Performance Analysis of Artificial Neural Networks Training Algorithms and Transfer Functions for Medium-Term Water Consumption Forecasting

Lemuel Clark P. Velasco, Angelie Rose B. Granados, Jilly Mae A. Ortega, Kyla Veronica D. Pagtalunan
Mindanao State University-Iligan Institute of Technology
Iligan City, The Philippines

***Abstract*—Artificial Neural Network (ANN) is a widely used machine learning pattern recognition technique in predicting water resources based on historical data. ANN has the ability to forecast close to accurate prediction given the appropriate training algorithm and transfer function along with the model's learning rate and momentum. In this study, using the Neuroph Studio platform, six models having different combination of training algorithms, namely, Backpropagation, Backpropagation with Momentum and Resilient Propagation and transfer functions, namely, Sigmoid and Gaussian were compared. After determining the ANN model's input, hidden and output neurons from its respective layers, this study compared data normalization techniques and showed that Min-Max normalization yielded better results in terms of Mean Square Error (MSE) compared to Max normalization. Out of the six models tested, Model 1 which was composed of Backpropagation training algorithm and Sigmoid transfer function yielded the lowest MSE. Moreover, learning rate and momentum value for the models of 0.2 and 0.9 respectively resulted to very minimal error in terms of MSE. The results obtained in this research clearly suggest that ANN can be a viable forecasting technique for medium-term water consumption forecasting.**

Keywords—Artificial neural network; backpropagation; water consumption forecasting

I. Introduction

Artificial Neural Networks (ANN) is a mathematical model inspired from how brain neurons learn and perform pattern recognition. ANN have been used as a technique for predicting and forecasting in various areas including finance, power generation, medicine, water resources and environmental science [1]-[3]. ANNs are composed of one or more processing units called artificial neurons or perceptrons. Basically, ANNs consist of three layers, namely, the input layer, the hidden layer and the output layer. The input layer represents the model inputs and the output layer represents the model outputs. The hidden layer consists of nodes that during optimization attempt to functionally map the model inputs to the model outputs. The basic idea of an ANN is that the network learns from the input data and the associated output data with the help of training algorithms and transfer functions [3]-[6]. Back propagation training algorithm is a supervised learning method based on the gradient descent of the quadratic error function and is considered as the universal function approximator [4], [5]. During the learning process, the gradient descent method is used to minimize the total error or mean error of the output computed by the network [1], [6]. The activations of the input nodes are multiplied by the weighted connections and are passed through a transfer function at each node in the first hidden layer. The activations from the first hidden layer are then passed to the neurons in the next layer, and this process is repeated until the output activations are obtained from the output layer. The output activation values and the target pattern are compared and the error signal is calculated based on the difference between target and calculated pattern. This error signal is then propagated backwards to adjust network weights so that network will generate correct output for the presented input pattern. The training patterns are presented repeatedly until the error reaches an acceptable value or other convergence criteria are satisfied [5]. As this technique involves performing computation backwards it is named as backpropagation.

ANN modelling approaches have been embraced enthusiastically by practitioners in water resources, as they are perceived to overcome some of the difficulties associated with traditional statistical approaches [1], [4], [7]. With the changing landscape and climate brought about by weather phenomena and unprecedented human activities, water as a very important environmental resource should be managed scientifically with the use of tools and techniques that will optimize usage management and conservation. Decision-makers can utilize machine learning platforms and models in analyzing huge volumes of data related to water management that can in turn be used to develop applications that will generate valuable inputs for short, medium and long term planning. These applications involved in water consumption forecasting use historical data to predict medium-term consumption helpful for decision makers in making critical decisions involving supply planning, reservoir or urban infrastructure changes, staging treatment and distribution system improvements [3], [6,]. Predictive applications involving forecasting water consumption are as important as descriptive applications since these applications give foresight on trends and patterns using machine learning models [8]. With the use of ANN an accurate and reliable prediction of future water consumption can help decision-makers to take necessary measures according to the possible crises and limitations.

Neuroph, a lightweight Java neural network framework for developing common neural network architectures implements multilayer perceptron having various backpropagation training algorithms and transfer functions. A major challenge in the implementation of ANN in water consumption forecasting is the choice of the appropriate ANN model design involving training algorithms and transfer functions that can yield the smallest error from the actual water consumption. Consequential to this challenge is water consumption data preparation through data normalization to ensure the avoidance of slow neural network training [7]-[9]. Data normalization is a process of final data preparation for network training so that the normalized data are shaped to meet the network input layer requirements. With the proper analysis of the water consumption data and the formulation of the appropriate ANN model, an ANN technique can be a viable solution to generate close to perfect prediction values for the prediction of water consumption. The aim of this research then is to determine an appropriate data normalization and ANN model design for medium term water consumption forecasting. This study aims to contribute to the recent technology researches in machine learning by evaluating performance of ANN models that could help water utility companies in their decision-making, proper planning and management of water resources.

