Iraqi Journal for Applied Science (IJAS)
Vol. 1, No. 1, June, 2024, pp. 9-23
ISSN: 3006-5828, **DOI**:10.69923/IJAS.2024.010102

Preview the predictive performance of the STR, ENN, and STR-ENN hybrid models

Maryam Luqman Othman [1], **Nihad S. Khalaf Aljboori** [2]
[1] Department of Mathematic, College of Education for Women, Tikrit University, Tikrit, Iraq
[2] Department of Mathematic, College of Education for Women, Tikrit University, Tikrit, Iraq

Article Information

Article history:

Received: 19, 04, 2024
Revised: 22, 05, 2024
Accepted: 02, 06, 2024
Published: 30, 06, 2024

Keywords:
STR-ENN hybrid
STR
ENN
BP Algorithm
RNN

Abstract

This paper presents the mathematical structure of the STR decomposition model and the Elman neural network, in addition to the structure of the hybrid model combining the two previous models. The stages of analysis and verification of each model are discussed separately, and the paper proposes the use of the STR decomposition model based on the autoregressive equation and moving averages, while the STR-ENN model is a model that combines the STR model and the ENN neural network.

The studied data series represents the monthly US soybean oil contracts for the period from 1-2-1997 to 1-5-2022, and using the MATLAB-a2022 program, the results obtained from the hybrid algorithm were compared with the STR model and the ENN neural network individually, to find out which The models are better in terms of prediction, with the prediction accuracy criteria MAE and MSE used. The proposed STR-ENN model had the best predictive performance among the rest of the models, as it had an average absolute error value less than the other two models.

Corresponding Author:

Maryam Luqman Othman
Mathematic, College of Education for Women
Tikrit University, Tikrit, Iraq
Email: Mariam.loqman964@st.tu.edu.iq

1. INTRODUCTION

The fundamental question that must be asked is: What are the conditions that must be met for a model to be stable? To answer this question, the researchers were divided into three sections. The first section of the researchers was concerned with finding the stability conditions for the model, while the other section of them was finding the conditions for specific models. As for the remaining part of the researchers, they focused their attention on dividing the time series into parts so that the model would be easier to understand and give more accurate information about the nature of the time series. Therefore, many models related to the last type of this stability appeared, of which the trend and seasonality decomposition model was presented using STR regression as one of these models. Which was first proposed in 2015 by Dokumentov and Rob [1] followed by many improvements such as: In 2020, a new method for analyzing seasonal data was proposed. Unlike other analysis methods, STR allows for multiple seasonal and periodic components, covariates, seasonal patterns that may contain incorrect periods, and seasonality with complex topology [2]. Follow up with a study entitled “Forecasting Energy Load Time Series.” Using integrated memory network for decomposition and autoencoder [3], and a study on extending RobustSTL to handle multiple seasons. To speed up the computation, a special generalized ADMM algorithm is proposed to perform the decomposition efficiently [4].

Artificial neural networks are considered one of the most important areas of artificial intelligence and have wide application in various fields. One of these networks is the Elman Recurrent Neural Network, symbolized by the abbreviation ENN. The ENN is a recurrent neural network (RNN) designed to capture and store information. Contextuality in hidden layer. First introduced by Jeff Elman in 1990 to process data sequences [5], this network was developed and based on many subsequent studies such as: improved ENN based on a new hybrid optimization algorithm for this purpose [6], Zhang et al presented a paper showing how an Elman OIF neural network with feed-back to inputs can effectively solve nonlinear problems [7]. As for Li et al., they presented a study based on Elman neural network and using models. The results indicate that ENN can be considered a better modeling method for predicting indoor temperature due to its simpler network structure, smaller storage space, and better prediction accuracy [8]. Then Fan et al. presented a study in which they proposed a new parametric conjugate gradient method based on the secant equation to train ENN in an efficient manner [9], and in 2022 Guo et al. presented the state-of-health (SOH) estimation of a lithiumion battery (LIB) based on the ENN [10].

This paper aims to find a hybrid model that combines a time series model STR and a recursive neural network model ENN, then make a comparison between the hybrid model and the individual models separately and find out which models are better in terms of analysis and forecasting.

The first section briefly describes the structural structure of the hybrid STR, ENN, and STR-ENN models. The second section describes the steps for analyzing the data used in the study, which is the monthly average of US soybean oil contracts in dollars, and the approach used in this work for each model. As for the third, the results obtained from each model will be presented, with a comparison between the results obtained, and final section, the most important conclusions obtained.

2. Method

2.1 STR model

We assume that we have a time series x_t and the model STR is given in the form:

$$x_t = TC_t + \sum_{j=1}^{Q} S_t^{(j)} + \sum_{p=1}^{P} \alpha_{p,t}\varepsilon_{t,p} + I_t \tag{1}$$

TC_t is a periodic trend with smooth variation, while $S_t^{(j)}$ represents the change with seasonal components using complex topology, I_t represents the remainders of the series, while $\varepsilon_{t,p}$ are variables with coefficients $\alpha_{p,t}$ that can have time variation up to seasonality.

Assuming that the seasonal component has recurring patterns with slow or constant regular change over time. While the basic smooth trend component of the data (taking into account the presence of seasonality). The residual component contains only white noises and distinct patterns in the data. When the components take a non-linear direction, the data is processed using a transformation to make the data stable. The total number of parameters to be estimated is usually much larger than the length of the time series. Thus, we impose some regularization on the coefficient estimates. In fact, without some regularization, the trend, seasonal components and time variation coefficients cannot be determined. The innovation in the STR approach is the introduction of an estimation method that allows these registrations to be imposed in an efficient manner using the matrix of different coefficients, which transforms the estimation method into a linear model. In this way, the estimation becomes analogous to ridge regression "is a method of estimating the coefficients of multiple regression models in cases where the independent variables are closely related. It is a particularly useful method for alleviating the problem of multicollinearity in linear regression, which usually occurs in models with large numbers of parameters. Overall, the method provides improved efficiency in parameter estimation problems for an acceptable amount of bias [11]." Although the matrices involved can also become very large, sparse, so sparse matrix algebra methods can be used to reduce the computational burden. To explain the steps of modeling the STR model in detail, it is as follows [12]:

- Data Processing :The STR model is generally suitable for stable time series and cannot be applied to unstable time series. Therefore, the stability of the series must be checked before constructing the model.
- Determine the rank of the lag part of the regression Determine the rank of the time differences part, assuming we have a linear regression model
- Perform the nonlinear test, i.e. choose the linear model and the nonlinear STR model.
- Pattern recognition is the selection of a conversion function. After passing the nonlinear test, pattern recognition was performed, and a choice was made between the STR model or any other model

- Estimating the parameters of the STR model by several methods Parameter estimation methods mainly include the Gauss-Newton iteration method, the simulation annealing method, and the grid search method. This is often done by the maximum likelihood function
- Testing the evaluation and adaptability of the STR model. The model is mainly evaluated by comparing the value of the Akaike Information Criterion (AIC) and other criteria for the model: The adaptability test of the model tests whether $\{z\}$ represents a random series according to the autocorrelation coefficient of the residual series $\{z\}$.

2.2 Elman Neural Network

Elman Neural Network is abbreviated as ENN. This network is considered one of the types of recursive neural networks. Figure 1 represents the structure of the recurrent neural network.

