Baghdad Science Journal

Volume 22 | Issue 8 Article 23

8-27-2025

A New Approach for Selecting Optimal Neuron Number in Input Layer of Generalized Regression Neural Network Model

Muhammad Muhajir
Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Islam Indonesia, Yogyakarta, Indonesia, mmuhajir@uii.ac.id

Hermansah Hermansah
Department of Mathematics Education, Faculty of Teacher Training and Education, Universitas Riau Kepulauan, Kepulauan Riau, Indonesia, hermansah@fkip.unrika.ac.id

Aditya Pandu Wicaksono
Department of Accounting, Faculty of Business and Economics, Universitas Islam Indonesia, Yogyakarta, Indonesia, aditya.pandu@uii.ac.id

Lathifah Aliya Pratiwi
Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Islam Indonesia, Yogyakarta, Indonesia, 19611180@alumni.uii.ac.id

How to Cite this Article

Muhajir, Muhammad; Hermansah, Hermansah; Wicaksono, Aditya Pandu; and Pratiwi, Lathifah Aliya (2025) "A New Approach for Selecting Optimal Neuron Number in Input Layer of Generalized Regression Neural Network Model," *Baghdad Science Journal*: Vol. 22: Iss. 8, Article 23.
DOI: https://doi.org/10.21123/2411-7986.5034

BAGHDAD SCIENCE JOURNAL 2025;22(8):2738–2751

Scan the QR to view the full-text article on the journal website

RESEARCH ARTICLE

A New Approach for Selecting Optimal Neuron Number in Input Layer of Generalized Regression Neural Network Model

Muhammad Muhajir [1,*], **Hermansah** [2], **Aditya Pandu Wicaksono** [3], **Lathifah Aliya Pratiwi** [1]

[1] Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Islam Indonesia, Yogyakarta, Indonesia
[2] Department of Mathematics Education, Faculty of Teacher Training and Education, Universitas Riau Kepulauan, Kepulauan Riau, Indonesia
[3] Department of Accounting, Faculty of Business and Economics, Universitas Islam Indonesia, Yogyakarta, Indonesia

ABSTRACT

This study aimed to determine the input layer neurons for the Generalized Regression Neural Network (GRNN) model by using various classical methods, namely 1) Partial Autocorrelation Function (PACF), 2) frequency-based method, 3) frequency and Forward Selection (FS), 4) frequency and Backward Elimination (BE), 5) frequency and step-based methods, and 6) frequency method combined with the Least Absolute Shrinkage and Selection Operator (LASSO). These classical methods were combined with various parameters within GRNN, including smoothing parameters, forecasting strategies, and transformations. The most accurate model, with the lowest RMSE, MAE, MAPE, and SMAPE values, resulted from the combination of frequency and BE, rolling origin, MIMO, and additive transformation parameters. Additionally, a further approach is proposed by using binary dummy neurons in the input layer. Each best model obtained from the classical approach is given additional neurons in the input layer in the form of binary dummies. Thus, this approach combines the autoregressive lag approach to capture stochastic seasonal patterns and binary dummies to capture deterministic seasonal patterns. The empirical study results show that the GRNN model with the frequency and stepwise approach, and binary dummies, provides the best results. This is demonstrated by the lowest RMSE, MAE, MAPE, and SMAPE values. The results of this study also indicate that the forecasting accuracy of the proposed GRNN model significantly differs from the exponential smoothing, ARIMA, FFNN, and GRNN models. Based on these results, the approach in this study is an effective way to improve forecasting accuracy.

Keywords: Backward elimination, Binary dummy, Forward selection, Generalized regression neural network, Least absolute shrinkage, Selection operator

Introduction

Time series forecasting can be classified into linear and nonlinear methods. According to,[1] linear forecasting methods perform well on linear time series data but are less effective in modeling nonlinear and complex time series data. A more flexible approach for modeling both linear and nonlinear relationships is the Neural Networks (NN) model.[2] states that the NN model is a machine designed to simulate how the human brain works in performing certain functions or tasks. The NN model consists of several information-processing elements called neurons. The neurons in the NN model are arranged in groups, called layers. The arrangement of neurons within layers and the pattern of connections within and between layers is

Received 29 May 2024; revised 6 October 2024; accepted 8 October 2024.
Available online 27 August 2025

* Corresponding author.
E-mail addresses: mmuhajir@uii.ac.id (M. Muhajir), hermansah@fkip.unrika.ac.id (Hermansah), aditya.pandu@uii.ac.id (A. P. Wicaksono), 19611180@alumni.uii.ac.id (L. A. Pratiwi).

https://doi.org/10.21123/2411-7986.5034

referred to as network architecture. This architecture is one of the key characteristics that distinguish NN models. In general, three layers form the NN model: input layer, hidden layer, and output layer.[3–5] Research[3] focuses on how NN models are used to evaluate and improve the efficiency of energy systems involving solar panels, transformers, and energy consumption. Research[4] focuses on how NN models can be used to monitor and predict the performance of photovoltaic panel systems. Meanwhile, research[5] focuses on how NN models are applied in projecting air pollution levels and can be beneficial for air quality management.

In practice, NN models contain a limited number of parameters (weights). How to obtain an appropriate NN model, that is, how to determine the right combination between the number of neurons in the input layer and the hidden layer (which has implications for the number of optimal parameters).[6–8] Several references related to the description and implementation of NN modeling can be found in.[9–11] In[9] focus is on the use of NN models for nonlinear time series forecasting with applications in cigarette sales.[10] integrates NN models and stochastic pattern analysis for time series forecasting.[11] discusses the optimization of NN models for time series forecasting. Additionally, one NN modeling approach available in the literature is the method introduced by.[12] They introduced the Generalized Regression Neural Networks (GRNN) model for time series forecasting. In their approach, neurons in the input layer are created based on the frequency of time series data. The frequency attribute represents the quantity of time series data within a certain period. If the frequency of the time series data is m, then m successive lags starting from lag 1 are used. According to,[12] seasonal patterns can be more easily captured through the time series data frequency approach. However, in the work of,[12] deterministic seasonal dummy factors were not considered. According to,[13] deterministic seasonal dummies need to be created as neurons in the input layer to capture seasonal patterns, not just based on autoregressive lags. The econometric theory also suggests that stochastic and deterministic seasonality need to be modeled differently to achieve accurate forecasting.

The GRNN model has also undergone significant development in previous studies. Some of them include[14] which discusses recent developments in the GRNN model with the addition of regularization to improve forecasting accuracy on time series data;[15] which explores improvements to the GRNN model used for financial data prediction, demonstrating advances in forecasting techniques using the GRNN model;[16] which provides a comprehensive review of the application of GRNN models for air quality forecasting, reflecting recent developments in GRNN applications to environmental issues; and[17] which develops a hybrid GRNN model for multivariate time series analysis, potentially expanding the application of GRNN in various domains. However, they introduced GRNN model forecasting with input variables using external data, without considering autoregressive lags of the time series data to be predicted.

