مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

مقارنة بين أداء الشبكة العصبية إلمان قبل وبعد استخدام تحويل الموجات

هاوناز حسن صديق
جامعة السليمانية ،كلية الإدارة والاقتصاد
hawnaz.sidiq@univsul.edu.iq

ا.م.د محمد محمود فقى حسين
جامعة السليمانية ،كلية الإدارة والاقتصاد
Mohammad.faqe@univsul.edu.iq

المستخلص

يعد إنتاج النفط في العراق ركيزة أساسية لاقتصاد ، كما أنه يلعب دورًا رئيسيًا في أسواق النفط العالمية. ووفقًا لأحدث البيانات، يبلغ إنتاج البلاد نحو 4 ملايين برميل يوميًا، بعد تعديله ليتماشى مع لوائح أوبك +. وقد شهد الإنتاج تقلبات تاريخية نتيجة لعوامل داخلية، مثل تطوير البنية التحتية، والتحديات السياسية، والاتفاقيات الدولية. وخلال الفترة من 1992 إلى 2022، سجل إنتاج النفط في العراق نموًا ملحوظًا، رغم فترات التراجع التي صاحبت الصراعات أو التغيرات في السوق العالمية. وتتمتع البلاد باحتياطيات نفطية ضخمة، وتسعى باستمرار إلى توسيع قدراتها الإنتاجية من خلال الشراكات والاستثمارات في الحقول النفطية الرئيسية. يعتمد التحليل على المقارنه بين نموذج الشبكة العصبية إلمان قبل و بعد استخدام تحويل المويجات لتفسير البيانات، حيث تركز الدراسة على تطبيق تحويلات الموجات بهدف تحسين أداء النموذج، مع التركيز على تقليل متوسط مربعات الخطأ (MSE) تم استخدام موجات دوبشيز عند المستويين 1 و2، وأظهرت النتائج تفوق موجة دوبشيز من المستوى 2 في تقليل متوسط مربعات الخطأ مقارنة بغيرها. تؤكد هذه النتائج فعالية تحويلات الموجات، ولا سيما دوبشيز من المستوى 2، في تحسين دقة التنبؤات، مما يبرز أهمية الأساليب المعتمدة على الموجات في تحسين مقاييس الأداء.

الكلمات المفتاحية :السلاسل الزمنية، الموجات، الشبكات العصبية إلمان، متوسط الخطأ التربيعي

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 492- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

A Comparison of Elman Neural Network Performance Before and After Using Wavelet Transform

Hawnaz Hassan Sdiq
University of Sulaimani, College of Administration and Economics
hawnaz.sidiq@univsul.edu.iq

Assistant Prof Dr. Mohammad Mahmood Faqe Hussein
University of Sulaimani, College of Administration and Economics
Mohammad.faqe@univsul.edu.iq

Abstract

Iraq's oil production is a critical component of its economy and a major contributor to global oil markets. As of recent data, the country produces approximately 4 million barrels per day (bpd), having adjusted its output to align with OPEC+ regulations. Historically, Iraq's production has fluctuated, influenced by internal factors like infrastructure development, political challenges, and global agreements. Over the period from 1992 to 2022, Iraq's oil production has demonstrated significant growth, with occasional dips during periods of conflict or global market adjustments. The country possesses vast reserves and has consistently worked on expanding its production capacity through partnerships and investments in key oil fields ELA HOMEPAGE.The analysis is based on comparison between the Elman Neural Network before and after using wavelete transform to interpret the data. This study investigates the application of wavelet transforms in optimizing model performance, focusing specifically on reducing the mean square error (MSE). The analysis employed Daubechies wavelets at levels 1 and 2, with results demonstrating that the level 2 Daubechies wavelet outperformed others in minimizing MSE. The findings underscore the efficacy of wavelet transforms, particularly level 2 Daubechies, in enhancing the predictive accuracy of the model. These results highlight the significance of wavelet-based approaches in achieving improved performance metrics.

Keywords: Time Series, Wavelet, Elman Neural Networks, MSE

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

1. Introduction:

Time series analysis is widely used across several areas due to its ability to use the natural temporal structure of data via the sequential presentation of observations. An example of an approach that is often used in time series analysis is the Box-Jenkins ARMA (p,q) model. This model combines autoregressive (AR) and moving average (MA) components. Important techniques such as the partial autocorrelation function (PACF) and the autocorrelation function (ACF) may help you discover temporal patterns and seasonal affects in the data.[5][15]

Wavelet analysis provides a powerful framework for studying data across multiple scales by decomposing it into frequency-based components localized in both time and frequency domains. Unlike Fourier transforms, wavelets are particularly adept at handling non-stationary signals, enabling the detection of localized patterns. This makes them especially suitable for dynamic and irregular datasets, with applications in signal processing and time series analysis.[14][17]

The Elman Neural Network (ENN), introduced in 1990, enhances the Backpropagation (BP) framework by incorporating a "context layer" that retains historical information for subsequent iterations. This dynamic memory feature allows ENNs to effectively leverage temporal context and recognize sequential patterns, making them highly applicable to tasks involving time dependent data.This research aims to assess the Elman Neural Network's (ENN) performance by contrasting its accuracy and efficiency before to and after the implementation of Wavelet Transform (WT). The study's overarching goal is to determine if and by what margin WT improves

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

the network's feature extraction capabilities, therefore the accuracy of forecasts and classifications. The objective is to find out whether the ENN's performance is significantly enhanced across different datasets when WT is included.[7][18]

2. Material and Methods:

2.1Time Series:

A time series is a set that may be the measurements taken at regular or irregular intervals.In mathematics, a collection of vectors $\{\ (b), b \in T\}$ is defined, where the index parameter (T) denotes the time-space or the set of observations indexed by time. A random variable is the one that is denoted by (b).[3][4]

2.1.2Types of Time Series Data:

2.1.2.1 Stationary Time Series:

Stationary time series are crucial to time series analysis, serving as the foundation for model development and forecasting. Non-stationary time series are more commonly used in practical applications, particularly in business, economics, and industry, where processes frequently display non-stationary behavior. While stationary data is optimal for analysis, real-world settings typically include non-stationary datasets that necessitate particular processing.[1]

In many circumstances, non-stationary data can be turned into a stationary process using the differencing method, represented as:

$$\nabla Zt = Z_t - Z_{t-1}, \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Once this transformation is completed, it becomes possible to model the changes, anticipate future values of these changes, and generate projections for the original Non-stationary Time Series.

2.1.2.2 Non-Stationary Time Series :

The non-stationarity of a time series is a property of linear and nonlinear systems that causes the behavior of the system to fluctuate over time. The presence of mean shifts and fluctuations in process variance, together with a trend that varies over time, indicates non-stationarity (Kan, Tan, and Mathew, 2015). When a time series $\{Z_t\}$ becomes stationary after differencing, it is said to be homogeneous non-stationary. This may also be said as.[2][4]

$$Wt = Z_t - Z_{t-1} = (1-B)Z_t \ldots\ldots\ldots..(2)$$

or for higher-order differences,

$$Wt = (1 - B)^d Z_t, \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots....(3)$$

Non -Stationary around mean: when the mean of a series is not constant.

Non Stationary around Variance: Many time series are non-stationary because of their time-dependent variances and autocovariance, variance that changes through time.

