2020

Forecasting Kenya's GDP using a hybrid neural network and ARIMA model

Ngige, I. Wanjiru
Strathmore Institute of Mathematical Sciences
Strathmore University

Forecasting Kenya′s GDP using a Hybrid Neural Network and ARIMA Model

Ngige Isabel -067119

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Statistical Science at Strathmore University
Strathmore Institute of Mathematical Sciences
Nairobi, Kenya.

March 23, 2020

Declaration

Declaration by Student
I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Student′s name: Ngige Isabel Wanjiru

Signed: ______________________________

Date: ______________________________

Approval by Supervisors
This study thesis has been submitted for appraisal with my approval as university supervisor.

Supervisor′s name: Dr. Collins Odhiambo

Signed: ______________________________

Date: ______________________________

Supervisorś name: Dr. Hellen Osiolo

Signed: ______________________________

Date: ______________________________

Abstract

Background: Gross Domestic Product (GDP) is the market value of goods and services produced within a selected geographical area usually a country in a selected interval in time often a year and can be measured and forecasted in different ways for use by governments and other market participants.Specific users of information on GDP analysis include the United Nations′ Sustainable Development Goal assessment whose key indicator is economic growth as measured by GDP and the joint International Monetary Fund-World Bank methodology for conducting standardized debt-sustainability analyses in low-income countries.

Objective:The main objective of this study was to assess the superiority as suggested by Literature of a Hybrid Autoregressive Integrated Moving Average(ARIMA) and feed forward Artificial Neural Network (ANN) model over a pure ARIMA model in forecasting Kenya′s GDP.

Methods: The ARIMA and the additive ANN-ARIMA Hybrid model is used to forecast absolute GDP values and the comparative forecast accuracy is tested using the RMSE and visualization plots.The Box-Jenkins methodology is used to fit the ARIMA model while the feed-forward Neural Network Autoregressive(NNAR) structure is used to model the neural network portion of the hybrid model .

Results: The data analysis results indicate that the Hybrid model made up of the ARIMA(2,2,1) model and the NNAR (5,2) model does not outperform the pure ARIMA model in forecasting Kenya's short term out of sample GDP based on the results in which a value higher than the ARIMA model's Root Mean Squared Error (RMSE) value is obtained.The ARIMA forecasts outperform the Hybrid model forecasts by 30% on the same basis.However when the two models are compared against an industry benchmark- the IMF GDP forecasts, the Hybrid model yields forecasts closer to this benchmark than the ARIMA forecasts.In conclusion, the Hybrid model has great potential to compete favourably with the ARIMA model in forecasting short term GDP, if a method that accurately rather than arbitrarily specifies the optimum NNAR parameters is developed to reduce the computation costs of repeated trials that has been used in this study to obtain the Neural

Network specifications of the Hybrid model.This is further supported by the Diebold and Mariano statistic that pointed towards equal forecast accuracy ability between the two models.
Keywords: Autoregressive Integrated Moving Average,Artificial Neural Network,Gross Domestic Product,Neural Network Autoregressive, Root Mean Squared Error, International Monetary Fund

Contents

List of Figures

List of Tables

List Of Abbreviations

ARIMA Autoregressive Integrated Moving Average

ARMA Autoregressive Moving Average

AIC Akaike Information Criteria

ANN Artificial Neural Network

ACF Autocorrelation Function

ADF Augmented Dickey Fuller

BP Back-propagation

DWT Discrete Wavelet Transform

GARCH Generalized Autoregressive Conditional Heteroscedasticity

GDP Gross Domestic Product

IMF International Monetary Fund

IID Independent and Identically Distributed

LDCs Least Developed Countries

MLFN	Multilayer Feed Forward Neural Network
NN	Neural Network
NNAR	Neural Network Autoregressive
PACF	Partial Autocorrelation Function
RNN	Recurrent Neural Network
RMSE	Root Mean Squared Error
SAM	Social Accounting Matrix
SDGs	Sustainable Development Goals
TDNN	Time Delay Neural Network
VAR	Vector Autoregressive

Chapter 1

Introduction

1.1 Background to the study

The Gross Domestic Product (GDP) is the market value of goods and services produced within a selected geographical area usually a country in a selected interval in time often a year (Leamer, 2008). Nominal GDP is the market value in current prices of goods and services produced within the geographic borders of a country. Real GDP is the value in base year prices of goods and services produced within the geographic borders of a country. Real GDP is often preferred in modelling to uncover the true trend of GDP and to eliminate price shocks thus stabilizing variance since isolating and then removing inflation from the equation, provides a more accurate picture of a nation′s actual increases or decreases in economic output .

GDP growth statistics has been a primary way in which national economic performance is tracked since it is the aggregate statistic of all economic activity and it captures a broader coverage of the economy than other macro-economic variables (Wabomba, 2016). Moreover, there is demand for forecasts of major economic variables since the monetary policy in some countries is set with reference to expectations of output growth (Camba, 2001). Financial markets′ participants who rely on the impact of such policy frameworks to make decisions in the market also have a keen interest on such predictions of GDP. Furthermore, data on GDP is regarded as an important index for assessing the national economic development and for judging the operating status of macroeconomic activity as a whole (Ning, 2010).

GDP can be measured in three ways. The Expenditure approach consists of the cumulative household, business and government purchases of goods and services and net exports. The Production approach which is equal to the sum of the value added at every stage of production (the intermediate stages) by all industries within the country, plus taxes and subsidies on products in the period. The Income approach, which is equal to the sum of all factor income generated by production in the country (the sum of remuneration of employees, capital income, and gross operating surplus of enterprises i.e. profit, taxes on production and imports less subsidies) in a period (Wabomba, 2016).

Statistics on GDP are a vital basis for governments to set down economic developmental strategies and economic policies.GDP growth data can also be used to predict economic recessions. GDP growth data is essentially useful in planning a country′s yearly financial budget by pointing out areas of underdevelopment that require more funding or potential industries in which further investment into by the government can spur economic growth.

According to Hyndman (2018) forecasting in general can be done using different methods and models such as dynamic regression models, ARIMA Models and advanced forecasting methods such as the Neural Network Model (Hyndman, 2018). Different studies on GDP growth have used different categories of prediction models such as linear and non-linear regression models, time series models and artificial neural network based models to forecast GDP growth. Statistical linear regression models include the Autoregressive model (AR), Moving Average model (MA) and the Autoregressive Integrated Moving Average model (ARIMA). ARIMA models work best when data exhibits a stable or consistent pattern over time with a minimum amount of outliers such as is portrayed by developing countries' GDP. Other statistical models used in forecasting include non-linear multiple regression models such as Bilinear models, Generalized Autoregressive Conditional Heteroskedastic (GARCH) models and the Threshold Autoregressive (TAR) models. Econometric models used in forecasting GDP usually correlate GDP growth to other different macro-economic variables.

Artificial Intelligence Expert systems such as Artificial Neural Network models are analytic techniques modeled on the learning processes inspired by biological systems, particularly that of the human cognitive system and the neurological functions of the brain (Tamba, 2018). Neural Networks exhibit a high learning capability feature that makes them formidable in prediction. Integrated and mixed/hybrid models have also been used in forecasting economic phenomena. In Kenya for example, The Central Bank forecasts foreign exchange rates using a model derived from an analysis of both the Dornbusch-Frankel sticky price model and the portfolio balance models leading to a sticky-price hybrid model known as monetary/portfolio hybrid model (Were, 2013).

These models have been applied by different authors and practitioners in modeling GDP.Were et. al., 2013 built a class of ARIMA time series models to forecast Kenyan GDP. Camba (2001) used a dynamic factor analysis method to come up with an Automatic Leading Indicator Methodology for GDP forecasts which they compared to a Vector Autoregressive model, a Univariate Autoregressive model and a Bayesian Vector Autoregressive model.

The Kenyan GDP has over the years exhibited an increasing trend alongside other East African countries such as Uganda and Tanzania albeit below other bigger African economies such as South Africa, Egypt and Nigeria.

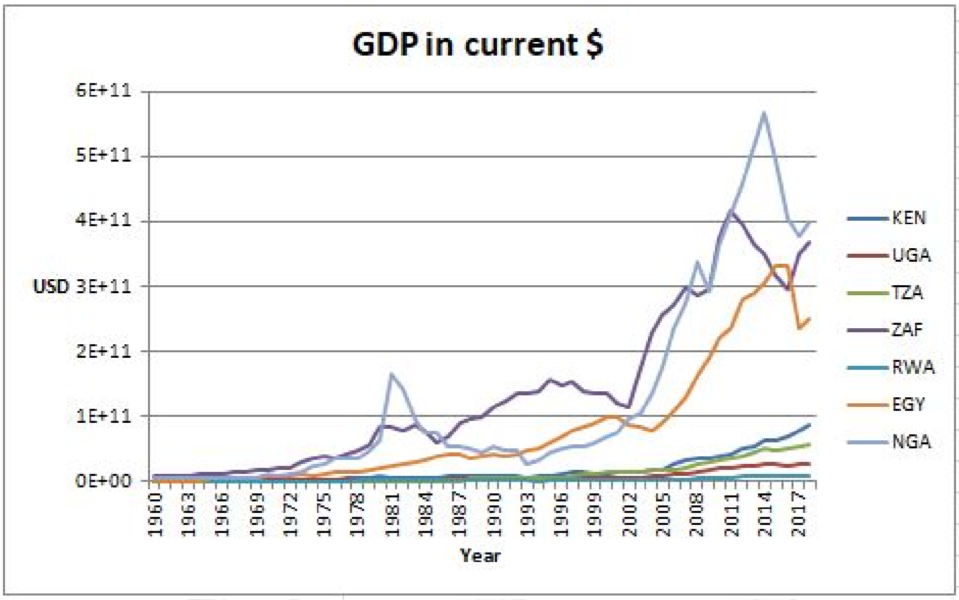

Figure 1.1: African Countries' Comparative GDP in Current USD terms (Source: The World Bank Metadata)

Neural networks is a non-linear regression technique that is appealing to use to study problems in economics and finance with poorly defined system models or noisy data or data with a strong presence of non-linear effects (Aminian et. al., 2006). According to the findings of Aminian et. al., (2006) a neural approach captures a significantly larger aspect of the ability of volatile financial markets to predict economic activity which consequently is an indicator to the existence of a non-linear component of the relationship.

ANN have also been used to model other economic phenomena. Binner (2005) compared a neural network model to ARIMA and VAR in the prediction of the inflation rate of the Euro and they concluded that such linear models (VAR and ARIMA) are a subset of non-linear models such as Neural

Networks (**?**). Tkacz (2001) also used neural networks to model real GDP of Canada and concluded that the neural network produces superior year-to-year forecasts of real GDP growth relative to all other models (Tkacz, 2001).

Artificial Neural Networks have expansively been used in other fields such as Energy, Agriculture and Medicine. In the Energy sector, Tamba et. al., (2018) analyzed as part of their literature survey on forecasting natural gas, over ten studies and publishing conducted from the year 2000 to 2015 on Neural Networks as used in forecasting and modeling energy consumption and production. In Agriculture, Neural Networks have been used in predicting rainfall since statistical techniques for rainfall forecasting cannot perform well for long-term rainfall forecasting due to the dynamic nature of climate phenomena (Tkacz, 2001).

A downfall to measuring the superiority of neural networks to linear models in prediction is that there is no standard technique to measure the relative forecasting advantage of a neural network model relative to a linear model (Aminian, 2006).This has resulted in the use of different methodologies for comparisons of results.

Researchers do not agree on what should be considered a good performance measure or not but prediction accuracy is the most important measure of performance (22). An accuracy measure is defined as the difference between the desired and the predicted value. The most prominent performance measures which may be used to indicate the prediction superiority of a model include; the mean absolute percentage error, mean squared error, root mean squared error, sum of squared error, mean absolute deviation and coefficient of determination.

The Diebold statistic, Diebold and Mariano(1995), tests hypothesis of equal predictive ability in two competing forecasts generated by different forecasting techniques. The root mean squared forecast error (RMSFE) statistic is also used to assess model superiority where the model with the least value is considered best since it represents a lower level of deviation of the forecasts from the actual value.

Another class of models used in GDP studies to monitor the effect of policy changes on GDP is the Computable General Equilibrium Models based on the Social Accounting Matrix framework which have been used to estimate the effect of changes in one part of the economy upon the rest.For example, Karadag and Westaway (1999) have analyzed the effects of policy change on VAT of consumer goods for the Turkish economy.This approach has been intensively used to examine the income generating process of countries to account for income generation from non-industrial demand sources.A more elaborate use of it is the development of a a three-sector social accounting matrix-based (SAM) macroeconomic model of China that generated simulation exercises to observe various effects of macroeconomic structural and policy changes, on the patterns of energy production and consumption given that China′s economic growth largely depends on industrialization which necessitates the increasing use of fossil fuel energy as an input for production leading to a series of negative externalities such as increased $C0_2$ emissions (Jiang et. al., 2017).

1.2 Problem Statement

The accurate analysis and prediction of GDP has great theoretical and practical value (Ning 2010). Raw historical and current data on GDP cannot be used to frame suitable economic development strategies, economic policies and to allocate funds to different priorities for government.It needs a reliable estimate of GDP in some period ahead, which is only possible by forecasting GDP as accurately as possible using a suitable model(Wabomba, 2016).The Monetary Policy Committee of the Central Bank of Kenya uses a Keynasian tradition based macro-econometric model in obtaining short-term forecasts of economic growth.However, complex macroeconomic models have been shown to be outperformed by the well-specified data-driven statistical benchmark models.In forecasting problems the principle of parsimony forces researchers to abandon complex models that are tweaked to the observed data in favour of simpler models that can generalize to new data sets (Vandekerckhove, 2015).This calls for the need to develop, for comparison purposes a modern parsimonious data-driven model as a benchmark for the computationally expensive macro-econometric model as used in Kenya.

Governments as well as other economic market participants rely on the country′s GDP outlook such that inaccurate forecasts of GDP may result in inappropriate decisions and policy setting which has the direct or indirect impact of impeding growth and development.Such misinformation is likely to reverse the very objectives of forecasting with significant cost implications, giving rise to the need for painstaking modeling using models that incorporate correct,relevant and well-informed assumptions for accurate and precise forecasting (Wallison, 2018).

Inappropriate or restrictive assumptions used in modeling economic data can adversely affect results and in turn impair the use of such results.Two models that have predominantly been used in modeling the GDP of different countries is linear models such as ARIMA as well as non-linear models such as the Feed-forward Artificial Neural Network . ARIMA models′ key assumption is that the underlying process is linear and therefore best models purely linear time series.However, according to Zhang (2003), real world economic time series are rarely purely linear or non-linear and often contain both linear and non-linear patterns thus the inadequacy of linear models to capture non-linearities in a mixed pattern data prompts for the use of a non-linear regression technique such as Artificial Neural Networks which have no presumed structure of the underlying relationship in data. Artificial Neural Networks are universal approximators that can be applied to a wide range of time series forecasting problems with a high degree of accuracy solving the issue posed by the linear ARIMA models.ANNs however have a number of shortcomings such as lack of a solid theoretical basis to explain its functionality and the need for subjective judgement in determining the number of input nodes and hidden layer nodes starting point.

However, when higher forecasting accuracy is needed then both theoretical and empirical findings have indicated that integration of different models can be an effective way of improving upon the individual model′s predictive performance, especially when the models in the ensemble are quite different (Khashei and Bijari, 2012). Since the modeling capability shortcomings of the individual models have given rise to the use of hybrid models and these hybrid architecture based models have been shown to perform better than their constituent models as illustrated by Taskaya and Ahmad (2005) studies,then this study suggests the use of a data driven hybrid model that encompasses both the linear based ARIMA and the non-linear based Feed-forward neural network, this will better capture both linear and non-linear aspects characteristic of GDP data with the potential result of more accurate forecasts for the Kenyan GDP.Khashei and Bijari(2012), Zhang (2003) and Jiang et al (2019) apply the hybrid model to forecast different phenomena.More specifically,GDP for developed countries have been modeled using hybrid models. However the model has not been applied to developing countries′ data which is uniquely different. This study therefore provides the first attempt to forecast the Kenyan GDP based on a hybrid ARIMA-ANN model.

