Published by Avanti Publishers

International Journal of Petroleum Technology

ISSN (online): 2409-787X

Interpretation of Neural Network Models – New Insights of Estimating Static Bottom-Hole Pressures of Gas Wells in Nigerian Petroleum Provinces

Clement E. Udoma[1], Anietie N. Okon[1,*], Stella U. Udoeyop[1] and Wilfred C. Okologume[2]

[1]*Department of Chemical and Petroleum Engineering, University of Uyo, Uyo, Nigeria*

[2]*Department of Petroleum Engineering, Federal University of Petroleum Resources, Effurun, Nigeria*

ARTICLE INFO

Article Type: Research Article

Academic Editor: Shastri Nimmagadda

Keywords:
Pseudo-reduce pressure
Poettmann integral value
Static bottom-hole pressure
Pseudo-reduce temperature
Neural network-based models

Timeline:
Received: October 07, 2023
Accepted: November 25, 2023
Published: December 05, 2023

Citation: Udoma CE, Okon AN, Udoeyop SU, Okologume WC. Interpretation of neural network models – new insights of estimating static bottom-hole pressures of gas wells in Nigerian petroleum provinces. Int J Petrol Technol. 2023; 10: 135-150.

DOI: https://doi.org/10.15377/2409-787X.2023.10.10

ABSTRACT

Methods to determine static bottom-hole pressure (BHP) from surface measurements include the average temperature and z-factor method, the Sukkar-Cornell method, the Cullender-Smith method, and the Poettmann method. Among these methods, the Poettmann method is preferable in the petroleum industry but with a concern for software developers, as the integral values to determine the static BHP are tabular. In this study, neural network-based models to predict the integral values using pseudo-reduced pressures and temperatures were developed. The 2-3-1, 2-4-1, and 2-5-1 neural-based models had overall correlation coefficients (R) of 0.9974, 0.99835, and 0.99745, respectively, for the maximum-minimum normalization method and R of 0.99745, 0.99805, and 0.9992 for the clip-scaling method. Comparing the models' predictions with the Lagrangian interpolated values resulted in R of 0.99895 and 0.9995 for the maximum-minimum and clip-scaling-based models. Thus, the developed models can predict Poettmann's integral values without table look-up to estimate static BHP in gas wells.

*Corresponding Authors

Email: anietieokon@uniuyo.edu.ng
Tel: +(234) 803 671 6815

1. Introduction

Pressure is the continuous physical force exerted on the object's surface in a vertical direction per unit area. The pressure generated by static fluids depends on gravity's depth, density, and acceleration. This also applies to the pressure generated by flowing liquids down the hole [1]. Thus, the focus is on static bottom hole pressure (BHP). The static BHP is the well's pressure in the opposite direction of the formation produced after the well is closed and the pressure stabilises. The oil reservoir comprises layers with special interlayer properties containing the three liquid phases of water, oil, and gas. The flow of these fluids in their phases can be controlled by individual layers separated by either permeable or impermeable interfaces [2]. The simultaneous flow of water, oil, and gas in vertical tubes is crucial in the oil industry [3]. Over the years, multiphase flow in oil plants has been studied, and many research materials have been published. The industry is interested in determining accurate pressure losses caused by multiphase flow in tube installations [4]. To optimize production, it is necessary to accurately determine the bottom hole pressure (BHP) [5]. Historically, BHP data have been obtained by direct measurements with a downhole gauge [6, 7]. As a result, there is much redundancy due to mechanical failure of the downhole equipment [8].

However, it is not economical or practical to obtain BHP data from drilling equipment (pressure sensors) [9-11]. Consequently, several methods have been developed to determine BHP directly from surface measurements [12]. These methods include the z-factor method of adjusting the average temperature and the wellbore pressure to the depth and gas density and are sufficiently accurate for all applicants [13]. In general, this method is applied to shallow gas wells. It is used with more rigorously correct methods, such as the Poettmann [14] method for deeper gas wells. This methodology proposed by Poettmann assumes an average but constant temperature value. Unlike average temperatures and z-factor methods, Poettmann allows z-factors to change with pressure, thus providing a more rigorous calculation technique.

Contrasting with the Poettmann method, Cullender and Smith [15] developed a method that does not simplify the hypothesis of variations in well temperature or z-factors [16, 17]. This method is more stringent than previous calculation methods and applies to a much wider range of gas well pressures and temperatures. Cullender and Smith also proposed to use the Simpson rule to achieve better accuracy than the trapezoidal rule. Therefore, the Poettmann method is suggested for more accurate results because it does not require repetition [18]. Unfortunately, the solution to determining Poettmann's integral value for static BHP is in table form. Therefore, to use this method to calculate the static pressure of the bottom hole (BHSP), a table search is required, and a table interpolation is required between the pressure and the temperature to determine the pseudo-reduced pressure (P_{pr}) and the pseudo-reduced temperature (T_{pr}). As a result, given the widespread acceptance of the Poettmann method, it is important to have a model that is not limited to table searches but applies to all Poettmann data sets. Therefore, this study developed a neural network-based model for determining Poettmann integral values to ameliorate table lookup challenges and estimate static BHP in gas wells.

2. Overview of Existing Static Bottom-hole Pressure Estimation Methods

Precise bottom-hole pressure (BHP) data is crucial for gas-reserve engineering calculations. The pressure gauges should ideally be positioned near the bottom of the well to allow for direct measurement of these pressures. Nevertheless, bottom-hole measurements are sometimes too costly and inefficient [16, 19]. In the past, bottom-hole pressure (BHP) was determined using bottom-hole pressure measurements. However, because of redundancy and economic drawbacks, BHP measurements appear less successful, which is why BHP prediction processes are used in estimating. Just the additive pressure exerted by the weight of the static fluid column needs to be calculated to estimate the static bottom-hole pressure from surface measurement [20]. Equation 1 illustrates that, in the case of a static gas column, the friction and kinetic energy effects are abolished, as they are equal to zero [13].

$$dP = \frac{0.01875\rho_g P}{zRT} cos\ \theta\ dL \tag{1}$$

where dP is the change in the pressure of the wellbore, z is the factor of gas deviation, R is the constant gas, T is the temperature, P is the pressure, ρ_g is the gas density, θ is the angle of the diagonal well, and dL is the change in the length of the well.

Ikoku [16, 19] claims that because of well circulation, the temperature profile in the shut-in gas well is not a straight line. As a result, Equation 1, or the universal vertical flow equation, is a particular instance of the static condition. Thus, every approach to estimate BHSPs from surface observations starts with Equation 1. Table **1** displays the four well-known methods for determining the static bottom-hole pressure from surface readings.

Table 1: Static bottom-hole pressure estimation methods and their limitations.

	Estimation Methods	Models	Limitations
i.	Average temperature and z-factor method	$\int_{P_{ws}}^{P_{ts}} \frac{dP}{P} = -\frac{0.01875\rho_g L\cos\theta}{\bar{z}\bar{T}} \int_0^L dL$ where P_{ts} is the static pressure, P_{ws} is the static pressure of the bottom hole, dP is the change in the wellbore pressure, $\bar{z}$ is the average gas deviation factor, $\bar{T}$ is the average temperature, P is the average pressure, ρ_g is the gas density, θ is the angle of the slanted well, L is the length of the well and dL is the change in the length of the well.	It applies only to shallow gas wells. It allows the calculation of bottom-hole pressures with a trial-and-error procedure.
ii.	Sukkar-Cornell [21] method	$\int_{0.2}^{P_{pr,wf}} \frac{\frac{zdP_{pr}}{P_{pr}}}{1+B\left(\frac{z}{P_{pr}}\right)^2} = \int_{0.2}^{P_{pr,ts}} \frac{\frac{zdP_{pr}}{P_{pr}}}{1+B\left(\frac{z}{P_{pr}}\right)^2} + \frac{0.01875\rho_g L\cos\theta}{T}$ $B = \frac{6.67\times10^{-4} f q_g^2 \bar{T}^2}{d^5 P_{pc}^2 \cos\theta}$ where P_{tf} is the flow pressure of the wellhead, P_{wf} is the flow pressure of the bottom of the well, P_{pr} is the pseudo-reduced pressure, P_{pc} is the pseudo-critical pressure, dP is the change in the pressure of the well, z is the gas deviation factor, $\bar{T}$ is the average temperature, P is the average pressure, ρ_g is the gas density, θ is the angle of the diagonal well, f is the Moody friction factor, d is the diameter of the production string, L is the length of the well, dL is the change in the length of the well, and 0.2 is the arbitrary constant	It is used for gas and condensate wells with less than 10,000 psi pressures. It has a table lookup for the solved integral.
iii.	Cullender-Smith [22] method	$\int_{P_{ws}}^{P_{ts}} \frac{zT}{P} dP = \frac{(I_{mp}+I_{ts})}{2}(P_{mp}-P_{ts}) + \frac{(I_{ws}+I_{mp})}{2}(P_{ws}-P_{mp})$ where, $P_{mp} = P_{ts} + \frac{\alpha}{I_{ts}+I_{mp}}$, and $P_{ws} = P_{mp} + \frac{\alpha}{I_{ws}+I_{mp}}$ $I = \frac{zT}{P}$ and $\alpha = 0.01875\rho_g L\cos\theta$ where I is the integrand estimated at the surface, midpoint, or bottom-hole conditions (denoted by the subscripts ts, mp, and ws, respectively)	It requires iteration. It allows a trial-and-error procedure. It applies to shallow and deep wells and sour gases and may involve digital computations.
iv.	Poettmann [14] method	$\int_{0.2}^{P_{pr,ws}} \frac{zdP_{pr}}{P_{pr}} = \int_{0.2}^{P_{pr,ts}} \frac{zdP_{pr}}{P_{pr}} + \frac{0.01875\rho_g L\cos\theta}{\bar{T}}$ where the various variables are as presented for other methods	The established solution for the Poettmann integral values to determine the static BHP is in table form and requires table lookup.

