Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A new recurrent neural network for forecasting seasonal time series

Eren Bas
Giresun University
Crina Grosan
King's College London
Erol Egrioglu
erol.egrioglu@giresun.edu.tr

Giresun University

Article

Keywords: Forecasting, Recurrent artificial neural networks, Particle swarm optimization, Seasonality

Posted Date: August 24th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10276183/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A new recurrent neural network for forecasting seasonal time series

Eren Bas[1], Crina Grosan[2], Erol Egrioglu[1,*],

[1] Department of Data Science and Analytics, Faculty of Arts and Science, Giresun University, Giresun, 28200, Türkiye.

[2] Division of Digital Health and Applied Technology Assessment, King's College London, UK

ABSTRACT

Special models are needed for forecasting seasonal time series because of their periodic structures. The artificial neural network models can forecast seasonal time series by using integrated decomposition techniques or including lagged variables about a period as inputs. This paper presents a new recurrent neural network for modelling seasonal time series. The architecture of the proposed network is designed by inspiring exponential smoothing methods and it has recurrent connections similar to exponential smoothing methods. A learning algorithm based on particle swarm optimization and a new objective function for training is proposed for the new neural network. The performance of the new neural network is investigated by using different seasonal time series and statistical hypothesis tests. The proposed network is superior to well-known deep and shallow artificial neural networks and it has the best forecasting performance according to the statistical hypothesis test results.

Keywords Forecasting, Recurrent artificial neural networks, Particle swarm optimization, Seasonality.

Introduction

Statistical time series forecasting methods are the most preferred forecasting methods for seasonal time series in the literature. Seasonal autoregressive integrated moving average model (SARIMA) and Winter's exponential smoothing methods have been commonly used. [1] proposed an automatic forecasting method (ETS) and a model taxonomy for the exponential smoothing methods. ETS can forecast seasonal time series. Many decomposition methods help in forecasting seasonal time series besides, seasonal forecasting methods. Some statistical methods have also developed in recent years. [2] proposed a forecasting seasonal time series based on kernel nearest neighbour nonparametric regression. In this study, parallel regression models are used for forecasting every different season. [3] proposed a seasonally integrated periodic autoregressive model. [4] proposed a Walsch transformation-based technique method for seasonal time series. [5] proposed a new approach based on constructed multiple time series with different sub-seasons from the original time series. [6] used an analytical hierarchy process for ranking seasonal time series forecasting methods. They found that SARIMA is the best method for forecasting seasonal time series methods.

Artificial neural networks are one of the methods for forecasting seasonal time series. For modelling seasonal time series with ANN, a few ways can be preferred. The simplest one is using seasonal decomposition techniques for the time series before modelling with ANNs. Another way is using a model which contains terms for seasonality. In the classical time series analysis models, taking lagged variables around the period as independent variables and creating a seasonality index are used to create models for seasonal time series. Taking lagged variables around the period as inputs is the most preferred method for analysing seasonal series with artificial neural networks.

In the literature, some studies considered computational intelligence techniques for forecasting seasonal time series. [7] proposed a hybrid method of SARIMA and support vector machines for seasonal time series forecasting. [8] proposed a

* Corresponding author.

E-mail address: erol.egrioglu@giresun.edu.tr (E. Egrioglu)

seasonal support vector regression forecasting method based on a genetic algorithm and tabu search. [9] proposed a new hybrid combination of fuzzy rules and computational intelligence methods such as evolutionary artificial neural networks and support vector machines for multi-step seasonal time series forecasting. [10] proposed a forecasting method for forecasting natural gas daily consumption. Their method is a two-step method for modelling seasonality and is based on a multi-layer perceptron. [11] used a convolutional neural network for forecasting seasonal time series. They suggested a CNN model without pooling layers. Because they determined the negative effects of pooling layers for this kind of problem. [12] proposed a new seasonal forecasting method based on first-order fuzzy transform on weather data sets and compared their results with ARIMA and artificial neural networks. [13] proposed a novel support vector regression approach for forecasting time series with multiple seasonal periods. [14] proposed similarity-based machine learning methods for seasonal time series forecasting. They forecasted monthly electricity demand in 35 European countries. [15] used LSTM, GRU, SRNN, Bi-LSTM and Bi-GRU for forecasting five-minute short-term solar irradiance seasonal prediction models. Some studies consider the hybridization of computational intelligence and statistical methods. [16] proposed a parallel and series-connected artificial neural network and statistical models for forecasting demand time series data with complex seasonal structures. [17] proposed parallel genetic algorithms for optimizing the SARIMA model and this approach gave more accurate forecast results. Following two studies considered grey prediction models for forecasting seasonal time series. [18] proposed a grey seasonal least square support vector regression model for seasonal time series forecasting. [19] proposed a grey seasonal model for forecasting seasonal energy forecasting. Fuzzy sets and their variants are used for creating forecasting methods for seasonal time series. [20] proposed a new hybrid approach based on SARIMA and a partial high-order bivariate fuzzy time series forecasting model for seasonal time series forecasting. [21] proposed a time-variant fuzzy time series method for seasonal time series forecasting. [22] proposed a bivariate forecasting method for seasonal time series forecasting. [23] proposed a hybrid method that uses fuzzy time series triple exponential smoothing. [24] proposed a forecasting technique based on fuzzy transformation and logic techniques for seasonal time series. [25] proposed an intuitionistic fuzzy seasonality regression method for forecasting air pollution seasonal time series.

Artificial neural networks should be redesigned for forecasting seasonal time series. [26] proposed a new artificial neural network for seasonal time series forecasting. [26]'s architecture used a multiplicative neuron model which was proposed by [27]. [26]'s architecture is a combination of the seasonal autoregressive and multiplicative neuron model. Artificial neural networks were used with decomposition techniques for forecasting seasonal time series. Artificial neural networks never work with the logic of exponential smoothing methods but it is well known that feedback on the model forecasts can enhance the forecasting performance of the models. The contribution of this study is listed below:

- A new recurrent neural network is proposed in this study. The architecture of the network employs feedback of forecasts at the seasonality period.
- A learning algorithm is proposed for the proposed new neural network.
- A new objective function "named weighted mean square error" is proposed for the training of the network. The proposed objective function gives bigger weights to the last observations of the time series.
- An early stopping strategy is employed in the training processes for preventing overfitting problems.
- Restarting strategy is employed in the training processes for avoiding local optimum traps.

