International Journal of Mathematics and Statistics Invention (IJMSI)
E-ISSN: 2321- 4767 P-ISSN: 2321-4759
www.ijmsi.org Volume 9 Issue 3 || March, 2021 || PP 32-38

Convergence of Nonlinear Least Square Fit to an Identified Time Series Model, Using Extreme and Guessed Initial Estimates

Uyodhu Amekauma Victor-Edema
Department of Mathematics/Statistics
Ignatius Ajuru University of Education, Port Harcourt. Nigeria.

Abstract: *This paper is an attempt to obtain a linear equation algorithm for a proposed time series model connecting two variables – the Nigerian current account and the exchange rate in United States dollars. The proposed time series model was identified and investigated for adequacy using the modified statistic (Li Mclead Q-test) and then transformed into a linear model. The transformed linear model was estimated by a system of linear equations. A linear equation algorithm and analogous that well explained the behaviour of the variables when the sum of squares function converged was obtained. The identified model fitted was verified for the justification of the simplification by adopting Guessed Initial Estimates or Extreme Initial Estimates recursive methods. The result offers an added verification and offers more backing that the time series model is adequate.*

Keywords: *Time series, Transfer Function Noise Model, Current Account, Modified Statistic, Linear equation algorithm.*

Date of Submission: 29-03-2021 Date of Acceptance: 12-04-2021

I. INTRODUCTION

There are several types of "mixed models" in different disciplines which have been given various names. Longitudinal models, panel data models, transfer function models, dynamic regression models, and linear system models are some of such models. In this study, a mixed model known as the transfer function model in time series analysis was considered. An explanatory model like the transfer function model is suitable because it integrates evidence regarding additional variables, instead of limiting the forecast to only historical values of the variable.

There exist some justifications for the selection of a time series explanatory model by a forecaster instead of a mixed or an explanatory model. The first reason is that the understanding of the system might be a problem and supposed it was implicit it might be somewhat complex to quantify the presumed connections that strongly influence the system and to manage the behaviour of such relationships.

The next reason is that the knowledge or forecast of the upcoming values of the multifarious predictors to facilitate the forecast of the key variables of interest is necessary, and it might be a complex task.

Thirdly, the key problem might be solely to forecast the future event and not just the reasons that led to the occurrence of the event. Lastly, the transfer function model also known as the time series explanatory model is likely to yield more reliable forecasts than other mixed models. The Transfer Function Model (TFM) is used to model an input (Xt) and an output time series (Yt) which is used for predicting upcoming values of the time series. In these models, Xt and Yt stand for the deviations from the symmetry of the structure output and input. In reality, the system is affected by some turbulence with an intention to distort or corrupt some amount of the Transfer Function (TF) predicted output Nt. To derive optimal predicted values with evidence from Xt and Yt series, the identified TF-noise model linking both series, Xt and Yt is needed for model adequacy. Therefore, this paper aims to investigate the identified transfer function model adequacy of the relationship between the output Yt (transfer function model) and the input time series Xt (Exchange rate) in [1]); hoping that the chosen model is hoped to be fascinating, then its future values forecast can be valid and reliable for adaption by policymakers.

Although the utilization of TFM has spanned over four decades, contemporary studies have expanded its application in areas such as scheduling in maintenance practices [2], and [3]. [4] explored the relationship between saving and investment in South Africa, using a transfer function model, where predicted variable yielded indication of imperfect capital mobility. The ARIMA (5,1,0) performed better than other models utilized and it was used also adopted for the process of pre-whitening. The estimated model thereafter was adopted to carry out a forecast for the venture series. The transfer function model diagnostic checking yielded sufficient information to clinch that the ARIMA (5,1,0) is effective, however, no parameters estimation procedure or iterations was done.

Similarly, [5], prescribed the use of the transfer function model as a dominant instrument provided there are suitable conditions for its use. It showed that to generalize the three phases (identification, estimation and checking, application) of Box-Jenkins univariate procedures we need a transfer function model framework.