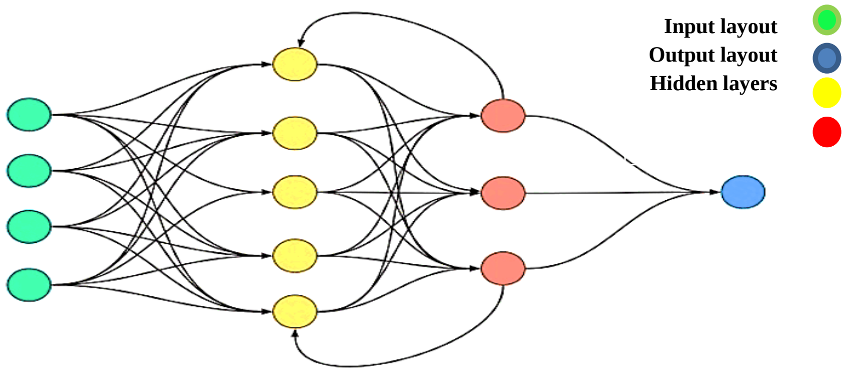


Figure 1: Structure of a recurrent neural network [13]

The ENN is referred to as a simple recursive network and is a special case of artificial recursion, differing from traditional two-layer networks in that the first layer contains a recurrent connection. That is, it is a simple artificial neural network consisting of three layers and has a back loop from the hidden layer to the input layer, called (Context layer). The outputs of the context layer are also inputs to the hidden layer. The ENN has a memory that allows it to detect and generate temporally variable patterns, and typically contains sigmoid artificial neurons in the hidden layer, and linear artificial neurons in the output layer. With this combination of artificial neurons, the transfer functions can approximate any function with arbitrary accuracy if only there are enough artificial neurons in the hidden layer. The ENN is similar to the Jordan network, except that the only difference is that the context units are fed from the output layer instead of the hidden layer [13]. Uses an Back Propagation algorithm Known as BP, this algorithm is a method that performs back propagation. The algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs that we want the network to calculate, and then the error (the difference between the actual and expected results) is calculated. The idea of the backpropagation algorithm is to reduce the output error so that the artificial neural network learns the training data. Training starts with random weights, and the goal is to adjust them so that the error is minimal to optimize the input loss at the input end of the neural network to obtain better effects without additional costs during inference time. This algorithm is known for its slow speed due to the constant and small learning rate coefficient used to update the weights of the neurons. The work of this algorithm includes two stages, which are (the training stage. Training depends on the decent gradient rule, which aims to adjust the weights and reduce system errors in the network, while the second stage is known as testing.

For multilayer networks, the output of one layer becomes the input to the next layer. The equations that describe this process are [13]:

$$a^{m+1} = f^{m+1}(w^{m+1}a^m + b^{m+1}), m = 1, \dots, M-1 \qquad (2)$$

Where a represents the output of the neuron, while f represents the activation function, w represents the vector of offsets in the neural network, b represents an adjustable numerical parameter representing the bias (displacement), and M represents the number of layers in the neural network.

<u>The steps of the algorithm can be summarized as follows:</u>

The first step: propagate the input forward across the network

$$a^0 = p \qquad (3)$$

Where p is a numerical constant

The second step: Calculate the output for each layer using equation (3), so that the outputs of the neurons in the last layer are considered the outputs of the network

$$a = a^M \tag{4}$$

The third step: spreading the sensitivities backward through the network using the following equations:

$$(s^M \to s^{M-1} \to \cdots \to s^2 \to s^1) \tag{5}$$

$$s^M = -2\grave{F}^M(n^M)(t-a) \tag{6}$$

$$s^m = \grave{F}^m(n^m)(w^{m+1})^T s^{m+1}, m = M-1, \ldots, 2, 1 \tag{7}$$

When

$$\grave{F}^m(n^m) = \begin{bmatrix} \grave{f}^m(n_1^m) & 0 & \ldots & 0 \\ \ldots & \grave{f}^m(n_2^m) & 0 & 0 \\ 0 & 0 & 0 & \grave{f}^m(n_s^m) \end{bmatrix} \tag{8}$$

Step Four: The weights and biases are updated using the approximate steepest descent rule. The approximate steepest descent or saddle point method is an extension of the Laplace method for approximating the integral, so the equations are used:

$$w^m(k+1) = w^m(k) + \alpha s^m (a^{m-1})^T \tag{9}$$

$$b^m(k+1) = b^m(k) + \alpha s^m \tag{10}$$

The ENN applies the error back propagation algorithm, but with recurrent back connections in the hidden layers, so that the equations of the modified ENN become as follows :

$$h_t = f_h(Q_h x_t + R_h h_{t-1} + l_h) \tag{11}$$

$$y_t = f_y\left(Q_y h_t + l_y\right) \tag{12}$$

So x_t represents the input vector, h_t and y_t represent the hidden layer and output vectors respectively, f_h and f_y represent the activation functions, and finally R, Q and l represent a vector and the parameter matrix.

2.3 STR-ENN hybrid

The model is predicated on the idea that the data series is split into three parts by the STR model: a seasonal component, a trend component, and a residual component. Verifying that the remaining time series is regressively processed before delivering all the components to the ENN network for training and testing the obtained data may be too restrictive. based on the seasonal model. The processes for creating the suggested model are shown below:

1. Accessing the data series under study.
2. Prior to developing the model, make sure the series is stable. The data is transformed into a stable time series by pre-processing techniques like the difference transformation approach.
3. The series is split into three parts using equation (1), with the trend component describing the fundamental smooth mean of the data (after taking seasonality into account). The data's distinctive patterns and noise make up the remaining component, which is thought to be uncorrelated.
4. The final stage in creating the seasonal model is estimating the regression coefficients in the residual component.
5. Make two groups out of the data for each component that was acquired in step (3): a test sequence and a training sequence.
6. The ENN was used to train the three acquired sequences in order to predict the test sets for each component independently.
7. To get a forecast for the original series, use equation (1) once more to combine the predictions made by the three components.
8. Utilizing the following formulas, determine the relative error value R as well as the mean absolute error (MAE), mean square error (MSE), and other values:

$$MAE = \frac{1}{T}\sum_{k=0}^{T}|\hat{y}_k - y_k| \tag{13}$$

$$MSE = \frac{1}{T}\sum_{k=0}^{T}(\hat{y}_k - y_k)^2 \tag{14}$$

$$R^2 = 1 - \sum_j (\bar{y}_k - y_k)^2 \Big/ \sum_j (\hat{y}_k - y_k)^2 \tag{15}$$

Where y_k represents the original observation value at time k, while $\hat{y}_k$ represents its prediction value, T represents the length of the data series, and $\bar{y}_k$ represents the values of the average observations.