2.2 Wavelet concept:

Wavelets are tiny waves that may be coupled together to create larger and more complicated waves. A comprehensive wavelet system that can capture

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

any wave type may be built by modifying and expanding basic wavelets. To build an orthonormal wavelet foundation in wavelet analysis, one uses the mother wavelet and the scaling function: the father wavelet. Because of its poor resolution, the Short-Time Fourier Transform is problematic when working with non-stationary events. To get over this constraint, however, the wavelet transform gives representations in both the frequency and temporal domains.[6][8][10]

2.2.1Wavelet Properties:

The key properties of wavelets are outlined as follows:[10][13]

1. The fact that the wavelet Ψ(.) function's zero average exists over the interval (-∞,+∞) implies that the waveform is oscillating and has to be situated in space.

$$\int_{-\infty}^{+\infty} \Psi(\mathrm{x})\mathrm{dx} = 0 \qquad \text{.....................(4)}$$

2. The square of the wavelet Ψ(.) function integrates into one, assuring compactness or limited length.

$$\int_{-\infty}^{+\infty} \Psi^2(\mathrm{x})\mathrm{dx} = 1$$

.......................(5)

3. The wavelet function's in-vanishing moments enable exact representation of signals using a finite sum.

$$\int_{-\infty}^{+\infty} x^k \Psi(x)dx = 0 \qquad \text{k= 0,1,.........N}$$

...................(6)

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

4. The scaling function's integral across one interval (-∞,+∞) equals one, standardizing the area under the function to unity.

$$\int_{-\infty}^{\infty} \emptyset(x)dx = 1$$
…………………………..(7)

2.2.2 Wavelet Types: The types of wavelets which have gained popularity throughout the development of wavelet analysis:

2.2.2.1 Discrete Wavelets:

There is a wide variety of discrete wavelets available, and each one has its own set of uses: [12][13]

1. Haar wavelet
2. Coiflet (6,12,18,24,30).
3. Daubechies wavelet (2, 4, 6, 8, 10, 12, 14, 16, 18, 20).

2.2.2.3 Discrete Wavelet Transform:

Discrete Wavelet Transform is an incredibly flexible tool for signal processing that finds applications in a wide range of fields, from computer science and engineering to mathematics and physics. To create a multiresolution representation of the signal, DWT breaks it down into the shifted and scaled copies of the compactly supported basis function, which is called the mother wavelet.

Given a signal vector with N observations (where N is an integer), the DWT may be expressed as:[8][11]

$$Z = WX$$ ……………….(8)

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

Z is a vector with N dimensions that includes wavelet coefficients and discontinuous scaling. Additional vectors may be constructed from these wavelet coefficients:

$$Z = [CD_j, CA_j] \quad ....(9)$$

In this context, $CD_j$is a Nj-long vector that represents the wavelet coefficients (details) connected to variations on a 2^j, scale, and CA_j is aN_j-long vector that represents the scaling coefficients (approximations or smooth components) linked to the average over a 2^j scale.The orthonormal N×N matrix W corresponds to the orthonormal wavelet basis that has been selected.

2.2.3.1 Haar wavelet:

The Haar wavelet, introduced by Alfred Haar in 1909, is recognized as the first and simplest wavelet. Its straightforward nature makes it a common starting point for beginners in wavelet theory. For N = 2, deriving the Haar wavelet involves satisfying three conditions: stability (${h_0}^2 + {h_1}^2$=1), accuracy $h_0 + h_1$=$\sqrt{2}$, and orthogonality $h_0 h_1$=0,These constraints yield a unique solution: $h_0 = 1$ and $h_1 = -1$,This leads to defining the scaling function as a step function.[6][10]

2.2.3.2 Coiflet(6,12,18,24,30):

Wavelets with compact support and higher-order vanishing moments are known as coiflets, which were created by Ingrid Daubechies. N denotes the quantity of vanishing moments for the scaling (ϕ) and wavelet (ψ) functions, and is shown as Coiflet(N). Coiflet(6,12,18,24,30) and its variants cater to a wide range of signal processing requirements by varying the filter length and vanishing moments. These wavelets strike a good compromise between

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 499- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

temporal and frequency representation; longer filters are necessary for greater N, but better frequency localization is achieved. Coiflets' effectiveness in evaluating data from both areas makes them popular.[11]

2.2.3.3 Daubechies wavelet (2, 4, 6, 8, 10, 12, 14, 16, 18, 20):

In 1988 and 1992, Daubechies introduced two types of compactly supported wavelets, each defined by a particular level of smoothness. These are referred to as Daubechies' maximally phase and minimally symmetric wavelets, constructed based on the standard condition described in the referenced equation.[6][10]

$$Z_0 + Z_1 + Z_2 + Z_3 = 2 \quad \ldots\ldots\ldots\ldots\ldots\ldots(10)$$

$$Z_0 - Z_1 + Z_2 - Z_3 = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots(11)$$

$$-Z_1 + 2Z_2 - 3Z_3 = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots(12)$$

$$Z_0 Z_2 + Z_1 Z_3 = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots(13)$$

$$Z^2{}_0 + Z^2{}_1 + Z^2{}_2 + Z^2{}_3 = 2 \quad \ldots\ldots\ldots\ldots\ldots\ldots(14)$$

The unique solution for the equation is exists:

$$Z_0 = \frac{1+\sqrt{3}}{4} \qquad Z_1 = \frac{3+\sqrt{3}}{4} \quad \ldots\ldots\ldots\ldots(15)$$

$$Z_2 = \frac{3-\sqrt{3}}{4} \qquad Z_3 = \frac{1-\sqrt{3}}{4} \quad \ldots\ldots(16)$$

2.3 Concept of Neural Network :

Neural networks and artificial intelligence working together to simulate how the brain processes and understands data. The method is called deep learning, and it uses a hierarchical network of interconnected neurons to simulate the way the human brain works. The result of this method is an adaptive system, which gives computers the ability to learn and advance over time. As a result, artificial neural networks are very adept at challenging tasks such as document summarization and picture identification.[9][16]

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

2.3.1 Supervised Learning Algorithms: The training of a model in supervised learning algorithms often involves the usage of labeled datasets. By studying the patterns in the examples given, these algorithms learn to convert incoming data into the results you want. Once trained, the model may use previously unknown data to generate predictions or judgments.

2.3.2 Reinforcement Learning Algorithms: Reinforcement learning is a subfield of machine learning algorithms that aims to train agents to adapt their decision-making to their environment. The agent learns from its actions and makes adjustments to its strategy to maximize cumulative rewards over time. It receives feedback in the form of rewards or penalties.

2.3.3 Unsupervised Learning Algorithms: Unsupervised learning algorithms seek to discover structures, correlations, or patterns in unlabeled data without being guided or instructed to produce certain results. The algorithms in question constitute a subset of MLP.

2.3.4 Elman Neural Network (ENN):

In 1990, Elman created a kind of RNN known as the Elman Neural Network. The buried layer's feedback loops allow it to receive input from both the current and prior instants. When dealing with problems involving discrete time series, ENN is useful because of the pattern-learning abilities and its ability to imitate nonlinear and dynamic systems. Backpropagation is one of the most well-known supervised learning methods. The ENN gains its self-referencing, repeating feature and is able to preserve its past outputs via the context layer.[7][16]

2.3.5 The Structure of Elman Neural Network:

Elman Neural Networks (ENNs) are a kind of layered neural network that looks like a classic multi-layer architecture. Most importantly, this

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

architecture incorporates a context layer that stores and recycles the results of buried layers. With this information, the ENN can better manage patterns and sequences that change over time thanks to its temporal memory.[7]

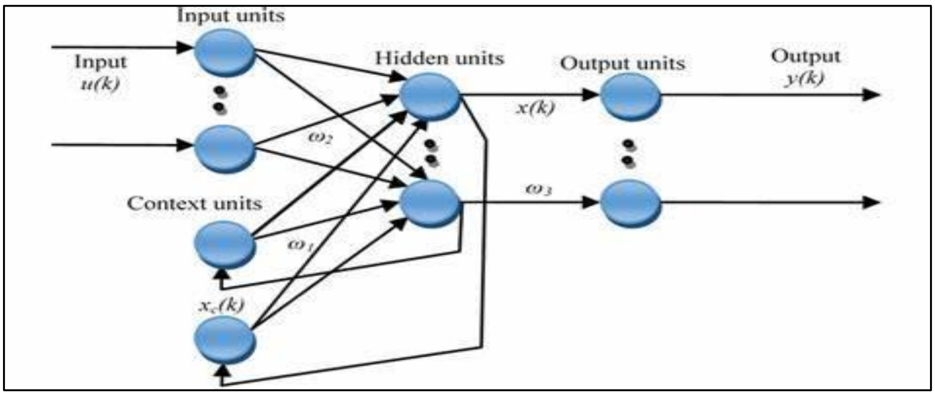


Fig 2.1The Structure of Elman Neural Network

There are a few different ways to train Elman networks: The weight matrices $w^{h,i}(t)$, $w^{h,c}(t)$, $w^{o,h}(t)$ reflect the external input, context, and output, respectively. The sentence reads as follows: "The sentence has an external input vector $x^1(t) = [\, x_1^1(t)\ , x_2^1(t)\ , \ldots\ldots\ldots\ldots x_n^1(t)]^T$, and an output vector $z_{(t)} = [z_1(t)\ , Z_2(t)\ , \ldots\ldots\ldots\ Z_n(t)]^T$, Based on the above information, the hidden neuron count is $w^{h,i}(t) \in R^{m\times n}$, $w^{h,c}(t) \in R^{m\times m}$, $w^{o,h}(t) \in R^{n\times m}$.

