International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

MODELING AND PREDICTION OF YEAR-BASED CHEMICAL ENGINEERING PLANT COST INDEX DATA USING FEEDFORWARD BACKPROPAGATION NEURAL NETWORK (FFBPNN) AND REGRESSION

Selvaraju Sivamani[1*], Kanakasabai Panchanathan[1], Umareddy Meka[1], Akhilesh Kumar Mishra[2], Saikat Banerjee[1], Naveen Prasad, B.S. [1], Raja Thiruvengadam[1]

[1]-Department of Engineering and Technology, University of Technology and Applied Sciences, Salalah, Oman

[2]Centre for Foundation Studies, Gulf College, Muscat, Oman

*Author for correspondence: sivmansel@gmail.com

Abstract:

This study presents a comparative analysis of Single Input Single Output (SISO) Feedforward Backpropagation Neural Network (FFBPNN) and conventional regression models for predicting year-based Chemical Engineering Plant Cost Index (CEPCI) data spanning 64 years from 1960 to 2023. The primary aim is to evaluate the predictive performance of both approaches using the coefficient of determination (R^2) and mean squared error (MSE) as key performance indicators. In the regression modelling phase, a variety of functions, linear, polynomial (up to sixth order), logarithmic, exponential, logistic, and power models, were examined. The best regression performance was obtained for hexic polynomial (n=6), yielding an R^2 of 0.9760, indicating strong, but not optimal predictive accuracy. In contrast, the FFBPNN architecture demonstrated superior performance with a model tuned by varying the data split ratio, number of neurons in the hidden layer, and activation functions for hidden and output layers. The best configuration was found to be data split ratio of 0.90 (training) - 0.05 (validation) - 0.05 (testing), 30 neurons in the hidden layer, tansig activation in the hidden layer, and purelin in the output layer. The training was carried out using the Levenberg-Marquardt algorithm with gradient descent with momentum-based learning over 10 epochs. This configuration achieved an R^2 of 0.9997 and an MSE of 10.80, significantly outperforming the regression models. The findings confirm that FFBPNN provides a more robust and accurate framework for modelling complex, nonlinear trends in year-based CEPCI data, offering substantial advantages over traditional regression methods in both precision and generalizability.

Keywords: Artificial neural networks, Nonlinear system modelling, Time series prediction, Model parameter optimization, Performance evaluation metrics

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Introduction:

Cost indices are essential tools in chemical engineering for estimating the cost of equipment, processes, or entire plants at different times. They provide a means to adjust historical cost data to present-day values by accounting for inflation, market dynamics, material costs, labour rates, and technological advancements [1,2]. Commonly used in feasibility studies, project evaluation, and cost estimation, these indices allow engineers to extrapolate or update costs without redoing full economic analyses [3]. In chemical engineering, cost indices are especially valuable for scaling equipment costs from past projects, adjusting capital investment estimates, and performing quick economic comparisons over time [4]. Several indices exist, such as the Marshall and Swift Index, Nelson-Farrar Refinery Index, and Chemical Engineering Plant Cost Index (CEPCI), each tailored for specific sectors or types of plants [5].

The CEPCI is one of the most widely used cost indices in the chemical process industries. Published monthly by Chemical Engineering magazine [6]. It tracks the relative changes in plant construction costs over time, offering a composite measure based on factors such as equipment (process, piping, instrumentation), construction labour, engineering and supervision, buildings, and materials [7]. CEPCI is normalized to a reference year (commonly 1957–1959 = 100), and it allows engineers to update capital costs from historical estimates to current values through the ratio, as given in Equation (1) [5]:

$$Updated\ Cost = Original\ Cost \times \frac{CEPCI\ (base\ year)}{CEPCI\ (current)} \quad (1)$$

CEPCI is particularly valuable because it is applicable across many types of chemical plants, regularly updated and accessible, and based on real-world economic and industrial data [8]. Its long historical record makes it suitable for trend analysis and predictive modelling, as demonstrated in studies involving machine learning and regression techniques.

Modelling and prediction are fundamental components of engineering and scientific analysis, enabling the understanding, simulation, and forecasting of real-world systems [9]. Modelling involves developing a mathematical or computational representation of a physical process or system based on underlying principles, empirical data, or both. These models can be deterministic or stochastic, linear, or nonlinear, and range from simple equations to complex algorithms such as machine learning models [10,11]. Prediction through modelling refers to the use of these models to estimate future behaviour or unknown outcomes based on input data [12]. In engineering, especially, predictive modelling supports decision-making, design optimization, and risk assessment. Depending on the system complexity, techniques may include statistical regression, differential equations, artificial neural networks, or hybrid approaches [13,14]. The accuracy of predictions relies heavily on model selection, training quality, input data, and evaluation metrics such as the coefficient of determination (R^2) and mean squared error (MSE) [15]. In recent years, data-driven models, particularly neural networks, have gained prominence for their ability to capture nonlinear patterns and make highly accurate predictions in complex systems [16].

