A COMPARATIVE ANALYSIS OF PRICE FORECASTING MODELS FOR BLACK PEPPER

By

AKSHAYA AJITH
(2021-19-004)

DEPARTMENT OF AGRICULTURAL STATISTICS
COLLEGE OF AGRICULTURE
VELLANIKKARA, THRISSUR - 680656
KERALA, INDIA

2024

A COMPARATIVE ANALYSIS OF PRICE FORECASTING MODELS FOR BLACK PEPPER

By

AKSHAYA AJITH
(2021-19-004)

THESIS

Submitted in partial fulfillment of the
requirement for the degree of

Master of Science in Agriculture

(AGRICULTURAL STATISTICS)

Faculty of Agriculture

Kerala Agricultural University

DEPARTMENT OF AGRICULTURAL STATISTICS

COLLEGE OF AGRICULTURE

VELLANIKKARA, THRISSUR - 680656

KERALA, INDIA

2024

DECLARATION

I, hereby declare that the thesis entitled **"A comparative analysis of price forecasting models for black pepper"** is a bonafide record of research work done by me during the course of research and the thesis has not previously formed the basis for the award to me any degree, diploma, fellowship or other similar title of any other University or Society.

Place: Vellanikkara

Date: 17/02/2024

Akshaya Ajith

(2021-19-004)

CERTIFICATE

Certified that the thesis entitled **"A comparative analysis of price forecasting models for black pepper"** is a bonafide record of research work done independently by **Ms. Akshaya Ajith (2021-19-004)** under my guidance and supervision and that it has not previously formed the basis for the award of any degree, diploma, fellowship or associateship to her.

Place: Vellanikkara
Date : 17/02/2024

Ms. Sajitha Vijayan M
Assistant Professor and Head
Department of Agricultural Statistics
College of Agriculture, Vellanikkara

CERTIFICATE

We, the undersigned members of the advisory committee of **Ms. Akshaya Ajith (2021-19-004),** a candidate for the degree of **Master of Science in Agriculture** with major field in **Agricultural Statistics,** agree that the thesis entitled **"A comparative analysis of price forecasting models for black pepper"** may be submitted by **Ms. Akshaya Ajith**, in partial fulfillment for the degree.

Ms. Sajitha Vijayan M
Assistant Professor and Head
Department of Agricultural Statistics
College of Agriculture, Vellanikkara

Dr. Anil Kuruvila
Professor
Department of Agricultural Economics
College of Agriculture, Vellanikkara

Ms. Dayana David
Assistant Professor
Department of Agricultural Statistics
College of Agriculture, Vellanikkara

Dr. Pratheesh P. Gopinath
Assistant Professor and Head
Department of Agricultural Statistics
College of Agriculture, Vellayani

ACKNOWLEDGEMENT

The preparation of this thesis has been one of the most significant academic challenges I have ever had to face. It marks the end of my journey in gaining my degree in M.Sc. Agricultural Statistics, fulfilling my passion towards the subject. I owe my deepest gratitude to all the personages without whose support, patience and guidance, the study would not have been completed.

First and foremost, I bow my head before the God Almighty for giving me the strength, knowledge, ability and opportunity to undertake this research study and to persevere and complete it successfully.

I would primarily like to express deep gratitude to my esteemed guide ***Ms. Sajitha Vijayan M*** *(Assistant Professor and Head, Department of Agricultural Statistics) for her expert advice, valuable guidance, practical suggestions, constant patience, inspiring encouragement, and timely help at various stages of my research work and thesis preparation and will be remembered forever. It was a great privilege to work and study under her guidance.*

I am really fortunate to get exemplary guidance, constant encouragement and compassion from ***Ms. Dayana David*** *(Assistant Professor, Department of Agricultural Statistics), one of my advisory committee members, for her valuable suggestions, kind advice, friendly approach and constant encouragement given throughout the research period.*

I extend my special thanks to ***Dr. Anil Kuruvila*** *(Professor, Department of Agricultural Economics), who is a member of my advisory committee, for his advices and suggestions that have served as the major contributions towards the completion of the project.*

I also take this opportunity to extend sincere thanks to ***Dr.Pratheesh P. Gopinath*** *(Assistant Professor and Head, Department of Agricultural Statistics, College of Agriculture, Vellayani) who is also an advisory committee member, for his guidance and useful critiques that helped me throughout my research.*

I am extremely grateful to my beloved father ***Mr. Ajithkumar M. A.*** *and mother* ***Mrs. Vanisree V R*** *for their love, prayers, caring and sacrifices for educating and preparing me for my future. Profound affection and love to my sister,* ***Krishna Ajith*** *who serves as a great support to me, who keeps on encouraging me to always make the best out of everything. I also thank my grandparents for their practical and emotional support.I expresss my gratitude to* ***Mr. Karthik, Ms. Dayana, Mr. Adarsh and Mr. Joyal Mathew*** *for supporting me to assist in scripting of codes in Python language.*

I owe a great deal of appreciation and gratitude to thank ***Mr. Ayyoob K.C****,* ***Ms. Nayana Naryanan, Mrs. Anagha V Gopal, Ms. Geethu P George, Mr. Vishnu B.R., Mr. Adarsh, Ms. Harshini, Ms. Jesma*** *for their immense support and technical help lent during urgent and needy times.*

I also express my gratitude to my dearest classmate ***Mr. Muhammed Sameer****, beloved seniors* ***Ms. Deenamol Joy and Ms. Anjana B Krishna,*** *junior* ***Mr. Abhishek Krishnan*** *and staff of the department* ***Sheena chechi*** *for their valuable support as well as* ***Naanu chettan, Leela chechi, Chandran chettan*** *and* ***Anil chettan*** *for their love through food in* ***Pamba Mess****.*

I would also like to greatly thank some people towards the final days of my research tenure who immensely supported me both physically and mentally and to whom I am greatly indebted, my dear friends, ***Joyal, Anjitha, Chenna, Subin, Shamna, Archana, Neethu, Ansu, Adheena and Nayana.***

I acknowledge the whole-hearted co-operation and gracious help rendered by each and every member of the ***College of Agriculture*** *during the period of study.*

For the whole journey, my head bows to Kerala Agricultural University for supporting me financially by provision of scholarship and letting my dreams come true...

Akshaya Ajith.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

ABBREVATIONS

ACF	Autocorrelation Function
ADF	Augmented Dickey Fuller test
AIC	Akaike Information Criterion
AR	Auto regressive
ARCH	Autoregressive Conditional Heteroskedasticity
ARIMA	Autoregressive Integrated Moving Average
ANN	Artificial Neural Network
CMA	Centred Moving Average
DES	Double Exponential Smoothing
GARCH	Generalized Autoregressive Conditional Heteroskedasticity
HWAS	Holt-Winters' Additive Seasonal
HWMS	Holt-Winters' Multiplicative Seasonal
LM	Lagrange Multiplier
LSTM	Long Short -Term Memory
MA	Moving Average
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MSE	Mean Square Error
PACF	Partial Autocorrelation Function

RMSE	Root Mean Square Error
SARIMA	Seasonal Autoregressive Integrated Moving Average
SIC	Schwartz Information Criteria
SES	Single Exponential Smoothing
TS	Time series
RNN	Recurrent Neural Network
TDNN	Time Delay Neural Network

Introduction

1. INTRODUCTION

Black pepper, often referred to as the "king of spices," stands as one of the most popular and widely consumed spices which shares a place on most dinner tables with salt. India ranks as the third largest producer in the world (International Pepper Community, 2023), also a significant consumer and exporter of black pepper, with Kerala and Karnataka producing the majority of the nation's output. The total area, production, and productivity of black pepper is 3,04520 ha, 90150 metric tonnes, and 0.30 metric tonnes/ha, respectively in India (GOI, 2023). Recently, Kerala ranked second in terms of black pepper acreage (76,160 ha), and production (33290 metric tonnes) but seventh in terms of productivity (0.44 metric tonnes/ha) (GOI, 2023).

Black pepper, derived from the *Piper nigrum* plant native to the Indian subcontinent, stands as a globally cherished spice with a rich history and widespread culinary use. Harvested in its green, unripe state and later dried to obtain its signature black peppercorns, this spice imparts a pungent and slightly spicy flavour due to the presence of piperine. The productive lifespan of black pepper plants is approximately 20 years, and the vines begin to experience a decrease in yield after consistently bearing fruit for about two decades. With potential medicinal benefits and a continued economic impact, pepper stands as a versatile and culturally significant spice with a rich history spanning thousands of years.

Pepper, additionally referred to as "black gold," was significant in the history of Kochi, Kerala. Since Kochi began to develop as a port in the fourteenth century, pepper and other spices were transported to other locations throughout the world by both the Arabs and the Europeans. Historically, the market value of pepper contributed to the development of the city of Kochi as a centre of international commerce. The first exclusive pepper exchange in India is situated in Kochi, established by the Indian Pepper and Spice Traders Association (IPSTA). The exchange is effectively regulated by the traditional stakeholders in the region, ensuring a lack of defaults in both supply and delivery of the commodity, as well as minimizing volatility.

Black pepper prices exhibit significant volatility, reaching over Rs. 500/kg, marking the highest recorded price in Kochi market since 2022, as reported by the Spices Board. This surge is attributed, in part, to apprehensions about a potential decline in production in the upcoming harvest, influenced by adverse weather conditions impacting the pepper yield. Fluctuations in prices of agricultural commodities affect supply and demand, consequently having a significant impact on consumers. Black pepper being a perennial crop, the significant fluctuations in prices throughout a year pose a major challenge for both farmers and consumers. Therefore, the examination of time series data related to black pepper prices is crucial.

A time series refers to a systematically arranged set of data points for a specific variable, recorded at regular time intervals. The analysis of a time series serves two primary objectives: firstly, to forecast upcoming values by applying a suitable predictive model to the dataset, and secondly, to decompose the time series into its four constituent elements—namely, trend, seasonal variation, cyclic variation, and random variation.

Forecasting the prices of agricultural commodities holds significance for farmers, governmental bodies, and agribusiness industries, providing valuable insights for policymakers, producers, and consumers to make informed decisions. Precise forecasts of agricultural commodity prices can help mitigate the risks associated with price fluctuations (Gu et al., 2022). Various forecasting models are accessible for time series data, broadly categorized into linear models, such as ARIMA models and Exponential Smoothing models, and nonlinear models, including artificial intelligence techniques.

The ARIMA model, introduced by Box-Jenkins (1970), gained popularity in time series forecasting due to its robust statistical properties and the systematic methodology it offers for constructing models. Exponential smoothing, another method employed in forecasting time series data, utilizes a moving average technique where past observations are assigned exponentially decreasing weights, with greater emphasis placed on recent observations. Different types of exponential smoothing models include Single Exponential Smoothing (SES), Holt's linear/Double Exponential Smoothing (DES), and Holt-Winters' Exponential Smoothing (HWMS). The selection of the most suitable model depends on the characteristics of the time series data being analysed.

The considerable fluctuations in the prices of black pepper clearly demonstrate the volatility in the commodity's price movements, presenting a notable challenge. It can be analysed through the statistical models like Auto Regressive Conditional Heteroskedasticity (ARCH) and Generalized Auto Regressive Conditional Heteroskedasticity (GARCH), which was introduced by Engle in 1988.

Price forecasting of agricultural commodities is being done for ages for various crops using traditional time series approaches. However, studies have found that in reality these series often have unknown nonlinear structure, which can be better understood using machine learning technology like artificial neural network (ANN), recurrent neural network (RNN), etc. A multivariate, non-linear, non-parametric statistical technique that is driven by data and self-adaptive is artificial neural network (ANN). One of its benefits is its adaptable functional structure, making it a universally applicable functional approximation (Zhang *et. al*.,1998). A Recurrent Neural Network (RNN) is a type of artificial neural network designed for processing sequential data by incorporating feedback loops that allow information to persist and be used in subsequent steps.

Black pepper is a major spice crop of Kerala. Throughout the years, the market value of black pepper has displayed significant fluctuations. An upswing in pepper prices affects consumers by raising their food expenses, while a decline in prices below the production cost has repercussions for the farmers. Concerning predicting the price of black pepper, no studies have been conducted so far. It is crucial to comprehend to what extend the fluctuations can be projected for the foreseeable future in order to draw appropriate policy conclusions.

In this context, the present study, 'A comparative analysis of price forecasting models for black pepper' is conducted, with the following objective:

Comparison of price forecasting models using time series and Artificial Neural Network approaches, for prices of black pepper.

Review of Literature

2. REVIEW OF LITERATURE

A thorough review of prior literature is advantageous in the formulation of concepts, methodologies, and analytical instruments for any research endeavour. In this chapter, an attempt has been undertaken to meticulously examine the prior research literature pertinent to the current study. This chapter is divided into five sections: Trend analysis and decomposition of time series components, Exponential smoothing models for price forecasting, ARIMA model for price forecasting, Price volatility, Artificial Neural Networks for price forecasting and Comparison of forecasting models.

2.1 Trend analysis and decomposition of time series components

Adanaliglu and Yercanm (2012) conducted an analysis to fit seasonal indices for tomato prices in Antalya and Turkey. The study utilized a 10-year dataset spanning from January 2000 to December 2010, and identified an increasing price trend during September and October, and a decreasing price trend in January. Monthly seasonal indices were computed to analyse the variations in tomato prices from 2000 to 2010. The findings revealed that the peak monthly seasonal index was observed in April, while the lowest occurred in June. Prices in April were around 70 per cent higher than the overall monthly average, whereas in June, they were 45 percent lower.

Kumaraswamy and Sekar (2014) conducted a study on the growth rates of potato cultivation along with the area, production, and yield, as well as the seasonal variations in potato prices across major markets in India. Utilizing a dataset covering a span of 23 years, the analysis revealed a positive growth rate in the potato cultivation area at the national level, while in Tamil Nadu, there was a declining trend in the cultivation area. The study also revealed an increasing price trend during the months of May to November.

The seasonality of the coconut oil, coconut, and copra prices in the Kochi, Alappuzha, and Kozhikode markets of Kerala, using monthly data from January 1980 to December 2015 was examined by Indraji (2016). The study found that the price of coconut oil followed a similar pattern in all the three markets, while the price of copra showed an increasing trend from June to August, and the price of coconut increased from August to September.

In their study, Sabu *et al*. (2018) analysed the dynamics of prices of black pepper during the pre-liberalization and post liberalization periods. The authors employed time series data encompassing monthly domestic and international prices of black pepper from January 1980 to December 2017 to estimate the components of the TS. Their findings indicated that the variability of prices increased during the post liberalization period. Additionally, the pre-liberalization era showed a higher frequency and magnitude of stochastic factors influencing black pepper prices.

Sutradhar *et al.* (2019) conducted a study on the seasonal and cyclical fluctuations in the prices of natural rubber in India, both domestically and internationally, over a span of 18 years from 2000-01 to 2017-18. Their findings revealed that the prices were at their lowest in the months of January and November, while they peaked in July and June. Additionally, the study explained that the natural rubber markets, both domestic and international, exhibited an erratic cycle.

Kachroo *et al.* (2021) analysed trend and seasonality in the prices and arrivals of Coriander, Castor seeds, Soybean, Turmeric, and Jeera for the period of 2004-2020 by collecting secondary data from CMIE (Centre for Monitoring Indian Economy) database. The seasonality of prices and arrivals for specific agricultural commodities throughout the year has been determined through the utilization of the moving average method. The findings of this investigation indicated that a significant and positive growth rate has been found in the prices of the chosen commodities. Jeera showed the most substantial growth rate in arrivals at 9.87 per cent, trailed by Turmeric at 9.05 per cent, with Coriander registering the lowest growth rate at 5.06 per cent. Conversely, the prices of the commodities displayed a different scenario, with Coriander having the highest rate at 7.55 per cent, Soybean at 7.24 per cent, and Jeera at the lowest rate of 5.69 per cent.

2.2 Exponential smoothing models

Using price data from May 1987 to May 2001, Vasanthakumar *et al.* (2005) conducted a study to forecast the price of various types of teak using an exponential smoothing model. They used the trial-and-error method to determine that selection weights are more important after observing that the single parameter exponential smoothing model performs better for forecasting in short-term using MS-Excel. Every year, the cost of living increased for all classes because of the growing population, increased demand, and limited availability of teak timber.

Huertas *et al.* (2007) analysed and evaluated the forecasting performance of Holt-Winters exponential smoothing model on the international tourism arrivals to Spain by the time 2007-08. This was based on primary data obtained from a survey of citizens of ten other significant countries of origin regarding their intended future travel to Spain.

Indraji (2016) used prices of copra, coconut oil, and coconuts to calculate seasonal variations. She also proposed appropriate forecast models for the markets of Alappuzha, Kochi, and Kozhikode. The HWMS model was determined to be the most suitable forecast model for the coconut oil price at the Kochi and Alappuzha markets. For all markets, the HWMS model was determined to be the best forecast model for copra.

Booranawong and Booranawong (2017) forecasted monthly lime prices in Thailand using the data from January 2011 to December 2015. They revealed that the Double Exponential Smoothing (DES) model shows suitable forecasting performances than the SES model and gave the smallest forecasting error measured by MAPE of 6.97 per cent.

2.3 ARIMA model for price forecasting

In 1981, Carter and Rausser conducted a study to evaluate the effectiveness of futures markets of soyabean by analysing their predictive capabilities in terms of bias and variability measures. Specifically, they investigated the "relative accuracy" condition of the soybean, soybean oil, and soybean meal futures markets using structurally-based ARIMA models. The results indicated that the constructed models surpassed the futures market in both short- and long-term forecasting.