1.3 Research Objectives

1. To assess the superiority of a hybrid ARIMA and feed forward ANN model over a pure ARIMA model in forecasting Kenya′s GDP.

2. To compare the accuracy of forecasts produced by different parameter specifications of the constituent models.

3. To quantify the extent to which the hybrid model improves on the

short-term forecasts of the pure ARIMA model.

4. To compare the hybrid model′s forecasts with forecasts from the International Monetary Fund.

1.4 Research Questions

1. How does the hybrid ARIMA- feed forward ANN model perform against the pure ARIMA model in forecasting Kenya′s GDP ?
2. Which parameter specifications of the constituent models produce more accurate GDP forecasts?
3. By what percentage does the hybrid model outperform the pure ARIMA model in short-term forecasting of GDP?
4. How does the hybrid model′ future forecast estimates vary from forecasts generated by the International Monetary Fund?

1.5 Significance of the study

The eradication of poverty is the first United Nations′ Sustainable Development Goal whose key indicator is economic growth as measured by GDP. The least developed countries $LDCs$ are expected to reach the Sustainable Development Goal target for GDP growth of at least 7 per cent by 2030 (United Nations, 2018). Therefore, the monitoring, tracking and predictive estimations of the GDP growth statistic amongst these countries is of great significance as it determines the amount of finances to be channelled towards productive investment by governments to spur economic growth. Further, reliable forecasts are used by governments and market participants for long term planning (Steinbuks, 2019).

The accurate forecasting of GDP is fundamental in monitoring the essential Debt to GDP ratio of any country. Ratios of external and public debt to GDP remain at the core of the joint IMF-World Bank methodology for conducting standardized debt-sustainability analyses in low-income countries, where a continually rising ratio above certain thresholds is a signal that debt is unsustainable (Porzecanski, 2018).

Causality relationships between GDP and other aspects of development such as energy consumption, financial markets, unemployment and CO_2 emissions exist as studied by different authors for example (Guptha,2018). Accurate GDP predictions therefore can be used to monitor the direction of movement of such counter-party variables upon establishing the exact relationship. Such uses of this key economic variable point to the essence and need for precise forecasting of GDP growth based on relevant and consistent assumptions.

Chapter 2

Literature Review

2.1 Introduction

A variety of models have been used in forecasting studies such as Time Series models, Econometric models, Artificial Neural Networks and Hybrid Models.This Literature analyses and reviews the model′ capacity to model and in turn forecast a mixed data set such as GDP time series as developed and used by different authors over the years.

2.2 Time series Models

A time series is a sequence of quantitative observations at successive times (Liu et. al., 2016). Time series forecasting is an important area of forecasting in which past observations of the same variable are collected and analyzed to develop a model describing the underlying relationship (Zhang, 2003). Time series models are often used in forecasting when limited knowledge is available on the underlying data generating process or when there is no satisfactory explanatory model that relates the prediction variable to other explanatory variables. The primary objective of time series analysis is to develop a mathematical model that can forecast future observations on the basis of available data (Khandelwal,2015).

A popular type of the linear time series model is the ARIMA model valued for the Box-Jenkins Methodology that is part of the model building process (Brockwell, 2016). The Methodology entails a three-step iterative process of model identification, parameter estimation and diagnostic checking (Box and Jenkins 1968).

ARIMA models are quite flexible in that they can represent several different types of time series, i.e., pure autoregressive (AR), pure moving average (MA) and combined AR and MA (ARMA) series. However, their major limitation is the pre-assumed linear form of the model (Zhang, 2003). Moreover, the Box–Jenkins methodology focuses on the low order autocorrelation, a model is considered adequate if low order autocorrelations are not significant even though significant autocorrelations of higher order still exist this therefore results in suboptimal models (Granger 1989). This disadvantage may however benefit a hybrid model since Granger (1989) further points out that for a hybrid model to produce superior forecasts the constituent models should be suboptimal.Wabomba (20160 establishes the ARIMA $(2, 2, 2)$ model as the best for modelling Kenyan GDP under only one criterion the AIC with relatively adequate in sample forecast results within 5

SARIMA $(p, d, q)(P, D, Q)m$ is an extension to the ARIMA model and is a popular linear model for forecasting seasonal time series. It adds three new hyper-parameters to specify the auto regression (AR), differencing (I) and moving average (MA) for the seasonal component of the series, as well as an additional parameter m which represents the number of time steps for a single seasonal period (10). For data sets that are not explicitly seasonal, the choice of whether to use SARIMA or ARIMA depends on the observations made from the visualization plots of the ACF and the PACF. For example the observation of a slow residual decay in the ACF plots despite having computed the first order of differencing may suggest the application of a seasonal difference.

ARIMAX Autoregressive Integrated Moving Average with Exogenous Inputs model is a model that can take the impact of covariates on the forecasting into account, improving the comprehensiveness and accuracy of prediction. The methodology employed is similar to that used in ARIMA modeling except for the need to estimate the cross correlation coefficient between the response series and the input series to determine the configuration of the ARIMAX model Yang et al(2018). ARIMAX models have the capacity to identify the underlying patterns in time-series data and to quantify the impact of environmental influences such that the influences of high-impact changes of both internal and external nature can be identified (Andrews,2013). Peter and Silvia (2012) apply the ARIMAX model and ARIMA model to quarterly GDP data and conclude that the ARIMA model outperforms ARIMAX in forecasting out of sample.

Non-linear time series models are used to overcome the restrictive linear assumption posed by ARIMA models when modelling mixed data. Non-linear time series models such as GARCH models however, are still limited with respect to capturing non-linearity because these models are developed for specific non-linear patterns; they are not capable of modeling other types of non-linearity in the time series (Zhang, 2003).

2.3 Econometric Models

Statistical (also called econometric) approaches forecast the current price by a weighted combination of the past prices and/or past or current values of exogenous variables (e.g., demand or weather forecasts), typically in a linear regression setting. Autoregressive terms are often used to account for the dependencies between today's prices and those of the previous days (Soytacs, 2019). The Central Bank of Kenya uses a relatively complex macroeconomic model to forecast economic variables such as GDP as informed by economic theory (Were et al, 2013).The economic theory basis for this model targets policy analysis and for evaluating the impact of various shocks as it shows the inter-linkages in the economy.

2.4 Artificial Neural Networks Models

Introduction

The three basic components of the neural network model are: a set of connecting links, an activation function and bias. The Neural Network can comprise of a single layered or multi layered network of neurons that is formed when a neuron links with the other neurons through a connection link (Darji, 2015).A neuron is a data processing unit. An ANN contains 3 layers: an input layer, an output layer, and one or more hidden layers. The hidden layer is useful for performing intermediary computations before mapping the input to the output layer.

The feed-forward multi-layer network in Figure 2 consists of an input layer, a hidden layer, each with an activation function, and an output layer that often uses a linear transfer function to avoid distortion of the predicted output. The ANN model performs a non-linear functional mapping from the past observations to the future value y_t i.e

$$y_t = f(y_{t-1}, y_{t-2}, \dots y_{t-p}, w) + \varepsilon_t. \tag{2.1}$$

where, w is a vector of all parameters and $f(.)$ is a function determined by the network structure and connection weights.

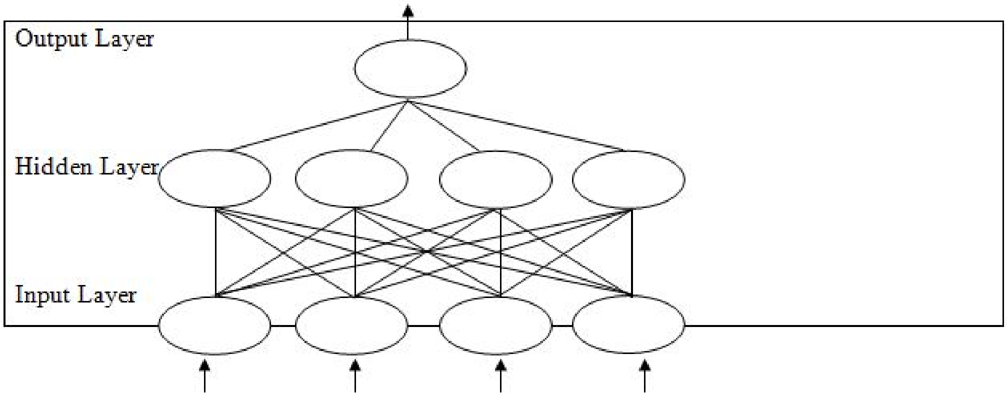

Figure 2.1: A typical Feed-forward Neural Network (Zhang, 2003)

The main advantage of artificial neural networks is their ability to represent non-linearity that may exist between input and output variables. Neural networks require little knowledge with respect to the nature of a time series therefore they have a capability of extracting patterns that are too complex to be noticed by humans (Taskaya and Ahmad, 2005). Furthermore, ANN models have an advantage over other classes of non-linear models in that they are universal approximators; almost non-parametric which can approximate a large class of functions with a high degree of accuracy (Zhang, 2003).

Multi-layered networks are capable of memorizing data due to the large number of synaptic weights available in the network. The hidden nodes of the hidden layer allow neural networks to capture the pattern in the data, and perform non-linear mapping between input and output variables (Darji, 2015). However, according to Darji (2015), rainfall the problem of over-fitting occurs due to model over fitting on the training data or the selection of a large number of input parameters (dimensions) as a result of many neurons or many hidden layers rendering the network unreliable when dealing with forecasting problems. However, the problem of over fitting can be reduced by using early stopping or regularization methods.Early stopping is the process of modifying a learning algorithm so as to prevent over fitting. This generally involves imposing some sort of smoothness constraint on the learned model (Girosi, Jones and Poggio, 1995).

Neural Network Architecture and Training

Different architectural structures of ANNs can be used in forecasting depending on different parameters such as size of data, training memory, delay, required number of connections, and learning speed. Examples of structures include; Multilayer Feed Forward Neural Network (MLFN), Recurrent Neural Network (RNN) and Time Delay Neural Network (TDNN). RNN allows information to pass backward using a set of feedback connections.

The training of a neural network involves an iterative training until the change in weights in a training cycle reaches a minimum value. This iterative training involves a set of input and corresponding output pairs for which the network studies the underlying mapping function by adjusting the free parameters of weights and biases to learn the input-output relationship described by the data set.

After training, a model is validated by checking whether it produces accurate output or not (Darji, 2015). The performance measure of a prediction model as traditionally used is based on the error difference between predicted and actual values. To minimize this error the synaptic weight in the training algorithm used can be updated by generating the predicted output as close to the actual output. To update the synaptic weights, different ANN back-propagation training algorithms such as conjugate gradient descent and Levenberg-Marquardt are used (Darji, 2015. The back-propagation (BP) learning algorithm has been shown to be the most popular and widely implemented in all neural networks paradigms (Wang, 2013).

The chosen training algorithm depends on the problem being solved or studied since there is no algorithm available which guarantees the global optimal solution for non-linear optimization problems. Different activation functions which provide the non-linear relationship between inputs and outputs can be applied to the neurons in the hidden and output layers. The four most commonly used activation functions by practitioners are; sigmoid, tanh, sine or cosine and linear (Zhang, 2003). Examples of the logistic and hyperbolic activation functions are shown below:

$$Sig(x) = \frac{1}{1 + exp(-x)}. \tag{2.2}$$

$$Tanh(x) = \frac{1 - exp(-2x)}{1 + exp(-2x)}. \tag{2.3}$$

A key task in ANN modeling for a time series is choosing an appropriate number of layers and of hidden nodes q, as well as selecting the dimension of input vector (the lagged observations), p. However, there exists some difficulty in determining q and p in practice as there are no theoretical developments to guide the selection process. Hence, in practice, experiments or trial and error methods are often conducted to select the appropriate values p and q (Wang,2013).This major gap limits the use of ANNs however this paper suggests the use of a hybrid model whose first stage of using an ARIMA linear model entails determining the Autoregressive structure of the data which can then be used to inform the ANN′s dimension of input vector p. The parameter selection issue is often further compounded by the existence of relatively small samples of the low frequency annual GDP data which limits the use of the Network that is heavily data dependent,this can be overcome by interpolation to generate past higher frequency data (quarterly) from the lower frequency data available(annual)through temporal disaggregation (Hodgess, 2019).

Dumitru (2013) outlines some challenges in the use of Artificial Neural Networks.The study points out that the mathematical theories used to guarantee the performance of an applied neural network are still under development and that ANNs are useful for prediction, but not always in understanding a model.To overcome these gaps the author suggests the continuous training and thorough testing of its results over and above combining multiple models.

Forecast Accuracy Measures

Diebold and Mariano (1995) formulated a statistic that tests the hypothesis of equal predictive ability in two competing forecasts that have been generated by different forecasting techniques. The use of such measures still remains largely disputable in the forecasting world.Other forecast accuracy measures such as the Root Mean Squared Forecast Error are outlined in the methodology section.

Armstrong and Callopy (1992) mention that the primary criteria for measuring forecast errors are reliability, construct validity, protection against outliers, and the relationship to decision making.The two most common point forecast accuracy evaluation metrics are the squared error loss and the absolute error loss such as the Root Mean Squared Error loss and the Mean Absolute Error, respectively.

2.5 Hybrid Models

Different studies have shown that ANNs can outperform other numerical and statistical models in forecasting albeit pegged on the particular nature of problem being solved and the specific forecast horizon of interest. In their survey of rainfall forecasting, Darji (2015) concludes that ANNs are suitable to predict yearly rainfall than other forecasting techniques. Consequently different literature and studies have been inconclusive about the superiority of ANN models over other models in forecasting. To resolve this some authors have recommended the use of Hybrid models prompted majorly by the fact that it is often difficult in practice to determine whether a time series under study is generated from a linear or non-linear underlying process (Zhang, 2003). Further, according to Zhang (2003) as previously mentioned real-world time series are rarely purely linear or non-linear and often contain both linear and non-linear patterns.

Zhang (2003) also argues that the method of model selection where a number of different models are tried and the one with the most accurate result is selected, is not necessarily the best for future use due to many potential influencing factors such as; sampling variation, model uncertainty and structure change. He points out that by combining different methods, the problem of model selection can be eased. Many empirical studies suggest that by combining several different models, forecasting accuracy can often be improved over the individual model. A number of combining schemes have been proposed for time series forecasting problems resulting in the so called hybrid models.

The linear ARIMA and the non-linear multilayer perceptrons have been jointly used in such hybrid models in order to capture different forms of relationship in the time series data (Khashei and Bijari, 2012) . Different studies have outlined these hybrid models as developed and used over time such as (Khashei, 2012) and (Wang, 2013). An important finding that emerges is that hybrid models are robust with regard to the possible structure change in the data (Zhang 2003).The essence of the model combination in forecasting is to use each model′s unique features to capture different patterns in the data.

Three distinctive variations of the hybrid ANN-ARIMA model exist in Literature.The models apply an ARIMA model to a given time series data set, then consider the error between the original and the ARIMA-predicted data as a non-linear component, and model it using an ANN in different ways.

Zhang (2003) proposes an additive ARIMA-ANN model and applies it to three data sets the Wolf′s sunspot data, the Canadian lynx data, and the British pound to US dollar exchange rate data. The study assumes that a time series is composed of a linear autocorrelation structure and a non-linear component which have to be extracted from the data. The study further proposes that the residuals of the ARIMA linear model will contain information about the non-linearity such that the modeling of these residuals using an ANN is justifiable. Further, the form of the hybrid model is assumed to be additive such that the final forecast is obtained by adding the ARIMA model′s forecast of the linear component to the ANN model′s forecast of the non-linear component.

$$y_t = L_t + N_t. \tag{2.4}$$

where L_t denotes the linear component and N_t denotes the non-linear component of the time series y_t .

A key critique of this study′s assumptions is that in a modular ARIMA neural network hybrid architecture, one model is always built on the residuals of the other model but if the latter model fails to model the errors of the prior then the performance of the overall hybrid model can be degenerated compared to its constituents′ performances (Taskaya and Ahmad,2005).

