Malaysian Journal of Computing, 8 (2): 1574-1588, 2023

eISSN: 2600-8238

VULGARIZED NEIGHBOURING NETWORK OF MULTIVARIATE AUTOREGRESSIVE PROCESSES WITH GAUSSIAN AND STUDENT-T DISTRIBUTED RANDOM NOISES

Rasaki Olawale Olanrewaju[1*], Ravi Prakash Ranjan [2], Queensley C. Chukwudum[3], Sodiq Adejare Olanrewaju[4]

[1*,2]*Africa Business School (ABS), Mohammed VI Polytechnic University (UM6P), Morocco.*
[3]*Department of Insurance and Risk Management, University of Uyo, Nigeria.*
. [4]*Department of Statistics, University of Ibadan, Ibadan, Nigeria.*
[1*]olanrewaju_rasaq@yahoo.com, [2]ravi.ranjan@um6p.ma,[3]queensley@yahoo.com, [4]sodiqadejare19@gmail.com.

ABSTRACT

This paper introduces the vulgarized network autoregressive process with Gaussian and Student-t random noises. The processes relate the time-varying series of a given variable to the immediate past of the same phenomenon with the inclusion of its neighboring variables and networking structure. The generalized network autoregressive process would be fully spelt-out to contain the aforementioned random noises with their embedded parameters (the autoregressive coefficients, networking nodes, and neighboring nodes) and subjected to monthly prices of ten (10) edible cereals. Global-α of Generalized Network Autoregressive (GNAR) of order lag two, the neighbor at the time lags two and the neighbourhood nodal of zero, that is GNAR (2, [2,0]) was the ideal generalization for both Gaussian and student-t random noises for the prices of cereals, a model with two autoregressive parameters and network regression parameters on the first two neighbor sets at time lag one. GNAR model with student-t random noise produced the smallest BIC of -39.2298 compared to a BIC of -18.1683 by GNAR by Gaussian. The residual error via Gaussian was 0.9900 compared to the one of 0.9000 by student-t. Additionally, GNAR MSE for error of forecasting via student-t was 15.105% less than that of the Gaussian. Similarly, student-t-GNAR MSE for VAR was 1.59% less than that of the Gaussian-GNAR MSE for VAR. Comparing the fitted histogram plots of both the student-t and Gaussian processes, the two histograms produced a symmetric residual estimate for the fitted GNAR model via student-t and Gaussian processes respectively, but the residuals via the student-t were more evenly symmetric than those of the Gaussian. In a contribution to the network autoregressive process, the GNAR process with Student-t random noise generalization should always be favoured over Gaussian random noise because of its ability to absolve contaminations, spread, and ability to contain time-varying network measurements.

Keywords: *Gaussian Generalized Network Autoregressive (GNAR), Global-a, Nodes, Student-t.*

Received for review: 12-07-2023; Accepted: 26-09-2023; Published: 10-10-2023
DOI: 10.24191/mjoc.v8i2.23103

1. Introduction

Complex dynamic systems have become an intricate part of our world. Many fields like neuroscience, engineering, biology, and finance are driven by these complex systems. One major approach applied to understand the complex behaviors exhibited by these systems are the graph-based models (Halim & Shuhidan 2022). These models have the ability to capture the spatial characteristic of both the static and dynamic processes exhibited by the system as well as the time dependence (Podlubny, 1999; Bai *et al.*, 2019).

More recently, there has been an increasing interest in the use of networks to model time series data. The underlying drive is the need to map time series data onto a multiplex model such as a network and then explore the different distinctive features of the data through the analysis of the complex network. Additionally, the basic assumption guiding a network specifically requires an adequate level of 'association' between the different attributes of the two closely connected vertices (Dahlhaus & Eichler, 2003; Kolaczyk, 2009).

One of the cognate traits and relevancy in complex systems and networking is the feature of graph representation. It is in line with the relevancy that Fortunato (2010) referred to graph networking or representation as community detection. It is otherwise called community structuring or clustering. It is the organization of acmes (vertices) in clusters, such that neighbours of vertices' edges join vertices of the same community structuring, and comparatively few edges join vertices of different clusters. In other words, community detection can be be thought of as autonomous compartments of a graph, or clustering playing a similar role. Community structuring or clustering is of prominence in work areas where systems, and clustering are represented as graphs. Its importance is for identifying modules and their well-defined inter-face that allows the classification of vertices in accordance with their structural position (Gregory, 2009; Dhumal & Kamde, 2015; Krampe, 2019). To model the varying attributes of a dynamic network, Khan & Niazi (2017) considered a scenario where the network model dictates the dependency structure of the time series (the attributes). The framework gave room for both the network model and time series to be modeled separately and then combined within a multivariate doubly stochastic time series setup. Chen (2022) and Bloemheuvel *et al.* (2022) also considered a similar study with the added perspective of incorporating exogeneous regressors to allow the target series to be regressed by its own historical time lags. With a specific focus on sensor networks, Gao *et al*. (2017) exploited spatial and time series information through graph-based neural networks. They developed the TISER-GCN regression neural network for processing long multivariate time series.