II. Methodology

A. Water Consumption Data Preparation

Eighteen-year water consumption data from a city's Waterworks System in the Philippines based from the monthly bills of January 1998 to December 2015 was used in this study. As shown in Table I, a total of 1,080 rows of data with eleven (11) corresponding columns containing the month, the billed accounts and the respective water consumed in cubic meter from five water consumption categories namely domestic, commercial, industrial, bulk and whole water consumed.

TABLE I. WATER CONSUMPTION DATA

Month	Domestic Billed Accounts	Domestic Water Onsumed	Commercial Billed Accounts	Commercial Water Onsumed	Industrial Billed Ccounts	Industrial Water Consumed	Bulk Billed Account	Bulk Water Onsumed	Whole Billed Ccounts	Whole Water Onsumed
1	14657	830613	1138	152931	20	137422	2	4173	15817	1125139
2	14487	833589	1173	155840	19	124218	2	3249	15681	1116896
3	14153	761834	1107	131138	19	120886	2	3652	15281	1017510
4	13337	816552	1111	141278	19	121263	2	3479	14469	1082572
5	14430	789987	1137	125028	19	114681	2	2803	15588	1032499

Data normalization is a means of fitting the data within unity so that all data values will take on a value range of 0 to 1 [9]-[11]. It is one of the most significant pre-processing strategy which has a significant impact on the accuracy and performance of any model such that the sole purpose of data normalization is to guarantee the quality of the data before it is fed to a model [11]. Furthermore, the normalization process for the raw inputs has great effect on preparing the data to be suitable for the training. Due to the different units of the data, it is important to normalize the input and output data in the model development. It is required to normalize all the datasets between 0 and 1 to fit the data within unity [6], [12]. Two normalization methods namely Min-Max normalization and Max normalization were used and tested in this study. For the Min-Max normalization, a function was used to normalize the water consumption values using equation (1):

$$\bar{x} = \frac{x - x_{MIN}}{x_{MAX} - x_{MIN}} \tag{1}$$

where $\underline{x}$ is the normalized data point, x is the original data point, x_{MIN} and x_{MAX} are the minimum and maximum of the dataset, respectively. On the other hand, Max normalization was used to normalized the water consumption values using equation (2):

$$L_s = \frac{L}{L_{max}} \tag{2}$$

where L is the actual load, L_{max} is the maximum load during the day and L_S is the normalized load. Each of these normalization methods were applied into different models. Each of these normalization methods was applied into different formulated models having different training algorithms and transfer functions with comparison conducted using Mean Square Error (MSE) for every neural network model testing.

After normalizing the dataset, the data was then partitioned into training and testing sets. Approximately 95% of the dataset was assigned as the training set containing 1026 records of the water consumption data from January 1998 to December 2014 while 5% of the dataset was assigned as the testing set containing 54 records of the water consumption data from January 2015 to December 2015.

B. ANN Model Design Evaluation

The type of neural network used in this study was multilayer perceptron neural network with three layers: an input layer, one hidden layer and an output layer. The number of variables used as input parameters were then determined. There is no general rule for selecting the number of neurons in a hidden layer. It only depends on the complexity of the system being modeled [13]. The most popular approach in finding the optimal number of neurons in hidden layer is by trial and error [4], [6], [13]. Moreover, according to research, researchers conducted a study evaluating the number of neurons in the hidden layer but still none was accurate [14]. Thus, trial and error approach was used in this study to determine the optimum neurons in hidden layer of the network. In order to determine the optimum number of hidden neurons, several formulae on how to ascertain the optimum neurons in the second layer was also considered in this study. Hidden neuron formulae were gathered and determined from the academic journals as shown in Table II. Also, there are five neurons in the output layer representing the next month water consumption in each category.

There are several types of transfer functions, however this study only used Gaussian and Sigmoid function since these transfer functions can produce positive values between 0 and 1 which corresponds to the training and testing data sets that were normalized in a scale of 0 and 1. Training then commenced with an important consideration that the size of the steps taken in weight space during training is a function of a number of internal network parameters which includes the learning rate and momentum [1], [12]. Factors such as learning rate and momentum affect the performance in the learning process of the network. Learning rate is a parameter that determines the size of the weights adjustment each time the weights are changed during training while the momentum is a factor used to speed up convergence and maintain generalization performance of the network [15], [16]. The choice of appropriate parameters has a major impact on the performance of the backpropagation algorithm [1], [15], [16]. A good selection of these parameters could speed up and improve in great measure the learning process to reach the goal, although a universal answer does not exist for such configuration [16]. Furthermore, authors believe that choosing the learning rate can be done by trial and error [1], [3]. In this case, the learning rate and momentum value used in this study was done by trial and error. Both learning rate and momentum parameter were usually in the range between 0.1 and 0.9 [12]. Training attempts were conducted using all combinations of learning rate and momentum. This was done to select the learning rate parameter to be used in training the models. After each training run, the training results was then evaluated and compared with the results achieved in the previous runs to select the best run.