Nabi and Shahahuddin (1998) employed the Box-Jenkins univariate ARIMA model to predict the price of natural rubber in Malaysia. Their study was conducted on the average monthly rates of natural rubber spanning from January 1977 to May 1987.

The ARIMA (2,0,2) model was utilized to forecast the costs of different grades of rubber. The resulting MAPE was observed to fluctuate between 2.65 per cent and 3.30 per cent.

Rangoda *et al.* (2004) conducted a study on the volatility of coconut and coconut product prices in Sri Lanka spanning from January 1974 to December 2004. The data was analysed utilizing six standard time series models, including the moving average method, decomposition method, exponential smoothing models, and ARIMA. The models were compared using MAPE and Mean Absolute Deviation and indicated that ARIMA and single exponential methods were the most effective in forecasting wholesale prices.

In 2006, Menon *et.al.* utilized an ARIMA (2,1,0) model to predict the prices of cardamom. This was achieved by analysing monthly costs spanning from January 1985 to December 2005, with the assistance of ACF and PACF to identify the model's parameters. The price series underwent rigorous testing to ensure stationarity, independence of residuals, and significance of autocorrelations. Subsequently, short-term price forecasting was conducted for a period of five months, from August 2005 to December 2005.

Bharathi (2009) analysed price behaviour of mulberry silk cocoon in Ramnagar and Siddlaghatta market. The secondary data on monthly arrivals and prices of cocoons were used for the study period from 1998 to 2008 from the respective markets. She used the Box-Jenkins ARIMA model to forecast the monthly arrivals and prices of mulberry silk cocoons. SARIMA$(1,1,3)(1,1,1)_{12}$ and SARIMA$(0,1,0)(1,1,1)_{12}$ were identified as the best model with minimum AIC values for the monthly arrivals and prices of cocoon in Ramnagar market whereas in Siddlaghatta market, SARIMA$(2,1,1)(1,1,1)_{12}$ and SARIMA$(0,1,0)(1,1,1)_{12}$ were the suitable models with minimum AIC values for the monthly arrivals and prices of cocoon.

Considering the conclusions reached by Kumar *et al.* (2011), the optimal model for predicting the price of potatoes in the Bangalore market was determined to be ARIMA $(0,1,0)$ $(0,1,1)_{12}$, based on monthly price data spanning from April 1999 to March 2008. The model yielded a MAPE of 18.28 per cent and AIC value of 1198.85.

The ACF and PACF of residuals were found to be lies within the standard interval revealed that, there is no autocorrelation among the residuals and Ljung-Box 'Q' statistic were found to be insignificant showing white noise of series. Thus, the model was found adequate.

Chaudhary and Tingre (2014) have identified ARIMA (0,1,0) as the optimal model for predicting the price of green gram, with MAPE of 6.01 per cent and Bayesian Information Criterion (BIC) value of 1989. This conclusion was drawn based on the monthly average price data from January 2001 to September 2012, obtained from the Ankola market. Model parameters were analysed using the Statistical Package for Social Sciences (SPSS).

In 2014, Krishnakutty made an endeavour to employ various ARIMA models to predict the prevailing yearly prices of teak in girth classes 1, 2, and 3 in the state of Kerala. He identified the prices for the expenses of girth classes 1, 2, and 3 using ARIMA(1,2,1) and ARIMA(0,2,2), respectively, with a confidence level of 95 per cent.

Tadjini *et al*. (2014) conducted a study to predict the yearly prices of wood-based panels in Iran spanning from 1986 to 2009. The authors employed various forecasting techniques, including ARIMA, Holt's Winter, and exponential methods. Among these methods, the ARIMA (2,1,1) model was determined to be the most effective, exhibiting a lower root mean square error for the Veenar wood panel. The model parameters were estimated using MINITAB software.

Naveena *et.al* (2014) made an attempt to study the best model to forecast the coconut production in India for time series data from 1951-2012.They revealed that ARIMA (1, 1, 1) was the best fit with minimum root mean square error value of 907.86. The ACF and PACF of residuals were found to be within the standard interval, thus revealing that there is no autocorrelation among the residuals. Ljung-Box 'Q' statistic was performed and found to be non-significant indicating adequacy of the model selected.

Jalikatti and Patil (2015) endeavoured to predict the monthly prices of onions in the Hubli district of Karnataka from 1996-97 to 2010-11. The analysis indicated a consistent rise in both arrivals and prices in chosen market. The monthly seasonal indices for onion arrivals were found to be elevated shortly after the harvest, while the price seasonal index was low during the harvest season and high during the lean season. They determined that the ARIMA (1,1,1) $(2,1,1)_{12}$ model was suitable for the series, as evidenced by the non-significant AIC and Q-statistic. Subsequently, they made projections for the following two years.

Indraji (2016) used prices of copra, coconut oil, and coconuts to calculate seasonal variations. She also proposed appropriate forecast models for the markets of Alappuzha, Kochi, and Kozhikode. The ARIMA (0,1,1) model was thought to be appropriate for predicting the price of coconuts at the Alappuzha marketplace. Also, SARIMA $(1,1,1)(1,0,1)_{12}$ models were thought to be appropriate for the price of coconut oil in Kozhikode market.

Guha and Bandyopaday (2016) forecasted future gold prices in India based on monthly data from November 2003 to December 2014 to mitigate the risk in purchases of gold. The value of Durbin-Watson (DW) was found to be 0.091 for the sample data of the gold price which indicated that the data is suitable for time series analysis. The results described that ARIMA (1,1,1) was the best model to forecast gold prices compared to other models with the value of R-square as 0.993, minimum RMSE of 719.18 and MAPE value of 3.245.

Deshmukh *et al.* (2016) stated that ARIMA (1,1,1) was the appropriate model for forecasting the milk production when compared to the Vector autoregression model based on the secondary data collected from Food and Agriculture Organization Corporate Statistical Database (FAOSTAT) and National Dairy Development Board (NDDB) (1991 to 2012). Stationarity of the data was checked with ACF and PACF, after confirming the stationarity, ARIMA and Vector Autoregression (VAR) models were used. The models were tested for reliability using R square, RMSE, MAPE, Schwartz Bayesian Criteria (SBC), and AIC.

In order to understand the components of time series data on pricing, to create statistical models for price volatility, and to investigate the integration between international and Indian tea prices, Joy (2021) did a study on this aspect. In order to predict tea prices in North and South India from January 2021 to April 2021, price forecasting models such as the exponential smoothing model and ARIMA model were fitted. The best forecast model for tea prices in North India was found to be SARIMA $(0,1,3)(0,1,1)_{12}$, whereas SARIMA$(0,1,1)(1,0,1)_{12}$ was chosen for tea prices in South India.

2.4 Price volatility

Kuruvila *et al.* (2012) utilized the GARCH model to assess the volatility levels in various time periods across both international and domestic markets. Their findings indicated that, during the post-WTO period, the monthly nominal prices of pepper, cardamom, tea, and coffee in the Indian market demonstrated increased volatility. Although international markets also exhibited high volatility, it was notably lower when compared to the volatility observed in domestic markets.

In their study, Yang *et al.* (2001) investigated the impact of agricultural liberalization policies on the volatility of agricultural commodity prices, employing Generalized Autoregressive Conditional Heteroscedasticity (GARCH) models. The findings of the research suggested that the implementation of the liberalization policy led to heightened price volatility for three key grain commodities—corn, soybeans, and wheat. However, oats experienced minimal change in volatility, while cotton witnessed a decrease in price volatility.

In 2015, Bhavani and colleagues investigated the volatility of prices in the domestic chilli markets by employing ARCH and GARCH models over the timeframe spanning from 1997 to 2011. The analysis revealed a consistent fluctuation in all markets, with Nagpur market exhibiting particularly pronounced and heightened volatility.

Dudhat (2017) conducted an analysis of the price volatility of groundnut in significant domestic markets located in Andhra Pradesh, Gujarat, and Tamil Nadu, utilizing data spanning from 1996 to 2016. The examination of price volatility involved the application of the GARCH model, with the Gaussian GARCH model identified as the most effective. The study revealed a greater persistence of volatility in the Rajkot market in Gujarat compared to the other two markets.

Cermak (2017) conducted a research endeavour aiming to identify and predict the volatility of wheat prices. The study utilized weekly time series data spanning from 2005 to 2015, employing stochastic models of conditional heteroskedasticity. The results indicated that the GARCH (1,1) model emerged as the suitable model for capturing the volatility of wheat prices.

2.5 Artificial Neural Networks for price forecasting

Huang *et al.* (2004) used ANN to forecast flows in Apalachicola River Using long term observations of rainfall and river flow during 1939-2000. The model used a feed-forward, backpropagation network structure with an optimized conjugated training algorithm. The correlation coefficients for daily, monthly, quarterly, and yearly flow forecasting between predictions and actual observations are 0.98, 0.95, 0.91, and 0.83, respectively. When comparing the forecasted flow rates from an ANN model to those from a conventional ARIMA forecasting model, the findings demonstrated that the ANN model outperforms the ARIMA model in accurately predicting river flow.

Ajjan *et al.* (2009 a) focused on the development and comparison of ANN and ARIMA models for forecasting the price of Poovan banana in Trichy, Tamil Nadu. Through their analysis, the researchers found that the ANN model outperformed the ARIMA model, particularly in terms of forecasting accuracy. This determination was based on the calculation of MAPE, where the ANN model exhibited a significantly lower MAPE value compared to the ARIMA model with parameters (1,0,1). The superior performance of the ANN model suggests its efficacy in capturing and modeling the complex patterns and relationships inherent in the price dynamics of Poovan banana in Trichy.

Sreekanth *et al.* (2009) applied the artificial neural network (ANN) model, i.e. standard feed-forward neural network trained with Levenberg–Marquardt algorithm for forecasting groundwater level at Maheshwaram watershed, Hyderabad, India. The model was found best fit as its RMSE was 4.50 and R^2 was 0.93.

Gan-qiong *et al*. (2010) undertook the task of predicting wholesale prices of tomatoes in China at daily, weekly, and monthly intervals, employing both ARIMA and ANN models. The data spanned from 1996 to 2010. Notably, the research findings highlighted a correlation between the accuracy of predictions and the length of the time interval, with shorter intervals yielding higher precision. Daily price forecasting emerged as the most accurate, followed by weekly predictions. The study revealed that the ANN model excelled particularly in forecasting one cycle ahead, indicating its suitability for short-term predictions. Impressively, the ANN model achieved accuracy rates exceeding 80% for all three forecasting types, with daily price forecasting surpassing an exceptional accuracy threshold of 90%. This underscores the efficacy of the ANN model, especially in capturing the intricate patterns inherent in short-term fluctuations of tomato prices in the Chinese wholesale market.

Jha and Sinha (2013) reported a simple way of building short–term memory into the structure of a neural network is through the use of time delay, which can be implemented at the input layer of the neural network. The TDNN model utilized in the study, showcases its efficacy in agricultural price forecasting by outperforming the ARIMA model in terms of forecast accuracy measures such as RMSE and Mean Absolute Deviation for longer forecast horizons of 6 and 12 months, while the ARIMA model excels for shorter horizons of 1 and 3 months. The result demonstrated the superiority of ANN over linear model methodology using monthly wholesale prices series of soybean and rapeseed-mustard.

Choudhary *et al.* (2019) addresses the challenges of forecasting agricultural commodity prices, particularly focusing on potato prices in the Delhi market. The study highlights the limitations of traditional statistical models, such as the ARIMA model, due to their pre-assumed linear relationships, and proposes the use of ANNs as a non-linear, nonparametric, and data-driven alternative. By employing empirical mode decomposition (EMD) and ANN models, the research demonstrated the effectiveness of the ensemble model in outperforming single models in terms of forecast accuracy, as evidenced by root mean square error and directional prediction statistics. The study provided valuable insights into the application of advanced forecasting techniques in the agricultural domain, offering potential benefits for farmers, policy planners, and agro-based industries.

Naveena *et al.* (2017) provides a comprehensive analysis of the Indian coffee market and the application of advanced forecasting models to predict coffee prices. The study covers the period from January 1995 to February 2016, using monthly wholesale price data for Robusta coffee seeds. The authors successfully employed an ARIMA model and a neural network architecture to forecast future prices, addressing the complexity and variation of Robusta coffee prices. The findings indicate that the hybrid ARIMA-ANN model outperformed individual ARIMA and ANN models, demonstrating its superiority in forecasting Indian Robusta coffee prices as it had lowest MAPE value of 1.16 per cent.

Rathod *et al.* (2018) conducted an experiment to forecast the oilseed production in India using TDNN, non-linear support vector regression (NLSVR) and ARIMA model. Their findings revealed that the artificial intelligence techniques, TDNN with MAPE value of 8.09 per cent and NLSVR with MAPE value of 3.85 per cent, outperformed the traditional ARIMA (MAPE value of 11.06 per cent) model in forecasting oilseed production.

Kurumatani (2020) proposed a time series forecasting method for the future prices of agricultural products in his study. The author implemented two methods for time series forecasting based on RNN, one of which is called time-alignment of time point forecast (TATP), and another one is called direct future time series forecast (DFTS). He predicted that, in addition to the error rate, the criterion for conservation of the statistical characteristics of the probability distribution function from the original past time series to the future time series in the forecasted future is also important. The study reported that after intensive training, TATP of long short-term memory (LSTM) shows superior performance not only in accuracy, but also the conservation of the statistical characteristics of the probability distribution function from the original past time series compared to TATP of other RNNs.

The dual input attention long short-term memory model (DIA-LSTM) was proposed by Gu *et al.* (2022) as an effective way to anticipate agricultural commodities prices. By using the DIA-LSTM model to analyse the monthly prices of radish and cabbage in the South Korean market, they have assessed the model. The DIA-LSTM is capable of identifying the feature correlation and temporal correlations of multivariate time series input data since it is trained using a variety of variables that influence the price of agricultural commodities, including trading volume and meteorological data.

2.6 Comparison of forecasting models

Gutierrez *et al.* (2004) conducted a study to compare different statistical models and artificial intelligence techniques for ammonia concentration forecasting in an eel (Anguilla Anguilla L.) intensive rearing system which is located in Spain. They applied multiple linear regression, univariate time series models (exponential smoothing and ARIMA models) and neural networks (ANNs) to forecast the daily average ammonia concentration in rearing tanks with water recirculation. The result stated that the nonlinear ANN model provided a better prediction of daily average ammonia concentration than multiple linear regression and univariate time series analysis.

Mani *et al.* (2005) compared flood prediction models by developing ARIMA and artificial neural networks (ANNs) based on the availability of historical daily stream flow data over the period of 29 years from 1972 to 2000 of River Godavari and its tributaries in India. The result obtained revealed that the ANN model outperformed the ARIMA model to predict the hydrological modeling of flood. The ANN model output was compared based on the correlation coefficient (r), Absolute Average Relative Error (AARE), RMSE and Nash-Sutcliffe Coefficient of Efficiency (CoE).

Jyothi (2011) used various traditional models of forecasting like single, double and triple exponential smoothing models, moving average and ARIMA models to evaluate the price behaviour of turmeric in India based on the prices of turmeric from 1980-81 to 2008-09 in Nizamabad and Erode markets. Different models were compared, out of which double smoothing exponential smoothing model was appropriate for Nizamabad and Winters multiplicative model was the better for Erode market.

Kumar *et al.* (2011) evaluated the performance of Holt Winter's exponential smoothing model and seasonal ARIMA model for forecasting price of onion in Bangalore market on the basis of monthly average price of onion from April, 1999 to March, 2010. They found ARIMA $(1,1,0)(0,1,1)_{12}$ to be the best fit with MAPE of 16.14 per cent.

Reeja (2011) applied Moving Average model, single and double exponential smoothing models, ARIMA and ANN models for forecasting price of natural rubber in

Kottayam and Bangkok markets. They revealed ANN as the best model in Kottayam market and SARIMA(0,1,0)(1,0,1)$_{12}$ in Bangkok market.

Adebiyi *et al.* (2014) evaluated the forecasting performance of ARIMA and artificial neural network model with published stock data obtained from New York Stock Exchange. The empirical results revealed the superiority of neural network model over ARIMA model.

In 2015, Sangsefidi and colleagues employed ARIMA and GARCH models to predict the prices of various agricultural products, such as potato, onion, tomato, and veal. A comparison was made between the outcomes of the ARIMA model and ARCH family models. The findings revealed that the ARIMA method produced estimates with a lower relative error compared to the ARCH model. In essence, the ARIMA model exhibited superior performance over the ARCH model.

Sharma and Burark (2015) conducted an exhaustive comparison of ARIMA and ANN models in order to know the best model for forecasting moth bean. The various forms of ARIMA and ANN were employed to predict the future prices of moth bean in Churu market. Their study revealed that the ARIMA (0,1,2) model was the most suitable for forecasting the prices of moth bean.

In 2016, Naveena conducted research focused on forecasting the price and export of Indian coffee using different time series models. Within the study, forecasting models, including Exponential Smoothing, ARIMA, GARCH and ANN were constructed for the analysis of both price and export trends. The assessment of the different forecasting models relied on measures such as RMSE and MAPE. The result indicated that the ARIMA (0,1,1)(0,0,0) model proved to be the most effective for forecasting Indian Arabica price. Additionally, for Robusta coffee price, the AR(3)-GARCH (3,1) models demonstrated superior performance. In the context of Indian coffee export, the Artificial Neural Network (ANN) model outperformed the other models.