The average temperature and deviation factor approach is the most straightforward BHSP method that is typically used. The sole requirement is that the temperature and pressure be average [16, 19]. Applying the average temperature and deviation factor approach to shallow gas wells is its only drawback. Additionally, Ikoku [16, 17] noted that one of the simplest approaches for determining bottom-hole pressure is the Sukkar and Cornell [21] method, which enhances accuracy and eliminates the need for trial-and-error computations. To determine the z-factor, knowing the pseudo-reduced temperature (T_{pr}) and pressure (P_{pr}) is compulsory. Cullender and Smith [22] developed a method that makes no simplifying assumptions for the variation of either

temperature or the z-factor in the wellbore. As a result, this method is more rigorous than the earlier approaches and applies to a much wider range of gas-well pressures and temperatures [13]. However, Cullender and Smith [22] applied trapezoidal numerical integration to evaluate the integral (Table **1**). They incorporate a two-step calculation procedure that uses an intermediate pressure value at the midpoint of the production string. Besides, Cullender and Smith [22] suggested Simpson's rule (Equation 2) to achieve better accuracy than the trapezoidal rule visible in Table **1** [16, 20].

$$P_{ws} = P_{ts} + \frac{6\alpha}{I_{ts} + 4I_{mp} + I_{ws}} \tag{2}$$

and the alpha (α) in Equation 2 is expanded in Table **1**.

On the other hand, Poettmann's [14] approach assumes an average but constant value for the temperature, unlike the average temperature and the z-factor method. Unlike the earlier methods for estimating static bottom-hole pressure, the Poettmann approach is straightforward, simple, and more accurate [18]. It does not require iteration and can be applied to a wider range of pressure and temperature conditions. This is due to the inclusion of the z-factor in the integrals (Table **1**). Thus, this method accounts for the z-factor's pressure variation but not temperature. According to Lee and Wattenberg [13], an even more accurate result can be achieved if the constant temperature assumption is eliminated from the Poettmann approach.

3. Methodology

This section handles the data source, the dataset preprocessing stages involved in the work, and the description of feed-forward back-propagation neural network training with MATLAB, including the hyperparameters of the neural networks.

3.1. Data Acquisition and Preparation

The datasets for this study were extracted from Lee and Wattenberg [13]. They consist of 2360 data points with pseudo-reduced pressures of the wellhead (P_{pr}) and pseudo-reduced temperatures (T_{pr}) as input variables and Poettmann's integral values as output variables. The statistical description and correlations of the extracted datasets, namely P_{pr}, T_{pr}, and Poettmann integral values are presented in Table **2** and Fig. (**1**).

Table 2: Statistical description of the neural-based models' input and output variables.

Variables	Maximum	Minimum	Range	Average	Std. Deviation
Pseudo-reduced Pressure (Ppr)	12.0	0.2	11.8	6.1076	3.4494
Pseudo-reduced Temperature (Tpr)	3.0	1.05	1.95	1.7375	0.5739
Poettmann integral values	4.249	0.0	4.249	2.8207	0.8442

The maximum-minimum normalization (Equation 3) and the clip scaling (Equation 4) approaches were used to reduce the dataset ranges. Thus, the maximum-minimum approach normalized the data sets to values between 0 and 1, while the clip method scaled the data sets to values between -1 and 1 [23].

$$y_{scaled} = \frac{y_i - y_{min}}{y_{max} - y_{min}} \tag{3}$$

$$y_{scaled} = 2\left(\frac{y_i - y_{min}}{y_{max} - y_{min}}\right) \tag{4}$$

where y_{scaled} denotes the scaled values for input or output parameters, y_i is the values of the non-normalized parameters, y_{min} and y_{max} represent the minimum and maximum values of the non-normalized parameters, respectively. According to Okon and Ansa [24] and Okon *et al*. [25], normalizing the datasets for the neural

network training is necessary for the following reasons: adequate adjustment of the network connecting weights for optimum prediction and reducing the sensitivity of the sigmoidal (i.e. transfer or activation) function to large datasets values.

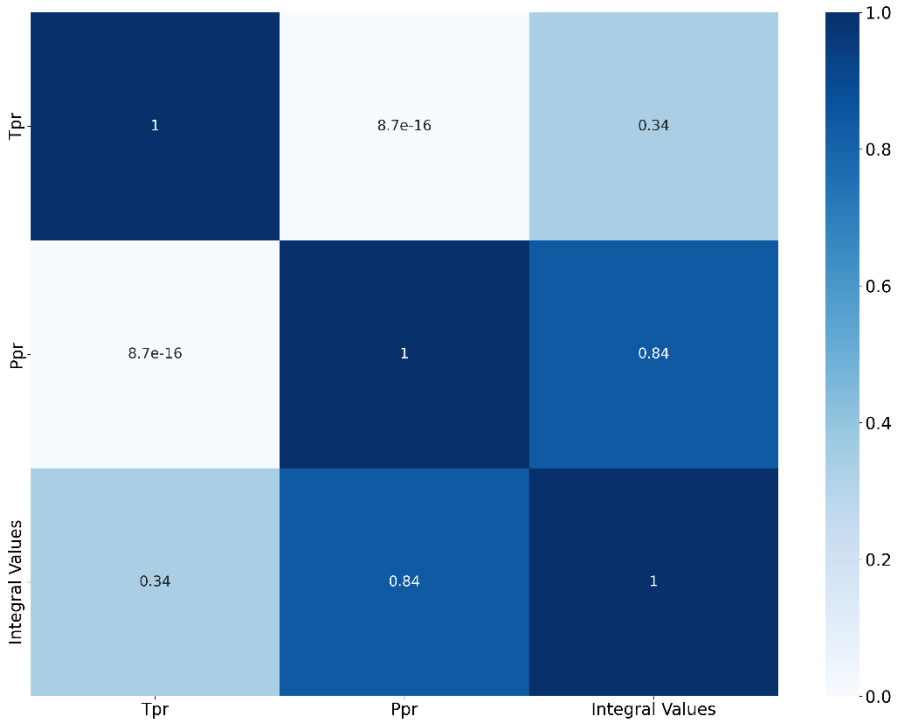

Figure 1: Correlations plot of the pseudo-reduced pressure and temperature with the Poettmann integral values.

3.2. Neural Network Development and Training

The neural network-based model to predict Poettmann's integral values was developed using MATLAB's neural fitting tool (nftool) (Matrix Laboratory) 2020a mathematical software. The normalized input (i.e., P_{pr} and T_{pr}) and output (i.e., integral values) datasets were exported from Microsoft Excel to the MATLAB nftool environment. Subsequently, the imported data sets were randomly partitioned into three parts, namely, the training set (70%), that is, 1652 data points; the validation set (15%) – 354 data points; and the testing set (15%) – 354 data points. The network training was supervised learning, as the target datasets (i.e. integral values) were provided to the network as output data. In the training phase, the network learning algorithm Levenberg-Marquardt (trainlm in MATLAB) adjusts the weights and biases based on the feedforward back-propagation (FFBP) method until the neurons learn the input-output data structure [25]. The network learning algorithm (Levenberg-Marquardt) was chosen because it is faster (i.e., takes less training time) and suitable for non-linear functions minimization compared to other learning algorithms, namely Bayesian regularization and scaled conjugate gradient, available in the MATLAB software. Besides, the activation functions of the non-linear (sigmoid, i.e. tansig in MATLAB) and linear (purelin in MATLAB) were implemented to introduce non-linearity and linearity to the outputs of hidden and output layers' neurons, respectively. The chosen activation functions are to ensure conformity with the maximum-minimum normalization approach output values range, that is, 0 to 1. According to Okon *et al*. [25], the weights and biases that produce the lowest error from the supervised datasets are the best generalizations of the network. Thus, the lowest MSE and the correlation coefficient (R) values were also used to select the best network topology (architecture).

After several trials of different network topologies, the best network performance was obtained with three topologies: 2-3-1, 2-4-1, and 2-5-1. For more information on the neural network stopping criteria, weights generation and adjustment, neural network computations, and generalisation, readers can read published works by Mahmoudi and Mahmoudi [26], Okon and Ansa [24], and Okon *et al*. [25]. As a result, Table **3** displays the

default values for the trained neural networks' parameters. The following are the fundamental steps in neural network learning:

i. reading the input datasets and the anticipated output;
ii. calculating the network output using weighted sums and transfer functions;
iii. comparing the predicted output of the network with the target;
iv. computing and updating the fitness (MSE) value based on the comparison;
v. repeat steps (ii) and (iii) until all training points have been used;
vi. appropriately modify weights to maximize fitness; and
vii. repeat steps (i) through (vi) until a satisfactory fitness level has been developed.

Table 3: Basic settings of neural networks learning parameters.