Based on the above studies, this research proposes an extension to the GRNN model for time series forecasting. In this study, the neurons in the input layer of the GRNN model will be created based on the frequency of time series data (autoregressive lags) and deterministic seasonal dummies. The number of neurons in the hidden layer will be determined as half the number of neurons in the input layer. Additionally, several classical approaches are proposed for selecting the autoregressive lags in the GRNN model, including the Partial AutoCorrelation Function (PACF), Forward Selection (FS), Backward Elimination (BE), stepwise, and Least Absolute Shrinkage and Selection Operator (LASSO) methods. Therefore, this research has two main novelties. First, it selects the autoregressive lags in the GRNN model using several classical approaches. Second, it uses deterministic seasonal dummies in the form of binary dummies as neurons in the input layer of the GRNN model. Moreover, each classical approach is combined with various parameters in the GRNN model, such as smoothing parameters, forecasting strategies, and transformations. The smoothing parameter determination methods considered in this study are the rolling origin and fixed origin methods. The forecasting strategies considered are the recursive method and Multiple Input Multiple Output (MIMO). The transformations considered are the additive and multiplicative methods. The details of each parameter have been discussed in previous studies.[18–20]

Subsequently, the performance of the proposed method is compared with several methods available in the literature, namely the Exponential Smoothing (ETS) model described in,[21] the ARIMA model described in,[22] the FFNN model described in,[23] and the GRNN model described in.[12] Meanwhile, the accuracy comparison will be measured using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percent Error (MAPE), and Symmetric Mean Absolute Percentage Error (SMAPE). The proposed method is also applied to forecast real data, namely the monthly data on the number of deaths due to accidents in the USA and the monthly inflation rate data in Indonesia. Forecasting the number of deaths due to accidents in the USA is quite popular because of its complex pattern and has been discussed in several studies. Meanwhile, forecasting the inflation rate data in Indonesia can be useful for estimating what will happen in the future and can contribute ideas to policymakers in determining future policies.

This research is structured as follows: In Section 1 explain the background and contributions of this research. Section 2 briefly describes GRNN modeling for time series forecasting. Section 3 presents an empirical study with two real data sets. Conclusions are provided in Section 4.

Materials and methods

GRNN model

The GRNN represents a variation of the radial basis neural network, originally introduced by,[24] with its primary application lying in classification and regression tasks. The GRNN consists of three distinct layers, namely the input, hidden, and output layers. In the hidden layer, radial basis neurons reside, and their centers correspond to the training examples. Typically, the radial basis function takes the form of the multivariate Gaussian function:

$$G(x,\ x_i) = exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{1}$$

In Eq. (1), x_i and σ denote the center and the smoothing parameter, respectively, with x representing the input vector. The output of a hidden layer neuron depends on the proximity of the input vector to the center, adjusted by the smoothing parameter.

To compute the output for an input pattern x from a training set consisting of n training patterns and their associated targets $\{x_1,\ x_2, \ldots,\ x_n\}$ and $\{y_1,\ y_2, \ldots,\ y_n\}$, two primary steps were included. Firstly, the hidden layer generated weights contingent on the proximity of x to the training patterns:

$$w_i = \frac{exp\left(-\frac{\|x-x_i\|^2}{2\sigma^2}\right)}{\sum_{j=1}^{n} exp\left(-\frac{\|x-x_j\|^2}{2\sigma^2}\right)} \tag{2}$$

These weights are collectively summed to one, representing each training pattern's contribution to the final result. Secondly, the output layer computes the output as:

$$\hat{y} = \sum_{i=1}^{n} w_i\, y_i \tag{3}$$

This resulted in a weighted average of the training targets, with weights reflecting the proximity of the input to the training patterns, favoring closer patterns. The smoothing parameter σ regulated the degree of smoothing in the output. A large σ led to small, similar weights for all targets, leading to an output close to the mean of the targets. However, a small σ assigned significant weights only to targets with patterns closely matching the input.

GRNN model algorithm

This section describes the procedure of the GRNN modeling algorithm using some classical approaches, assuming that $y(t)$ is the time series data to be predicted. The algorithms used closely follow some improvements by[12] and.[13] Furthermore, the steps contained in the algorithm are explained as follows:

1. Preprocessing
The initial process begins by identifying whether the data $y(t)$ contains a trend component. If the data $y(t)$ is identified as having a trend component, first differencing is performed to remove this trend. Once the data is free from the trend, the next step is to identify whether the data has a seasonal component. If a seasonal component is identified in the data, a deterministic seasonal dummy variable is created to be used as input in the model's input layer. Finally, the processed data is linearly normalized within the range [0, 1].

2. Autoregressive lags
Autoregressive lags are determined based on the frequency of the data $y(t)$. If the frequency of the data $y(t)$ is equal to m, then consecutive lags from 1 to m are considered potential autoregressive lags/input variables. For example, for quarterly and monthly data, lags 1:4 and 1:12 are considered potential autoregressive lags/input variables, respectively. Additionally, autoregressive lags as neurons in the input layer are determined using several classical approaches, namely through the Partial Auto-Correlation Function (PACF), frequency, frequency and Forward Selection (FS), frequency and Backward Elimination (BE), frequency and stepwise, as well as frequency and Least Absolute Shrinkage and Selection Operator (LASSO). In the latter four methods, significant autoregressive lags/input variables are determined using variable selection methods.

3. Number of neurons
The number of neurons in the hidden layer is determined as half of the number of input variables (neurons in the input layer).

4. Selecting the smoothing parameter

The determination of the smoothing parameter value considers the rolling origin and fixed origin methods. Strategies discussing the rolling origin and fixed origin methods can be found in.[12]

5. Multi-step ahead strategy

Each strategy in step 4 is combined with several parameters in the GRNN model, namely the smoothing parameter, forecasting strategy, and transformation. The considered forecasting strategies are the recursive method and Multiple Input Multiple Output (MIMO). The considered transformations are the additive and multiplicative methods.

Results and discussion

The purpose of this research is to determine the optimal number of input layer neurons for the Generalized Regression Neural Network (GRNN) model by using several classical approaches and different parameter combinations. Firstly, it aims to identify the most effective classical approach in selecting input layer neurons for the GRNN model. Secondly, it seeks to test the effectiveness of parameter combinations in the GRNN model, including the smoothing parameter, forecasting strategy, and transformation, to determine the model with the best accuracy. Thirdly, it aims to introduce and evaluate a new approach that utilizes additional binary neurons in the input layer, This approach combines autoregressive lags and binary dummy variables to capture both stochastic and deterministic seasonal patterns. Lastly, it aims to compare the forecasting accuracy of the proposed GRNN model with existing models such as exponential smoothing, ARIMA, Feed-Forward Neural Network (FFNN), and other GRNN models. The objective is to demonstrate improvements in forecasting accuracy.

Accidental deaths

In this study, the first case study was conducted using data on the number of deaths due to accidents in the USA. The data used is monthly data taken from January 1973 to June 1979. This data has been discussed in several studies and is quite popular due to its complex pattern. Additionally, this data is assumed to fit the characteristics of the proposed method. The data is divided into two parts: training data and testing data. The training data consists of data from January 1973 to December 1978, while the testing data consists of data from January 1979 to June 1979. This data can be accessed and found on the official website of the National Highway Traffic Safety Administration (NHTSA).[25]

In this study, several classical approaches are proposed for determining the neurons in the input layer of the General Regression Neural Network (GRNN) model. First, the neurons in the input layer are determined using the Partial Auto-Correlation Function (PACF) approach. Second, the neurons in the input layer are determined using the frequency approach, which considers the time and frequency attributes of the time series data. The time attribute indicates the time unit of each observation point, while the frequency attribute indicates the quantity of data within a certain period, usually defined annually. For example, monthly data has a frequency of 12, quarterly data has a frequency of 4, four-month data has a frequency of 3, semi-annual data has a frequency of 2, and annual data has a frequency of 1. Third, the determination of neurons in the input layer is selected using the frequency and Forward Selection (FS) approach. Fourth, the determination of neurons in the input layer is selected using the frequency and Backward Elimination (BE) approach. Fifth, the determination of neurons in the input layer is selected using the frequency and stepwise approach. Sixth, the determination of neurons in the input layer is selected using the frequency and Least Absolute Shrinkage and Selection Operator (LASSO) approach. Each classical approach is combined with several parameters in the GRNN model, including smoothing parameters, forecasting strategies, and transformations. The smoothing parameter values are determined using the rolling origin and fixed origin methods. The forecasting strategies considered include the recursive method and Multiple Input Multiple Output (MIMO). The transformations considered include the additive and multiplicative methods. The hyperparameter values used are determined based on the results of.[12]

The results of each parameter combination for the GRNN model using the PACF approach are shown in Table 1. The results for the GRNN model using the frequency approach are shown in Table 2. The results for the GRNN model using the frequency and FS approach are shown in Table 3. The results for the GRNN model using the frequency and BE approach are shown in Table 4. The results for the GRNN model using the frequency and stepwise approach are shown in Table 5. The results for the GRNN model using the frequency and LASSO approach are shown in Table 6. The best model is selected based on the smallest RMSE, MAE, MAPE, and SMAPE values among the various models built.