A further downside Zhang's additive hybrid model is that the model supposes that the relationship between the linear and non-linear components is additive and this may underestimate the relationship between the components and degrade performance, if there is not any additive association between the linear and non-linear elements and the relationship is different (Taskaya and Ahmad,2005).

The Khashei (2012) study proposes a generalized hybrid ARIMA-ANN Model and apply it to the same data sets; Wolf's sunspot data, the Canadian lynx data, and the British pound against the United States dollar exchange rate data. This model does not have the above-mentioned additive assumption of the traditional hybrid ARIMA and ANN model however a time series is still considered as an unspecified function of a linear and a non-linear component.

$$y_t = f(L_t, N_t). \tag{2.5}$$

The first stage in this approach entails fitting an autoregressive integrated moving average (ARIMA) model similar to the traditional hybrid models to model the linear component. However, the residuals from the first stage are assumed to contain both the non-linear relationship that the linear model is not able to model and may also contain a residual linear relationship Khashei and Bijari (2012). In the second phase a multilayer perceptron (ANN) is used in order to simultaneously model the non-linear and probable linear relationships that may have remained in the residuals after fitting ARIMA and also the non-linear and linear relationships in the original data. Such that the final non-linear function determined by the neural network is based on the original data, the ARIMA forecasts as well as the ARIMA residuals.

Wang et al (2013) propose a multiplicative ARIMA-ANN model and test it on two similar data sets as the previous authors and one new data set respectively; Wolf′s sunspot data, the Canadian lynx data and the IBM stock price data. With similar assumptions as Zhang (2003), the multiplicative model also assumes that the time series has both linear and non-linear components, where the first phase of modeling entails fitting the ARIMA model and obtaining the pre-assumed non-linear residuals which are then modeled using the ANN model. In this study though, the final forecasts are obtained by multiplying the forecasts from ARIMA with the forecasts from the ANN.

$$y_t = L_t^* N_t. \tag{2.6}$$

Wang et al (2013) conclude that the multiplicative model is generally better than Zhang′s additive model with respect to forecasting the 3 real-world data sets.The multiplicative ARIMA-ANN is a variation of Zhang′s additive ANN-ARIMA model for forecasting and the conclusion drawn on its use is that the new combinatorial model is effective in obtaining more accurate forecasting as compared to existing models Wang(2013).

Prior to using hybrid models some authors have applied some methods to the data to define its structure to justify and inform further the use of the hybrid model.Two such studies apply the Discrete Wavelet Transform and the moving average filter prior to fitting the hybrid model. Babu and Reddy (2014) argue that if the nature of the given time series is taken into account before applying the models then more accurate forecasts can be obtained. They use the moving-average filter and obtain higher prediction accuracy. Discrete Wavelet Transform can be adopted to filter the data into the approximation and the detail parts before applying the Hybrid model, this way the ARIMA and the ANN would have more potential to capture the linear and non-linear patterns in the data effectively (Pannakkong, 2017). Discrete Wavelet Transform is a component of the wider spectral analysis which is a purely descriptive analysis that provides a rigorous and versatile way to define formally and quantitatively cyclical series components and, by means of filtering, it provides a reliable extraction method (Iacobucci, 2005).

A slight variation of these 3 models has been proposed by (Khandelwal, 2015) who used a discrete wavelet transform to decompose four sets of real world data into high(detailed) and low(approximate) frequency components which respectively pick up the higher and lower frequency components of the series. This was then followed by a reconstruction procedure with an inverse discrete wavelet transform after which an ARIMA Model was fitted to the reconstructed detailed part and forecasts generated. The detailed part is thus assumed to contain only the linear elements of the data. An ANN was then fitted to the corresponding residuals together with the approximate part. Finally, the combined forecasts were obtained through adding the two component-wise forecasts. They concluded that the proposed method yielded notably better forecasts than ARIMA, ANN, and Zhang′s hybrid model. The discrete wavelet transform is described as;

$$\begin{aligned} \mathrm{Z}_t &= A_j(t) + \sum_{j=1}^{J}(D_j(t)) && (2.7) \\ &= \sum_{k=1}^{K} C_{j,k}\Phi_{j,k}(t) + \sum_{k=1}^{K}\sum_{j=1}^{J} D_{j,k}\psi_{j,k}(t). && (2.9) \end{aligned}$$

where Z_t denotes the time series at period t,$A_J(t)$ denotes the approximation of the highest decomposition level (J), $D_j(t)$ denotes the detail of decomposition level j,$C_{j,k}$ and $D_{j,k}$ denote the coefficient of approximation and detail respectively, at decomposition level j and period k,$\Phi_{j,k}(t)$ and $\psi_{j,k}(t)$ denote high(detail) and low(approximation) pass filters respectively at decomposition level j and period k respectively, K denotes the total number of time series while J denotes the total levels of decomposition.

Pannakkong (2019) also applies only one level of the DWT decomposition to Zhang's hybrid model by using Daubechies wavelet basis function on Thailand's cassava export data but unlike Khandelwal et al (2015) the approximation part and detailed parts of the decomposition are considered to each contain both linear and non-linear elements resulting in two sets of residuals; the residuals of the approximation and residuals of the detailed. The conclusion drawn is that better forecasting accuracy results if the DWT, ARIMA and ANN are combined in extracting linear and non-linear components of the approximation and the detail without a prior linear or non-linear assumption against seven other versions of the hybrid model on three distinct cassava types data sets.

2.6 Conclusion

The aforementioned forecasting models build up from the linear and non-linear time series models to the Econometric and ANN Models reveals a number of gaps and challenges with regard to accurately forecasting mixed data series.Linear Model′ assumptions are too restrictive, non-linear models suit only specific pre-determined non-linear data patterns, Econometric models tend to be too complex and computationally expensive for forecasting but best suits policy analysis studies and finally ANNs lack a firm theory basis to explain its functionality and computation process. The Hybrid models however have emerged as a remedy to the limitations posed by these individual models due to their accommodation of mixed linear and non-linear data series.

Moreover, these models have been applied to different economic and non-economic data from developed countries and to the best of this study′s analysis and research, forecasting studies based on developing countries′ data which has unique characteristics, is yet to be explored fully.

Therefore with both theoretical and empirical findings suggesting that combining different methods can be an effective and efficient way to improve mixed data forecasts such as Zhang (2003), this study proposes the comparison of GDP forecasts using ARIMA then combining ARIMA and ANN models to form a hybrid model based on Zhang's additive methodology mentioned in the latter part of this Literature review, which can then be assessed against the individual model′s results.

Chapter 3

Methodology

3.1 Introduction

In light of the conclusions from the literature review, it is of essence to note that no one size model fits all such that different models seem to work best for different data sets.

3.2 Reseach Design

The study has an experimental research design as the objective of the study is to forecast short term GDP of 3 years.

3.3 Data Collection

Data used is secondary. Data is obtained from The Kenya National Bureau of Statistics and The World Bank Meta data.Annual GDP data from 1960 to 2018 is obtained from the International Monetary Fund database, due to the need for uniformly and consistently measured data, quarterly data that could have been used to meet the bulky data demands of Neural Networks is inconsistent due to the re-basing of all quarterly data from 2009 onwards such that quarterly data before 2009 has a different basis from quarterly data after 2009. Further, given the highly sensitive nature of the forecast values, interpolated data from annual values to quarterly values would likely present further deviations and errors.

3.3.1 Selection of the Period of Study

The period of study that will be used is based on data availability.

3.4 Data Analysis

Time series data of GDP consists of successively generated observations over time such that data is ordered with respect to time and successive observations may be dependent. The underlying process to the data may indicate long-term behaviour such as trend and or cyclical fluctuations that depict seasonality. Using different models to capture such underlying processes can enable forecasts of the likely observation at a time point in future to be made.Based on the literature reviewed, the preferred classes of models that we will explore are ARIMA and the hybrid ARIMA-ANN Model based on Zhang′s specification.

ARIMA Model

ARIMA models do not involve independent variables in their construction, but rather make use of the information in the series itself to generate forecasts thus they rely heavily on autocorrelation patterns in the data (Wabomba, 2016). This model assumes that time t is a discrete variable, y_t denotes the observation at time t, the weights $\theta_i \in$ and $\alpha_i \in$ and $_t$ denotes the zero mean random noise at time t.

The Moving Average (MA) model is a time series model which uses past errors as explanatory variables. Let $\mu_t(1, 2, 3, \dots)$ be a white noise process, a sequence of independently and identically distributed (iid) random variables with $E(\mu_(t) = 0$ and $Var(\mu_(t)) = \sigma^2$. Then the q^th order MA model is given as:

$$y_t = \mu_t + \theta_1\mu_{t-1} + \theta_2\mu_{t-2} + \dots + \theta_q\mu_{t-q}. \tag{3.1}$$

The model is expressed in terms of past errors and thus we estimate the coefficients $\theta_j, j = 1, 2$ and then use the model for forecasting such that only q errors will affect the current level y_t but higher order errors do not affect it . According to (17), an autoregressive model of order p, an AR (p) can be expressed as;

$$y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \ldots + \alpha_p y_{t-p} + \mu_t. \tag{3.2}$$

The model is expressed in terms of past values and therefore, we estimate the coefficients and use the model for forecasting. In this case, all previous values will have cumulative effects on the current level.

The ARMA (p, q) model (short for autoregressive moving average) is a combination of AR (p) and MA(q) models and its key assumption is that time series data is stationary. This model assumes that y_t is generated through the process:

$$y_t = \sum_{i=1}^{q} \theta_i \mu_{t-i} + \sum_{i=1}^{p} \alpha_i y_{t-i} + \mu_t. \tag{3.3}$$

where $_t$ is a zero mean noise term.

Since time series data are usually not realizations of a stationary process as some of them may contain deterministic trends resulting in strong serial correlations the differential method is considered resulting in the ARIMA (p, d, q) model of which ARMA (p, q) is a special type where the differences order is zero (16). Non-stationary series of data then becomes well represented by ARIMA (p, d, q) models by allowing the differencing of the data where the parameters of the model are defined as; p the number of autoregressive terms, d the number of times of differencing and q the number of moving average terms. The parameters are ultimately substituted with integer values to quickly indicate the specific ARIMA model being used.

A preliminary Box-Jenkins analysis with a run sequence plot of the initial GDP data will be run as the starting point in determining an appropriate model. This will entail the 4 iterative stages of identification, estimation, diagnostic checking and forecasting. The conceptual framework of the Box Jenkins Methodology is as in Figure 3.

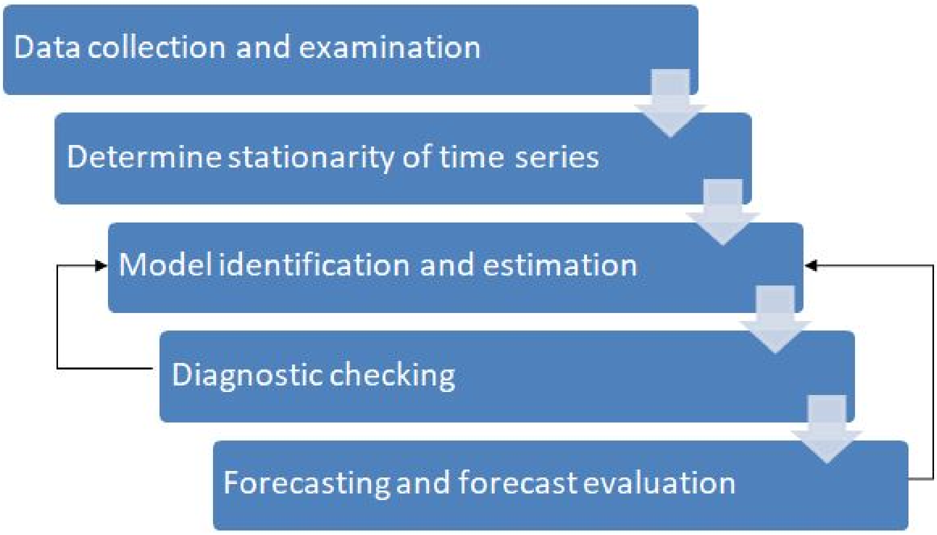

Figure 3.1: ARIMA forecasting procedure using Box-Jenkins Methodology

Preliminary procedures will include adjusting the data to form a stationary series followed by identifying seasonality and conducting seasonal differencing if necessary. Autocorrelation and partial autocorrelation functions of the time series can then be used to determine which Autoregressive and or Moving Average component if any will be used in the model, after fitting the automatic ARIMA as a starting point. Partial Autocorrelation function measures the correlation between an observation k periods ago and the current observation, after controlling for observations at intermediate lags (all lags<k) and is useful for telling the maximum order of the AR. The Auto correlation function plot is useful for telling the order of the Moving Average process. We can use maximum likelihood, the method of least squares, or Yule-Walker equations for the parameters $(c, ,)$ estimation. This stage of model identification and estimation will also involve the use of the Akaike Information Criterion (AIC) of selecting a model from a set of models. The chosen model is the one which has the lowest AIC.

The diagnostic checking involves the analysis of the residuals where we plot the standardized residuals, the autocorrelation function of the residuals, and the p-values for Ljung-Box Q statistic. At this stage, the assumptions of the ARIMA model are checked, e.g. the hypothesis of errors being independently and normally distributed (Kokkinen, 2016).

The forecasting stage will entail using the estimated model to generate forecasts and their confidence limits. According to Kokkinen (2016) the forecasting process expression can be found recursively and is given by:

$$\hat{y}_{t+1} = c + \alpha_1 y_t + \alpha_2 y_{t-1} + \ldots + \alpha_p y_{t-p} + \theta_1 \hat{\mu}_t + \theta_2 \hat{\mu}_{t-1} + \ldots + \theta_q \hat{\mu}_{t-q}. \quad (3.4)$$

Hybrid Model

Since it is difficult to completely know the characteristics of the data in a real problem, the hybrid methodology that has both linear and non-linear modeling capabilities can be a good strategy for practical use. By combining different models, different aspects of the underlying patterns may be captured.

We will use the hybrid model as developed by Zhang (2003), this hybrid approach applies ARIMA and ANN separately for modeling linear and non-linear components of a time series. We have the general equation represented as;

$$y_t = L_t + N_t. \quad (3.5)$$

where y_t is the observation at time t, L_t denotes the linear component and N_t the non-linear component.

ARIMA is fitted to the linear component and the corresponding forecast $\hat{L}_t$ at time t is obtained. The residual at time t is given by $e_t = y_t - \hat{L}_t$ unlike the residual of the multiplicative model which is given by $e_t = y_t \hat{L}_t$. According to Zhang, the residuals dataset after fitting ARIMA contains only the non-linear component and so can be properly modeled through an ANN. We will use p input nodes based on a starting point informed by the autoregressive parameter p obtained in the ARIMA fitting.The ANN for residuals has the following form;

$$e_t = f(e_{t-1}, e_{t-2}, \ldots e_{t-p}) + \varepsilon_t. \quad (3.6)$$

where f is a non-linear function, estimated by the ANN and ε_t is the white noise. A single hidden layer feed forward ANN with one output node is most commonly used in forecasting applications (Wang, 2013). A $p \times 1$ ANN has the following output;

$$y_t = \varphi_0 + \sum_{j=1}^{q} \theta_j g\left(\theta_{0j} + \sum_{i=1}^{p} \theta_{ij} y_{t=1}\right) + \varepsilon_t. \quad (3.7)$$

where $\varphi_j(j = 0, 1, 2, \ldots, q)$, $\theta_i(i = 0, 1, 2, \ldots, q)$ are the weights $\varphi_(0)\theta_0 j$ are the bias terms, g is the activation function, y_t is the time series, the parameters are p the number of input nodes as informed by the ARIMA autoregressive terms and q the number of hidden nodes and
$varepsilon_t$ is the white noise.The model will use a rolling window such that if p is the number of predictor values in this case the residuals of the predicted ARIMA, then the first ANN predicted value occurs at time $p + 1$. We will take the logistic function as is often used as the hidden layer activation function g.

$$g(x) = \frac{1}{1 + exp(-x)}. \tag{3.8}$$

Based on the additive model if $\hat{N}_t$ is the forecast of this ANN, then the ultimate hybrid forecast at time t is obtained as;

$$\hat{y}_t = \hat{L}_t + \hat{N}_t. \tag{3.9}$$

The combined forecasts are obtained through adding these two component-wise forecasts. The approach is elaborately presented in the Algorithm in Figure 3. Comparatively, the multiplicative model as outlined in the literature review proposes that if $\hat{N}_t$ is the forecast of this ANN, then the ultimate hybrid forecast at time t is obtained as;

$$\hat{y}_t = \hat{L}_t^* \hat{N}_t. \tag{3.10}$$

3.5 Forecasting and Forecast Evaluation

The ARIMA model as well as the Hybrid model will each be individually used to obtain short term forecasts of GDP after splitting the data into the training set and test set. The data split into training and testing will be performed using the 80 : 20 split of the 59 data points available, as this ratio values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance (James et al., 2013).According to Tashman (2000), the most important guide in deciding the number of periods to withhold as test data N is the longest term H of the forecast, in this case H is 3 where N must be at least as large as H, which is met by the 80 : 20 data split. Further, as much as the k-fold cross validation method has been shown in many studies and Literature to form the best basis for data splitting, this study is time series based such that rather than using k-fold cross-validation, we utilize hold-out cross-validation where a subset of the data that is temporally split is reserved for validating the model performance because the validation set must come chronologically after the training subset.The modeling will be done using the software R Studio.