Parameters	Values
Number of input neuron	2
Number of hidden layers	1
Number of neurons in the hidden layer	3, 4, 5
Number of the output neuron	1
Input activation function	*Tansig*
Output activation function	*Purelin*
Learning algorithm (trainlm)	Levenberg-Marquardt
Mean square error (MSE)	1.0×10^{-5}
Number of epochs	1000
Training rate	0.7

4. Results and Discussion

4.1. Performance of the Developed Neural Networks

According to Song *et al*. [27], neural networks' data learning and non-linear modelling abilities are factors for their consideration over other machine learning methods. Also, it is more easily represented mathematically than any other machine learning approach. Thus, this study's choice of the neural network model is based on the mentioned advantages. As mentioned earlier, the networks' topologies are based on their training, validation, and testing best performance (i.e., mean square error (MSE) and correlation coefficient (R)). They are visible in Figs. (**2-5**) and Fig. (**A1-A6**) (in Appendix) for the maximum-minimum and clip normalization datasets. The developed neural networks are feed-forward backpropagation networks with 2-3-1, 2-4-1, and 2-5-1 topologies. This implies that the networks have two input neurons and one output neuron. At the hidden layer, the networks have three, four, and five neurons connected to the output neuron. Figs. (**2**-**3**) present the MSE and R performance of the networks trained with maximum-minimum normalised datasets. These figures show that the various networks' validation, testing, and overall performance were better than the training performance. This observation is visible in the MSE and R performance of the various networks, as the MSE trend for validation and testing was less than that of the training stage. Again, the R trends for validation, testing, and overall performance were higher (i.e., close to 1) than the training performance. Thus, the results obtained depict that the different topologies trained with maximum-minimum datasets had an overall R-value of 0.99747, 0.9990, and 0.99929 for the 2-3-1, 2-4-1 and 2-5-1, respectively, and MSE values of 1.630×10^{-4}, 9.104×10^{-4} and 5.906×10^{-5}. The implication of these statistical values implies that the neural network predictions were a good fit with Poettmann's integral values.

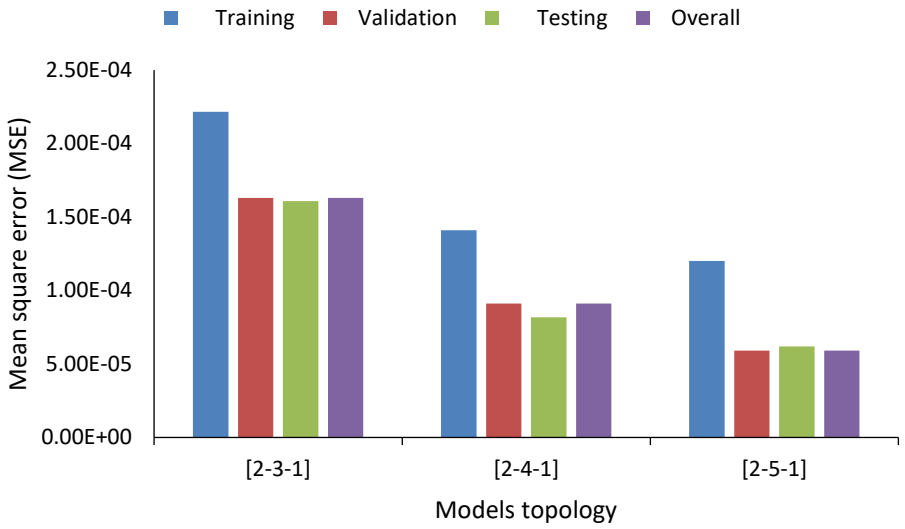

Figure 2: Mean square errors of the various topologies training stages with maximum-minimum normalized datasets.

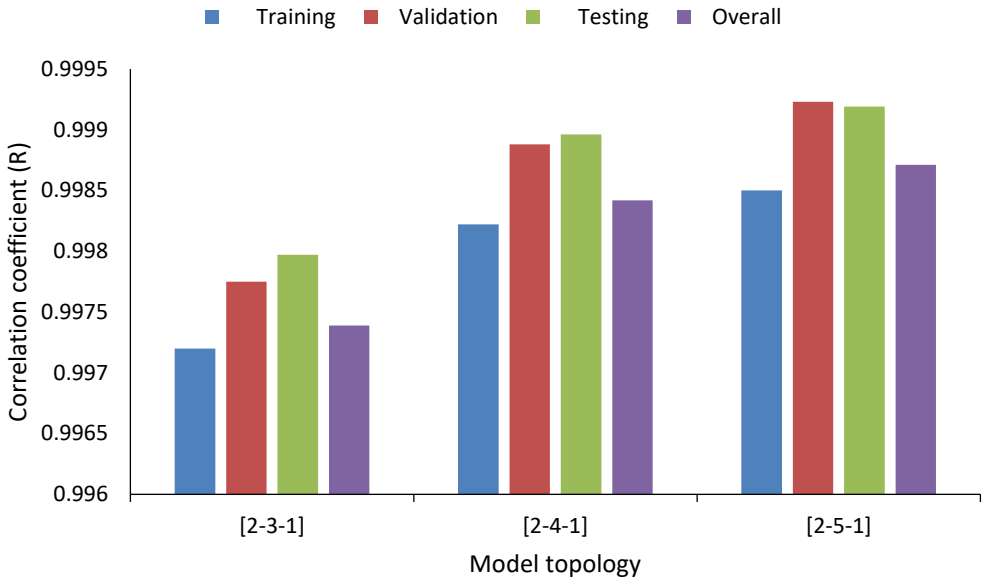

Figure 3: Correlation coefficient of the various topologies training stage with maximum-minimum normalized datasets.

Again, the regression performance of the networks' predicted output and the provided output (i.e., target) for training, validation, and testing stages of the various neural networks trained with the maximum-minimum normalized datasets are in Fig. (**B1-B3**) in the Appendix. The statistical indicators in Figs. (**2-3**) and diagonal trends from the networks' regression plots (Fig. (**B1-B3**) in the Appendix indicate that the network predictions (overall performance) were close to the Poettmann integral values. The assertation is observed in the R-values for the various networks, which in all cases are closer to 1 (an average of 0.99817), and their corresponding MSE values were closer to zero (about 1.044×10^{-4}). Statistically, these R and MSE values are strong indicators of the developed neural networks' good prediction performance. The R-value implied that the neural networks can predict the Poettmann integral values with 99.82% certainty. Again, Al-Bulushi *et al*. [28] and Tugwell and Livinus [29] reported that a close prediction between actual field and model-predicted datasets would result in a unit diagonal trend, as visible in Fig. (**B1-B3**).

On the other hand, the performance of the networks trained with the clip-normalized datasets is in Figs. (**4-5**). In these figures, the general performance of the various topologies (i.e., 2-3-1, 2-4-1, and 2-5-1) is in sync with the

results obtained in Figs. (**2**-**3**) for the networks trained with maximum-minimum normalized datasets. In addition, the various networks' output-target (i.e., cross plots) performance during training, validation, and testing stages using the clip-normalized datasets are visible in Fig. (**B4-B6**) in the Appendix. The performance indicators in Table **4** and the diagonal alignment of data points in Fig. (**B1-B3**) showed that the networks' overall performance was close to the Poettmann integral values. This is because the R and MSE values obtained for the various networks are closer to 1, with their respective MSE values closer to zero. From a statistical standpoint, the R and MSE values obtained for the networks trained with clip-normalized datasets strongly indicate the networks' good prediction performance [25, 28].

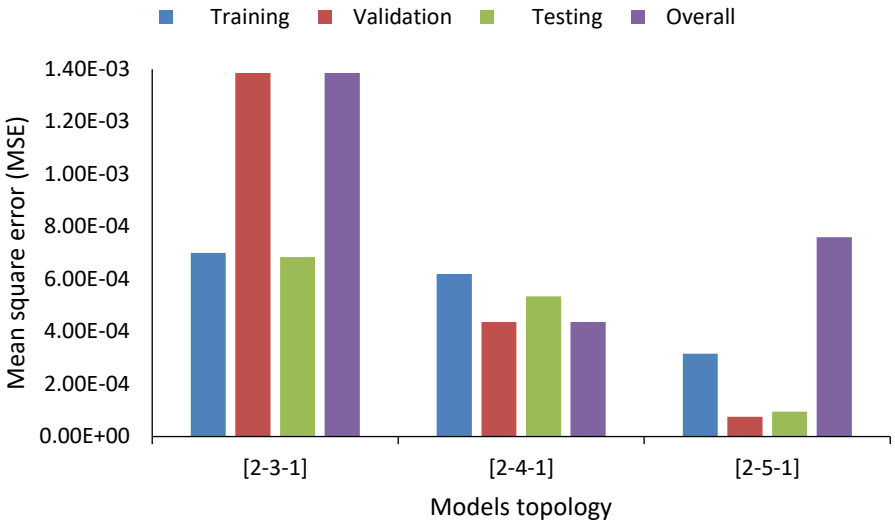

Figure 4: Mean square errors of the various topologies training stages with clip-scaling datasets.

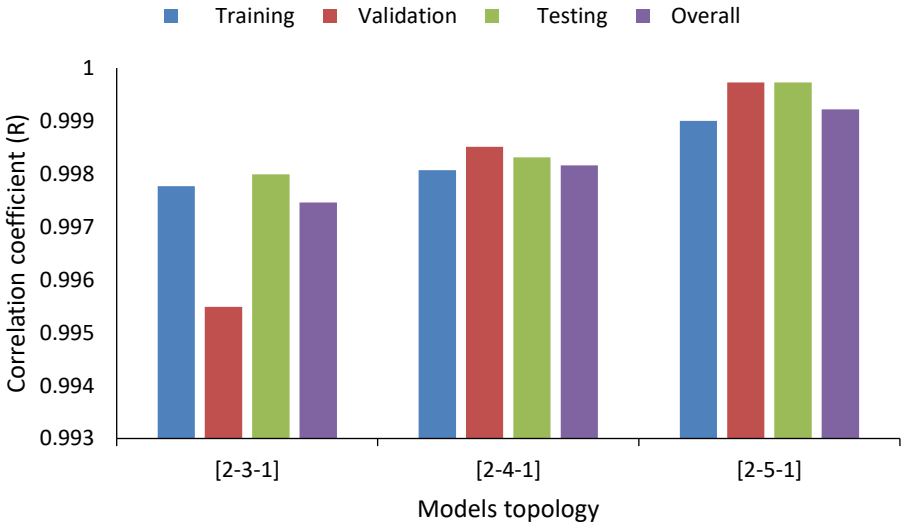

Figure 5: Correlation coefficient of the various topologies training stage with clip-scaling datasets.