The Proposed New Recurrent Artificial Neural Network

In real-world applications, time series data frequently exhibit seasonal fluctuations that must be accurately accounted for during the modeling phase. To address this issue, this study presents a new recurrent neural network (RNN) framework designed to capture seasonal dynamics without the need for prior data preprocessing techniques. The structural design of the proposed network is divided into two distinct components. The first component uses a convolutional neuron topology.

This component processes the p-lagged observations of the time series $(x_{t-j}, j = 1,2,\dots,p)$ utilizing corresponding weights $(w_j, j = 1,2,\dots,p)$ and biases $(b_j, j = 1,2,\dots,p)$. To introduce non-linearity into this stage, a logistic activation function is integrated. The intermediate output of this first segment, denoted as ξ_t, is determined via Equation (1).

$$\xi_t = \frac{1}{1+\exp(\prod_{j=1}^{p}(w_j x_{t-j}+b_j))} \tag{1}$$

In the subsequent part, the seasonal period of the time series is represented by s. The operational inputs for this stage consist of x_{t-1} and x_{t-s}. Furthermore, feedback inputs originating from the network's historical outputs, specifically, $\hat{x}_{t-1}$ and $\hat{x}_{t-s}$ are incorporated to sustain the recurrent behaviour. The mathematical calculation for the output of this second phase ζ_t is defined in Equation (2).

$$\zeta_t = \alpha_3(\alpha_1 x_{t-1} + (1-\alpha_1)\hat{x}_{t-1}) + (1-\alpha_3)(\alpha_2 x_{t-s} + (1-\alpha_2)\hat{x}_{t-s}), \;\; 0 \le \alpha_1, \alpha_2, \alpha_3 \le 1 \tag{2}$$

Within this formulation, α_1 serves as the smoothing coefficient for the level component of the series, while α_2 represents the seasonal smoothing coefficient. The third parameter, α_3, acts as a weight factor. All three coefficients are restricted to the continuous interval $[0,1]$.

The final layer of the network synthesizes these two distinct operational paths by implementing a combination parameter, β, which is likewise constrained within the $[0,1]$ range. The ultimate forecasting output $\hat{x}_t$ is generated according to Equation (3).

$$\hat{x}_t = \beta\xi_t + (1-\beta)\zeta_t \,, \;\; 0 \le \beta \le 1 \tag{3}$$

The overall architectural schema of the developed network is illustrated in Fig. 1. As displayed in the diagram, B denotes the backshift operator, which operates on the principle of $B^k x_t = x_{t-k}$.

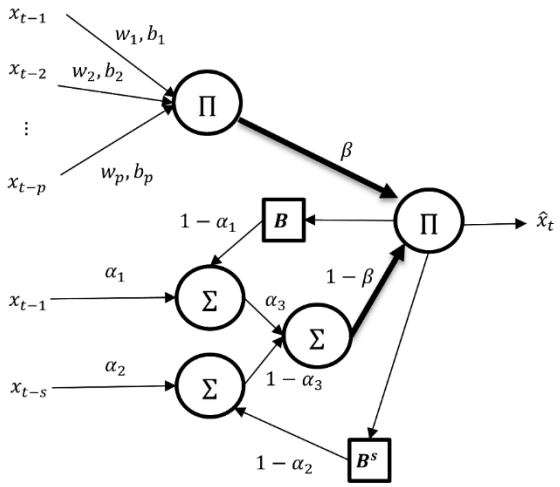

Fig. 1. The architecture of the proposed artificial neural network

A Learning Algorithm based on Particle Swarm Optimization for the Proposed Artificial Neural Network

Fitting the weights and biases of the introduced network topology necessitates a robust optimization framework. Traditional gradient-based learning mechanisms often suffer from localized optimization traps, becoming stagnant at local minima. To remedy this vulnerability, a heuristic learning approach driven by Particle Swarm Optimization (PSO) is established to extract the global optimum for the parameters during time series forecasting.

Standard evaluation criteria such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) given by Equations (4) and (5) treat all historical errors with equal significance. However, in time series forecasting, accurately capturing the most recent observations is generally more critical than fitting distant past data. To mitigate this imbalance, this paper presents an alternative error metric termed Weighted Mean Squared Error (WMSE). The underlying mechanism of WMSE allocates exponentially higher weights to later observations while diminishing the impact of early errors. WMSE is formulated by Equation (6).

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(x_t - \hat{x}_t)^2 \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(x_t - \hat{x}_t)^2} \tag{5}$$

$$WMSE = \frac{1}{n}\sum_{t=1}^{n}\frac{t}{n}(x_t - \hat{x}_t)^2 \tag{6}$$

By designating WMSE as the objective function during the training loop, the network is guided toward superior forecasting performance. Since PSO explores the parameter space globally, the risk of local minimum entrapment is significantly curtailed. To further boost computational efficiency and mitigate potential overfitting hazards, early stopping and dynamic re-starting criteria are embedded into the training architecture.

The sequential execution of the developed learning scheme is detailed below:

Step 1. The primary dataset is split into three independent subsets: training, validation, and testing. The validation matrix is utilized to tune the lag hyperparameter (p), whereas the test matrix serves to evaluate the generalization capacity.

Step 2. A universal boundary set $[p_{low}, p_{up}]$ is defined for the lag parameter (p). These thresholds are chosen based on the seasonal pattern and the empirical autocorrelation function (ACF) profile. For seasonal data structures, it is advised to configure the upper boundary as equivalent to a single seasonal period, while the lower boundary is set to 1.

Step 3. Establish the operational parameters for the particle swarm optimization engine.
pn: Total population size of the swarm.
$maxitr$: Maximum iteration threshold allowed for convergence.
$vmaps$: Velocity restriction limit to prevent chaotic particle displacement.
$[iw_1, iw_2]$: Dynamic range allocated for the inertia weight adjustment.
$[c_1^{initial}, c_1^{final}]$: Boundary values for updating the cognitive coefficient.
$[c_2^{initial}, c_2^{final}]$: Boundary values for updating the social coefficient.
The baseline experimental setups for these parameters can be configured as $pn = 30, maxitr = 1000, vmaps = 1, [iw_1, iw_2] = [0.4, 0.9], [c_1^{initial}, c_1^{final}] = [1,2], [c_2^{initial}, c_2^{final}] = [1,2]$.

Step 4. Define the control mechanisms for the training execution.
Rsp: Trigger index for the dynamic re-starting strategy.
Fil: Maximum tolerable failure index limiting stagnated iterations.
For execution stability, these threshold benchmarks can be designated as taken $Rsp = 30$ and $Fil = 20$

Step 5. Initialize the outer loop control condition by setting $p = p_{low}$

Step 6. Generate the baseline coordinate matrices and velocity vectors for the swarm. The total dimensional space for positions corresponds to $2p + 4$, as graphically itemized in Fig. 2.