Based on the modeling results in Table 1, it can be empirically seen that the GRNN model provides the smallest error with the parameter combination of PACF, rolling origin, MIMO, and multiplicative transformation. Meanwhile, in Table 2, the best GRNN

Table 1. Comparison of the results of combining GRNN model parameters and several models for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
PACF, rolling origin, recursive, no transformation	709.9820	614.7765	6.592633	6.870116
PACF, fixed origin, recursive, no transformation	715.3952	617.0577	6.609311	6.897377
PACF, rolling origin, MIMO, no transformation	564.5140	482.8702	5.268065	5.453042
PACF, fixed origin, MIMO, no transformation	569.8324	496.1738	5.402179	5.585038
PACF, rolling origin, recursive, additive transformation	687.7312	573.7833	6.169700	6.435547
PACF, fixed origin, recursive, additive transformation	688.6903	573.5240	6.165028	6.428075
PACF, rolling origin, MIMO, additive transformation	544.3736	416.1514	4.543474	4.715519
PACF, fixed origin, MIMO, additive transformation	574.4101	458.0508	4.978384	5.172186
PACF, rolling origin, recursive, multiplicative transformation	643.3624	548.5627	5.918375	6.147855
PACF, fixed origin, recursive, multiplicative transformation	727.3667	608.5760	6.549740	6.841279
PACF, rolling origin, MIMO, multiplicative transformation	524.9925	389.0132	4.243316	4.397312
PACF, fixed origin, MIMO, multiplicative transformation	602.9755	494.7956	5.369198	5.575885

Table 2. Comparison of the results of combining GRNN model parameters with input layer neurons using the frequency method for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
Frequency, rolling origin, recursive, no transformation	409.5742	345.5921	3.773155	3.872991
Frequency, fixed origin, recursive, no transformation	410.4594	344.6501	3.755772	3.856906
Frequency, rolling origin, MIMO, no transformation	406.9522	342.1923	3.729924	3.829330
Frequency, fixed origin, MIMO, no transformation	406.9522	342.1923	3.729924	3.829330
Frequency, rolling origin, recursive, additive transformation	307.8812	253.0103	2.756263	2.793664
Frequency, fixed origin, recursive, additive transformation	311.9937	255.2490	2.782325	2.824921
Frequency, rolling origin, MIMO, additive transformation	290.5309	229.4725	2.495509	2.530309
Frequency, fixed origin, MIMO, additive transformation	290.5824	229.4583	2.495264	2.530017
Frequency, rolling origin, recursive, multiplicative transformation	309.7977	253.7515	2.753546	2.787782
Frequency, fixed origin, recursive, multiplicative transformation	315.2464	258.2687	2.807541	2.849007
Frequency, rolling origin, MIMO, multiplicative transformation	293.2696	231.0252	2.502836	2.535675
Frequency, fixed origin, MIMO, multiplicative transformation	293.6085	231.0389	2.502571	2.535302

model is obtained through a parameter combination of frequency, fixed origin, MIMO, and multiplicative transformation. In Table 3, the best GRNN model is obtained through a parameter combination of frequency and FS, rolling origin, MIMO, and multiplicative transformation. In Table 4, the best GRNN model is obtained through a parameter combination of frequency and BE, rolling origin, MIMO, and additive transformation. In Table 5, the best GRNN model is obtained through a parameter combination of frequency and stepwise, rolling origin, recursive, and additive transformation. In Table 6, the best GRNN model is obtained through a parameter combination of frequency and LASSO, rolling origin, recursive, and multiplicative transformation. The best model is selected based on the smallest error metric. Overall, the GRNN model with the parameter combination of frequency and BE, rolling origin, MIMO, and additive transformation is the most accurate model with the lowest RMSE, MAE, MAPE, and SMAPE values.

Table 3. Comparison of the results of combining GRNN model parameters with neurons in the input layer via frequency and FS method for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
FS, rolling origin, recursive, no transformation	376.1982	338.4367	3.757855	3.817981
FS, fixed origin, recursive, no transformation	467.2387	378.6923	4.141043	4.270128
FS, rolling origin, MIMO, no transformation	397.1495	328.0769	3.569796	3.663670
FS, fixed origin, MIMO, no transformation	1051.306	803.2445	8.840320	9.083745
FS, rolling origin, recursive, additive transformation	392.8880	318.3445	3.435093	3.511580
FS, fixed origin, recursive, additive transformation	299.6357	238.7202	2.583006	2.623696
FS, rolling origin, MIMO, additive transformation	291.9438	231.0427	2.516775	2.546658
FS, fixed origin, MIMO, additive transformation	498.1187	387.0953	4.234266	4.364239
FS, rolling origin, recursive, multiplicative transformation	316.9674	245.2819	2.718285	2.767718
FS, fixed origin, recursive, multiplicative transformation	321.1058	264.0585	2.868777	2.914808
FS, rolling origin, MIMO, multiplicative transformation	288.0549	231.0068	2.503299	2.527631
FS, fixed origin, MIMO, multiplicative transformation	1208.724	909.6647	9.966224	10.59156

Table 4. Comparison of the results of combining GRNN model parameters with neurons in the input layer via frequency and BE method for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
BE, rolling origin, recursive, no transformation	370.5660	298.3767	3.274228	3.357548
BE, fixed origin, recursive, no transformation	392.7925	327.5897	3.592025	3.678446
BE, rolling origin, MIMO, no transformation	395.0538	324.3461	3.526581	3.619213
BE, fixed origin, MIMO, no transformation	406.7469	339.9487	3.702804	3.801881
BE, rolling origin, recursive, additive transformation	292.3550	239.7750	2.607317	2.639740
BE, fixed origin, recursive, additive transformation	291.6519	235.3825	2.548808	2.585541
BE, rolling origin, MIMO, additive transformation	282.6505	224.0709	2.429304	2.460385
BE, fixed origin, MIMO, additive transformation	754.1560	550.0543	6.013864	6.287733
BE, rolling origin, recursive, multiplicative transformation	290.6687	240.8437	2.615176	2.637529
BE, fixed origin, recursive, multiplicative transformation	292.5794	229.7534	2.508365	2.549210
BE, rolling origin, MIMO, multiplicative transformation	287.1174	227.3108	2.455321	2.484268
BE, fixed origin, MIMO, multiplicative transformation	955.9835	714.6564	7.733571	8.120301