Choosing a forecasting model from a set of viable candidates should be based on in-sample and out-of-sample performance since some models may perform fairly well in-sample, but not necessarily out-of-sample (6). Thus in this study the quality of the two forecasting models will be judged based on the out-of sample forecasting accuracy .

The penalty received for an error $(e_(t+h,t) = y_(t+h,t) - y_(t+h))$ depends on the selected loss function $L(\hat{y}_{t+h,t}, y_{t+h})$. The individual models$'$ prediction accuracy will therefore be assessed using the commonly used the Root Mean Squared Error and Theil's Inequality Co-efficient (U) to rank their prediction performance. Theil$'$s U is such that it will always lie between 0 and 1. The closer U is to 0, the better the model's forecasting power and the statistic reaches its lower boundary of 0 at perfect forecasts.According to Bliemel (1973) if the co-efficient is larger than 1 then the forecasting method used is to be rejected because it cannot beat the most simple no-change extrapolation.

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T} e_{t+h,t}^2}. \tag{3.11}$$

$$U = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{y}_{t+1,t}^2 - y_{t+1})}}{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{y}_{t+1,t})^2} + \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_{t+1})^2}}. \tag{3.12}$$

Graphical inspection will also be used where plots of different forecasts against the true realization will be drawn to observe and assess the model$'$s comparative prediction performance against the already observed out of sample GDP values.

Chapter 4

Results

4.1 Exploratory Analysis

4.1.1 Data

The data used as previously mentioned is the annual raw GDP values from the World Development Indicators database in current United States dollars from 1960 to 2018.The time series data has 59 observations. A run sequence plot of the same as in Figure 4.1 reveals that the annual Kenyan GDP data has a long-term increasing trend from 1960 to 2018.

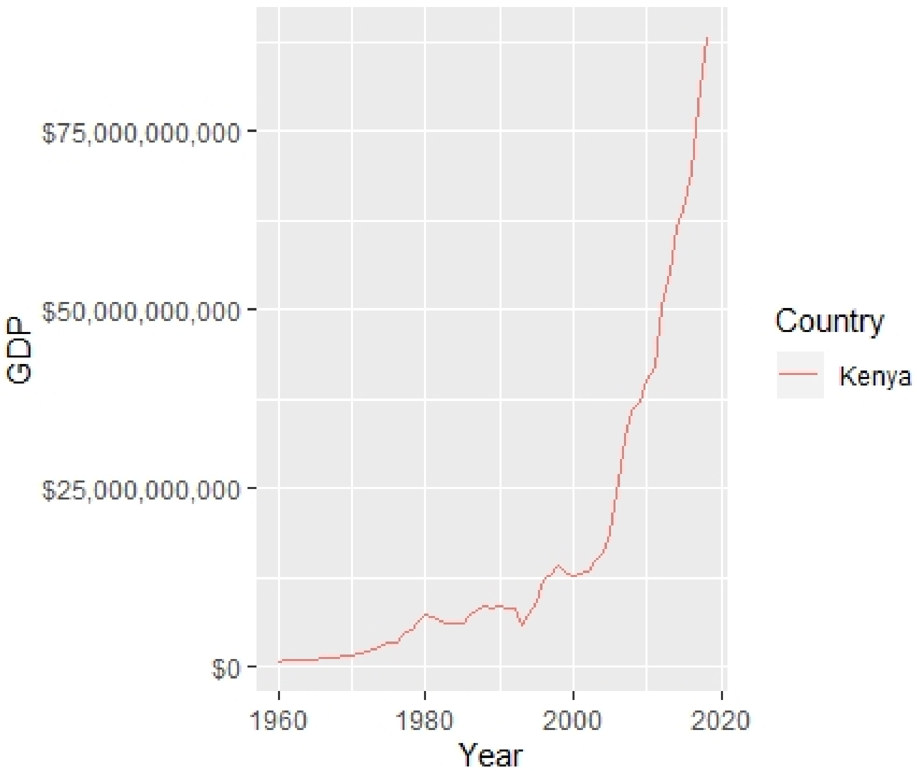

Figure 4.1: Kenyan GDP Time series Plot

4.1.2 Data Transformation

Given that the ARIMA model is the first model in the hybrid structure we require stationary data as stipulated by its methodology.Stationary data has a constant mean and variance (Wabomba, 2016).A visual inspection of the time series plot reveals an increasing trend as exhibited in Figure 4.1 implying that both mean and variance are not constant.The Autocorrelation plot in Figure 4.2 of the raw GDP time series exhibits a slowly decaying sequence echoing the non-stationarity of the series. Further, the ADF test with the null hypothesis of non-stationarity and the alternative hypothesis of a stationary time series yields a p-value of 0.99 which is greater than 0.05 at a 95% confidence interval, presenting strong evidence against rejecting the null hypothesis of a non-stationary time series.

The data is transformed using the logarithmic transformation and by further obtaining the second differences of the natural logarithm of the series, as used by other authors such as Wabomba (2016).The transformed series exhibits stationarity evidenced by the ADF test that yields a p-value of 0.01 which is less than the threshold of 0.05 thus there is sufficient evidence to conclude that the transformed times series is stationary.The transformed data run sequence plot exhibits stationarity as shown in Figure 4.3.

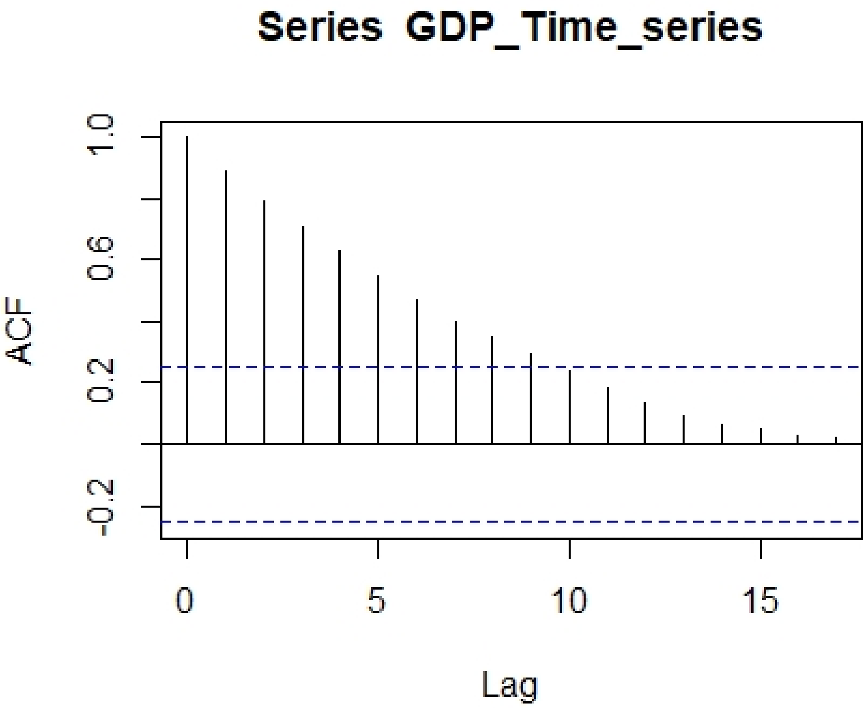

Figure 4.2: Kenyan GDP ACF Plot

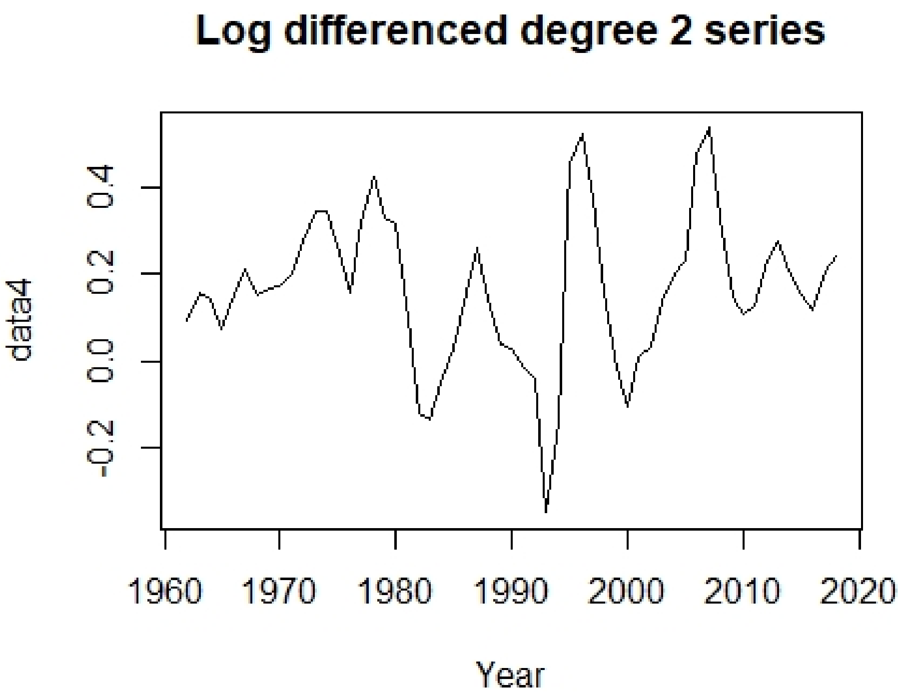

Figure 4.3: Transformed Data Run Sequence Plot

4.2 ARIMA Model Estimation

The Autocorrelation plot and the Partial autocorrelation plot of the transformed series exhibits significance at lags $(1, 4)$ and 2 respectively as in Figure 4.4. This informs the choice of the Autoregressive order and Moving Average order respectively of the model.

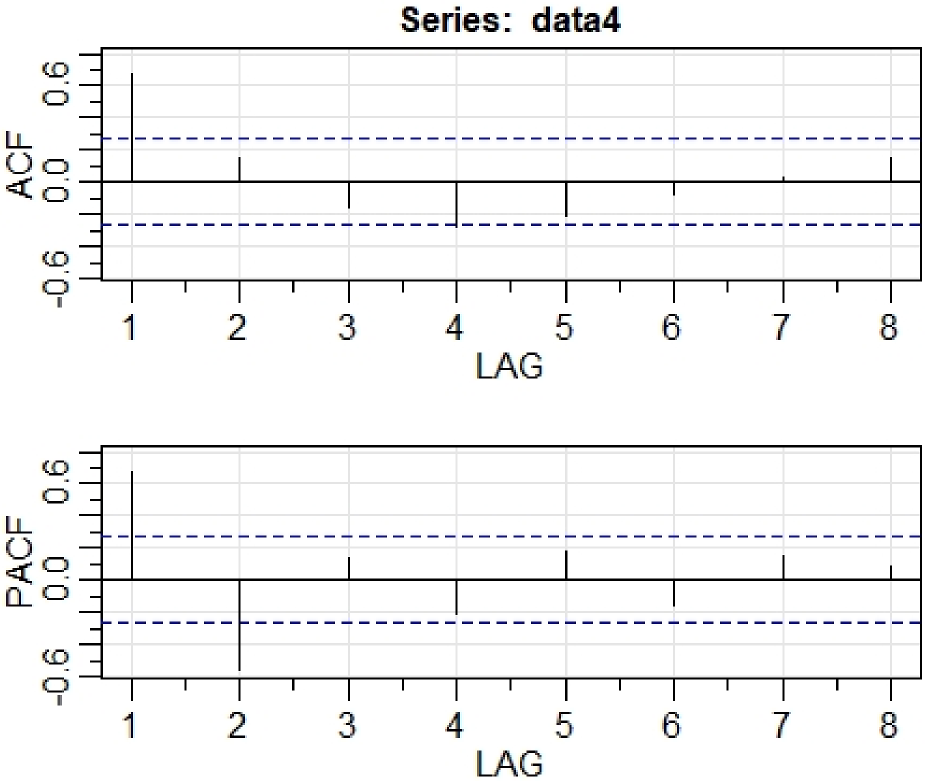

Figure 4.4: Transformed Data ACF and PACF Plot

The data is split into a training set and a validation set in the ratio 80 : 20.As mentioned in the Literature Review this basis is chosen due to the three main reasons; the temporal dependency nature of time series that requires the validation set to chronologically come after the training set, the proven results of this ratio having neither excessively high bias nor very high variance and the satisfaction of the condition N(the number of test data points must be atleast equal to H (the forecast horizon of the study).

The 3 best comparative fits of the ARIMA model for the transformed GDP training set data based on the lowest Akaike Information Criteria are ARIMA(2,2,1), ARIMA (2,2,2) and ARIMA(2,2,4). The ARIMA(2,2,2) model is found to be the best and most parsimonious model in sample among the aforementioned ARIMA models, based on four criteria the AIC,log-likelihood criteria, the Ljung Box statistic and the Augmented Dickey Fuller Unit Root Test of the models' residuals.The summary results are as in Table 4.1.

ARIMA(2,2,2) has the least AIC value as well as equally competitive loglikelihood value and RMSE.All three models have residuals consistent with the stationarity assumption at low value lags as displayed by their residual acf and pacf plots in in Appendix A.

The models' residuals are also analyzed using the Augmented Dickey Fuller Unit Root Test with the null and alternate hypothesis as:

H_0 Unit root present thus residuals are non-stationary
H_1 Unit root absent thus residuals are stationary

All three models have p-values less than 0.05 with the test statistic being lower than the critical value of -1.95 at a 5% significance level as summarized in Table 4.1, therefore the null hypothesis of non-stationarity is rejected.This result is consistent with the Box Jenkins Methodology that requires the residuals to be stationary.

Further residual diagnosis using the Ljung Box test results in p-values greater than 0.05 as in Table 4.1 thus we do not reject the null hypothesis of zero autocorrelations for errors, this indicates that the models have sufficiently captured linear autocorrelation patterns in the training set series.

However, on the validation data ARIMA(2,2,1) appears to outperform the other two models as it exhibits the least validation data set RMSE and the lowest Theil's U co-efficient which is closest to zero as summarized in Table 4.1. Bliemel (1973) mentions that as the Theil U co-efficient reaches the lower bound 0 then the forecasts can be deemed as perfect, the co-efficient is derived from the models' standard error.Low values of the RMSE indicate the preferred model as this metric estimates the standard deviation of a typical observed value from the fitted model's prediction. A large RMSE value generally means that the fitted model is failing to account for important features underlying the data. The ARIMA (2,2,1) model is therefore the best among the three for out of sample forecasts and will therefore its residuals will be used to analyze the non-linearities as prescribed by the additive Hybrid model methodology, since the objective of the study is to obtain a model that accurately forecasts short term GDP out of sample.