Vijay and Mishra (2018) demonstrated the efficiency of the ANN model over the ARIMA model by conducting a study on the area and production of Pearl millet (bajra) from 1955-56 to 2014-15 in Karnataka based on various evaluation metrics, including

RMSE, MAPE, and MSE. This suggests that the non-linear machine learning techniques employed in the ANN model can effectively capture the heterogeneous trend in the data set, leading to improved forecasting accuracy. The findings highlight the superior performance of the ANN model compared to the ARIMA model in terms of forecasting pearl millet production, demonstrating the potential of non-linear machine learning techniques, particularly artificial neural networks, in improving the accuracy and effectiveness of forecasting agricultural crop production.

Chi, Y. N, and Chi, O. (2021) conducted a study to predict the monthly global price of bananas using the data from January 1990 to November 2020 in USA. Empirically, the results showed that the Multi-Layer Perceptron (MLP) neural network model performed better compared to ARIMA(4,1,2)(1,0,1) due to its smaller MSE value. The MLP neural network model can provide useful information in the decision-making process.

Mahto *et al*. (2021) in their article named "Short-Term Forecasting of Agriculture Commodities in Context of Indian Market for Sustainable Agriculture by Using the Artificial Neural Network", compared ANN model and ARIMA models of the time series data for soybean seeds for the period of five years (Jan 2014–Dec 2018), in Akola district market of Maharashtra, and sunflower seeds for the period of six years (Jan 2011–Dec 2016), in Kadari district market of Andhra Pradesh. It was observed that the ANN is a better forecasting model than the ARIMA model by considering the two forecasting performance parameters MAPE and RMSE.

Rathoda *et al. (*2022) studied the rice prices under COVID-19 lockdown using time series models such as ARIMA, ANN and Extreme Machine Learning (ELM) and to forecast price changes during similar crisis scenarios with the best model selected among them. The results showed that rice prices increased by Rs.0.92/kg during the COVID-19 lockdown. The ELM intervention model provided better results compared to other models due to its ability to detect nonlinear patterns in time series data.

Kumari *et al.* (2022) compared different statistical models to predict area, production and yield of citrus in Gujarat state. The study used secondary data on citrus production from 1991-92 to 2016-17 and analysed it using ARIMA and Exponential smoothing models. The results showed that the ARIMA model outperformed the exponential smoothing model in predicting the area, production, and yield of citrus. The study suggested that the government should adopt the ARIMA model to make policy decisions regarding citrus production.

Kumari *et al.* (2023) conducted a study to analyse and compare the efficiency of different traditional models like ARIMA, SARIMA, ARCH and GARCH to the deep artificial intelligence techniques like ANN and RNN in forecasting the prices of banana in Gujarat on the time series data from January 2009 to December 2019. Empirical result showed that RNN was the best fitted model among all other models of prediction due to less error accuracy measures. The structure of the RNN model included an input layer with three input values representing the previous lag values of the price, a hidden layer with ten neurons, and an output layer with a single neuron. In order to introduce non-linearity and capture complex features, the Rectified Linear Unit (ReLU) activation function was applied in the hidden layer. Meanwhile, a linear activation function was utilized in the output layer to enable the direct prediction of continuous values.

Materials and Methods

3.MATERIALS AND METHODS

The data used and the statistical techniques adopted for the study "A comparative analysis of price forecasting models for black pepper" are elaborated in this chapter in the following headings.

3.1 Data

Monthly and weekly average price of garbled and ungarbled black pepper for Kochi market of Kerala from 2000 to 2020 with 252 and 1094 data points respectively formed the database for the study which was obtained from Spices India and Spices Market Weekly, publications of Spices Board, Kerala.

Prior to delving into the statistical techniques employed in the study, essential terms are elucidated in section 3.2.

3.2 Important terminologies

Time series (TS):

A TS is a dynamic situation which reveals the good deal of variations overtime. It refers to a systematically arranged set of data points for a specific variable, recorded at regular time intervals, which can span weeks, months, quarters, years, and so on.

Mathematically, TS can be defined as the functional relationship $Y_t = f(t)$ where Y_t is the value of the variable under study at time t.

It can be analysed to study the different components in the time series and also to forecast the series.

Lag:

Lag is the interval of time between an observation and an earlier observation.

As for instance, Y_{t-k} , where Y_t is the value of time series at time t, lags Y_t by k periods.

Lead:

The time elapsed between one observation and a subsequent observation is referred to as lead. $Y_{t\pm k}$ thus has a k-period lead over Y_t .

Moving Average :

The mean of k successive observations is the moving average of order k.

MA of order $k = \frac{\sum_{t=1}^{k} Y_t}{k}$, when k =12, MA of order 12 is obtained.

Autocorrelation:

It refers to the correlation between time series values observed at different time points. Let Y_t be the time series value at time t, then autocorrelation at lag k, is given by

$$r_k = \frac{\sum_{t=1}^{n-k}(Y_t - \bar{Y})(Y_{t\pm k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2}$$

Where, k denotes the length of the time lag

n is the number of observations

Y_t is the value of the variable at time t

$\bar{Y}$ is the mean of all the data

$\frac{1}{n}\sum_{t=k+1}^{n}(Y_t - \bar{Y})(Y_{t\pm k} - \bar{Y})$ is the auto-covariance term

Autocorrelation Function (ACF):

It is the correlation patterns observed between a time series and its past values at different time lags, such as 1, 2, 3, and so on.

Correlogram:

It is a graphical representation created by plotting autocorrelation values against specific time lags. This visual tool is crucial for swiftly interpreting autocorrelation patterns. Correlograms are standard tools used in time series analysis. They play a key role in identifying seasonal patterns within a time series and assessing the data's stationarity. This information is vital for selecting the good TS forecasting model.

Partial Autocorrelation:

It is employed to measure the degree of association between $\mathrm{Y_t}$ and $\mathrm{Y_{t-k}}$, after accounting for the effects of other time lags and denoted by α_{k}. The partial autocorrelation coefficient of order k can be calculated by regressing Y_t against $Y_{t-1}, \ldots, Y_{t-k}$.

$$Y_t = b_0 + b_1 Y_{t-1} + b_2 Y_{t-2} + \cdot \cdot \cdot + b_k Y_{t-k} + \epsilon_t$$

PACF(k) measures the "direct" impact of Y_{t-k} on Y_t.

Partial Autocorrelation Function (PACF):

The autocorrelation pattern for 1,2,3, …. lags is known as PACF.

PACF plot:

The plot derived by graphing partial autocorrelations against their respective lags is referred to as the PACF plot.

Stationary Time series:

A TS is considered stationary when its statistical attributes, like mean and variance, stay constant over time. In simpler terms, if the process generating a time series maintains a steady mean and variance irrespective of the observed time span, the time series is categorized as stationary.

Time Plot:

In order to effectively explore a time series, it is crucial to visually represent the data. A time plot is created by graphing time series data against their respective time points. This visualization method aids in comprehending the time series pattern and detecting trends, seasonal patterns, and other features within the dataset.

3.3 Decomposition of black pepper prices

In the analysis of a time series related to pepper prices, the factors influencing the prices can be categorized into four distinct components. These components help break down the complexities of the time series data and provide a clearer understanding (Croxton *et al.*, 1979). The four components are as follows:

1. Trend: The trend component represents the long-term movement or direction in the black pepper prices. It captures the underlying pattern in the data, indicating whether prices are generally rising, falling, or stable over an extended period.

2. Seasonal Variation: Seasonal variation accounts for the regular, repeating patterns in black pepper prices that occur within specific time frames, such as months, seasons, or years. This component helps identify consistent price fluctuations that follow a predictable pattern over these recurring periods.

3. Cyclic Variation: Cyclic variation refers to the rhythmic, non-seasonal ups and downs in black pepper prices that do not have a fixed period. Unlike seasonal patterns, cyclic variations are more extended and irregular, often influenced by economic cycles, market trends, or other long-term factors.

4. Irregular Variation: Irregular variation, also known as residual or random variation, represents the erratic and unpredictable fluctuations in black pepper prices that cannot be attributed to trends, seasonal patterns, or cyclic movements. These irregularities could be caused by unforeseen events, market shocks, or other random factors affecting prices unexpectedly.

Understanding and isolating these four components are essential for a comprehensive analysis of the black pepper price time series data, as they provide valuable insights into the various factors at play, enabling more accurate predictions and informed decision-making.

3.3.1 Trend analysis

The trend for monthly and weekly prices of garbled and ungarbled black pepper for Kochi market were studied by fitting suitable model. The following models were fitted:

Linear trend: $Y_t = a + b_t + e_t$

Quadratic trend: $Y_t = a + b_t + c_t^2 + e_t$

Exponential trend: $Y_t = a\,b^t$

$t = 2000\ to\ 2020$ and $e_t \sim N\ (0, \sigma^2)$

From the different models, suitable model was selected based on Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE).

MAPE is given by,

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_t - \widehat{Y_t}|}{Y_t} * 100$$

Where,

Y_t– Actual price at time t

$\widehat{Y_t}$- Fitted price

n- Number of observations

RMSE is given by,

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} e_t^2}$$

Where,

$e_t = Y_t - \widehat{Y_t}$, e_t is the difference between actual and fitted price

$n =$ Number of observations

3.3.2 Decomposition model

For decomposition of the black pepper price data into four components, a multiplicative model was assumed as outlined by Croxton *et al.* (1979) and Spiegel (1992). Let Y_t be the price at time t. Then the multiplicative model is,

$$Y_t = T_t * S_t * C_t * I_t$$

$T_t, S_t, C_t\ and\ I_t$ are the components of TS at time t , where,

T_t: Trend

S_t: Seasonal variation

C_t: Cyclic variation

I_t: Irregular variation

t: January 2000 to December 2020

All the four components were estimated for price of garbled and ungarbled black pepper in Kochi market using R software and represented graphically.

3.3.3 Calculation of seasonal indices

Calculation of seasonal indices was carried out for the 12 months (January to December) using both monthly and weekly price data of black pepper (both garbled and ungarbled) spanning from January 2000 to December 2020. This analysis aimed to grasp the seasonal patterns in price behaviour over the specified period.

Assuming multiplicative model, $Y_t = T_t * S_t * C_t * I_t$, seasonal indices were calculated using ratio to moving average method.

The steps involved in ratio to moving average method are given below:

- Calculate the centred 12 month moving average (CMA) of time series data, Y_t .

  CMA values will give the estimates of combined effect of trend and cyclic variation.

  i.e., $CMA = T_t * C_t$

- Express the original data as the percentage of the CMA values.
- This percentage will represent the seasonal (S_t) and irregular components (I_t).

  i.e., $\frac{Y_t}{CMA} * 100 = S_t * I_t$

- By averaging the above percentages over years, the irregular components will get eliminated. The resultant value will be preliminary seasonal indices, S.

- The sum of preliminary seasonal indices (S) may not be equal to 1200 and so we need to adjust the value by multiplying throughout by the factor $\frac{1200}{s}$. The result will give the seasonal indices from January to December.

Seasonal indices of the black pepper for Kochi market were plotted against corresponding months. The resultant graph is called seasonal plot.

3.4. Forecasting models for prices of black pepper

In TS forecasting, the goal is predicting how a sequence of observations will unfold in the future. This can be achieved by examining previous observations of the identical variable and constructing a model that characterizes the inherent pattern of the TS. By using known data (Y_t), model fit values ($\widehat{Y_t}$) are calculated, allowing the computation of errors ($e_t = Y_t - \widehat{Y_t}$). Subsequently, forecasted values ($F_{t+1}, F_{t+2}, F_{t+3}$ ...) are generated. When the new observations are available, forecasted errors ($e_{t+1}, e_{t+2}, e_{t+3}$ ...) can be used for evaluation.

Exponential smoothing (Single, Holt's linear (Double), Holt-Winters' additive and multiplicative exponential smoothing) models, ARIMA models, SARIMA models, ANN models and RNN models were fitted to forecast the prices of black pepper and the best model was selected. The five models are explained from section 3.4.1. to 3.4.5.

3.4.1 Exponential smoothing models

Exponential smoothing is a specific type of moving average technique employed in time series analysis to generate smoothed data for forecasting. This method assigns exponentially decreasing weights to past observations in order to predict future values. The selection of one or more smoothing parameters is crucial, as it directly influences the weights assigned to the values. Various smoothing models are available for TS forecasting, and the choice of these models dictates the explicit determination of smoothing parameters, influencing the overall accuracy of the forecasts. The different exponential smoothing models used were,

1. Single Exponential Smoothing model (SES)

2. Double Exponential Smoothing model (DES)

3. Holt-Winters' exponential smoothing model

3.4.1.1 Single Exponential Smoothing model (SES)

The single exponential forecasting method involves taking the forecast from the preceding period and adjusting it using the forecast error, i.e. forecast for subsequent period is: $F_{t+1} = F_t + \alpha(Y_t - F_t)$; where α is a constant ranging between 0 and 1. A significant α value (e.g., 0.9) smooths the forecast very little, while a small α value (e.g., 0.1) smooths the forecast considerably. Exponential smoothing is the same as forecasting from the most recent observation when $\alpha = 1$. A grid of values for α, such as $\alpha = 0.1, 0.2, \ldots, 0.9$, was used to pick the values that resulted in the lowest MAPE. Hence, the updated forecast is essentially the previous forecast with an adjustment accounting for the error in the last forecast. This method is applied for making short-term forecasts in a time series devoid of trend or seasonality.

3.4.1.2 Double Exponential Smoothing model (DES)

Holt (1957) expanded single exponential smoothing into linear exponential smoothing to enable the forecasting of data exhibiting trends but lacking seasonality. The prediction in Holt's linear exponential smoothing is determined by employing two smoothing constants, α and β, both ranging between 0 and 1. The forecasting process involves three equations, outlined as follows:

Level: $L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + b_{t-1})$

Trend: $b_t = \beta (L_t - L_{t-1}) + (1 - \beta) b_{t-1}$

Forecast: $F_{t+m} = L_t + b_{tm}$

where,

L_t: level of the time series at time t,

b_t: estimate of trend (slope) of the time series at time t

F_{t+m}: forecast for m periods ahead of t

The combination of α and β which provides the least MAPE was selected.

3.4.1.3 Holt-Winters' Exponential Smoothing model

Holt's method was extended by 'Winter (1960) to capture 'seasonality' directly. Three smoothing equations—one for level, one for trend, and one for seasonality—are the foundation of the Holt-Winters approach. Depending on whether seasonality is modelled in a multiplicative or additive manner, Holt-Winters' technique has two models: the Multiplicative Seasonal (HWMS) model and the Additive Seasonal (HWAS) model. For the present study, model used was HWMS as the seasonality is modelled in a multiplicative manner.

The equations for HWMS model:

Level: $L_t = \alpha \frac{y_t}{S_{t-12}} + (1 - \alpha)(L_{t-1} + b_{t-1})$

Trend: $b_t = \beta (L_t - L_{t-1}) + (1 - \beta) b_{t-1}$

Seasonality: $S_t = \gamma \frac{x_t}{Lt} + (1 - \gamma) S_{t-s}$

Forecast: $F_{t+m} = (L_t + b_{tm}) S_{t-s+m}$

where, $L_t$is the level of the time series at time t, b_t is the estimate of trend (slope) of the time series at time t. S_t is the seasonal component at time t. For monthly data, $s = 12$. F_{t+m} is the forecast for m periods ahead of t. α, β and γ are smoothing constants, each taking values between 0 and 1. The combination of α, β and γ which yields minimum value for MAPE was chosen.

From among different exponential smoothing models fitted for the prices, the best model was chosen based on closeness of actual and fitted price plots, MAPE and RMSE. Exponential smoothing models for the black pepper prices in Kochi market was fitted using the R software.

3.4.2 Auto Regressive Integrated Moving Average (ARIMA) model

In mid 1970's, George Box and Gwilym Jenkins created and promoted ARIMA technique for time series forecasting. Thus, it is likewise called Box-Jenkins ARIMA technique. The model is based on the assumption that the time series is stationary. Box and Jenkins introduced a pragmatic three-stage approach for identifying a suitable model, where a good model entails the minimum number of estimated parameters required to effectively capture the patterns in the provided data. The process involves distinct steps, including identification, estimation, diagnostics, and forecasting.

Model Identification

Examining the time series for stationarity is the first step in identifying ARIMA modeling. If the mean and variance of a time series remain constant, it is referred to as stationary. Time series plots, ACF and PACF plots can all be used to detect the presence of non-stationarity. Differencing is a technique employed to transform a non-stationary time series into a stationary one. When dealing with time series data containing seasonality and exhibiting non-stationarity, applying seasonal differencing to the data can render it stationary.

Unit root tests are statistical procedures used to assess stationarity of time series. The Augmented Dickey Fuller (ADF) test is the unit root test that is most frequently utilized. This test involves estimating the following regression equation:

$$Y_t' = \phi\, Y_{t-1} + b_1 Y_{t-1}' + b_2 Y_{t-1}' + \cdots + b_p Y_{t-1}'$$

where, $Y_t' = (Y_t - Y_{t-1})$

By using the method of least squares, the value of ϕ is calculated and tested for deviation from unity : $H_0: \phi = 1$

$$H_1: \phi < 1$$

$$t_{\phi=1} = \frac{\hat{\phi} - 1}{\mathrm{SE}\,(\hat{\phi})}$$

where, $\hat{\phi}$ is the least square estimate of ϕ.