Table 5. Comparison of the results of combining GRNN model parameters with input layer neurons using frequency and stepwise method for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
Stepwise, rolling origin, recursive, no transformation	502.8382	444.3461	4.865029	5.005971
Stepwise, fixed origin, recursive, no transformation	504.9673	442.2106	4.849423	4.997548
Stepwise, rolling origin, MIMO, no transformation	406.9533	342.1933	3.729936	3.829343
Stepwise, fixed origin, MIMO, no transformation	407.4665	342.7141	3.735759	3.835449
Stepwise, rolling origin, recursive, additive transformation	327.6593	261.7070	2.834338	2.877329
Stepwise, fixed origin, recursive, additive transformation	329.6950	263.4534	2.853903	2.898082
Stepwise, rolling origin, MIMO, additive transformation	382.1196	293.6514	3.218788	3.297918
Stepwise, fixed origin, MIMO, additive transformation	379.9672	281.0378	3.065087	3.138579
Stepwise, rolling origin, recursive, multiplicative transformation	349.9429	281.1396	3.045904	3.098482
Stepwise, fixed origin, recursive, multiplicative transformation	349.6249	277.7384	3.006767	3.053012
Stepwise, rolling origin, MIMO, multiplicative transformation	382.3682	296.4796	3.245580	3.323304
Stepwise, fixed origin, MIMO, multiplicative transformation	382.6730	295.1734	3.228731	3.305582

The empirical results also show that the GRNN model using the frequency and BE approach is increasingly accurate in determining the neurons in the input layer. On the other hand, this approach is used to obtain optimal neurons following the principle of parsimony. Rolling origin is an optimization method used to obtain smoothing parameter values that minimize forecasting accuracy metrics. The rolling origin method often provides better smoothing parameter values compared to the fixed origin method. Meanwhile, the MIMO approach is used for multi-step ahead forecasting and provides better accuracy in the first case study of this research. On the other hand, the additive transformation approach is used to handle trends and seasonal patterns. This approach preprocesses and transforms time series data to improve forecasting accuracy. In the first case study of this research, the additive transformation approach provides better accuracy compared to the multiplicative transformation approach.

Table 6. Comparison of the results of combining GRNN model parameters with neurons in the input layer via frequency and LASSO method for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
LASSO, rolling origin, recursive, no transformation	524.5195	467.6025	5.113271	5.261883
LASSO, fixed origin, recursive, no transformation	531.0631	465.0677	5.105445	5.270498
LASSO, rolling origin, MIMO, no transformation	444.3687	378.4995	4.148260	4.261248
LASSO, fixed origin, MIMO, no transformation	584.6139	465.5895	5.033009	5.211662
LASSO, rolling origin, recursive, additive transformation	410.6028	308.4574	3.346757	3.431368
LASSO, fixed origin, recursive, additive transformation	419.0369	310.3657	3.363384	3.450402
LASSO, rolling origin, MIMO, additive transformation	393.2072	289.2256	3.153711	3.232661
LASSO, fixed origin, MIMO, additive transformation	393.6802	289.9694	3.162978	3.242678
LASSO, rolling origin, recursive, multiplicative transformation	367.7770	287.7060	3.109221	3.163186
LASSO, fixed origin, recursive, multiplicative transformation	367.7776	287.7066	3.109227	3.163192
LASSO, rolling origin, MIMO, multiplicative transformation	394.4957	292.8938	3.185381	3.261408
LASSO, fixed origin, MIMO, multiplicative transformation	403.9807	305.4434	3.334918	3.420445

Table 7. Comparison of results between the GRNN model and several models for data on the number of deaths due to accidents in the United States.

	RMSE	MAE	MAPE	SMAPE
GRNN-PACF-dummy	67.12593	36.52342	0.420418	0.420479
GRNN-Frequency-dummy	40.59484	22.93888	0.265000	0.264980
GRNN-Frequency-FS-dummy	57.62788	35.19532	0.419549	0.419220
GRNN-Frequency-BE-dummy	42.50247	23.34174	0.277160	0.277243
GRNN-Frequency-Stepwise-dummy	32.47380	17.98457	0.207064	0.207139
GRNN-Frequency-LASSO-dummy	51.33380	30.76869	0.354217	0.354572
ETS	262.6980	202.6369	2.322380	2.320347
ARIMA	285.3613	200.9519	2.348831	2.375394
FFNN	68.55017	49.99667	0.580383	0.580134
GRNN	307.8812	253.0103	2.756263	2.793664

Furthermore, an advanced approach is proposed using binary dummy neurons in the input layer. For each best model obtained in Tables 1 to 6, additional neurons are created in the input layer in the form of binary dummies. Binary dummy neurons (S-1, where S is the frequency length) have been shown to capture deterministic seasonal patterns well, based on previous studies. Here, the neurons in the input layer apply two approaches: First, the neurons in the input layer are determined based on autoregressive lags with several classical approaches. Second, binary dummy neurons are created as deterministic seasonal dummies. The econometric theory also suggests that stochastic seasonality and deterministic seasonality need to be modeled differently to achieve accurate forecasting. Therefore, this research combines both approaches, where the autoregressive lag approach is used to capture stochastic seasonal patterns, and binary dummies are used to capture deterministic seasonal patterns. Additionally, the performance of the proposed GRNN model is also compared with several models available in the literature, such as the Exponential Smoothing (ETS) model described in,[17] the ARIMA model described in,[18] the FFNN model described in,[19] and the GRNN model described in.[12] The results of the empirical comparison can be seen in Table 7 for each GRNN model approach compared with several models available in the literature. The best model is selected based on the smallest RMSE, MAE, MAPE, and SMAPE values.

The empirical study results show that the GRNN model with the frequency, stepwise, and binary dummy approach provides the best results. This is demonstrated by the lowest RMSE, MAE, MAPE, and SMAPE values. The approach using the frequency of time series data has proven effective in capturing stochastic seasonal patterns better. Meanwhile, the binary dummy approach has proven to capture deterministic seasonal patterns very well. On the other hand, the stepwise approach is one of the standard methods for variable selection. Here, this stepwise method is used to obtain optimal neurons in the input layer, and this approach has not been previously applied by other researchers in GRNN modeling. At the same time, this method aligns with the principle of parsimony. The graphical illustration of real data plots, in-sample fitting, and out of sample forecasts using each GRNN model approach and several models available in the literature is shown in Fig. 1.

Indonesian inflation

In this study, the second case study was conducted on the inflation rate data in Indonesia. The observed data is monthly data from January 2005 to December 2022. The training data consists of the first 204 data points (from January 2005 to December 2021), and the last 12 data points are used as testing data (from January 2022 to December 2022). This data can be found and accessed on the official website of the Indonesian Central Bureau of Statistics (BPS).[26]

A similar method to the first case study was applied in executing the neurons in the input layer of the GRNN. After comprehensive modeling and analysis, the GRNN appeared as the most accurate, showing the lowest RMSE, MAE, MAPE, and SMAPE values (see Table 8). This exceptional performance was achieved through a parameter combination including frequency and BE, rolling origin, recursive methods, and additive transformation. The results also emphasized the effectiveness of the model in obtaining precise input layer neurons when using frequency and BE methods. This method adhered to the principle of parsimony, which prioritized simplicity. Additionally, the rolling origin method was identified as a robust optimization strategy for determining the smoothing parameter value that minimized forecasting errors. In comparison to the fixed origin method, the rolling origin method consistently yielded superior smoothing parameter values. In this second case study, the recursive method was used for forecasting