II. METHODS AND ALGORITHM

A transfer function model is used to define an input time series "X_t" and a matching output time series "Y_t" for a specific physical system. This model can be represented by

$$Y_t = \frac{w(\mathrm{B})}{\delta(\mathrm{B})}\mathrm{X}_{t-b}$$

(1)

and in expanded form as;

$$Y_t - \delta_1 Y_{t-1} - \delta_2 Y_{t-2} - \ldots - \delta_r Y_{t-r} = w_0 \mathrm{X}_{t-b} - w_0 \mathrm{X}_{t-b} - w_1 \mathrm{X}_{t-b-1} - w_2 \mathrm{X}_{t-b-2} - \ldots . w_s \mathrm{X}_{t-b-s}$$

(2)

where b is the delay parameter, $\delta_1, \delta_2, \ldots, \delta_r$ are parameters of the corresponding output time series, and $w_0, w_1, w_2, \ldots, w_s$ are parameters of the input time series.

In reality, the system is affected by some turbulence with an intention to distort or corrupt some amount of the transfer function (TF) predicted output n_t. Therefore, the joint transfer function-noise model is given as;

$$Y_t = \frac{w(\mathrm{B})}{\delta(\mathrm{B})}\mathrm{X}_{t-b} + n_t$$

(.3)

or

$$Y_t = V(\mathrm{B})\mathrm{X}_t$$

(4)

where $$V(\mathrm{B}) = \frac{w(\mathrm{B})}{\delta(\mathrm{B})}\mathrm{B}^b$$

Equation (4) can transform from a nonlinear model to a linear model, that is

$$\mathrm{Y}_t = \phi_1 \mathrm{Y}_{t-1} + \phi_2 \mathrm{Y}_{t-2} + \ldots \phi_r \mathrm{Y}_{t-r} + w_0 \mathrm{X}_{t-1} + w_1 \mathrm{X}_{t-2} + \ldots + w_s \mathrm{X}_{t-s}$$

(5)

where $\phi_1, \phi_2, \ldots, \phi_r$ and $w_0, w_1, w_2, \ldots, w_s$ are parameters to be estimated.

The transformed model can be estimated by systems of linear equations, which is

$$\begin{pmatrix} \sum \mathrm{Y}_{t-1}\mathrm{Y}_t \\ \sum \mathrm{Y}_{t-2}\mathrm{Y}_t \\ . \\ . \\ . \\ \sum \mathrm{Y}_{t-r}\mathrm{Y}_t \\ \sum \mathrm{X}_{t-1}\mathrm{Y}_t \\ \sum \mathrm{X}_{t-2}\mathrm{Y}_t \\ . \\ . \\ . \\ \sum \mathrm{X}_{t-s}\mathrm{Y}_t \end{pmatrix} = \begin{pmatrix} \sum \mathrm{Y}^2_{t-1} & \sum \mathrm{Y}_{t-1}\mathrm{Y}_{t-2} & \cdots \sum \mathrm{Y}_{t-1}\mathrm{Y}_{t-r} & \sum \mathrm{Y}_{t-1}\mathrm{X}_{t-1} & \sum \mathrm{Y}_{t-1}\mathrm{X}_{t-2} & \cdots \sum \mathrm{Y}_{t-1}\mathrm{X}_{t-s} \\ \sum \mathrm{Y}_{t-2}\mathrm{Y}_{t-1} & \sum \mathrm{Y}^2_{t-2} & \cdots \sum \mathrm{Y}_{t-2}\mathrm{Y}_{t-r} & \sum \mathrm{Y}_{t-2}\mathrm{X}_{t-1} & \sum \mathrm{Y}_{t-2}\mathrm{X}_{t-2} & \cdots \sum \mathrm{Y}_{t-2}\mathrm{X}_{t-s} \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ \sum \mathrm{Y}_{t-r}\mathrm{Y}_{t-1} & \sum \mathrm{Y}_{t-r}\mathrm{Y}_{t-2} & \cdots \sum \mathrm{Y}^2_{t-r} & \sum \mathrm{Y}_{t-r}\mathrm{X}_{t-1} & \sum \mathrm{Y}_{t-r}\mathrm{X}_{t-2} & \cdots \sum \mathrm{Y}_{t-r}\mathrm{X}_{t-s} \\ \sum \mathrm{X}_{t-1}\mathrm{Y}_{t-1} & \sum \mathrm{X}_{t-1}\mathrm{Y}_{t-2} & \cdots \sum \mathrm{X}_{t-1}\mathrm{Y}_{t-r} & \sum \mathrm{X}^2_{t-1} & \sum \mathrm{X}_{t-1}\mathrm{X}_{t-2} & \cdots \sum \mathrm{X}_{t-1}\mathrm{X}_{t-s} \\ \sum \mathrm{X}_{t-2}\mathrm{Y}_{t-1} & \sum \mathrm{X}_{t-2}\mathrm{Y}_{t-2} & \cdots \sum \mathrm{X}_{t-2}\mathrm{Y}_{t-r} & \sum \mathrm{X}_{t-2}\mathrm{X}_{t-1} & \sum \mathrm{X}^2_{t-2} & \cdots \sum \mathrm{X}_{t-2}\mathrm{X}_{t-s} \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ \sum \mathrm{X}_{t-s}\mathrm{Y}_{t-1} & \sum \mathrm{X}_{t-s}\mathrm{Y}_{t-2} & \cdots \sum \mathrm{X}_{t-s}\mathrm{Y}_{t-r} & \sum \mathrm{X}_{t-s2}\mathrm{X}_{t-1} & \sum \mathrm{X}_{t-s}\mathrm{X}_{t-2} & \cdots \sum \mathrm{X}^2_{t-s} \end{pmatrix} \begin{pmatrix} \hat{\phi}_1 \\ \hat{\phi}_2 \\ . \\ . \\ . \\ \hat{\phi}_r \\ \hat{w}_0 \\ \hat{w}_1 \\ . \\ . \\ . \\ w_s \end{pmatrix}$$