9. The best model for predicting the data series is the one that has the best predictive indications in the test set.

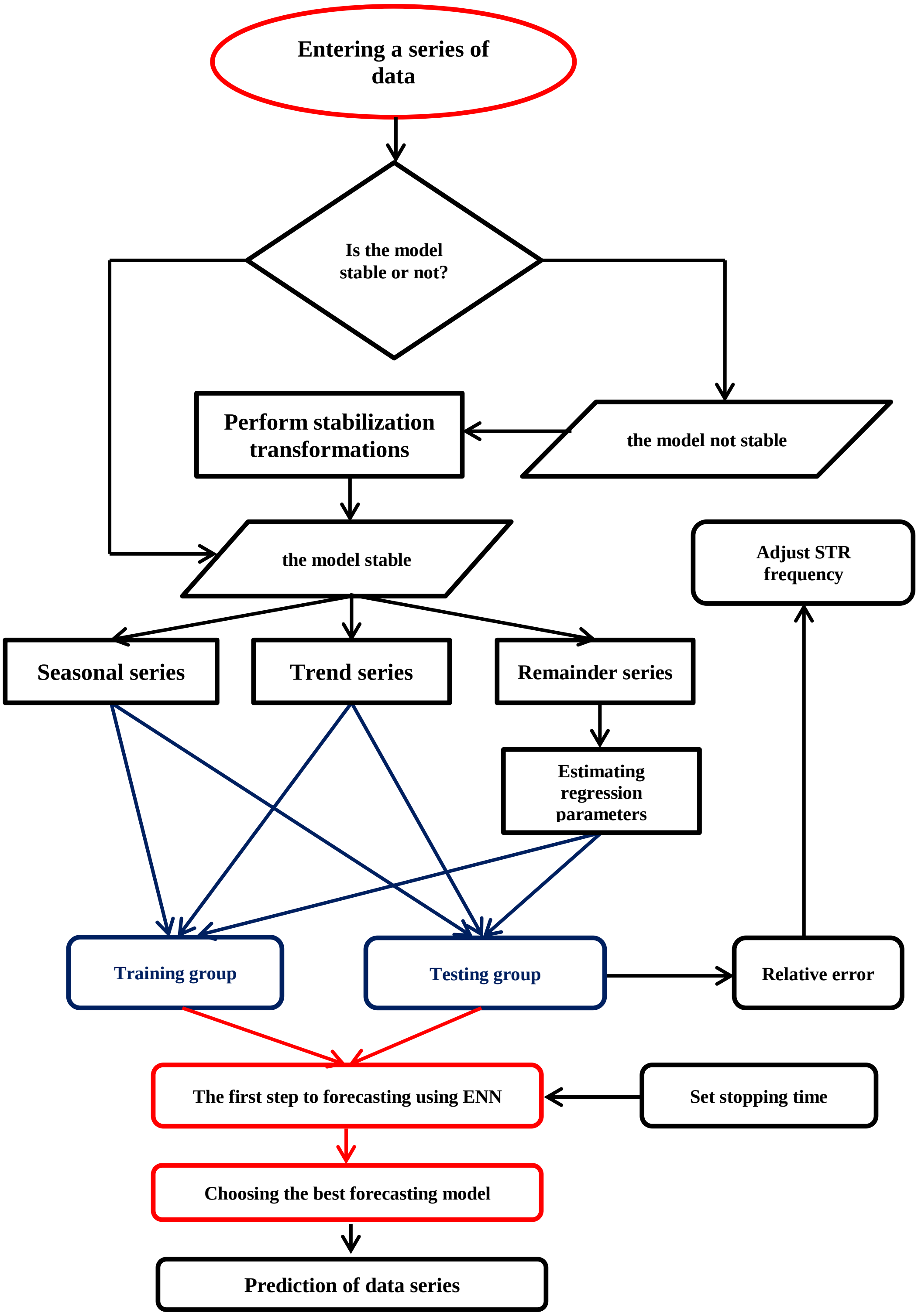


Figure 2: Structure of the STR-ENN model [by Author]

3. Data analysis

The studied data series represents the monthly US soybean oil contracts for the period from 1-2-1997 to 1-5-2022. The time series data was divided into two parts. The first section is the main section for building the structure of the series, which will be relied upon in the analysis of the models used. The second section, which is the evaluation group, represents the last 8 observations of the data series, which will be relied upon to examine the predictive performance of the three models used in this study. Where Figure 3 represents the data series.

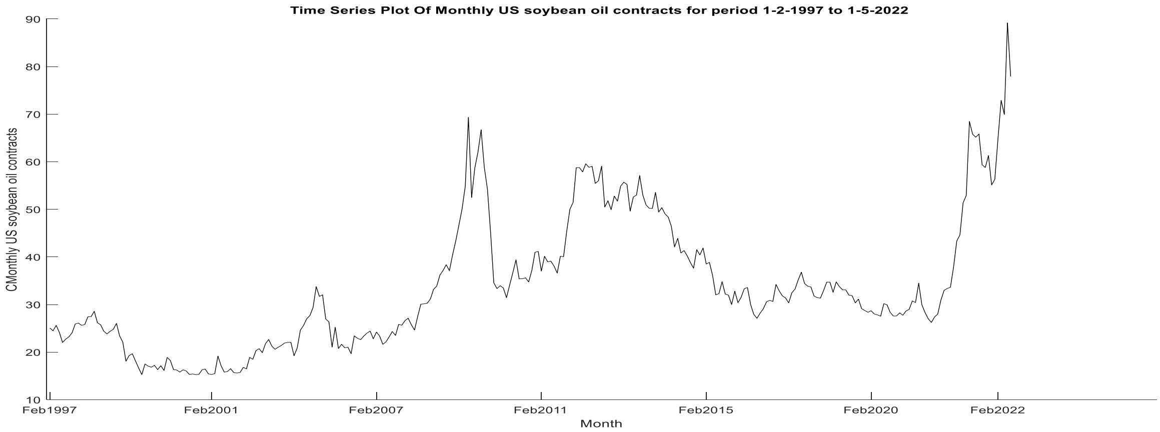


Figure3: Data series

After plotting the data series, the ACF and PACF functions are plotted to determine the stability of the series, Figure 4 shows ACF and PACF for data series.

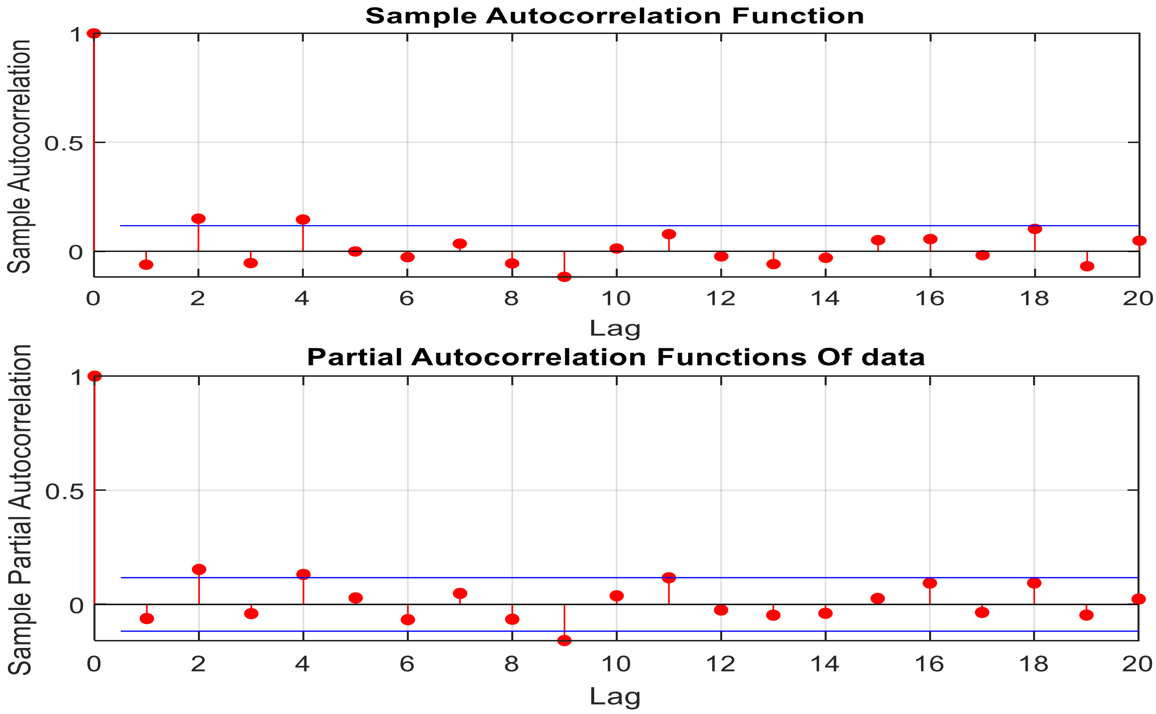


Figure 4: ACF and PACF for data series

To obtain stability of the series, So that the lags values are within the confidence interval [14], [15], [16] we use the logarithm transformation for this purpose, Figure 5 shows converted data series

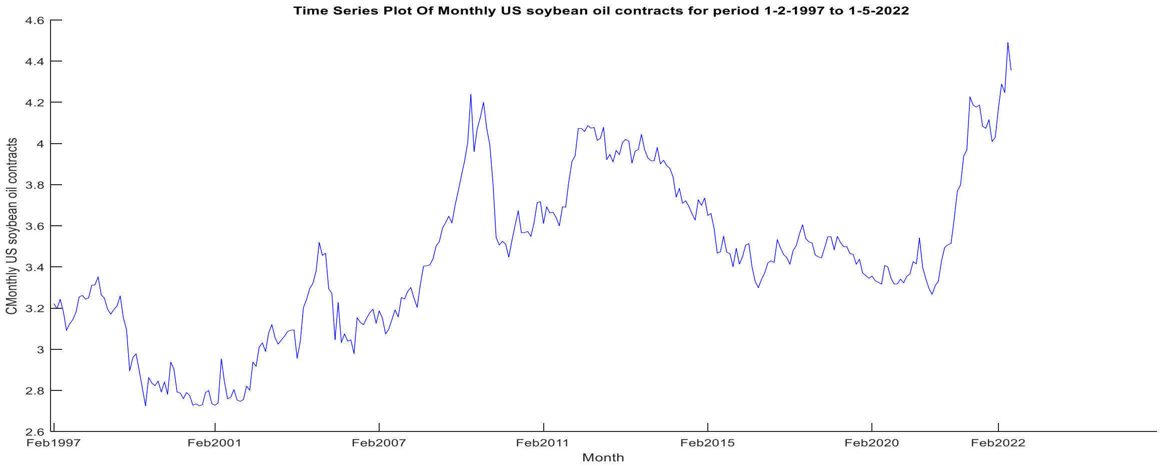


Figure 5: Converted data series

3.1 Data analysis by ENN

Two groups comprise the data series. 80% of the observations are from the series under study. The training set has 20%. The first group is called the test set. It has 10 hidden layers and a stopping time of 1000 stops. The training set's values are in table (1). The training methods' values are in table (2).