To specifically observe the behaviour of the chosen ARIMA (2,2,1) model for short term GDP forecasting, we choose a forecast horizon $h = 3$ and perform a rolling regression based forecast with a varying rolling window that starts from the number of data points in the training set $W = 45$ to $W = 54$ based on the last possible $N - h$ value which is $57 - 3 = 54$.This yields an RMSE of 0.0517 for the chosen best ARIMA model. Figure 4.5 shows the resulting rolling regression forecast results compared against the actual test data values. This provides a basis to later compare the model to

Table 4.1: ARIMA Parameter Estimates and Key Model Assessment Statistics

Statistic/Parameter	ARIMA(2,2,1)	ARIMA(2,2,2)	ARIMA(2,2,4)
AR(1)	0.3622	1.3012	.5466
AR(2)	−0.0106	−0.4081	−0.9642
MA(1)	1	0	−0.3265
MA(2)	0	−1	−0.7277
MA(3)	0	0	0.7494
MA(4)	0	0	0.1507
Intercept	0.1554	0.1469	0.1487
AIC	−61.94	−62.65	−61.19
Loglikelihood	35.97	37.32	38.59
Training series RMSE	0.1034	0.0985	0.0965
Ljung Box Test p-value	0.5074	0.7046	0.2401
ADF Test Statistic	−3.8782	−3.8517	−3.8043
Theil's U	0.2619	0.3067	0.5161
Validation set RMSE	0.0653	0.0765	0.1287
Validation set RMSE using rolling forecasts for h=3	0.0517	-	-

the IMF 3 years' ahead forecasts from 2019.

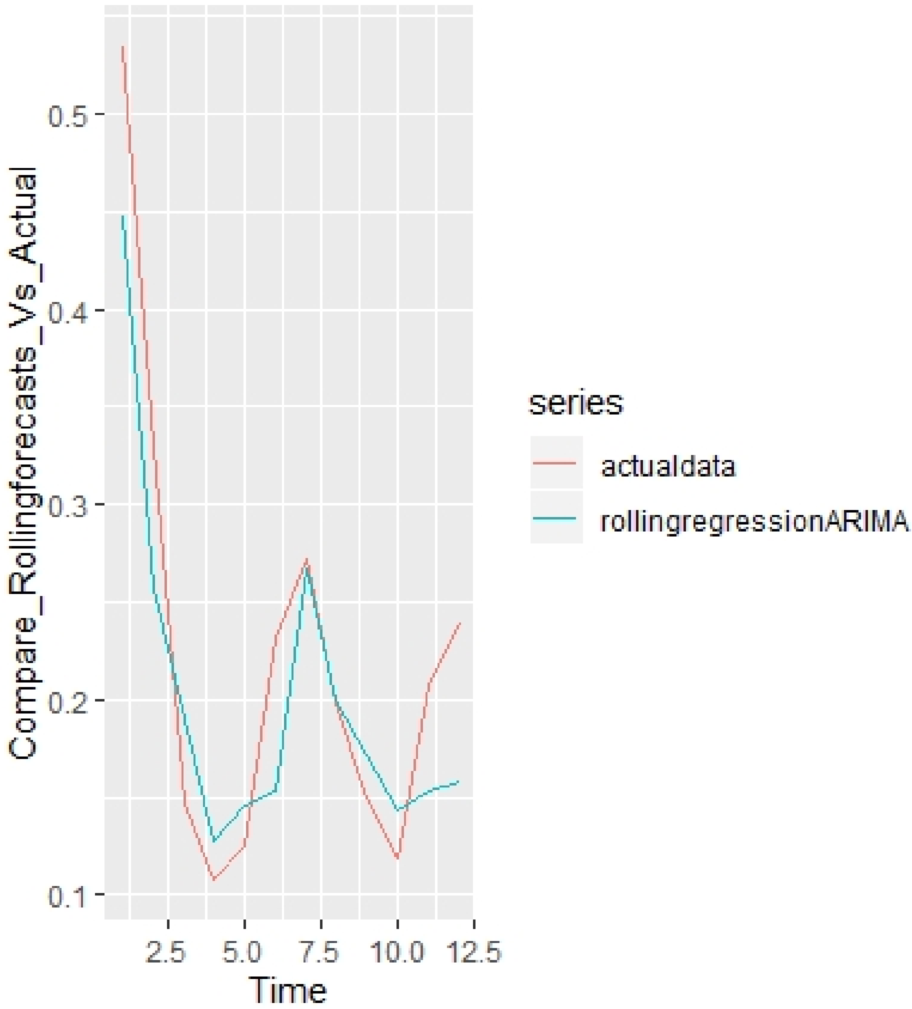

Figure 4.5: Rolling regression ARIMA forecast against actual data

4.3 Hybrid Model Fitting

The residuals from the ARIMA(2,2,1) model are fitted into the Artificial Neural Network model based on the Neural Network Autoregressive structure (NNAR) in the software R-studio.The model is defined by the parameters NNAR (p,k) where p is the level of the lagged input residual values while k is the number of nodes in the hidden layer and is trained based on a specified number of repeats, in this study we arbitrarily vary the repeats between 30 and 100. The NNAR model in the software R is designed to model only one hidden layer, according to Triebe, Laptev and Rajagopal (2019) a single hidden-layer network has been shown both theoretically and empirically to be capable of modelling any type of functional relationship.

For validation purposes the 45 residual are further split into a training and validation set in the ratio 80 : 20 resulting to 36 testing data points and 9 validation data points.The 45 residuals are those arising from the fitted ARIMA (2,2,1)Model whose 80 : 20 split of the 59 entries resulted in 47 training data points and 12 validation data points, however due to the Autoregressive lag 2, 2 data points are lost resulting in the fitting of 45 data points only and in turn 45 residuals.

Training of the model is done by varying the number of hidden nodes in the hidden layer, the number of repeated runs and the level of autoregressive lags of the input values, and observing the corresponding impact on the in sample RMSE, to obtain the best model.This method of varying the parameters and observing the resultant impact has also been used by Zhang and Berardi (2001) where they varied the input lag order from 1 to 5 and the number of hidden layer nodes from 2 to 8 to obtain the best parameters for their forecasting model, these low values of specifications are not too computationally expensive with regard to the run time in R-studio.

Rather than an arbitrary starting point,we assume that the linear autoregressive order 2 from the ARIMA model fitting can inform the non-linear autoregressive lag order of the residuals,therefore the neural network starts off with a lag to level 2 borrowing from the full data set linear autoregressive order deduced from the transformed data pacf plot that showed linear autoregressive significance spike at lag 2 as indicated in Figure 4.4. Triebe et al (2019) also designed an Autoregressive Net Neural Network model such that that the parameters of the input layer were equivalent to the Autoregressive coefficients of their ARIMA model.

Four models are fitted based on variations of the 3 aforementioned parameters: repeats, lagged input order and number of hidden nodes in the hidden layer, resulting in RMSE values as summarized in Table 4.2.

The RMSE values and visualization of the forecast plots is used to compare the forecast accuracy of the 4 fitted models.In the test set, the best model is the NNAR(5,2) since it has the least RMSE. However, in the training data the NNAR (2,30) results in the best model on the similar least RMSE basis. Visual assessment of the run sequence plots of the two model's test set forecast shows that NNAR(2,30) over-fits the data thus giving rise to the least value of RMSE in the training set of 0.00441 as shown in Table 4.3. Figure 4.6 also shows the two model's visualization plot.Therefore, the

Table 4.2: ANN Parameter Estimates and Key Model Assessment Statistics

Statistic/Parameter	**NNAR 1**	**NNAR 2**	**NNAR 3**	**NNAR 4**
p	2	5	2	2
k	2	2	30	2
Repeats	30	30	30	100
Training set RMSE	0.0717	0.0577	0.00441	0.0719
Test set RMSE	0.1000	0.1213	0.2617	0.1069
Test set RMSE using the rolling window	-	0.1483	-	-

NNAR (5,2)is preferred on an out of sample basis.Other plots with a varying number of repeats and hidden layers were also plotted as in Appendix B to assess their impact on forecasts,however, none of them beats the NNAR(5,2).

The plotted short term forecasts of the NNAR(5,2) appear reasonable against the actual test data set plot as in Figure 4.7.Therefore, the conclusion drawn is that the hybrid model based on the NNAR(5,2) with 5 lagged input variables and 2 hidden layers is the best among the 4 fitted models.

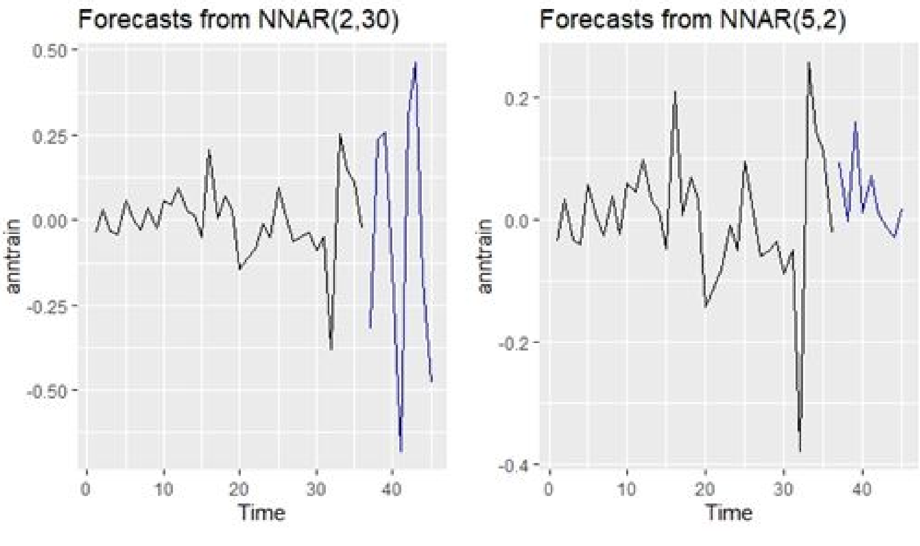

Figure 4.6: NNAR (5,2) forecasts against NNAR (2,30) forecasts

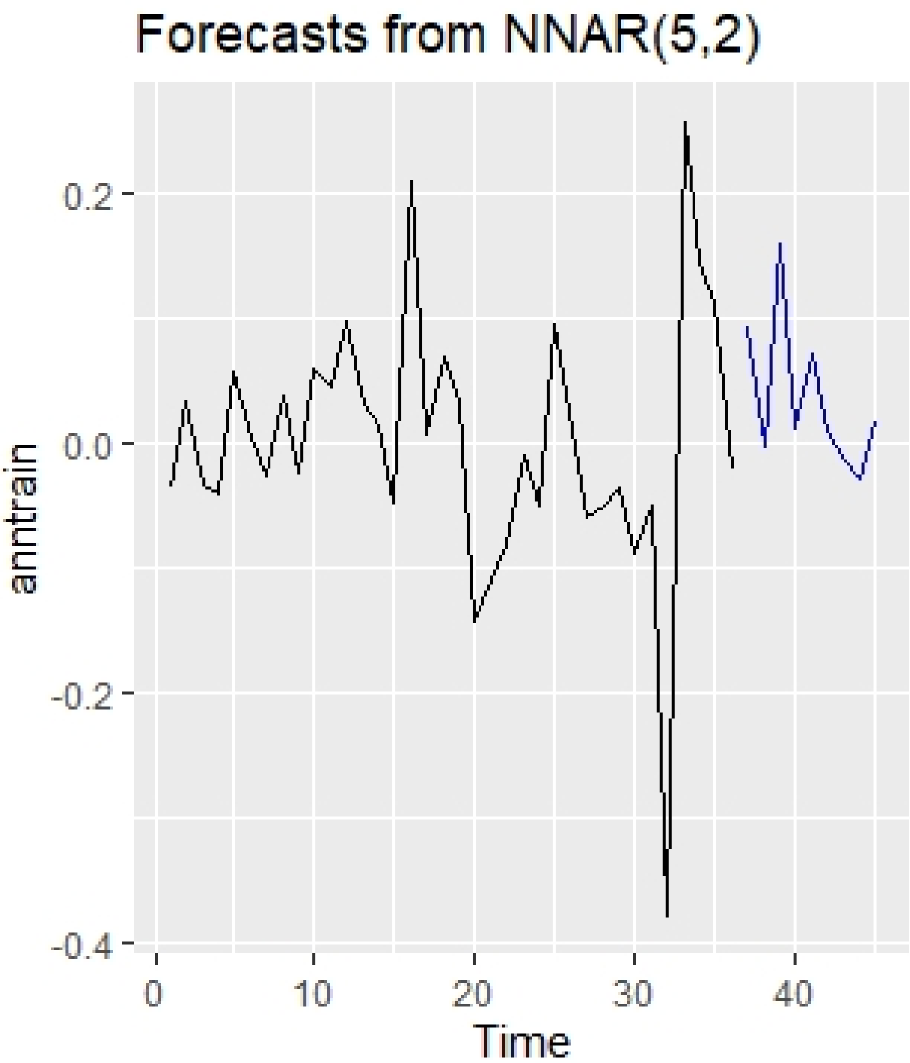

Figure 4.7: NNAR(5,2)forecast against actual residuals plot

4.3.1 Overall Comparative Forecasting Accuracy Evaluation

The best ARIMA model based on validation set forecasts is compared with the hybrid model using the RMSE criterion, resulting in the validation set RMSE indicated in Table 4.3. This shows that the Hybrid Model fails to outperform the ARIMA model based on this single criterion.Further, the time plot of the two model's forecasts against the actual test data in Figure 4.8 indicates that the ARIMA(2,2,1) model has the closest and most consistent pattern to the actual test data.

Table 4.3: Hybrid Versus ARIMA RMSE

Statistic	**ARIMA(2,2,1)**	**HYBRID**	**% RMSE difference**
RMSE	0.0653	0.0869	33.0781%

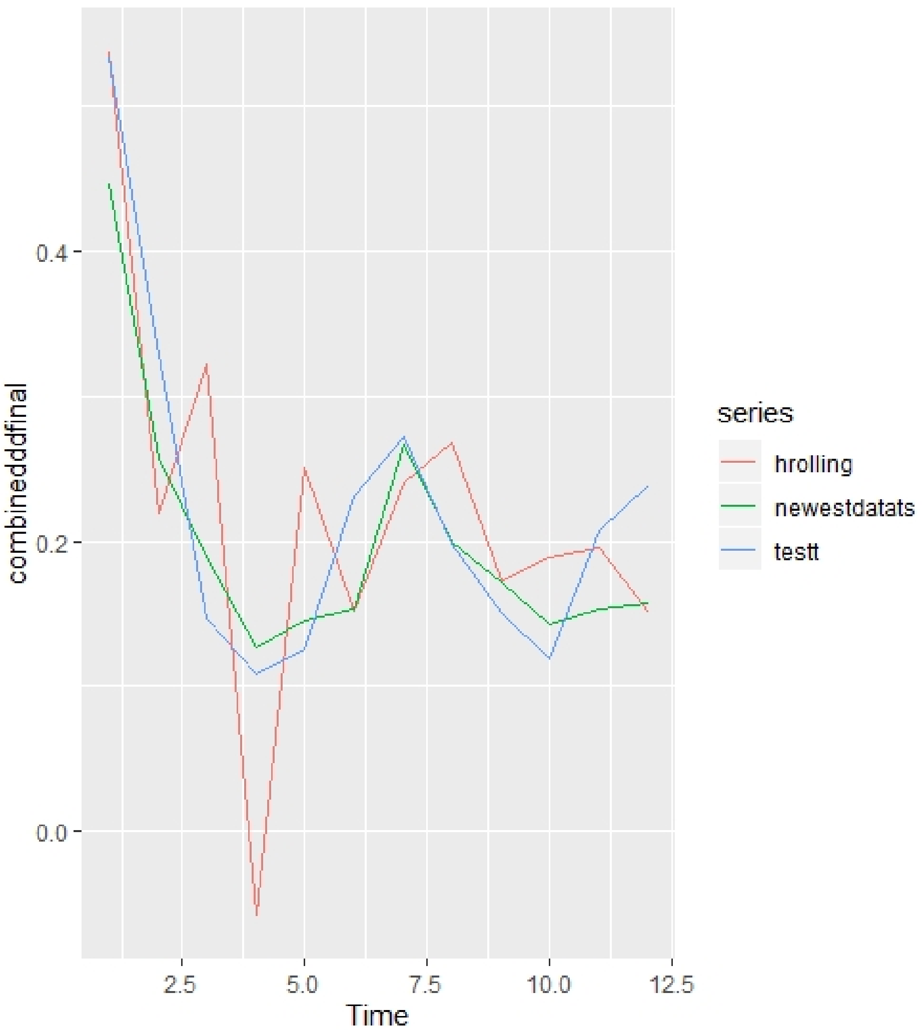

Figure 4.8: Rolling regression based ARIMA,Hybrid model against Actual data

Armstrong and Callopy (1992) study the error measures for generalizing about forecasting methods and they conclude that the RMSE is able to describe the magnitude of the forecast error in terms that would be relatively more useful to decision makers compared to unit free measures as well as relative measures such as Mean Absolute Percentage Error and the Relative Absolute Error, respectively.Lack of existence of other forecast precision measures of the hybrid model limits our comparison basis, moreso the ANN part of the hybrid structure lacks specifically applicable and tailored measures from the reviewed Literature.