If $\hat{\phi}$ is negative and significant, the TS is considered stationary, if $\phi = 1$, TS is non- stationary.

The next stage of the identification process is to determine the starting values for the orders of the seasonal and non-seasonal parameters, p, q and P, Q. They could be identified by looking for the significant autocorrelation and partial autocorrelation coefficients.

B. Estimation and testing

In the identification phase, one or more models that provided statistically adequate representations of the TS data were tentatively chosen. After the selection of preliminary models, the values for p and q were estimated. Accurate estimates of the model parameters were obtained using the R software through least squares analysis, as outlined by Box and Jenkins (1970).

Diagnostics of the model

To assess the quality of the fitted model, ACF and PACF plots of residuals, along with the Ljung-Box Q statistic for residuals, were employed. (Manoj and Madhu, 2014).

(a) Using criteria such as Schwarz-Bayesian Information Criteria (SBC) or Akaike Information Criteria (AIC), the best model among the various models could be chosen.

AIC is provided as ; $AIC = -2logL + 2m$

where, $m = p + q + P + Q$, L is the likelihood function.

$-2 \, log \, L$ is approximately equal to $\{2n(1 + log2\pi)\} + nlog\sigma^2$; σ^2 is the model MSE,

AIC is rewritten as

$$AIC = \{n(1 + log2\pi)\} + nlog\sigma^2 + 2m$$

$$SBC = log\sigma^2 + (m \, logn)/\, n$$

Since the first term is typically omitted as it is a constant. The model with the lowest AIC/SBC value will be chosen, and its residuals will resemble white noise.

(b) Plot of residual ACF

Upon fitting the suitable ARIMA model, the goodness of fit was assessed by examining the autocorrelation function (ACF) of residuals. A model is considered a good fit when the majority of the sample autocorrelation coefficients of the residuals fall within the bounds of $\frac{\pm 1.96}{\sqrt{n}}$, where n represents the number of observations forming the basis of the model. This observation indicates that the residuals resemble white noise, confirming the adequacy of the model fit.

(c) Portmanteau test: Ljung-Box test

Performing diagnostic checks is essential to assess the appropriateness of the model and suggest any required adjustments after fitting the initial model to the data. The significance of autocorrelations in residuals can be tested using the Portmanteau test as part of the diagnostic evaluation. Ljung-Box Q^* statistic is given by

$$Q^* = n(n+2)\sum_{k=1}^{h} \frac{1}{n-k} r_k^2$$

where 'h' is the maximum lag being considered and 'n' is the number of observations in the series. If the residuals are white noise, the statistic Q^* has a chi-square (χ^2) distribution with $(h - m)$ degrees of freedom where 'm' is the number of parameters fitted in the model. The series is not white noise when Q^* is greater than table value of chi-square at $(h - m)$ degrees of freedom.

3.4.2.1 Representation of ARIMA

After identifying the ARIMA model, the model parameters are estimated, and essential diagnostic procedures are undertaken. There are two categories: Seasonal ARIMA (SARIMA) and non-seasonal ARIMA.

(a) Non seasonal ARIMA model

ARIMA (p, d, q) is the standard notation for the non-seasonal ARIMA model, where $p, d, and\ q$ stand for the orders of auto-regression, integration (differencing), and moving average, respectively. For example, given a TS process $\{Y_t\}$,

AR (p) model is given by

$$Y_t = \mu + \varphi_1 Y_{t-1} + \varphi_2 Y_{t-2} + \cdots + \varphi_p Y_{t-p} + \varepsilon_t$$

MA (q) model is given by

$$Y_t = \mu - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} + \varepsilon_t$$

A stationary ARMA (p, q) process is defined by the equation

$$Y_t = \mu + \varphi_1 Y_{t-1} + \varphi_2 Y_{t-2} + \cdots + \varphi_p Y_{t-p} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} + \varepsilon_t$$

i.e., $\left(\mathbf{1} - \sum_{i=1}^{p} \boldsymbol{\varphi}_i \, \boldsymbol{B}^i \right) \boldsymbol{Y}_t = \left(\mathbf{1} - \sum_{j=1}^{q} \boldsymbol{\theta}_j \, \boldsymbol{B}^j \right) \boldsymbol{\varepsilon}_t$

$Y_t, Y_{t-1} \ldots Y_{t-p}$ are the TS values at time $t, t-1, t-2 \ldots t-p$;

B is the backshift operator such that $B_i Y_t = Y_{t-i}$,

$\varepsilon_t, \varepsilon_{t-1}, \ldots, \varepsilon_{t-p}$'s are errors at time $t, t-1, t-2 \ldots t-p$; independently and normally distributed with zero mean and constant variance σ^2 .

If the TS is non-stationary, the ARIMA (p, d, q) model is given as

i.e., $\left(\mathbf{1} - \sum_{i=1}^{p} \boldsymbol{\varphi}_i \, \boldsymbol{B}^i \right) (\mathbf{1} - \boldsymbol{B})^{\boldsymbol{d}} \, \boldsymbol{Y}_t = \left(\mathbf{1} - \sum_{j=1}^{q} \boldsymbol{\theta}_j \, \boldsymbol{B}^j \right) \boldsymbol{\varepsilon}_t$

where, ϕ_i , $i = 1, 2 \ldots p$ are AR parameters

$\theta_j, j = 1, 2 \ldots q$ are MA parameters

$(1 - B)^{d} Y_t$ is the non-seasonal difference of order d on Y_t

(b) Seasonal ARIMA Model

When seasonality is present in the TS data, the seasonal ARIMA model is applied. In other words, a pattern recurs after a predetermined amount of time. For monthly data, significant autocorrelation and partial autocorrelation at lags of 12, 24, etc., can be used to detect the presence of seasonality. Seasonal differencing can be used to make non-stationary seasonal data stationary. That is, the difference between an observation and its corresponding observation from the preceding year.

$SARIMA\ (p, d, q)\ (P, D, Q)_s$ model is given by,

$$\left(1 - \sum_{i=1}^{p} \varphi_i\, B^i\right)\left(1 - \sum_{i=1}^{p} \varphi_i\, B^{is}\right)(1 - B)^d (1 - B^s)^D\, Y_t = \left(1 - \sum_{j=1}^{q} \theta_j\, B^{js}\right)\left(1 - \sum_{j=1}^{Q} \Theta_j\, B^s\right)\varepsilon_t$$

Non seasonal AR(p)

Seasonal AR(P)

Non seasonal difference

Seasonal difference

Non-seasonal MA(q)

Seasonal MA(Q)

where,

$\Phi_i\ , i = 1,2 \ldots p$ are the seasonal AR parameters

$\Theta_j, j = 1,2 \ldots Q$ are the seasonal MA parameters

d, D =order of non-seasonal, seasonal differencing respectively

$s =$ no. of seasons (Since the data was collected on a monthly basis, $s = 12$)

3.4.2.2 Forecasting

Forecasting is the process of projecting the future values of times series using the fitted model.

3.4.2.2.1 Forecast Accuracy Measures

The reliability of the ARIMA model is evaluated using forecast accuracy indicators.

If Y_t is the actual observation for time period t and F_t is the forecast for the same period, then the error is defined as

$$e_t = Y_t - F_t \ ;$$

F_t is estimated using data $Y_1, \ldots, Y_{t-1}$ and is a one step ahead forecast. Suppose there are n observations and n forecasts, then there will be n error terms, and the following statistical forecast accuracy measures can be used.

$$\text{MAE} = \frac{1}{n} \sum_{t_{i=1}}^{n} |e_t|$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{t_{i=1}}^{n} e_t^2}$$

$$\text{MAPE} = \frac{1}{n} \sum_{t_{i=1}}^{n} \frac{|Y_t - F_t|}{Y_t} * 100$$

The best suitable ARIMA/SARIMA model is selected for black pepper prices based on the agreement between the fitted and actual price plots, residual ACF and PACF plots MAPE and RMSE. The R software was used for fitting the SARIMA model.

3.4.3 ARCH family models

The fluctuation of commodity prices over a period of time refers to the term price volatility. As the study uses the time series data which consist of volatility, the ARCH family models can be employed. The entire process of fitting ARCH and GARCH models is conducted using the 'fgarch', 'tseries', and 'FinTS' packages within the R software. In order to examine the ARCH effect in the time series data, the Heteroscedasticity LM test or ARCH LM test was utilized.

Heteroscedasticity LM Test or ARCH LM Test

The Heteroscedasticity LM test (Tsay, 2005) is employed to assess the presence of volatility or the ARCH effect in the time-series data.

H_0: There is no ARCH effect.

Steps:

1) Run the regression of the model using Ordinary Least Squares (OLS) for the stationary series and collect residuals.

2) The residuals are squared, and then regress them on q own lags to test for ARCH of order q, i.e. run the regression

$$\hat{\varepsilon}_t^2 = \sum_{i=1}^{q} \gamma_i \, \hat{\varepsilon}_{t-q}^2 + \varepsilon_t$$

3) Obtain the statistic $N * R^2$, where N is the number of observations and R^2 is the coefficient of determination. This statistic follows a chi-squared distribution with q degrees of freedom.

4) If the value of the test statistic is less than the critical value from the χ^2 distribution, then accept the null hypothesis or if the value of p (with respect to chi-square) is greater than 0.05, then accept null hypothesis.

3.4.3.1 ARCH model

The Autoregressive Conditional Heteroscedasticity (ARCH) model is a significant advancement in the realm of time series modeling, particularly for capturing and understanding volatility dynamics. Introduced as one of the pioneering models, ARCH provides a framework to model conditional heteroscedasticity, where the variance of a time series is not constant but changes based on past errors. This model allows for a representation of volatility by considering the squares of past errors as a crucial factor influencing the current conditional variance leaving. the unconditional. variance constant. The ARCH(q) model for the series $\{\varepsilon_t\}$ is defined by specifying the conditional distribution of ε_t (error) given the information available up to time $t-1$.

ARCH(q) model is given by

$$\varepsilon_t \sim N(0, h_t) \qquad (1)$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \, \varepsilon_{t-i}^2 \qquad (2)$$

where, h_t is the conditional variance at time t , α_0 >0 and $\alpha_i \geq 0$, for all i and $\sum_{i=1}^{q} \alpha_i < 1$ are required to be satisfied to ensure non-negative and finite unconditional variance of stationary $\{\varepsilon_t\}$ series.

3.4.3.2 GARCH model

The ARCH model has certain limitations. Firstly, when the ARCH model's order is high, it necessitates the estimation of a large number of parameters. Secondly, the conditional variance of the ARCH(q) model exhibits a characteristic where the unconditional autocorrelation function of squared residuals, if it exists, decays very rapidly compared to typical observations unless the maximum lag q is large. To address these challenges, the GARCH model has been introduced. In the GARCH model, the conditional variance is also a linear function of its own lags. This model represents a weighted average of past squared residuals with declining weights that never reach zero completely. It provides parsimonious models that are easy to estimate and, even in its simplest form, has demonstrated surprisingly effective results in predicting conditional variances.

The GARCH (p, q) model for the series $\{ \varepsilon_t \}$ is given by

$$\varepsilon_t \sim N(0, h_t)$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \, \varepsilon_{t-i}{}^2 + \sum_{j=1}^{p} \beta_j \, \sigma_{t-j}^2 \qquad (3)$$

where, h_t is the conditional variance at time t ,

$\alpha_0 > 0$, $\alpha_i \geq 0$, $\beta_j \geq 0$, $\sum_{i=1}^{q} \alpha_i + \sum_{j=1}^{p} \beta_j < 1$,

where, $\alpha_i = 0$ for $i > q$ and $\beta_j = 0 \; for \; j > p$

GARCH models differentiate between the predictable and unpredictable components of prices and enable the variance of the unpredictable element to vary over time. The estimation of such time-varying conditional variances was accomplished by utilizing the GARCH (1,1) model, as proposed by Bollerslev (1986).

A. Identification of models

The commencement of time series analysis often involves creating a graphical representation of the time series. The primary step in the modeling process is to assess the stationarity of the series since estimation procedures are applicable only to stationary series. The presence of stationarity can be determined using the Augmented Dickey-Fuller (ADF) test. If the model is identified as non-stationary, achieving stationarity can be accomplished by differencing the series. During this step, it is essential to examine the volatility or ARCH effect in the time-series data using the ARCH LM test. In this test, the null hypothesis states the absence of the ARCH effect or volatility. If the p-value (with respect to chi-square) is less than 0.05, then it is permissible to apply ARCH family models for the stationary series; otherwise, such models cannot be employed. The minimum RMSE and MAPE are used to select the best model from the set of ARCH and GARCH models.

B. Estimation of the parameters

During the stage of identification, models are provisionally selected based on their apparent ability to statistically represent the available data. The Maximum Likelihood Estimation (MLE) method is then employed to estimate the parameters of the chosen model along with their standard errors.

C. Diagnostic checking of the model

Performing diagnostic checks is crucial to validate the adequacy of the selected model. If the model is considered statistically inadequate, the entire process of identification, estimation, and diagnostic checking is reiterated until a suitable model is identified. To assess the goodness of the fitted model, methods such as the Serial Correlation LM test and the Ljung-Box test for residuals can be employed.

Serial Correlation LM test

This test is similar to the Heteroscedasticity Lagrange's Multiplier test. However, in this case, the null hypothesis is that there is no serial correlation in residuals. If the p-value (with respect to the chi-square statistic) is greater than 0.05, then the null hypothesis is accepted.

D. Forecast Accuracy Measures

Several forecasting measures is used to determine the best volatility model using the MAPE and RMSE.

3.4.4 Artificial Neural Network

The biological neuron system serves as the inspiration for the Artificial Neural Network (ANN), a data processing model. It consists of a large number of interconnected processing units called neurons, which collaborate to solve problems. Operating in a non-linear manner, it processes data in parallel across the nodes. A group of connected input/output units, each with a distinct weight, make up a neural network. It executes the calculations through a learning process. The network learns in the learning phase by modifying the weights to anticipate the right class label of the provided inputs (Zhang *et. al*., 1998).

ANN is a mathematical model that aims to replicate the functionality and architecture of biological neural networks. An artificial neuron is the foundation of any artificial neural network; it is a straightforward mathematical model (function). Three fundamental sets of rules make up a model: activation, summation, and multiplication (Aggarwal, 2018). At the artificial neuron's input, the inputs are weighted, meaning that every input value is multiplied by a unique weight. All weighted inputs and bias are added by the artificial neuron's sum function in the central region. An activation function, also known as a transfer function, receives the sum of the previously weighted inputs and bias at the exit of an artificial neuron. The processing element are divided into two parts. The first part simply aggregates the weighted inputs; the second part, which is the transfer function—also called the activation function. An activation function serves as a critical element in the model, introducing non-linearity to enable the network to learn complex patterns and relationships within the data. It operates on the output of each neuron, transforming the weighted sum of inputs into the neuron's output.

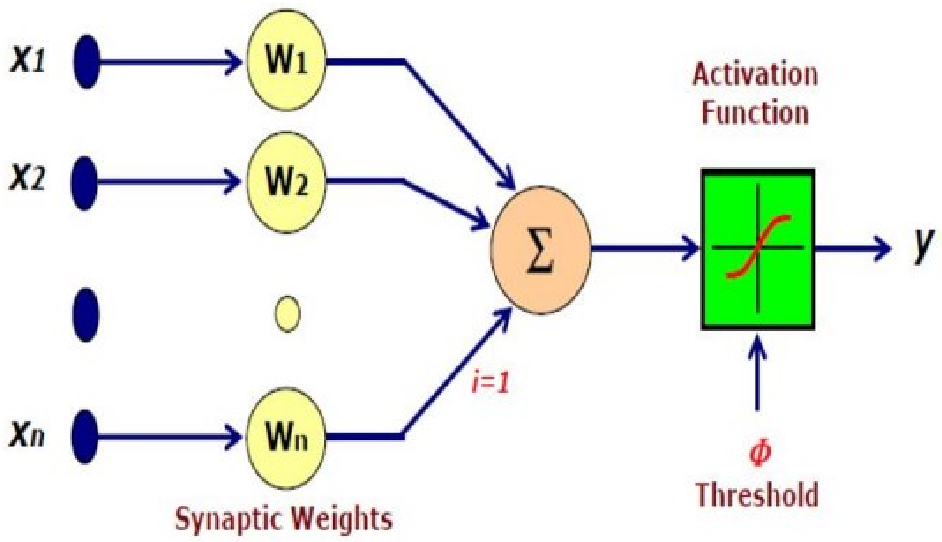


Figure 3.1 Working principle of Artificial Neural Network

The block diagram of a model of neuron is shown in Figure 3.1 (Haykin, 1997) on the basis of designing ANN. Main elements of a neural network are:

Input layer: The neuron receives a set of n inputs, x_i, $i = 1,2, \dots, n$, from its neighbouring neurons and a bias which is equals to 1. There is a weight (w_i) assigned to each input. As a result, the individual neurons, network connectivity, weights corresponding to the connections between neurons, and activation function of neurons can all be used to characterize a neural network. The weighted sum of the inputs determines the state or activity of a neuron and is given by

$$a = \sum_{i=1}^{n+1} w_i \; x_i \; + b = W^T X$$

where, $X = \{x_1 \, x_2 \dots x_n \; 1\}^T$, b is the bias and w_i represents the weights corresponding to inputs.