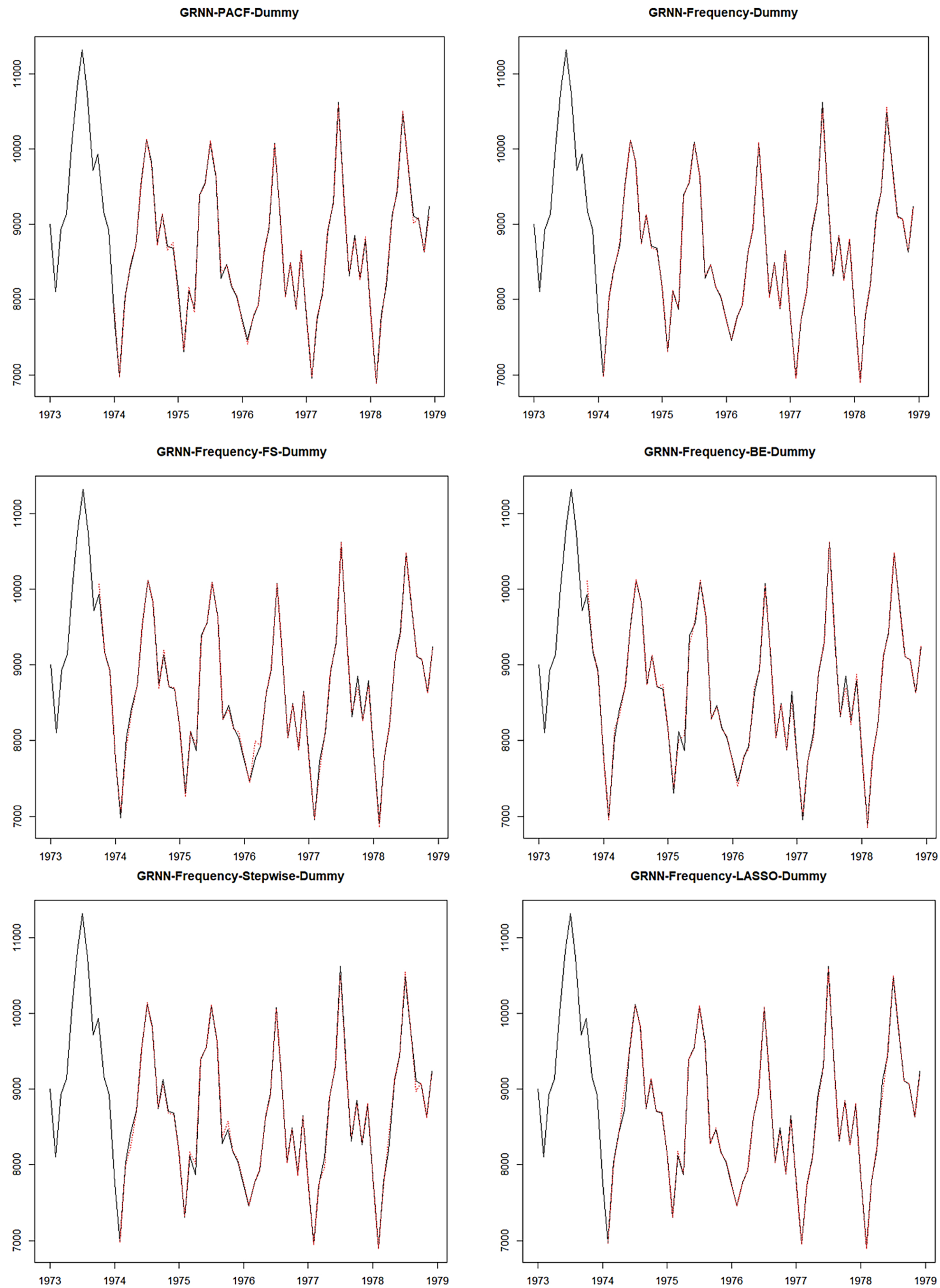

Fig. 1. Plots of real data, in-sample fitting, and out-sample forecasting using the proposed GRNN model approach and several models considered for the number of deaths due to accidents in the USA.

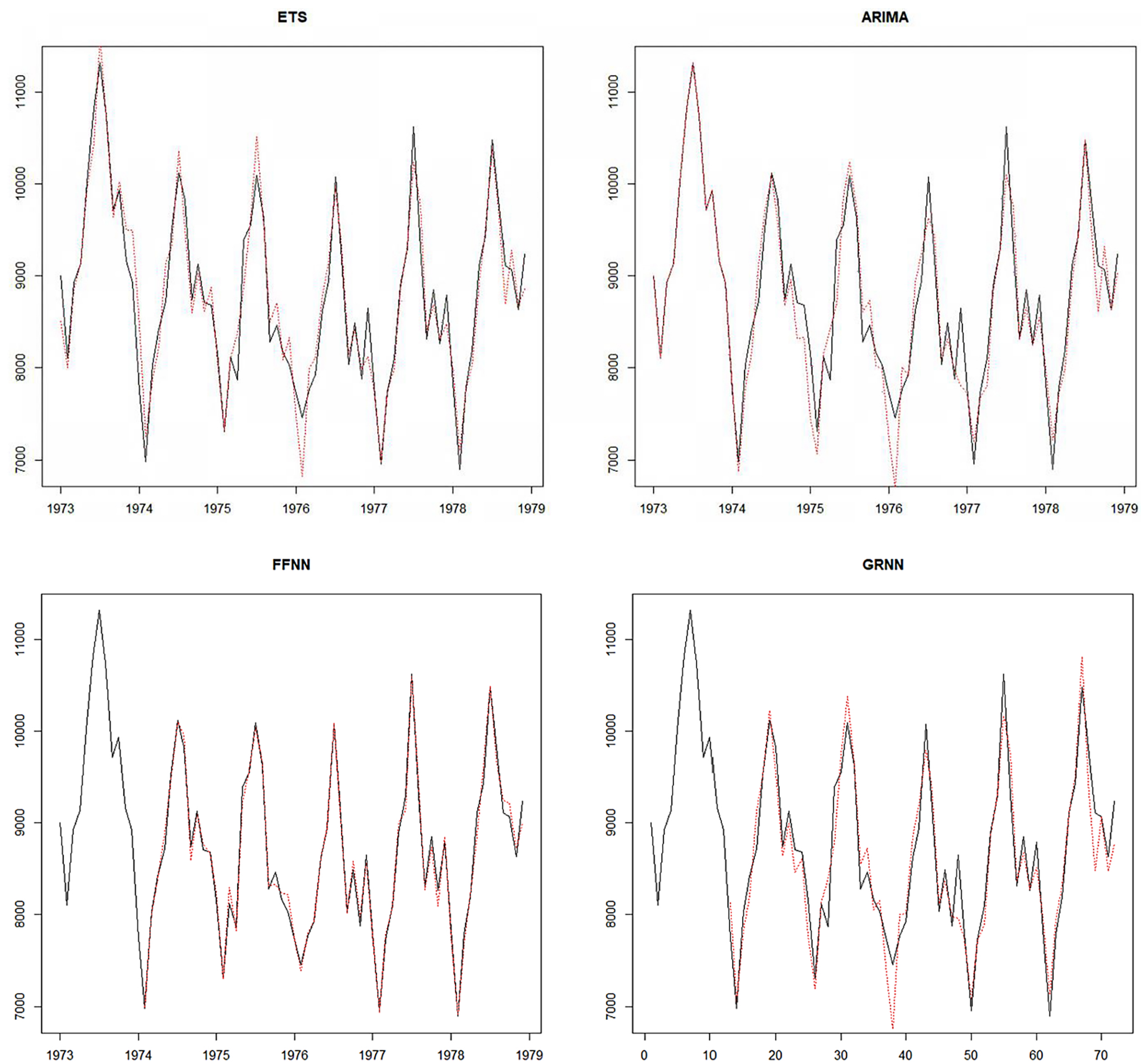

Fig. 1. Continued.

Table 8. Comparison of the outcomes of combining GRNN model parameters with neurons in the input layer via frequency and BE for Indonesian Inflation data.