TABLE II. FORMULAE FOR HIDDEN NEURONS

Authors	**Hidden Neuron Formula**	**No. of Hidden Neurons**
Lipae, J.L. and Deligero, E.P. (2012)	$\sqrt{\boldsymbol{N_i N_o}}$ Where $\boldsymbol{N_i}$ is the input neuron and $\boldsymbol{N_o}$ is the output neuron	7
Sheela, K.J. and Deepa, S.N. (2013)	N-1 where N is the input-target relation	10
	$N_h = n + n_o - 1/2$ where n is the number of inputs and n_o is the number of outputs	16
	$N_h = (N_{in} + \sqrt{N_p}) / L$ where L is the number of hidden layer, N_{in} is the number of input neuron and N_p is the number of input sample	26
Param, Sowjanya (2015)	2/3 the size of the input layer plus the size of the output layer	12

This research evaluated the performance of different ANN models based on the type of training algorithms and transfer functions of the neural network. Training algorithms such as Backpropagation, Backpropagation with Momentum and Resilient Propagation were used in this study. Backpropagation is one of the most widely used training algorithms for training feedforward neural networks. This type of network configuration is the most commonly in use due to its ease of training [10]. The Backpropagation algorithm modifies network weights to minimize the mean squared error between the desired and the actual outputs of the network. Furthermore, Backpropagation uses supervised learning in which the network is trained using data for which the input as well as the desired outputs is known, one of the most well-known variants is the Backpropagation with Momentum [15],[17]. Momentum was added that for faster training. With this change, the weight change continues in the direction it was heading. This weight change, in the absence of error, would be a constant multiple of the previous weight change. The momentum term is an attempt to try to keep the weight change process moving, and thereby not gets stuck in its local minima which make the convergence faster and the training more stable in some cases [17]. On the other hand, Resilient Propagation (Rprop) was one of the best performing first order learning algorithms for multilayer neural networks[10],[18],[19]. The basic principle of Rprop is to eliminate the harmful influence of the size of the partial derivative on the weight step in which only the sign of the derivative is considered to indicate the direction of the weight update.

These training algorithms were paired with Sigmoid and Gaussian transfer functions. There are several types of transfer functions, this study however only used Sigmoid and Gaussian function since these transfer functions can produce positive values between 0 and 1 which corresponds to the training and testing data sets that were normalized in a scale of 0 and 1. Each combination of training algorithm and transfer function represent one model. Shown in Table III are the formulated models.

TABLE III. FORMULATED ANN MODELS

Model	**Training Algorithm**	**Transfer Function**
Model 1	Backpropagation	Sigmoid
Model 2	Backpropagation	Gaussian
Model 3	Backpropagation with Momentum	Sigmoid
Model 4	Backpropagation with Momentum	Gaussian
Model 5	Resilient Propagation	Sigmoid
Model 6	Resilient Propagation	Gaussian

During evaluation, test runs were conducted in each model by feeding the training dataset into the network and trained using the Backpropagation algorithm, Backpropagation with Momentum and Resilient Propagation. Backpropagation algorithm was used to modify the network weights in order to decrease the value of the error function on subsequent tests of the inputs. The process of adjusting weights was continued until the error is less than some desired limit after which the network is considered trained. The training process of the ANN models will stop when the network output error has reached its minimal value [5], [12]. Error measure was then computed to assess the neural network's accuracy since accuracy is the most important criteria in evaluating forecasting models and in choosing between competing models. In each test run, error measure was calculated to determine and be compared with the predictive capability of the models. In order to evaluate the ANN models, Mean Square Error (MSE) was calculated as the error measure. MSE was used as it penalizes extreme errors obtaining partial derivative with respect to the weights and that it lies close to the heart of the normal distribution [1], [18]. Among the designed models, the model that produces the smallest MSE was chosen as the neural network model.