Comparing the performance of the different network topologies, the overall R and MSE values showed that the 2-5-1 network for both maximum-minimum and clip-normalized datasets performed slightly better than the 2-3-1 and 2-4-1 networks. This observation is attributed to the number of neurons (i.e. five neurons) at the 2-5-1 network's hidden layer, which is more than the 2-3-1 and 2-4-1 networks. The overall R-values for the 2-5-1 topology indicated that the developed network could predict Poettmann's integral values with 99.870% and 99.92% certainty with the maximum-minimum and clip normalization methods.

Table 4: Neural network performance for the clip scaling approach.

Topology	Training		Validation		Testing		Overall	
	MSE	R	MSE	R	MSE	R	MSE	R
2-3-1	6.995e-4	0.99777	1.386e-3	0.99549	6.842e-4	0.99799	1.386e-3	0.99746
2-4-1	6.196e-4	0.99807	4.367e-4	0.99851	5.335e-4	0.99831	4.367e-4	0.99816
2-5-1	3.163e-4	0.9990	7.593e-5	0.99973	9.502e-5	0.99973	7.593e-4	0.99922

According to Sircar *et al*. [30] and Effiong *et al*. [31], the mathematical representation of the neural network computations parameters (i.e., inputs, weights, biases, and output) in vector form and transfer or activation functions (i.e., *tansig* and *purelin*) are related, as expressed in Equation 5;

$$y_{ANN} = f_{purelin}\left[\sum_{j} f_{tan\,sig}\left(\sum Iw_{ij}x_i + b_i\right)Lw_{kj} + b_k\right] \tag{5}$$

where y_{ANN} denotes the neural network predicted output (in normalized form), $f_{purelin}$ is the output neuron activation function (i.e. *purelin*), Lw_{kj} is the hidden layer neurons' weights from the *j*th neuron to the *k*th output layer neuron, $f_{tan\,sig}$ represents the transfer function (*tansig*) at the hidden neuron, Iw_{ij} is the input neuron weights from the *i*th neuron to the *j*th hidden layer neuron, and x_i denotes the input variables. Again, b_i and b_k are the hidden and output neurons' biases, respectively.

Therefore, using the various network topologies in this study, the neural network mathematical representation for the Poettmann integral values P_{value} prediction is expanded in Equations 6 through 8. Equation 6 represents the 2-3-1 network topology, while Equations 7 and 8 represent the 2-4-1 and 2-5-1 network topologies, respectively;

$$(P_{value})_{ANN} = \sum_{i=1}^{1}\left\{f_{purelin}\left[\sum_{i=1}^{3}\sum_{j=1}^{2} f_{tan\,sig}\left((P_{pr}j_1 + T_{pr}j_2)_i + b_i\right)\right] \times Lw_{ij} + b_{ki}\right\} \tag{6}$$

$$(P_{value})_{ANN} = \sum_{i=1}^{1}\left\{f_{purelin}\left[\sum_{i=1}^{4}\sum_{j=1}^{2} f_{tan\,sig}\left((P_{pr}j_1 + T_{pr}j_2)_i + b_i\right)\right] \times Lw_{ij} + b_{ki}\right\} \tag{7}$$

$$(P_{value})_{ANN} = \sum_{i=1}^{1}\left\{f_{purelin}\left[\sum_{i=1}^{5}\sum_{j=1}^{2} f_{tan\,sig}\left((P_{pr}j_1 + T_{pr}j_2)_i + b_i\right)\right] \times Lw_{ij} + b_{ki}\right\} \tag{8}$$

where $(P_{value})_{ANN}$ is the neural network predicted oil flow rate in normalized form? The variables j_1 and j_2 are the weights of the network inputs: P_{pr} and T_{pr} to the hidden layer neurons; Lw_{ij} represents the hidden layer weights that connect the output layer neuron; b_i and b_k are biases at the hidden and output neurons, respectively. Then, $f_{purelin}$ and $f_{tan\,sig}$ were defined in Equation 5. The sigmoidal function $\sigma(z)$, that is, *tansig* ($f_{tan\,sig}$) in Equations 6 through 8, is expressed in Equation 9 as [30];

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{9}$$

where z represents the computation $(P_{pr}j_1 + T_{pr}j_2)_i + b_i$.

The weights and biases of the various neural network-based models for Poettmann integral value prediction are in Tables **5** and **6**.

Table 5: Network weights and biases using the max.-min. normalized datasets.

Network Topology	Input layer Weights		Input Biases (b_i)	Hidden Layer Weight (Lw_i)	Output Biases (b_k)
	j_1 (P_{pr})	j_2 (T_{pr})			
2-3-1	-0.89320	-0.18915	-0.07286	-0.57402	-4.37937
	-0.38262	2.41253	2.60654	0.470652	
	-3.70729	0.09959	-4.73012	-4.40663	
2-4-1	1.24439	-2.25987	-311497	2.149481	-3.87005
	-0.89375	2.03561	2.76692	3.080096	
	0.72459	0.06494	0.49369	0.962884	
	5.13128	-0.11212	6.03298	3.079741	
2-5-1	-0.52614	5.24644	5.27148	0.21425	-4.43369
	2.57188	-6.22888	-4.48779	-0.04145	
	-1.08699	0.82044	-1.18028	-0.32412	
	-0.61727	-0.41521	-0.40705	-0.75189	
	-5.04332	0.07311	-6.13619	-4.18262	

Table 6: Network weights and biases using the clip-scaling datasets.

Network Topology	Input Layer Weights		Input Biases (b_i)	Hidden Layer Weight (Lw_i)	Output Biases (b_k)
	j_1 (P_{pr})	j_2 (T_{pr})			
2-3-1	-0.89004	-0.18703	-0.05371	-0.56593	-83.73643
	-0.35040	2.354191	2.582028	0.49999	
	-3.46156	0.095305	-6.00162	-83.7471	
2-4-1	0.219689	-0.32706	-0.55468	-0.554679	-132.3021
	-0.15742	0.29047	0.598824	0.598825	
	-0.08235	0.21545	0.876287	0.876287	
	5.028566	-0.09714	7.269345	7.269345	
2-5-1	0.501128	1.292399	3.516369	1230.675	-1198.769
	0.441548	0.012952	3.312989	224.5664	
	-0.57831	-1.34445	-2.997368	797.5542	
	4.940618	0.007162	8.191205	266.2791	
	-0.63871	-1.39099	-2.845391	-276.0254	

4.2. Explicit Representation of the Neural-Based Models

Explicit presentation of the developed neural network-based models will make them purposeful. Therefore, explicit representation of the 2-3-1 topologies models for the maximum-minimum normalization and clip scaling data sets was considered.

4.2.1. Model Based on Maximum-Minimum Normalization Approach

The explicit presentation of the neural-based model for predicting Poettmann's integral values based on the maximum-minimum normalization method is expressed in Equation 10.

$$P_{value} = 4.249(P_{value})_{ANN} \tag{10}$$

where $(P_{value})_{ANN}$ is the neural network predicted Poettmann's integral value (i.e., output) in normalised form, which is expressed in Equation 11 and P_{value} denotes the Poettmann integral value in the de-normalize form;

$$(P_{value})_{ANN} = [A + B + C - 4.37937] \tag{11}$$

Thus, $(P_{value})_{ANN}$ depicts the computation at the output layer neuron where the variables A, B, and C are expanded in Equations 12 through 14. These variables A, B, and C denote the computations at the hidden layer first, second, and third neurons, respectively, to the output layer neuron;

$$A = \left[\frac{-0.57402}{1 + e^{-\left(-0.89320(P_{pr})_n - 0.18915(T_{pr})_n - 0.07286\right)}}\right] \tag{12}$$

$$B = \left[\frac{0.47065}{1 + e^{-\left(-0.38262(P_{pr})_n + 2.41253(T_{pr})_n + 2.60654\right)}}\right] \tag{13}$$

$$C = \left[\frac{-4.40663}{1 + e^{-\left(3.70729(P_{pr})_n + 0.09959(T_{pr})_n - 4.73012\right)}}\right] \tag{14}$$

and the variables $(P_{pr})_n$ and $(T_{pr})_n$ are the normalized input parameters (i.e. pseudo-reduced pressure and temperature) expressed as $\left(\frac{P_{pr}-0.2}{11.80}\right)$ and $\left(\frac{T_{pr}-1.05}{1.95}\right)$, respectively.

4.2.2. Model Based on Clip Scaling Approach

Based on the clip scaling method, the explicit presentation of the neural-based model for predicting Poettmann's integral values is expressed in Equation 15.

$$P_{value} = 2.1245[1 + (P_{value})_{ANN}] \tag{15}$$

where $(P_{value})_{ANN}$ represents the neural network predicted Poettmann's integral value (i.e. output) in normalized form, expressed in Equation 16, and P_{value} denotes the Poettmann integral value in the de-normalized form.

$$(P_{value})_{ANN} = [A + B + C - 83.73643] \tag{16}$$

Therefore, the computation at the output layer neuron (i.e. Equation 16) presents the variables A, B, and C in Equations 17 through 19. These variables A, B, and C denote the computations at the hidden layer first, second, and third neurons, respectively, to the output layer neuron;

$$A = \left[\frac{-0.56593}{1 + e^{-\left(-0.89004(P_{pr})_n - 0.18703(T_{pr})_n - 0.05371\right)}}\right] \tag{17}$$

$$B = \left[\frac{0.49999}{1 + e^{-\left(-0.35040(P_{pr})_n + 2.35419(T_{pr})_n + 2.58203\right)}}\right] \tag{18}$$

$$C = \left[\frac{-83.7471}{1 + e^{-\left(3.46156(P_{pr})_n + 0.095305(T_{pr})_n - 6.00162\right)}}\right] \tag{19}$$

and the variables $(P_{pr})_n$ and $(T_{pr})_n$ are the normalized input parameters (i.e. pseudo-reduced pressure and temperature) expressed as $2\left(\frac{P_{pr}-0.2}{11.80}\right) - 1$ and $2\left(\frac{T_{pr}-1.05}{1.95}\right) - 1$, respectively.

