doi.org/10.1109/UKRICIS.2008.4798969

Turner, E. (2006). *Comparison of infiltration equations and their field validation with rainfall simulation*. University of Maryland. https://drum.lib.umd.edu/bitstream/handle/1903/4218/umi-umd-4033.pdf?sequence = 1&isAllowed = y

Vandevivere, P., & Baveye, P. (1992). Saturated hydraulic conductivity reduction caused by aerobic bacteria in sand columns. *Soil Science Society of America Journal*, *56*(1), 1–13. https://doi.org/10.2136/sssaj1992.03615995005600010001x

World Health Organization Regional Office for the Eastern Mediterranean. (2006). A compendium of standards for wastewater reuse in the Eastern Mediterranean region.

Zeleňáková, M., Čarnogurská, M., Šlezingr, M., Słyś, D., & Purcz, P. (2013). A model based on dimensional analysis for prediction of nitrogen and phosphorus concentrations at the river station Ižkovce, Slovakia. *Hydrology and Earth System Sciences*, *17*(1), 201–209. https://doi.org/10.5194/hess-17-201-2013

Zhang, Z., Zheng, T., & Vairappan, C. (2007). A novel learning method for Elman neural network using local search. *Neural Information Processing – Letters and Reviews*, *11*(8), 181–188.

Index 1

1. **Code for Data Division in MLP model**:

```
l = length(Target);
vallength = int64(l∗0.15); % For Validation
testlength = int64(l∗0.15); % For Testing
trainlength = l-(testlength+vallength);
trainvalue = zeros(1,l);
valvalue = zeros(1,vallength);
testvalue = zeros(1,testlength);
trainindex = zeros(1,trainlength);
valindex = zeros(1,vallength);
testindex = zeros(1,testlength);
datamean = zeros(20,3);
Alltrainindex = zeros(20,trainlength);
Allvalindex = zeros(20,vallength);
Alltestindex = zeros(20,testlength);
maxnumber = max(Target);
minnumber = min(Target);
maxindex = find(Target = = maxnumber);
minindex = find(Target = = minnumber);
originalrindex = zeros(20,(testlength+vallength+2));
modifiedrindex = zeros(20,(testlength+vallength));
for n = 1:1:20
   rindex = randperm(l,(testlength+vallength+2));
   originalrindex(n,:) = rindex;
   rindex(rindex = = maxindex(1)) = [];
   rindex(rindex = = minindex(1)) = [];
   if((length(rindex)-(testlength+vallength)) = = 2)
      rindex(testlength+vallength+2) = [];
      rindex(testlength+vallength+1) = [];
   end
   if((length(rindex)-(testlength+vallength)) = = 1)
      rindex(testlength+vallength+1) = [];
   end
   modifiedrindex(n,:) = rindex;
   valindex = rindex(1,1:vallength);
   testindex = rindex(1,(vallength+1):(testlength+vallength));
   ss = 1;
   for p = 1:1:l
      if(p ∼ = rindex)
         trainindex(1,ss) = p;
         ss = ss+1;
      end
   end
   trainvalue = Target(trainindex);
   valvalue = Target(valindex);
   testvalue = Target(testindex);
   datamean(n,1) = mean(trainvalue);
   datamean(n,2) = mean(valvalue);
   datamean(n,3) = mean(testvalue);
   Alltrainindex(n,:) = trainindex;
   Allvalindex(n,:) = valindex;
   Alltestindex(n,:) = testindex;
end
disp(datamean);
Bestmeanindex = input("Enter the best mean index ");
Bestdatamean90 = datamean(Bestmeanindex,:);
Besttrainindex90 = Alltrainindex(Bestmeanindex,:);
Bestvalindex90 = Allvalindex(Bestmeanindex,:);
Besttestindex90 = Alltestindex(Bestmeanindex,:);
Train_Target = Target(Besttrainindex90);
Train_Input = Input(:,Besttrainindex90);
Val_Target = Target(Bestvalindex90);
Val_Input = Input(:,Bestvalindex90);
Test_Target = Target(Besttestindex90);
Test_Input = Input(:,Besttestindex90);
```