III. Results and Discussion

A. Water Consumption Data Preparation Results

A comparison of two normalization techniques namely the Min-Max normalization and Max normalization was performed for the purpose of determining which of the two techniques yields a more accurate model. Each of the normalization techniques was tested to see whether the normalization technique has a significant effect on the neural network accuracy based on MSE values. The comparison was made by training the neural network using different combination of transfer functions namely Sigmoid and Gaussian and training algorithms such as Backpropagation, Backpropagation with Momentum and Resilient Propagation with 7 hidden neurons. The training dataset was used and then fed into the network. The learning parameters like learning rate and momentum of 0.2 and 0.7 respectively, was used during training. The MSE for each network that used the Min-Max normalization technique was calculated and presented in Table IV, respectively showing the MSE results using different training algorithms and transfer functions.

TABLE IV. Min-Max Normalization MSE Results

Training Algorithm	Mean Square Error	
	Sigmoid	Gaussian
Backpropagation	0.003874424	0.003885264
Backpropagation with momentum	0.003914189	0.003913525
Resilient propagation	0.005360615	0.006531334

In using the Min-Max normalization in normalizing the water consumption data, the lowest mean square error was achieved using the combination of Backpropagation training algorithm and Sigmoid transfer function while the highest mean square error was achieved using the Resilient Propagation as the training algorithm and Gaussian transfer function. The researchers observed that the MSE values of Sigmoid and Gaussian activation are below 0.0039 when Backpropagation and Backpropagation with Momentum were used as the training algorithm. As shown in Fig. 1, even though Sigmoid and Gaussian activation have close MSE results, Sigmoid with Backpropagation as the training algorithm has lower MSE values than Gaussian.

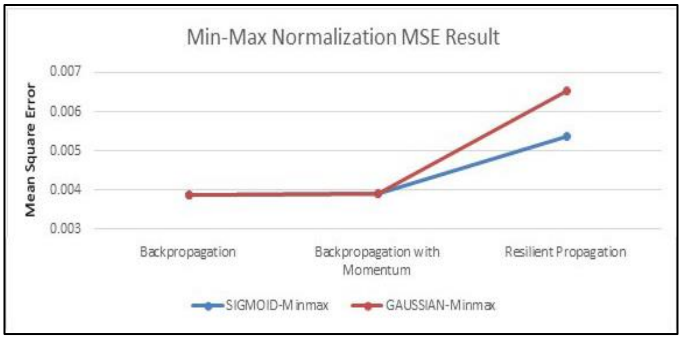

Fig. 1. Min-max normalization MSE result graph.

The MSE results using different training algorithms and transfer functions for each network that used the Max normalization technique was calculated and presented in Table V.

TABLE V. Max Normalization MSE Results

Training Algorithm	Mean Square Error	
	Sigmoid	Gaussian
Backpropagation	0.004959902	0.005191838
Backpropagation with momentum	0.005250859	0.005312595
Resilient propagation	0.003896138	0.003882556

In using the Max normalization in normalizing the water consumption data, the lowest mean square error was achieved using the Resilient Propagation training algorithm and Gaussian transfer function while the highest mean square error was achieved using the Backpropagation with Momentum algorithm and Gaussian transfer function. As shown in Fig. 2, the researchers observed that the MSE of Sigmoid and Gaussian activation showed lowest values when Resilient Propagation training algorithm was used. Even though the Sigmoid and Gaussian activation have close MSE results, Gaussian activation has lower MSE values than Sigmoid.

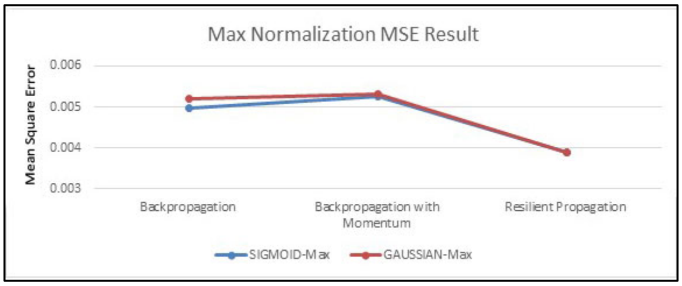

Fig. 2. Max normalization MSE result graph.

In comparing the two normalization techniques, the Min-Max normalization technique yielded better MSE values than the Max normalization technique. As shown in Fig. 3, among the four normalization tests, the data that used Min-Max normalization and Backpropagation as the training algorithm with Sigmoid transfer function produced the lowest MSE while the data that used the Max normalization and Backpropagation with Momentum as the training algorithm with Gaussian transfer function produced the highest error.