Similarly, representing the 2-4-1 and 2-5-1 topologies models can be presented with Equations 11 and 16 modifications to indicate details for the additional neurons in the hidden layers. The values to present the explicit neural-based models for the 2-4-1 and 2-5-1 topologies are in Tables **5** and **6**.

4.3. Comparison of the Developed Neural-based Models Predictions with Poettmann Integral Values

The performance of the developed neural network-based models for predicting Poettmann integral values was evaluated using some statistical yardsticks, namely coefficient of determination (R^2), correlation coefficient (R), mean absolute error (MAE), average relative error (ARE), average absolute relative error (AARE), mean square error (MSE) and root mean square error (RMSE). The neural-based models predicted or estimated integral values were compared with the actual Poettmann integral values extracted from Lee and Wattenbarger [13]. The results obtained are depicted in Table **7** and Fig. (**6**). In Table **7**, the R^2 and R values show that the developed neural-based models predicted integral values were close to the actual values provided by Poettmann [14]. This statement is because the R^2 and R values for the various neural-based models were close to 1, which statistically means a good fit between the predicted and actual values [29, 31]. Again, the good fits of the developed neural-based models are further observed in Fig. (**6**) as the predicted integral values aligned diagonally with the actual Poettmann integral values. Furthermore, these statistical values in Table **7**, as visualized in Figs. (**7**-**8**), imply that the predictions of the developed neural-based models were closer to the actual Poettmann integral values for estimating static bottom-hole pressure. The assertion is because the estimated statistical indicators MAE, ARE, AARE, MSE, and RMSE values were far less than 1, as visible in Figs. (**7**-**8**).

Table 7: Statistical performance of the neural-based models' predictions.

Models	R^2	R	MAE	ARE	AARE	MSE	RMSE
Maximum-Minimum Normalised Neural-Based							
2-3-1	0.99480	0.99740	0.03866	-0.00204	0.02211	0.01667	0.12911
2-4-1	0.99670	0.99835	0.02778	-0.00147	0.01697	0.01681	0.12966
2-5-1	0.99730	0.99745	0.02295	-0.00118	0.01420	0.01768	0.13298
Clip Normalised Neural-Based							
2-3-1	0.99490	0.99745	0.03766	-0.00102	0.02194	0.01682	0.12970
2-4-1	0.99610	0.99805	0.02997	-0.00165	0.01837	0.0170	0.13037
2-5-1	0.99840	0.99920	0.01379	-0.00186	0.09790	0.01666	0.12908

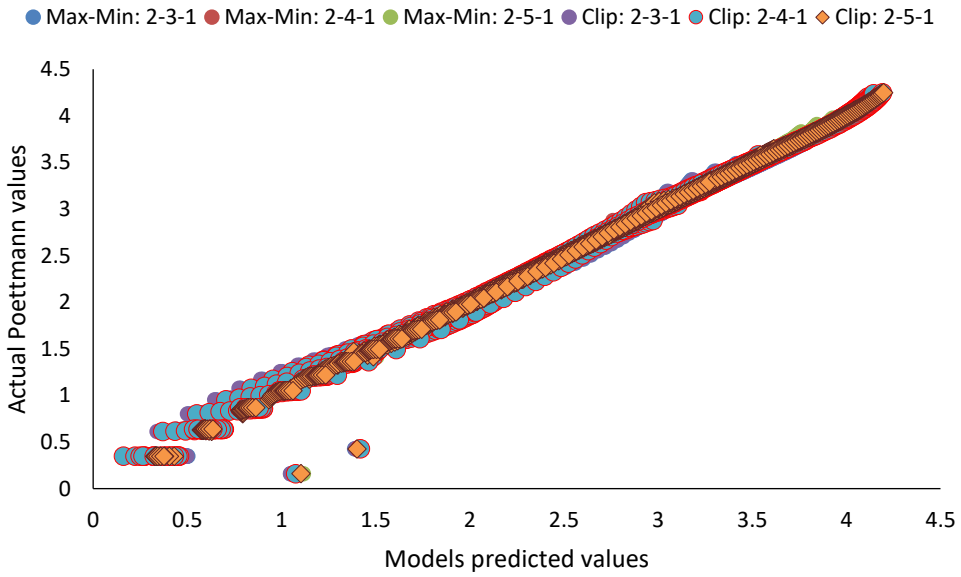

Figure 6: Cross-plot of the models predicted and actual Poettmann integral values.

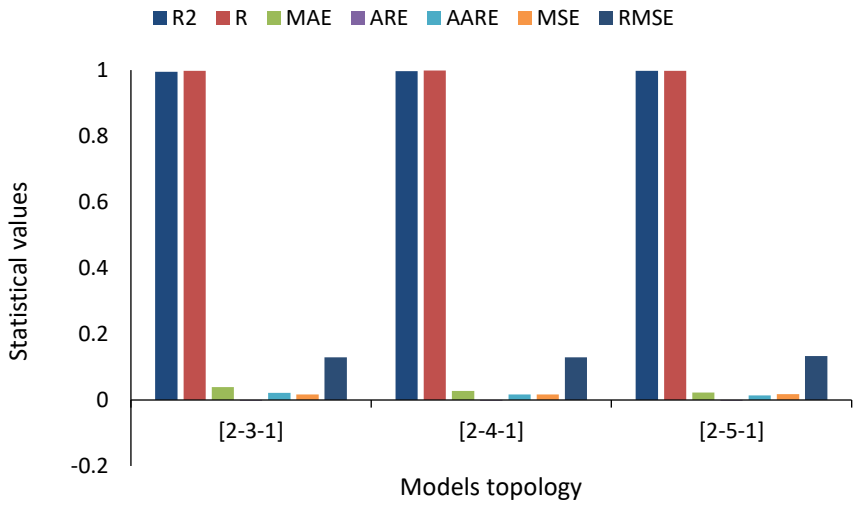

Figure 7: Statistical indicators of the max-min normalized models.

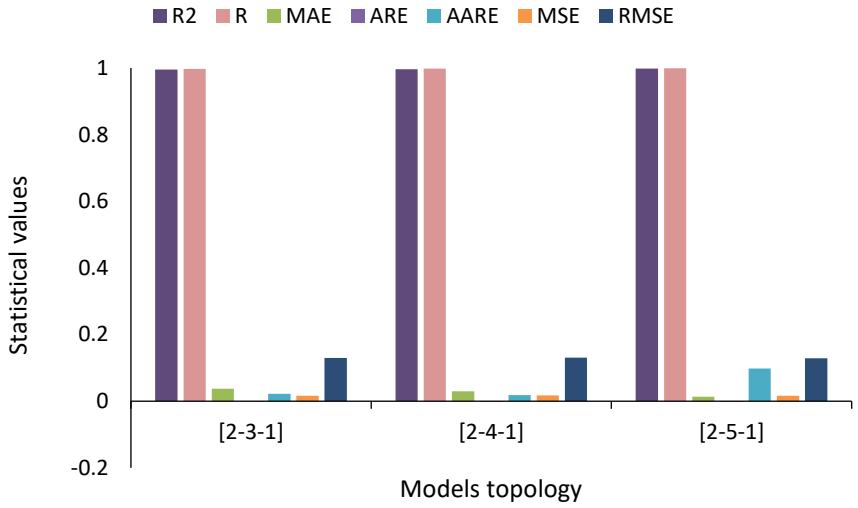

Figure 8: Statistical indicators of the clip-normalized models.

4.4. Generalization and Comparison of the Developed Neural-based Models Predictions and with Lagrangian Interpolation

The effectiveness of any developed model is its capability to predict new sets of data to prove its generalization and applications [32, 33]. As stated previously, the established method to determine the integral values of Poettmann for estimating static BHP is a table lookup. Most of the time, interpolation between two data points is required to obtain the corresponding integral value since the calculated values of P_{pr} and T_{pr} may differ from the exact values provided in the Poettmann integral table. As a result, the generalizability potential of the developed neural-based models was determined using 500 randomly generated P_{pr} datasets at a constant T_{pr} (across the various intervals). The predicted integral values of the developed models were compared with the estimated integral values of the Lagrangian interpolation approach in Equation 20.

$$P_{value}(P_{pr}) = P_{value}(P_{pr_1})\left(\frac{P_{pr_2} - P_{pr}}{P_{pr_2} - P_{pr_1}}\right) + P_{value}(P_{pr_2})\left(\frac{P_{pr} - P_{pr_1}}{P_{pr_2} - P_{pr_1}}\right) \quad (20)$$

where $P_{value}(P_{pr})$ is the required integral value at estimated P_{pr}, $P_{value}(P_{pr_1})$ and $P_{value}(P_{pr_2})$ are the lookup integral values from Poettmann's table at estimated P_{pr_1} and P_{pr_2}, respectively.