The first approximation to the solution of the linear equation above is taken by using Gauss-Seidel iteration method. Firstly, a recursive computation of the output from the input time series using a developed algorithm in Jupyter notebook was done as stated in the steps below. Two runs were made of the nonlinear least square technique by two diverse sets of values, which are extreme and guessed initial estimates.

Step 1: The transfer function model is multiplied out to form a linear model and then used for estimating after the estimation of the model parameters, as deliberated.

Step 2: The parameters of the model are estimated with the ordinary least square method (OLSM). The value of the parameters are selected to reduce the Sum of the Squared Residuals (SSR) between the estimated values and the real data.

Step 3: The non-linear approximation method is then used to estimate the identified parameters above to optimize the probability of the observed series given the parameter values. The following criteria are used by the methodology in parameter estimation.

At a minimal change of 0.001 in all parameter estimates between iterations, the estimation process stops.

At a minimal change of 0.001 in the SSR between iterations, the parameters estimation procedure stops.

Step 4: Based on the diagnosis verification step, the residuals from the fitted model are inspected alongside adequacy. It is achieved through correlation via the goodness-of-fit test and residual plots of ACF using the Chi-Square test. The model is then refined in step one above if the residuals are correlated. Else, the autocorrelations are white noise and the model is suitable for the representation of our time series [6]..

Note: The factored model could be checked and refitted to show the justification of the simplification using Guessed Initial Estimates (GIE) or Extreme Initial Estimates (EIE) recursively method.

However, every model building required the following steps: (1) Identification, (2) Estimation, (3) Diagnostics Checking, and (4) Forecasting (where intended).

III. DIAGNOSTICS CHECKING OF TRANSFER FUNCTION WITH NOISE MODEL

Before accepting the model in Equation (2.4) to be an appropriate illustration of the system, autocorrelation and cross-correlation verifications were adopted, using the modified statistic (Li Mclead Q-test)) approximate test for model adequacy. The test was done under the following hypothesis;

Hypothesis:

H_0: The identified TFM is adequate

H_1: The identified TFM is not adequate

However, the modified statistics for testing the transfer function model is

$$\hat{Q} = m(m+2)\sum_{k=1}^{K}(m-k)^{-1} r^2_{e_t e_t}$$

(6)

where $\hat{Q}$ is defined in terms of the ARIMA model situation, m is the number of parameters, k is the number lag considered and $r^2_{e_t e_t}$ is the k cross-correlation of the residual $\hat{e}_t$. $\hat{Q}$ is roughly circulated as chi-square (χ^2_α) with n-p-q degrees of freedom(df). It is worthy of note that the df in χ^2_α depends not on the number of parameters in the transfer function model; where m= n-p-q but on the number of parameters in the noise model.