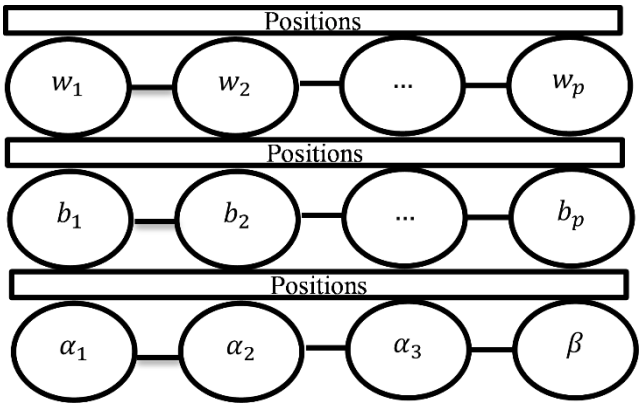

Fig. 2. The positions of a particle

The structural coordinates comprise the feedforward weights, system biases, exponential smoothing parameters, and network combination coefficients. Specifically, the candidate arrays are designated as $PS_{ik},, i = 1,2, \dots, pn\ , k = 1,2, \dots, 2p + 4$. The initialization maps utilize a uniform distribution: initial network weights $(w_j, j = 1,2, \dots, p)$ and bias points $(b_j, j = 1,2, \dots, p)$ are drawn within the $[-1, 1]$ interval. Concurrently, the smoothing components $(\alpha_1, \alpha_2, \alpha_3)$ along with the combination factor (β) are initialized uniformly across the $[0, 1]$ domain. Swarm velocity indicators $(v_{ik}, i = 1,2, \dots, pn\ , k = 1,2, \dots, 2p + 4)$ are initialized via a uniform distribution over the range $[0, 1]$.

Step 7. Calculate the fitness metrics for every individual particle in the swarm. The fitness performance is mapped directly to the WMSE score computed via Equation (6) on the training split. Concurrently, the personal historical best $(gbest)$ vector and the global swarm standard matrix $(PSbest)$ are established.

Step 8. Compute updated velocity fields and coordinate paths using Equations (7) through (13). In these formulations, $iter$ represents the current optimization loop index, whereas $maxitr$ denotes the predefined maximum execution cap.

$$v_{i,k}^{(\mathrm{itr})} = w^{(\mathrm{itr})} v_{i,k}^{(\mathrm{itr}-1)} + c_1^{(\mathrm{itr})} r_1 \left(PSbest_{i,k}^{(\mathrm{itr})} - PS_{i,k}^{(\mathrm{itr})}\right) + c_2^{(\mathrm{itr})} r_2 \left(gbest_k^{(\mathrm{itr})} - PS_{i,k}^{(\mathrm{itr})}\right) \tag{7}$$

$$v_{i,k}^{(\mathrm{itr})} = \min(vmaps, \max\left(-vmaps, v_{i,k}^{(itr)}\right) \tag{8}$$

$$PS_{i,k}^{(\mathrm{itr})} = PS_{i,k}^{(\mathrm{itr}-1)} + v_{i,k}^{(\mathrm{itr})} \tag{9}$$

The time-varying inertia weight alongside the cognitive and social scaling parameters are adjusted iteratively according to the following formulas:

$$w^{(\mathrm{itr})} = (iw_1 - iw_2)\frac{maxitr-\mathrm{itr}}{maxitr} + iw_2 \tag{10}$$

$$c_1^{(\mathrm{itr})} = (c_1^{final} - c_1^{initial})\frac{itr}{maxitr} + c_1^{initial} \tag{11}$$

$$c_2^{(\mathrm{itr})} = (c_2^{initial} - c_2^{final})\frac{maxitr-\mathrm{itr}}{maxitr} + c_2^{final} \tag{12}$$

Furthermore, to guarantee mathematical validity, the coordinate positions relating to the architectural parameters ($\alpha_1, \alpha_2, \alpha_3$ and β) are strictly bounded within the [0,1] operational range using Equation (15):

$$PS_{i,k}^{(\mathrm{itr})} = \min\left(1, max\left(0, PS_{i,k}^{(itr)}\right)\right), i = 1,2,\dots,pn;\ \ k = 2p+1,\dots,2p+4 \tag{13}$$

Step 9. Re-evaluate and refresh the global coordinate tracker ($gbest$) alongside the swarm's historical optimal metrics ($PSbest$).

Step 10. Increment the counter tracking the re-starting threshold ($rsc = rsc + 1$). If the state satisfies the condition $rsc > Rsp$, a global swarm regeneration is triggered, causing all coordinates and velocity vectors to be re-initialized, while the counter is reset to zero.

Step 11. Assess the convergence criteria via the early stopping rule formalized in Equation (14), where esc monitors the failure index loop. The variable$fgbest^{(t)}$defines the global best fitness score captured during iteration t.

$$esc = \begin{cases} esc + 1\ , if\ \frac{fgbest^{(t)} - fgbest^{(t-1)}}{fgbest^{(t)}} < 10^{-3} \\ 0,\ \ otherwise \end{cases} \tag{14}$$

If the condition $esc > Fil$ is satisfied, the optimization engine is stopped early; otherwise, the sequence shifts back to Step 8.

Step 12. Evaluate RMSE values using the verification/validation data subset to analyze the generalized performance of the trained network configuration. At this stage, RMSE is selected as the primary metric instead of WMSE. This selection is intentional, as the primary objective is to evaluate the model's standard one-step-ahead forecasting capability across all validation data points with equal weight distribution.

Step 13. Adjust the lag condition via $p = p + 1$. If the condition if $p \le p_{up}$ remains true, loop back to Step 6; otherwise, route the flow directly to Step 14.

Step 14. Identify the optimum hyperparameter (p) configuration based on the out-of-sample validation assessment. The optimal lag parameter is selected by identifying the specific model structure that yields the minimum RMSE value on the validation dataset.

Step 15. Deploy the chosen network configuration featuring the optimized hyperparameter values on the untouched test dataset

to compute final forecasting performance metrics.

Investigating the Performance of the New Artificial Neural Network

The performance of the proposed network is examined by using six seasonal time series data sets. The performance of the proposed method is evaluated according to the one-step ahead forecasting success. In the application, time series are divided into three parts as training, validation and test set. While the optimal weights of artificial neural networks are determined with the training data, the optimal hyperparameters are selected with the validation set. The results obtained from the test set for the best hyperparameter setup were used to compare the methods. Error metrics were calculated for one step ahead forecasting performance in test sets. The MATLAB function created for the artificial neural network proposed in this study and the data sets used in the study are published at https://github.com/erole1977/SRANN and are available for public use.