Thus, graph-based models can bring to light latent dependencies that exist between the variables in a given dataset. In view of this, quite a number of methods have been developed which specifically deal with the reconstruction of a complex network from either a univariate or a multivariate time series. Some earlier literature has included the multiplex recurrence networks proposed by Zhang & Small (2006) designed to test the oil-water flow to capture the information on spatial flow. Baudry & Robert (2019) developed the pseudo-periodic time series transform algorithm. They focused on investigating the statistical properties of the different chaotic time series networks. Their result indicated distinct topological structures stemming from the hierarchy of unsteady periodic orbits that were already enclosed in the chaotic attractor. Some authors have worked on the generation of time series from deep learning networks. The autoregressive implicit quantile network was designed by Guibert *et al*. (2020) to analyze time series data in a bid to learn in an efficient way the basic time dependencies of the driving stochastic process. In a similar vein, Krampe (2019) developed the random vector functional enlace neural networks and the vector autoregressive elastic-net model-to-model mortality and construct life tables. Nikolaev *et al*. (2013) to model the time-dependent variance, which usually features in time series data, presented a mixing neural network.

Focusing on the application in various fields, Gao *et al*. (2014) presented a computational algorithm applied to quantitatively define autoregressive patterns in a time series based on regression models. Gregory (2009) worked-out the largest connected component of networks of scientist collaborators working at the Santa Fe Institute (SFI). It was affirmed that there were 118 vertices that make-up residing scientists at SFI and their collaborators. Fortunato (2010) confirmed that the edges are the placement between scientists that have published at least one paper together, while it was observed that the network clique connote authors of the same paper are all linked to each other with few connections between most groups. Gao *et al*. (2019) constructed a spatial-temporal convolutional neural network for an electroencephalograph (EEG).

In relation to connecting networks, Ibrahim & Awang (2022) developed a sensor network for spatially dispersed and dedicated monitors in order to access the physical conditions of an organized data measurements extracted at a particular location. They adopted the Compressive Sensing (CS) algorithm in order to enhance long lifetime network, reduced energy consumption, and a simple routing scheme. However, their implemented CS algorithm helped to increase networking life by 9.7%, but failed to incorporate time-varying schemes.

Furthermore, Zhen *et al*. (2019) and Olanrewaju *et al*. (2022) focused on a new methodology of converting a multidimensional time series into a complex network established upon a correlation coefficient matrix. U.S. crude oil price data from twenty-three (23) regions was used for their analysis. In examining the heteroscedasticity displayed by short-term series, An *et al*. (2020) showed that the diversity of the fluctuations affect the length of the short period. These short-term fluctuation series were based on the autoregressive generalized autoregressive conditional heteroscedasticity model having a weighted edge. Given the high rate of death and spread of COVID-19, some studies have applied the generalized network autoregressive (GNAR) time series model to model the cases of deaths/survivors Urrutia *et al*., (2022) and to investigate the economic response of COVID-19 for different countries via purchasing managers' indices. In both cases the GNAR performed significantly well.

Our study leans towards the research undertaken by Knight *et al*. (2020) who analyzed multivariate time series based on an already existing underlying network where each node was assumed to depend not only on its previous values, but also on the previous values of its neighbours as well as further down it the historical tree. In particular, they adopted the Gaussian Independent, and Identically Distributed (iid) assumption for the random noise. This poses some constraints and excludes observations that are heavy-tailed in nature such as financial data. We relax this assumption by adopting the student-*t*-distributed white noise to relax the inclusion of heavy-tailed observations and to allow flexibility in modeling random errors. This clear distinction between the random errors when modeling on complex networks has received little attention in previous studies and a comparative analysis has rarely been undertaken. Culurciello (2017), Olanrewaju & Oseni (2021) and Ibrahim & Awang (2022) noted after an extensive literature review the lack of a neural network that could capture the features exhibited by financial returns, that is, extreme fat tails, and leptokurtic characteristics raised this concern. The author then went ahead to study nonlinear neural networks for forecasting conditional mean and variance where both the Gaussian and student-*t* neural networks were investigated. The conclusions made with respect to the superiority of the latter to the former when dealing with extreme leptokurtic data align with that of Briegel & Tresp (2000) who also incorporated this distinction when constructing their dynamic neural regression model by replacing Gaussian errors with a more flexible noise model that is based on the student-*t*-distribution. They showed experimentally that the student-*t*-distributed noise model gives rise to online learning algorithms that are more stable than their Gaussian error counterparts.

Knight *et al*. (2020) proposed the Generalized Network Autoregressive (GNAR) processes with Gaussian random noise, in this paper, we shall be presenting vulgarized neighbouring network of the multivariate autoregressive process with student distributed random noise as an extension to the GNAR process. We shall be applying both the GNAR and the vulgarized neighboring network of the multivariate autoregressive process with student-*t*-distributed to the monthly-accord wholesale of prices (in naira (#)) of cereals in Kano state, Nigeria. The coefficients and performances of the two processes would be estimated and juxtaposed. In a clearer term, the core and novel objectives of this write-up are that, a vulgarized neighboring network of multivariate autoregressive process with student-*t*-distributed random noise will be expounded as an extension and alternative to the Generalized Networking Autoregressive (GNAR) process, such that the former's parameter estimation was carried-out via Expectation-Maximization (EM) algorithm. In addition, the GNAR and vulgarized neighboring network of the multivariate autoregressive process with student-*t*-distributed random noise will be applied to contaminated time-varying series of prices of cereals in Kano state, Nigeria in order to juxtapose their noise effectiveness.