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Regression is a fundamental statistical technique used to model the relationship between a dependent variable and one or more independent variables. It is commonly employed for prediction and trend analysis [17,18]. Linear regression assumes a straight-line relationship, while more advanced forms such as polynomial, exponential, or logarithmic regression and capture nonlinear trends. However, traditional regression models often struggle with capturing complex, highly nonlinear relationships in real-world data [19,20]. Address such limitations, artificial neural networks (ANNs) have emerged as powerful alternatives.

Inspired by the structure of the human brain, ANNs consist of interconnected layers of neurons (nodes) capable of learning patterns from data without explicitly being programmed with rules [21]. ANNs are particularly effective for modelling nonlinear and dynamic systems, making them suitable for predictive tasks across various domains. There are several types of ANN architectures, including feedforward neural networks (FNN), the simplest form where data moves in one direction from input to output, radial basis function networks (RBFN), used for function approximation and classification, recurrent neural networks (RNN), capable of handling sequential or time-dependent data due to feedback connections, convolutional neural networks (CNN), primarily used in image and spatial data processing, modular and hybrid networks, combinations designed for specialized tasks. Among these, the Feedforward Backpropagation Neural Network (FFBPNN) is one of the most widely used and well-established architectures [22-24]. In FFBPNN, information flows in one direction from input to output while the backpropagation algorithm adjusts the weights during training to minimize prediction error. This architecture is particularly effective for function approximation, time series prediction, and pattern recognition, especially when dealing with historical or year-indexed data [25,26].

FBPNNs have been widely applied across diverse engineering fields due to their ability to model complex, nonlinear relationships [27]. In the domain of hydrological modelling, Samantaray and Sahoo (2020) compared the performance of BPNN, FFBPNN, and CFBPNN algorithms in predicting rainfall-runoff behaviour in an arid watershed. Although BPNN slightly outperformed FFBPNN in terms of accuracy, the latter still showed strong prediction capabilities with an R value of 0.9925 in training and 0.9611 in testing phases, underscoring its potential for runoff prediction tasks [28]. Similarly, sediment transport modelling benefited from FFBPNN in the work by Rahul et al. (2021), where it outperformed Support Vector Machines (SVM) in predicting suspended sediment concentration (SSC) in the Ganga River. Here, FFBPNN demonstrated higher precision with a validation R=0.955 and Nash–Sutcliffe Efficiency (NSE) of 0.912, indicating its suitability for water resource management [29].

Expanding to combustion engineering, Lalmi et al. (2024) applied FFBPNN to predict swirling flow characteristics within a combustion chamber. The model effectively replicated spatial and velocity profiles of the vortex flow field and exhibited strong generalization ability, validating its application in energy systems involving complex fluid dynamics [30]. In the environmental domain, Hosseinzadeh et al. (2018) employed FFBPNN to model the efficiency of non-thermal plasma for removing BTEX pollutants from waste gases. Among the tested ANN variants and

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Response Surface Methodology (RSM), FFBPNN yielded the highest accuracy, achieving an R2R2 of 0.9736 and thus confirming its superiority in multi-variable environmental modelling [31]. Likewise, Li et al. (2016) utilized FFBPNN to predict adsorption performance in Rotating Packed Beds (RPB). With optimal topology and high prediction accuracy, the model outperformed both alternative ANN structures and nonlinear regression models, emphasizing its usefulness in process optimization in chemical separation technologies [32].

In the context of agricultural forecasting, Balaji and Vairavan (2015) implemented FFBPNN to predict rice production in Tamil Nadu. By evaluating multiple statistical error metrics, they concluded that the Absolute Relative Error (ARE) was the most effective indicator for minimizing prediction errors, thereby validating the reliability of FFBPNN in food security planning [33]. Finally, Wajahat et al. (2018) applied FFBPNN and LRNN to rainfall-runoff modelling in the Barak Basin. The FFBPNN, with a 3-9-1 architecture using a log-sigmoid transfer function, yielded superior performance metrics, reinforcing its utility in hydrological and disaster management systems [34].

The reviewed studies collectively highlight the versatile applications of regression analysis across various engineering domains. Begin with, Banerjee et al. (2025) applied multiple linear regression to analyse pressure drop in gas-solid fluidized beds, focusing on how variables like bed height and gas speed influence key response factors such as drag coefficient and power consumption. The regression model effectively clarified the relationship between multiple variables and helped isolate their individual effects [35]. Similarly, Sivamani et al. (2023) explored how regression modelling can predict mass density and specific volume in aqueous surfactant solutions (SLS and CTAB). By fitting different polynomial and exponential models, they found that the quintic model provided the best fit, with near-unity R^2 and negligible SSE, highlighting regression's capability in physicochemical data analysis [36].

In a related approach, Sivamani et al. (2020) used curve fitting regression to model and optimize Vickers hardness in laser cladding of Inconel 625. The regression-based model facilitated process parameter optimization using derivative and heuristic techniques, emphasizing regression’s role in manufacturing process optimization [37]. Moreover, Saravanaraj et al. (2022) demonstrated the applicability of empirical regression in economics by modelling the electricity consumer price index and inflation trends in Dhofar, Oman. A variety of models were evaluated, with the quintic model again yielding the highest accuracy based on the coefficient of determination (R^2), signifying regression’s adaptability even in macroeconomic forecasting [38].