Hidden layer: The hidden layer is primarily responsible for processing input information in various ways. The quantity of hidden layers varies from one algorithm to another. The primary determinant of the neural network's architecture was the number of neurons in each layer and the hidden layers in a particular problem. Therefore, choosing the ideal number of hidden layers for a neural network is crucial.

Activation function

An activation function is a function that is differentiable and is utilized to smooth the result of cross product between the neurons and the weights. In the context of ANN, activation function of a node plays a crucial role in determining the output of that node based on a provided input or set of inputs. Following are some of the commonly used activation functions for input x:

Sigmoid or Logistic activation function : f (x) = $\frac{1}{1+e^{-x}}$; range (0, 1)

Hyperbolic tangent function : $f(x) = \tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$; range(-1, 1)

Gaussian activation function : $f(x) = e^{-x^2}$; range (0, 1]

Rectified Linear units (ReLu) : $(x)=0$ $for\ x \leq 0$

x $for\ x > 0$; range [0, ∞)

Sinusoid activation function : $(x) = \sin(x)$; range [-1, 1]

Output layer: The output layer gives the results after the computation.

3.4.4.1 MAJOR TYPES OF ARTIFICIAL NEURAL NETWORK

3.4.4.1.1 Feed-forward Artificial Network

A feed-forward artificial neural network is characterized by the requirement that data flows in a unidirectional manner from input to output, without incorporating any feedback loop. The network imposes no limitations on the number of layers, the type of transfer function, or the connections between artificial neurons. The most basic form of a feed-forward artificial neural network is a single perceptron, capable of learning only situations that can be linearly separated. Rumelhart *et al.* (1986) developed a basic feed-forward artificial neural network with multiple layers for analytical applications.

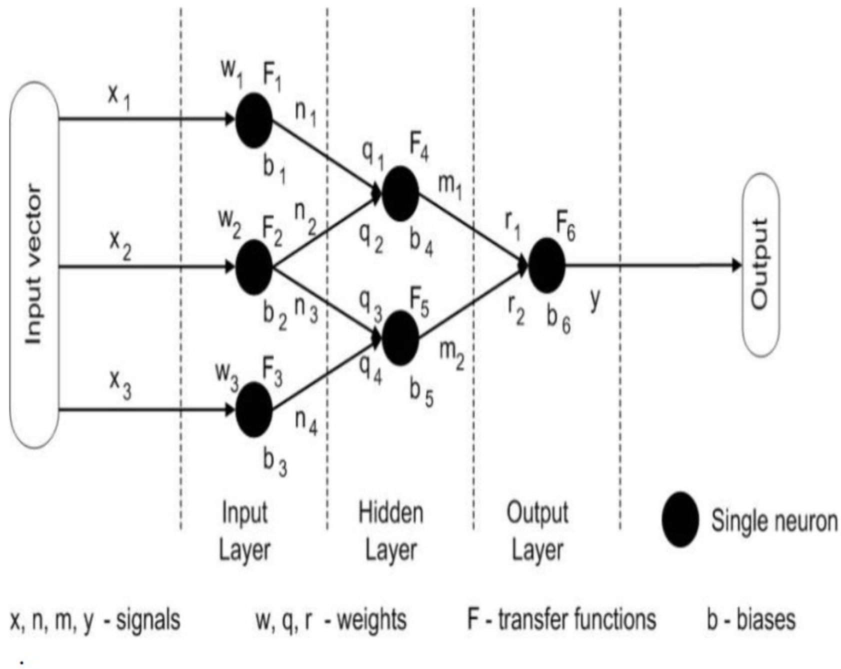


Figure 3.2 Architecture of Feed-forward Artificial Neural Network

3.4.4.1.2 Time-delay neural network

There are mainly two approaches to modeling time series data using neural networks. The initial approach entails the direct inclusion of time information by incorporating recurrent connections from output nodes to the preceding layer, as introduced by Elman in 1990. The second approach involves imparting an implicit representation of time, where a static neural network, such as a multilayer perceptron, is endowed with dynamic properties, as elucidated by Haykin in 1999. The dynamic attributes can be introduced into a neural network, either in the form of short-term or long-term memory, depending on the desired retention time.

In order to effectively handle temporal data in neural networks, it is essential to incorporate a form of short-term memory to impart dynamic capabilities. A straightforward method to incorporate short-term memory involves employing time delays, which can be implemented at the input layer of the network. An example of such an architectural design is a Time-delay neural network (TDNN), which is done in the current research. The incorporation of time delays in the neural networks is inspired by neurobiology, as single delays are widely observed in the brain and hold significance in the neurobiological processing of information. In TDNN, the activation function for node i at time t can be expressed as follows:

$$y_t = f(t) = f\left(\sum_{j=1}^{q}\sum_{d=0}^{p} w_{ij}\,(t-d)y_j(t-d)\right)$$

where $y_i(t)$ the output node i at time t, $w_{ij}(t)$ is the connection weight between node i and j at time t, p is the number of tapped delays, q is the number of nodes connected to node i from preceding layer, d denotes the time delays and f is the activation function, typically the logistic sigmoid**.**

For this research, we specifically focus on scenario where tapped delays are present only in the input layer. The TDNN configurations with a single hidden layer can be denoted as I:Hs:Ol, where I is the number of nodes in the input layer, H is the number of nodes in the hidden layer O is the number of nodes in the output layer, s represents the logistic sigmoid transfer function and l indicates the linear transfer function. Figure 3.3 provides architecture of TDNN model.

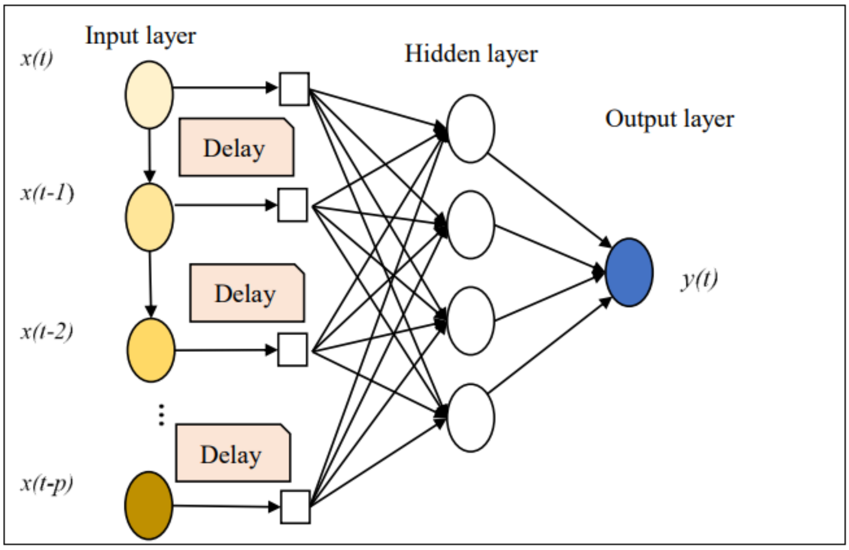


Figure 3.3 Architecture of Time-delay Neural Network

The determination of the mapping of input features to perceptron outputs in a neural network is not fixed and is crucial for generating a network with preferred characteristics. In this context, the identification of the number of layers, neurons, and parameters in the neural network plays a pivotal role. The aim is to create a network that effectively captures the relationship between input and output variables.

Determining the suitable number of hidden layers and neurons is a challenging task, particularly when dealing with specific problems. Neural networks typically incorporate a modest number of hidden nodes to facilitate efficient training, especially in cases involving intricate, non-linear relationships. The determination of the number of layers and neurons is achieved through empirical experimentation and exploration of various parameters.

In Figure 3.3, there is an illustration of a TDNN, showcasing an architecture suitable for a specific problem. The hidden nodes are essential for capturing complex patterns in the data, particularly in the context of time series analysis. However, having an excessive number of hidden nodes can result in overfitting the model, impeding its generalization ability. The decision on the number of hidden layers is often based on empirical observations of the data and the desired level of accuracy. While linear statistical problems may suffice with a single hidden layer, more intricate tasks such as time series prediction may benefit from additional hidden layers. Achieving a balance between model complexity and training efficiency is essential in practice. Too few hidden nodes can lead to underfitting, while an excess may cause overfitting. The process involves iterative adjustments to the network architecture and parameters to attain satisfactory results. Experimentation and analysis are crucial for developing an effective neural network model as the determination of the optimal number of hidden nodes are difficult.

Determining the number of output nodes, denoted as y(t), is a straightforward aspect of the study. A single output node has been employed, and the forecasting involves predictions using an iterative approach akin to the Box-Jenkins method. This method utilizes the forecasted value as an input for predicting future values. The architecture of the network is characterized by the interconnections between nodes in different layers, where fully connected networks are prevalent in forecasting and various applications. In fully connected networks, all nodes in one layer are linked to every node in the next higher layer.

Another critical aspect of modelling involves the selection of the activation function. The activation function defines the relationship between a node's inputs and outputs, introducing a level of nonlinearity beneficial for most neural network applications. While any differentiable function is theoretically suitable as an activation function, in practice, only a select few "well-behaved" functions—those that are bounded, monotonically increasing, and differentiable—are commonly utilized. The logistic sigmoid transfer function is widely favoured for both hidden and output nodes. However, when dealing with forecasting scenarios that involve continuous target values, opting for a linear activation function for output nodes is a reasonable choice.

The construction of a TDNN model typically involves the utilization of training and test samples. The training sample is employed for developing the TDNN model, while the test sample is used to evaluate the model's forecasting capabilities. In certain instances, a third sample, known as validation sample, may be introduced to address overfitting concerns or determining the termination point of the training process. Using a single test set for both validation and testing is not uncommon, especially in scenarios involving limited datasets.

The selection of training and testing samples plays a crucial role in influencing the adequacy of TDNNs. Deciding how to partition the data to training and test sets is a key consideration. Although there is no one-size-fits-all solution to this problem, various factors such as problem characteristics, data type, and available data size should be considered when making this decision. Many researchers adhere to a rule of dividing the data into 90 per cent for training and 10 per cent for testing, or variations like 70 per cent vs. 30per cent or 80 per cent vs 20 per cent.

The quantity of the data allocated for network training is influenced by factors such as the training method, network structure, and the complexity of the specific problem or the level of noise present in the dataset. It's important to note that the forecasting performance of the TDNN typically improves with an increase in the size of the training sample.

In this study, 80 per cent of the monthly and weekly prices of black pepper is used as training set and rest 20 per cent is used as the testing set. The price for years 2021 and 2022 were used for the validation purposes.

The ‘nnetar’ package in R software was used to develop the TDNN model in the study.

Forecast Accuracy Measures

Several forecasting measures is used to determine the best TDNN model using the MAPE and RMSE values. The best suitable TDNN model is selected for black pepper prices based on the agreement between the fitted and actual price plots, residual ACF and PACF plots, MAPE and RMSE values. The model is also evaluated using the Ljung-Box test.

3.4.4.1.3 Recurrent Neural Network

An ANN with a recurrent topology is called a Recurrent Neural Network (RNN). It is having a similarity to the feed-forward neural network, with the exception that back propagations are unrestricted. Under certain conditions, information is transmitted both forward and backward and the network eventually develops an internal state that enables it to exhibit dynamic temporal behaviour. Using their internal memory, RNN can perform any sequence of inputs. Any basic building block, known as artificial neuron in a fully RNN is directly connected to every other basic building block in every direction, making it the most basic type of recurrent artificial neural network. Other RNNs, such as Hopfield, Jordan, bi-directional networks, Elman, and others, are some subsets of RNN.

RNNs connect numerous layers of networks, allowing details from prior time steps, along with the output nodes, to persist into subsequent time steps. As each layer processes input and parameters, it takes into account the output from preceding layers, providing the network with a type of memory.

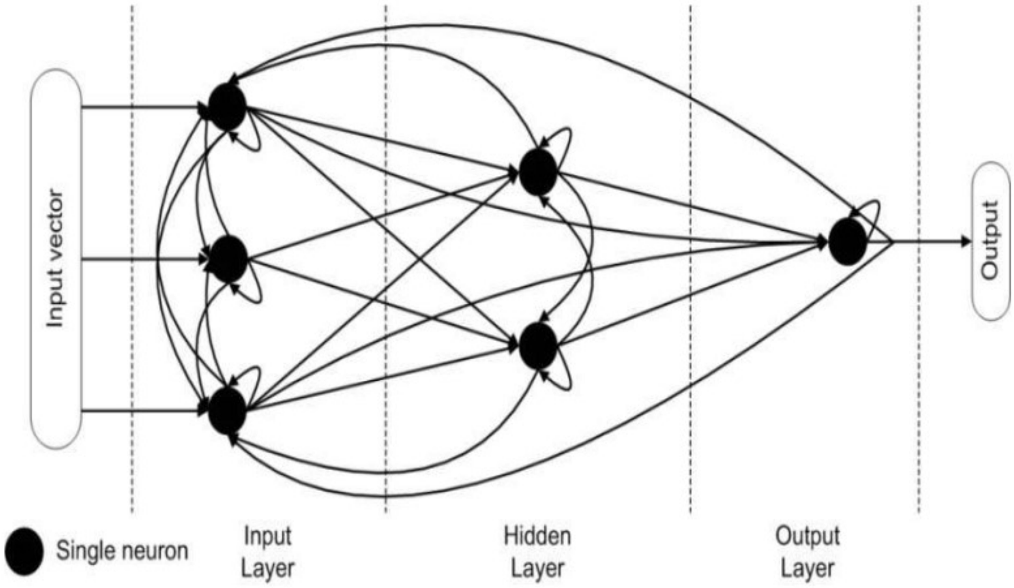


Figure 3.4 Architecture of Recurrent Neural Network

The primary issue with conventional RNNs is the problem of vanishing gradient, where the vanishing gradient diminishes and eventually reaches close to zero. This challenge is effectively addressed by the Long Short-Term Memory network model (LSTM), a type of RNN that incorporates memory and forget cells. The rationale behind choosing this neural network over a simple RNN lies in the limitation of simple RNNs to leverage information from the distant past due to the absence of long-term memory. This limitation hampers the ability of simple RNNs to effectively learn patterns with extended dependencies. In order to address this issue, the LSTM network has been opted, a specialized type of RNN, as it possesses the capability to learn patterns with long dependencies and is adept at detecting complex patterns. LSTM model functions through write, read, and delete operations using distinct cells within its hidden layers, facilitated by three gates: the input gate, forget gate, and output gate. The movement of information from a layer to another occurs via the cell state. Firstly, the 'forget gate' allows essential information to flow via the cell state. The forget gate, consisting of a point-wise multiplication and sigmoid layer, retains relevant information (when the sigmoid value is 1) and discards unnecessary details (when the sigmoid value is 0). The 'update state' enables the addition of new information through the 'tanh layer', generating a new value for updating, with a sigmoid layer determining which values to update. The resulting new cell state is a combination of the forget and update states. The sigmoid layer in the output state determines the relevant information in the new cell state, which is normalized using the tanh layer and is crucial for generating the output. The visual representation of LSTM model is given in Figure 3.5.

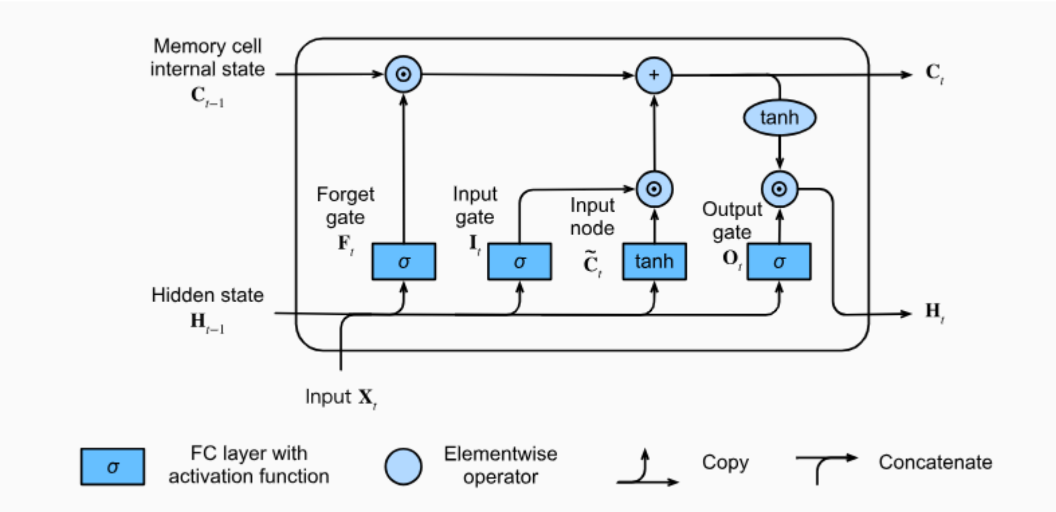


Figure 3.5 LSTM cell and its components

In this proposed study, the LSTM neural network has been employed to better understand the price behaviour of black pepper.

LSTM MODEL IMPLEMENTATIONS

Step 1

The LSTM model was implemented using the Python library Keras and TensorFlow. The models were trained using 80 percent of the data, while the remaining 20 percent was designated as the test set. Additionally, the data underwent normalization and differencing during the pre-processing stage.