In the first application, the classic Box & Jenkins airline data set totals of international airline passengers observed monthly between January 1949 and November 1959 is used. The line graph of the logarithmic air passenger time series is given in Fig. 3.

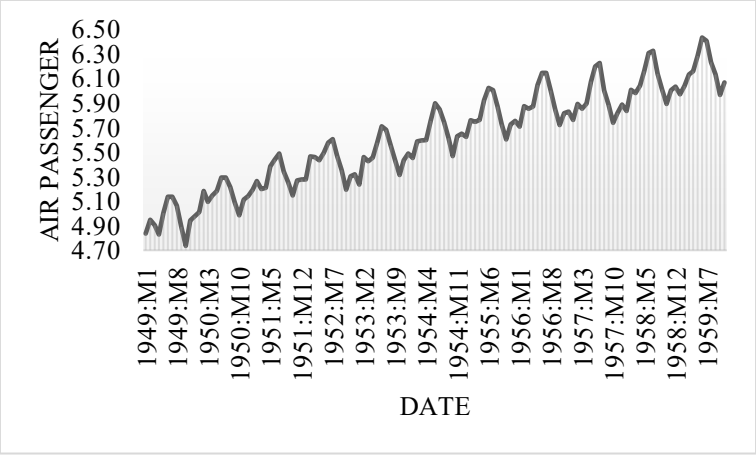

Fig. 3. The graph of logarithmic air passenger time series

As illustrated in Fig. 3, the logarithmic air passenger time series is characterized by prominent seasonal patterns and an upward trend component. Given these structural attributes, seasonal forecasting methodologies are highly suitable for modelling this dataset. For this specific time series, the seasonal period is established as 12 ($s = 12$). In the empirical execution, the total data points allocated for both the validation set and the test set are selected equally as 30. To benchmark the forecasting capability of the newly introduced architecture, the time series dataset is concurrently modelled using three prominent alternative methods: Long Short-Term Memory (LSTM), Pi-Sigma Artificial Neural Network (PS-ANN), and Simple Recurrent Neural Network (SRNN). For a fair comparison, the hyperparameter tuning phase for each competing approach is conducted based strictly on their respective validation set performances. The range for the input lag hyperparameter (p) is restricted between $p_{low} = 1$ and $p_{up} = 12$ utilizing a step-wise increment of one across all evaluated methods. Regarding the network architecture design, the number of hidden layer units is systematically tested between 1 and 12 (with an increment of one) for LSTM, PS-ANN, and SRNN models. Additionally, the depth of the network is varied by setting the number of hidden layers to 1 and 2 specifically for the LSTM and SRNN frameworks.

During the initial phase of the experimental application, the optimal hyperparameter configurations that yield the best validation performance for each model are determined. These identified optimal settings are consolidated and presented in Table 1.

The best hyperparameters of the methods (s=12)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers

LSTM	5	5	1
PS-ANN	2	12	1
SRNN	11	1	2
Proposed Network	-	11	1

Table 1. The best hyperparameters of the methods for logarithmic air passenger time series

Following the parameter tuning phase, all forecasting models are executed using their optimal hyperparameter configurations. To ensure statistical reliability and eliminate initialization bias, each method is executed across 30 independent runs with randomized initial weights and biases. Table 2 provides a comprehensive statistical summary—including the mean, standard deviation, minimum, and maximum values—of RMSE metrics calculated over the untouched test subset.

Furthermore, to validate the superiority of the proposed framework from a statistical standpoint, a non-parametric hypothesis test, specifically the Wilcoxon signed-rank test, is conducted. This statistical test compares the location parameters of two distinct distributions, thereby enabling a robust verification of whether the performance gap between the proposed network and the competitive baselines is statistically significant. The calculated p-values (significance levels) derived from the signed-rank test are reported in the final column of Table 2.

Methods	Statistical Results				
	Mean	Std. Dev.	Min	Max	Significance Value
LSTM	0.270925	0.066675	0.175569	0.451312	1.73E-06
PS-ANN	0.218179	0.031567	0.182673	0.301468	1.73E-06
SRNN	0.317480	0.142098	0.141036	0.537852	1.73E-06
Proposed Network	0.089248	0.019802	0.076731	0.172161	

Table 2. RMSE statistics for the test set of logarithmic air passenger time series

The empirical results outlined in Table 2 clearly indicate that the proposed network dramatically outperforms the alternative models. It achieves the lowest mean RMSE (0.089248) and the lowest minimum error (0.076731), while maintaining a highly stable structure as evidenced by the lowest standard deviation (0.019802). Moreover, the remarkably low significance values (1.73E-06) confirm that the superior forecasting performance of the proposed method over LSTM, PS-ANN, and SRNN is statistically significant and not a result of random chance.

To further validate the generalization and robustness of the developed framework, a second empirical application is executed using the Australian beer consumption dataset. This time series consists of quarterly observations spanning from the first quarter of 1956 (Q1\ 1956) through the first quarter of 1994 (Q1\ 1994). The historical trajectory and structural behaviour of this dataset are graphically illustrated via the line plot in Fig. 4.

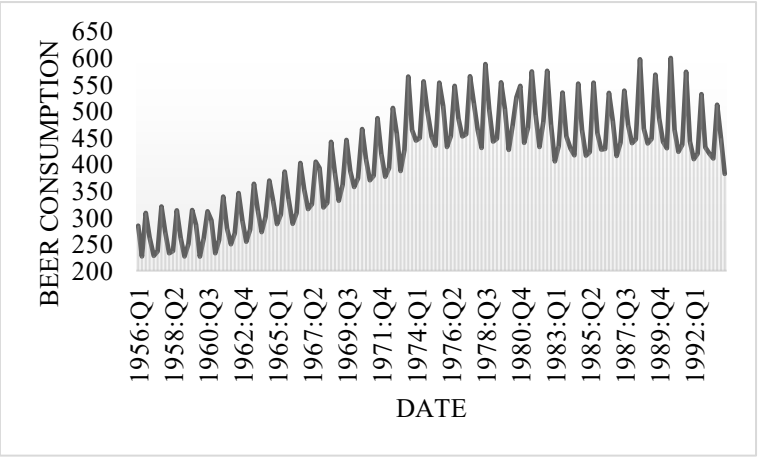

Fig. 4. The graph of the Australian beer consumption time series

As with the previous dataset, an initial hyperparameter optimization process is carried out to configure the optimal architectures for each competing model under a seasonal period configuration of four ($s = 4$). The resulting optimal hyperparameter arrangements—encompassing the number of hidden layer units, input lags (p), and the depth of hidden layers—are compiled and documented in Table 3.