	RMSE	MAE	MAPE	SMAPE
BE, rolling origin, recursive, no transformation	0.543854	0.541246	9.896515	10.41634
BE, fixed origin, recursive, no transformation	0.742335	0.611862	13.39000	13.80697
BE, rolling origin, MIMO, no transformation	0.686484	0.573046	10.43080	11.16774
BE, fixed origin, MIMO, no transformation	0.778400	0.688283	14.02450	14.64733
BE, rolling origin, recursive, additive transformation	0.013852	0.013852	0.251412	0.251096
BE, fixed origin, recursive, additive transformation	1.251738	1.041282	23.12957	26.89259
BE, rolling origin, MIMO, additive transformation	0.266359	0.213734	3.887281	4.008223
BE, fixed origin, MIMO, additive transformation	0.253993	0.253993	4.609681	4.718433
BE, rolling origin, recursive, multiplicative transformation	0.818129	0.818129	14.84808	16.03881
BE, fixed origin, recursive, multiplicative transformation	0.839931	0.687184	16.35711	16.53441
BE, rolling origin, MIMO, multiplicative transformation	0.916764	0.720889	16.50967	18.06761
BE, fixed origin, MIMO, multiplicative transformation	1.582075	1.384154	32.45919	39.86472

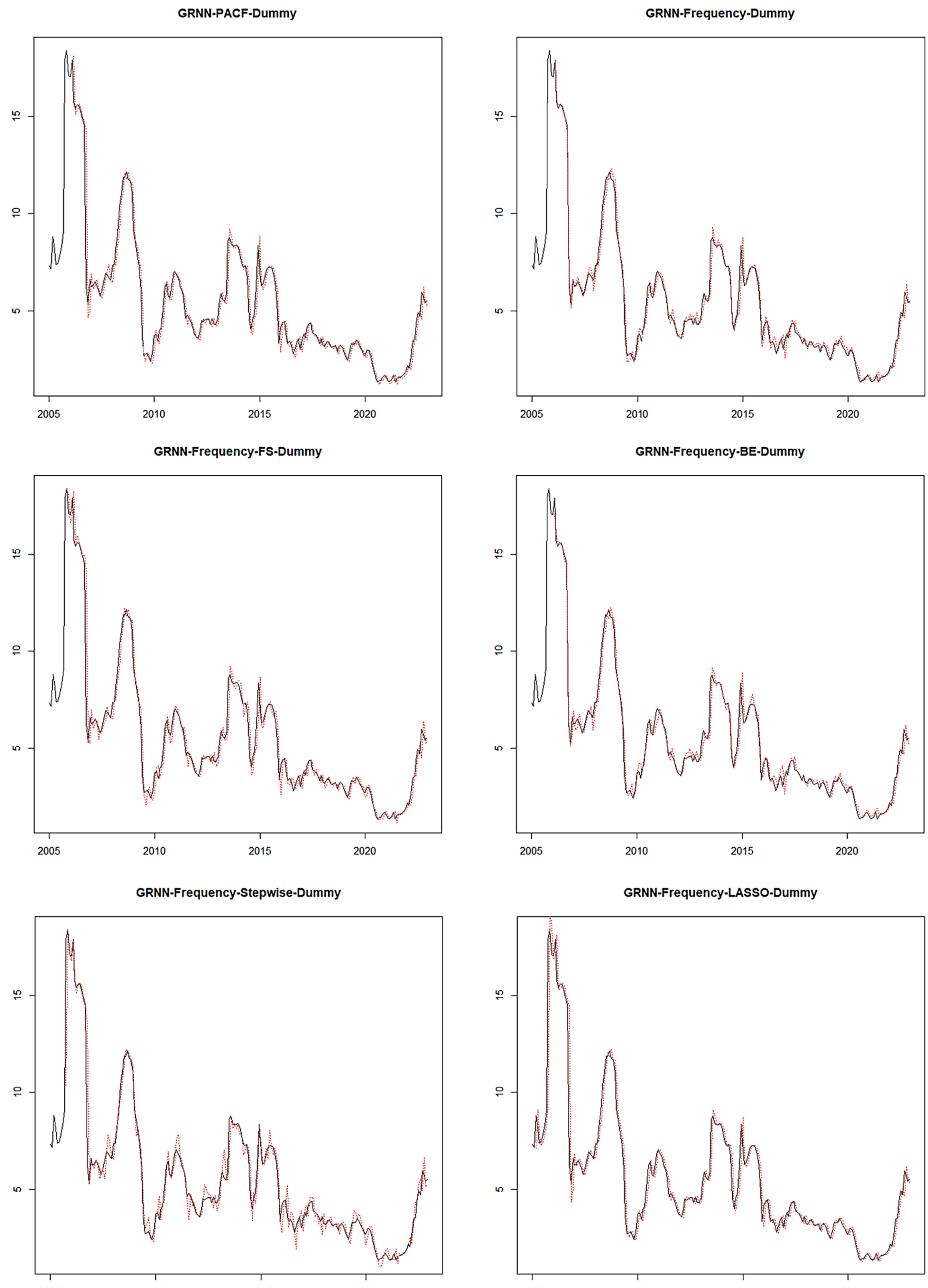

Fig. 2. Plots of real data, in-sample fitting, and out-sample forecasting using the proposed GRNN model approach and several models considered for inflation rate data in Indonesia.