2. Method

Assuming two or more univariate autoregressive processes ϕ_i at each node/vertex/graph of a network/ data structure depending on both immediate past values of the node and previous time neighbouring nodes give a multivariate time series. In other words, neighbouring nodes will be part of the autoregressive networking structure of the observed time series as proposed by (Knight *et al*., 2020).Working from the (Knight *et al*., 2020)'s perspective: Let $(\mathrm{N}\times 1)$ be the vector nodal time series, $Y_t = \left(Y_{1,t}, \ldots, Y_{N,T}\right)^T$, that is, networking size with N-nodes indexed as $i \in [1, N]$. Describing the network structure within and between the nodes via network structure of time-varying associate weights (otherwise known as linking weight), say "η". The time, $1 \le t \le T$ for the generalized autoregressive process of the order $(k,[\mathrm{m}]) \in ¥ \times ¥_0^k$ of Y_t is

$$Y_{i,t} = \sum_{j=i}^{K} \left(\phi_{i,j} Y_{i,t-j} + \sum_{c=1}^{C} \sum_{m_j \in ¥_0 = r=1}^{m_j} \alpha_{j,n,c} \sum_{p \in ¥_i^{(t)}(i)} \eta_{i,p,c}^{(t)} Y_{p,t-j} \right) + \varepsilon_{i,t} \tag{1}$$

Where,

"k" belongs to the set of natural number as the optimal time lag, $[m] = (m_1, m_2, \ldots, m_k)$ such that, $m_j \in ¥_0$, the maximum phrase of dependence of neighbor at time lag j; $¥_0 = \{0\} \cup ¥$.

$N_t^{(n)}(i)$ is the n^th^-stage of neighbourhood nodal set "i" at time "t".

$\eta_{i,p,c}^{(t)}$ is the linking probability (connecting weight) between nodes "i" and "p" at time "t" with their associated route covariate "c", such that $0 \le \eta_{i,p,c}^{(t)} \le 1$.

ϕ_{ij} is the "j" lag of autoregressive processes at node "i".

ϕ_i is the autoregressive processes at node "i".

η is the networking weight or connecting weight.

N is the networking size of a nodal time.

$\mathbb{N}$ is the set of Natural number

$\mathbb{N}_0$ is the set of Natural number minus zero, that is, $\mathbb{N}-\{0\}$

$\alpha_{j,n,c} \in \mathbb{R}$ is the effect of nth-stage neighbours at lag "*j*" together with covariate $c = 1, \ldots, C$

$\varepsilon_{i,t}$ is the random noise.

Knight *et al.* (2020) adopted the Gaussian Independent and Identically Distributed (iid) Assumption for the random noise, but in this research, the student-*t*-distributed white noise would be adopted in the course of this work to relax the inclusion of heavy-tailed observations and to allow flexibility in modelling random errors. The objective of the study is to estimate embedded parameters, fit a real-life dataset (financial dataset), and predict from the vulgarized neighbouring network of multivariate autoregressive processes with adopted student-*t*-distributed white-noise and compare with generalized network autoregressive processes with independently and identical distributed Gaussian random noise.

2.1. Parameter estimation via expectation-maximization (EM) algorithm

In this section, we shall discuss the parameter estimation of the GNAR model with student-*t* random noise. It is to be noted that the group label is $\eta_{i,p,c}$ is the latent (connecting weight). The parameter estimation and group detection would be constrained to be time variant. We are interested in modeling the dynamics of Y_t such that all the effects are invariant with nodes of homogenous type. To do that we assume the nodes in the network are classified into *k*-groups, where each group is characterized by a specific set of parameters. We start by making some denotations:

Let,

$$G_{it} = \sum_{c=1}^{C} \sum_{m_j \in \mathbb{N}_0}^{m_j} \alpha_{j,n,c} \sum_{p \in \mathbb{N}_i^{(t)}(i)} \eta_{i,p,c}^{(t)} Y_{p,t-j} \tag{2}$$

$$Y_t^{(k)} = \left(Y_{i,t} : i \in \mathbb{N}_k\right) \in \mathbb{R}^{\mathbb{N}_k} \tag{3}$$

$$G_t^{(k)} = \left(G_{it} : i \in \mathbb{N}_k\right) \in \mathbb{R}^{\mathbb{N}_k} \tag{4}$$

$$\Theta_k = \left(\phi_{oj}, \phi_{1j}, \phi_{2j}, \ldots, \phi_{ij}\right)^T \in \mathbb{R}^{\mathbb{N}_k} \tag{5}$$

$$\varepsilon_t^{(k)} = \left(\varepsilon_{it} : i \in \mathbb{N}_k\right) \tag{6}$$

Then Equation (1) could be written as

$$Y_t^{(k)} = G_t^{(k)} \Theta_k + \varepsilon_t^{(k)} \tag{7}$$

Subsequently, the Ordinary Least Squares (OLS) estimator can be obtained for the k^{th} -group as:

$$\hat{\Theta}_k = \left(\sum_{j=1}^{T} G_{t-1}^{(k)T} G_{t-1}^{(k)} \right)^{-1} \left(\sum_{j=1}^{T} G_{t-1}^{(k)T} Y_t^{(k)} \right) \tag{8}$$

Recall, that the latent variable $\eta_{i,p,c}^{(t)}$ which indicates the linking probability (connecting weight) between nodes "*i*' and "*p*" at time "*t*" with their associated route covariate. Let $"\Phi"$ denote the overall parameter space.