The best hyperparameters of the methods (s=4)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers
LSTM	4	5	1
PS-ANN	7	2	1
SRNN	5	5	2
Proposed Network	-	10	1

Table 3. The best hyperparameters of the methods for the Australian beer consumption time series

Once the ideal model frameworks are established, each network structure is trained and evaluated over the out-of-sample test split. To minimize standard initialization discrepancies, the descriptive statistics of the RMSE values are recorded across multiple randomized iterations. Table 4 presents the comprehensive statistical summary of these test results, reporting the mean, standard deviation, minimum, and maximum error boundaries for all evaluated methodologies.

Methods	Statistical Results				
	Mean	Std. Dev.	Min	Max	Significance Value
LSTM	40.9320	4.4091	32.6711	50.3163	1.73E-06
PS-ANN	39.7342	8.5378	21.1220	56.3719	3.88E-06
SRNN	50.4756	21.9295	23.3462	83.2645	3.52E-06
Proposed Network	25.3116	3.9521	21.0738	36.6430	

Table 4. RMSE statistics for the test set of the Australian beer consumption time series

According to the comparative findings presented in Table 4, the proposed network consistently achieves the superior forecasting rank by exhibiting the lowest mean RMSE value (25.3116). In addition to providing the lowest average error, the proposed framework demonstrates remarkable structural stability, as evidenced by its minimal standard deviation score (3.9521).

Furthermore, the final column of Table 4 details the significance levels derived from the non-parametric hypothesis testing. The calculated p values for all competitive baselines (LSTM, PS-ANN, and SRNN) fall substantially below the standard significance threshold of 10^{-3}. These statistical indicators definitively confirm that the forecasting superiority of the newly introduced architecture over the alternative forecasting models is highly significant and statistically verified rather than anomalous.

Following the successful validation on economic and consumption metrics, a third empirical assessment is conducted on an environmental time series: the Ankara air pollution (SO_2) dataset. This environmental indicator consists of monthly records tracked continuously from March 1994 until April 2005. The chronological fluctuations and inherent distribution properties of this air quality index are depicted in the line graph shown in Fig. 5.

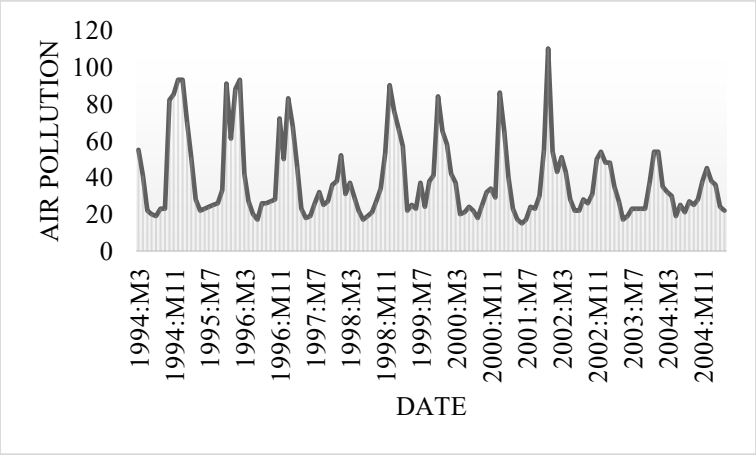

Fig. 5. The graph of the Ankara air pollution data time series

For the environmental analysis involving the Ankara air pollution (SO_2) dataset, the structural tuning configurations optimized for a seasonal cycle of twelve ($s = 12$) are systematically evaluated. Table 5 consolidates the most effective hyperparameter selections—specifically detailing the hidden unit dimensions, optimal lag parameters (p), and the targeted layer depths—that minimize out-of-sample error during the validation sequence.

The best hyperparameters of the methods (s=12)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers
LSTM	3	4	1
PS-ANN	12	2	1
SRNN	4	10	2
Proposed Network	-	1	1

Table 5. The best hyperparameters of the methods for the Ankara air pollution data time series

Following the validation phase, each network structure is subjected to verification on the independent test dataset. Table 6 summarizes the vital empirical criteria, presenting a comparative display of the mean, standard deviation, minimum, and maximum values of RMSE distributions gathered over multiple evaluation trials.

Methods	Statistical Results				
	Mean	Std. Dev.	Min	Max	Significance Value

LSTM	17.0311	9.6328	6.5062	36.3134	3.41E-05
PS-ANN	14.2774	6.5576	6.2641	27.3344	1.60E-04
SRNN	10.4540	1.7991	4.9260	14.2810	2.83E-04
Proposed Network	8.6426	0.7003	6.0602	8.8267	

Table 6. RMSE statistics for the test set of the Ankara air pollution data time series

The quantitative evidence summarized in Table 6 demonstrates that the proposed recurrent framework exhibits an exceptional forecasting capability, registering the lowest mean RMSE (8.6426) alongside the lowest standard deviation. An interesting nuance in the empirical behaviour occurs when evaluating the absolute minimum error metric, where the SRNN model achieves a slightly lower value (4.9260) compared to the proposed architecture (6.0602). However, relying solely on an isolated minimum metric does not represent overall predictive reliability. When factoring in the average model performance and architectural stability—evidenced by the remarkably compact standard deviation of the proposed framework—the introduced model establishes its clear superiority over LSTM, PS-ANN, and SRNN for this air quality series.

Furthermore, the statistical integrity of these results is evaluated through non-parametric hypothesis testing. The computed p-values for all competing methodologies are markedly lower than the conventional 10^{-3} threshold, which mathematically proves that the forecasting dominance of the proposed neural network is statistically significant and highly robust.

To further assess the model's robustness across macro-level utilities, a fourth empirical implementation is carried out on the Turkey electricity consumption dataset. This national energy series encompasses monthly observed metrics logged continuously between January 2002 and December 2012. The specific fluctuation profiles, trend behaviours, and seasonal dependencies characterizing this electricity consumption index are graphically presented in the line chart in Fig. 6.