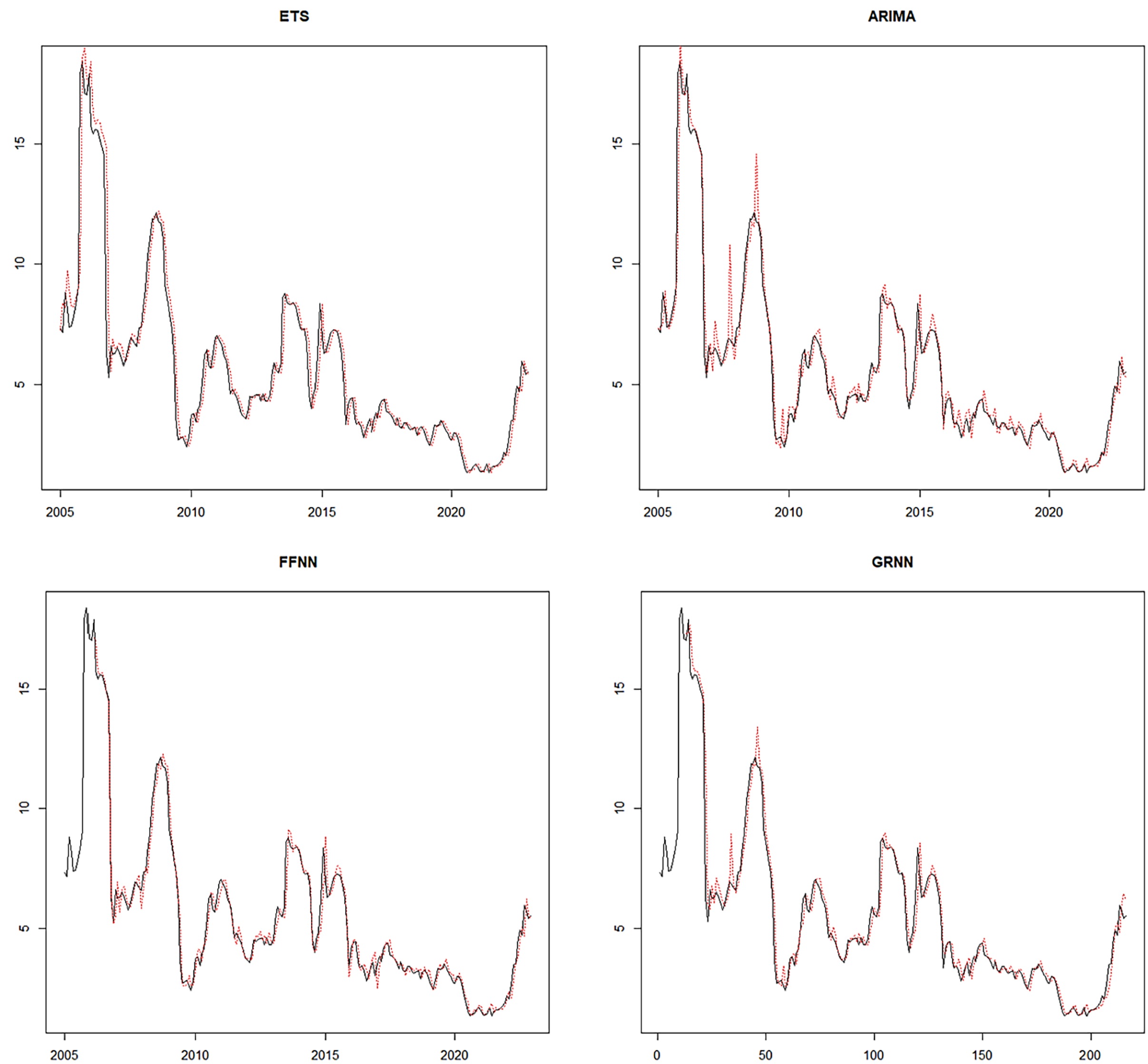

Fig. 2. Continued.

Table 9. Comparison of the results of the GRNN model and several other models for Indonesian Inflation data.

	RMSE	MAE	MAPE	SMAPE
GRNN-PACF-dummy	0.570993	0.357796	8.257855	8.213663
GRNN-Frequency-dummy	0.337627	0.235763	5.885520	5.808237
GRNN-Frequency-FS-dummy	0.405146	0.305084	7.168880	7.149291
GRNN-Frequency-BE-dummy	0.356746	0.268284	6.379361	6.346651
GRNN-Frequency-Stepwise-dummy	0.458697	0.307158	7.287032	7.242743
GRNN-Frequency-LASSO-dummy	0.930035	0.485018	9.363246	9.286715
ETS	1.033346	0.508369	9.407505	9.028196
ARIMA	0.773869	0.418744	7.997439	7.748047
FFNN	0.465528	0.322660	6.939367	6.911206
GRNN	0.654972	0.562850	13.70966	14.51901

multiple steps, resulting in significantly improved accuracy. The additive transformation strategy was applied to address trend and seasonal patterns. This method comprised preprocessing and modifying time series data, ultimately enhancing forecasting accuracy. It was observed that the additive transformation method outperformed the multiplicative transformation method in terms of accuracy.

In each modeling strategy, additional neurons were incorporated into the input layer in the form of binary dummies. Table 9 shows the results of an empirical investigation, comparing each GRNN with various models from the literature. The empirical results show that GRNN using the binary dummy method does not effectively capture deterministic seasonal trends. This is supported by the fact that the best model without a binary dummy yields lower RMSE, MAE, MAPE, and SMAPE values when compared to those with the binary dummy. Graphical illustrations of plots of real data, in-sample fitting, and out-sample forecasts using each of the proposed GRNN model approaches and several models available in the literature are shown in Fig. 2.

Conclusion

This study shows that the GRNN model with the frequency and Backward Elimination (BE) approach provides the best accuracy. The combination of frequency and BE with rolling origin, MIMO, and additive transformation produces a model with the lowest RMSE, MAE, MAPE, and SMAPE values. The frequency and BE approach help optimize the neurons in the input layer, following the principle of parsimony. Rolling origin maximizes the smoothing parameter, while MIMO improves long-term forecasting accuracy. The multiplicative transformation addresses trends and seasonal patterns. On the other hand, the addition of binary dummy neurons in the input layer combines the autoregressive lag approach and binary dummy to capture stochastic and deterministic seasonal patterns. The GRNN model using the frequency and stepwise approach along with a binary dummy shows the best results. The stepwise method is also effective in variable selection and determining the optimal number of neurons. The binary dummy approach is very effective in capturing deterministic seasonal patterns. These findings suggest that this combination of approaches consistently outperforms other methods in the literature for data with seasonal components, such as the data on the number of deaths due to accidents in the USA. However, this combination of approaches is not recommended for data without seasonal components, such as the inflation rate data in Indonesia, where only the autoregressive lag approach is advised. Future research can explore the combination of autoregressive lags and external variables to further improve the GRNN model's accuracy, developing further optimization methods, such as more sophisticated parameter tuning techniques, to enhance the performance of the GRNN model in the context of data with complex seasonal and trend patterns. Analyzing how variations in historical data and data frequency affect the GRNN model's performance, integrating the latest machine learning techniques, such as deep learning or ensemble methods, to see if they can improve model accuracy compared to classical approaches currently used, and conducting case studies with data from other domains to test the generalizability of these findings.

Acknowledgment

The authors are grateful to all relevant agencies, particularly the Directorate of Research and Community Service Universitas Islam Indonesia for financial support, as well as colleagues who assisted with data collection and analysis.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images, that are not ours, have been included with the necessary permission for re-publication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at Universitas Islam Indonesia, Indonesia.

Authors' contribution statement

MM contributed to the conception, design, analysis, revision, and proofreading of the manuscript. H designed the study, analyzed the data, and acquired the data. APW participated in the interpretation of the results, and LAP was involved in drafting the manuscript. All authors read and approved the final manuscript.

References

1. Samarasinghe S. Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition. New York: Auerbach Publications. 2016;570 p. https://doi.org/10.1201/9780849333750.
2. Mehlig B. Machine Learning with Neural Networks: An Introduction for Scientists and Engineers. UK: Cambridge University Press; 2021;250 p. https://doi.org/10.1017/9781108860604.
3. Gupta R, Ekata, Batra CM. Performance Assessment of Solar-Transformer-Consumption System Using Neural Network