To assess the extent to which the hybrid model forecasts differs from the ARIMA model forecast with regard to forecasts with a horizon of $h = 12$ on the test data, we use two criteria the Diebold and Mariano test statistic and the percentage difference between the two model's RMSE. As tabulated in Table 4.3 the RMSE percentage difference criterion shows that the Hybrid model is less accurate than the ARIMA model by 30% on an RMSE basis.This difference is significant and is largely attributed to the partially arbitrary specification of the parameters of the ANN part of the Hybrid model. Other measures other than the percentage difference in RMSE of forecasts is desirable since the robustness of one model over the other would be assessed more specifically giving deeper insight into any additional forecasting gains of one model over the other.In other studies such as Marcellino (2008), the linear model proved to outperform the non-linear models in modelling GDP and Inflation with conclusions being drawn that the best model yielded gains of 10–15% with respect to the best linear specifications.

The Diebold and Mariano test is also applied to compare the two model's forecast ability.The null hypothesis is that the two forecasts have the same accuracy while the alternative hypothesis is that the two forecasts have different levels of accuracy. The Diebold-Mariano statistic obtained is -1.3378 such that the null hypothesis of no difference is not rejected since the computed Diebold-Mariano statistic falls within the range of -1.96 to 1.96 at 5% level of significance. This result shows that based on the two model's residuals, there is no significant difference in GDP forecast accuracy between the Hybrid and pure ARIMA model.This is unlike the aforementioned RMSE criteria result.

4.3.2 Comparison with International Monetary Fund Forecasts

To compare the short-term performance of the hybrid model with the IMF forecasts for real GDP growth,we generate the forecasts between 2019 and 2021 using both ARIMA (2,2,1) and the Hybrid model.The results are as in Table 4.4. Assuming the IMF forecasts are the industry benchmarks then the RMSE for the ARIMA and the Hybrid Model against the IMF forecasts can be obtained.Both the ARIMA model and the Hybrid model perform relatively equally against the IMF benchmark since their RMSE values have

Table 4.4: Transformed GDP Data Forecast Comparison 2019 to 2021

Year	**ARIMA(2,2,1)**	**HYBRID**	**IMF**
-		ARIMA(2,2,1)NNAR(5,2)	
2019	0.1637	0.1559	0.2461
2020	0.1593	0.1602	0.1093
2021	0.1605	0.1216	0.0624
Average RMSE	0.0794	0.0688	

a marginal difference of 0.01, with the Hybrid model having the lower RMSE indicating that its 3 years forecast resembles the IMF forecasts more closely than the ARIMA model.

Chapter 5

Conclusion and Recommendations

5.1 Conclusion

Based on the results discussed in this Chapter, this study concludes that the hybrid model composed of the ARIMA(2,2,1) model and the Neural Network Auto-regressive feed-forward model with the structure NNAR(5,2) representing 2 hidden layers of the fifth lag order of the ARIMA residuals fails to outperform the pure ARIMA model in forecasting the Kenyan GDP out of sample since it does not produce precise forecasts as measured by the RMSE metrics.However, when compared out of sample against the IMF forecasts as a benchmark the Hybrid model performs marginally better than the ARIMA for short term 3 year forecasts.The Neural Network architecture specification requires a proven and more definite methodology that can yield optimum parameters rather than arbitrarily informed parameters.

There are very few forecast accuracy measures specifically tailored to assess the Artificial Neural Network model given the general limited understanding of its inner workings.As a component model in the hybrid model the Autoregressive Neural Network model's performance needed to be assessed prior to choosing its best specification for accurate forecasts, therefore this study used the hold out validation criteria on the data to provide a basis for assessing the trained neural network moreso using the test dataset RMSE.

5.2 Recommendations

The current study has certain limitations that can be improved on in further studies.Chief amongst them is the Box–Jenkins methodology used in ARIMA modelling that focuses on the low order autocorrelation only.Further the amount of data used in this study is not sufficient enough to generalize the performance of the hybrid model to other countries' data, therefore a wider data set would give the Hybrid model a better competitive chance given that the ANN portion of the Hybrid model requires data in bulk to better capture the relationships within the data and subsequently producing more accurate forecasts.

Several interesting directions also arise for future research such as investigating higher order autoregressive lags if a sufficiently large database is available where models that incorporate higher order autocorrelations can be explored as part of the hybrid structure. Also, other structures of the ANN as incorporated in software can be explored with availability of a large dataset such as the Multilayer Perceptron structure in R studio and other non-bounded structures in Python software can be explored. Moreover, future work on non-linear modelling of time series could extend the Auto-regressive Neural Network to include a non-linear Moving Average component for the Neural Network residuals.

Bibliography

[1] Camba-Mendez, Gonzalo, Kapetanios, George, Smith, Richard J, Weale, Martin R. 2001. An automatic leading indicator of economic activity: forecasting GDP growth for European countries. The Econometrics Journal, 4(1), S56S90.

[2] Aminian, Farzan and Suarez, E Dante and Aminian, Mehran Walz, Daniel T., 2006. Forecasting economic data with neural networks. Computational Economics Journal, 28(1), 7188.

[3] Andrews, Bruce H., Dean, M.D., Swain, R., Cole, C., 2013. Building ARIMA and ARIMAX models for predicting long-term disability benefit application rates in the public/private sectors Society of Actuaries, 154.

[4] Box, George E.P., Jenkins, G.M., 1968. Some recent advances in forecasting and control. Journal of the Royal Statistical Society. Series C (Applied Statistics), 17(2), 91109.

[5] Brockwell, P.J., Davis, R.A. 2016. Introduction to time series and forecasting. Springer, 17(2), 91109.

[6] Kokkinen,A., Wouters, H.,2016. Euro area and European Union GDP flash estimates at 30 days. 34th IARIW General Conference Dresden, Germany.

[7] Girosi, F., Jones, M. Poggio, T. 1995. Regularization theory and neural networks architectures. Neural computation Journal, 7(2), 219269.

[8] Granger, C.J,1989. Invited review combining forecaststwenty years later, Journal of Forecasting,8(3), 167173

[9] Guptha, K Siva Kiran and Rao, R Prabhakar, 2018. causal relationship between Financial development and economic growth: an experience with BRICS economies, Journal of Social and Economic Development,20(2) 308326

[10] Hyndman, Rob J and Athanasopoulos, George. 2018. Forecasting: principles and practice. OTexts

[11] Iacobucci, A. 2005. Spectral analysis for economic time series. New Tools of Economic Dynamics,203219. publisher=Springer

[12] James, G., Witten, D., Hastie, T., and Tibshirani, R. 2013. An introduction to statistical learning,112 Springer

[13] Khandelwal, Ina and Adhikari, Ratnadip and Verma, Ghanshyam Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition, Procedia Computer Science,48,173179,

[14] Khashei, M. Bijari, M., 2012 Which methodology is better for combining linear and nonlinear models for time series forecasting? Journal of Industrial and systems Engineering (JISE)

[15] Leamer, E.E.,2008 Macroeconomic patterns and stories, Springer Science Business Media

[16] Liu, C., Hoi, S., Zhao, P., and Sun, J., 2016. Online arima algo- rithms for time series prediction, Thirtieth AAAI conference on artificial intelligence.

[17] Yoo, J., and Maddala, G.S., 1991 Risk premia and price volatility in futures markets, The Journal of Futures Markets (1986-1998),11(2) 165.

[18] Ning, W., Kuan-jiang, B., and Zhi-fa, Y., 2010. Analysis and forecast of Shaanxi GDP based on the ARIMA Model, Asian Agricultural Research,2(1812-2016-143365) 3441.

[19] Pannakkong, W., Huynh, V., and Sriboonchitta, S. 2019. A Novel Hybrid Autoregressive Integrated Moving Average and Artificial Neural Network Model for Cassava Export Forecasting, International Journal of Compu- tational Intelligence Systems,12(2) 10471061.

[20] Porzecanski, A.C., 2018. Debunking the Relevance of the Debt-to-GDP Ratio, World Economics, March.

[21] Steinbuks, Jevgenijs Assessing the accuracy of electricity production forecasts in developing countries, International Journal of Forecasting,35(3) 11751185.

[22] Tamba, J.G., Essiane,S.N., Sapnken, E.F., Ko, F.D.,Nsouandele, J.L., Soldo, B., and Njomo, D., 2018 natural gas: a literature survey, International Journal of Energy Economics and Policy,8(3) 216249.

[23] Taskaya-Temizel, T., and Ahmad, K., 2005. Are ARIMA neural network hybrids better than single models?, Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.,5, (31923197)

[24] Tkacz, G., 2001 Neural network forecasting of Canadian GDP growth, International Journal of Forecasting,17(1) 5769.

[25] Soytas, U., Sari, R., 2019. Routledge Handbook of Energy Economics, Routledge

[26] United Nations Department of Economic and Social Aairs World Economic Situation and Prospects 2018, United Nations, 24

[27] Wabomba, M.S., Mutwiri, M.P., Mungai, F., 2016. and forecasting Kenyan GDP using Autoregressive Integrated Moving Average (ARIMA) models, Science Publishing GroupScience Journal of Applied Mathematics and Statistics,4(2) 6473.

[28] Wallison, P.J., and Nabil, R., 2018. Why Expert Economic Forecasts Are Often Wrong, The International Economy,32(3) 5051. publisher=International Economy Publications, Incorporated

[29] Wang, Li and Zou, Haofei and Su, Jia and Li, Ling and Chaudhry, S., 2013. An ARIMA-ANN hybrid model for time series forecasting, Systems Research and Behavioral Science,30(2) 244259. publisher=Wiley Online Library

[30] Were, Maureen and Kamau, A and Sichei, M and Kiptui, M A theoretical framework for Kenyas central bank macro econometric model, Africa growth initiative working paper,

[31] Yang, Mofeng and Jiaohong, XIE and Peipei, MAO and Wang, Chao and Zhirui, Y.,2018 Application of the ARIMAX Model on Forecasting Freeway Trac Flow, CICTP 2017: Transportation Re- form and ChangeEquity, Inclusiveness, Sharing, and Innovation,593602 publisher=American Society of Civil Engineers Reston, VA

[32] Zhang, G.P., 2003 Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing Journal,50 159175. publisher=Elsevier

[33] Diebold, F Mariano and Mariano, R., 1995 R.(1995). Comparing predictive accuracy, Journal of Business economic statistics,20(1)

[34] Steinbuks, J., 2019. Assessing the accuracy of electricity production fore- casts in developing countries, International Journal of Forecasting,35 (3) 11751185.

[35] Vandekerckhove, J., Matzke, D., Wagenmakers, E., and others, 2015 Model comparison and the principle of parsimony, Oxford handbook of computational and mathematical psychology,300319. publisher=Oxford University Press Oxford, UK

[36] Jiang, S., Xiao, R., Wang, L., Luo, X., Huang, C., Wang, J., Chin, K., and Nie, X., 2019 Combining Deep Neural Networks and classical time series regression models for forecasting patient ows in Hong Kong, IEEE Access,7(118965118974) publisher=IEEE

[37] Karadag, M. and Westway, T., 1999. A SAM-based computable general equilibrium model of the Turkish economy, WP, Center for International Financial and Economic Research,

[38] Jiang, X., and Khan, H.A.,2017 Structural change and energy use in China: A SAM-based CGE analysis, The Chinese Economy,50(6) 405424 Taylor Francis

[39] Hodgess, Erin M and Mhoon, K., 2019 Temporal Disaggregation of Time Series Revisited, Management,7(4) 293299.

[40] Babu, C.N., and Reddy, B.E., 2014 A moving-average filter based hybrid ARIMAANN model for forecasting time series data, Applied Soft Computing,23 2738.

[41] Marcellino, Massimiliano; 2008 A linear benchmark for forecasting GDP growth and inflation? Journal of Forecasting, 27(4)305-340,Wiley Online Library.

[42] Peter, Durka and Silvia, Pastorekova, 2012. ARIMA vs. ARIMAXwhich approach is better to analyze and forecast macroeconomic time series, Proceedings of the 30th International Conference Mathematical Methods in Economics, Karvina, Czech Republic, 1113.

[43] Dumitru, C., and Maria, V., 2013. Advantages and Disadvantages of Using Neural Networks for Predictions., Ovidius University Annals, Series Economic Sciences, 13(1).

[44] Armstrong, J Scott and Collopy, Fred., 1992. Error measures for generalizing about forecasting methods: Empirical comparisons, International journal of forecasting,8(1).

[45] Rappai, Gabor,2015. Modelling Non-Equidistant Time Series Using Spline Interpolation, Journal of the Hungarian Central Statistical,22.

[46] Triebe, Oskar and Laptev, Nikolay and Rajagopal, Ram 2019. AR-Net: A simple Auto-Regressive Neural Network for time-series. arXiv preprint arXiv:1911.12436.

[47] Tashman, Leonard J 2000. Out-of-sample tests of forecasting accuracy: an analysis and review. International journal of forecasting: 164437-450,Elsevier.

[48] Bliemel, Friedhelm (1973) Theil's forecast accuracy coefficient: A clarification. SAGE Publications Sage CA: Los Angeles, CA.