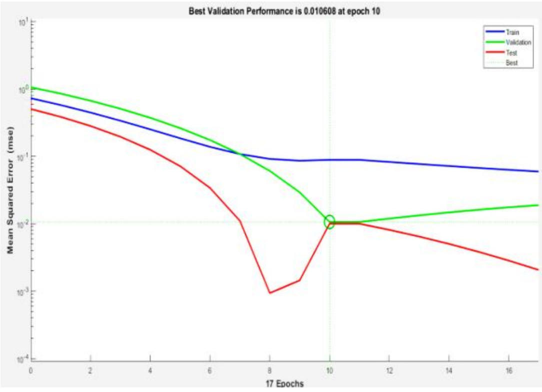
Table1: Values of the training set for the ENN

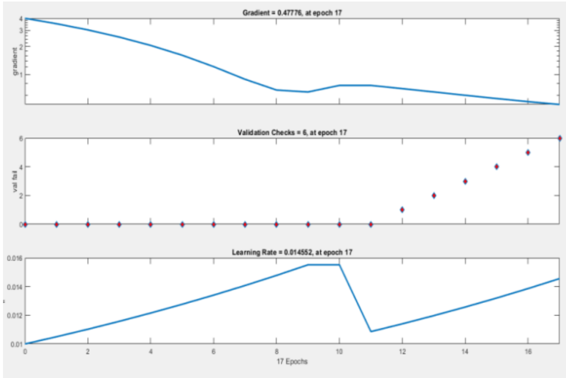
Units	Initial value	Stopped value	Target value
Epoch	0	17	1000
Elapsed Time	-	00:00:05	-
Performance	0.727	0.0594	0
Gradient	4.02	0.478	1e-05
Validation Checks	0	6	6

Table2: Values of training algorithms in ENN

Training algorithm	**Value**
Data division	Random
Training	Gradient Descent with moments and adaptive LR
Performance	MSE
Calculations	MEX

As follows, Figure 6 represents the values of the training and validation set for the best validity and training state.





a b

Figure 6: a- The values of the training and validation set for the best validity, b- training state

As shown in the figure above (b), the Gradient values are equal to 0.47776, while the validation checks value is equal to 6, and finally the learning rate value is equal to 0.014552 at cycle 17.
The error values in the graph with 20 bins can be observed in Figure 7 as follows:

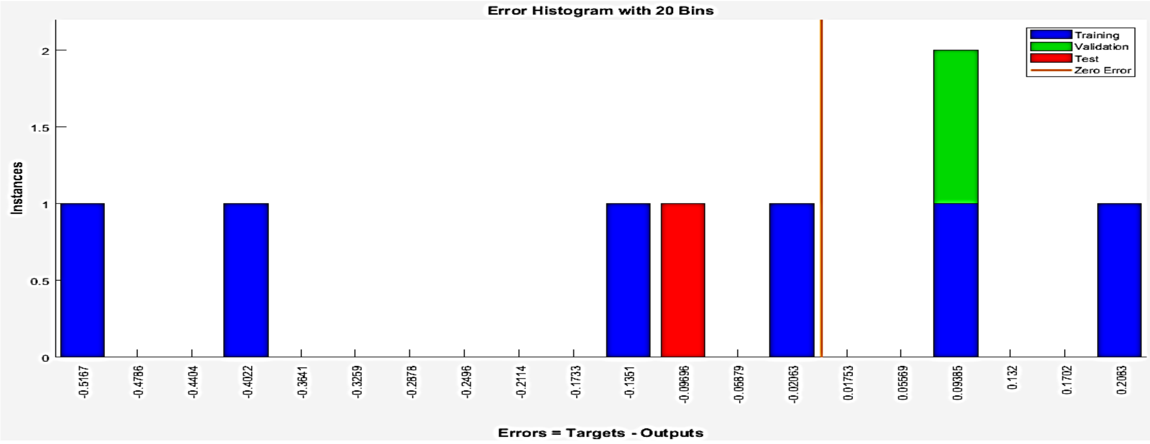
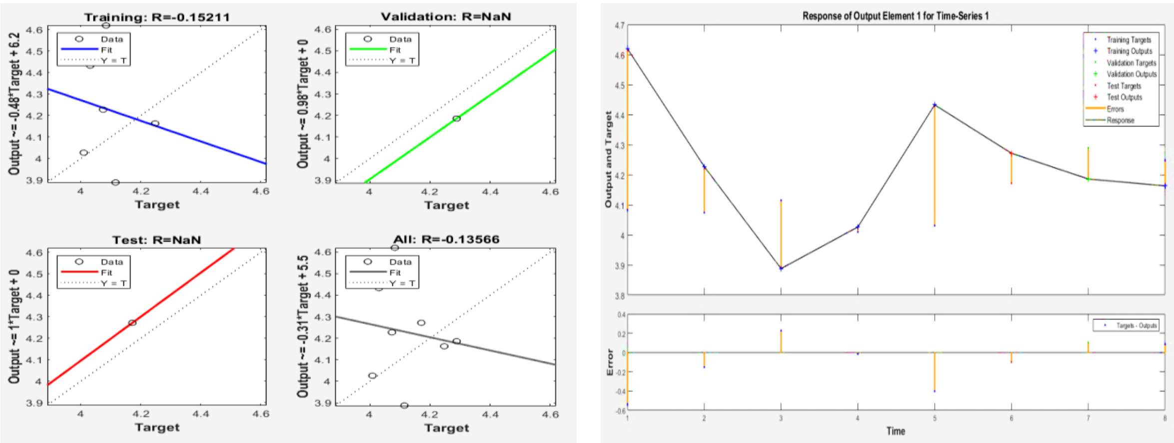


Figure7: Error values for a histogram with 20 boxes in an ENN network

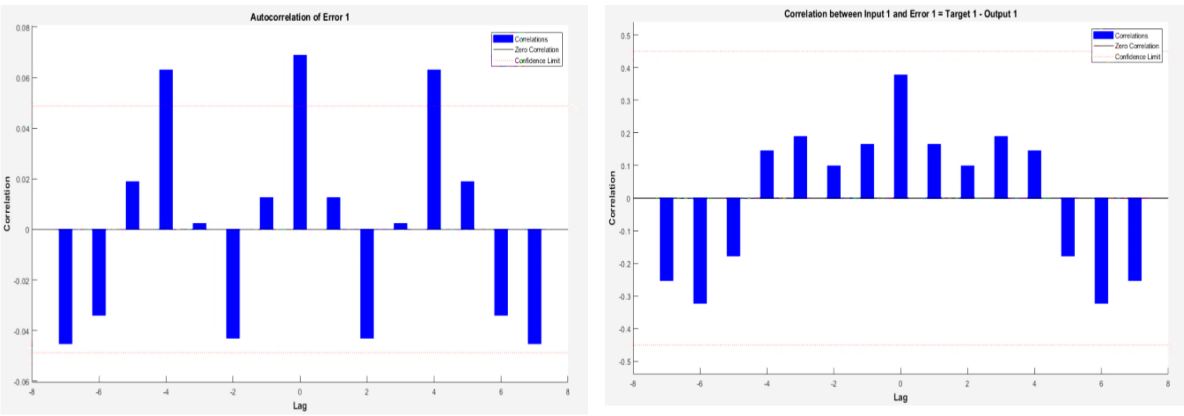
The error histogram is a graph of the errors between the target values and the expected values after training the feedforward neural network (ENN). Since the error graph is in the last third of the graph of the series, I have 20 bins on it, it corresponds to an error of 0.00029 but close to 1 and the validation and test data set is between 1 and 2. Therefore, Figure 8 shows the regression of training the ENN neural network at the end cycle 17, as the linear regression of goals in relation to outputs is drawn, and Output element-1 for time series-1 as follows:



A B

Figure8: A-ENN training regression, B- Output element-1 for time series-1

As for drawing the autocorrelation function for errors in the ENN neural network with 8 time differences (Lags), shown in Figure 9 (a), while b in Figure 9 represents the correlations between the input and output errors of the ENN.



a b

Figure 9: a- Autocorrelation of ENN errors, b- the correlations between the input and output errors of the ENN

3.2 Data analysis by STR

To start analyzing the data with the seasonal STR model, we draw the converted data series as shown in Figure 5. Then, we divide the series into its three parts: trend, seasonality, and residual. Figure 10 represents a plot of the components of the time series.