The hidden layer's output vector, denoted as c(t-1), is equal to the set $c(t-1) = [c_1(t-1)\ , c_2(t-1), \ldots c_m(t-1)]^T$

Returned to the buried layer as an additional input vector, is defined as the whole input vector. $x_{(t)} = [\ x_1^1(t)\ , x_2^1(t)\ , \ \ldots\ldots\ldots\ldots x_n^1(t)\ , x_{n+1}^2(t), \ldots\ldots\ldots x_k^2(t)]^T$

$[[\, x_{(t)}^2]^T\ [x_{(t)}^1]^T]^T = [\, x_1^1(t)\ , x_2^1(t)\ , \ldots.\, x_n^1(t)\ , c_1^2(t-1), \ldots c_m^2(t-1)]^T$ ………(17)

where $k = m + n$

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

The activation function of the output layer takes the sigmoid function The output vector can be computed by equations:

The activation function of the output layer takes the sigmoid function The output vector can be computed by equations:

$$Z_{i(t)} = f\left(a_i^0(t)\right) = \frac{1}{1+\exp\left(-a_i^0(t)\right)} \quad , \quad i= 1,2,\dots\dots\dots n$$

………………..(18)

$$a_i^0(t) = \sum_{j=1}^{m} w_{ji}^{o,h}(t) * h(t), \qquad i = 1,2,3,\dots\dots n$$

…………………..(19)

For relationships among the input layer, the context layer, and the hidden layer, define the complete input weight matrix as:

$$w^h(t) = [w^{h,i}(t), w^{h,c}(t)] \in R^{m\times k}$$

So the output of the complete input vector, the activation function of the hidden layer takes the sigmoid function:

$$h_j(t) = f\left(a_j^h(t)\right) = \frac{1}{1+\exp(-a_j^{o,h}(t))} \ , \ j = 1,2,\dots\dots\dots m$$

…..…………..…(20)

$$a_j^{o,h}(t) = \sum_{i=1}^{k} W_{jl}^h(t) * x_l(t) \ , j = 1,2,\dots\dots\dots\dots m$$

……………….(21)

The target of the ENN training algorithm is to minimize the mean-square error

$$MSE(t) = \frac{\|e(t)\|^2}{2}$$

$$e(t) = d(t) - Z(t)$$

Here, are the desired outputs.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

$w^{o,h}(t+1) = w^{o,h}(t) - \mu \frac{\partial E(t)}{\partial w^{o,h}(t)} = w^{o,h}(t) + \mu y'(t)e(t)h^T(t)$

……(22)

$w^{h}(t+1) = w^{h}(t) - \mu \frac{\partial E(t)}{\partial w^{h}(t)} = w^{h}(t) + \mu h'(t)[\, w^{o,h}(t)$

$]^T z'(t)e(t)x^T(t)$ … ….(23)

Here, μ is the learning rate of the EBP, and

$z'(t) = diag[f'(a_1^0(t))f'(a_2^0(t), \dots \dots \dots \dots . f'(a_n^0(t))] \in R^{n*n}$

………..…(24)

$h'(t) = diag[f'\left(a_1^h(t)\right) f'\left(a_2^h(t), \dots \dots . f'\left(a_m^h(t)\right)\right] \in R^{m*m}$

……..….(25)

Elman networks can be trained following occurs at each epoch [18]: The external input, context, and output weight matrix are represented as $w^{h,i}(t)$, $w^{h,c}(t)$, $w^{o,h}(t)$ respectively.it contains a -dimensional external input vector $x^1(t) =[\, x_1^1(t), x_2^1(t), \dots\dots x_n^1(t)]^T$, and a -dimensional output vector $z_{(t)} = [z_1(t), Z_2(t), \dots\dots Z_n(t)]^T$, The number of hidden neurons is, and therefore $w^{h,i}(t) \in R^{m \times n}$, $w^{h,c}(t) \in R^{m \times m}$, $w^{o,h}(t) \in R^{n \times m}$.

The output vector of the hidden layer $c(t-1) = [c_1(t-1), c_2(t-1), \dots c_m(t-1)]^T$

connected back to the hidden layer as another input vector, so the complete input vector is defined as:

$x_{(t)} = [\, x_1^1(t), x_2^1(t), \dots\dots x_n^1(t), x_{n+1}^2(t), \dots \dots x_k^2(t)]^T =$
$[[\, x_{(t)}^2]^T [x_{(t)}^1]^T]^T = [\, x_1^1(t), x_2^1(t), \dots \dots \dots \dots . x_n^1(t), c_1^2(t-1), \dots \dots c_m^2(t-1)]^T$ …………(26)

where $k = m + n$

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

The activation function of the output layer takes the sigmoid function The output vector can be computed by equations:

$Z_{i(t)} = f\left(a_i^0(t)\right) = \frac{1}{1+\exp\left(-a_i^0(t)\right)}$, i= 1,2,………n …………..…..(27)

$a_i^0(t) = \sum_{j=1}^{m} w_{ji}^{o,h}(t) * h(t)$, $i = 1,2,3,\ldots\ldots n$ …………...….(28)

For relationships among the input layer, the context layer, and the hidden layer, define the complete input weight matrix as:

$w^h(t) = [w^{h,i}(t), w^{h,c}(t)] \in R^{m\times k}$

So the output of the complete input vector, the activation function of the hidden layer takes the sigmoid function:

$h_j(t) = f\left(a_j^h(t)\right) = \frac{1}{1+\exp(-a_j^{o,h}(t))}$, $j = 1,2,\ldots\ldots\ldots m$ …..…………..…(29)

$a_j^{o,h}(t) = \sum_{i=1}^{k} W_{jl}^h(t) * x_l(t)$, $j = 1,2,\ldots\ldots\ldots.. m$ ……………….…(30)

The target of the ENN training algorithm is to minimize the mean-square error

$MSE(t) = \frac{\|e(t)\|^2}{2}$

$e(t) = d(t) - Z(t)$

Here, are the desired outputs.

$w^{o,h}(t+1) = w^{o,h}(t) - \mu \frac{\partial E(t)}{\partial W^{o,h}(t)} =$

$w^{o,h}(t) + \mu y'(t) e(t) h^T(t)$ …….……….(31)

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

$w^{h}(t+1) = w^{h}(t) - \mu \frac{\partial E(t)}{\partial w^{h}(t)} = w^{h}(t) +$

$\mu h'(t)[w^{o,h}(t)]^{T} z'(t)e(t)x^{T}(t)$ … … . . ….(32)

Here, μ is the learning rate of the EBP, and

$z'(t) = \text{diag}[f'(a_1^0(t))f'(a_2^0(t), \ldots \ldots \ldots \ldots . f'(a_n^0(t))] \in$

R^{n*n} …………(33)

$h'(t) = \text{diag}[f'\left(a_1^h(t)\right)f'\left(a_2^h(t), \ldots \ldots . f'\left(a_m^h(t)\right)\right] \in R^{m*m}$

…………(34)

2.3.5 Training an Elman NN:

Elman networks have two training options: train and adapt. While using the train function, the following happens at each training epoch:[16]

1- The network is programmed with the whole input sequence, and then it computes the outputs, compares them to the goal sequence, and produces an error sequence.

2- Ignoring effects of delayed recurrent connections, mistakes are backpropagated at each time step to approximate weight and bias gradients.

3–After training the suggested function, the weights are updated using the estimated gradient in accordance with the user-selected backpropagation technique.

With the adapt function, the following actions are performed at each time step during training:

1-Network makes a mistake after receiving the input vectors.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

2. The weights and biases' error gradients are computed by backpropagating the error. Due to the removal of contributions from delayed recurrent connections, this is again an estimate.
3- The user's preferred learning function is utilized to update the weights via the approximation gradient; learning is the recommended function. Elman networks rely on estimating the error gradient for training and adaptation, which makes them less dependable than other varieties. Additional hidden neurons are often needed for them to outperform other learning techniques. As the number of hidden neurons increases, the likelihood of efficiently dividing the input space and finding the best weights also increases.