Using the earlier mentioned statistical indices and cross-plot, the generalizability of the developed models with Lagrange interpolation was assessed. Table **8** and Fig. (**9**) present the developed neural-based models' generalisation and comparison performance to determine the Poettmann integral values. In Table **8**, the statistical measures between the neural-based models' predictions and Lagrange interpolation estimations resulted in statistical indices (R^2, R, MAE, ARE, AARE, MSE, and RMSE) values that show that the neural-based models predicted integral values agreed with the Lagrange interpolated integral values. Noticeable in these statistical indices are the R^2 and R values, which are closer to 1, indicating a strong agreement between the two approaches.

Also, the cross-plot in Fig. (**9**) depicts a diagonal trend of the data points along a unit slope (i.e., 45° from the origin) for the various neural-based models. This observation shows that the developed neural-based models' predictions with Lagrange interpolation were close. As Al-Bulushi *et al*. [28] reported, a good agreement between the neural-based models' predicted integral values and the Lagrangian interpolated integral values is evidenced by the diagonal trend in Fig. (**9**). Again, the different overcome (i.e., predicted integral values) in Fig. (**9**) from the neural-based models are close to the Lagrange interpolated values, as is observed in Fig. (**6**) for the model's predicted values and the actual Poettmann integral values. This observation implies that any neural-based model or topology would predict the Poettmann integral value with a good degree of fitness (certainty) of 99.75%. Thus, the developed neural-based models would predict the Poettmann integral values for estimating static bottom-hole pressure without a table lookup. This functionality of the developed neural-based models can be exploited in petroleum engineering software applications.

Table 8: Statistical performance of the developed neural-based models with Lagrangian interpolation.

Models	R^2	R	MAE	ARE	AARE	MSE	RMSE
Maximum-Minimum Normalised Neural-Based							
2-3-1	0.9951	0.99755	0.04749	-0.00913	0.04194	0.01243	0.11150
2-4-1	0.9972	0.99860	0.03447	-0.00182	0.03001	0.00579	0.07609
2-5-1	0.9979	0.99895	0.02704	-0.00185	0.02340	0.00408	0.06391
Clip Normalised Neural-Based							
2-3-1	0.9950	0.99750	0.04727	-0.00533	0.04241	0.01251	0.11184
2-4-1	0.9966	0.99830	0.03747	-0.00170	0.03281	0.00677	0.08225
2-5-1	0.9990	0.99950	0.01775	-0.00285	0.01343	0.00146	0.03825

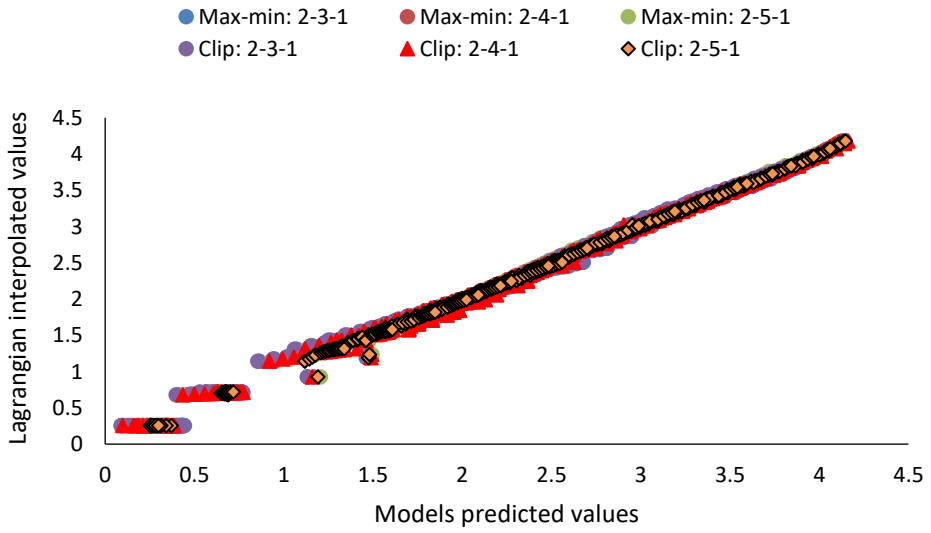

Figure 9: Comparison of the developed model predicted integral values With the Lagrangian interpolation.

5. Conclusion

The Poettmann approach for estimating static bottom-hole pressure is essential in reservoir engineering as it gives a more accurate estimation. Although it is the most reliable and accurate method, it is impeded by table lookup and interpolation between P_{pr} and T_{pr} to estimate Poettmann's integral values. This study developed the feedforward back-propagation (FFBP) neural network model to predict the integral values using different network topologies: 2-3-1, 2-4-1, and 2-5-1, and dataset normalization approaches: maximum-minimum and clip. From the study, the under-listed conclusions are drawn based on the performance of the neural-based models:

i. with the maximum-minimum and clip normalized datasets, the neural-based models predicted values (P_{value}) were close to the Poettmann integral values with the coefficient of determination (R^2) and coefficient of correlation (R) of values of about 99.7% for the various models;

ii. other statistical indicators: MAE, ARE, AARE, MSE and RMSE values are significantly low values to showcase the good fit of the developed models' predictions;

iii. the generalization performance of the developed neural-based models predicted integral, and the Lagrangian interpolated values were close with good coefficient fitness; and

iv. the developed neural-based models can predict Poettmann's integral values with 99.7% certainty (accuracy) without table look-up to determine static bottom-hole pressure in gas wells.

Conflict of Interest

The authors declare that there is no conflict of interest between the authors.

Funding

No funding is received from any funding agency for this work.

References

[1] Akinsete O, Adesiji BA. Bottom-hole pressure estimation from wellhead data using artificial neural network. SPE Nigeria Annual International Conference and Exhibition, Lagos, Nigeria, 5-7 August 2019. https://doi.org/10.2118/198762-MS

[2] Akintola SA, Olakunle F. Pressure drop determination for multiphase flow in a vertical well tubing. Trends Pet Eng. 2021; 1(2): 1-7. https://doi.org/10.53902/TPE.2021.01.000510

[3] Ebrahimi A, Khamehchi E. A robust model for computing pressure drop in vertical multiphase flow. J Nat Gas Sci Eng. 2015; 26: 1306-16. https://doi.org/10.1016/j.jngse.2015.08.036

[4] Faraji F, Santim C, Chong PL, Hamad F. Two-phase flow pressure drop modelling in horizontal pipes with different diameters. Nuclear Eng Des. 2022; 395: 111863. https://doi.org/10.1016/j.nucengdes.2022.111863

[5] Akinseye CA, Omotara OO, Giwa A. Determination of flowing bottom-hole pressure of a well using modified Guo's model. J Environ Sci Comput Sci Eng Technol. 2020; 9(2): 270-86. https://doi.org/10.24214/jecet.C.9.2.27086

[6] El-Saghier RM, Abu El-Ela M, El-Banbi A. A model for calculating bottom-hole pressure from simple surface data in pumped wells. J Petrol Explor Prod Technol. 2020; 10: 2069-77. https://doi.org/10.1007/s13202-020-00855-y

[7] Zhang C, Zhang R, Zhu Z, Song X, Su Y, Li G, *et al*. Bottom-hole pressure prediction based on hybrid neural networks and Bayesian optimization. Pet Sci. 2023. https://doi.org/10.1016/j.petsci.2023.07.009

[8] Yue P, Yang H, He C, Yu GM, Sheng JJ, Guo ZL, *et al*. Theoretical approach for the calculation of the pressure drop in a multibranch horizontal well with variable mass transfer. ACS Omega. 2020; 5(45): 29209-21. https://doi.org/10.1021/acsomega.0c03971

[9] Osman EA, Ayoub MA, Aggour MA. Artificial neural network model for predicting bottom-hole flowing pressure in vertical multiphase flow. SPE Middle East Oil and Gas Show and Conference, Kingdom of Bahrain: SPE; 12-15 March 2005. https://doi.org/43.1098/11262-MS

[10] Mohammadpoor M, Shahbazi K, Torabi F, Qazvini AR. A new methodology for prediction of bottomhole flowing pressure in vertical multiphase flow in Iranian oil fields using Artificial Neural Networks (ANNs). SPE Latin American and Caribbean Petroleum Engineering Conference, Lima, Peru: SPE; 1-3 December 2010. https://doi.org/10.2418/1009-MS

[11] Pastorek N, Young KR, Eustes A. Downhole sensors in drilling operations. Proceedings of the 44th Workshop on Geothermal Reservoir Engineering Stanford University, Stanford, California: 11-13 February 2019, pp. 1-9.

[12] Ashena R, Moghadasi J, Ghalambor A, Bataee M, Ashena R, Feghhi A. Neural networks in BHCP prediction performed much better than mechanistic models. International Oil and Gas Conference and Exhibition, Beijing, China, 8-10 June 2010. https://doi.org/10.2118/130095-MS

[13] Lee J, Wattenbarger RA. Gas reservoir engineering. Richardson, Texas, USA: SPE; 1996; p. 5. https://doi.org/10.2118/9781555630737

[14] Poettmann FH. The calculation of pressure drop in the flow of natural gas through pipe. Trans AIME. 1951; 192: 100-13.

[15] Cullender MH, Smith RV. Practical solution of gas flow equations for wells and pipelines with large temperature gradients. Trans AMIE. 1956; 207: 281-7.

[16] Ikoku CU. Natural Gas Engineering. Tulsa, Oklahoma: PennWell Publishing Company; 1980, pp. 317-55.

[17] Ikoku CU. Natural Gas Reservoir Engineering. Malabar, Florida: Krieger Publishing Company, 1992.