So,

$$L(\Phi) = \prod_{i=1}^{N} \prod_{k=1}^{K} \left[\prod_{t=1}^{T} \alpha_k \Upsilon\left(Y_{it}^{(k)} - G_{it}^{T} \Theta_k \right) \right]^{\eta_{i,p,c}} \tag{9}$$

Such that $\Upsilon(.)$ is the Student-*t* Probability Density Function (PDF):

$$\Upsilon(Y_t) = \frac{\sqrt{\left(\frac{v+1}{2} \right)}}{\sqrt{\left(\frac{v}{2} \right)} \sqrt{v\pi}} \left(1 + \frac{Y_t^2}{v} \right)^{-\frac{v+1}{2}} \qquad Y_t \in \left(-\infty, \infty \right) \tag{10}$$

Where "v" is the degree of freedom.

Adopting the EM-algorithm for the parameter estimation, we just need to set an initial value for $\hat{\Phi}^{(0)}$, then iteration of the two steps (EM-algorithm) in the f^{th} $\left(f \geq 1 \right)$ iteration.

First Step (E-Step): Estimating $\eta_{i,p,c}$ by its posterior average $\eta_{i,p,c}^{(f)}$. Here

$$\eta_{i,p,c}^{(k)} = E\left(\eta_{i,p,c} / \hat{\Phi}^{(f-1)} \right) = \frac{\hat{\alpha}_k^{(f-1)} \prod_{t=1}^{T} \Upsilon\left(\hat{\Lambda}_{it,k}^{(f-1)} \right)}{\sum_{k=1}^{K} \hat{\alpha}_k^{(f-1)} \prod_{t=1}^{T} \Upsilon\left(\hat{\Lambda}_{it,k}^{(f-1)} \right)} \tag{11}$$

Where,

$$\hat{\Lambda}_{it,k}^{(f-1)} = \frac{\left(Y_{it} - G_{i(t-1)}^{T} \hat{\Phi}_k^{(f-1)} \right)}{\hat{\sigma}_k^{(f-1)}}$$

(Since Student-*t* distribution is a ratio of independent Normal-variate and Chi-Square distribution with degree of freedom (v)).

Second Step (M-Step): Given $\eta_{i,p,c}^{(f)}$ Equation (11) is then maximize regarding to $\alpha_k, \Phi_k, \sigma_k$ to have

$$\hat{\Phi}_k^{(f)} = \left(\sum_i \eta_{i,p,c}^{(f)} \sum_t G_{it} G_{it}^T\right)^{-1} \left(\sum_i \eta_{i,p,c}^{(f)} \sum_t G_{it} Y_{it}\right) \quad (12)$$

$$\hat{\sigma}_k^{2(f)} = \left(T \sum_i \eta_{i,p,c}\right)^{-1} \left[\sum_i \eta_{i,p,c} \sum_t \left(Y_{it} - G_{it}^T \hat{\Phi}_k^{(f)}\right)^2\right] \quad (13)$$

$$\hat{\alpha}_k^{(f)} = \frac{\sum_i^N \eta_{i,p,c}^{(f)}}{N} \quad (14)$$

The steps are to be repeated until the EM-algorithm converges for desired results of the estimators. It is to be noted that $\hat{\Phi}_k^{(f)}$ of Equation (12) is in a similar spirit to Equation (7). In particular, the EM estimation of Φ_k can also be perceived as a weighted OLS estimator, such that, the weights are the latent group variables $\eta_{i,p,c}$. Additionally, the estimation of σ_k^2 and α_k in Equation (13) and Equation (14) can be fully grasped in a similar way.

3. Numerical results

The monthly-concord wholesale prices (in naira (#)) of cereals in Kano state, Nigeria. The prices of the cereals were from 2007 to 2019 and were recorded in naira (#). The uniformly time variant prices' dataset was obtained from the Ministry of Agriculture and Natural Resources (MANR), Kano state, Nigeria. The prices of cereals in particular are sorghum, groundnut, beans, rice, maize, millet, gcorn, cowpea, wheat, and cassava. The average monthly prices of the cereals were regulated prices of the edible grains by MANR. The time series dataset was obtained from the Ministry of Agriculture and Natural Resources (MANR), Kano state, Nigeria. The dataset was a monthly uniform time-varying harmonized and regulated price of the edible grains by the ministry.

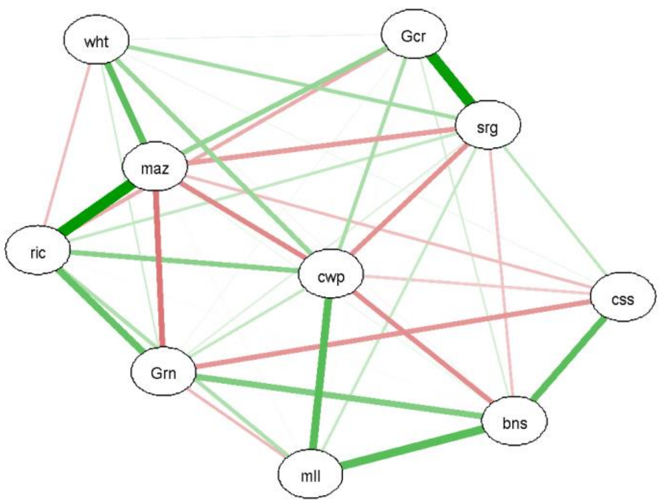

Figure 1. Networking Graph of the Prices of Cereals.