2.3.6 Testing an Elman Neural Network (ENN): include testing how well the trained network performed. We can evaluate the ENN's generalizability by feeding it new, unknown data and seeing how well it predicts and classifies. It is common practice to follow these steps during testing:I have read [7][9]

1. A new dataset, called the test data, is fed into the network after training. There was no use of this dataset during training.

2. Through recurrent connections, the Elman neural network sequentially analyzes inputs from the past. At each step, the context layer influences future outputs based on the hidden layers and their output.

3. Results Assessment: As a third step, the network uses the input data to make predictions or categorizations. To determine how accurate these outputs are, they are compared to the actual values in the test data.

4. Metrics for performance: In regression, metrics like as mean squared error (MSE) evaluate the network's capacity to internalize patterns and apply them

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

to fresh data. The same is evaluated in classification using measures like F1 score, recall, accuracy, and precision.

5. is to make sure the network isn't overfitting; this happens when it memorizes its training data too well and can't go beyond that. It may be possible to tell whether this is true by comparing its performance in testing and training.

3. Result and Discussions:

3.1 Data Description: We employed time series models and Elman neural networks to analyze crude oil production data spanning 372 months, from January 1992 to December 2022.The data was obtain from (https://countryeconomy.com/energy-and-environment/crude-oil/production/iraq).Used EViews14 and MATLAB 14b software for analyzing the data.

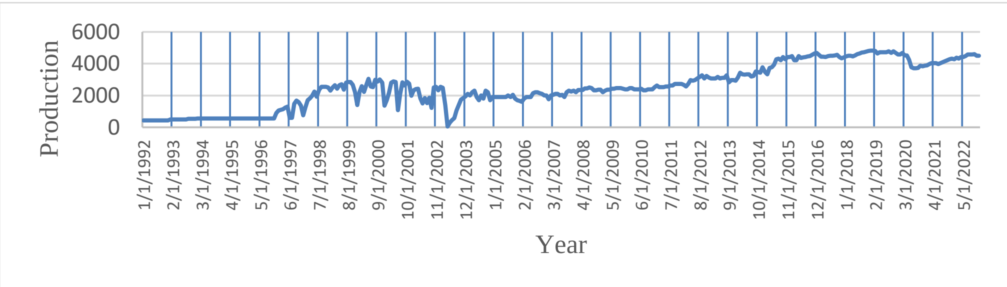


Figure 3.1 Oil Production in Iraq during the year (1992-2022)

3.2 Steps for Applying Elman Neural Network

a. Test the data for Stationary

Time series stationarity may be assessed using a number of statistical tests, one of which is the Augmented Dickey-Fuller (ADF) test. For a time series to be considered stationary, its mean, variance, and autocorrelation must all stay the same during the course of the data. Contrarily, statistical features of

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

a non-stationary series may vary over time due to cycles, trends, or other patterns

Hypothesis Test:

H_0: The time series is non-stationary because it has a unit root.

H_1: The time series is non stationary because it does not have a unit root.

Table (1): Represent the ADF Test of Oil Crude Production

Test	Test Statistic	P-Value	Critical Value
Augmented Dickey-Fuller test	-0.986985	0.7589	1% :-3.448
			5% : -2.869
			10%:-2.5709

We are unable to reject the null hypothesis and come to the conclusion that the time series is probably non-stationary because the p-value (0.7589) is higher than 0.05 and the test statistic falls below the crucial thresholds.

b. Take the first differencing to achieve stationarity in the data:

This method is widely used to transform a non-stationary time series into a stationary one. It involves subtracting the previous value in the series from the current value. By doing so, the data series becomes stationary, helping to remove trends or seasonal patterns.

First Differencing Steps:

1. Calculate the Difference: Compute the first difference using the formula $\Delta Y_t = Y_t - Y_{t-1}$ for each time step t.
2. Assess Stationarity: Since the test statistic is lower than the critical thresholds and the p-value is greater than 0.05 (0.7589), the null hypothesis cannot be rejected, indicating that the time series remains non-stationary.
3. Verify with ADF Test: After differencing, conduct the Augmented Dickey-Fuller (ADF) test to confirm whether the series has achieved

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

stationarity. The results of the first differencing test are presented in the table below

Table (2): Represent the ADF Test of Oil Crude Production After First Difference.

Test	Test Statistic	P-Value	Critical Value
Augmented Dickey-Fuller test	-9.621839	0.0000	1% :-3.448
			5% : -2.869
			10%:-2.5709

The null hypothesis is rejected since the test statistic (-9.621839) is more negative than any of the essential values and the p-value (0.0000) is exceptionally low. This provides strong evidence that the time series does not have a unit root and is stationary.

c. Using train, testing, and validating for Appling ELMAN before and after using Wavelet transform

The method would be a little different from the last example if you wanted to use an Elman Neural Network (ENN) model before executing Wavelet Transform. Rather than training the model after converting the data, you would:

i. First, separate the data into sets for testing, validation, and training.

ii. Before using the Wavelet Transform, train the Elman Neural Network.

Table (3) : Train, Testing, and Validating for Appling ELMAN before and after using Wavelet transform

(Training , Validation , Testing	Ratio	MSE	RMSE	MAE	R Training	R Validation	R Testing	R All	R^2 Validation
(%80 , %10 , %10)	1:02:01	0.0031	0.055677644	0.523	0.953	0.9845	0.984	0.9647	0.96924025
(%80 , %10 , %10)	1:02:02	0.0096	0.09797959	0.515	0.9287	0.9471	0.67517	0.8922	0.89699841
(%70 , %15 , %15)	1:02:03	0.0043	0.065574385	0.53	0.95321	0.9145	0.96439	0.9513	0.83631025
(%70 , %15 , %15)	1:02:04	0.0058	0.076157731	0.524	0.91386	0.9767	0.96086	0.9324	0.95394289

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

(%80 , %10 , %10)	1:02:05	0.0004948	0.022243426	0.523	0.99418	0.9964	0.99536	0.9944	0.99281296
(%60 , %20 , %20)	1:02:06	0.0004502	0.021217681	0.525	0.99499	0.99677	0.98996	0.995	0.993550433
(%60 , %20 , %20)	1:02:07	0.001	0.031622777	0.52	0.99591	0.99128	0.92667	0.9888	0.982636038
(%80 , %10 , %10)	1:02:08	0.002	0.04472136	0.525	0.9806	0.9722	0.9709	0.9781	0.94517284
(%70 , %15 , %15)	1:02:09	0.000493	0.022203603	0.525	0.9948	0.9959	0.9935	0.9944	0.99181681
(%60 , %20 , %20)	1:02:10	0.0016	0.04	0.527	0.98919	0.97204	0.96224	0.9817	0.944861762
(%60 , %20 , %20)	1:02:11	0.001	0.031622777	0.523	0.99008	0.99004	0.98697	0.9887	0.980179202
(%70 , %15 , %15)	1:02:12	0.0026	0.050990195	0.532	0.97923	0.96007	0.9756	0.9715	0.921734405
(%60 , %20 , %20)	1:02:13	0.0013	0.036055513	0.524	0.98754	0.97692	0.99647	0.9857	0.954372686
(%80 , %10 , %10)	1:02:14	0.000745	0.027294688	0.526	0.9982	0.99566	0.9552	0.9917	0.991338836
(%80 , %10 , %10)	1:02:15	0.0003712	0.019266551	0.524	0.99545	0.99725	0.99701	0.9958	0.994507563

The best model, based on MSE, is **Row 15**, where the training, validation, and testing, all ratio is **80%**, **10%**, and **10%** respectively. This model shows the best accuracy with the lowest MSE, the highest R², and strong overall performance across training, validation, and testing datasets.

d. The Structure of the Elman Neural Network <u>before</u> using Wavelet Transform:

A basic Elman network, in which data is fed directly into the network to learn sequential dependencies, has the same structure as an Elman Neural Network (ENN) prior to the use of Wavelet Transform. Importantly, the data is not subjected to the Wavelet Transform before to being transmitted to the network.