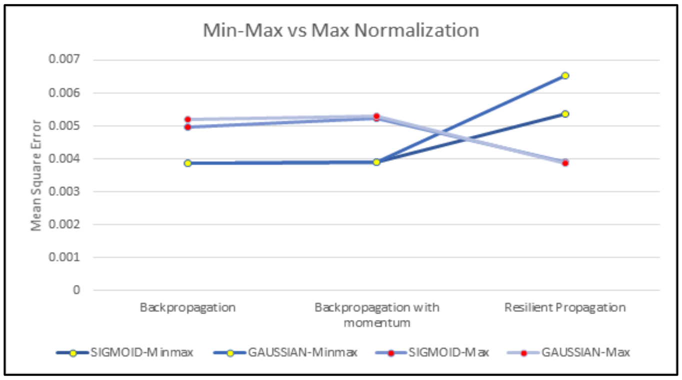

Fig. 3. Min-max normalization MSE result graph.

The researchers observed that error curves are not in almost linear pattern. In Resilient Propagation, Sigmoid and Gaussian using Min-Max and Max normalization technique have opposite results. The values of Sigmoid and Gaussian using Min-Max normalization are higher than the values of Sigmoid and Gaussian using Max normalization. Even though the MSE values are within 0 to the 0.007 range, the values have a significant difference between one another. Thus, among Min-Max and Max normalization techniques, the type of normalization method used has a huge and significant effect on the performance of the network in terms of accuracy. Furthermore, since Min-Max normalization has the lowest MSE, it was used as the normalization technique for the training dataset. Studies on different types of normalization techniques in data mining as a preprocessing engine conducted also concluded that Min-Max normalization is the best design for training data set because it has a higher percentage of accuracy compared to Max normalization, Z-score normalization and decimal point normalization [10], [20]. Moreover, the Min-Max normalization technique was also used in water demand prediction using artificial neural networks and support vector regression to avoid having more weight being assigned to features with larger values [21]. Studies conducted on predictive analytics also support the results of this study that Min-Max normalization is better than Max normalization [9]-[11], [20], [21].

B. ANN Model Design Evaluation Results

Designing the architecture of an ANN model includes the identification of the number of neurons for input, hidden and output layers, as well as the performance analysis of the training algorithms and the transfer functions. As shown in Fig. 4, there were 11 input neurons which represent the month, billed accounts in each category and the water consumed in each category while the number of neurons in the output layer was 5 representing the next month water consumption in each category. A neural network with one hidden layer has the tendency to perform very well [2], [5], [7]. Thus, the researchers used only 1 hidden layer.

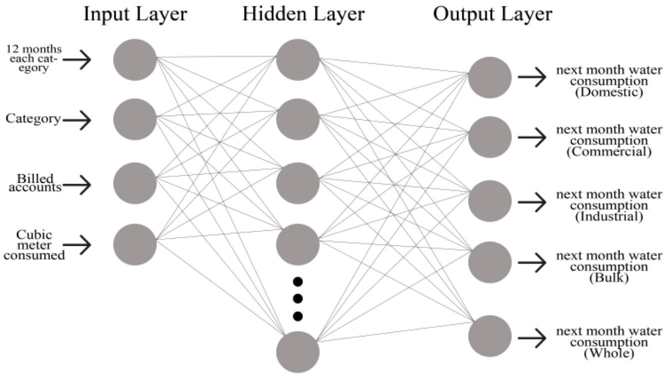
Input Layer
Hidden Layer
Output Layer
12 months each category
Category
Billed accounts
Cubic meter consumed
next month water consumption (Domestic)
next month water consumption (Commercial)
next month water consumption (Industrial)
next month water consumption (Bulk)
next month water consumption (Whole)

Fig. 4. The proposed ANN architecture.

Determining the number of hidden neurons does not have a standardized or theoretical approach to calculate the number of neuron in the hidden layer [14]. In order to select the appropriate number of hidden neurons to be used in this study, the researchers conducted series of tests with results shown in Table VI.

TABLE VI. HIDDEN NEURONS TEST RESULTS

idden Neuron Equation	Hidden Neurons	Total MSE (M^3)
$\sqrt{\boldsymbol{N_i N_o}}$ where $\boldsymbol{N_i}$ is the input neuron and $\boldsymbol{N_o}$ is he output neuron	7	0.00394277592437603
N-1 where N is the input-target relation	10	0.0039617700063948325
$N_h = n + n_o - ½$ where n is the number of inputs and n_o s the number of outputs	16	0.0040196430018757 27
$N_h = (N_{in} + \sqrt{N_p}) / L$ where L is the number of hidden layer, V_{in} is the number of input neuron and V_p is the number of input sample	26	0.0042473199122742 59
2/3 the size of the input layer plus the ize of the output layer	12	0.0044927624199001 46

The hidden neuron equations presented in the first column of the table was calculated to determine the hidden neuron. Each number of hidden neuron was tested and yielded results in MSE. Among the hidden neuron choices, the first equation with 7 hidden neurons yielded the lowest MSE while the last equation with 12 hidden neurons produced the highest MSE. As a result, 7 hidden neurons were selected because it yielded the least MSE that is 0.00394.