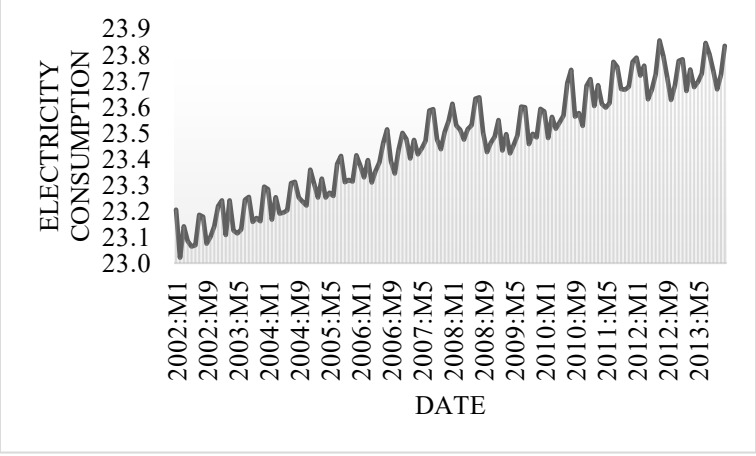

Fig. 6. The graph of Turkey's electricity consumption time series

For the final empirical verification, the forecasting capabilities of the competitive methodologies are evaluated using the national electricity consumption dataset for Turkey. Operating under a monthly seasonal cycle framework ($s = 12$), the validation sub-matrix is exploited to identify the ideal architectural layout for each model. The resulting optimal structural parameters—comprising hidden unit dimensions, layer depths, and ideal input lags (p)—are systematically documented in Table 7.

The best hyperparameters of the methods (s=12)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers
LSTM	5	3	1
PS-ANN	12	2	1
SRNN	1	9	2
Proposed Network	-	11	1

Table 7. The best hyperparameters of the methods for Turkey's electricity consumption time series

Following structural identification, the performance profiles of the initialized networks are verified on the independent test subset. To minimize any performance anomalies resulting from random parameter initialization, the evaluation metrics are aggregated across multiple distinct operational loops. Table 8 presents the resulting empirical distributions, showcasing the mean, standard deviation, minimum, and maximum RMSE values for each algorithm.

Methods	Statistical Results				
	Mean	Std. Dev.	Min	Max	Significance Value
LSTM	0.0679	0.0051	0.0643	0.0867	1.73E-06
PS-ANN	0.0628	0.0096	0.0467	0.0841	2.35E-06
SRNN	0.1666	0.0752	0.0642	0.2844	1.73E-06
Proposed Network	0.0440	0.0061	0.0366	0.0557	

Table 8. RMSE statistics for the test set of the Turkey electricity consumption time series

The quantitative findings summarized in Table 8 demonstrate the clear superiority of the newly introduced architecture over traditional forecasting models in national energy forecasting. The proposed framework secures the top performance rank by achieving the lowest mean RMSE (0.0440) along with the lowest absolute minimum error (0.0366) and lowest maximum error cap (0.0557). Concurrently, it exhibits remarkable stability and consistency across runs, as demonstrated by its low standard deviation score (0.0061).

To assess the statistical validity of this predictive superiority, a non-parametric Wilcoxon signed-rank test is conducted. The resulting significance metrics, reported in the final column of Table 8, fall substantially below the strict 10^{-3} standard threshold. These empirical indicators definitively demonstrate that the superior accuracy achieved by the proposed neural network architecture across the test partition is highly robust and statistically significant rather than a consequence of random initialization.

To provide a comprehensive empirical validation across diverse climatic and environmental metrics, a fifth application is carried out using the monthly air pressure dataset for Giresun City, Turkey. This geographical time series encompasses monthly records tracked continuously from January 1990 until November 2021. The specific historical fluctuations and cyclical dependencies characterizing this air pressure index are graphically presented in the line chart in Fig. 7.

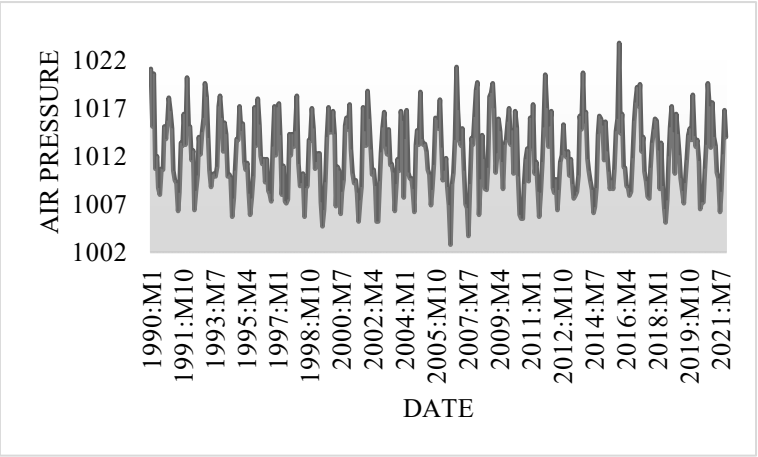

Fig. 7. The graph of the Giresun City in Turkey air pressure time series

Operating under a monthly seasonal cycle configuration ($s = 12$), the validation split is exploited to identify the ideal network layout for each competitive baseline. The resulting optimal structural parameters—comprising hidden layer unit counts, hidden layer depths, and ideal input lags (p)—are systematically documented in Table 9.

The best hyperparameters of the methods (s=12)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers
LSTM	1	4	1
PS-ANN	8	2	1
SRNN	5	9	2
Proposed Network	-	8	1

Table 9. The best hyperparameters of the methods for the Giresun City in Turkey air pressure time series

Following structural identification, the performance profiles of the initialized networks are verified on the independent test subset. To minimize performance anomalies resulting from random parameter initialization, the evaluation metrics are aggregated across 30 distinct operational loops using randomized initial weights and biases. Table 10 presents the resulting empirical distributions, showcasing the mean, standard deviation, minimum, and RMSE values for each algorithm.

The quantitative findings summarized in Table 10 demonstrate the clear superiority of the newly introduced architecture over traditional forecasting models in barometric time series forecasting. The proposed framework secures the top performance rank by achieving the lowest mean RMSE (1.9232) along with the lowest absolute minimum error (1.9172) and lowest maximum error cap (1.9651). Concurrently, it exhibits exceptional stability and consistency across runs, as demonstrated by its remarkably low standard deviation score (0.0086).

To assess the statistical validity of this predictive superiority, a non-parametric Wilcoxon signed-rank test is conducted. The resulting significance metrics, reported in the final column of Table 10, fall substantially below the strict 10^{-3}standard threshold. These empirical indicators definitively demonstrate that the superior accuracy achieved by the proposed neural network architecture across the test partition is highly robust and statistically significant rather than a consequence of random initialization bias.

