A Hybrid GSTARX-Jordan RNN Model for Forecasting Space-Time Data with Calendar Variation Effect

F Hikmawati, Suhartono, D D Prastyo
Department of Statistics, Institut Teknologi Sepuluh Nopember
Kampus ITS Sukolilo, Surabaya 60111, Indonesia
*e-mail: fauzah.hikmawati@gmail.com

Abstract. Generalized Space-Time Autoregressive (GSTAR) is one of the space-time models. The GSTAR model has its limitations of not being able to model a nonlinear time series, and this can be overcome by applying a hybrid model on GSTAR. This research aims to propose modeling hybrid Time Series Regression (TSR) and hybrid GSTARX-Jordan RNN, where TSR and GSTARX model as a linear component involving the predictor variable, which is an effect of calendar variation and Jordan-RNN as a nonlinear component. This research focused on a simulation study to evaluate the goodness of the model hybrid GSTARX-Jordan RNN. There were some scenarios experimented, i.e. simulation studies in data that have linear noise and non-linear noise. The results showed that a hybrid GSTARX-FFNN, GSTARX-DLNN, and GSTARX-Jordan RNN model is the best model for predicting simulation data containing trend, seasonality, calendar variations, and nonlinear noise series compared with TSR, and GSTARX models. In general, it is in line with the results of the 2018 M4 forecasting competition show that combined models or hybrid models tend to provide more accurate forecast performance than forecast results with individual models.

Keywords: Calendar Variation, GSTAR, Hybrid GSTARX-Jordan RNN, Space-Time

1. Introduction
Space-time Autoregressive (STAR) model was introduced by Pfeifer and Deutsch in 1980 [1]. STAR is a model that combines dependency between time and location at multivariate time series data. Suhartono [2] did a study about the comparison of the stages of model formation and forecast accurately between STAR and VARIMA model at a time series and location data. Ruchjana [3], Borovkova, Lopuhaa and Ruchjana [4] developed the Generalized Space-Time Auto-Regressive (GSTAR) model as elaboration from STAR model. GSTAR can be applied at data with heterogeneous location variables, with giving weight for each location to produce a space-time model with parameters that can be different for time and location dependencies. In this study, GSTAR is used in modeled data in that containing trend, seasonal, and calendar components variation.

Time series data such as economic data have seasonal patterns that are influenced by external events such as holidays, sales promotions and other policy changes [5]. These external events cause the presence of the calendar variation pattern in time series data. Thus, must add predictor variables on the GSTAR model (then known as GSTARX model) i.e., calendar variation effect. It is also possible that data is formed from linear and nonlinear structures [6]. The GSTAR model has the limitation of being unable to modeling a nonlinear time series. Therefore, the GSTAR model with calendar variation effect (GSTARX) will be combined with Artificial Neural Network (ANN), that called a hybrid GSTARX-ANN model to resolve linear and nonlinear data structures simultaneously.
The advantage of ANN is unnecessary to define a specific model form because the model is formed adaptive based on features presented from the data [7]. Minimize the forecast error depending on the depth of the ANN model measured from the number of hidden layers, increasingly in ANN's architecture will provide better forecast results. ANN with a hidden layer more than one known as one of the Deep Learning Neural Network (DLNN) model form. There are some models in ANN, Jordan Recurrent Neural Network (Jordan-RNN) is one of the ANN models. Jordan-RNN is a model that has similarities with a multilayer perceptron with an additional single context layer [8]. The memory neurons in the context layer of Jordan-RNN derived from the output layer. RNN modeling continues to be developed on several occasions, and either is Wysocki & Ławrynczuk [9], who did modeling Jordan-RNN as predictive control of the dynamic system. Although the Jordan RNN model is a model that has a very simple recurrent architecture and when used for modeling higher dynamic processes provides some unavoidable inaccuracies, this model was used successfully on the predictive control [9].