While FFBPNNs have proven effective in modelling nonlinear systems across various fields, and regression models remain popular for their simplicity and interpretability, few studies compare these methods specifically for long-term forecasting of the CEPCI. CEPCI data exhibits complex, nonlinear trends over decades, challenging conventional modelling. Existing research lacks a direct, thorough comparison of FFBPNNs and regression models on CEPCI prediction, limiting guidance on the most accurate and reliable approach. Addressing this gap

Received: July 26, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

will improve cost estimation practices in chemical engineering economic analyses. Hence, this study is novel in providing a comprehensive comparative analysis between FFBPNNs and multiple regression models specifically for long-term prediction of the CEPCI over 64 years. Unlike prior research, which focuses on individual modelling techniques or other engineering data, this work rigorously optimizes and evaluates FFBPNN architectures alongside diverse regression functions to determine the superior method for capturing complex nonlinear trends in CEPCI. The findings offer new insights into enhancing cost estimation accuracy and robustness in chemical engineering economic forecasting.

From the research gap and novelty statement, the present study aims to develop and evaluate predictive models using FFBPNN and conventional regression techniques for accurate forecasting of the CEPCI over a 64-year period. The objectives are as follows: (i) To compile year-based CEPCI data from 1960 to 2023 for modelling purposes; (ii) To construct and optimize various regression models (linear, polynomial, logarithmic, exponential, logistic, power) for CEPCI prediction; (iii) To design and fine-tune FFBPNN architectures by varying data split ratios, neuron numbers, and activation functions of hidden and output layers; (iv) To compare the predictive performance of regression models and FFBPNNs using R^2 and MSE metrics; and (v) To identify the most accurate and robust modelling approach for long-term CEPCI forecasting.

Methods:

Collection of CEPCI data:

The CEPCI data used in this study were sourced from the monthly publications of Chemical Engineering magazine, which has been consistently releasing updated cost indices since the mid-20th century. This comprehensive dataset, spanning from 1960 to 2023, reflects industry-standard cost factors including equipment, labour, materials, and construction, making it a dependable and widely accepted reference for cost estimation and economic analysis in the chemical process industries [39].

Regression analysis:

The CEPCI data was verified to ensure consistency and readiness for analysis. Regression analysis was conducted in Microsoft Excel. Various regression models, including linear, polynomial (quadratic, cubic, quartic, quintic and hexic), logarithmic, exponential, power, and logistic functions, as given in Equations (2)-(11), were selected and fitted to the historical data. Each model's performance was evaluated using the coefficient of determination (R^2) and mean squared error (MSE) to assess goodness of fit and prediction accuracy [38]. The models were compared to identify the best-performing regression function, which was then validated and used for CEPCI prediction.

Linear model: $y = \alpha_0 + \alpha_1 x$ (2)

Quadratic model: $y = \alpha_0 + \alpha_1 x + \alpha_2 x^2$ (3)

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Cubic model: $y = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$ (4)

Quartic model: $y = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4$ (5)

Quintic model: $y = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4 + \alpha_5 x^5$ (6)

Hexic model: $y = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4 + \alpha_5 x^5 + \alpha_6 x^6$ (7)

Logarithmic model: $y = a_1 . \ln(x) + b_1$ (8)

Exponential model: $y = a_2 e^{b_2 x}$ (9)

Power model: $y = a_3 x^{b_3}$ (10)

Logistic model: $y = \frac{a_4}{e^{-b_4(x-x_0)}}$ (11)

where y is the CEPCI, x is year, α_0 is intercept, α_1 is linear coefficient, α_2 is quadratic coefficient. α_3 is cubic coefficient, α_4 is quartic coefficient, α_5 is quintic coefficient, α_6 is hexic coefficient, a_1 is slope, b_1 is intercept, a_2 is value of y at x=0, b_2 is inclining rate, a_3 is scale factor, b_3 is exponent, a_4 is maximum value of y, b_4 is steepness, and x_0 is inflection point.

FFBPNN modelling:

MATLAB served as the primary platform for conducting the analysis in this study. The FFBPNN modelling procedure for CEPCI prediction began with organizing the dataset into input (year) and output (CEPCI index) variables. The working principle of FFBPNN involves two main phases: forward propagation and backpropagation. During forward propagation, the input passes through the network layer by layer, where each neuron applies a weighted sum followed by an activation function to produce an output. The network output is compared with the target CEPCI value, and the error is calculated [40]. In the backpropagation phase, this error is propagated backward through the network using gradient descent to update the weights and biases, minimizing the overall prediction error. This iterative learning continues until the network reaches a satisfactory level of performance. The trained FFBPNN model is then tested on unseen data to validate its generalization capability, demonstrating superior predictive accuracy for modelling complex, nonlinear CEPCI trends [9].