Methods	Statistical Results				
	Mean	Std. Dev.	Min	Max	Significance Value

LSTM	2.0665	0.0842	1.8624	2.2038	3.18E-06
PS-ANN	2.6145	0.2705	2.3107	3.6074	1.73E-06
SRNN	3.4587	0.4779	2.0343	3.6932	1.73E-06
Proposed Network	1.9232	0.0086	1.9172	1.9651	

Table 10. RMSE statistics for the test set of the Giresun City in Turkey air pressure time series

The final experimental verification is conducted using the air temperature time series of Giresun City, Turkey. This environmental dataset consists of monthly observations monitored over a long-term horizon from January 1990 to November 2021. The cyclical fluctuations, clear seasonal peaks, and troughs inherent to this climate metric are graphically detailed in the line plot shown in Fig. 8.

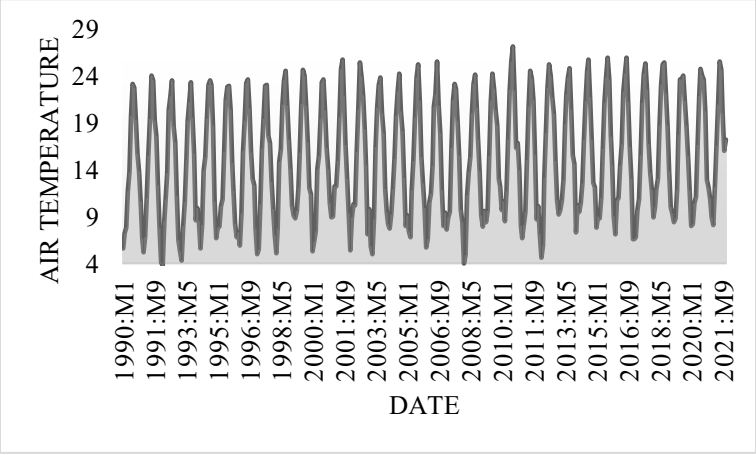

Fig. 8. The graph of the Giresun City in Turkey air temperature time series

Given the monthly nature of the climate indicators, the network architectures are optimized considering a seasonal period of twelve ($s = 12$). Utilizing the validation dataset partition, the structural configurations that minimize forecasting errors are systematically extracted. Table 11 summarizes these optimal hyperparameter choices for each evaluated framework, explicitly detailing the hidden layer counts, hidden unit allocations, and chosen lag horizons (p).

The best hyperparameters of the methods (s=12)

	The number of hidden layer units	Number of inputs (p)	The number of hidden layers
LSTM	5	5	1
PS-ANN	11	1	1
SRNN	3	12	2
Proposed Network	-	12	1

Table 11. The best hyperparameters of the methods for the Giresun City in Turkey air temperature time series

Following structural identification, the performance profiles of the finalized networks are verified on the independent test subset. To eliminate random initialization bias, each framework is trained over 30 independent loops using randomized initial weight vectors and bias matrices. Table 12 provides a comprehensive statistical breakdown—reporting the mean, standard deviation, minimum, and maximum values—of the calculated test RMSE distributions.

Methods	Statistical Results

	Mean	Std. Dev.	Min	Max	Significance Value
LSTM	1.6515	0.0754	1.5360	1.8800	1.73E-06
PS-ANN	2.6255	0.6163	1.7938	4.3134	1.73E-06
SRNN	2.7901	1.5769	1.5549	6.1693	1.73E-06
Proposed Network	1.4127	0.0303	1.3410	1.4343	

Table 12. RMSE statistics for the test set of the Giresun City in Turkey air temperature time series

The empirical metrics summarized in Table 12 demonstrate that the proposed architecture establishes clear predictive dominance over traditional time series methods. The proposed framework reports the lowest mean RMSE (1.4127) along with the lowest maximum error cap (1.4343). More importantly, it demonstrates outstanding structural consistency and robustness across random runs, achieving a remarkably small standard deviation score (0.0303). In contrast, alternative architectures such as the PS-ANN and SRNN demonstrate considerably higher variability, as evidenced by their high standard deviation values.

To ensure the validity of these outcomes, a non-parametric Wilcoxon signed-rank test is deployed as a rigorous statistical hypothesis test. The calculated p-values, presented in the final column of Table 12, are significantly lower than the standard 10^{-3} threshold across all competitive methods (LSTM, PS-ANN, and SRNN). These statistical indicators mathematically confirm that the predictive superiority delivered by the newly introduced neural network configuration is robust, reliable, and highly significant.

Conclusion and Discussion

This study introduces a novel recurrent artificial neural network (RNN) topology engineered specifically for forecasting seasonal time series without requiring any prior data decomposition. While the existing literature heavily relies on classical statistical models like SARIMA and ETS, or standard computational intelligence techniques that handle seasonality merely by adding lagged inputs, the proposed framework directly integrates seasonal smoothing logic and dynamic feedback loops into its architectural design.

The core contributions of this research, as outlined in the introduction, were successfully validated through a series of comprehensive empirical applications across six diverse real-world datasets: logarithmic air passenger numbers, Australian beer consumption, Ankara air pollution (SO_2) national electricity demand in Turkey, and barometric pressure and temperature profiles for Giresun City.

Unlike traditional feedforward multi-layer perceptron or vanilla RNNs, the introduced network incorporates a dual-component design featuring a multiplicative neuron model alongside a seasonal recurrent smoothing mechanism. This structural design allows the network to capture complex non-linear dynamics and seasonal dependencies simultaneously. The empirical results across all six distinct datasets confirm that incorporating the network's historical forecasts at the seasonal period (s) significantly enhances predictive performance.

Standard optimization metrics such as MSE and RMSE assign equal weights to all historical data points, ignoring the fact that recent observations carry more critical signals regarding immediate future trends. The introduced WMSE objective function addresses this limitation by allocating progressively higher weights to the most recent data points. Training the network with WMSE successfully guided the optimization engine toward configurations that adapt rapidly to structural trend shifts, as demonstrated by the consistently low mean error values achieved on the out-of-sample test subsets.

To overcome the localized optimization traps common to derivative-based algorithms, an adaptive PSO mechanism was developed. By integrating structural early stopping rules and dynamic re-starting criteria, the learning engine avoided overfitting and bypassed local minima traps. This optimization stability is demonstrated by the outstandingly low standard

deviation scores achieved by the proposed method across 30 independent runs, significantly outperforming highly volatile architectures like SRNN and PS-ANN.

In all six validation scenarios, the proposed architecture consistently secured the lowest mean RMSE scores compared to robust deep learning baselines like LSTM and recurrent frameworks like SRNN. Non-parametric hypothesis validation via the Wilcoxon signed-rank test produced significance levels substantially below the 10^{-3} threshold in every case study. This mathematically confirms that the predictive superiority of the proposed framework is highly robust and statistically significant across economic, environmental, and utility forecasting applications.