Networking analysis refers to the structuring of variables represented by nodes in relation to edges between nodes. Nodes are sometimes referred to as vertices, and edges otherwise known as links, while networks are called graphs for analysing group-level or individual-level networks based on times series data, longitudinal time series data or cross-sectional data. Based on Figure 1 above, the centrality of the nodes/edges in the network of prices of cereals understudy are Gcr, srg, ric and maz which stand for Gcorn, Sorghum, Rice, and Maize respectively. The betweenness centrality for each centralized nodes mentioned is the shortest paths that cater across the mentioned nodes is the deep green color that connected the price of rice and maize; price of Gcorn and Sorghum. This literally mean closeness in the prices of rice and maize; and Gcorn and Sorghum, compared to others. The other light green colors measure the strength of the eigenvector centrality node, nodes with deeper green color are those that are linked to many other nodes which are in turn, connected to many others. Contrarily, the red colors measure how the centralized nodes were unconnected to many other nodes, which are in turn, connected to many others.

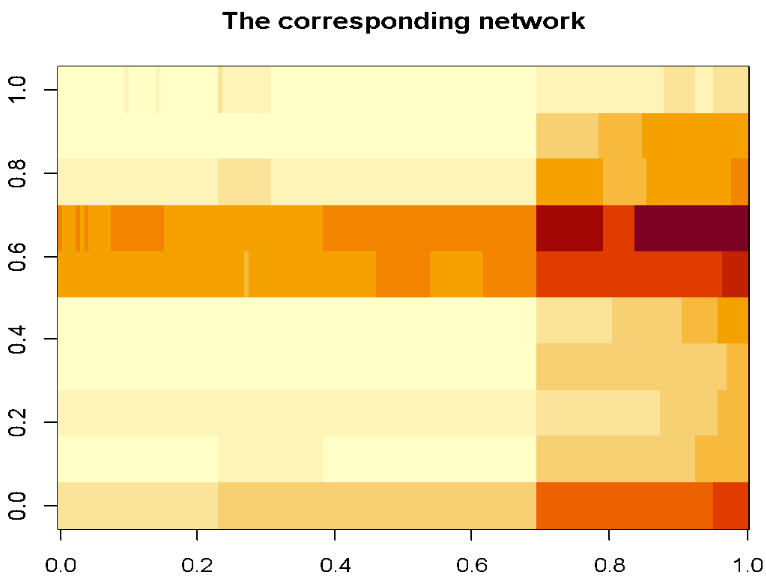

Figure 2. Greyscale Differencing Time Series of the Prices of Cereals.

Clustering is a tool used in operating network analysis of the group of nodes that are based on graph topology. It is otherwise known as community detection. It entails a multivariate time series in line with the real-inferred network that gives insights into inter-variable relationships. Based on Figure 2 above, from the vertical axis (*y*-axis), the deeper inferred from 0.5000 to 0.7000 versus 0.7000 to 1.0000 the same deeper inferred of the *x*-axis suggested an overlapping graph, such that node of the prices of cereals can be partitioned into three sets, the connected, correlated and unordered paired. Concisely, it means 70% (0.7000) of the cereals prices explained and inter-explained one another. It is sometimes called the GNAR autocorrelation.

3.1 Discussion

In time series, we normally select the optimal or best lag (order) of an AR process via its minimum corresponding Partial Autocorrelation Function (PACF) or AutocorrelationFunction (ACF) value. Consequently, the selected lag values for the corresponding BICs instead of the PACF for the study. Instead of using the word "BIC", the "sort" was used because its criteria are for selecting the optimal order. The lag of order two (2) for autoregressive has the minimum sort in Table 1.

Table 1. Selecting the Best GNAR Order using the BIC for the Networking of Autoregressive Process via Student-*t*-distribution

Order	2	7	3	8	4	6	1	2	9
Sort	**-18.2850**	-18.1910	-18.2610	-18.2140	-18.1970	-18.1680	-16.4400	**-18.2850**	-16.4320

The histogram in Table 3 defines the network regression parameters on the first two neighbor sets at time lag two, that is nodal of in- and out-degree of node two (2). The model that minimized BIC, in this case, was the second model, GNAR(2, [2, 0]), a model with two autoregressive parameters and network regression parameters on the first two neighbor sets at a time lag of two. In fact, in order two (2), the sorted BIC was -18.285 for the minimum BIC among others. Among the nine (9) reoccurence PACFs of the same nine (9) lags of the AR process, AR of order two with two nodes gave a mínimum model selection criteria via BIC. The corresponding coefficients of the selected model are in Table 2 and Equation (9). It means we have autoregressive coefficients at optimal at lag two (2) for two nodes. Additionally, there are two effects of the n^{th}-stage neighbors at lag two (2) such that the random noise $\varepsilon_{i,t}$ takes the form of student-*t*-distribution.