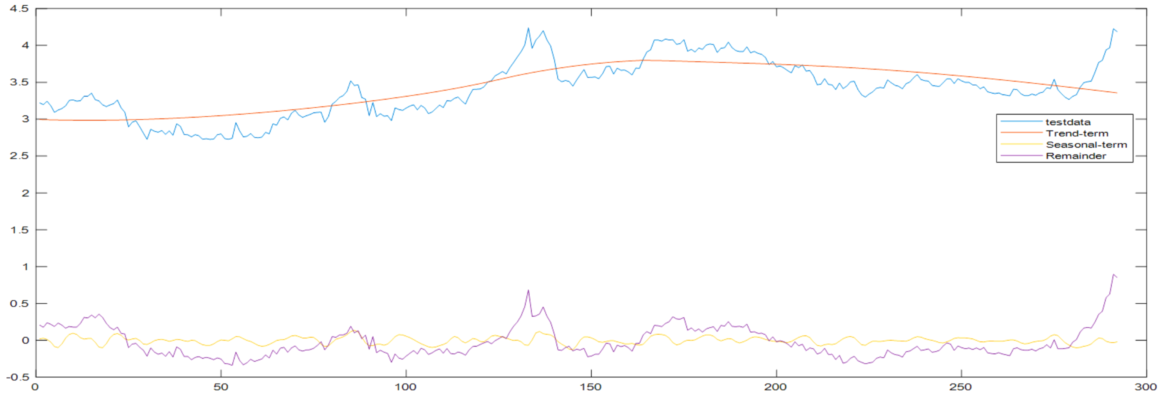


Figure10: plotting the components of the time series

In analyzing the STR model according to equation (1), we will use the autoregressive model to choose the best rank for the regression. We will do this based on Akaike and Bayesian information criteria (AIC and BIC). We determine it using equations (16) and (17).

$$AIC = -2\,l(\hat{\theta}) + 2k \quad [17] \tag{16}$$

$$BIC = -2l(\hat{\theta}) + k\ln n \quad [18] \tag{17}$$

Here, $l(\hat{\theta})$ is the estimated maximum likelihood. n is the number of observations in the series. And, k is the number of estimated parameters. Accordingly, Table 3,has the AIC and BIC values for different ranks.

Table 3 Values of AIC and BIC information criteria for different ranks

ARMA(p,q)	AIC	BIC
ARMA(1,0)	710.4620	747.1953
ARMA(1,1)	700.1871	714.8804
ARMA(1,2)	704.1871	726.2270
ARMA(1,3)	708.1871	737.5737
ARMA(2,0)	712.4620	752.8686
ARMA(2,1)	702.1871	720.5537
ARMA(2,2)	706.1871	731.9003
ARMA(2,3)	710.1871	743.2470
ARMA(3,0)	714.4620	758.5419
ARMA(3,1)	703.7043	725.7443
ARMA(3,2)	737.0909	707.7043
ARMA(3,3)	711.7043	748.4376

The last step of the STR model analysis is to draw the predicted sub-series one by one. , Figure 11 represents a plot of the estimated the subseries, and predicted subseries.

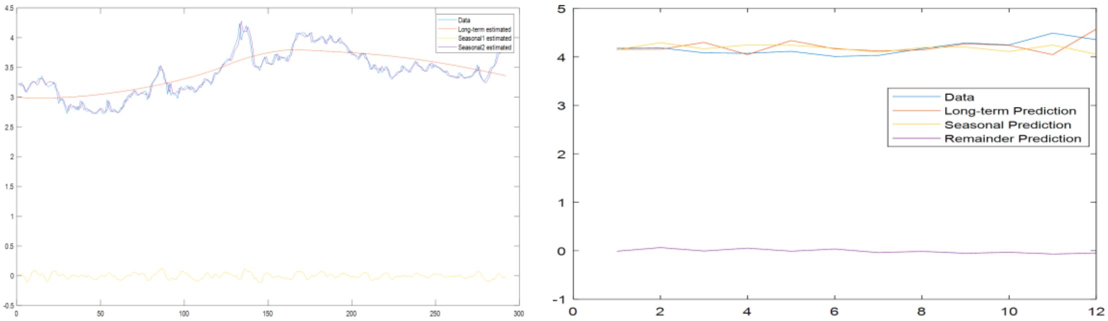


Figure 11 the plot of the estimated the subseries, and predicted subseries

3.3 Data analysis by STR-ENN

The first stage must be the description stage. In this stage, Figure 3 shows the drawing of the data series. Checking stability follows this. We do this by drawing autocorrelation functions, as in Figure 4. Then, we take the logarithm transformation to stabilize the data series.

The second stage is to give the time series numerical stability. Then, we use the seasonal STR model to divide the time series into three parts. They are seasonal, trend, and residual. Figure 12 represents the sub-parts of the time series.

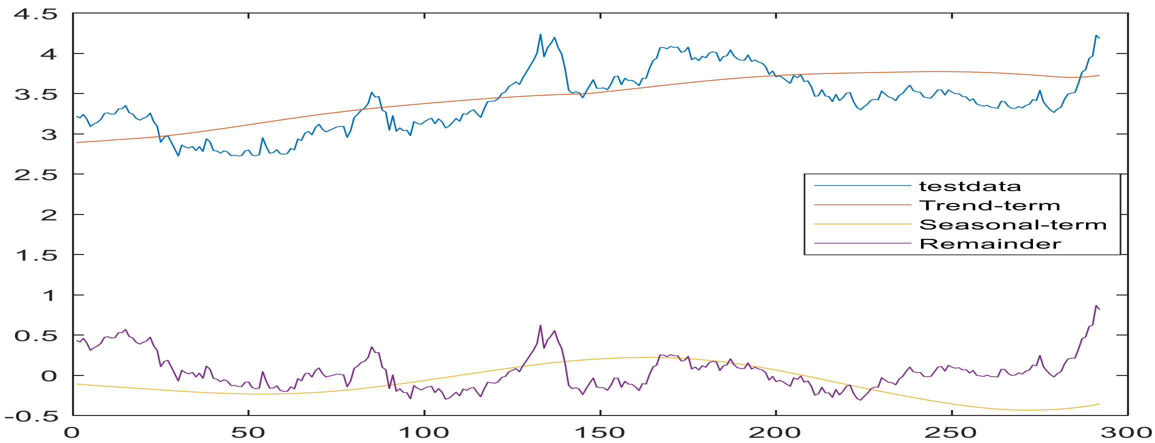


Figure12: Sub-parts of the time series

From Figure 12, It is evident that the data exhibits a high trend, the residuals that are close to zero indicate that the seasonal and trend components are accurate. Consequently, the time series exhibits little noise after processing.

The fourth stage, The team split the initial series. But first, we estimate the regression parameters of the residual series. Then, we run the neural network. This prepares them for the final step: sending them to the neural network. Each subseries of the three sequences is divided into two groups, a test sequence of 80%, and a training sequence of 20%, with 10 hidden layers with a stopping time of 1000. The average values are computed and reflect the value 5.9519 that is used in the calculation of the relative error as described in equation 15. Table 4 displays the values of the training set, while Table 5 displays the values of the training methods.