This study used three ANN architectures to know the goodness of the forecast model. First is hybrid GSTARX and Feed Forward Neural Network model with one hidden layer (GSTAR-FFNN), second is hybrid GSTARX and Deep Learning Neural Network model (GSTAR-DLNN) with two hidden layers, and the neural network structure is developed in this research is repeated type model i.e. hybrid GSTARX and Jordan RNN model (GSTAR-Jordan RNN) with one hidden layer and single context unit. The purpose of this research is to know performance of the hybrid GSTARX-JRNN model in capturing spatio-temporal data patterns that containing components trend, seasonal, and calendar variation compared with TSR and GSTAR model use criteria Root Mean Square Error (RMSE).

2. Research method

2.1. Time Series Regression

Time Series Regression (TSR) is used to form the calendar variation model. The calendar variation model is a time series model used to forecast data based on seasonal patterns with varied periods [10]. In general, the calendar variation model is based on the regression method if there are trends, seasonality, and calendar variation effect on the data, then the model can be written as

\[ Z_{i,j} = \delta t + \sum_{m=1}^{12} \gamma_m S_{m,j} + \sum_{g} \nu_g L_{g,i} + \sum_{g} \nu_g L_{g,i+1} + N_{i,j}. \]  

by \( \delta \) is a linear trend parameter, \( S_{1,j}, S_{2,j}, ..., S_{12,j} \) are monthly seasonal dummy variable, and \( \nu \), and \( \nu \) are calendar variation parameter.

2.2. Model Generalized Space-Time Autoregressive (GSTAR)

GSTAR is a time series analysis that considers characteristics of location factors. GSTAR models are the specifications of the Vector Autoregressive (VAR) model, which assumes the research locations are heterogeneous [3]. These characteristics are demonstrated by providing location-weighted or spatial (W). If given a series of \( \{Z(t) : t = 0, \pm 1, \pm 2, ..., T\} \) that it is a multivariate time series from \( N \) location, then the order autoregressive of GSTAR model is \( p \) and the spatial order \( \lambda_1, \lambda_2, ..., \lambda_p \) or defined by GSTAR \( (p; \lambda_1, \lambda_2, ..., \lambda_p) \) can be written as following [11]

\[ Z(t) = \sum_{k=1}^{p} \Phi_{k0} + \sum_{l=1}^{L} \Phi_{l} W^{(l)} Z(t-k) + a(t). \]  

In this study used uniform weights assume that the locations used are homogeneous. Generalized Least Square (GLS) employed the parameter estimation of the GSTAR model. GSTAR model involving exogenous variables is known GSTARX model. GSTARX is a development of multivariate time series models for data space-time involving an exogenous variable.
2.3. Recurrent Neural Network (RNN)

RNN is one of the types of neural network models. Jordan-RNN is a type of RNN model in which the neuron on the context layer obtained information from the output layer [8]. Context layer contains delay neurons, and delay neuron is a neuron that saves the memory is located on the activation values in the previous step, then neurons will release the previously saved amounts to the network in the next level. The equation of Jordan-RNN model following by

\[
\hat{N}_t = f^\top \left[ \sum_{j=1}^{p} w_{j}^{\top} h_j + \sum_{j=1}^{q} w_{j}^{\top} J_{t}^{(j)} + u_{t}^{\top} c_{t} \right].
\]

2.4. Hybrid GSTARX-Jordan RNN Model

The first hybrid model was introduced by Zhang [6] the purpose is to improve the accuracy of the forecast. Hybrid models combine linear components and nonlinear components. In this study, hybrid modeling was conducted in two phases. The first step is a modeling of the patterns trend, seasonal, and calendar variation using TSR. The next step is modeling of the residual from the previous step using GSTARX-Jordan RNN. Flowchart of the hybrid GSTARX-Jordan RNN is illustrated in figure 1.

![Flowchart of the hybrid GSTARX-Jordan RNN](image)

**Figure 1.** Hybrid GSTARX-Jordan RNN Modeling Flowchart

with, \( \mathbf{N}^{(1)}_{t} \) is a lag-1 residual model of TSR and \( \mathbf{W}^{(2)}_{t-1} \) is a lag-1 weighted of residual TSR model.

\[ \hat{Z}_{t|y} = \hat{Z}_{t|y}^{(1)} + \hat{N}_{t|y}^{(2)} \]