Neuroph, a Java open source framework designed to develop artificial neural network was used in training the ANN models. It contains an implementation for most of the mainstream ANNs and learning algorithms, such as multilayer perceptron network and backpropagation learning algorithm. Training dataset was fed into each model during training. The models used the same set of learning parameters to maintain the credibility of the evaluation results. Learning parameters such as maximum error, learning rate and momentum were set. Maximum error was set as the stopping criteria during training. If the error on the training or selection test drops below the given target values, the network is considered to have trained sufficiently well, and training is terminated. The error never drops to zero or below, so the default value of zero is equivalent to not having a target error [22]. The learning rate parameter determines the size of the weights adjustment each time the weights was updated during training. A learning rate of 0.0 does not learn [16]. The momentum parameter was a factor used to speed network training for escaping the local minima to avoid error fluctuation [13]. Both learning rate and momentum parameter were usually in the range between 0.1 and 0.9 and choosing the learning rate can be done by trial and error [3], [12]. Fig. 5 shows the graphical representation of the overall results for the test of the learning rates. Different combination of learning rate and momentum yielded varying results in terms of MSE. The researchers observed that the higher the learning rate, the bigger the MSE was produced. Generally, a small learning rate can ensure the reduction of the error function but may slow the convergent process, while a large learning rate can speed up the learning process but may cause divergence [6], [16]. According to an author, if the selected learning rate is too large, then the local minimum may be overstepped constantly, resulting in oscillations and slow convergence to the lower error state [23]. As a result, learning rate of 0.9 yielded the highest MSE among the others. Likewise, if the learning rate is too low, the number of iterations required may be too large, resulting in slow performance.

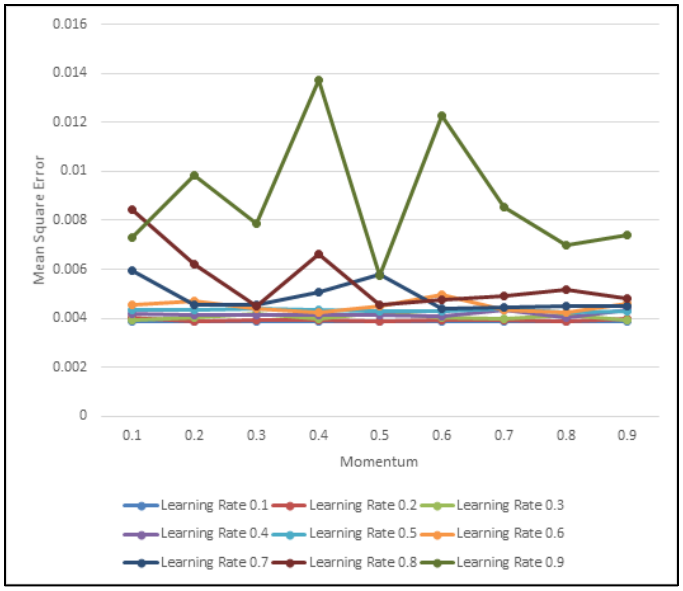

Fig. 5. Learning parameters test results.

Contrary to results shown in learning rate 0.1 and 0.2 combined with momentum 0.1 to 0.9, the number of iteration was not too large and yielded the least MSE. Thus, this study used the combination of momentum 0.1 to 0.9 and learning rate 0.1 and 0.2 in training the models because it performed well in terms of MSE. Each combination of learning rate and momentum corresponds to 1 training attempt. Generally, there is a total of 108 training attempts where each model has 18 training attempts. Table VII shows the training attempts with the smallest MSE obtained in each model.

TABLE VII. ANN MODEL TRAINING RESULTS

Model	Maximum Error	Learning Rate	Momentum	MSE
Model 1	0.01	0.2	0.9	0.003870079156963867
Model 2	0.01	0.1	0.3	0.00455617301987756
Model 3	0.01	0.1	0.4	0.003870627416201587
Model 4	0.01	0.1	0.2	0.004689457453932001
Model 5	0.01	0.1	0.9	0.00398609747754698
Model 6	0.01	0.2	0.2	0.003990659899236666