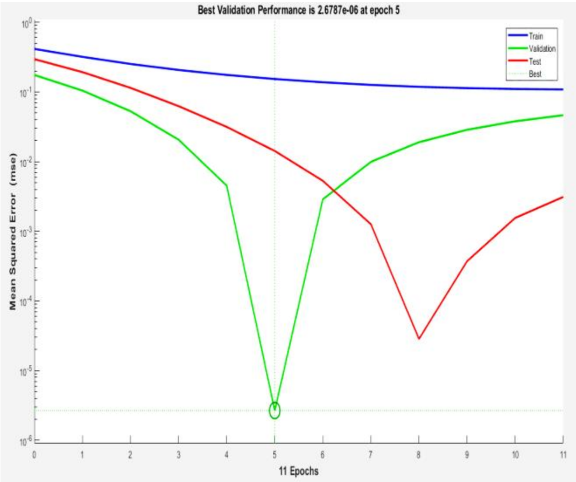
Table4: Values of the training set for the STR-ENN

Units	Initial value	Stopped value	Target value
Epoch	0	0	11
Elapsed Time	-	-	00:00:08
Performance	0.414	0.414	0.109
Gradient	3.32	3.32	0.417
Validation Checks	0	0	6

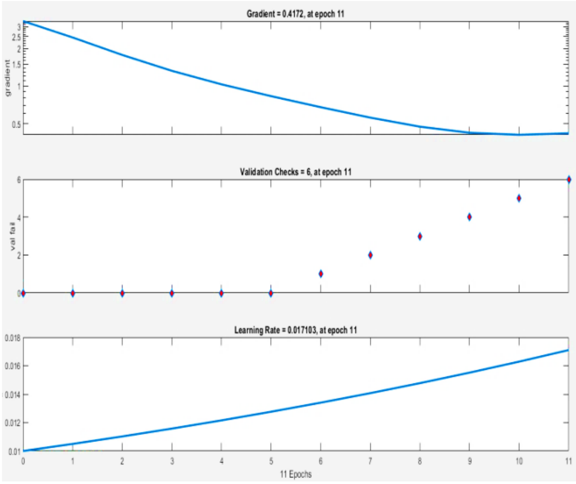
Table5: Values of training algorithms in STR-ENN

Training algorithm	**Value**
Data division	Random
Training	Gradient Descent with moments and adaptive LR
Performance	MSE
Calculations	MEX

As follows, Figure 13 represents the values of the training and validation set for the best validity and training state.



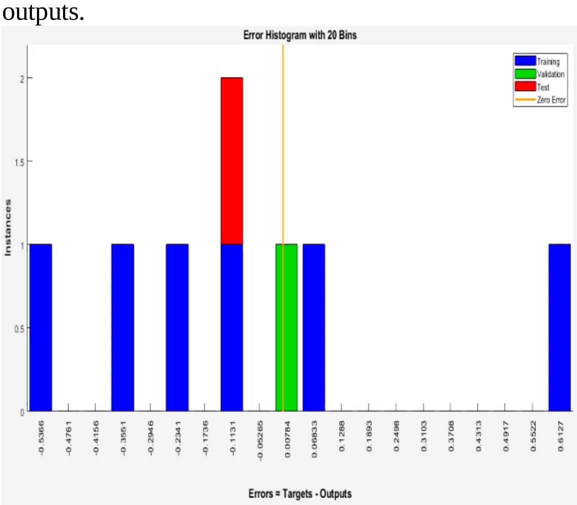
a



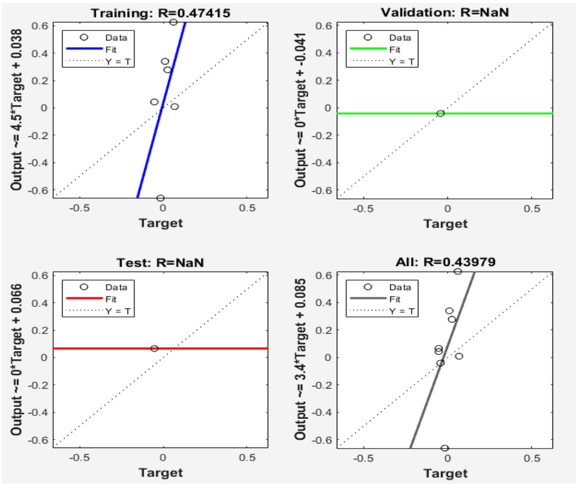
b

Figure13: a-the values of the training and validation set for the best validity, b- training state

Training state it means the state of the neural network after the completion of the training phase. As shown in the figure above, the Gradient values are 0.4172. The validation value is 6, and the learning rate is 0.017103 at epoch 11. We can observe the error values in the graph with 20 bins in Figure 14(a). Figure 14(b) shows the STR-ENN neural network's training regression at cycle 211. It plots the linear regression of goals versus outputs.



a



b

Figure14: a- Error values for a histogram with 20 boxes in an STR-ENN network ,b-STR-ENN training regression

The error histogram is a graph of the errors. The errors are between the target and expected values. These are from training the (ENN). These error values show how much expected values differ from target values. They can be negative. Or, the boxes are the number of vertical bars you notice on the chart. Here, we divide the total error range into 20 smaller bins. The y-axis shows the number of samples from the dataset. It shows the samples in each bin. The response of output element-1 to time series-1 is shown as follows:

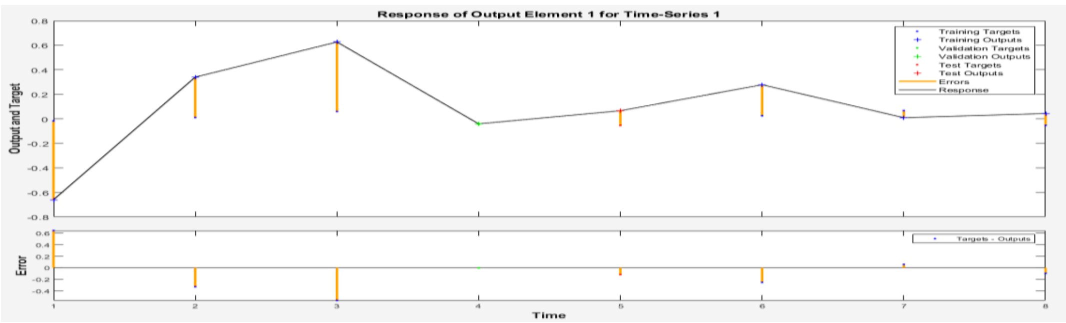
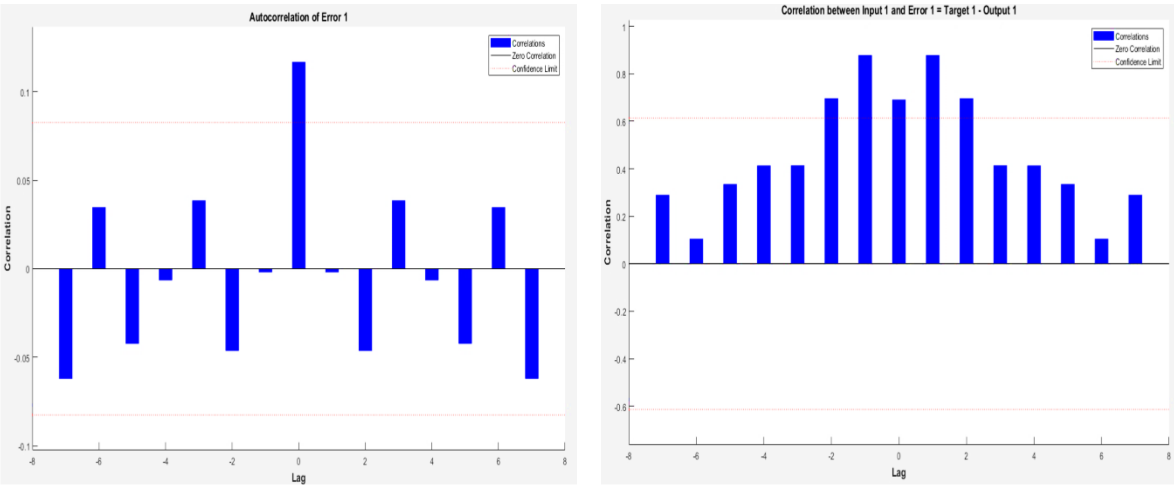


Figure15: The response of output element-1 to time series-1

The response of the output element-1 to time series-1 means that the time series is target t (i.e. the target of the time series) and the time series is the output y (i.e. the output of the time series), so that they are plotted on the same axis, showing the errors between them (i.e. solving the multi-input time series problem And the outputs using learning tools in the ENN network). This network has 8 time differences (lags), as shown in Figure 16(a), there is only one value outside the confidence interval, Figure 16(b) shows the Autocorrelations. It shows Autocorrelation between the input and output errors of the ENN neural network.



a b

Figure16: a- Autocorrelation of STR-ENN errors
b- Autocorrelation between input and output errors of STR-ENN neural network

Finally, the outputs of each sub-series (seasonal, directional, remainder) are obtained for the STR-ENN network.