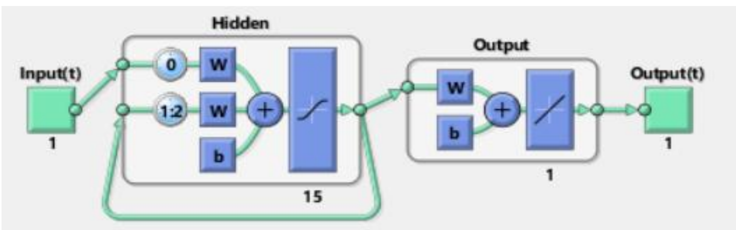


Figure 3.2: The Configuration of the Best ELMAN NN (1:2,15) for Oil Crude Production

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 511- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

[Reference: Researchers by (Matlab Software)]

The topology of an Elman neural network with certain parameters seems to be defined by the model you are referring to, Elman Neural Network [1, 15, 1, 1]. The Elman Neural Network [1, 15, 1, 1] consists of one input unit, fifteen hidden units, one output unit (single prediction), and one context unit (for remembering prior time steps). Sequential data analysis and time series forecasting are two examples of sequence-based jobs that employ this arrangement to describe dependencies across time increments.

e. The regression plot generally has four graphs showing for the best training:

The following figure demonstrates the usage of a regression plot to compare the actual data used as targets with the predicted data used as outputs. The standard format for a regression plot includes four graphs: training, validation, test, and combined. That proved our neural network and architecture is sound.

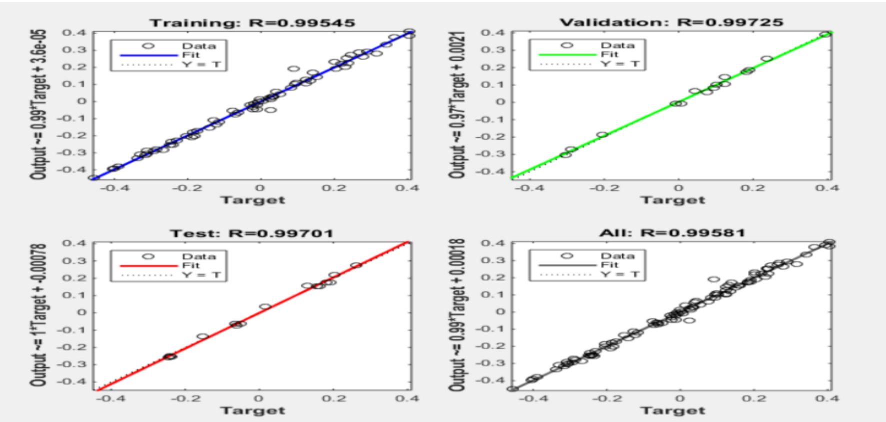


Figure 3.3: Regression Plots Displaying the Elman NN (1:2,15) for Oil Crude Production data

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

With a training R^2 score of 0.99092, the model was able to accurately explain 99.09% of the variance in the training dataset. The model fits the training data nearly perfectly when the R^2 value is high. An R^2 value of 0.9945 was used for validation purposes. To prevent overfitting and fine-tuning the model's parameters, the validation dataset is used. An R^2 value of 0.9945 indicates that the model adequately accounted for 99.45% of the variance observed in the validation data. This shows that the model has efficiently learned the patterns and can apply them to new data without being overfit. R^2 = 0.9940 on the testing dataset, which is used to evaluate the final model's performance. After fitting the data, the model correctly accounts for 99.40% of the variance, as shown by the R^2 value of 0.99701. With its close proximity to the validation R^2, the model demonstrates its consistent and reliable performance throughout the testing and validation datasets. Test, validation, and training sets' combined R^2 values are 0.9916. An R^2 of 0.9916 indicates that the model satisfactorily explains 99.16% of the overall heterogeneity in the dataset. If the number is high, it means the model is doing a good job of capturing the data's underlying patterns and generalizing effectively.

f. The time-series response plot for confirmed cases using Elman Neural Network:

Elman NN's time-series response plot for validated instances is shown in the figure below. Additionally, it reveals the exact start and end times of the training, testing, and validation processes. Elman NN (1:2,15) shown to be true in the test. The data was consistently mirrored by the model, as seen by the modest errors in the training, testing, and validation subsets and the uniformly dispersed outputs around the response curve.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 513- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

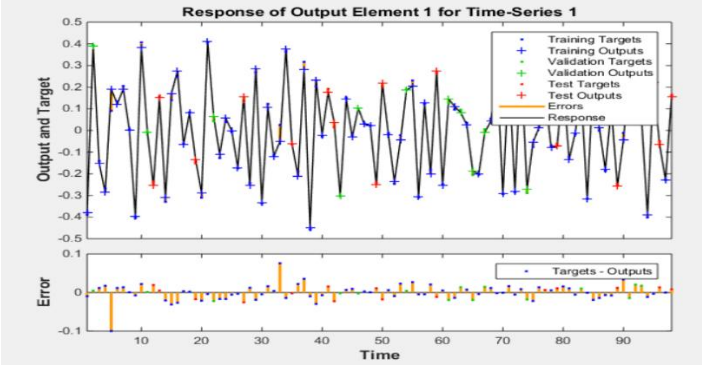
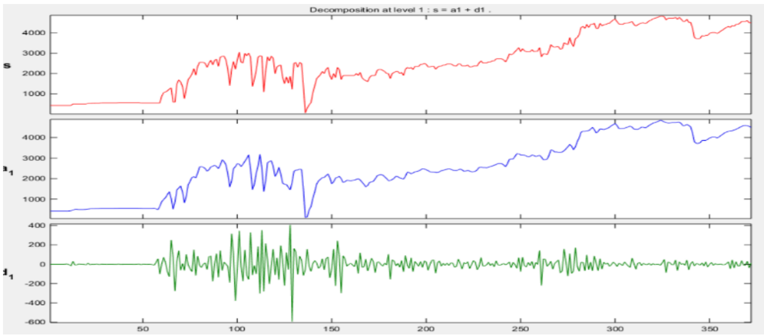


Figure 3.4: The Time-Series Response for the ELMAN NN (1:2,15) Model of Oil Crude Production

g. Using train, testing, and validating for Appling ELMAN After using (db3_1 level 1)

Construct three subsets of your data first: testing, validation, and training. To extract multi-scale features, apply wavelet transforms to the data using db3_1 (level 1). The Elman network uses the wavelet coefficients as input characteristics, which include detail and approximation components. Make sure the Elman network understands temporal dependencies from the wavelet-transformed data by training it with the training set. During training, keep an eye on the model's performance using the validation set, and modify the hyperparameters to avoid overfitting. Test the model on the testing set after training to see how well it generalizes. Lastly, evaluate the model's performance using suitable measures like MSE or accuracy.



DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Figure3.5: Wavelet not De-noise before ELMAN NN db3 level (1) Model of Oil Crude Production

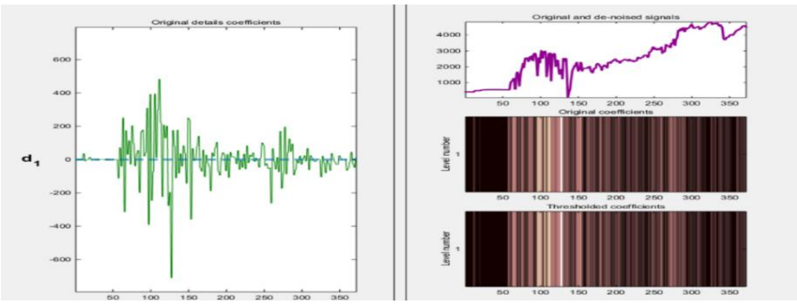


Figure 3.6: Wavelet with De-noise before ELMAN NN db3 level (1) Model of Oil Crude Production

Use db3 (level 1) wavelet denoising to eliminate noise from the time-series data on oil crude production. The Elman neural network model uses the denoised coefficients—which include detail and approximation components—as clean input features. Utilize these properties to train the Elman network to forecast oil crude production, improving accuracy by reducing the influence of noise.

h. Using train, testing, and validating for Appling ELMAN After using (db3_1)

To begin, divide the data into sets for testing, validation, and training. In order to extract features from the time-series data, apply the wavelet transform using db3_1 (level 1), paying particular attention to the approximation and detail coefficients. Train the Elman network on the training set using these wavelet coefficients as input. In order to adjust hyperparameters and prevent overfitting, track performance on the validation set during training. Lastly, test the learned model's prediction accuracy and generalization using the testing set.