The data was normalized to improve training efficiency and then split into three subsets: training (50-90%), validation (5-25%), and testing (5-25%). The FFBPNN architecture was designed with a single input neuron (year), a hidden layer with varying neurons (5-35), and a single output neuron (predicted CEPCI value). The activation functions used for the hidden and the output layers are tansig, logsig, purelin and poslin at various combinations. The network was trained and learned using the Levenberg-Marquardt (LM) and gradient descent with momentum (GDM) algorithms, known for its fast convergence and high accuracy, over ten epochs.

Received: July 26, 2025

International Journal of Applied Mathematics
Volume 38 No. 3s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Performance metrics:

The coefficient of determination (R^2) and mean squared error (MSE) are two key performance metrics used to evaluate the accuracy of predictive models [41]. R^2 indicates the proportion of variance in the dependent variable that is predictable from the independent variable(s). It ranges from 0 to 1, where a value closer to one signifies a better fit, meaning the model explains most of the variability in the data. It is calculated using Equation (12).

$$\%R^2 = \left(1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}\right) \times 100 \quad (12)$$

where y_i is observed value, $\hat{y}_i$ is predicted value, $\bar{y}$ is mean of observed values, and n is number of data points.

MSE measures the average of the squares of the prediction errors, i.e., the average squared difference between predicted and actual values. Lower MSE values indicate higher accuracy and better model performance. It is calculated using Equation (13).

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n} \quad (13)$$

Results And Discussion:

CEPCI data:

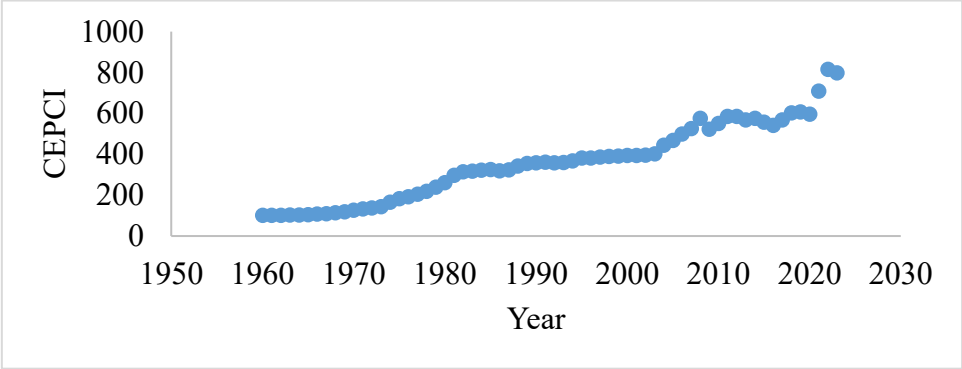

Figure 1. CEPCI data from 1960 to 2023

Figure 1 illustrates the trend of the CEPCI from 1960 to 2023. A gradual increase is observed from 1960 to the late 1970s, followed by a more rapid rise during the early 1980s. From the mid-1980s through the early 2000s, the CEPCI exhibits a relatively stable or moderately increasing trend. A significant escalation is evident after 2005, with noticeable fluctuations around the 2008 financial crisis. From 2010 onward, the index continues to rise, with sharp increases post-2020 likely reflecting economic recovery efforts and inflationary pressures following the COVID-19 pandemic. Overall, the trend indicates long-term growth with intermittent volatility.

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Regression modelling:

Table 1. Models developed for relationship between the CEPCI and year

Model	R^2	MSE
Linear model: $y = -19222 + 9.8324x$	95.41	1565.46
Quadratic model: $y = 154858 - 165.01x + 0.0439x^2$	95.93	1388.17
Cubic model: $y = -20000000 + 257666x - 12.978x^2 + 0.0022x^3$	96.26	1275.66
Quartic model: $y = 4000000000 - 7000000x + 5379.7x^2 - 1.8031x^3 + 0.0002x^4$	97.21	951.78
Quintic model: $y = 100000000000 - 300000000x + 249933x^2 - 124.61x^3 + 0.0311x^4 - 0.000003x^5$	97.25	938.14
Hexic model: $y = 30000000000000 - 100000000000x + 100000000x^2 - 85175x^3 + 32.063x^4 - 0.0064x^5 + 0.0000005x^6$	97.60	818.81
Logarithmic model: $y = 19574\ln(x) - 148337$	95.35	1585.91
Exponential model: $y = 4 \times e^{0.0322x}$	93.07	2363.24
Power model: $y = 4 \times 10^{-210}x^{64.24}$	93.23	2308.69
Logistic model: $y = \frac{499.3058}{e^{-0.005474(x-x_0)}}$	92.18	2666.67

Table 1 summarizes the performance of various regression models, linear, polynomial (from quadratic to hexic), and nonlinear (logarithmic, exponential, power, and logistic), applied to model the CEPCI data over time. Each model's equation is provided along with the coefficient of determination (R^2) and mean squared error (MSE), which collectively indicate the model's fit and prediction accuracy.