2. Code for MLP model:

```
%= = = = = = = = = = = = = = = = = = = = = = = = = =
%                    MLP (FeedForward Network)
%= = = = = = = = = = = = = = = = = = = = = = = = = =


epochs = 1000;
val_max = 100;
node_number = 4;
nodes = [7,8,9,10];     % No. of nodes in each hidden layers
number_time = 10;            % No. of times to run for each configuration


All_Train_Regression = zeros(node_number,number_time);
All_Train_MSE = zeros(node_number,number_time);

All_Val_Regression = zeros(node_number,number_time);
All_Val_MSE = zeros(node_number,number_time);

All_Test_Regression = zeros(node_number,number_time);
All_Test_MSE = zeros(node_number,number_time);

All_Overall_MSE = zeros(node_number,number_time);
All_Overall_Regression = zeros(node_number,number_time);


for i = 1:1:node_number               % Number of node loop
   for j = 1:1:number_time
      %Training
      x = nodes(i);
      net = feedforwardnet([x,x]);            % Provide the nodes in each
HL
      net.trainParam.epochs = epochs;
      net.trainParam.max_fail = val_max;
      net.divideFcn = ‘divideind’;
      net.divideParam.trainInd = Besttrainindex90;
      net.divideParam.valInd = Bestvalindex90;
      net.divideParam.testInd = Besttestindex90;
      net = train(net,Input,Target);
      % Train Output
      Train_Output = net(Train_Input);
      Train_error = gsubtract(Train_Target,Train_Output);
      Train_Relativeerror = Train_error./Train_Target;
      Train_MAE = mae(Train_error);
      Train_MSE = mse(net,Train_Target,Train_Output);
      Train_Regression = regression(Train_Target,Train_Output);
      All_Train_Regression(i,j) = Train_Regression;
      All_Train_MSE(i,j) = Train_MSE;
      % Validation Output
      Val_Output = net(Val_Input);
      Val_error = gsubtract(Val_Target,Val_Output);
```

(*continued*)

```
      Val_Relativeerror = Val_error./Val_Target;
      Val_MAE = mae(Val_error);
      Val_MSE = mse(net,Val_Target,Val_Output);
      Val_Regression = regression(Val_Target,Val_Output);
      All_Val_Regression(i,j) = Val_Regression;
      All_Val_MSE(i,j) = Val_MSE;
      % Testing Output
      Test_Output = net(Test_Input);
      Test_error = gsubtract(Test_Target,Test_Output);
      Test_Relativeerror = Test_error./Test_Target;
      Test_MAE = mae(Test_error);
      Test_MSE = mse(net,Test_Target,Test_Output);
      Test_Regression = regression(Test_Target,Test_Output);
      All_Test_Regression(i,j) = Test_Regression;
      All_Test_MSE(i,j) = Test_MSE;
      % Overall Output
      Output = net(Input);
      Overall_error = gsubtract(Target,Output);
      Overall_Relativeerror = Overall_error./Target;
      Overall_MAE = mae(Overall_error);
      Overall_MSE = mse(net,Target,Output);
      Overall_Regression = regression(Target,Output);
      All_Overall_Regression(i,j) = Overall_Regression;
      All_Overall_MSE(i,j) = Overall_MSE;
      %Saving Workspace
      eval([‘save MLP_model_node_’,int2str(i),’_times_’,int2str(j)]);
   end
end
```