[18] Brar GS, Aziz K. Analysis of modified isochronal tests to predict the stabilized deliverability potential of gas wells without using stabilized flow data. J Pet Technol. 1978; 30(2): 297-304. https://doi.org/10.2118/6134-PA

[19] Ikoku CU. Natural gas production engineering. New York: John Wiley & Sons; 1984.

[20] Feng Q, Gao S, Zhang J, Ye X. Well bottom-hole flowing pressure evaluation method in 48 Block of S Gas Field. Paper presented at the 5th International Conference on Energy and Environmental Protection, Shenzhen, China, 17-18 September 2016.

[21] Sukkar YK, Cornell D. Direct calculation of bottom-hole pressures in natural gas wells. Trans AIME. 1955; 204: 43-8.

[22] Cullender MH, Smith RV. Practical Solution of Gas flow equations for wells and pipelines with large temperature gradients. Trans AIME. 1956; 207: 281-7.

[23] Barjouei HS, Ghorbani H, Mohamadian N, Wood DA, Davoodi S, Moghadasi J, *et al*. Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. J Pet Explor Prod. 2021; 11: 1233-61. https://doi.org/10.1007/s13202-021-01087-4

[24] Okon AN, Ansa IB. Artificial neural network models for reservoir-aquifer dimensionless variables: influx and pressure prediction for water influx calculation. J Pet Explor Prod Technol. 2021; 11(4): 1885-1904. https://doi.org/10.1007/s13202-021-01148-8

[25] Okon AN, Effiong AJ, Daniel DD. Explicit neural network-based models for bubble point pressure and oil formation volume factor prediction. Arab J Sci Eng. 2023; 48: 9221-57. https://doi.org/10.1007/s13369-022-07240-3

[26] Mahmoudi S, Mahmoudi A. Water saturation and porosity prediction using back-propagation artificial neural network (BPANN) from well log data. J Eng Technol. 2014; 5(2): 1-8.

[27] Song S, Wu M, Qi J, Wu H, Kang Q, Shi B, *et al*. An intelligent data-driven model for virtual flow meters in oil and gas development. Chem Eng Res Design. 2022; 186: 398-406. https://doi.org/10.1016/j.cherd.2022.08.016

[28] Al-Bulushi N, King PR, Blunt MJ, Kraaijveld M. Development of artificial neural network models for predicting water saturation and fluid distribution. J Pet Sci Eng. 2009; 68: 197-208. https://doi.org/10.1016/j.petrol.2009.06.017

[29] Tugwell KW, Livinus A. Predictive models for oil in place for oil rim reservoirs in the Niger Delta using machine learning approach. Pet Petrochem Eng J. 2023; 7(3): 1-16. https://doi.org/10.23880/ppej-16000361

[30] Sircar A, Yadav K, Rayavarapu K, Bist N, Oza H. Application of machine learning and artificial intelligence in oil and gas industry. Pet Res. 2021; 6: 379-91. https://doi.org/10.1016/j.ptlrs.2021.05.009

[31] Effiong AJ, Etim JO, Okon AN. Artificial intelligence model for predicting formation damage in oil and gas wells. SPE Nigeria Council 45th Nigeria Annual International Conference and Exhibition, 2-4 August 2021. http://dx.doi.org/10.2118/207129-MS

[32] El-Bouchefry K, de Souza RS. Learning in big data: Introduction to machine learning. In: Skoda P, Adam F, Eds. Knowledge discovery in big data from astronomy and earth observation. Elsevier; 2020; pp. 225-49. https://doi.org/10.1016/B978-0-12-819154-5.00023-0

[33] Taye MM. Understanding of machine learning with deep learning: architectures, workflow, applications, and future directions. Computers. 2023; 12(5): 91. https://doi.org/10.3390/computers12050091

Appendix

Maximum-minimum normalisation method

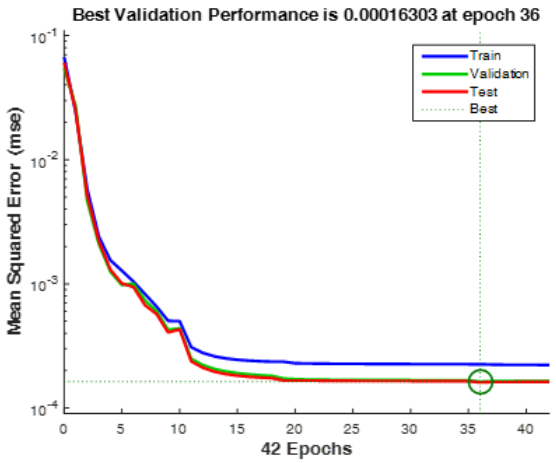

Figure A1: Topology 2-3-1 learning performance

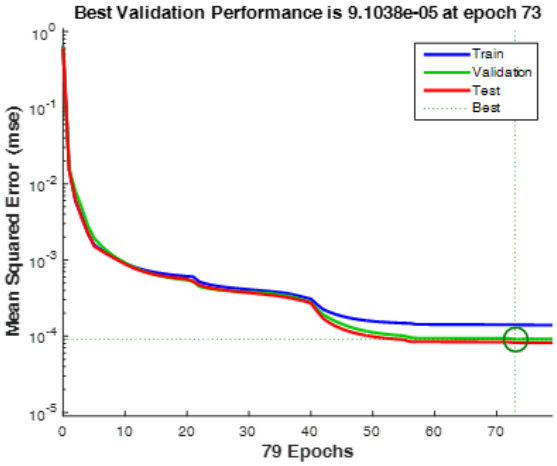

Figure A2: Topology 2-4-1 learning performance

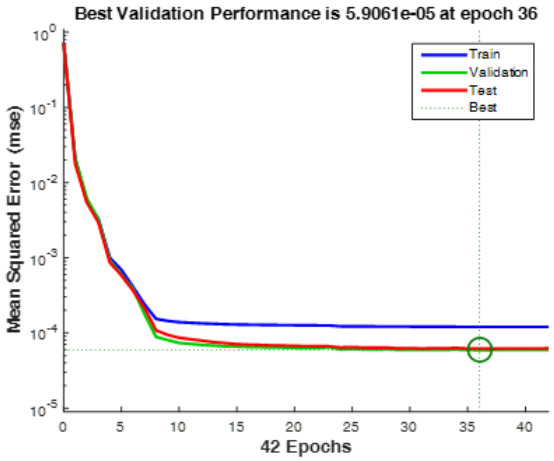

Figure A3: Topology 2-5-1 learning performance

Clip-scaling method

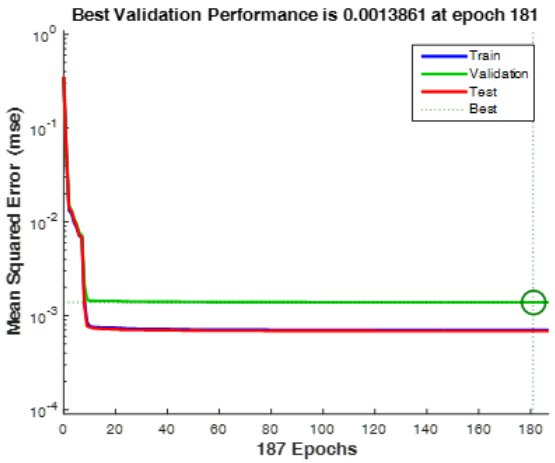

Figure A4: Topology 2-3-1 learning performance

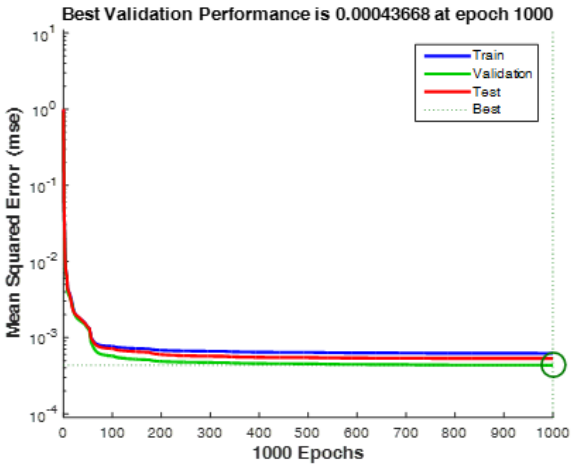

Figure A5: Topology 2-4-1 learning performance

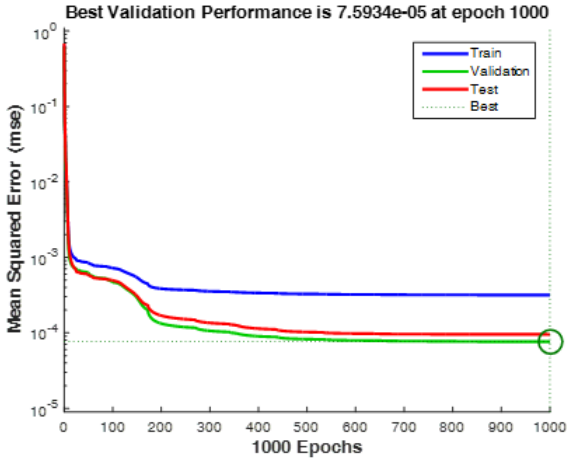

Figure A6: Topology 2-5-1 learning performance

Regression plots of the various neural networks training, testing, and validation, as well as the overall performance.