Among all models, the hexic (sixth-degree polynomial) model demonstrated the best performance, with the highest R^2 value of 97.60% and the lowest MSE of 818.81, indicating a superior ability to capture the complex trends in the data. The quintic and quartic models followed closely with R^2 values of 97.25 and 97.21%, and MSE values of 938.14 and 951.78, respectively. These results support the effectiveness of higher-order polynomials in modelling the intricate behaviour of the CEPCI data. The cubic, quadratic, and linear models showed slightly lower R^2 values of 96.26, 95.93, and 95.41%, respectively, with corresponding MSEs increasing to 1275.66, 1388.17, and 1565.46. While these models still reflect strong correlations, their lower performance suggests a reduced ability to represent more nuanced or nonlinear variations, particularly in periods of rapid change.

Among the nonlinear models, the logarithmic model achieved the highest R^2 value of 95.35%, outperforming the power 93.23%, exponential .93.07%, and logistic 92.18% models. This indicates that the CEPCI data follows a growth pattern that slows over time, rather than

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

exhibiting unchecked exponential or power-law growth. Despite their lower R^2 values and higher MSEs, these nonlinear models may still offer advantages in terms of interpretability or theoretical alignment with specific physical or economic growth behaviours.

In conclusion, while simpler models provide a good approximation, higher-degree polynomial models, particularly the hexic model, most effectively capture the underlying patterns in the CEPCI data. However, selecting an appropriate model should also involve consideration of overfitting risks, model interpretability, and predictive reliability, especially when extrapolating beyond the observed data range.

ANN modelling:

Effect of neurons in a hidden layer on performance of model:

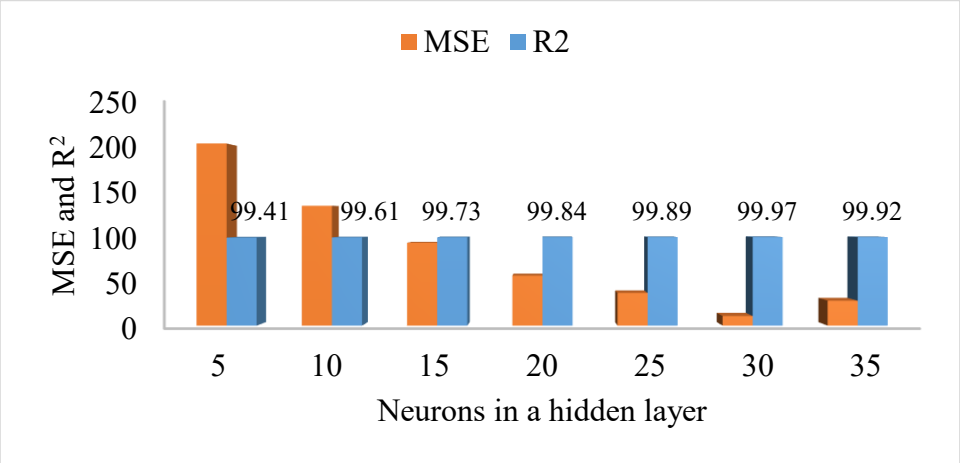

Figure 2. Effect of neurons in a hidden layer on model performance metrics (Training function: Levenberg-Marquardt algorithm, learning function: gradient descent with momentum, ten epochs, activation functions for hidden and output layers: tansig-purelin, and data split ratio (training-testing-validation):90-5-5)

Figure 2 illustrates the performance of a FFBPNN in terms of MSE and coefficient of determination (R^2) for different numbers of neurons in the hidden layer (ranging from 5 to 35). As the number of neurons increases, there is a clear improvement in model performance. Initially, with five neurons, the model shows a higher MSE of 205.67 and a lower R^2 of 99.41%, indicating a poor fit. As neurons increase to 10 and 15, MSE decreases (to 135.4 and 93.97, respectively), and R^2 rises slightly (to 99.61 and 99.73%), showing modest improvements.

Significant gains are seen between 15 and 25 neurons, where MSE drops sharply from 93.97 to 36.94, while R^2 increases to 99.89%, suggesting that the model is capturing the underlying data pattern more accurately. The optimal performance is observed around 30 neurons, where MSE reaches its minimum (10.8) and R^2 is at 99.97%, indicating near-perfect prediction capability. Beyond 30 neurons, at 35, MSE slightly increases to 28.24, though R^2 remains remarkably high at 99.92%, suggesting diminishing returns and the potential onset of overfitting.

Received: July 26, 2025

International Journal of Applied Mathematics
Volume 38 No. 3s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In summary, increasing neurons improves accuracy up to a point, with 30 neurons providing the best trade-off between minimizing error and maximizing predictive strength.