2.5. Best Model Selection

The selection of the best models in this study used the criteria Root Mean Square Error (RMSE) for in-sample data and out-sample data. Hyndman and Koehler [12] explained that to apply and compare different methods on the same scale of data, RMSE can be used because it will be the same with the scale of the data [2]. The formula of RMSE as follows:

\[
RMSE = \sqrt{\frac{1}{L} \sum_{t=1}^{L} (\hat{Z}_{t|y} - \hat{\mathbf{Z}}_{t|y})^2}
\]

3. Data

3.1. Simulation Study

Simulated studies conducted to know the performance of hybrid GSTARX-Jordan RNN model to provide a model of time series data that contains component trend, seasonal, calendar variation, and a linear or nonlinear noise sequence. The stages of conducting simulated studies follow:
3.1.1. Generating data that has a component effect trend, seasonal, and calendar variation with the following equation,

\[ Z_i(t) = T_i(t) + S_i(t) + V_i(t) + N_i(t), \quad i = 1, 2, 3, 4 \] stating the location with,

i. The trend components used are linear components with the following equations,

\[ T_i(t) = \delta_i(t), \quad \text{with} \quad \delta_1(t) = 0.3, \quad \delta_2(t) = 0.17, \quad \delta_3(t) = 0.22, \quad \delta_4(t) = 0.15 \] (5)

ii. Component seasonal obtained with the following equation,

\[ S_i(t) = \gamma_1(t)S_{1,t} + \gamma_2(t)S_{2,t} + \ldots + \gamma_3(t)S_{3,t} \] (6)

where,

\[ S_1(t) = 16.5S_{1,t} + 19.79S_{2,t} + 21S_{3,t} + 19.79S_{4,t} + 16.5S_{5,t} + 12S_{6,t} + 7.5S_{7,t} + 4.21S_{8,t} + 3S_{9,t} + 4.21S_{10,t} + 7.5S_{11,t} + 12S_{12,t} \]

\[ S_2(t) = 14S_{1,t} + 15.46S_{2,t} + 16S_{3,t} + 15.46S_{4,t} + 14S_{5,t} + 12S_{6,t} + 10S_{7,t} + 8.54S_{8,t} + 8S_{9,t} + 8.54S_{10,t} + 10S_{11,t} + 12S_{12,t} \]

\[ S_3(t) = 15.5S_{1,t} + 18.15S_{2,t} + 19S_{3,t} + 18.15S_{4,t} + 15.5S_{5,t} + 12S_{6,t} + 8.5S_{7,t} + 5.9S_{8,t} + 5S_{9,t} + 5.9S_{10,t} + 8.5S_{11,t} + 12S_{12,t} \]

\[ S_4(t) = 14.55S_{1,t} + 16.35S_{2,t} + 17S_{3,t} + 16.35S_{4,t} + 14S_{5,t} + 12S_{6,t} + 9.5S_{7,t} + 7.5S_{8,t} + 7S_{9,t} + 7.5S_{10,t} + 9.5S_{11,t} + 12S_{12,t} \]

iii. Components for the calendar variation in this simulated study are only at the time of  \( t \) and  \( t-1 \), with the following equation,

\[ V_i(t) = \alpha_1V_{i,t-1} + \ldots + \alpha_4V_{i,t-4} + \beta_1V_{i,t-1} + \ldots + \beta_4V_{i,t-4} \] (7)

where,

\[ V_1(t) = 8.13V_{1,t} + 47.46V_{2,t} + 66.5V_{3,t} + 82.38V_{4,t} + 62.29V_{5,t-1} + 48.21V_{6,t-2} + 33.07V_{7,t-3} + 32.42V_{8,t-4} \]

\[ V_2(t) = 3.9V_{1,t} + 11.14V_{2,t} + 22.84V_{3,t} + 29.59V_{4,t} + 19.77V_{5,t-1} + 13.39V_{6,t-2} + 11.6V_{7,t-3} + 9.05V_{8,t-4} \]

\[ V_3(t) = 3.74V_{1,t} + 15.99V_{2,t} + 33.78V_{3,t} + 36.4V_{4,t} + 28.7V_{5,t-1} + 20.46V_{6,t-2} + 18.39V_{7,t-3} + 16.12V_{8,t-4} \]