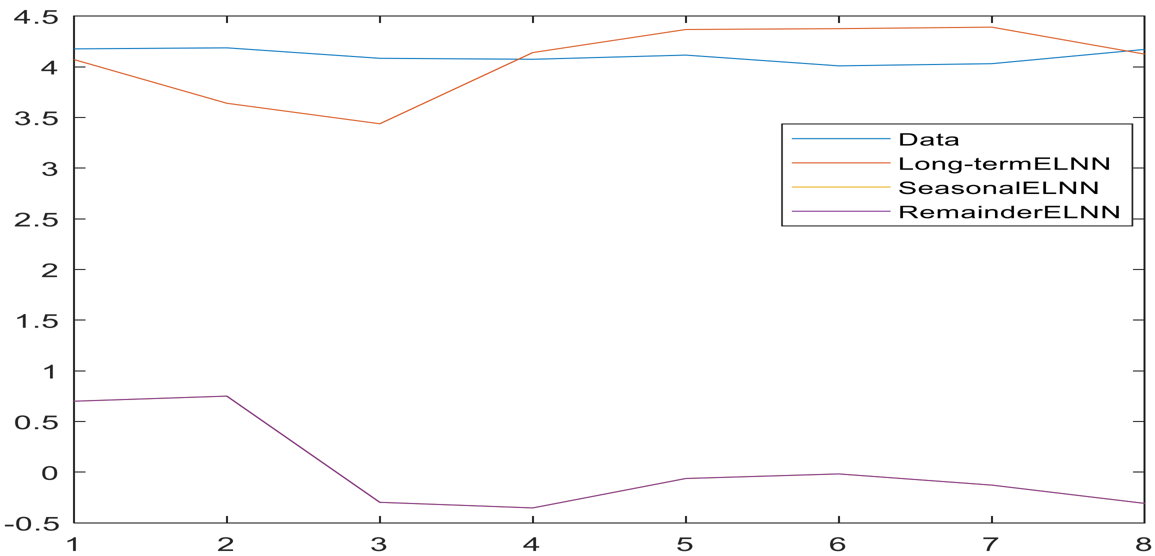


Figure17: the outputs of each sub-series are obtained for the STR-ENN network

4. Comparing the performance of the models used in the analysis Description of the data used

The primary goal of analyzing any time series is to obtain future predictions for the studied series, and as we mentioned at the beginning of the third chapter, by dividing the series into two groups. What is important here is the second group, which aims to examine the predictive performance of each model. To compare them, we used the mean absolute error (MAE) [19], and mean square error (MSE) [20]. We defined these in Equations 12 and 13. Figure18 shows the evaluation group's predictions for the ENN neural network. Figures 19 and 20 show their predictions for the STR and STR-ENN models. They summed them using Equation 1. Table 6 shows the predictions for the three models. It also shows the original observations for the evaluation group, with the criteria. Mean absolute error and mean square error.

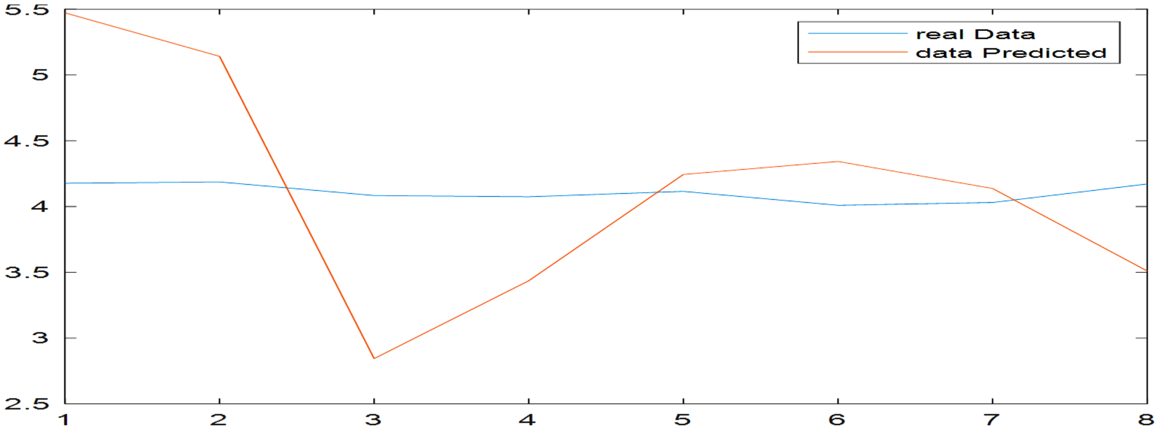


Figure18: Orth group values for the data series with predictions for the ENN model

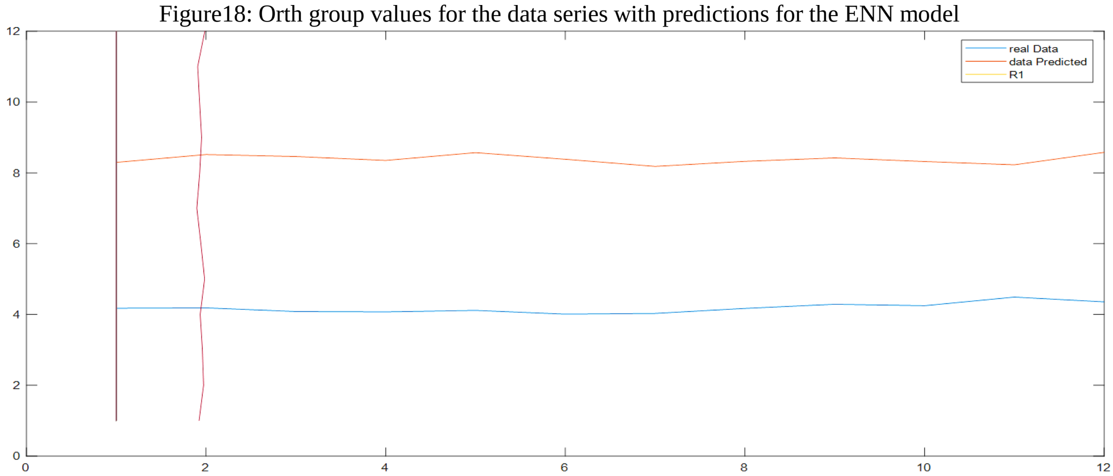


Figure19: Orth group values for the data series with predictions for the STR model

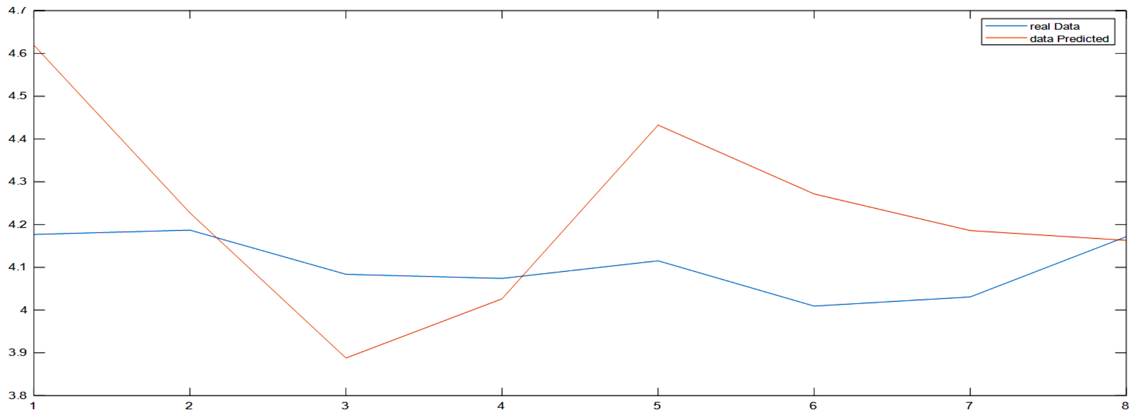


Figure20: Orth group values for the data series with predictions for the STR-ENN model

Table 6: Prediction values for the three models, in addition to the values of the original observations for the evaluation group, with the criteria of MAE and MSE.