IV. RECURSIVE COMPUTATION OF OUTPUT FROM THE' INPUT TIME SERIES USING A DEVELOPED ALGORITHM IN JUPYTER NOTEBOOK

Algorithm / Source Code

```
  def RSS(X, Y, m):
    temp = 0
    for I in range(X[0].shape[0]):
      predicted_value = ((m[0]*X[0][i]) +  (m[1]*X[1][i]) +  (m[2]*X[2][i]) +  (m[3]*X[3][i]) +  (m[4]*X[4][i]) +
(m[5]*X[5][i]))
      actual_value = Y[i]
      temp = temp + ((predicted_value – actual_value)**2)
    return (np.sqrt(temp/X[0].shape[0]))
  def predictedValueForSpecificRow(X, I, m):
    return (m[0]*X[0][i]) +  (m[1]*X[1][i]) +  (m[2]*X[2][i]) +  (m[3]*X[3][i]) +  (m[4]*X[4][i]) +  (m[5]*X[5][i])
# GRADIENT DECSCENT FUNCTION
def gradientDescentAlgorithm(X, Y, learning_rate):
  print('Trianing MLR model using Gradient Descent')
```

```
maximum_iterations = 400
has_converged = False
rows_count = X[0].shape[0]
m = [0.04, -0.01, 0.05, -3000, 1200 , 1500]
error = RSS(X, Y, m)
print('initial value of RSS(Cost Function) is {} when model parameter is {}'.format(error, m))
I = 0
while not has_converged:
  g1 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[0][i] for I in range(rows_count)])
  g2 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[1][i] for I in range(rows_count)])
  g3 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[2][i] for I in range(rows_count)])
  g4 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[3][i] for I in range(rows_count)])
  g5 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[4][i] for I in range(rows_count)])
  g6 = (1.0/ rows_count)*sum([(predictedValueForSpecificRow(X, I, m) – Y[i])*X[5][i] for I in range(rows_count)])
  temp1 = m[0] – learning_rate * g1
  temp2 = m[1] – learning_rate * g2
  temp3 = m[2] – learning_rate * g3
  temp4 = m[3] – learning_rate * g4
  temp5 = m[4] – learning_rate * g5
  temp6 = m[5] – learning_rate * g6
  m[0] = temp1
  m[1] = temp2
  m[2] = temp3
  m[3] = temp4
  m[4] = temp5
  m[5] = temp6
  current_error = RSS(X, Y, m)
  if I % 20 == 0:
    print('iteration {} Current value of RSS is {} based on updated values of model parameters:{}'.format(I + 1, np.round(current_error), np.round(m, 2)))
  35rror = current_error
  I = I + 1
  if I == maximum_iterations:
    print('Training process halted as number of iteration maxed up')
    has_converged = True
return np.round(m, 2)
```

V. RESULTS

Hence, the TFM with added noise (Edema 2020) is

$$Y_t = Y_{t-1} + w_0 X_{t-2} + \frac{e_t}{1 - \phi_1 B - \phi_2 B^2}$$

(7)

VI. DIAGNOSTIC CHECKING

Prior to the acceptance of the model in equation (6) as a satisfactory illustration of the system, cross-correlation checks, and autocorrelation were applied, using the modified statistics approximate hypothesis test for model adequacy in section three, equation (6); That is;

$$\hat{Q} = m(m+2)\sum_{k=1}^{K} \frac{r_{e_t e_t}^2}{m-k} =$$
(8)

where m=n-p-q=57-2-0=55, K=20 and $\sum_{k=1}^{20}(m-k)^{-1} r_{e_t e_t}^2$ = 0.0057.

Comparison of $\hat{Q}$ with the χ_{α}^2 table for $K - p - q = 20 - 2 - 0 = 18$ ($\chi_{0.05}^2(18)$ =9.39) df offers no basis for questioning the adequacy of the model.
The model in equation (6) is nonlinear. Parameters can be used as rough estimates. Hence, the identified transfer function model with added noise is adequate. Conversely, to attain parsimony in parameterization, we simplify the model by factorization and the least square approximation technique turns out to be tremendously possible since the error minimum appears to lie on a surface on the parameter space.
We have

$$(1-\mathrm{B})Y_t = w_0 \mathrm{X}_{t-2} + \frac{e_t}{1-\phi_1\mathrm{B}-\phi_2\mathrm{B}^2}$$

$$(1-\mathrm{B})\left(1-\phi_1\mathrm{B}-\phi_2\mathrm{B}^2\right)Y_t = \left(1-\phi_1\mathrm{B}-\phi_2\mathrm{B}^2\right)w_0\mathrm{X}_{t-2} + e_t$$

$$\left(1-(1+\phi_1)\mathrm{B}-(\phi_1-\phi_2)\mathrm{B}^2+\phi_2\mathrm{B}^3\right)Y_t = \left(w_0 - w_0\phi_1\mathrm{B} - w_0\phi_2\mathrm{B}^2\right)\mathrm{X}_{t-2} + e_t$$

$$Y_t = (1+\phi_1)Y_{t-1} + (\phi_1-\phi_2)Y_{t-2} - \phi_2 Y_{t-3} + w_0\mathrm{X}_{t-2} - w_0\phi_1\mathrm{X}_{t-3} - w_0\phi_2\mathrm{X}_{t-4} + e_t$$
(9)