Effect of data split ratio on performance of model:

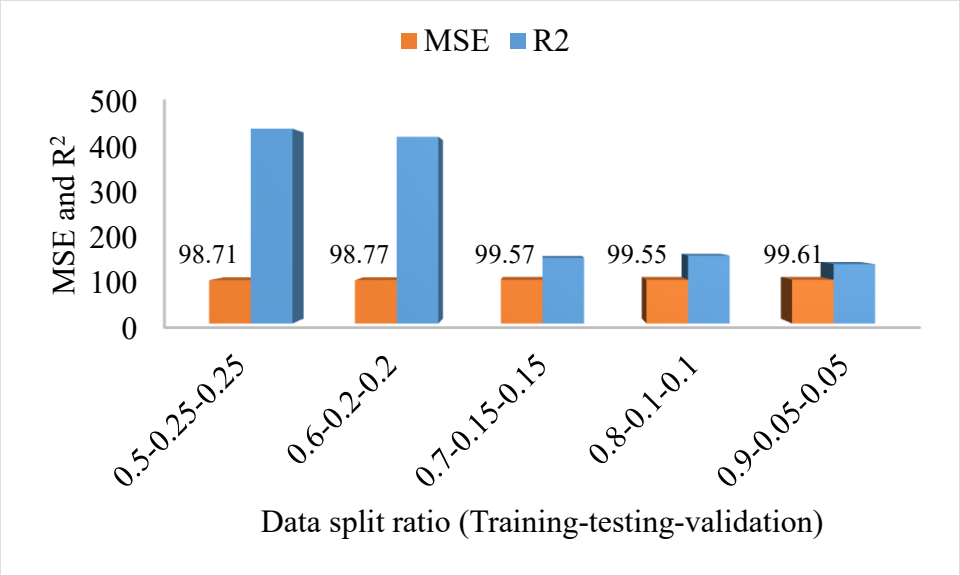

Figure 3. Effect of data split ratio (training-testing-validation) on model performance metrics (Training function: Levenberg-Marquardt algorithm, learning function: gradient descent with momentum, ten epochs, activation functions for hidden and output layers: tansig-purelin, and neurons in a hidden layer: 30)

Figure 3 illustrates the impact of different data split ratios (for training, testing, and validation) on the MSE and R^2 in a FFBPNN model. As the training data proportion increases from 50% to 90%, the MSE gradually increases from 98.71 to 99.61, while R^2 shows a slight decline from around 426 (in scaled units) at the 0.5–0.25–0.25 split to about 99.61 at the 0.9–0.05–0.05 split. The highest R^2 appears at the 0.5–0.25–0.25 and 0.6–0.2–0.2 splits, indicating strong generalization when more data is reserved for testing and validation. However, as the training data increases beyond 70%, the model may slightly overfit, as reflected in the plateauing of R^2 and marginal MSE differences. The differences among MSE values across the splits are minor, indicating stable performance across various data distributions.

In summary, the chart clearly shows that allocating 90% of data for training, with 5% each for testing and validation, results in the most optimal performance of the FFBPNN model. This split achieves the lowest Mean Squared Error (MSE) and the highest coefficient of determination (R^2), indicating excellent model accuracy and generalization. This suggests that a larger training set enables the neural network to learn the underlying patterns more effectively, while still preserving sufficient testing and validation data to ensure robust performance evaluation.

Effect of activation functions for hidden and output layers on performance of model:

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 2. Effect of activation functions of hidden and output layers on performance of model (Training function: Levenberg-Marquardt algorithm, learning function: gradient descent with momentum, ten epochs, data split ratio (training-testing-validation):90-5-5 and neurons in a hidden layer: 30)

Activation function of layer		R^2	MSE
Hidden	Output		
tansig	tansig	99.90	35.51
logsig	tansig	99.91	31.66
purelin	tansig	93.86	2122.70
poslin	tansig	98.67	458.47
tansig	purelin	99.97	10.80
logsig	purelin	99.82	61.94
purelin	purelin	94.74	1817.35
poslin	purelin	97.57	860.78
tansig	logsig	-9.33	37803.22
logsig	logsig	-6.75	36911.07
purelin	logsig	-9.36	37813.43
poslin	logsig	-9.67	37923.71
tansig	poslin	-8.51	37521.90
logsig	poslin	-6.87	36952.37
purelin	poslin	-9.05	37706.52
poslin	poslin	-8.87	37644.98

The activation function in the output layer plays a pivotal role in determining the range and nature of the model's predicted output, which directly affects performance metrics like R^2 and MSE. Unlike the hidden layer, which primarily introduces nonlinearity and facilitates complex feature extraction, the output layer governs how the final output is mapped, whether it remains bounded, unbounded, linear, or nonlinear. In the Table 2, activation functions such as purelin (linear) in the output layer—when paired with suitable hidden layer functions like tansig enabled accurate modeling of continuous target values, as reflected in the highest R^2 (99.97%) and lowest MSE (10.80). This suggests that for regression-type problems, linear output activation (purelin) is often the most appropriate choice, as it does not restrict the output range and allows the network to predict any real number.

Received: July 26, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Conversely, the use of nonlinear activations like logsig, and poslin, in the output layer led to significant performance degradation, with some combinations yielding negative R^2 values and extremely high MSEs. These nonlinear functions tend to saturate the output to specific ranges ([0,1] for logsig, or [-1,1] for tansig), which may hinder the model's ability to approximate target values outside those ranges. This mismatch between the activation function's output domain and the data's scale result in poor prediction accuracy. Therefore, choosing an appropriate output layer activation function that aligns with the data distribution and problem type, especially for regression, is crucial for optimal model performance. The empirical findings reaffirm that purelin is best suited for continuous output regression tasks, while nonlinear functions in the output layer may severely limit the network's predictive power in such contexts.