Table 2. Coefficients of the GNAR (2, [2, 0]) for the Networking of Autoregressive Process via Student-*t*-distribution.

Coefficients	**Est.**	**Std. Error**	**t-value**	**Pr(>|t|)**	**Residual Error**	**Adjusted R-squared**	**GNAR BIC**	**GNAR MSE**	**node-specific AR models**	**GNAR MSE with VAR model**
dmatalpha1	0.6187	0.0244	25.359	0.0020	0.4373	0.9008	-39.230	0.0073	0.0014	0.0001
dmatphil.1	0.0432	0.0136	3.1880	0.0015						
dmatphil1.2	-0.0099	0.0169	-0.5320	0.5946						
dmatalpha2	0.3117	0.0239	13.0450	0.0020						

Keys: Std.Error = Standard Error; Est.=Estimate

Mathematically,

$$Y_{i,t\ (\text{student}-t)} = \sum_{j=1}^{2}\left(0.04324_{1,j}\, y_{1,t-j} - 0.0099_{2,j}\, y_{2,t-j} + \sum_{c=1}^{C}\left(0.61869_{1,n} + 0.31165_{2,n}\right)0.0014_{c}\, y_{t-j}\right) \tag{15}$$

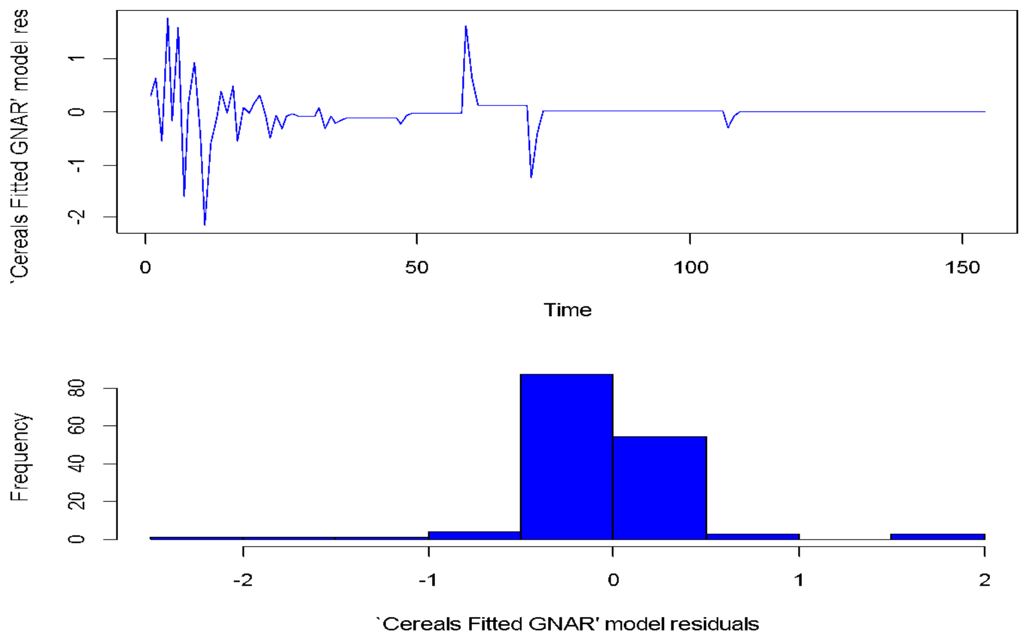

Figure 3. Plot and Histogram of the Residuals for the Student-*t*-GNAR Random Noise.

Table 3. Selecting the best GNAR order using the BIC for the Networking of Autoregressive Process via Gaussian Distribution.

Order	3	4	2	1	6	5	7	8	9
Sort	-39.2640	-39.2630	**-39.2800**	-39.2300	-39.2230	-39.2230	-39.1900	-39.1560	-39.1560

The model that minimized BIC, in this case, was the **second** model, GNAR(2, [2, 0]), a model with two autoregressive parameters and network regression parameters on the first two neighbor sets at a time lag of one. In fact, at order two (2), the sorted BIC was -39.2800 for the minimum BIC among others.

Table 4. Coefficients of the GNAR (2, [2, 0]) for the Networking of Autoregressive Process via Gaussian Distribution.

Coefficients	**Est.**	**Std. Error**	**t-value**	**Pr(>|t|)**	**Residual Error**	**Adjusted R-squared**	**GNAR BIC**	**GNAR MSE**	**node-specific AR models**	**GNAR MSE with VAR model**
dmatalpha1	0.9749	0.0269	36.2250	0.0020	0.1869	0.9971	-18.1683	0.1584	0.8437	0.0146
dmatphil.1	0.0028	0.0029	0.9870	0.3240						
dmatphil1.2	0.0010	0.0037	0.2730	0.7850						
dmatalpha2	0.0278	0.0270	1.0310	0.3030						

Mathematically,

$$Y_{i,t(Gaussian)} = \sum_{j=1}^{2}\left(0.0028_{1,\mathrm{j}}\, y_{1,t-j} + 0.0010_{2,\mathrm{j}}\, y_{2,t-j} + \sum_{c=1}^{C}\left(0.9749_{1,n} + 0.0278_{2,n}\right)0.8437_{c}\, y_{t-j}\right) \tag{16}$$