General Trends: Using a larger percentage of data for training (e.g., 80% training and 10% testing/validation) tends to result in lower and more stable

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

MSE, RMSE, and MAE values, suggesting that a bigger training set may improve model performance.

Training-Validation-Test Split: While the model is typically robust across splits, its performance is somewhat impacted by various data splits (e.g., 70-15-15, 80-10-10, or 60-20-20).

Top-Performing Setups: Particularly on validation and testing sets, the rows with the split of (%60, %20, %20) and the smallest MSE values (e.g., MSE = 0.00021) typically have very high R-squared values (near 1), suggesting outstanding performance.

In order to choose the optimal model, we should search for the model with the lowest mean squared error (MSE) value. The model with the highest MSE (0.00021) is the one that matches the row with the (%60, %20, %20) ratio (1:02:03). This model's greater prediction accuracy isindicated by its lower mean square error (MSE). It also exhibits excellent correlation and R^2 values, making it the best option according to MSE.

i. The Structure of the Elman Neural Network After using db3_1

After using the Wavelet transform with db3_1, the optimal Elman Neural Network (NN) configuration has adjusted hyperparameters for improved performance. By lowering noise and identifying important characteristics in the data, the wavelet transform increases the accuracy of the network. For the task at hand, this setup demonstrates enhanced generalization and predictive ability.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

Table (4) : Train, Testing, and Validating for Appling ELMAN After Frist level using Wavelet transform

(Training, Validation, Testing	Ratio	MSE	RMSE	MAE	R Training	R Validation	R Testing	R All	R^2 Validation
(%70 , %15 , %15)	1:02:01	0.004	0.06325	0.521	0.95441	0.97011	0.96946	0.954	0.941113
(%70 , %15 , %15)	1:02:02	0.0051	0.07141	0.5166	0.92906	0.98723	0.94056	0.941	0.974623
(%60 , %20 , %20)	1:02:03	0.00021	0.0146	0.5235	0.99785	0.99632	0.99834	0.998	0.992654
(%70 , %15 , %15)	1:02:04	0.00074	0.02729	0.5245	0.99014	0.99217	0.99552	0.992	0.984401
(%60 , %20 , %20)	1:02:05	0.0021	0.04583	0.5234	0.97224	0.99411	0.98388	0.976	0.988255
(%70 , %15 , %15)	1:02:06	0.0015	0.03873	0.5211	0.98235	0.9883	0.98746	0.984	0.976737
(%80 , %10 , %10)	1:02:07	0.00061	0.02473	0.5272	0.99602	0.99488	0.98347	0.993	0.989786
(%70 , %15 , %15)	1:02:08	0.0018	0.04243	0.5272	0.99395	0.98455	0.92779	0.979	0.969339
(%80 , %10 , %10)	1:02:09	0.0047	0.06856	0.5311	0.94694	0.98952	0.9148	0.949	0.97915
(%80 , %10 , %10)	1:02:10	0.0056	0.07483	0.5263	0.92082	0.97962	0.98244	0.935	0.959655
(%80 , %10 , %10)	1:02:11	0.00053	0.02304	0.5224	0.99507	0.9929	0.99322	0.994	0.98585
(%80 , %10 , %10)	1:02:12	0.00081	0.02844	0.5268	0.98972	0.99427	0.99559	0.991	0.988573
(%80 , %10 , %10)	1:02:13	0.0013	0.03606	0.5221	0.98306	0.99844	0.99276	0.986	0.996882
(%60 , %20 , %20)	1:02:14	0.0018	0.04243	0.5234	0.97151	0.99022	0.99839	0.98	0.980536
(%80 , %10 , %10)	1:02:15	0.00052	0.02281	0.53264	0.99457	0.99473	0.99269	0.994	0.989488

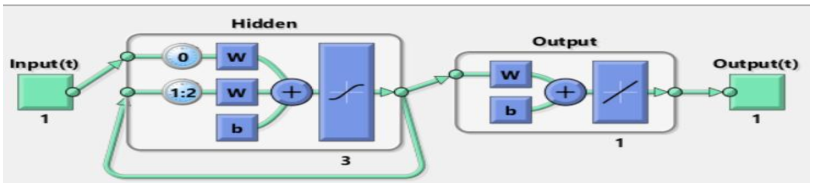
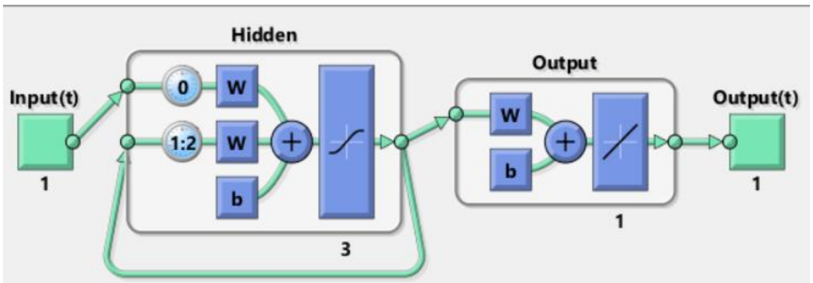


Figure 3.7: The Configuration of the Best Elman NN After Wavelet of the First Level (1:2,3) for Oil Crude Production [Reference: Researchers by (Matlab Software)]



j. The regression plot generally has four graphs showing for the best training:

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

As shown in the figure below, a regression plot is used to compare the target data (actual) with the projected data (outputs). Regression plots often include four graphs: one for training, one for validation, and one for testing; and finally, one for integrating all of them, following Wavelet at the first level. Our neural network construction was confirmed to be accurate.

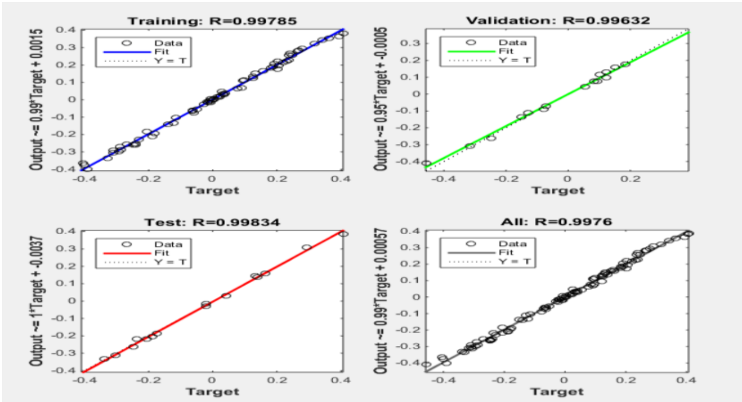


Figure 3.8 : Regression Plots Displaying the Elman NN (1:2,3) for Oil Crude Production After Wavelet First Level data

The overall objective of this analysis would be to use wavelet analysis and neural networks in conjunction to anticipate future oil production or comprehend the dynamics of production data. One might evaluate the model's performance graphically with the help of the regression plots.

k. Elman NN Training Time-Series Response:

Elman NN's time-series response plot for validated instances is shown in the image below. In addition, it reveals the exact intervals that are going to be used for training, testing, and validation. The Elman NN (1:2,3) test result is shown. The data was accurately represented by the After Wavelet For First Level model, as shown by the modest errors in the training, testing, and validation subsets and the uniformly distributed outputs on the response curve.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 518- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

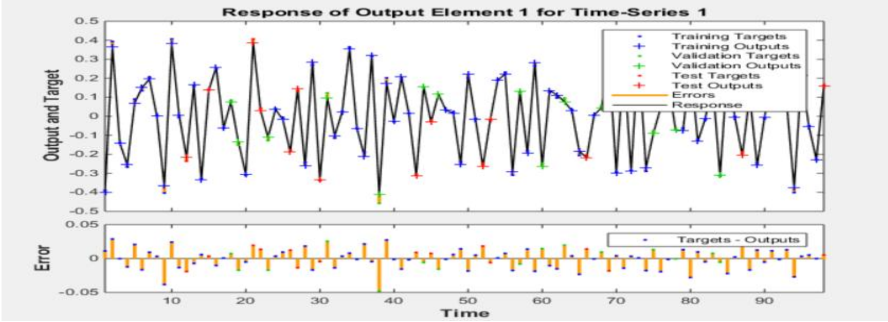
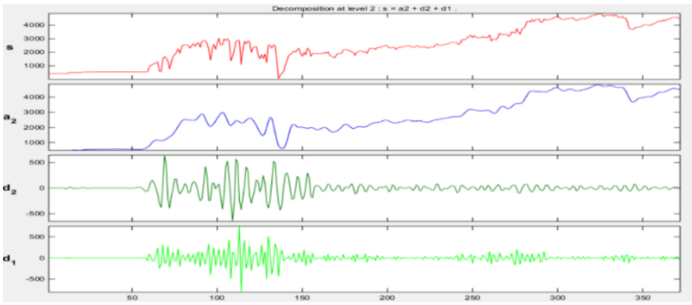


Figure 3.9 : The Time-Series Response for the ELMAN NN (1:2,3) Model After using Wavelet For First-level Oil Crude Production

l. Using train, testing, and validating for Appling ELMAN <u>After</u> using (db7_2)

Through this process, the objective is to combine the strengths of the Elman neural network (which models temporal dependencies) and wavelet decomposition (which captures both long-term trends and short-term fluctuations) to create a robust predictive model that can accurately forecast future values of a time-series variable (such as crude oil production).



DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

Figure 3.10: Wavelet not De-noise after ELMAN NN db7 level (2) Model of Oil Crude Production

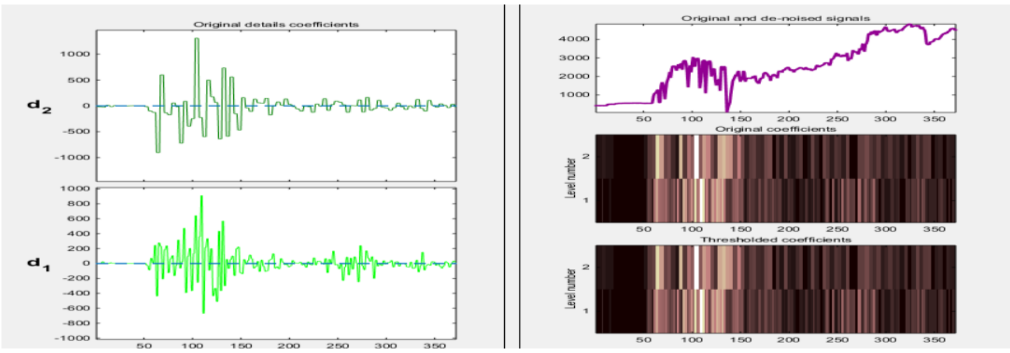


Figure 3.11: Wavelet with De-noise before ELMAN NN db7 level (2) Model of Oil Crude Production

Table below provides an overview of various models for the Crude Oil Production Series After Wavelet Second level. The optimal model is determined as the one with the lowest values for these measures, including MSE and R-square (Validation), in comparison to others. Validation of residual randomness for the chosen model is carried out using Elman Neural Networks, with After Wavelet.

Table (5) : Train, Testing, and Validating for Appling ELMAN After second level usin Wavelet transform.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

(Trining , Validation , Testing	Ratio	MSE	RMSE	MAE	R Training	R Validation	R Testing	R All	R^2 Validation
(%60 , %20 , %20)	1:02:01	0.0049	0.07	0.5255	0.93928	0.9439	0.9918	0.9434	0.89095
(%60 , %20 , %20)	1:02:02	0.0074	0.08602	0.5209	0.95885	0.66683	0.97401	0.91328	0.44466
(%70 , %15 , %15)	1:02:03	0.0017	0.04123	0.5254	0.9786	0.98871	0.97771	0.98124	0.97755
(%70 , %15 , %15)	1:02:04	0.00013	0.01149	0.5242	0.99875	0.99851	0.99801	0.99854	0.99702
(%70 , %15 , %15)	1:02:05	0.0025	0.05	0.5242	0.98134	0.91365	0.99095	0.97125	0.83476
(%60 , %20 , %20)	1:02:06	0.0026	0.05099	0.5243	0.96327	0.98851	0.98931	0.97035	0.97715
(%60 , %20 , %20)	1:02:07	0.0032	0.05657	0.5191	0.99096	0.69932	0.99727	0.9634	0.48905
(%60 , %20 , %20)	1:02:08	0.0014	0.03742	0.5276	0.99404	0.99129	0.94743	0.9839	0.98266
(%80 , %10 , %10)	1:02:09	0.0026	0.05099	0.5251	0.97473	0.95292	0.98195	0.97236	0.90806
(%80 , %10 , %10)	1:02:10	0.009	0.09487	0.5352	0.97836	0.59243	0.95465	0.89768	0.35097
(%80 , %10 , %10)	1:02:11	0.000799	0.02827	0.524	0.99534	0.9883	0.97418	0.99099	0.97674
(%60 , %20 , %20)	1:02:12	0.000645	0.02541	0.5254	0.99503	0.99446	0.98543	0.99286	0.98895
(%70 , %15 , %15)	1:02:13	0.000479	0.02191	0.5252	0.99412	0.99443	0.99704	0.99459	0.98889
(%70 , %15 , %15)	1:02:14	0.0003201	0.01789	0.5231	0.99681	0.99745	0.99467	0.9964	0.99491
(%80 , %10 , %10)	1:02:15	0.0015	0.03873	0.5258	0.98101	0.99664	0.99136	0.98386	0.99329

at this instance, the 70%-15%-15% split (as shown at 1:02:04) performs better for generalization than other models. The training-validation-testing split ratio selection has a substantial impact on model performance. To sum up, Model 4 (1:02:04) is the most precise and broadly applicable model.

m. Configuration of the Best ELMAN NN After Wavelet of the Second Level of db7 (1:2,4):

In general, this term describes the setup of a time series forecasting model that employs an Elman Neural Network trained on preprocessed data using a second-level Daubechies (db7_2) wavelet transform. While the Elman NN uses temporal dependencies to make precise predictions, the wavelet transform is utilized to extract significant features from the data. The precise architecture or

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

Al-Ghary Journal of Economic and Administrative Sciences Vol. 21 (No.1) 2025 PP. 521- 526

مجلة الغري للعلوم الاقتصادية والادارية

مجلد (21) عدد (1) 2025

setup details for this model are described in the (1:2,4) section, but further context is required to fully understand it.

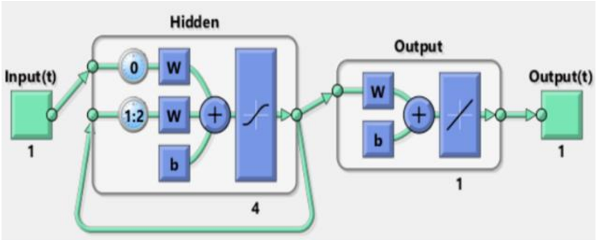


Figure 3.12 The Configuration of the Best ELMAN NN After Wavelet of the Second Level of db7 (1:2,4) for Oil Crude Production[Reference: Researchers by (Matlab Software)]

n. The regression plot generally has four graphs showing for the best training (After using Wavelet):

As shown in the figure below, a regression plot is used to compare the target data (actual) with the projected data (outputs). In a typical regression plot, there are four graphs that display the best results: training, validation, testing, and a combined result, following the second level of wavelets. This supported the validity of our neural network architecture.

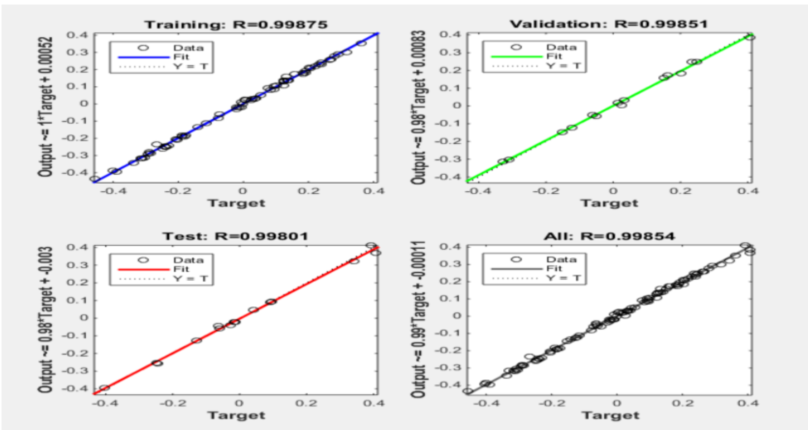


Figure 3.13: Regression Plots Displaying the Elman NN (1:2,4) for Oil Crude Production After Wavelet Second-Level data

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

o. Elman NN Training Time-Series Response (After using Wavelet):

Elman NN's time-series response plot for confirmed cases is shown in the figure below. It further reveals the intervals used for testing, validation, and training. According to the results, Elman NN (1:2,4). Results from the training, testing, and validation subsets showed minor errors and an evenly distributed response curve, suggesting that the After Wavelet For Second Level model accurately portrayed the data.