Substitute the parameters ϕ_1 = 0.4689, ϕ_2 = 0.5346 and w_0 =-39977.93 into Equation (3.3) gives

$$Y_t = 1.47Y_{t-1} - 0.066Y_{t-2} - 0.54Y_{t-3} - 4.0\times10^4\mathrm{X}_{t-2} + 1.87\times10^4\mathrm{X}_{t-3} + 2.14\times10^4\mathrm{X}_{t-4} + e_t$$
(10)

[7] suggested that the least square estimates and their approximate standard errors can be used to obtain parameters of the transfer function model when performing generalized least squares estimation of the regression equation (10) assuming the noise e_t follows some autocorrelated time series ARIMA model. A linear equation algorithm and analogous that will behaviour well when the sum of squares function converged. The factored model may be checked and refitted to show the justification of the simplification by using Guessed initial estimates or extreme initial estimates recursively method.

VII. ESTIMATION OF THE NON LINEAR TRANSFER FUNCTION MODEL

Using the guessed initial estimates recursively method, the estimates are derived with the conditional least square algorithm described in Section (II) to obtained Table I.

Let $Y_t = \hat{\phi}_1 Y_{t-1} + \hat{\phi}_2 Y_{t-2} - \hat{\phi}_3 Y_{t-3} + \hat{w}_0\mathrm{X}_{t-2} - \hat{w}_1\mathrm{X}_{t-3} - \hat{w}_2\mathrm{X}_{t-4} + e_t$
(10)

where $\hat{\phi}_1 = 1+\phi_1$, $\hat{\phi}_2 = \phi_1 - \phi_2$, $\hat{\phi}_3 = \phi_2$, $\hat{w}_0 = w_0$, $\hat{w}_1 = w_0\phi_1$ and $\hat{w}_2 = w_0\phi_2$

Set the guessed initial estimates ϕ_1 = 0.1, ϕ_2 = 0.5 and w_0 =-10000

Table I: Convergence of Nonlinear Least Square Fit to the Data set, Using Guessed Initial Estimates

Iteration	$\hat{\phi}_1$	$\hat{\phi}_3$	w_0	**Sum of Square**
1	0.3015	0.2855	-9700.95	2718498
21	0.6773	0.6613	-9330.36	12349
41	0.6773	0.6613	-9330.36	12349
61	0.6773	0.6613	-9330.36	12349
81	0.6772	0.6613	-9330.36	12348

101	0.6772	0.6613	-9330.36	12348
121	0.6772	0.6613	-9330.36	12348
141	0.6772	0.6613	-9330.35	12348
161	0.6771	0.6613	-9330.35	12348
181	0.6771	0.6613	-9330.35	12347
201	0.6771	0.6613	-9330.35	12347
221	0.6771	0.6613	-9330.35	12347
241	0.6770	0.6613	-9330.35	12347
261	0.6770	0.6613	-9330.34	12347
281	0.6770	0.6613	-9330.34	12346
301	0.6770	0.6614	-9330.34	12346
321	0.6769	0.6614	-9330.34	12346
341	0.6769	0.6614	-9330.34	12346
361	0.6769	0.6614	-9330.34	12346
381	0.6769	0.6614	-9330.33	12345

Footnote: ϕ_1 equal to coefficient of Y_{t-1} "$(1-\hat{\phi}_1)$"; ϕ_2 equal to coefficient of Y_{t-3} "$\hat{\phi}_3$" and w_0 equal to coefficient of X_{t-2} "$\hat{w}_0$".

From Table I, the parameters are ϕ_1 = 0.3231, ϕ_2 = 0.6614 and w_0 = -9330.33 compare to the estimated parameters ϕ_1 = 0.4689, ϕ_2 = 0.5346 and w_0 =-39977.93. This result shows that guessed initial values method does not give a close estimate to two of parameters, before the convergence of nonlinear least square fit to the data set.