From the generalized networking autoregressive process coefficients via student-*t*-distribution, the model performance indexes of Residual Error=0.4373, Adjusted R-squared=0.9008, GNAR BIC=-39.2298, GNAR MSE=0.0073, GNAR Mean Squared Error (MSE) with VAR model=0.0001. It implies a residual error of 43% (0.4300) was recorded and that 90% (0.9000) of the time-variant data points of the prices of cereals used contributed to the model as shown in Table 4. The GNAR error of forecasting recorded was tantamount to 0.7340% (0.0073) in comparison to 0.01% (0.0001) GNAR MSE with Vector Autoregressive (VAR) model. GNAR model with student-*t* random noise produced the smallest BIC of -39.2298 compared to a BIC of -18.1683 by GNAR by Gaussian. The residual error via Gaussian was 99% (0.9900) compared to the one of 90% (0.9000) by student-*t*. Additionally, GNAR MSE for error of forecasting via student-*t* was 15.105% (0.1511) less than that of the Gaussian. Similarly, student-*t*-GNAR MSE for VAR was 1.59% (0.0159) (lesser than that of the Gaussian-GNAR MSE for VAR. Comparing the histogram of Figure 3 to Figure 4, the two histograms produced a symmetric residual estimate for the fitted GNAR model via student-*t* and Gaussian processes respectively, but the residuals via the student-*t* were more evenly symmetric than of the Gaussian.

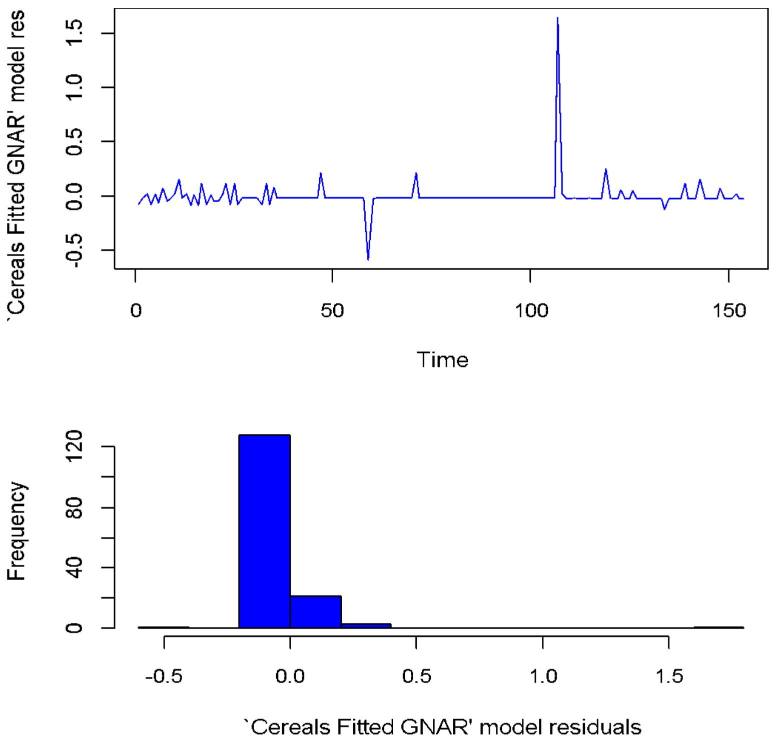

Figure 4. Plot and Histogram of the Residuals for the Gaussian-GNAR Random Noise.

4. Conclusions

Generalized Network Autoregressive (GNAR) process with Gaussian and Student-*t* random noises were proposed. The processes relate the time-varying series of a given variable to the immediate past of the same phenomenon with the inclusion of its neighboring variables and networking structure. In conclusion, Generalized Networking Autoregressive (GNAR) process with student-*t* random noise gave a minimum GNAR BIC=-39.2298, GNAR MSE=0.0073, GNAR Mean Squared Error (MSE) with VAR compared to the GNAR process with Gaussian random noise. Furthermore, the residuals from the fitted GNAR process with student-*t* random noise were more evenly symmetric on the histogram real number line. The GNAR process can be extended to Generalized-Error-Distribution (GED), Extreme-Value-Distributions (EVDs) or missing value distributions.

Acknowledgement

We would like to acknowledge the Ministry of Agriculture and Natural Resources (MANR), Kano state, Nigeria for providing the dataset used in this research.

Funding

The authors received no specific funding for this work.

Author Contribution

Rasaki Olawale Olanrewaju and Ravi Prakash Ranjan wrote the research methodology, conducted the statistical analysis and oversaw the the entire write-up of the research, while Queensley C. Chukwudum and Sodiq Adejare Olanrewaju came-up with the literature and introduction write-ups.

Conflict of Interest

Authors have no conflict of interest to declare.

References

An, S. Gao, X., Jiang, M., Feng, S., Wang, X. & Wen, S. (2020). Dynamic heteroscedasticity of time series interpreted as complex networks Chaos An Interdisciplinary. *Journal of Nonlinear Science*, 30(2), 023-133. doi.org/10.1155/2019/9107167.

Baudry, M. & Robert, C.Y. (2019). A machine learning approach for individual claims reserving in insurance *Applied Stochastic Models in Business and Industry*, 35(1), doi:10.1002/asmb.2455.