	Real data	Analysis by ENN	Analysis by STR	**Analysis by STR-ENN**
1	4.1768	5.4727	8.2940	**4.6194**
2	4.1869	5.1419	8.5163	**4.2273**
3	4.0836	2.8440	8.4600	**3.8879**
4	4.0740	3.4360	8.3493	**4.0263**
5	4.1153	4.2440	8.5694	**4.4326**
6	4.0093	4.3427	8.3828	**4.2715**
7	4.0307	4.1369	8.1819	**4.1860**
8	4.1716	3.5096	8.3245	**4.1631**
	MAE	0.6710	4.1992	**0.1972**
	MSE	0.6389875	18.321425	0.05395875

5. Conclusions

Unlike other decomposition approaches, STR is unique in its ability to handle various seasonal and cyclic components, covariates, non-integer seasonal patterns, and complicated seasonality topologies. It is applicable to time series data having regular time indices, such as hourly, daily, weekly, monthly, or quarterly data. STR is derived from a regularized optimization approach, making it slightly connected to ridge regression. Due to its reliance on a statistical model, we can readily calculate confidence intervals for components, which is not feasible with the majority of current decomposition techniques (such as STL, X-12-ARIMA, SEATS-TRAMO, etc.).

ENN This network is considered one of the types of recursive neural networks that is characterized by the presence of a layer within the hidden layers called the Context layer, and the outputs of the Context layer are inputs to the hidden layer.

STR-ENN is a hybrid method that combines the STR model with the ENN neural network to improve prediction performance for the time series under study.

The results we reached during this study can be summarized as follows: The hybrid method is distinguished from existing methods in that it divides the time series into three components, and each component is analyzed separately using a neural network. This detail is very useful for analyzing heterogeneous time series. Despite the simplicity of the STR model, ease of implementation, and very fast compared to the hybrid model, the STR-ENN model consumed 00:00:08 time when analyzing, while the ENN neural network model consumed 00:00:05 time, meaning that the hybrid model almost... It is very close to the execution speed of the ENN model, and the STR-ENN model has proven its efficiency in analysis, especially for this type of long-term series in which there is a large number of seasonality and a large pattern of directionality that is repeated, as it contains the lowest values for MAE and MSE. The labor market need, whether economic, engineering, medical, etc., requires reliance on the hybrid model. While the STR model proved poor in predicting long sequences.

We also suggest, as future studies, that different decomposition models be used than the STR model, as well as the use of neural network models to create hybrid models to reach the highest prediction accuracy.

ACKNOWLEDGEMENTS

I would like to thank my supervisor who worked with me to complete this research paper

References

[1] D. Alexander and R. J. Hyndman, "STR: A seasonal-trend decomposition procedure based on regression," *Monash econometrics and business statistics working papers,* vol. 13, no. 15, 2015.

[2] A. Dokumentov and . R. J. Hyndman, "STR: Seasonal-trend decomposition using regression," *INFORMS Journal on Data Science,* vol. 1, no. 1, pp. 50-62, 2022.

[3] B. Jatin and D. Toshniwal, "Energy load time-series forecast using decomposition and autoencoder integrated memory network," *Applied Soft Computing,* vol. 93, p. 106390, 2022.

[4] Q. Wen, Z. Zhang, Y. Li and L. Sun, "Fast RobustSTL: Efficient and robust seasonal-trend decomposition for time series with complex patterns," 2022.

[5] J. L. Elman, Finding structure in time, Wiley Online Library, 1990.

[6] D. Yu, Y. Wang, H. Liu, K. Jermsittiparsert and N. Razmjooy, "System identification of PEM fuel cells using an improved Elman neural network and a new hybrid optimization algorithm," *Energy Reports,* vol. 5, pp. 1365-1374, 2019.

[7] X. Li, Z. Han, T. Zhao, J. Zhang and D. Xue, "Modeling for indoor temperature prediction based on time-delay and Elman neural network in air conditioning system," *Journal of Building Engineering,* vol. 33, p. 101854, 2023.

[8] Q. Fan, Z. Zhang and X. Huang, "Parameter conjugate gradient with secant equation based Elman neural network and its convergence analysis," *Advanced Theory and Simulations,* vol. 5, no. 9, p. 2200047, 2022.

[9] Y. Guo, D. Yang, Y. Zhang, L. Wang and K. Wang, "Online estimation of SOH for lithium-ion battery based on SSA-Elman neural network," *Protection and Control of Modern Power Systems,* vol. 7, no. 3, pp. 1-17, 2022.

[10] R. B. Cleveland, W. S. Cleveland, J. E. McRae and I. Terpenning , "STL: A seasonal-trend decomposition," *J. Off. Stat,* vol. 6, no. 1, pp. 3-73, 1990.

[11] R. J. Hyndman and G. Athanasopoulos, Forecasting: principles and practice, OTexts, 2018.

[12] A. Krenker, B. Janez and K. Andrej, "Introduction to the artificial neural networks," *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech*, pp. 1-18, 2011.

[13] P. K. Satpathy, . K. S. Pradyumna and N. M. Mihi, "Elman neural network backpropagation based evaluation of critical busbars in power systems with renewable sources," *international Journal Of Renewable Energy Research*, vol. 2, no. 5, pp. 532-541, 2015.

[14] N. A. Noori and A. A. Mohammed, "Dynamical approach in studying GJR-GARCH (Q, P) models with application," *Tikrit Journal of Pure Science*, vol. 26, no. 2, 2021.

[15] N. S. Khalaf, H. H. Abdullah and N. A. Noori, "The Impact of Overall Intervention Model on Price of Wheat," *Iraqi Journal of Science*, pp. 853-862, 2024.

[16] H. H. Abdullah and A. A. Mohammed, "Stability Study of Exponential Double Autoregressive model with application," *NeuroQuantology*, vol. 20, no. 6, p. 4812, 2022.

[17] H. H. Abdullah and A. A. Mohammed, "Stability Study of Logarithmic Double Autoregressive Model with Application," *Journal of Algebraic Statistics*, vol. 13, no. 3, pp. 3206-3218, 2022.

[18] K. F. Hameed and N. S. Khalaf, "Stability Study of SATER Models in a Dynamic Approach with Application," *Mathematical Statistician and Engineering Applications*, vol. 71, no. 1, pp. 773-790, 2023.

[19] N. A. Noori, "Exploring the Properties, Simulation, and Applications of the Odd Burr XII Gompertz Distribution," *Advances in the Theory of Nonlinear Analysis and its Application*, vol. 7, no. 4, pp. 60-75, 2023.

[20] A. A. Khalaf, N. A. Noori and M. A. Khaleel, "A new expansion of the Inverse Weibull Distribution: Properties with Applications," *Iraqi Statisticians journal*, pp. 52-62, 2024.

BIOGRAPHIES OF AUTHORS

Mariam Loqman, graduated from Tikrit University, College of Education for Women, Mathematics Department, for the academic year 2020-2021, and is a master's student in mathematics in the research year, and is currently continuing at Tikrit University in the College of Education for Women email: Mariam.loqman964@st.tu.edu.iq

Nihad shareef khalaf al jboori, Field (General): Mathematics Field (Specialization): Time series College: College of Education for Women Education : Master's in Applied Mathematics email: nihad.shreef16@tu.edu.iq.