Set the extreme initial estimates ϕ_1 = 0.40, ϕ_2 = 0.50 and w_0 = -30000.0

TableII: Convergence of Nonlinear Least Square Fit to the data set, Using Extreme Initial Estimates

Iteration	ϕ_1	ϕ_2	w_0	Sum of Square
1	0.2183	0.2184	-29780.21	2039372.00
21	0.5002	0.5004	-29500.01	12515.00
41	0.5002	0.5004	-29500.01	12515.00
61	0.5002	0.5004	-29500.01	12514.00
81	0.5002	0.5004	-29500.01	12514.00
101	0.5001	0.5004	-29500.00	12514.00
121	0.5001	0.5004	-29500.00	12514.00

141	0.5001	0.5004	-29500.00	12514.00
161	0.5001	0.5004	-29500.00	12513.00
181	0.5000	0.5004	-29500.00	12513.00
201	0.5000	0.5004	-29500.00	12513.00
221	0.5000	0.5004	-29490.99	12513.00
241	0.5000	0.5004	-29490.99	12513.00
261	0.4999	0.5004	-29490.99	12512.00
281	0.4999	0.5004	-29490.99	12512.00
301	0.4999	0.5004	-29490.99	12512.00
321	0.4999	0.5004	-29490.99	12512.00
341	0.4998	0.5004	-29490.98	12512.00
361	0.4998	0.5004	-29490.98	12511.00
381	0.4998	0.5004	-29490.98	12511.00

Footnote: ϕ_1 equal to the coefficient of Y_{t-1} "$\left(1-\hat{\phi}_1\right)$"; ϕ_2 equal to the coefficient of Y_{t-3} "ϕ_3" and w_0 equal to the coefficient of X_{t-2} "$\hat{w}_0$"

In Table II, the parameters are ϕ_1 = 0.5002, ϕ_2 = 0.5004 and w_0 = -29490.98 compare to the estimated parameters ϕ_1 = 0.4689, ϕ_2 = 0.5346 and w_0 =-39977.93. This result shows that the extreme initial values method gives closer estimates to all three parameters, before the convergence of nonlinear least square fit to the data set.

In addition, Table I shows that convergence occurs after six iterations while convergence occurs after five iterations in Table II. Based on the available results, in realistic conditions, several inputs can be handled without serious estimation difficulties. Also, forecasting can be done using the identified transfer function model with noise.

VIII. CONCLUSION

The transfer function–noise model result offers an extra verification and offers more backing to the time series. Finally, the identified transfer function noise model linking the series Y_t and X_t is adequate (or fit to the series Y_t and X_t).

REFERENCES

[1]. Victor-Edema, U. A., & Essi, D. I., (2020). A transfer function modelling of Nigerian current account (net) and exchange rate. *International Journal of Statistics and Applied Mathematics.* 5(4). 177-185.

[2]. Nwobi-Okoye, C. C. & Igboanugo, A. C. (2012). Performance evaluation of hydropower generation system using transfer function modelling. *International Journal of Electrical Power,* 43(1), 245-254.

[3]. Nwobi-Okoye, C. C., Okiy, S. & Igboanugo, A. C. (2015). Performance evaluation of multi-input-single-output (MISO) production process using transfer function and fuzzy logic: case study of a brewery. *Ain Shams Engineering Journal*. 6(2). 541-551.

[4]. Moroke, N. D. (2015), "Box-Jenkins Transfer function Framework Applied to saving-Investment Nexus in the South African context". *Journal of Governance and regulation*; 4(1).

[5]. Makridakis S., Wheelwright S. C, and Hyndman, R. J. (1998). "Forecasting methods and applications" John Wiley and Sons. Inc.

[6]. Naill, P. E. & Momani, M. (2009). Time series analysis model for rainfall data in Jordan: case study for using time series analysis. *American Journal of Environmental Sciences.* 5(5), 599-604.

[7]. Pankratz, A. (1991). *Forecasting with dynamic regression models.* Wiley, New York.

Uyodhu Amekauma Victor-Edema. "Convergence of Nonlinear Least Square Fit to an Identified Time Series Model, Using Extreme and Guessed Initial Estimates." *International Journal of Mathematics and Statistics Invention (IJMSI)*, vol. 09(03), 2021, pp. 32-38.