\[ V_4(t) = 1.73V_{1,t} + 7.87V_{2,t} + 14.2V_{3,t} + 18.48V_{4,t} + 14.21V_{5,t-1} + 10.67V_{6,t-2} + 10.43V_{7,t-3} + 8.85V_{8,t-4} \]

iv. The components for the noise sequence used in this simulation study consist of noise with linear and nonlinear patterns with each of the following equations, Linear Noise Pattern

\[ N_i(t) = \phi_1(t)N_{1,t-1} + \phi_2(t)N_{2,t-1} + \phi_3(t)N_{3,t-1} + \phi_4(t)N_{4,t-1} + \alpha_i \] (8)

where,

\[ N_1(t) = 0.4N_{1,t-1} + 0.18N_{2,t-1} + 0.18N_{3,t-1} + 0.18N_{4,t-1} + \alpha_1(t), \]

\[ N_2(t) = 0.19N_{1,t-1} + 0.37N_{2,t-1} + 0.19N_{3,t-1} + 0.19N_{4,t-1} + \alpha_2(t), \]

\[ N_3(t) = 0.2N_{1,t-1} + 0.2N_{2,t-1} + 0.35N_{3,t-1} + 0.2N_{4,t-1} + \alpha_3(t), \] and

\[ N_4(t) = 0.21N_{1,t-1} + 0.21N_{2,t-1} + 0.21N_{3,t-1} + 0.33N_{4,t-1} + \alpha_4(t) \]

Nonlinear Noise Pattern

\[ N_1(t) = 3.5N_{1,t-1} \exp(-0.2N^2_{1,t-1})N_{1,t-1} + 0.9N_{2,t-1} \exp(-0.2N^2_{2,t-1})N_{2,t-1} + 1.5N_{3,t-1} \exp(-0.2N^2_{3,t-1})N_{3,t-1} + 3N_{4,t-1} \exp(-0.2N^2_{4,t-1})N_{4,t-1} + \alpha_1(t) \]

\[ N_2(t) = 1.8N_{1,t-1} \exp(-0.2N^2_{1,t-1})N_{1,t-1} + 4N_{2,t-1} \exp(-0.2N^2_{2,t-1})N_{2,t-1} + 1.1N_{3,t-1} \exp(-0.2N^2_{3,t-1})N_{3,t-1} + 1.5N_{4,t-1} \exp(-0.2N^2_{4,t-1})N_{4,t-1} + \alpha_2(t) \]

\[ N_3(t) = 1.2N_{1,t-1} \exp(-0.2N^2_{1,t-1})N_{1,t-1} + 2.3N_{2,t-1} \exp(-0.2N^2_{2,t-1})N_{2,t-1} + 4.5N_{3,t-1} \exp(-0.2N^2_{3,t-1})N_{3,t-1} + 1.3N_{4,t-1} \exp(-0.2N^2_{4,t-1})N_{4,t-1} + \alpha_3(t) \]

\[ N_4(t) = 1.9N_{1,t-1} \exp(-0.2N^2_{1,t-1})N_{1,t-1} + 2.1N_{2,t-1} \exp(-0.2N^2_{2,t-1})N_{2,t-1} + 1.3N_{3,t-1} \exp(-0.2N^2_{3,t-1})N_{3,t-1} + 3.9N_{4,t-1} \exp(-0.2N^2_{4,t-1})N_{4,t-1} + \alpha_4(t) \]

where \( \alpha_i = MN(0, \Sigma) \) and between location correlates.
3.1.2. Get Time-series data $Z_{i,t}$, with $i = 1, 2, 3, 4$ which means there are 4 locations and $t = 1, 2, \ldots, 144$, With the following scenarios

i. Scenario I, data that has a linear noise pattern

ii. Scenario II, data that has a nonlinear noise pattern

There is the time series plot from generating data that has the trend, seasonal, and calendar variation components at both scenarios, which is shown in figure 1 and figure 2 as follows.

\[
\Sigma = \begin{bmatrix}
1 & 0.6 & 0.45 & 0.55 \\
0.6 & 1 & 0.57 & 0.61 \\
0.45 & 0.57 & 1 & 0.49 \\
0.55 & 0.61 & 0.47 & 1
\end{bmatrix}
\]

Figure 1. Time Series Plot Data Simulation Scenario I

Figure 2. Time Series Plot Data Simulation Scenario II
4. Result

4.1. Time Series Regression (TSR) Model
This section shows the RMSE value of TSR model results on four locations with ten replications. Figure 3 shows that the RMSE value of the out-sample data from the TSR model is still quite high, especially in scenario II. It because data in scenario II consist of nonlinear noise patterns and the TSR model can only capture data patterns with linear noise.