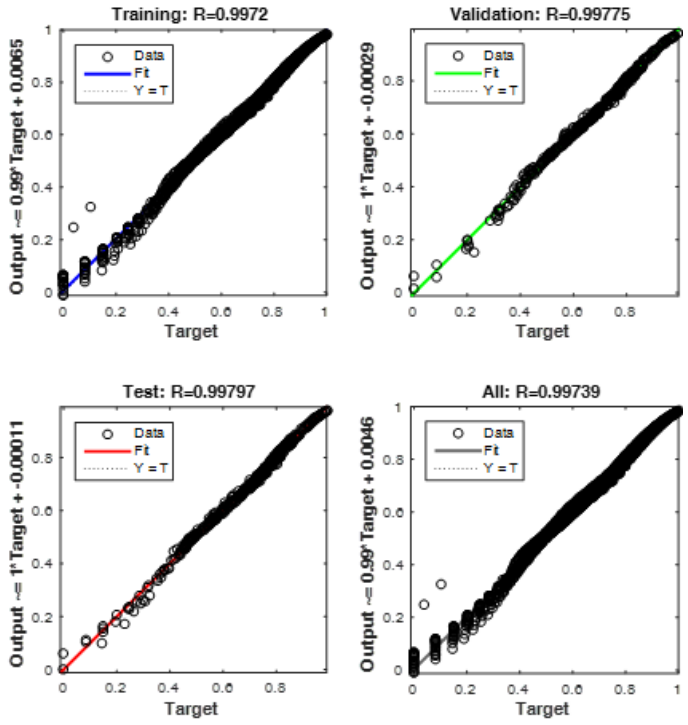

Figure B1: Regression plot of 2-3-1 network

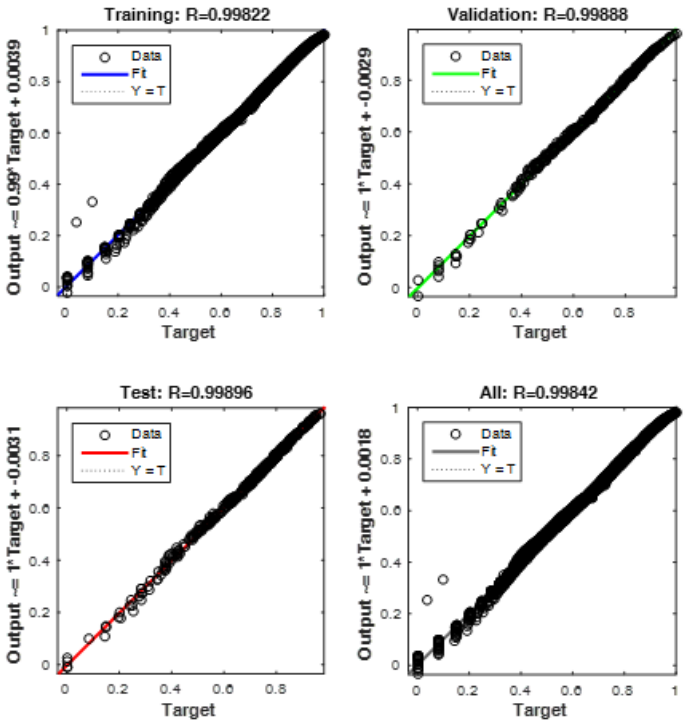

Figure B2: Regression plot of 2-4-1 network

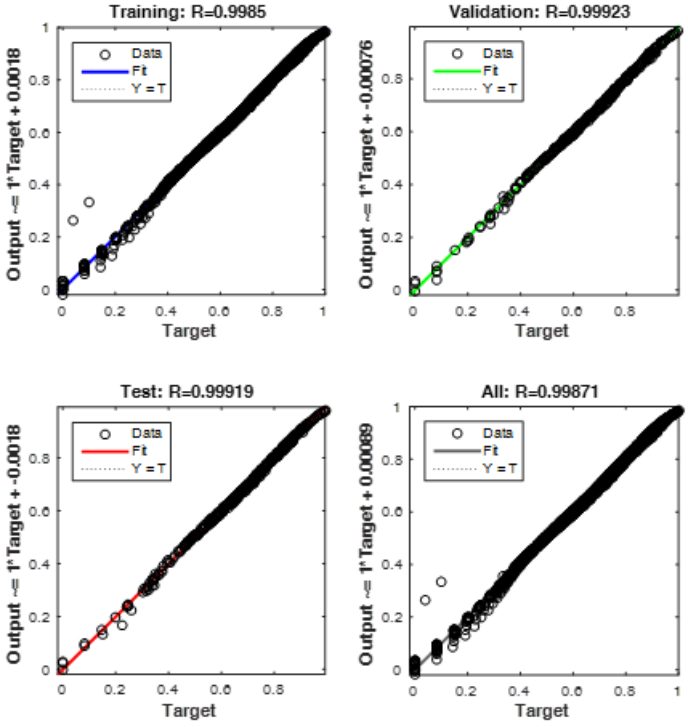

Figure B3: Regression plot of 2-5-1 network

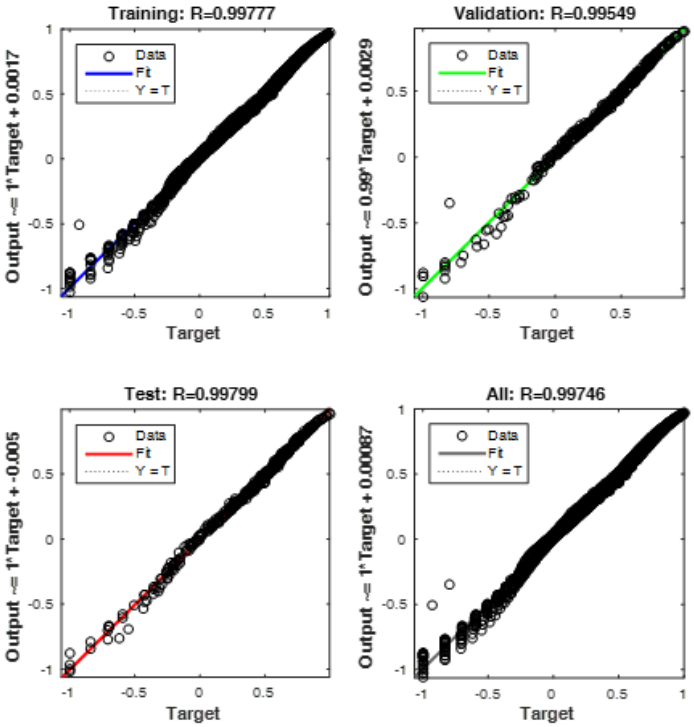

Figure B4: Regression plot of 2-3-1 network

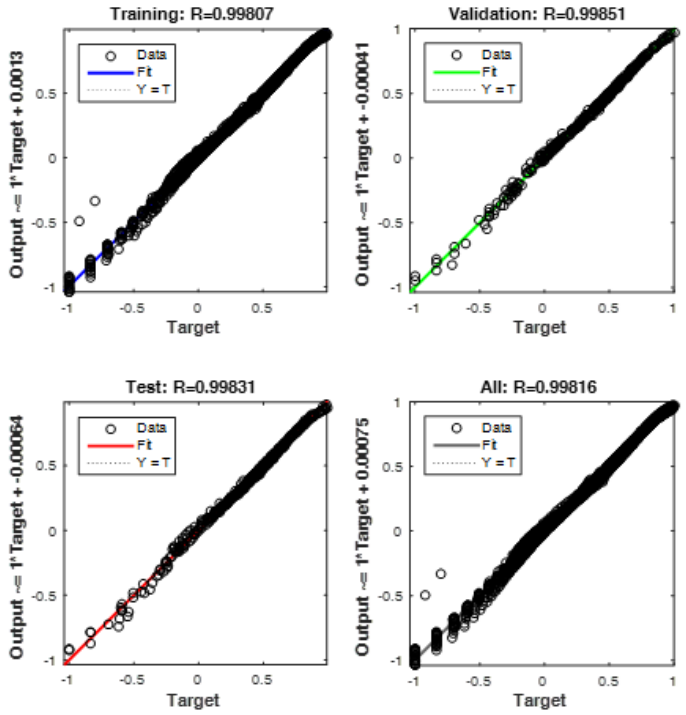

Figure B5: Regression plot of 2-4-1 network

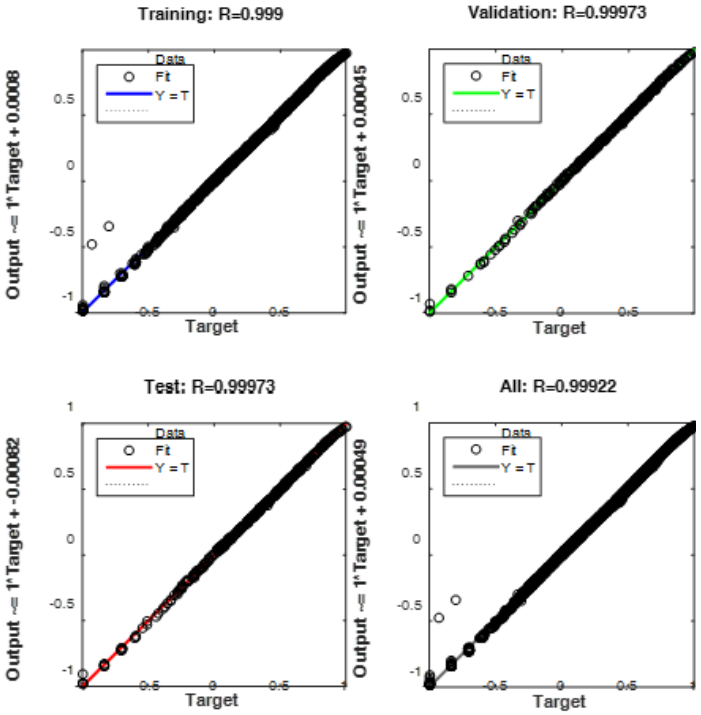

Figure B6: Regression plot of 2-5-1 network