As observed, Model 1 has better precision on prediction which was trained using Backpropagation algorithm and Sigmoid transfer function with 0.2 and 0.9 values of learning rate and momentum respectively. The researchers found out that the values of 0.2 and 0.9 for the learning rate and momentum yielded the fastest learning convergence to the minimal number of errors during training attempts. Although the selection of the learning rate and momentum is an essential task, other factors like training algorithm and activation function are more vital as the results shows that different combination of this yielded differential result. The researchers observed that Sigmoid transfer function paired to Backpropagation and Backpropagation with Momentum algorithms yielded the least MSE compared to Gaussian activation function. In other words, Sigmoid transfer reached a very good overall approximation [10], [24]. Moreover, when Sigmoid and Gaussian were trained using Resilient Propagation algorithm, it yielded results that were close to each other as shown in Fig. 6. Among the training algorithms, Backpropagation outperformed the Backpropagation with Momentum and Resilient Propagation. Backpropagation training algorithm yielded the least MSE value while Resilient Propagation yielded the highest MSE value. The result shows that the MSE values obtained by the Backpropagation and Backpropagation with Momentum were very close to each other but Backpropagation training algorithm has smaller values than the Backpropagation with Momentum algorithm. This implies that Backpropagation performed better than Backpropagation with Momentum. This was supported by studies who also used Backpropagation algorithm as the best model for water demand prediction [8], [25].

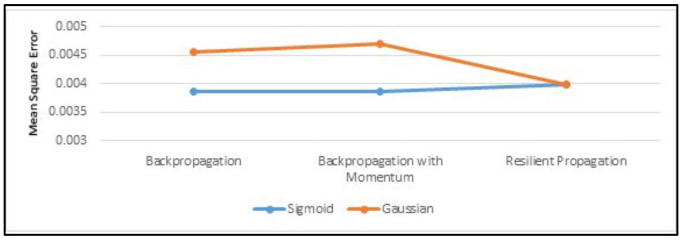

Fig. 6. Comparison of training algorithms with transfer functions.

IV. Conclusion and Recommendations

This study attempted to conduct a performance analysis of different ANN training algorithms, transfer functions, learning rates and momentum to discover a suitable ANN model for month-ahead water consumption prediction. Two normalization techniques namely Min-Max normalization and Max normalization were compared in the water consumption data preparation phase with results showing that Min-Max scaling yielded better results in terms of MSE values. Six models were then formulated with different combination of training algorithms and transfer functions. The type of neural network used in this study was a multilayer perceptron having three layers in each model. Neuroph Studio, a Java based Neural Network IDE was then used to simulate the designed models. Learning parameters such as maximum error, learning rate and momentum were set, showing that a learning rate and momentum value of 0.2 and 0.9 respectively, performed as the better combination of learning parameter with a minimal error in terms of MSE. After the models were compared and tested, among the 6 models, Model 1 which was composed of Sigmoid transfer function and Backpropagation training algorithm has the best precision in forecasting and yielding the smallest value of MSE equal to 0.003870079156963867. The results of this study show that Sigmoid activation function being paired with different training algorithm yielded better results compared to Gaussian. Moreover, Backpropagation algorithm performs better when compared to other training algorithms.

It is recommended that the use of additional input factors could yield better results in training the network model. The study could potentially be improved if other variables that affect water consumption are to be examined. Other ANN frameworks could also be used to expand performance analysis conducted in this study. One or more ANN frameworks can be compared with the performance results of this study and contrast if other frameworks have better or the same performance with that of Neuroph. Other normalization techniques, combination of learning parameters, training algorithms and activation functions could also be explored as enhancements for future work. With the results of the performance analysis from the different training algorithms and activation functions being shown in this research, future directions can be geared towards the use of the best performing model for a chosen validation set and an evaluation of how close is its prediction to the corresponding actual water consumption.

References

[1] H. R. Maier, and G. C. Dandy, “Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications,” Environmental Modelling & Software 15, Elsevier Science Ltd., pp. 101-124, 2000.

[2] L. C. P. Velasco, N. R. Estoperez, R. J. R. Jayson, C. J. T. Sabijon, and V. C. Sayles, "Day-ahead Base, Intermediate, and Peak Load Forecasting using K-Means and Artificial Neural Networks," International Journal of Advanced Computer Science and Applications, Volume 9 Issue 2, The Science and Information Organization Limited, DOI : 10.14569/IJACSA.2018.090210, pp. 62-67, 2018.

[3] J. R. Mohammed and H. M. Ibrahim, “Multilayer Perceptron - Multiactivation Function Artificial Neural Network Model for Municipal Water Demand Forecasting,” ARPN Journal of Engineering and Applied Sciences, Vol. 8, No. 12, pp. 1019-1038, 2013.

[4] M. A. Al-Zahrani, and A. Abo-Monasar, “Urban residential water demand prediction based on artificial neural networks and time series models,” Water Resources Management, 29(10), pp. 3651-3662, 2015.

[5] M. Djeddou and B. Achour, “The Use of a Neural Network Technique for the Prediction of Sludge Volume Index in Municipal Wastewater Treatment Plant,” Larhyss Journal, pp. 351-370, 2015.