Step 2

Creation of LSTM model has been executed. There are 3 LSTM layers which are the input, hidden and output layer. Total number of parameters are 1,26,452. All the parameters will be used to train the model. After creating the model, we have to train the model on the parameters. The model makes predictions on the training data during training process, and loss function is used to calculate the error between these predictions and the actual target values. The training loss is the average of these errors across all training samples. The goal of training is in adjusting the weights and biases of the model to minimize this training loss. After each epoch, the weights of the model get updated and the new epoch works on those updated values that process continues in every epoch. Epoch can be defined as number of times an algorithm visits the dataset. Mean Absolute Error (MAE) has been used as the measure to continue the epoch until MAE reaches a minimum.

Step 3

After training the model, the LSTM model is tested using testing data. Finally, the forecast evaluation metrics which include MAPE and RMSE are obtained for the model.

Hyperparameters

Determination of the values of the hyperparameters involves a search operation and testing of a combination of hyperparameters, that yield the best results for the final model. MAPE and RMSE were integral to this evaluation in selecting the most optimal parameters. The following values were chosen which include those of neuron input layer, sequence length determining LSTM memory duration, the epoch for countering overfitting, the neuron hidden and output layers along with the activation function and optimization function which is provided in Table 3.1.

Table 3.1. Values/Functions used for the parameters in the LSTM model

Parameters	**Chosen value/function**
Neurons input layer	100
Sequence length	10
Epochs	50
Neurons hidden layer	100
Neurons output layer	1
Activation Function	Tanh
Optimization Function	Adam

Forecasting accuracy measures

The developed price forecasting models were evaluated based on different performance metric or goodness of fit measures. The model performance metrics used in this study are MAPE and RMSE. The model is also evaluated using the Ljung-Box test.

3.5 Comparison of models and model validation

Accuracy and validity are closely intertwined, although they operate in distinct directions. While results may be valid, they may not necessarily be accurate, as highlighted by Markidakis and Hibbon in 1979. Conversely, an accurate forecast may lack validity. Therefore, accuracy serves as a metric for assessing the appropriateness of the forecast. In the current investigation, the evaluation of models was conducted based on the accuracy of forecasts, and this assessment was subjected to various tests.

The accuracy of forecast for the various models was selected by the accuracy measures like RMSE and MAPE (Prathima, 2018). The six forecasting models are compared based on lower values of MAPE and RMSE in order to find the best suitable model for forecasting monthly weekly price of garbled and ungarbled black pepper. The best selected model will be validated on price data for 2021 and 2022.

The R and Python codes used for the analysis of data in the study are provided in Appendix No.1.

Results and Discussion

4. RESULTS AND DISUSSION

The results of the study “A comparative analysis of price forecasting models for black pepper” are discussed in this section under the following titles:

4.1 Analysis of monthly price of garbled black pepper in the Kochi market

4.2 Analysis of monthly price of ungarbled black pepper in the Kochi market

4.3 Analysis of weekly price of garbled black pepper in the Kochi market

4.4 Analysis of weekly price of ungarbled black pepper in the Kochi market

4.1 Analysis of monthly price of garbled black pepper in the Kochi market

The results obtained from the analysis of monthly price of garbled black pepper are presented below:

4.1.1 Pattern for monthly price data of garbled black pepper

The time plot for monthly average prices of garbled black pepper in the Kochi market from 2000 to 2020 were used to study the price pattern and it exhibited some notable patterns which is shown in Figure 4.1. Overall, the figure portrays a dynamic economic landscape with wide fluctuations in the price of black pepper, including periods of growth, decline, and recovery.

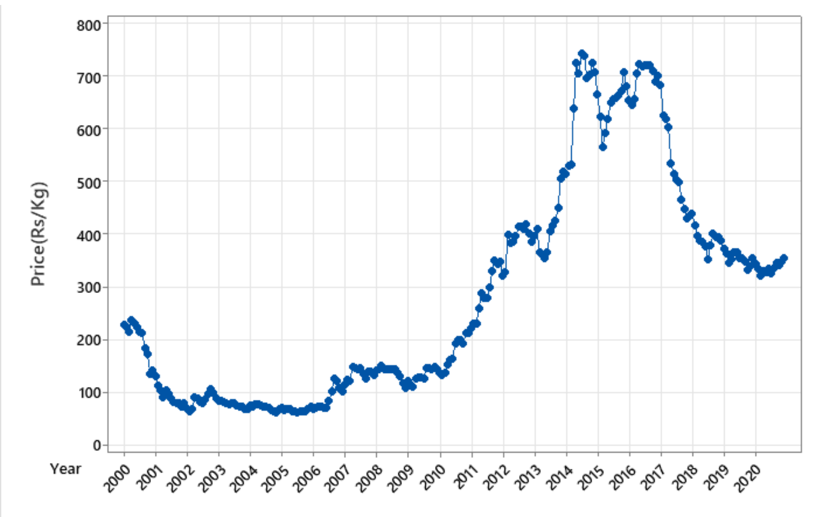


Figure 4.1 Price pattern for monthly price of garbled black pepper

4.1.2 Trend Analysis

Trend is defined as the general tendency of a time series data to increase or decrease during a long period of time. Trend analysis was carried out for monthly average price of garbled black pepper. Different functional forms like linear, quadratic, exponential, etc. were tried and the suitable model in each case was chosen based on the MAPE and RMSE values.

The trend equation fitted for monthly price of garbled black pepper and their accuracy measures are provided in Table 4.1. Graphical plots revealing the pattern of the series based on the different trend models tried are plotted in Figure 4.2 – 4.4.

Table 4.1 Trend equations for monthly price of garbled black pepper

Functional Form	**Trend equation**	**MAPE**	**RMSE**
Linear Trend Model	$Y_t = 12.9 + 2.144 \times t$	52.4	122.01
Exponential Trend Model	$\boldsymbol{Y_t = 67.6373 \times (1.00901^{\wedge}t)}$	**39.9**	**106.47**
Quadratic Trend Model	$Y_t = 1.5 + 2.412 \times t - 0.00106 \times t^2$	53.7	123.51

As observed from the table, lower MAPE value of 39.9, shows the adequacy of the exponential trend model in explaining the trend of monthly price of garbled black pepper.

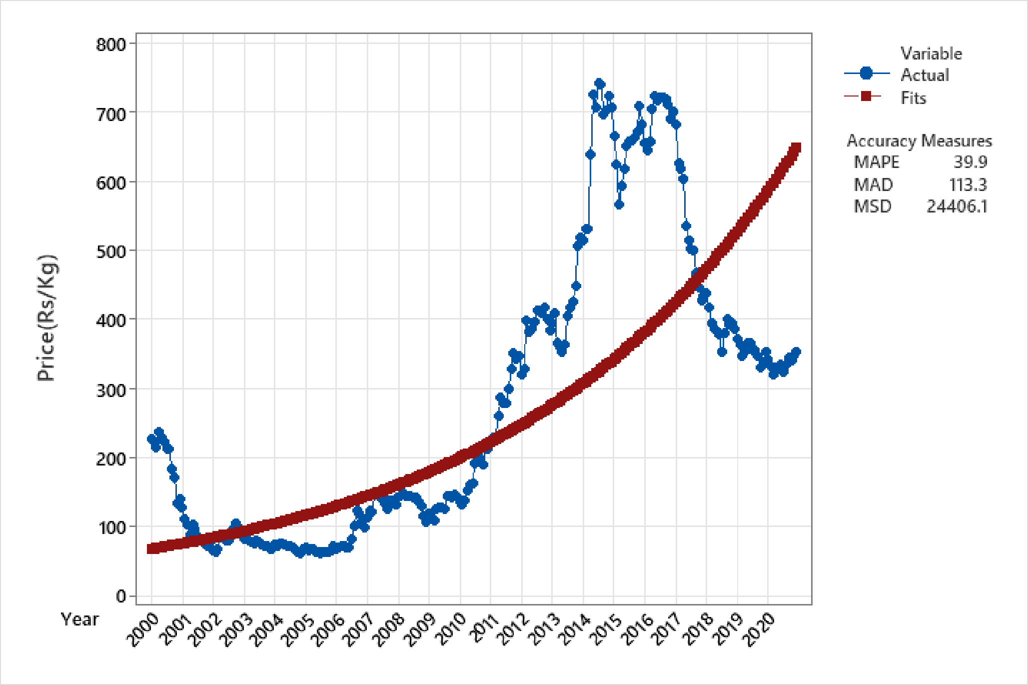


Figure 4.2 Linear trend plot for monthly price of garbled black pepper

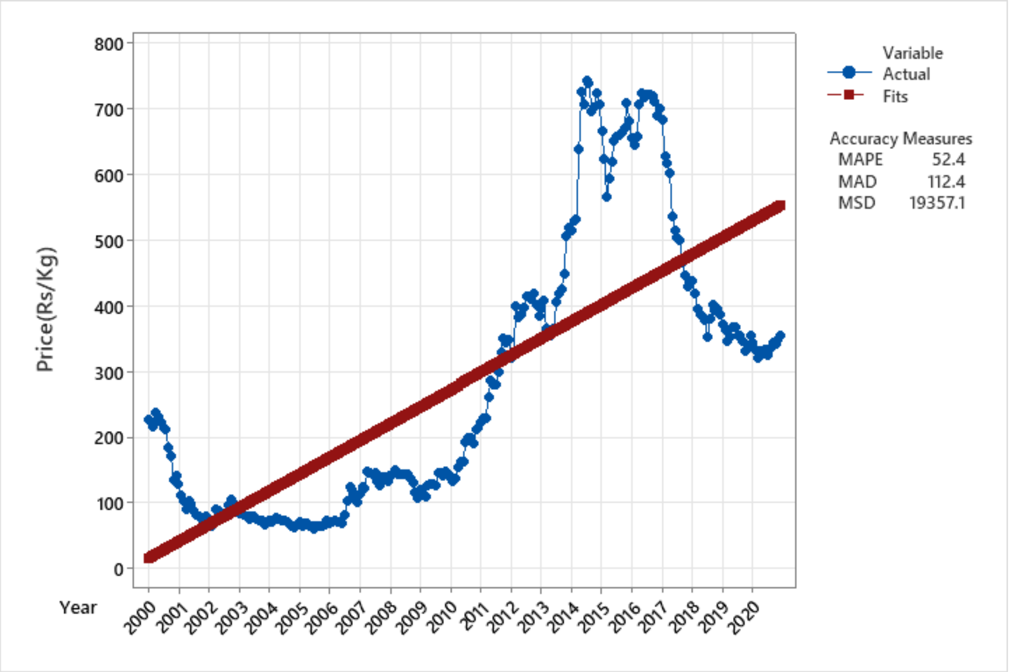


Figure 4.3 Exponential trend plot for monthly price of garbled black pepper

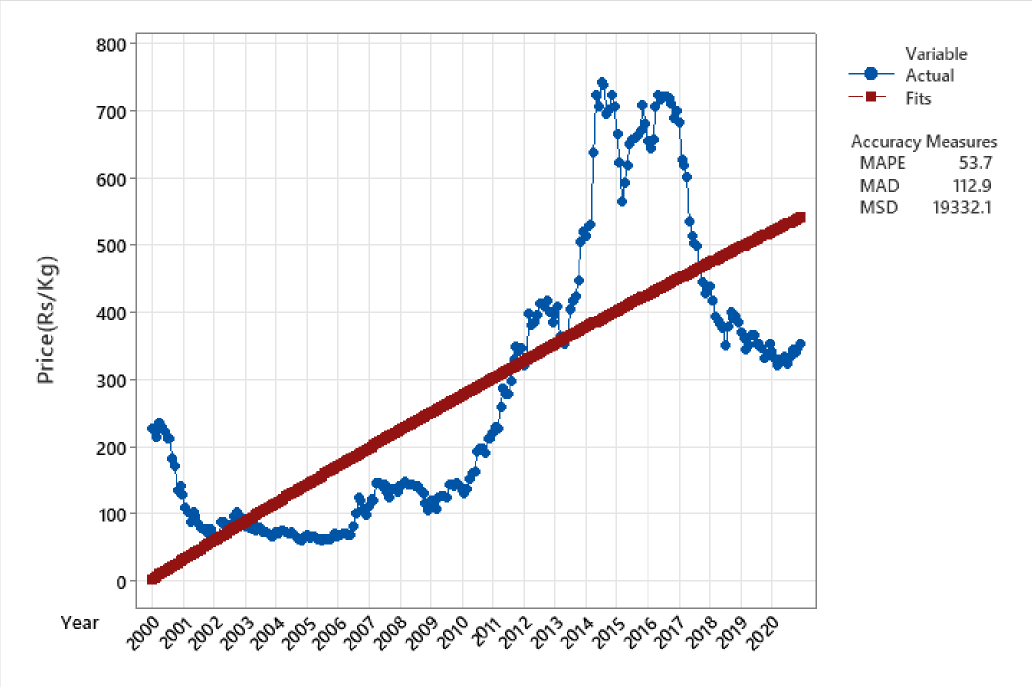


Figure 4.4 Quadratic trend plot for monthly price of garbled black pepper

4.1.3 Decomposition for monthly price of garbled black pepper

A time series is a sequence of data points ordered chronologically (Anderson, 1971). The time series data for monthly price of garbled black pepper is decomposed into four components, as described in time series analysis: trend, seasonal variation, cyclic variation, and irregular using a multiplicative model. The result is presented in Figure 4.5, which consists of four panels. In the first panel, the observed monthly price of black pepper has been plotted. The second panel illustrates the trend in the prices, while the third and fourth panels depict the seasonal and random variations, respectively.

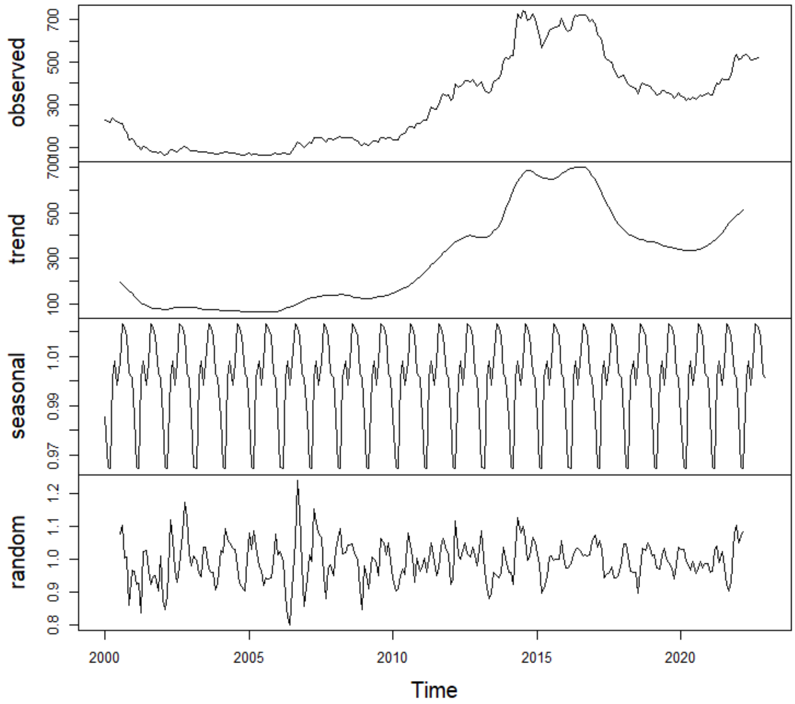


Figure 4.5 Decomposition for monthly price of garbled black pepper

4.1.3.1 Seasonal indices for monthly price of garbled black pepper

Decomposition of monthly price of garbled black pepper during January 2000-December 2020 indicated the seasonality pattern in every year. Seasonal indices were calculated for 12 months (January to December) from the monthly price data of garbled black pepper from January, 2000 to December, 2020 to understand the seasonal behaviour of prices.

Seasonal indices are calculated by ratio to moving average for monthly garbled black pepper data provided in Table 4.2 and seasonal plot is shown in Figure 4.6.

Table 4.2 Seasonal indices for monthly prices of garbled black pepper

Month	Seasonal indices (%)
January	98.21
February	97.71
March	95.47
April	100.58
May	100.95
June	99.99
July	101.25
August	100.79
September	102.79
October	101.88
November	100.28
December	100.63

Black pepper is seasonal in production and hence the prices exhibit considerable seasonality. The months of January, February, and March exhibit lower seasonal indices suggesting relatively lower prices during this period. This corresponds with the flowering stage of black pepper in May and June, where the vines begin yielding in the third year but reach full bearing capacity in the sixth or seventh year. The seasonal indices start to rise in April and continue to increase through May, June, and July, reaching a peak in September. This corresponds with the harvest season, which typically starts in November and extends to March, with variations in different regions.

The indices suggest lower prices during the harvest season, aligning with the increased supply of black pepper in the market. The higher seasonal indices from September to November indicate a period of relatively elevated prices. Additionally, the indices for November and December are relatively higher, corresponding to the harvesting period in November and the continued arrivals in the market, which are prominent from February to May. The interplay between seasonal indices and the months of harvest and market arrivals underscores the seasonality in black pepper production and its direct impact on pricing dynamics.