Approach. Baghdad Sci J. 2022;19(4):865–874. https://doi.org/10.21123/bsj.2022.19.4.0865
4. Kumaravel G, Kirthiga S, Al Shekaili MMH, Al Othmani QHSA. A Solar Photovoltaic Performance Monitoring and Statistical Forecasting Model Using a Multi-Layer Feed-Forward Neural Network and Artificial Intelligence. Baghdad Sci J. 2024;21(5(SI)):1868–1877. https://doi.org/10.21123/bsj.2024.10736.
5. Cabaneros SM, Calautit JK, Hughes BR. A Review of Artificial Neural Network Models for Ambient Air Pollution Prediction. Environ Model Softw. 2019;119:285–304. https://doi.org/10.1016/j.envsoft.2019.06.014.
6. Bui DT, Tsangaratos P, Nguyen V-T, Liem NV, Trinh PT. Comparing the Prediction Performance of a Deep Learning Neural Network Model with Conventional Machine Learning Models in Landslide Susceptibility Assessment. Catena. 2020;188:104426. https://doi.org/10.1016/j.catena.2019.104426.
7. Song Y, Yue L, Wang Y, Di H, Gao F, Li S, *et al.* Research on BP Network for Retrieving Extinction Coefficient from Mie Scattering Signal of Lidar. Measurement. 2020;164:108028. https://doi.org/10.1016/j.measurement.2020.108028.
8. Farrell MH, Liang T, Misra S. Deep Neural Networks for Estimation and Inference. Econometrica. 2021;89(1):181–213. https://doi.org/10.3982/ECTA16901.
9. Sun Z, Li X, Zhang H, Ikbal MA, Farooqi AR. A GA-BP Neural Network for Nonlinear Time-Series Forecasting and its Application in Cigarette Sales Forecast. Nonlinear Eng. 2022;11(1):223–231. https://doi.org/10.1515/nleng-2022-0025.
10. Khan ABF, Kamalakannan K, Ahmed NSS. Integrating Machine Learning and Stochastic Pattern Analysis for the Forecasting of Time-Series Data. SN Comput Sci. 2023;4(5):484. https://doi.org/10.1007/s42979-023-01981-0.
11. Ashour MAH. Optimized Artificial Neural Network Models to Time Series. Baghdad Sci J. 2022;19(4):899–904. https://doi.org/10.21123/bsj.2022.19.4.0899
12. Martínez F, Charte F, Frías MP, Martínez-Rodríguez AM. Strategies for Time Series Forecasting with Generalized Regression Neural Networks. Neurocomputing. 2022;491:509–521. https://doi.org/10.1016/j.neucom.2021.12.028.
13. Hermansah, Rosadi D, Abdurakhman, Utami H. Time Series Forecasting with Trend and Seasonal Patterns Using NARX Network Ensembles. Math Stat. 2021;9(4):511–520. https://doi.org/10.13189/ms.2021.090410.
14. Leung MT, Chen A-S, Daouk H. Forecasting Exchange Rates Using General Regression Neural Networks. Comput. Oper Res. 2000;27(11–12):1093–1110. https://doi.org/10.1016/S0305-0548(99)00144-6.
15. Bukhari AH, Raja MAZ, Sulaiman M, Islam S, Shoaib M, Kumam P. Fractional Neuro-Sequential ARFIMA-LSTM for Financial Market Forecasting. IEEE Access. 2020;8:71326–71338. https://doi.org/10.1109/ACCESS.2020.2985763.
16. Song C, Wang L, Hou J, Xu Z, Huang Y. The Optimized GRNN Based on the FDS-FOA Under the Hesitant Fuzzy Environment and its Application in Air Quality Index Prediction. Appl Intell. 2021;51(11):8365–8376. https://doi.org/10.1007/s10489-021-02350-1.
17. Wang G, Wu T, Wei W, Jiang J, An S, Liang B, *et al.* Comparison of ARIMA, ES, GRNN and ARIMA-GRNN Hybrid Models to Forecast the Second Wave of COVID-19 in India and the United States. Epidemiol Infect. 2021;149:1–9. https://doi.org/10.1017/S0950268821002375.
18. Martínez F, Frías MP, Pérez-Godoy MD, Rivera AJ. Dealing with seasonality by narrowing the training set in time series forecasting with kNN. Expert Syst Appl. 2018;103:38–48. https://doi.org/10.1016/j.eswa.2018.03.005.
19. Martínez F, Charte F, Rivera AJ, Frías MP. Automatic Time Series Forecasting with GRNN: A Comparison with Other Models. In: Rojas I, Joya G, Catala A, editors. Advances in Computational Intelligence (IWANN 2019). Lecture Notes in Computer Science vol. 11506. Cham: Springer; 2019. 198–209. https://doi.org/10.1007/978-3-030-20521-8_17.
20. Makridakis S, Spiliotis E, Assimakopoulos V. The M4 Competition: 100,000 Time Series and 61 Forecasting Methods. Int J Forecast. 2020;36(1):54–74. https://doi.org/10.1016/j.ijforecast.2019.04.014.
21. Hyndman RJ, Koehler AB, Snyder R, Grose S. A State Space Framework for Automatic Forecasting Using Exponential Smoothing Methods. Int J Forecast. 2002;18(3):439–454. https://doi.org/10.1016/S0169-2070(01)00110-8.
22. Hyndman RJ, Khandakar Y. Automatic Time Series Forecasting: The forecast Package for R. J Stat Softw. 2008;27(3):1–22. https://doi.org/10.18637/jss.v027.i03.
23. Hermansah, Rosadi D, Abdurakhman A, Utami H. Selection of Input Variables of Nonlinear Autoregressive Neural Network Model for Time Series Data Forecasting. Media Stat. 2020;13(2):116–124. https://doi.org/10.14710/medstat.13.2.116-124.
24. Specht DF. A General Regression Neural Network. IEEE Trans Neural Netw. 1991;2(6):568–576. https://doi.org/10.1109/72.97934.
25. National Highway Traffic Safety Administration. Fatality Analysis Reporting System (FARS). 2024.
26. Badan Pusat Statistik (BPS). Indeks Harga Konsumen (IHK) Dan Inflasi. 2024.

توقع بيانات السلاسل الزمنية باستخدام طريقة جديدة على نموذج شبكة الأعصاب للتراجع المعمم

محمد مهاجر[1]، هيرمانشاه[2]، أديتيا بندو ويكاكسونو[3]، لطيفة عليا براتيوي[1]

[1] *قسم الإحصاء، كلية الرياضيات والعلوم الطبيعية، الجامعة الإسلامية الإندونيسية، يوغياكارتا، إندونيسيا.*

[2] *قسم تعليم الرياضيات، كلية إعداد المعلمين، جامعة رياو كيبولاوان، كيبولاوان رياو، إندونيسيا.*

[3] *قسم المحاسبة، كلية الأعمال والاقتصاد، الجامعة الإسلامية الإندونيسية، يوغياكارتا، إندونيسيا.*

المستخلص

هدفت هذه الدراسة إلى تحديد طبقة الإدخال للخلايا العصبية لنموذج الشبكة العصبية التراجعية العامة (GRNN) باستخدام طرق تقليدية مختلفة، وهي: 1) دالة الارتباط الذاتي الجزئي 2 ،(PACF)) طريقة قائمة على التردد، 3) التردد والاختيار الأمامي 4 ،(FS)) التردد والإزالة الخلفية 5 ،(BE)) التردد والطرق القائمة على الخطوات، و6) طريقة التردد المدمجة مع مشغل الانكماش والاختيار المطلق الأقل (LASSO) . تم دمج كل من هذه الطرق التقليدية مع معلمات مختلفة داخل GRNN، بما في ذلك معلمات التنعيم واستراتيجيات التنبؤ والتحويلات. تم الحصول على النموذج الأكثر دقة، بأقل قيم MAPE، MAE، RMSE، وSMAPE، من خلال الجمع بين معلمات تتضمن التردد وBE، الأصل المتداول، MIMO، والتحويل الإضافي. بالإضافة إلى ذلك، تم تقديم طريقة تتكون من خلايا عصبية ثنائية اصطناعية في طبقة الإدخال. في هذه الطريقة التقليدية، كان النموذج الأمثل يتكون من خلايا عصبية إضافية في طبقة الإدخال على شكل محاكاة ثنائية. وبالتالي، جمعت هذه الطريقة بين التأخر الذاتي المتراجع والمتغيرات الثنائية الوهمية لالتقاط الأنماط الموسمية العشوائية والحتمية على التوالي. أظهرت الدراسة التجريبية أن استخدام التردد والأساليب التسلسلية، مع المحاكيات الثنائية، أنتجت النتائج الأكثر دقة، كما يتضح من أقل قيم MAPE، MAE، RMSE، وSMAPE. كما أظهرت النتائج فروقًا كبيرة في دقة التنبؤ بين نموذج GRNN المقترح والنماذج الموجودة مثل التنعيم الأسي، FFNN، ARIMA، وGRNN. بناءً على هذه النتائج، أثبتت المنهجية المقدمة في هذه الدراسة أنها وسيلة فعالة لتعزيز دقة التنبؤ.

الكلمات المفتاحية: الاستبعاد العكسي، الدمية الثنائية، الاختيار الأمامي، شبكة العصبونات العامة للانحدار، أقل انكماش مطلق ومشغل للاختيار.