Bloemheuvel, S., Van den Hoogen, J. Jozinović, D. et al. (2022). Graph neural networks for multivariate time series regression with application to seismic data. *International Journal of Data Science and Analytics (2022)*. doi:/10.1007/s41060-022-00349-6.

Briegel, T. & Tresp, V. (2000). *Dynamics Neural Regression Models*, Collaborative Research Center 386, Discussion Paper. doi:10.5282/ubm/epub.1571.

Chen , F. (2022). *New methods for multivariate, survival and time series data with networks.* PhD thesis, University of Bristol.

Culurciello, E. (2017). *Neural Network Architectures, Towards Data Science.*

Dahlhaus, R. & Eichler, M. (2003). *Causality and Graphical Models for Time Series, Highly Structured Stochastic Systems* (In PJ Green, NL Hjort, S Richardson (eds.)), Oxford University Press, Oxford, 115–137.

Dhumal, A. & Kamde, P. (2015). Survey on Community Detection in Online Social Networks. *International Journal of Computer Applications*, 121(9), 35-41, doi: 10.5120/21571-4609.

Fortunato, S. (2010). *Community detection in graphs*. Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Viale S. Severo 65, 10133, Torino, Italy. arXiv:0906.0612v2[physics.soc-ph] 25 Jan 2010.

Gao, X., An, H., Wei, F., Huang, X., Li. H. & Zhong, W. (2014). Characteristic of the transmission of autoregressive sub-patterns in financial time. *Sci. Rep*., 4, 6290–6290.

Gao, Z., Dang, W., Yang, Y. & Cai, Q. (2017). Multiplex multivariate recurrence network from multi-channel signals for revealing oil-water spatial flow behavior, *Chaos*, 27, 035809.

Gao, Z., Wang, X., Yang, Y., Mu, C., Cai, Q., Dang, W. & Zuo, S. (2019). EEG-based spatio-temporal convolutional neural network for driver fatigue evaluation. *IEEE Trans. Neural Netw. Learn. Syst.*, 30.

Gregory, S. (2009). in Complex Networks, edited by S. Fortunato, R. Menezes, G. Mangioni, & V. Nicosia (Springer, Berlin, Germany), volume 207 of Studies on Computational Intelligence, 47–62.

Guibert, Q., Piette, P. & Planchet, F. (2020). *Modeling and forecasting mortality with Machine Learning Approaches.* In: Insurance data analytics: some case studies of advanced algorithms and applications. Editors: Planchet, F. and Robert, C.Y.

Halim, Z. & Shuhidan, S.M. (2022). Towards development of robust machine learning model for Malaysian corporation. *Malaysian Journal of Comput*ing, 7(1), 1011-1026. doi.org/10.24191/mjoc.v7i1.16702.

Ibrahim, M.A., Awang, N. ((2022). An Improvement of resource consumption in wireless sensor network (WSN) using compressive sensing. *Malaysian Journal of Computing*, 7(1), 1036-1046. doi.org/10.24191/mjoc.v7i1.15852.

Khan, B.S. & Niazi, M.A. (2017). *Network Community Detection: A Review and Visual Survey Project*: Visual Analysis of Scientometric Networks.

Knight, M., Leeming, K., Nason, G. & Nunes, M.A. (2020). Generalized network autoregressive processes and the gnar package. *Journal of Statistics Software*, 96, 1–36.

Kolaczyk, E. (2009). *Statistical Analysis of Network Data: Methods and Models.* SpringerVerlag, New York.

Krampe, J. (2019).Time series modeling on dynamic networks. *Electronic Journal of Statistics*, 13(2), 4945-4976. doi.org/10.1214/19-EJS1642.

Nikolaev, N., Tino, P. & Smirnov, E.N. (2013). Time-dependent series variance learning with recurrent mixture density networks. *Neroucomputing*, 122, 501-512.

Olanrewaju, R.O., Jallow, M.A. & Olanrewaju, S.A. (2022). An Analysis of the Atlantic Ocean Random Cosine and Sine Alternate Wavy ARIMA Functions. *International Journal of lntelligent Systems and Applications (IJISA)*, 14(5), 22-34. doi:10.5815/ijisa.2022.05.03.

Olanrewaju, R.O. & Oseni, E. (2021). GARCH and its Variants' Model: An Application of Crude Oil Distributions in Nigeria. *International Journal of Accounting Finance and Risk Management*, 6(1), 25-35. doi.10.11648/j.ijafrm.20210601.14.

Podlubny, I. (1999). Bai, Y., Jin, X., Wang, X., Su, T., Kong, J. & Lu, Y. (2019). Compound autoregressive network for prediction of multivariate time serie. *Hindawi Complexity*,11. doi.org/10.1155/2019/9107167.

Urrutia, P., Wren, D., Vogiatzis, C. et al. (2022). SARS-CoV-2 Dissemination using a network of the US Counties. *Operations Research Forum*, 3(29). doi.org/10.1007/s43069-022-00139-7.

Zhang, J. & Small, M. (2006). Complex network from pseudo-periodic time series: Topology versus dynamics. *Phys. Rev. Lett.*, 96, no. 238701.

Zhen, Z., Tian, L. & Lu, J. (2019). Analysis of oil prices' interaction in the USA based on complex networks. *Energy Sources B Econ. Plan. Policy*, 14, 1–12.