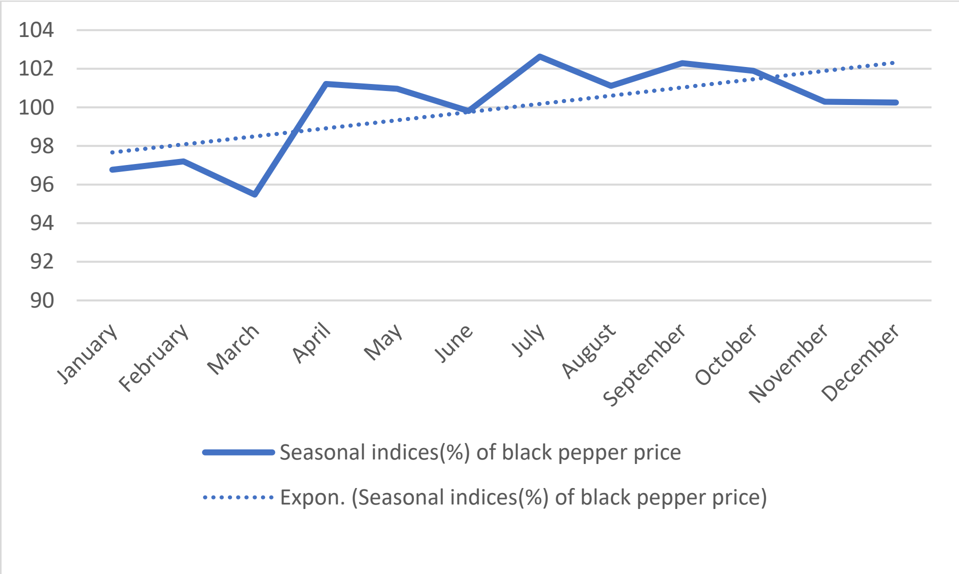


Figure 4.6 Seasonal plot for monthly price of garbled black pepper

4.1.3.2 Cyclical variation for monthly price of garbled black pepper

Price cycles indicate deviations in price levels from the average trend due to the recurring patterns of economic booms and recessions. These cycles, which span several years and exhibit various periodicities, are particularly evident in the production and pricing of black pepper.

The cyclical pattern of monthly price of garbled black pepper is depicted in Figure 4.1. The first 11-year cycle was from 1983 to 1993 and the second cycle from 1993, showed some fluctuation near the peak values and reached the lowest value in 2004 (Sabu, 2015). The third cycle started from 2005 when the prices started gradually increasing until reaching a peak at Rs. 174.22 in 2010. This upward trend is part of the cyclical variation, reflecting a period of growth and possibly influenced by factors such as increased demand, favourable weather conditions, or changes in market dynamics.

Subsequently, the prices experience a sharp spike in 2011, reaching Rs. 288.01, indicating a peak in the cycle. This could be attributed to specific factors like a surge in demand, reduced supply, or other market forces. The years from 2011 to 2018 witnessed increased price volatility, likely influenced by various economic, political, or industryspecific factors and there was a peak of above Rs.700 in 2014 indicating the demand of black pepper in the market. Following the peak, there is a noticeable downward trend in prices, reaching a low of Rs. 336.47 in 2019. This decline represents the trough of the cycle, possibly influenced by factors such as increased production, changes in consumer preferences, or economic conditions.

The cyclical variation continues as prices start to rise again in 2020, reaching Rs. 419.46. This upward movement suggests a new phase of the cycle, possibly influenced by factors such as improved market conditions, changes in agricultural practices, or global economic trends.

4.1.3.3 Irregular variation for monthly price of garbled black pepper

The random effect is the residual effect after the trend, seasonal and cyclical effects have been removed from the original observations. As observed from the fourth panel of Figure 4.5, monthly price of garbled black pepper exhibited significant irregular variations during 2000 to 2020. They represent random effect such as demand and supply shocks on account of climatic aberrations or due to speculative factors.

4.1.4 Forecast for monthly price of garbled black pepper

A series of models comprising exponential smoothing model, ARIMA/SARIMA, ARCH/GARCH and ANN were fitted to forecast monthly price of garbled black pepper. The best model was selected based on the forecast accuracy measure: MAPE and RMSE. The results are provided in following subsections:

4.1.4.1 Exponential smoothing models for monthly price of garbled black pepper

Exponential smoothing is a particular moving average technique applied to time series in order to produce smoothed data to make forecasts. The exponential smoothing method, weighs past observations by exponentially decreasing weights to forecast future values. Holt-Winters' Multiplicative Seasonal (HWMS) model was identified best among the different exponential smoothing models like SES, DES, HWAS and HWMS, for the monthly price of garbled black pepper based on criteria like agreement between observed and fitted price plots, MAPE and RMSE. The fit of the HWMS model for monthly price of black pepper is provided in Figure 4.7 and the actual and fitted values are in close agreement. The estimates of parameters of HWMS model are provided in Table 4.3. Various accuracy measures for HWMS model are provided in Table 4.4.

Table 4.3 Estimates of parameters for the HWMS model for monthly price of garbled black pepper

Parameter	**α**	**B**	**γ**
Estimate	0.9116	0.1319	0.08

Based on the above values for the parameters, HWMS model to forecast the monthly price of black pepper are as given below,

Level: L_t= 0.911* ($\frac{Y_t}{S_{t-12}}$)+.0884*(L_{t-1}– 8.462)

Trend: b_t = 0.1319($L_t - L_{t-1}$)- 7.3486

Seasonality: $S_t = S_{t-12}$

Forecast: F_{t+m}= $(L_t + b_{tm})S_{t-12+m}$

The values of L_t, b_t and S_t were substituted in the forecast equation $F_{t+m}$to obtain the price.

Table 4.4 Model accuracy measures by HWMS model for monthly price of garbled black pepper

Accuracy measure	**Value**
RMSE	17.87
MAPE	5.28

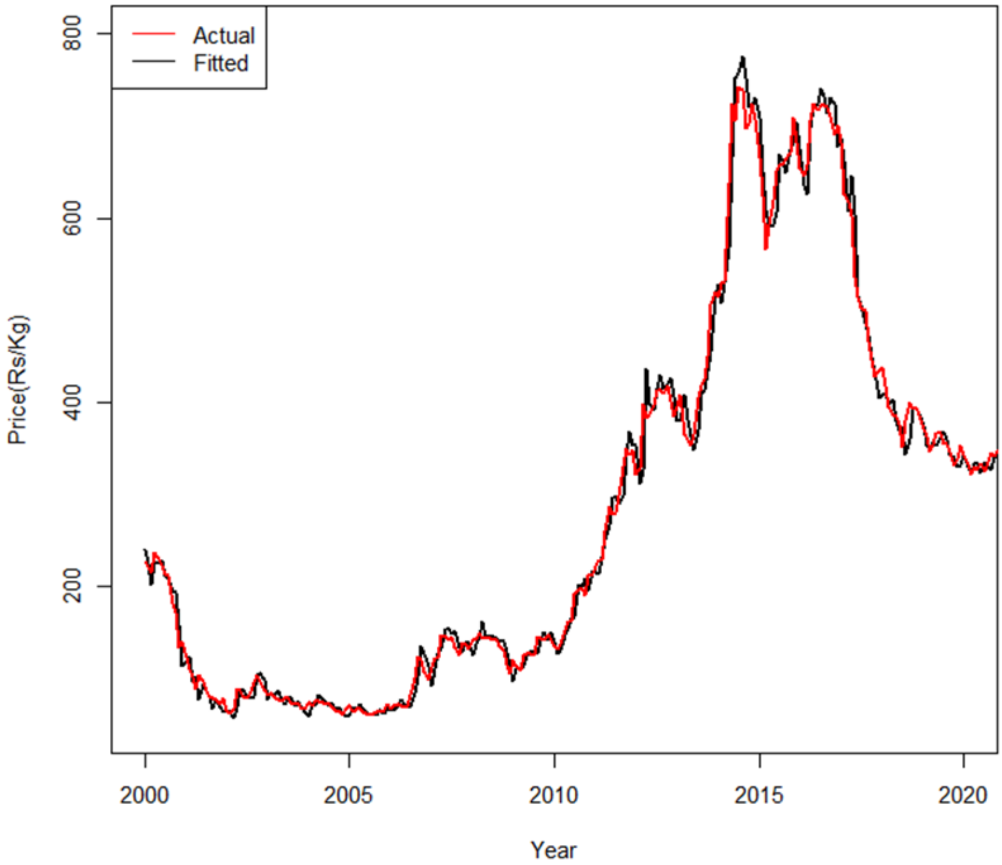


Figure 4.7 Actual and fitted plot for HWMS model for monthly price of garbled black pepper

The residual plot for monthly price of garbled black pepper from HWMS model is given in Figure 4.8 along with the ACF and PACF residual plots (Figure 4.9). It showed that most of the ACF and PACF values lie within the confidence limits indicating the adequacy of the model.

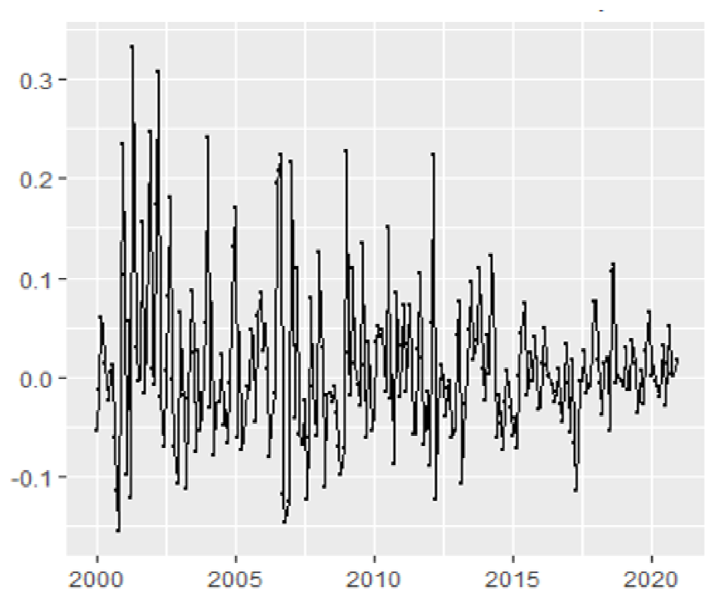


Figure 4.8 Residual plot for monthly price of garbled black pepper for HWMS model

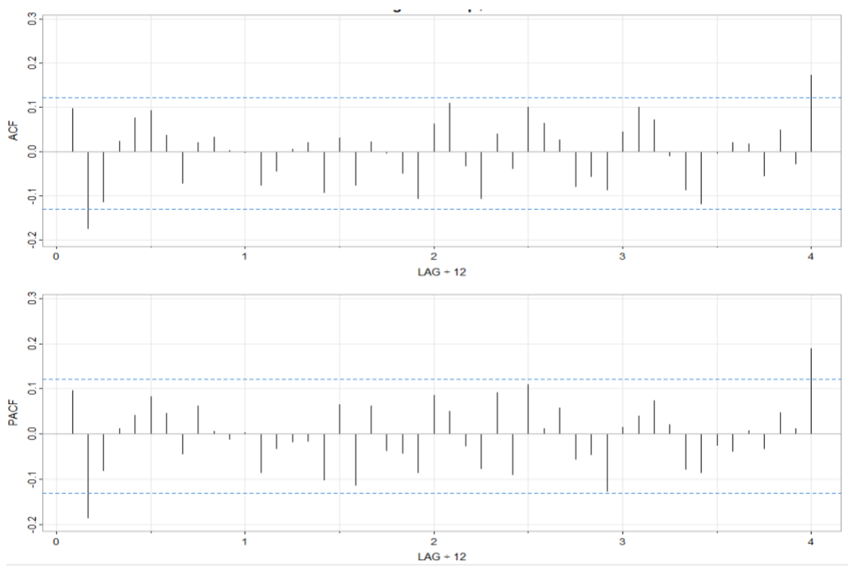


Figure 4.9 Residual ACF and PACF plots for monthly price of garbled black pepper for HWMS model

4.1.4.2 ARIMA model for monthly price of garbled black pepper

The time plot for monthly price of black pepper indicates that it is non-stationary. Autocorrelations and partial autocorrelation were computed for monthly price of garbled black pepper are provided in Table 4.5. Significance of autocorrelations upto 16 lags confirm the non stationarity of the data and partial autocorrelation value has very high value for lag 1.

Table 4.5 ACF and PACF values for monthly price of garbled black pepper

			Ljung-Box Statistic				
Lag	**Auto-correlation**	**SE**	**Value**	**Df**	**Probability (p)**	**Partial Auto-correlation**	**SE**
1	0.996	0.063	252.83	1	<0.001	0.996	0.063
2	0.989	0.062	503.37	2	<0.001	-0.262	0.063
3	0.983	0.062	751.53	3	<0.001	0.047	0.063
4	0.975	0.062	996.83	4	<0.001	-0.134	0.063
5	0.967	0.062	1239.02	5	<0.001	0.000	0.063
6	0.957	0.062	1477.49	6	<0.001	-0.153	0.063
7	0.946	0.062	1711.45	7	<0.001	-0.124	0.063
8	0.935	0.062	1940.72	8	<0.001	0.019	0.063
9	0.924	0.062	2165.48	9	<0.001	0.062	0.063
10	0.913	0.061	2385.75	10	<0.001	-0.006	0.063
11	0.901	0.061	2601.38	11	<0.001	-0.017	0.063
12	0.889	0.061	2812.29	12	<0.001	-0.001	0.063
13	0.877	0.061	3018.19	13	<0.001	-0.066	0.063
14	0.864	0.061	3219.14	14	<0.001	0.037	0.063
15	0.853	0.061	3415.49	15	<0.001	0.034	0.063
16	0.841	0.061	3607.24	16	<0.001	-0.020	0.063

p<0.01 indicates significance of autocorrelation

The ACF and PACF plots for monthly price of garbled black pepper is depicted in Figure 4.10. It was evident that in the ACF plot, spikes upto 16 lags fall above the confidence limit and PACF plot showed spikes for a number of lags (1, 2, 4 and 6) beyond confidence limits also indicated the non-stationarity of the time series.

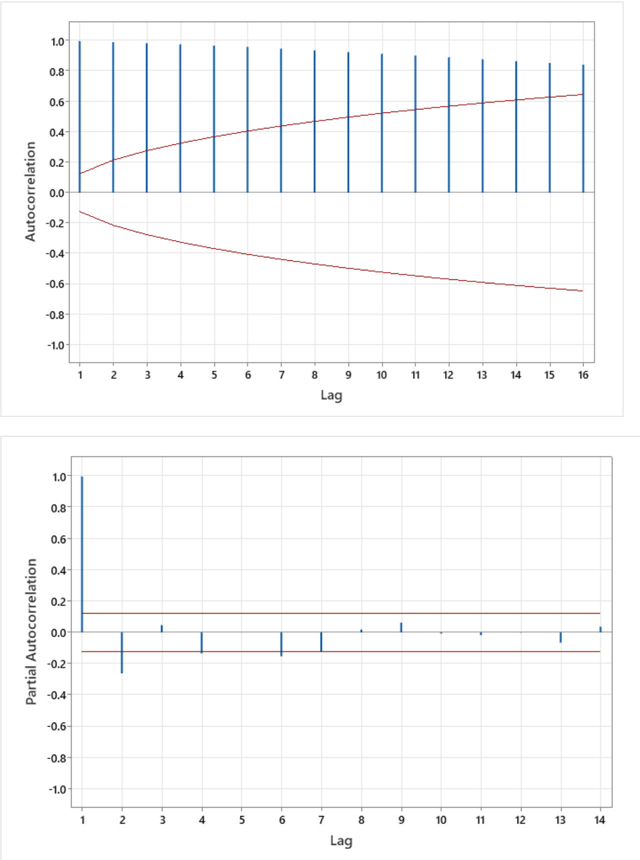


Figure 4.10 ACF and PACF plots for monthly price of garbled black pepper

The stationarity of the data was evaluated using ADF test. Following the initial differencing of the data, the ADF test was repeated. The significance of the ADF test statistic, as indicated in Table 4.6, affirmed the necessity of taking the first difference to achieve stationarity in the data.

Table 4.6 ADF test with critical values for monthly price of garbled black pepper

Garbled Black pepper price	ADF test statistic	Probability(p)	Critical values
Actual	-1.4491	0.8078	0.869
First difference	-9.9902**	0.01	-2.32

** indicates significant at 5 per cent level($p<0.05$)

Among several ARIMA models tried, the tentative models were chosen based on the value of MAPE, AIC and RMSE. SARIMA (2,1,2) $(3,0,2)_{12}$ was chosen as the best forecast model for monthly price of garbled black pepper which is depicted in Table 4.7.

Table 4.7 Identified SARIMA(p, d, q)(P, D, Q)$_{12}$ models for monthly price of garbled black pepper

Tentative models	MAPE	AIC	RMSE
SARIMA(2,1,2)(1,0,1)$_{12}$	4.71	2154.91	17.16
SARIMA(2,1,2)(3,0,2)$_{12}$	**4.71**	**2153.46**	**16.79**

The parameters of the model SARIMA(2,1,2)(3,0,2)$_{12}$ along with their tests of significance are provided in Table 4.8.