In summary, this research bridges a critical gap between classical exponential smoothing philosophies and modern deep learning methodologies. The proposed network provides a highly stable, accurate, and mathematically sound alternative for complex seasonal forecasting problems. Future research trajectories may explore expanding this recurrent framework to accommodate multiple and nested seasonal periods.

Credit authorship contribution statement

Eren Bas: Conceptualization, Methodology, Data curation, Investigation, Software, Writing – original draft. **Crina Grosan:** Conceptualization, Methodology, Supervision, Validation, Writing – original draft. **Erol Egrioglu:** Writing – original draft, Supervision, Validation, Methodology, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The raw data required to reproduce the above findings are available to download from [https://github.com/erole1977/SRANN].

Funding

No Funding.

References

1. Hyndman, R.J., Koehler, A.B., Ord J.K., & Snyder, R.D. Forecasting with exponential smoothing. Springer Berlin Heidelberg (2008).
2. Martínez, F., Frías, M.P., Pérez-Godoy, M.D., & Rivera, A.J., Dealing with seasonality by narrowing the training set in time series forecasting with kNN. Expert Systems with Applications 103, 38-48 (2018).
3. Baragona, R., Battaglia, F., & Cucina, D. Periodic autoregressive models for time series with integrated seasonality. Journal of Statistical Computation and Simulation, 91(4), 694-712 (2021).
4. Bajalinov, E., & Duleba, S. Seasonal time series forecasting by the Walsh-transformation based technique. Central European Journal of Operations Research 28(3): 983-1001 (2020).
5. Li, X., Petropoulos, F., & Kang, Y. Improving forecasting by subsampling seasonal time series. International Journal of Production Research 61(3): 976-992 (2022).
6. Naim, I., Mahara, T., & Khan, S.A. Ranking of univariate forecasting techniques for seasonal time series using analytical hierarchy process. International Journal of Industrial and Systems Engineering, 35(2), 196-215 (2020).
7. Chen, K.Y., & Wang, C.H. A hybrid SARIMA and support vector machines in forecasting the production values of the machinery industry in Taiwan. Expert Systems with Applications 32(1), 254-264 (2007).

8. Pai, P.F., Lin, K.P., Lin, C.S., & Chang, P.T. Time series forecasting by a seasonal support vector regression model. Expert Systems with Applications 37(6), 4261-4265 (2010).
9. Štěpnička, M., Cortez, P., Donate, J.P., & Štěpničková, L. Forecasting seasonal time series with computational intelligence: On recent methods and the potential of their combinations. Expert Systems with Applications, 40(6), 1981-1992 (2013).
10. Laib, O., Khadir, M.T., & Mihaylova, L. Toward efficient energy systems based on natural gas consumption prediction with LSTM Recurrent Neural Networks. Energy 177, 530-542 (2019).
11. Liu, S., Ji, H., & Wang, M.C. Nonpooling convolutional neural network forecasting for seasonal time series with trends. IEEE Transactions on Neural Networks and Learning Systems 31(8) 8823990, 2879-2888 (2020).
12. Di Martino, F., & Sessa, S. Seasonal time series forecasting by F1-fuzzy transform. Sensors (Switzerland) 19(16), 3611 (2019).
13. Valente, J.M., & Maldonado, S. SVR-FFS: A novel forward feature selection approach for high-frequency time series forecasting using support vector regression. Expert Systems with Applications 160, 113729 (2020).
14. Dudek, G., & Pełka, P. Pattern similarity-based machine learning methods for mid-term load forecasting: A comparative study. Applied Soft Computing 104, 107223 (2021).
15. Jaihuni, M., Basak, J.K, Khan, F., Okyere, F.G., Sihalath, T., Bhujel, A., Park, J., Lee, D.H., & Kim, H.T. A novel recurrent neural network approach in forecasting short term solar irradiance. ISA Transactions 121, 63-74 (2022).
16. Bahrami, M., Khashei, M., & Amindoust, A. A parallel-series hybridization of seasonal intelligent based statistical model for demand forecasting. Journal of Modelling in Management. 17(4), 1126-1143 (2021).
17. Farsi, M., Hosahalli, D., Manjunatha, B.R., Gad, I., Atlam, E.S., Ahmed, A., Elmarhomy, G., Elmarhoumy, M., & Ghoneim, O.A. Parallel genetic algorithms for optimizing the SARIMA model for better forecasting of the NCDC weather data. Alexandria Engineering Journal 60(1), 1299-1316 (2021).
18. Zhou, W., Cheng, Y., Ding, S., Chen, L., & Li, R. A grey seasonal least square support vector regression model for time series forecasting. ISA Transactions 114, 82-98 (2021).
19. Zhou, W., Pan, J., Tao, H., Ding, S., Chen, L., & Zhao, X. A novel grey seasonal model based on cycle accumulation generation for forecasting energy consumption in China. Computers and Industrial Engineering 163, 107725 (2022).
20. Egrioglu, E., Aladag, C.H., Yolcu, U., Basaran, M.A., & Uslu, V.R. A new hybrid approach based on SARIMA and partial high order bivariate fuzzy time series forecasting model. Expert Systems with Applications 36(4), 7424-7434 (2009).
21. Liu, H.T, & Wei, M.L. An improved fuzzy forecasting method for seasonal time series. Expert Systems with Applications 37(9), 6310-6318 (2010).
22. Bulut, E. Modeling seasonality using the fuzzy integrated logical forecasting (FILF) approach. Expert Systems with Applications 41 (4 PART 2), 1806-1812 (2014).
23. Saleena, A.J., & Jessy John, C. A new hybrid model based on triple exponential smoothing and fuzzy time series for forecasting seasonal time series. Springer Proceedings in Mathematics and Statistics 307, 179-190 (2020).
24. Nguyen, L., & Novák, V. Forecasting seasonal time series based on fuzzy techniques. Fuzzy Sets and Systems 361, 114-129 (2019).
25. Ho, C.H., Chang, P.T., Hung, K.C., & Lin, K.P. Developing intuitionistic fuzzy seasonality regression with particle swarm optimization for air pollution forecasting. Industrial Management and Data Systems 119(3), 561-577 (2019).
26. Aladag, C.H., Yolcu, U., & Egrioglu, E. A new multiplicative seasonal neural network model based on particle swarm optimization. Neural Processing Letters 37(3), 251-262 (2013).
27. Yadav, R.N., Kalra, P.K., & John, J. Time series prediction with single multiplicative neuron model. Applied Soft Computing 7, 1157-1163 (2007).