[49] Zhang, G Peter and Berardi, VL., 2001. Time series forecasting with neural network ensembles:an application for exchange rate prediction. Journal of the operational research society.52,6,652-664 Springer

Appendices

5.2.1 Appendix A: ARIMA Model Diagnostic Plots

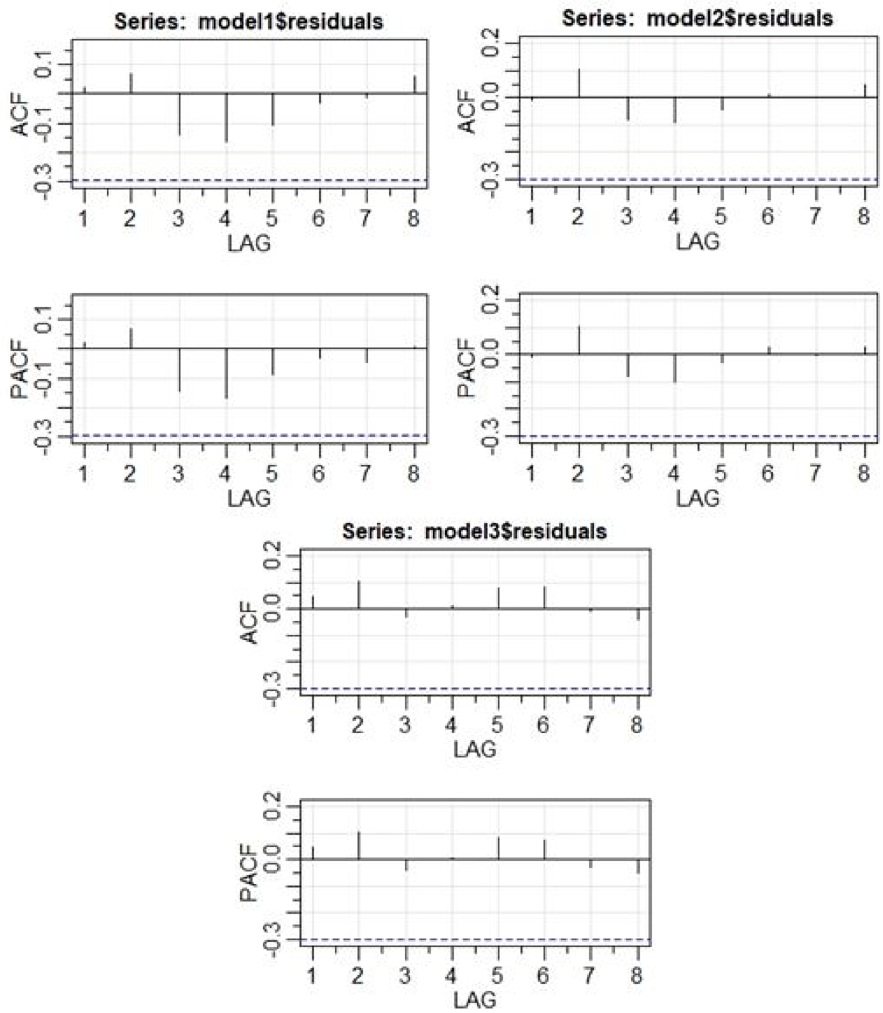

Figure 5.1: Fitted ARIMA models' ACF and PACF Plots

5.2.2 Appendix B: Data Transformation and NNAR Plots

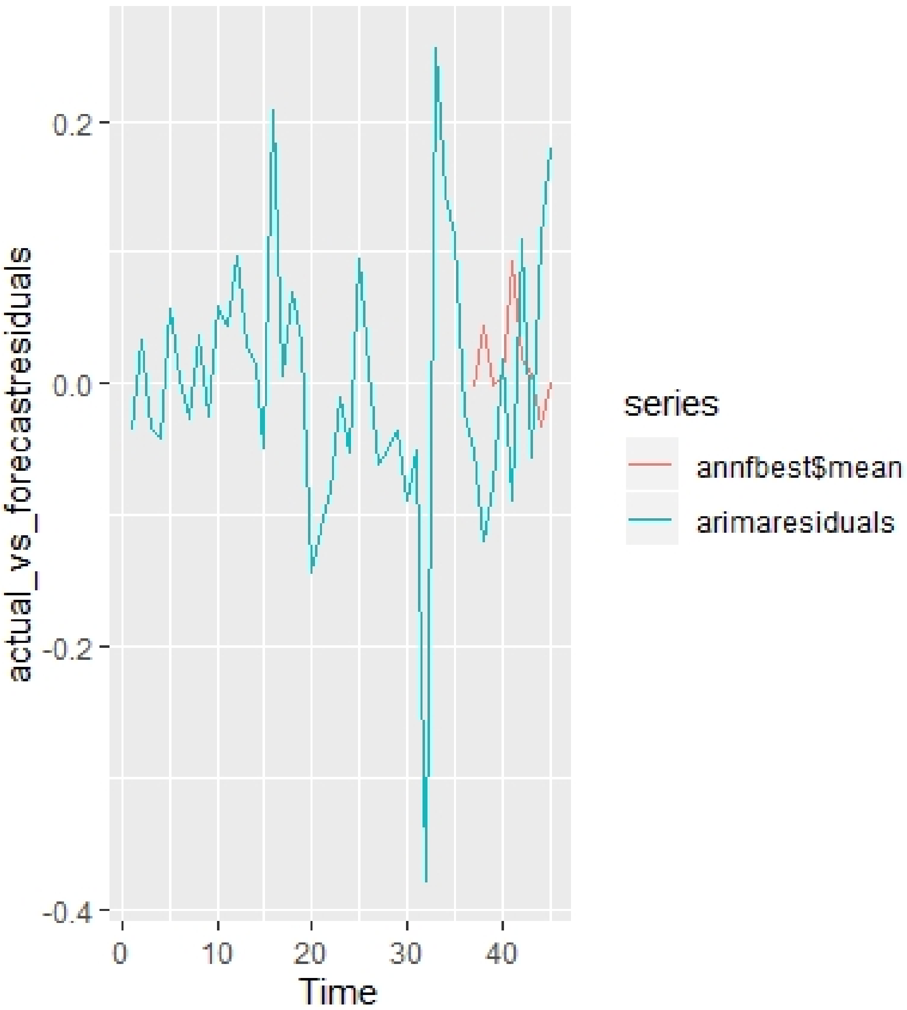

Figure 5.2: Transformed Data ACF and PACF Plot

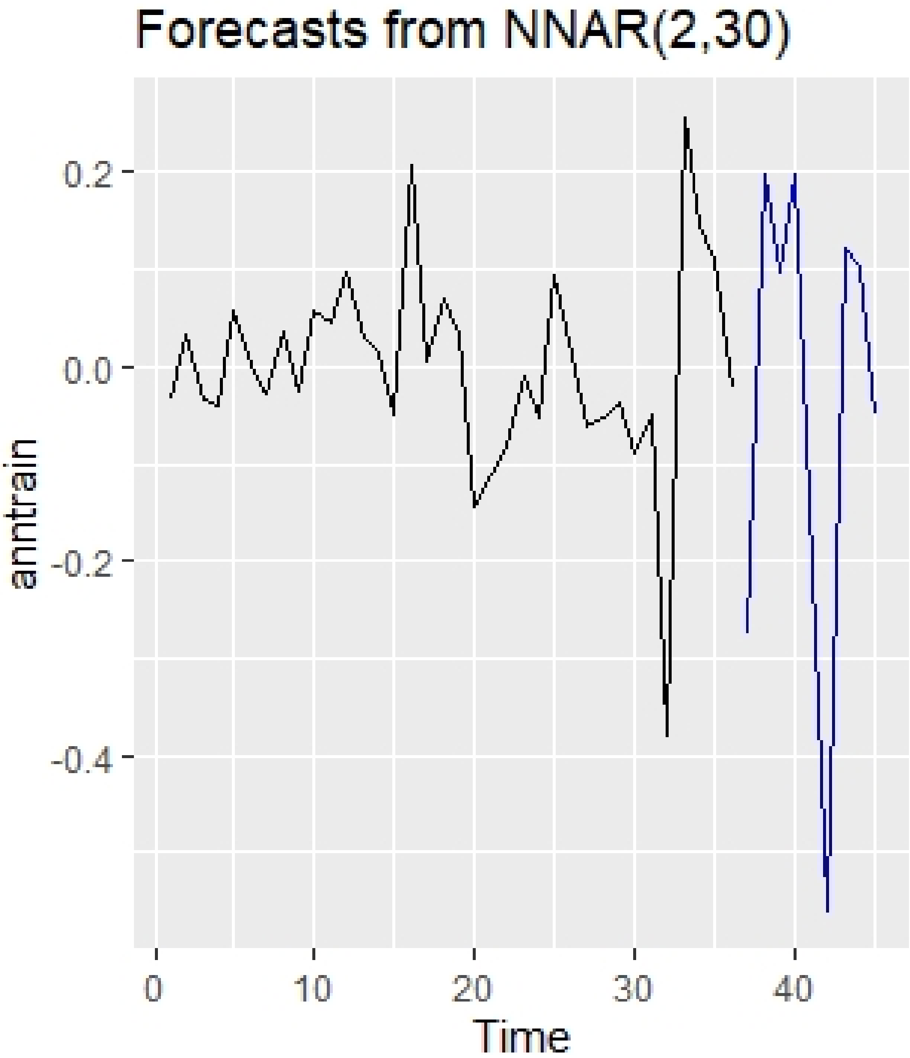

Figure 5.3: 100 ANN repeats

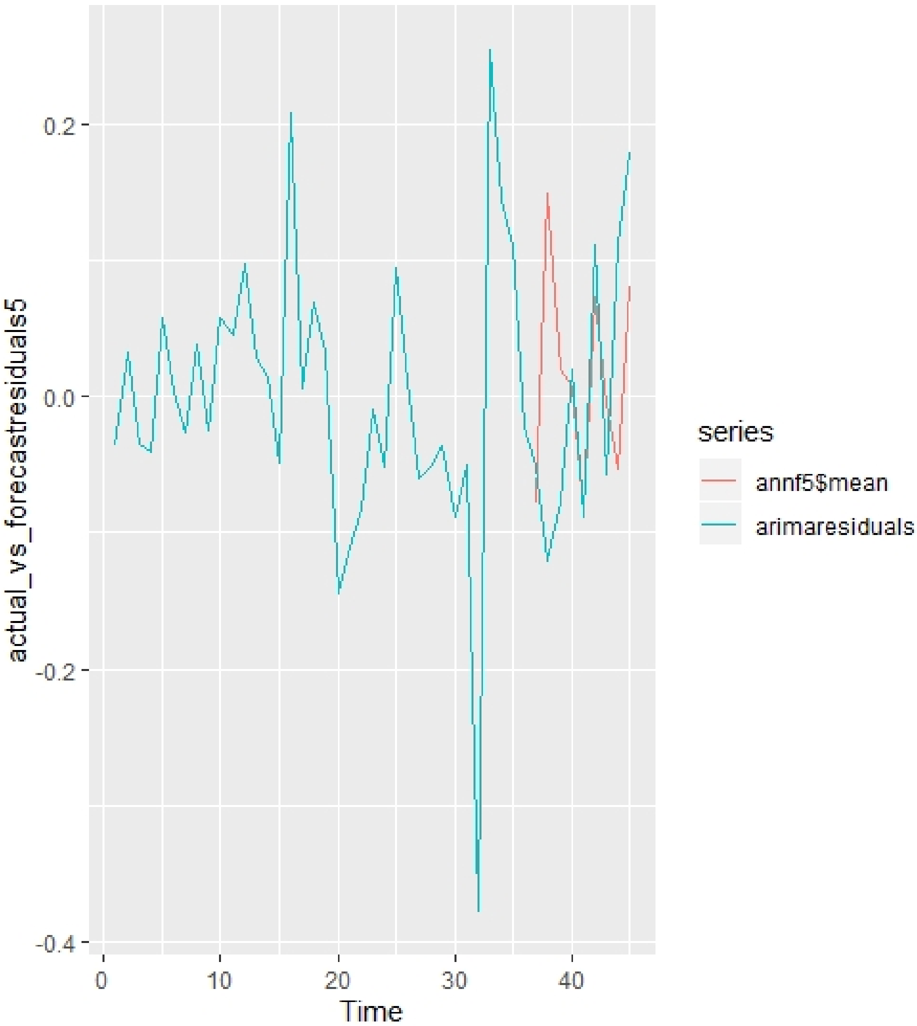

Figure 5.4: 10 hidden layers ANN

5.2.3 Appendix C: Turnitin Report

5.2.4 Appendix D: R Script

```
R Studio Code
###KENYA'S GDP FORECASTING USING ARIMA VERSUS HYBRID ARIMA+ANN MODEL FOR SHORT TERM
> #Load necessary files
> library(TTR)
> library("forecast")
> library(fpp)
> library("dataseries")
> library(WDI)

> #Read data from wdi database
> gdp=WDI(country=c("KE"), indicator=c("NY.GDP.MKTP.CD"), start=1960, end=2018)
> names(gdp)=c("Code","Country", "GDP","Year")
> head(gdp)
  Code Country         GDP Year
1   KE   Kenya 87908262520 2018
2   KE   Kenya 78757391333 2017

> #other important libraries
> library(ggplot2)
> library(scales)
> library(astsa)
>
> # annual  GDP data run sequence plot and convert data to ts
> ggplot(gdp, aes(Year, GDP, color=Country, linetype=Country)) +geom_line() + scale_
>
> data2=ts(gdp$GDP)
> data3 = ts(rev(data2),start=1960 ,end=2018)
> plot(data3, ylab="Annual GDP", xlab="Year")
#check for seasonality in annual data,none found
> decompose(data3)
Error in decompose(data3) : time series has no or less than 2 periods
>
> #view likely autoregressive/moving average characteristics of raw data-data 3
> acf(data3)
> pacf(data3)
> acf2(data3)
       ACF  PACF
 [1,] 0.89  0.89
 [2,] 0.79 -0.01
 [3,] 0.71  0.02
```

```
> #transform data via ln fxn to detrend and to transform to stationary series via di
> datalogged=log(data3)
> plot(datalogged)
> ndiffs(x=datalogged)
[1] 1
> plot(diff(datalogged, 1))
> databetween=diff(datalogged, 1)
> data4=(diff(datalogged, 2))
> plot(data4, main="Log differenced degree 2 series", xlab="Year")

> #Test stationarity BEFORE AND AFTER THE TRANSFORMATIONS
> adf.test(data3)

Augmented Dickey-Fuller Test

data:  data3
Dickey-Fuller = 4.4032, Lag order = 3, p-value = 0.99
alternative hypothesis: stationary

Warning message:
In adf.test(data3) : p-value greater than printed p-value
> adf.test(data4)

Augmented Dickey-Fuller Test

data:  data4
Dickey-Fuller = -3.992, Lag order = 3, p-value = 0.01629
alternative hypothesis: stationary

> #split the data to training and testing set
> require(caTools)
> library(caret)
> set.seed(101)
>#45 train and 12 test ie 80:20 split
> train= data4[1:45]
> test=data4[46:57]

> #try the manual arima modelling with different orders
> #model 1
> model1 = arima(train, order=c(2, 0, 1))
> acf2(model1$residuals)
       ACF  PACF
[1,]  0.02  0.02
[2,]  0.07  0.07
```

```
> urdftest1=ur.df(model1$residuals,type=c("none"),selectlags=c("AIC"))
> Box.test(model1$residuals, lag = 5, type = c("Ljung-Box"),fitdf=3)

Box-Ljung test

data:  model1$residuals
X-squared = 3.2473, df = 2, p-value = 0.1972

> summary(model1)

Call:
arima(x = train, order = c(2, 0, 1))

Coefficients:
         ar1      ar2     ma1  intercept
      0.3622  -0.0106  1.0000     0.1554
s.e.  0.1545   0.1545  0.1362     0.0466

sigma^2 estimated as 0.01069:  log likelihood = 35.97,  aic = -61.94

Training set error measures:
                     ME      RMSE        MAE      MPE     MAPE      MASE       ACF1
Training set 0.00081463 0.1033885 0.07594598 26.87002 87.44374 0.7212027 0.02205024
> acf2(model1$residuals)
> #if we had more data we'd explore lag 13 residuals in the ann as highlighted by pa
>
> Box.test(model1$residuals,lag =7,  type = c("Ljung-Box"),fitdf=3)

Box-Ljung test

data:  model1$residuals
X-squared = 3.3097, df = 4, p-value = 0.5074

> Box.test(model2$residuals, lag = 7, type = c("Ljung-Box"),fitdf=4)

Box-Ljung test

data:  model2$residuals
X-squared = 1.4039, df = 3, p-value = 0.7046

> Box.test(model3$residuals, lag = 7,type = c("Ljung-Box"),fitdf=6)

Box-Ljung test
```

```
data:  model3$residuals
X-squared = 1.38, df = 1, p-value = 0.2401

>
> checkresiduals(model1)


> urdftest1=ur.df(model1$residuals,type=c("none"),selectlags=c("AIC"))
> urdftest2=ur.df(model2$residuals,type=c("none"),selectlags=c("AIC"))
> summary(urdftest2)

###############################################
# Augmented Dickey-Fuller Test Unit Root Test #
###############################################

Test regression none


Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)

Residuals:
     Min       1Q   Median       3Q      Max
-0.37460 -0.04932  0.00484  0.05682  0.22829

Coefficients:
           Estimate Std. Error t value Pr(>|t|)
z.lag.1     -0.8893     0.2309  -3.852 0.000404 ***
z.diff.lag  -0.1124     0.1607  -0.699 0.488251
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1023 on 41 degrees of freedom
Multiple R-squared:  0.4956,Adjusted R-squared:  0.471
F-statistic: 20.15 on 2 and 41 DF,  p-value: 8.057e-07


Value of test-statistic is: -3.8517

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.62 -1.95 -1.61
> #best in-sample model ARIMA (2,0,2)
```

```
> model2

Call:
arima(x = train, order = c(2, 0, 2))

Coefficients:
         ar1      ar2    ma1     ma2  intercept
      1.3012  -0.4081  0.000  -1.000     0.1469
s.e.  0.1388   0.1396  0.185   0.185     0.0155

sigma^2 estimated as 0.009697:  log likelihood = 37.32,  aic = -62.65
>
#Fit the best ARIMA Model to the TRAIN Data and obtain forecasts to compare to test
> model1 <- Arima(train,order=c(2,0,1))
> ft1=forecast(model1,h=12)
> plot(ft1)
> arimapointforecast1=ft1$mean
>
> #####obtain what goes into the hybrid from ARIMA
> whatgoesintothehybrid=arimapointforecast1
>
> #############
> model2fit <- Arima(train,order=c(2,0,2))
> ft2=forecast(model2fit,h=12)
> plot(ft2)
> arimapointforecast2=ft2$mean
>
> model3fit <- Arima(train,order=c(2,0,4))
> ft3=forecast(model3fit,h=12)
> plot(ft3)
> arimapointforecast3=ft3$mean
> #RMSE for test data,Theil's innequality, MAPE
> accuracy(ft1$mean,test)
                 ME       RMSE        MAE      MPE     MAPE
Test set 0.02927012 0.06530553 0.05877928 4.174188 28.27423
> library(DescTools)
> TheilU(test,ft1$mean)
[1] 0.2618693
> #best model ARIMA (2,0,1) test data based out of sample
> model1

Call:
arima(x = train, order = c(2, 0, 1))
```

```
Coefficients:
         ar1      ar2     ma1  intercept
      0.3622  -0.0106  1.0000     0.1554
s.e.  0.1545   0.1545  0.1362     0.0466

sigma^2 estimated as 0.01069:  log likelihood = 35.97,  aic = -61.94
> acf2(model1$residuals)
       ACF  PACF
[1,]  0.02  0.02
[2,]  0.07  0.07
> acf(model1$residuals)
 ######################################################################################
> #Fit ANN on the ARIMA forecast residuals for short term forecast, 30 networks to f
> #different random starting weights,
> #AN NNAR
> #split residuals data to train nd test
> arimaresiduals=model1$residuals
> arimaresiduals
Time Series:
Start = 1
End = 45
Frequency = 1
 [1] -0.035535001  0.032816840 -0.034727275 -0.040928298  0.057299225  0.003488106 -
> anntrain=arimaresiduals[1:36]
> anntest=arimaresiduals[37:45]
>
> #fit different specifications of number of hidden nodes
> #and different number of runs/repeats in NNAR Model
> set.seed(1245)
>
> #model1
> annmodel= nnetar(anntrain,size=2,p=2,repeats=30)
> summary(annmodel)
          Length Class        Mode
x         36     ts           numeric
m          1     -none-       numeric
p          1     -none-       numeric
P          1     -none-       numeric
scalex     2     -none-       list
size       1     -none-       numeric
subset    36     -none-       numeric
model     30     nnetarmodels list
nnetargs   0     -none-       list
fitted    36     ts           numeric
```

```
residuals 36     ts           numeric
lags       2     -none-       numeric
series     1     -none-       character
method     1     -none-       character
call       5     -none-       call
> print(annmodel)
Series: anntrain
Model:  NNAR(2,2)
Call:   nnetar(y = anntrain, p = 2, size = 2, repeats = 30)

Average of 30 networks, each of which is
a 2-2-1 network with 9 weights
options were - linear output units

sigma^2 estimated as 0.005291
#model2,3,4 vary the lags, hidden layer and runs respectively
> annmodel2= nnetar(anntrain,size=2,p=5,repeats=30)
> print(annmodel2)
Series: anntrain
Model:  NNAR(5,2)
Call:   nnetar(y = anntrain, p = 5, size = 2, repeats = 30)

Average of 30 networks, each of which is
a 5-2-1 network with 15 weights
options were - linear output units

sigma^2 estimated as 0.003106
#Other variations
> annmodel5= nnetar(anntrain,size=30,p=2,repeats=30)
> print(annmodel5)
Series: anntrain
Model:  NNAR(2,30)
Call:   nnetar(y = anntrain, p = 2, size = 30, repeats = 30)

Average of 30 networks, each of which is
a 2-30-1 network with 121 weights
options were - linear output units

sigma^2 estimated as 4.438e-05
> #load new libraries
> library(sweep)
> library(tibble)
> library(tidyverse)
> sweep::sw_glance(annmodel)
```

```