![Figure 3. RMSE value of the TSR Model on Data Scenario I and Scenario II](image)

4.2. GSTAR Model
In this stage, GSTAR model using uniform weights and is only limited order GSTAR(1,1). Table 1 provides the summary of RMSE from GSTAR model.

| Location | RMSE Scenario I In Sample | RMSE Scenario I Out Sample | RMSE Scenario II In Sample | RMSE Scenario II Out Sample |
|----------|---------------------------|----------------------------|----------------------------|----------------------------|
| Location 1 | 14.152                    | 45.624                     | 15.465                     | 50.356                     |
| Location 2 | 5.221                     | 13.960                     | 7.174                      | 17.561                     |
| Location 3 | 6.836                     | 19.339                     | 8.580                      | 22.908                     |
| Location 4 | 3.540                     | 10.049                     | 5.886                      | 14.051                     |

Table 1 showed that the RMSE value from the GSTAR model is very high at all locations because this model has not been able to capture calendar variation effect and can modeling data with higher GSTAR orders is possible. RMSE out-sample is higher than in-sample in both scenarios.

4.3. GSTARX Model
The order of the models used is GSTAR (1,1) for data scenario I and GSTAR (2,1) for data scenario II. The weights used in GSTARX model are uniform weights assuming that each location is homogeneous. Figure 4 presents the results of RMSE value from GSTARX model. The RMSE value is still quite high especially in the simulation data scenario II. The reason is the GSTARX model has not been able to capture data with nonlinear noise.
Figure 4. Boxplot RMSE value Model GSTARX on Data Scenario I and Scenario II

4.4. Jordan-RNN Model
The Jordan-RNN model was employed the input lag $t-1$. The Jordan networks from 1 to 5, 10, and 15 hidden neurons are trained. In this model, the activation function used in the hidden layer is tanh, while in the layer output is a linear activation function. Therefore, these results have high forecasting error on both scenarios, because the input lag used is limited and not adjusted to the data pattern. One of the best model forms gained on both scenarios of each ten replications as follow:

The best model of scenario I first replication with four input and one neuron in the hidden layer

$$\hat{Y}_i = f^{\alpha} \left[ 0.99 f^{\alpha} \left[ (14.63 + 3.38) Y_{t-1}^{(1)} + (1.82 + 3.38) Y_{t-1}^{(2)} + (3.45 + 3.38) Y_{t-1}^{(3)} + (2.18 + 3.38) Y_{t-1}^{(4)} \right] - 3.38 \right]$$

The best model of scenario II first replication is four input and two neurons in the hidden layer

$$f_1^{\alpha} = (0.29 + 0.299) Y_{t-1}^{(1)} + (0.012 + 0.299) Y_{t-1}^{(2)} + (0.089 + 0.299) Y_{t-1}^{(3)} + (-0.134 + 0.299) Y_{t-1}^{(4)} - 0.299$$

$$f_2^{\alpha} = (8.13 - 1.34) Y_{t-1}^{(1)} + (5.309 - 1.34) Y_{t-1}^{(2)} + (8.29 - 1.34) Y_{t-1}^{(3)} + (4.5 - 1.34) Y_{t-1}^{(4)} + 1.34$$

$Y_{t-1}^{(i)}$ is standardized of $Y_{t-1}$ and the $i$th are number of locations

where, $\hat{Y}_i = \begin{bmatrix} \hat{Y}_i^{(1)} \\ \hat{Y}_i^{(2)} \\ \hat{Y}_i^{(3)} \\ \hat{Y}_i^{(4)} \end{bmatrix}$ and