Conclusion:

The comparative analysis between regression and FFBPNN modeling for CEPCI data prediction reveals significant insights. The regression analysis demonstrated that higher-order polynomial models, particularly the hexic (6^{th} degree) model with an R^2 of 97.60%, offer superior fitting accuracy over simpler linear or nonlinear models. This suggests that increasing model complexity can effectively capture nuanced trends in the data. However, the risk of overfitting and reduced generalization must be carefully considered. In contrast, the FFBPNN model achieved even greater predictive performance, with an optimal configuration reaching R^2 of 99.97 and a remarkably low MSE of 10.8. The neural network's flexibility, especially when tuned with appropriate training algorithms, learning functions, data split ratios, and activation functions (notably tansig in the hidden layer and purelin in the output), enabled it to model highly nonlinear patterns more effectively than regression models. Unlike polynomial regression, FFBPNN balances high accuracy with better generalization, making it a more robust choice for dynamic and complex datasets like CEPCI. Thus, while polynomial regression, especially up to the hexic model, offers a strong analytical approach, FFBPNN proves to be the most effective modeling technique, capable of capturing both the structure and variability of CEPCI data with exceptional precision.

References:

1) Mignard, D. (2014). Correlating the chemical engineering plant cost index with macro-economic indicators. *Chemical Engineering Research and Design*, *92*(2), 285-294.
2) Vatavuk, W. M. (2002). Updating the CE plant cost index. *Chemical Engineering*, *109*(1), 62-70.
3) Lemmens, S. (2016). Cost engineering techniques and their applicability for cost estimation of organic Rankine cycle systems. *Energies*, *9*(7), 485.
4) Gama, V. R. V. (2021). CostApp: a cost estimation tool developed using C# and WPF for the chemical engineering field.
5) Peters, M. S., Timmerhaus, K. D., & West, R. E. (2003). *Plant design and economics for chemical engineers* (Vol. 4). New York: McGraw-hill.

Received: July 26, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

6) Pintelon, L., & Geeroms, K. (1997). Computational model for a Belgian chemical engineering plant cost index. *International journal of production economics*, *49*(2), 101-115.
7) Earl, P. H. (1977). Specifying escalation indexes to aid in process plant cost forecasting. *Engineering and Process Economics*, *2*(4), 281-294.
8) Towler, G., & Sinnott, R. (2021). *Chemical engineering design: principles, practice and economics of plant and process design*. Butterworth-Heinemann.
9) Sivamani, S., Prasad, B. N., Nithya, K., Sivarajasekar, N., & Hosseini-Bandegharaei, A. (2022). Back-propagation neural network: Box–Behnken design modelling for optimization of copper adsorption on orange zest biochar. *International Journal of Environmental Science and Technology*, *19*(5), 4321-4336.
10) Seel, N. M. (2017). Model-based learning: A synthesis of theory and research. *Educational Technology Research and Development*, *65*, 931-966.
11) Mosavi, A., Ozturk, P., & Chau, K. W. (2018). Flood prediction using machine learning models: Literature review. *Water*, *10*(11), 1536.
12) Jahed Armaghani, D., Hasanipanah, M., Mahdiyar, A., Abd Majid, M. Z., Bakhshandeh Amnieh, H., & Tahir, M. M. (2018). Airblast prediction through a hybrid genetic algorithm-ANN model. *Neural Computing and Applications*, *29*, 619-629.
13) Technow, F., Messina, C. D., Totir, L. R., & Cooper, M. (2015). Integrating crop growth models with whole genome prediction through approximate Bayesian computation. *PloS one*, *10*(6), e0130855.
14) Lee, Y. H., Bang, H., & Kim, D. J. (2016). How to establish clinical prediction models. *Endocrinology and Metabolism*, *31*(1), 38.
15) Lin, C. P., Cabrera, J., Yang, F., Ling, M. H., Tsui, K. L., & Bae, S. J. (2020). Battery state of health modeling and remaining useful life prediction through time series model. *Applied Energy*, *275*, 115338.
16) Wang, Z., Hong, T., & Piette, M. A. (2020). Building thermal load prediction through shallow machine learning and deep learning. *Applied Energy*, *263*, 114683.
17) Sarstedt, M., Mooi, E., Sarstedt, M., & Mooi, E. (2019). Regression analysis. *A concise guide to market research: The process, data, and methods using IBM SPSS Statistics*, 209-256.
18) Wald, A. (1947). A note on regression analysis. *The Annals of Mathematical Statistics*, *18*(4), 586-589.
19) Alexopoulos, E. C. (2010). Introduction to multivariate regression analysis. *Hippokratia*, *14*(Suppl 1), 23.
20) Tyagi, K., Rane, C., & Manry, M. (2022). Regression analysis. In *Artificial intelligence and machine learning for EDGE computing* (pp. 53-63). Academic Press.
21) Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, *22*(5), 717-727.
22) Yin, C., Rosendahl, L., & Luo, Z. (2003). Methods to improve prediction performance of ANN models. *Simulation Modelling Practice and Theory*, *11*(3-4), 211-222.