3. Matlab Code for Data Division in ENN Model:

```
l = length(Target);
Test_length = int64(l∗0.6);
Train_length = l-(Test_length);
Train_value = zeros(1,l);
Test_value = zeros(1,Test_length);
Train_index = zeros(1,Train_length);
Test_index = zeros(1,Test_length);
datamean = zeros(20,2);
All_Train_Index = zeros(20,Train_length);
All_Test_Index = zeros(20,Test_length);
maxnumber = max(Target);
minnumber = min(Target);
maxindex = find(Target = = maxnumber);
minindex = find(Target = = minnumber);
original_rand_index = zeros(20,(Test_length+2));
modified_rand_index = zeros(20,(Test_length));
for n = 1:1:20
  rand_index = randperm(l,(Test_length+2));
  original_rand_index(n,:) = rand_index;
  rand_index(rand_index = = maxindex(1)) = [];
  rand_index(rand_index = = minindex(1)) = [];
  if((length(rand_index)-(Test_length)) = = 2)
    rand_index(Test_length+2) = [];
    rand_index(Test_length+1) = [];
  end
  if((length(rand_index)-(Test_length)) = = 1)
    rand_index(Test_length+1) = [];
  end
  modified_rand_index(n,:) = rand_index;
  Test_index = rand_index;
  ss = 1;
  for p = 1:1:l
    if(p ∼ = rand_index)
      Train_index(1,ss) = p;
      ss = ss+1;
    end
  end
  Train_value = Target(Train_index);
  Test_value = Target(Test_index);
  datamean(n,1) = mean(Train_value);
  datamean(n,2) = mean(Test_value);
  All_Train_Index(n,:) = Train_index;
  All_Test_Index(n,:) = Test_index;
end
disp(datamean);
Bestmeanindex = input("Enter the best mean index ");
Bestdatamean90 = datamean(Bestmeanindex,:);
Besttrainindex90 = All_Train_Index(Bestmeanindex,:);
Besttestindex90 = All_Test_Index(Bestmeanindex,:);
Train_Target = Target(Besttrainindex90);
Train_Input = Input(:,Besttrainindex90);
Test_Target = Target(Besttestindex90);
Test_Input = Input(:,Besttestindex90);
```

4. Matlab Code for ENN Model:

```
% = = = = = = = = = = = = = = = = = = = = = = = = = =
%                    Elman Neural Network
% = = = = = = = = = = = = = = = = = = = = = = = = = =
epochs = 1000;
goal_no = 3;            % No. of goals you want to try
goal = [0.05,0.04,0.03]; %list of goals to try
number_time = 10;         % No. of times to run for each
configuration


All_Train_Regression = zeros(goal_no,number_time);
All_Train_MSE = zeros(goal_no,number_time);

All_Test_Regression = zeros(goal_no,number_time);
All_Test_MSE = zeros(goal_no,number_time);

All_Overall_MSE = zeros(goal_no,number_time);
All_Overall_Regression = zeros(goal_no,number_time);

All_Overall_Max_RE = zeros(goal_no,number_time);
All_Overall_Min_RE = zeros(goal_no,number_time);

for i = 1:1:goal_no        % Goal loop
  for j = 1:1:number_time

    %Training
    net = elmannet(1:2,[9,9]);     % Provide the nodes in each HL
    net.trainParam.epochs = epochs;
    net.trainParam.goal = goal(i);
    net = train(net,Train_Input,Train_Target);

    % Train Output
    Train_Output = net(Train_Input);
    Train_error = gsubtract(Train_Target,Train_Output);
    Train_Relativeerror = Train_error./Train_Target;
    Train_MAE = mae(Train_error);
    Train_MSE = mse(net,Train_Target,Train_Output);
    Train_Regression = regression(Train_Target,Train_Output);
    All_Train_Regression(i,j) = Train_Regression;
    All_Train_MSE(i,j) = Train_MSE;

    % Testing Output
    Test_Output = net(Test_Input);
    Test_error = gsubtract(Test_Target,Test_Output);
    Test_Relativeerror = Test_error./Test_Target;
    Test_MAE = mae(Test_error);
    Test_MSE = mse(net,Test_Target,Test_Output);
    Test_Regression = regression(Test_Target,Test_Output);
    All_Test_Regression(i,j) = Test_Regression;
    All_Test_MSE(i,j) = Test_MSE;

    % Overall Output
    Output = net(Input);
    Overall_error = gsubtract(Target,Output);
    Overall_Relativeerror = Overall_error./Target;
    Max_RE = max(Overall_Relativeerror);
    Min_RE = min(Overall_Relativeerror);
    Overall_MAE = mae(Overall_error);
    Overall_MSE = mse(net,Target,Output);
    Overall_Regression = regression(Target,Output);
    All_Overall_Regression(i,j) = Overall_Regression;
    All_Overall_MSE(i,j) = Overall_MSE;
    All_Overall_Max_RE(i,j) = Max_RE;
    All_Overall_Min_RE(i,j) = Min_RE;

    %Saving Workspace
    eval([’save ENN_model_node_’,int2str(i),’_times_’,int2str(j)]);
  end
end
```