$Y_{t-1}^{(i)} = \begin{bmatrix} Y_{t-1}^{(1)} \\ Y_{t-1}^{(2)} \\ Y_{t-1}^{(3)} \\ Y_{t-1}^{(4)} \end{bmatrix}$

4.5. Hybrid GSTARX-Jordan RNN Model
At this step, the inputs of hybrid GSTARX-Jordan RNN are residual from TSR model with uniform weight and lag $t-1$. The number of neurons used in the hidden layer as well as the activation function of the hidden layer and the output layer as in the previous Jordan-RNN model. The figure below illustrates the best model architecture on the first replication in both scenarios.
Figure 5. Architecture Hybrid Model GSTARX-Jordan RNN in Data Scenario I and II

Figure 5 shows the best model architecture from hybrid GSTARX-Jordan RNN at the scenario I first replication with one neuron in the hidden layer, but at scenario II first replication with 15 neurons in the hidden layer. Results from ten replications at both scenarios have a different number of neurons in each best models at each replication.

4.6. Best Model Selection
In this section, it will compare to the entire ratio RMSE value. Figure 6 shows the ratio result of an RMSE value on each model used. From the graph below we can see that the hybrid GSTARX-DLNN model has the least RMSE value and has similarities RMSE with other models except for the TSR model because it still had quite high RMSE at the scenario I that have linear noise. In scenario II, it revealed that modeling using hybrid GSTARX-FFNN, GSTARX-DLNN, and GSTARX-Jordan RNN has RMSE smaller than the TSR and GSTARX models and has similarities RMSE on three hybrid models. It suggests that hybrid models have higher accuracy to forecast data with nonlinear noise patterns. This results obtained in line with M4-Competition [13].

Figure 6. Ratio RMSE value of each Model in the scenario I and scenario II

5. Conclusion
The results of this simulation study shown that RMSE from the hybrid model and GSTARX model have similarities at data with linear noise pattern. The hybrid and GSTARX models yield tend to similar forecasting accuracy when handling data consist of linear noise. However, those models give more accurate than the TSR model. Otherwise, the hybrid models more accurate than the TSR and GSTARX models to forecast data with nonlinear noise pattern. These results in line with M4-Competition results. Thus, hybrid models tend to be more accurate than individual forecasting models [13].
Acknowledgement
This research was supported by DRPM under the scheme of “Penelitian Tesis Magister”, project No. 793/PKS/ITS/2019. The authors most thank full to the General Director of DIKTI for supporting and to anonymous referees for their useful suggestions. This paper and the research behind it would not have been possible without the exceptional support from them.

References
[1] Pfeifer P E and Deutsch S J 1980 Simulation and Computation 9 (5) 551-562.
[2] Suhartono 2003 Penerapan dan perbandingan model State-Space dengan model Space-Time STAR untuk peramalan data deret waktu dan lokasi. Surabaya: ITS LEMLIT
[3] Ruchjana B N 2002 Forum Statistika dan Komputasi 1-6.
[4] Borovkova S A, Lopuhaa H P, and Ruchjana B 2002 Proceedings of the 17th International Workshop on Statistical Modeling 2002 Eds: Stasinopoulos M and Touloumi G C 143-152.
[5] Wei W W S 2006 Time Series Analysis: Univariate and Multivariate Methods. Second Edition. New York: Addison Wesley.
[6] Zhang G P 2003 Neurocomputing 50 159-175.
[7] Zhang G P and Berardi V L 1998 Health Care Management Science 1 (1) 29-37.
[8] Lewis N D 2016 Deep Learning Made Easy with R. South Carolina: Createspace Independent Publishing Platform.
[9] Wysocki A and Lawrynczuk M 2015 Poland IEEE 149
[10] Lee M H, Suhartono, and Hamzah N A 2010 Regional Conference on Statistical Science. Kuala Lumpur. 349-361.
[11] Wutsqa D U, Suhartono, and Sutijo B 2010 ICMSA Kuala Lumpur 752-761.
[12] Hyndman R J and Koehler A B 2006 International Journal of Forecasting. 679-688.
[13] Makridakis S, Spiliotis E, and Assimakopoulos V 2018 International Journal of Forecasting. 34 802-808.