Received: July 26, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

23) Ingre, B., & Yadav, A. (2015, January). Performance analysis of NSL-KDD dataset using ANN. In *2015 international conference on signal processing and communication engineering systems* (pp. 92-96). IEEE.
24) Kshirsagar, P., & Akojwar, D. S. (2015). Classification and Prediction of Epilepsy using FFBPNN with PSO. In *IEEE international conference on communication networks* (Vol. 17).
25) Priyadarshini, K., Alagarsamy, M., Sangeetha, K., & Thangaraju, D. (2024). Hybrid RNN-FFBPNN Optimized with Glowworm Swarm Algorithm for Lung Cancer Prediction. *IETE Journal of Research*, *70*(5), 4453-4468.
26) Saravanaselvan, A., & Paramasivan, B. (2024). FFBP neural network optimized with woodpecker mating algorithm for dynamic cluster-based secure routing in WSN. *IETE Journal of Research*, *70*(7), 6515-6524.
27) Makinde, F. A., Ako, C. T., Orodu, O. D., & Asuquo, I. U. (2012). Prediction of crude oil viscosity using feed-forward back-propagation neural network (FFBPNN). *Petroleum & Coal*, *54*(2).
28) Samantaray, S., & Sahoo, A. (2020). Prediction of runoff using BPNN, FFBPNN, CFBPNN algorithm in arid watershed: a case study. *International Journal of Knowledge-Based and Intelligent Engineering Systems*, *24*(3), 243-251.
29) Rahul, A. K., Shivhare, N., Kumar, S., Dwivedi, S. B., & Dikshit, P. K. S. (2021). Modelling of daily suspended sediment concentration using FFBPNN and SVM algorithms. Journal of Soft Computing in Civil Engineering, 5(2), 120-134.
30) Lalmi, D., Kifouche, A., Kaddour, A., Hadef, R., & Khodja, K. (2024). Application of FFBP neural network to the prediction of swirling flow. *Brazilian Journal of Technology*, *7*(4), e73336-e73336.\
31) Hosseinzadeh, A., Najafpoor, A. A., Jafari, A. J., Jazani, R. K., Baziar, M., Bargozin, H., & Piranloo, F. G. (2018). Application of response surface methodology and artificial neural network modeling to assess non-thermal plasma efficiency in simultaneous removal of BTEX from waste gases: Effect of operating parameters and prediction performance. *Process Safety and Environmental Protection*, *119*, 261-270.
32) Li, W., Wei, S., Jiao, W., Qi, G., & Liu, Y. (2016). Modelling of adsorption in rotating packed bed using artificial neural networks (ANN). *Chemical engineering research and design*, *114*, 89-95.
33) Balaji, S. A., & Vairavan, P. M. (2015). Effect of different statistical measures in error reduction in Feed Forward Back Propagation Neural Network (FFBPNN) to predict rice production. *International Journal of Applied Engineering Research*, *10*(24), 44570-44578.
34) Wajahat, A., Sil, S. B., & Ajay, G. (2018). Development of Rainfall-Runoff Model Using FFBPNN and LRNN for Silchar City—A Case Study. *Disaster Adv*, *11*, 19-23.
35) Banerjee, S., Sivamani, S., Prasad, N., & Gajendiran, V. (2025). Multiple Linear Regression Analysis for Prediction of Pressure Drop in Gas-Solid Fluidized Bed.
36) Sivamani, S., Ehilla, J. M., Banerjee, S., & Vijayanand, M. (2023). Regression modelling for concentration dependent mass density and specific volume of aqueous surfactant solutions. *Bulletin of the Chemical Society of Ethiopia*, *37*(4), 1047-1054.

Received: July 26, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

37) Sivamani, S., Vijayanand, M., Bala, A. U., & Varahamoorthi, R. (2020). Process Modelling and Optimization of Hardness in Laser Cladding of Inconel® 625 Powder on AISI 304 Stainless Steel. In *Green Materials and Advanced Manufacturing Technology* (pp. 137-160). CRC Press.
38) Saravanaraj, K. J., Gerardo, L., & Sivamani, S. (2022). Empirical modelling of consumer price index of electricity and percentage inflation for Dhofar governorate in Sultanate of Oman. *Asian Journal of Applied Business and Management*, *1*(1), 53-62.
39) Jenkins, S. (2023). Changes to Calculation Inputs for CEPCI Due to Discontinued Data Series. *Chemical Engineering*, *130*(7).
40) Selvaraju, S., Rajoo, B., & Banerjee, S. (2024, March). Artificial Intelligence based Neural Network Modelling of Bioethanol Production from Cassava Peel. In *2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies* (pp. 1-4). IEEE.
41) Serefoglu Cabuk, K., Cengiz, S. K., Guler, M. G., Topcu, H., Cetin Efe, A., Ulas, M. G., & Poslu Karademir, F. (2024). Chasing the objective upper eyelid symmetry formula; R^2, RMSE, POC, MAE, and MSE. *International Ophthalmology*, *44*(1), 303.

Received: July 26, 2025