[6] J. Liu, H. Savenije, and J. Xu, “Forecast of water demand in Weinan City in China using WDF-ANN model,” Physics and Chemistry of the Earth, 28, pp. 2019-224, 2003.

[7] M. Firat, M. E. Turan, and M. A. Yurdusev, “Comparative analysis of neural network techniques for predicting water consumption time series,” Journal of Hydrology, 384, pp. 46-51, 2010.

[8] L. C. P. Velasco, A. R. B. Granados, J. M. A. Ortega and K. V. D. Pagtalunan, "Medium-Term Water Consumption Forecasting using Artificial Neural Networks for a Water Utility," Proceedings of the 17th Conference of the Science Council of Asia, National Research Council of the Philippines, 2017

[9] T. Jayalakshmi and A. Santhakumaran, A. “Statistical Normalization and Back Propagation for Classification,” International Journal of Computer Theory and Engineering, pp. 89-93, 2011.

[10] L. C. P. Velasco, J. A. A. Dagaang , C. R. Villezas, P. N. C. Palahang, "Performance Analysis of Different Combination of Training Algorithms and Activation Functions in Predicting the Next Day Electric Load,", Proceedings of the 16th Philippine Computing Science Congress, 2016

[11] S. C. Nayak, B. B. Misra, and H. S. Behera, "Impact of Data Normalization on Stock Index Forecasting," International Journal of Computer Information Systems and Industrial Management Applications, Volume 6, pp. 257 – 269, 2014.

[12] A. Jain, A. K. Varshney, and U.C. Joshi, “Short-Term Water Demand Forecast Modelling at IIT Kanpur Using Artificial Neural Networks,” Water Resources Management, 15, pp. 299-321, 2011.

[13] J. Cheng, Q. S. Li, R, Xiao, "A new artificial neural network-based response surface method for structural reliability analysis,", Probabilistic Engineering Mechanics, Volume 23 Issue 1, Elsevier, pp. 51-63, 2008.

[14] S. Karsoliya, “Approximating Number of Hidden layer neurons in Multiple Hidden Layer BPNN Architecture,” International Journal of Engineering Trends and Technology,3(6), pp. 714-717, 2012.

[15] E. Istook, T. Martinez, “Improved backpropagation learning in neural networks with windowed momentum,” International Journal of Neural System, 12(3–4): pp. 303–318, 2002.

[16] N. O. Attoh-Okine, “Analysis of Learning Rate and Momentum Term in Back Propagation Neural Network Algorithm Trained to Predict Pavement Performance,” Journal of Advances in Engineering Software, 30(4), pp.293–302, 1999.

[17] S. K. Lahiri, K. C. Ghanta,"Artificial neural network model with parameter tuning assisted by genetic algorithm technique: study of critical velocity of slurry flow in pipeline," Asia-Pacific Journal of Chemical Engineering, Volume 5 Issue 5, 2009.

[18] L. C. P. Velasco, P. B. Bokingkito and J. T. Vistal, "Week-Ahead Load Forecasting using Multilayer Perceptron Neural Network for a Power Utility", Proceedings of the 17th Conference of the Science Council of Asia, National Research Council of the Philippines, 2017

[19] A. D. Anastasiadis, G. D. Magoulas and M. N. Vrahatis, "An Efficient Improvement of the RPROP Algorithm,", In Proceedings of the First International Workshop on Artificial Neural Networks in Pattern Recognition, 2003.

[20] L. Al Shalabi, Z. Shaaban and B. Kasasbeh, "Data Mining: A Preprocessing Engine," Journal of Computer Science, 2 (9), pp. 735-739, 2006.

[21] I. S. Msiza, F. V. Nelwamondo and T. Marwala "Water Demand Prediction using Artificial Neural Networks and Support Vector Regression," Journal of Computers, Volume 3 Number 11, 2018.

[22] P. P. Balestrassi, E. Popova, A.P. Paiva, and J. W. Marangon Lima, J.W., “Design of Experiments on Neural Network’s Training for Nonlinear Time Series Forecasting” pp. 1160-1178, 2009.

[23] A. Abrahan, "Artificial Neural Networks", John Wiley & Sons, 2005.

[24] A. K. Palit, and D. Popovic, “Computational Intelligence in Time Series Forecasting Theory and Engineering Applications” Springer, XXII, 2005.

[25] S. Behboudian, M. Tabesh, and M. Falahnezhad, “A Long-Term Prediction of Domestic Water Demand Using Preprocessing in Artificial Neural Network,” Journal of Water Supply: Research and Technology-Aqua, pp31-42, 2014.