# A tibble: 1 x 12
  model.desc  sigma logLik AIC   BIC            ME   RMSE    MAE   MPE  MAPE  MASE
  <chr>       <dbl> <lgl>  <lgl> <lgl>       <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl>
1 NNAR(2,2)  0.0727 NA     NA    NA    -0.00000706 0.0727 0.0523  38.4  93.7 0.553 -
> sweep::sw_glance(annmodel2)
# A tibble: 1 x 12
  model.desc  sigma logLik AIC   BIC          ME   RMSE    MAE   MPE  MAPE  MASE
  <chr>       <dbl> <lgl>  <lgl> <lgl>     <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl>   <

> #forecast 9 steps ahead to compare to the 9 test residuals data
> annmodelforecast=forecast(annmodel,h=9)
> autoplot(annmodelforecast)
>
>
> annmodelforecast2=forecast(annmodel2,h=9)
> autoplot(annmodelforecast2)
>
> #explore other possible variations
> annmodelforecast5=forecast(annmodel5,h=30)
> autoplot(annmodelforecast5)
> #compare ann forecasts and the testsample of the best ARIMA residuals
> accuracy(annmodelforecast,anntest)
                        ME       RMSE        MAE      MPE     MAPE      MASE
Training set -7.056737e-06 0.07274002 0.05228278 38.39789 93.67535 0.5534668 -0.0507
Test set     -1.131732e-02 0.10180822 0.08660651 91.11620 91.11620 0.9168186

> #best ann model models 3,4,6 since model 1 appears overfitted given the out of sam
annmodel2 as the best nnar5,2
> annmodel
Series: anntrain
Model:  NNAR(2,2)
Call:   nnetar(y = anntrain, p = 2, size = 2, repeats = 30)

Average of 30 networks, each of which is
a 2-2-1 network with 9 weights
options were - linear output units

sigma^2 estimated as 0.005291
> #extract best ann model's values (fitted and residuals)
> fittedannmodel3=sweep::sw_augment(annmodel3)
> fittedannmodel3
# A tibble: 36 x 4
   index  .actual  .fitted   .resid
   <int>    <dbl>    <dbl>    <dbl>
```

```
 1     1 -0.0355  NA       NA
 2     2  0.0328  NA       NA
 3     3 -0.0347  -0.0377   0.00294

> fittedannmode4=sweep::sw_augment(annmodel4)
> fittedannmode4
# A tibble: 36 x 4
   index  .actual   .fitted   .resid
   <int>    <dbl>     <dbl>    <dbl>
 1     1 -0.0355  NA         NA
 2     2  0.0328  NA         NA
 3     3 -0.0347   0.00112  -0.0358

> #fit best ann model 6 to the full residuals data set and forecast
> fittedannmodelbest= nnetar(arimaresiduals,size=2,p=5,repeats=30)
> annmodelforecast6=forecast(fittedannmodelbest,PI=TRUE,h=12)
> autoplot(annmodelforecast6)
> annmodelforecast6

> p=predict(fittedannmodelbest, n.ahead=12, se.fit=TRUE)
> #extract specific columns
> #Obtain ANN results- the lag 2 input model with 5 hidden neurons has an MsE of
>
> #Combine ANN and ARIMA model forecasts to form the hybrid model and COMPARE to
 Test data
> #hybrid=annforecast+arimaforecastfit
> #hybrid=arimapointforecast1+annmodelforecast6
>
> h2=as.numeric(whatgoesintothehybrid) + as.data.frame(annmodelforecast6$mean)
> #compare the hybrid vs arima vs actual test data
> h2,test, arimapointforecast1
> accuracy(as.numeric(arimapointforecast1),test)
                 ME       RMSE        MAE      MPE     MAPE
Test set 0.02927012 0.06530553 0.05877928 4.174188 28.27423
> accuracy(h2,test)
                 ME      RMSE        MAE      MPE     MAPE
Test set 0.04899705 0.0852141 0.06714081 18.80779 33.05433
>
> residd=test-arimapointforecast1
> residdd=test-h2
>
> dm.test(residd,residdd,alternative = "less",h=12,power=2 )

Diebold-Mariano Test
```

```
data:  e1e2
DM = -1.3378, Forecast horizon = 1, Loss function power = 2, p-value = 0.104
alternative hypothesis: less

Warning message:
In dm.test(residd, residdd, alternative = "less", h = 12, power = 2) :
  Variance is negative, using horizon h=1
> #At 5% level of significance the dm statistic -1.3378 The null hypothesis of no di
> #computed DM statistic falls within the range of -1.96 to 1.96.
> #To compare the three forecasts below (like with like comparison ie
log-differenced with log-differenced data)
> #Forecast 3 years using the hybrid model and pure ARIMA from 2019 to 2021
and compare with IMF
> #FIT best ARIMA to gdp annual data to 2018(whole dataset of 58 entries)
> #exp(arimapointforecast1+datalogged[45:59])
> #forecast ARIMA and HYBRID 3 EXTRA steps ahead(2019,2020,2021)
>
> #FIT ARIMA AND FORECAST
> model1fit2021 <- Arima(data4,order=c(2,0,1))
> ft2021=forecast(model1fit2021,h=3)
> plot(ft2021)
> arima2021pointforecast=ft2021$mean


> arimaresiduals2021=ft2021$residuals
> ##################################################IGNORE#######################
> modelkfit <- Arima(ft2021$mean,order=c(2,0,1))
> goesintotheforecasthybrid=modelkfit$residuals
>
> ##FIT THE ANN TO THE FORECASTED  ARIMA RESIDUALS
> fittedannmodelbest2021= nnetar(arimaresiduals2021,size=2,p=5,repeats=30)
> annmodelforecast62021=forecast(fittedannmodelbest2021,PI=TRUE,h=3)
> plot(annmodelforecast62021)
>
> ##OBTAIN THE HYBRID FORECAST
 h2021=as.numeric(arima2021pointforecast)+as.data.frame(annmodelforecast62021)
> arima2021pointforecast
Time Series:
Start = 2019
End = 2021
Frequency = 1
[1] 0.1636762 0.1592938 0.1604744
> h2021
```

```
     Point Forecast      Lo 80     Hi 80      Lo 95     Hi 95
2019      0.1675819 0.08775513 0.2442836 0.05138241 0.2854922
> #compare with imf WDI
> #convert WDI growth predictions to logged data for like to like comparison
> #2017 4.9%, 2018 6.3%, 2019 5.6%, 2020 6.0%, 2021 5.8%, 2022 5.8%, 2023 5.8%,
2024 5.9%
> #https://www.imf.org/external/datamapper/NGDP_RPCH@WEO/OEMDC/ADVEC/WEOWORLD/KEN
> #2018 absolute GDP was 87908262520
>
> projectactualgdpfrom2016to2017=69188755511*(1+0.049)
> projectactualgdpfrom2017to2018=78757391333*(1+0.063)
> projectactualgdpfrom2018to2019=87908262520*(1+0.056)
> projectactualgdpfrom2019to2020=projectactualgdpfrom2018to2019*(1+0.006)
> projectactualgdpfrom2020to2021=projectactualgdpfrom2019to2020*(1+0.058)
>
> actualgrowthabsolute=c(projectactualgdpfrom2016to2017,projectactualgdpfrom2017to20
+                        projectactualgdpfrom2018to2019,
projectactualgdpfrom2019to2020,projectactualgdpfrom2020to2021)
> actuallogged=log(actualgrowthabsolute)
> diflogged=diff(actuallogged,2)
> diflogged=as.numeric(diflogged)
> diflogged
[1] 0.24610630 0.10929683 0.06236241
> #compare difflogged(IMF) and h2021 and arima2021pointforecast  to compare IMF fore
> accuracy(c( 0.1559448,0.1601977, 0.1216490),as.numeric(diflogged))
                  ME       RMSE        MAE      MPE     MAPE
Test set -0.006675324 0.06888371 0.06678299 -35.0013 59.42475
> accuracy(arima2021pointforecast,as.numeric(diflogged))
                 ME       RMSE        MAE       MPE     MAPE
Test set -0.02189294 0.07941532 0.07684635 -56.52534 78.85447

###new method for manual rolling regression for ARIMA MODEL Shortterm h=3 forecasts
> train= data4[1:45]
> test=data4[46:57]
>
> modeltry <- Arima(train,order=c(2,0,1))
> prediction1=predict(modeltry,n.ahead = 3)
>
> train30= data4[1:48]
> modeltry2 <- Arima(train30,order=c(2,0,1))
> prediction2=predict(modeltry2,n.ahead = 3)
train31= data4[1:51]
> modeltry3 <- Arima(train31,order=c(2,0,1))
> prediction3=predict(modeltry3,n.ahead = 3)
```

```
>
> train32= data4[1:54]
> modeltry4 <- Arima(train32,order=c(2,0,1))
> prediction4=predict(modeltry4,n.ahead = 3)
>
> train33= data4[1:57]
> modeltry5 <- Arima(train33,order=c(2,0,1))
> prediction5=predict(modeltry5,n.ahead = 3)
>
> newestdata=c(prediction1$pred,prediction2$pred,prediction3$pred,prediction4$pred)
> newestdatats=as.ts(newestdata)
> h22=as.ts(h2)
> testt=as.ts(test)
>
> combineddd=cbind(testt,newestdatats,h22)
> autoplot(combineddd)
#compare to confirm if rolling method forecasts(2019-2021)
> # is similar to the forecast from the single fullfit arima (2019-2021)
> arima2021pointforecast
Time Series:
Start = 2019
End = 2021
Frequency = 1
[1] 0.1636762 0.1592938 0.1604744
> prediction5$pred
Time Series:
Start = 58
End = 60
Frequency = 1
[1] 0.1636762 0.1592938 0.1604744
> accuracy(test,newestdata)
                  ME       RMSE        MAE       MPE     MAPE
Test set -0.02072949 0.05171126 0.04233355 -8.808119 22.45774
>
> ###new method for manual rolling regression for ANN MODEL Shortterm h=3 forecasts
> anntrain=arimaresiduals[1:36]
> anntest=arimaresiduals[37:45]
>
> annmodel29= nnetar(anntrain,size=2,p=5,repeats=30)
> annmodelforecast29=forecast(annmodel29,h=3)
>
> anntrain30=arimaresiduals[1:39]
> annmodel30= nnetar(anntrain30,size=2,p=5,repeats=30)
> annmodelforecast30=forecast(annmodel30,h=3)
```

```
>
>
> anntrain31=arimaresiduals[1:42]
> annmodel31= nnetar(anntrain31,size=2,p=5,repeats=30)
> annmodelforecast31=forecast(annmodel31,h=3)
>
>
> anntrain32=arimaresiduals[1:45]
> annmodel32= nnetar(anntrain32,size=2,p=5,repeats=30)
> annmodelforecast32=forecast(annmodel32,h=3)
>
> newestdataANN=c(annmodelforecast29,annmodelforecast30,
+                  annmodelforecast31,annmodelforecast32)
>
>
> ##new hybrid using rolling regression
> hrolling=(newestdata)+(newestdataANN)
> anntestts=as.ts(anntest)
> newestdataANNts=as.ts(newestdataANN)
> combinedddfinal=cbind(hrolling,newestdatats,testt)
> autoplot(combinedddfinal)
>
> #compare to confirm if rolling method forecasts(2019-2021)
> # is similar to the forecast from the single fullfit ann (2019-2021)
> accuracy(anntest,newestdataANNts[1:9])
                 ME      RMSE       MAE       MPE     MAPE
Test set 0.01322104 0.1482623 0.1328885 -368.9723 2043.157
########### THE END ################
```