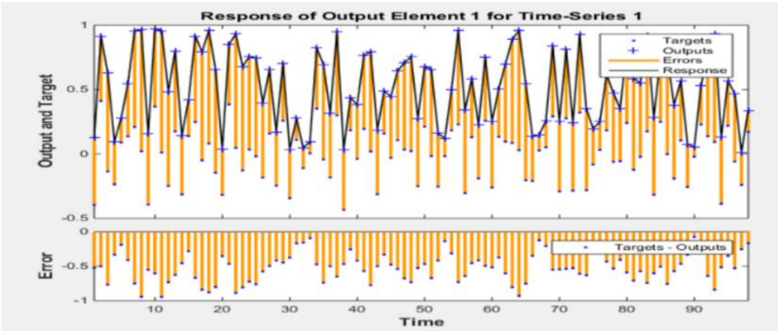


Figure 3.14:The Time-Series Response for the ELMAN NN (1:2,4) Model of the After Wavelet For Second-level Oil Crude Production

p. Comparison between Elman Neural Network and before and after using Wavelet transform:

To evaluate the impact of the wavelet, transform, we can compare the performance of the Elman NN before and after using wavelet preprocessing using several common metrics.

Table (6): Using differ metric to comparison among the models

Metrics	Before using Wavelet	After Using Wavelet	
	ELMAN NN (1:2,15)	db3_1(1:2,3)	db7_2 (1:2,4)
Mean Square Error	0.0003712	0.00021	0.00013
Root Mean Square Error	0.019266551	0.0146	0.01149
Mean Absolut Error	0.5239	0.5235	0.5242
Training Time (s) R^2	0.9909207025	0.9957046	0.99750156
Validation R^2	0.994507563	0.9926535424	0.9970222201
Testing R^2	0.9940289401	0.9966827556	0.9960239601
All R^2	0.94381225	0.996004	0.9970821316

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

In general, the Elman Neural Network performs much better when wavelet transforms are used, especially when the db7_2 wavelet is used. While without appreciably lengthening the training time, the db7_2 configuration yields the greatest results on the majority of measures, including more accurate predictions, a higher model fit (R^2), and enhanced generalization to unseen data.

4.Conclusion and Recommendations:

4.1 Conclusions

Based on the data analysis, we reached a set of conclusions as follows:

The Elman Neural Network's (ENN) performance is greatly improved by applying wavelet transformations, especially the db7_2 wavelet, which increases the ENN's predicted accuracy. Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) both drop significantly after applying wavelet preprocessing methods like db3_1 and db7_2 to the data, suggesting that the model's predictions are improved after preprocessing. An additional metric for the model's ability to explain data variation is the R-squared (R^2) value, which is enhanced by wavelet modifications. R^2 values for the validation and testing sets rise once wavelets are applied; db7_2 achieves the greatest R^2 across all three sets (training, validation, and testing), showing that it fits the data better and can generalize to new data better. Here we see how wavelet preprocessing improves the model's learning ability. Also, training time is just slightly longer when the db7 wavelet is used, proving that wavelet modifications hardly affect computing cost. The substantial gains in model correctness, even with this little increase, more than compensate for the additional processing cost. Wavelet preprocessing is a crucial step in improving neural network performance since it decreases error

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

metrics and increases R^2 values. It improves the input data, which makes the model more accurate and applicable to other situations. Wavelet transformations improve the network's pattern- and relationship-detecting capabilities by extracting useful features from raw input. Consequently, the ENN experiences enhanced stability and convergence during training. Because it strikes a good mix of precision and efficiency, the db7_2 wavelet stands out as the best. When applied to ENN as a whole, wavelet preprocessing proves to be an effective method for enhancing deep learning models.

4.2 Recommendations

Based on the conclusions, we got a set of recommendations as follows:

i. Using wavelet transforms, especially db7_2, prior to training the Elman Neural Network is advised due to the notable gains in MSE and RMSE. The model is better able to identify both high- and low-frequency patterns in the data thanks to this preprocessing phase, which produces predictions that are more accurate.

ii. It is recommended to give the db7_2 wavelet configuration priority when preparing data for time series forecasting jobs because to its superior outcomes (lowest MSE, RMSE, and greatest R^2). Performance enhancement and computational efficiency are best traded off with the db7_2 wavelet.

iii. Although the wavelet transforms clearly improve model accuracy, it's vital to keep in mind that employing wavelet preprocessing could marginally lengthen training times. The trade-off between enhanced performance and computational expenses should be carefully considered

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

for real-time applications or huge datasets. Alternatives such as db3_1 may be taken into consideration in these situations since they provide better performance with less computational load than db7_2.

Reference:

1-Abramovich, F., Bailey, T. C., & Sapatinas, T. (2000). Wavelet analysis and its statistical applications. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *49*(1), 1-29.

2-Adhikari, R., & Agrawal, R. K. (2013). An Introductory Study on Time Series Modeling and Forecasting. arXiv preprint arXiv:1302.6613.

3-Bisgaard, S., & Kulahci, M. (2011). Time Series analysis and Forecasting by Example. John Wiley & Sons.

4- Chatfield, C. (2000). Time-series Forecasting. CRC press.

5-Chatfield, C., & Xing, H. (2019). *The analysis of time series: an introduction with R*. Chapman and hall/CRC.

6-Ellis, J., Mavromatos, N. E., Nanopoulos, D. V., & Sakharov, A. S. (2003). Quantum-gravity analysis of gamma-ray bursts using wavelets. *Astronomy & Astrophysics*, *402*(2), 409-424.

7-Elman, J. L. (1990). *Finding Structure in Time*. Cognitive Science, 14(2), 179-211.

8-Fugal, D. L. (2009). Conceptual wavelets in digital signal processing: an in-depth. *Practical Approach for the Non-mathematician*.

9-Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks*. Springer.

10-Hamad, A. S. (2010). *Using Some Thresholding Rules in Wavelet Shrinkage to Denoise Signals for Simple Regression with Application in Rezgary Hospital–Erbil* (Doctoral dissertation, Doctoral dissertation, PhD.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.

مجلة الغري للعلوم الاقتصادية والادارية
مجلد (21) عدد (1) 2025

dissertation in statistics, college of Administration and Economic, university of Sulaimania, Iraq).

11-Hamdin, A. O., & Hussein, M. M. (2023). An application of Wavelet Markov Chains Model to Study Earthquake Occurrence. Tikrit Journal of Administrative and Economic Sciences, 19(61), 2.

12-Hussein, M. M. F. (2022). Comparison of Time Series Models before and after Using Wavelet Shrinkage Filtering to Forecast the Amount of Natural Gas in Iraq. *Cihan University-Erbil Scientific Journal*, *6*(1), 32-46.

13 -Lyashenko, V., Deineko, Z., & Ahmad, M. A. (2015). Properties of wavelet coefficients of self-similar time series.

14- Percival, D. B., & Walden, A. T. (2000). *Wavelet methods for time series analysis* (Vol. 4). Cambridge university press.

15-Peruggia, M. (2004). The analysis of time series: an introduction. Journal of the American Statistical Association, 99(467), 906.

16-Ren, G., Cao, Y., Wen, S., Huang, T., & Zeng, Z. (2018). A modified Elman neural network with a new learning rate scheme. Neurocomputing, 286, 11-18.

17-Torrence, C., & Compo, G. P. (1998). A practical guide to wavelet analysis. Bulletin of the American Meteorological society, 79(1), 61-78.

18- Zhang, G. P., & Qi, M. (2005). Neural network forecasting for seasonal and trend time series. European journal of operational research, 160(2), 501-514.

DOI: https://doi.org/10.36325/ghjec.v21i1.18766.