Table 4.8 SARIMA(2,1,2)(3,0,2)$_{12}$ model parameters for monthly price of garbled black pepper

Model Parameters	Estimate	SE	Z value	Probability
Non-seasonal difference	1			
AR Lag 2 (ϕ_2)	0.503***	0.108	4.673	0.00
MA Lag 2(θ_2)	-0.606***	0.116	-5.45	0.00
Seasonal difference	0			
AR Seasonal Lag 1(Φ_1)	-1.452***	0.191	-7.577	0.00
AR Seasonal Lag 2(Φ_2)	-0.575**	0.224	7.755	0.01
AR Seasonal Lag 3(Φ_3)	0.165**	0.075	4.219	0.02
MA Seasonal Lag 1(Θ_1)	1.538***	0.198	7.755	0.00
MA Seasonal Lag 2(Θ_2)	0.828***	0.196	4.219	0.00

**indicates significance at 5 per cent percentage level($p<0.05$)

***indicates significance at 1 per cent percentage level ($p<0.01$)

The general form of the SARIMA(2,1,2)(3,0,2)$_{12}$ model equation is as follows:

$$(1-\phi_2 B^2)(1-B)(1-\Phi_1 B^{12} - \Phi_2 B^{12} - \Phi_3 B^{12}) y_t =$$

$$(1-\theta_2 B^2)(1-\Theta_1 B^{12} - \Theta_2 B^{12})\epsilon_t \ , y_t = \log_e Y_t$$

Where, ϕ_2 = 0.503, Φ_1= -1.452, Φ_2=0.575, Φ_3=0.165, θ_2=-0.606, Θ_1=1.538, Θ_2=0.828

The SARIMA $(2,1,2)(3,0,2)_{12}$ model equation is as follows:

$$(1-0.503B^2)(1\text{-}B)(1+1.452B^{12}+0.575B^{12}+0.165B^{12})\, y_t =$$

$$(1+0.606B^2)(1\text{-}1.538^{12}\text{-}0.828B^{12})\epsilon_t$$

The plot of actual monthly price and fitted price of garbled black pepper using SARIMA$(2,1,2)(3,0,2)_{12}$ model is given in Figure 4.11.

Figure 4.11 Actual and fitted values of SARIMA(2,1,2) $(3,0,2)_{12}$ for monthly price of garbled black pepper

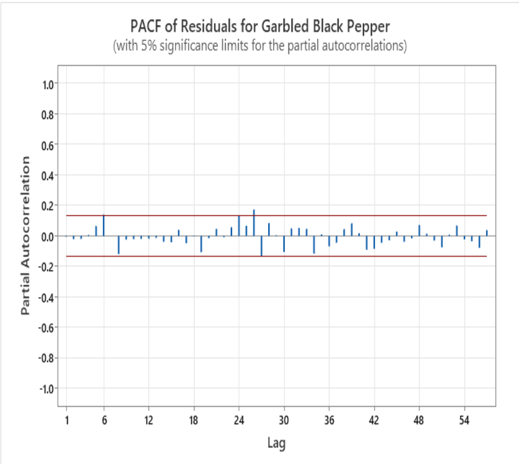
The adequacy of the model was also evaluated using the value of Ljung-Box 'Q' statistic is provided in Table 4.9 and was found to be insignificant. So, overall we can say SARIMA $(2,1,2)(3,0,2)_{12}$ model shows satisfactory result, among different ARIMA models.

Table 4.9 Ljung-Box 'Q' statistic for residuals of SARIMA (2,1,2)(3,0,2)$_{12}$ model

Statistic	p-value
20.894	0.1403^{NS}

NS: Non-significant

Residual ACF and PACF plots for the SARIMA (2,1,2)(3,0,2)$_{12}$ model fitted for monthly price of garbled black pepper is provided in Figure 4.12. It could be seen that majority of the spikes in the ACF and PACF plots fall within the critical values, thus indicating the adequacy of the fitted model SARIMA (2,1,2) (3,0,2)$_{12}$ for forecasting monthly price of garbled black pepper.



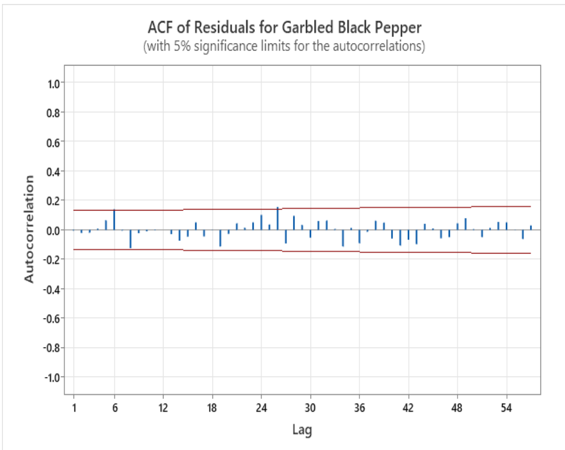


Figure 4.12 Residual ACF and PACF plots for monthly price of garbled black pepper for SARIMA(2,1,2) (3,0,2)$_{12}$

4.1.4.3 Volatility in monthly price of garbled black pepper

The volatility in pepper prices is highly evident from price movements of the commodity. As the study uses the time series data which consist of volatility, the ARCH family models can be employed. The monthly price of garbled black pepper for the period 2000-2020 were further used to choose the ARCH family models using R software.

The Heteroscedasticity LM test was carried out to check the volatility or ARCH effect in the time-series. The results of the test are presented in Table 4.10, which reveals that, there is an ARCH effect in the time series.

Table 4.10 Heteroscedasticity LM Test for first differenced

Statistic	p-value
21.25***	.000

*** Significant at 1 per cent level of significance

Several ARCH models were tested, but their coefficients were determined to be insignificant, highlighting the need for the implementation of the GARCH model.

4.1.4.3.1 GARCH(1,1) model for monthly price of garbled black pepper

The estimates of the GARCH (1,1) model fitted for the monthly price of garbled pepper is given in the Table 4.11. It showed that the constant term, ARCH and GARCH parameters are positive and significant indicating the volatility.

Table 4.11 Estimates of GARCH(1,1) model for monthly price of garbled black pepper

Parameter	Coefficient	Std error	t statistic	p-value
Constant term(α_0)	0.0014**	0.004	0.350	0.01
ARCH term(α_1)	0.1239**	0.051	2.387	0.017
GARCH term(β_1)	0.8248***	0.082	9.974	0.00

**Significant at 5 per cent level of significance

*** Significant at 1 per cent level of significance

The GARCH(1,1) model is given by,

$$h_t = 0.0014+0.1239\varepsilon_{t-i}{}^2+0.8248\sigma_{t-j}^2$$

The time varying volatility includes a constant (0.0014), a component which depends on past errors ($0.1239\varepsilon_{t-i}{}^2$) and a component which depends on weighted average of past squared residuals ($0.8248\sigma_{t-j}^2$). In the Table 4.11, the p-value for the t-statistic of the first order coefficient (2.38) and the second order coefficient (9.97) suggests a significant GARCH (1,1) coefficient.

Table 4.12 Model accuracy measures by GARCH(1,1) model for monthly price of garbled black pepper

Accuracy measure	Value
RMSE	17.56
MAPE	4.63

Residual analysis was carried out to check the adequacy of the selected model. The Serial Correlation LM test for residual is presented in Table 4.13. The large value of p (p=0.432 >0.05) reveals that, there is no serial correlation in the residuals. The Ljung-Box test of residual is presented in Table 4.14. The large value of p (p=0.365 > 0.05) with respect to Ljung-Box 'Q' statistic indicates that the residuals are normally distributed.

Table 4.13 Serial Correlation LM Test for residuals of GARCH(1,1) Model

Statistic	p-value
6.62	0.432^{NS}

NS: Non-significant

Table 4.14 Ljung-Box 'Q' statistic for residuals of GARCH(1,1) Model

Statistic	p-value
41.78	0.365^{NS}

NS: Non-significant

4.1.4.4 ANN model for monthly price of garbled black pepper

Time delayed neural network (TDNN) model was fitted for monthly price of garbled black pepper. The best time lagged neural network with single hidden layer was found for each series by conducting experiments with the basic cross validation method. Out of a total of 36 neural network structures, a neural network model with six lagged observations as input nodes and two hidden nodes (6:2s:1l) performed better than other competing models in respect of forecasting accuracy measures. This means that most accurate price forecast for the given series is obtained when the price of six preceding months is used as inputs.

The selected TDNN model is described in Table 4.15. along with the model accuracy measures for both training and testing set.

Table 4.15 Model accuracy measures by TDNN model for monthly price of garbled black pepper

Model	No. of parameters	MAPE		RMSE	
		Train	Test	Train	Test
6:2s:11	17	5.68	4.29	15.85	13.92

The actual monthly price along with the predicted values for monthly price of garbled black pepper using TDNN model is provided in Figure 4.13.

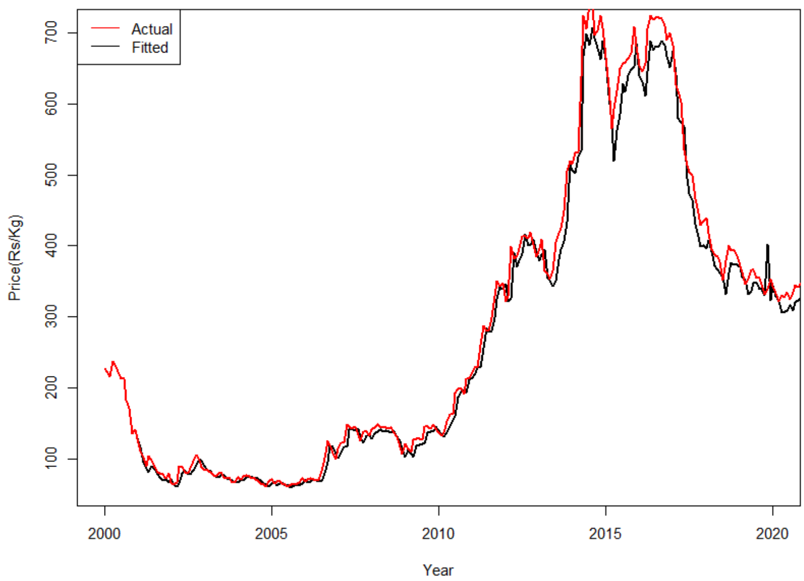


Figure 4.13 Actual and fitted plot for TDNN model for monthly price of garbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant which is shown in Table 4.16. So overall, we can say TDNN model shows satisfactory result.

Table 4.16 Ljung-Box 'Q' statistic for residuals of TDNN model

Statistic	p-value
21.58	**0.5804**^{NS}

NS: Non-significant

The residual plot from TDNN model is provided in Figure 4.14 which did not exhibit any specific pattern. The residual ACF and PACF plot is provided in Figure 4.15, majority of the spikes in the residual ACF and PACF are within the critical values indicating the adequacy of the model.

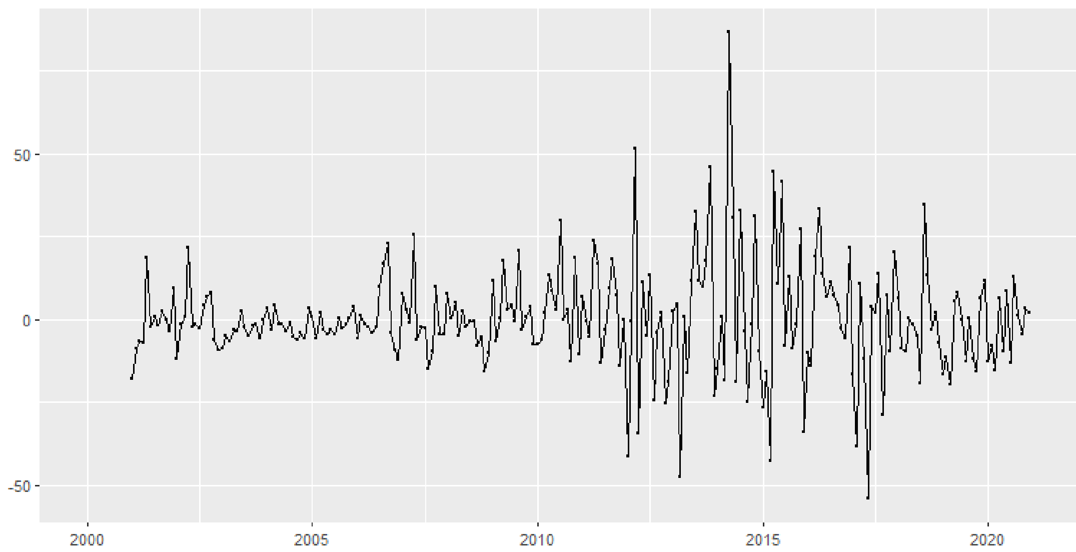


Figure 4.14 Residual plot for monthly price of garbled black pepper for TDNN model

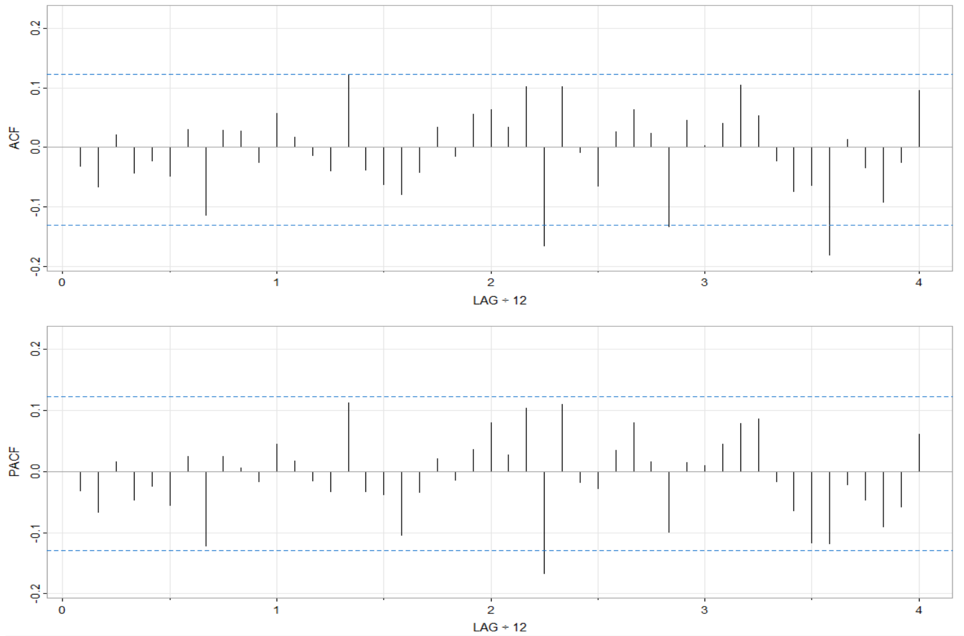


Figure 4.15 Residual ACF and PACF plots for monthly price of garbled black pepper based on TDNN model

4.1.4.5 LSTM model for monthly price of garbled black pepper

The LSTM model, a specialized type of RNN has been opted, as it possesses the capability to learn patterns with long dependencies and is adept at detecting complex patterns. Thus, it was fitted for monthly price of garbled black pepper.

The parameters used for the model was provided in Table 3.1. The LSTM model has been created with three layers by performing epoch of 50 times. During the training process, the model makes predictions on the training data, and the loss function is used to calculate the error between these predictions and the actual target values. The training loss is the average of these errors across all training samples. The goal of training is to adjust the weights and biases of the model to minimize this training loss. After each epoch, the weights of the model get updated and the new epoch works on those updated values that process continues in every epoch. In this model, Mean Absolute Error (MAE) has been used as the measure to continue the epoch until MAE reaches a minimum which is shown in Figure 4.16.

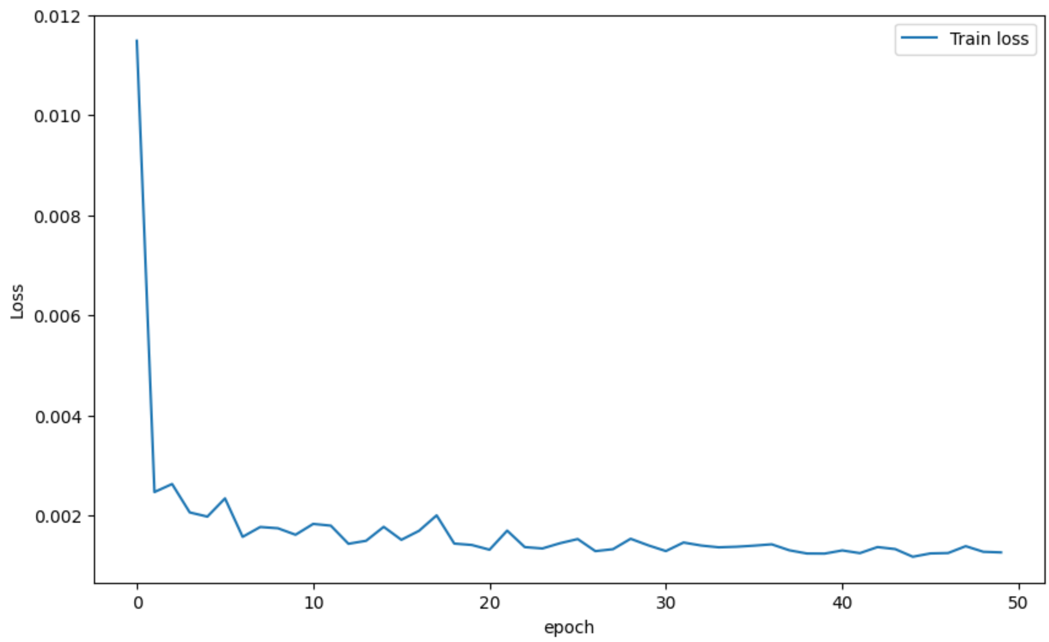


Figure 4.16 Training loss for LSTM model for monthly price of garbled black pepper

Various accuracy measures for LSTM model are provided in Table 4.17.

Table 4.17 Model accuracy measures by LSTM model for monthly price of garbled black pepper

Accuracy measure	**Value**
RMSE	28.50
MAPE	7.98

The actual monthly price along with the predicted monthly price of garbled black pepper using LSTM model is provided in Figure 4.17.

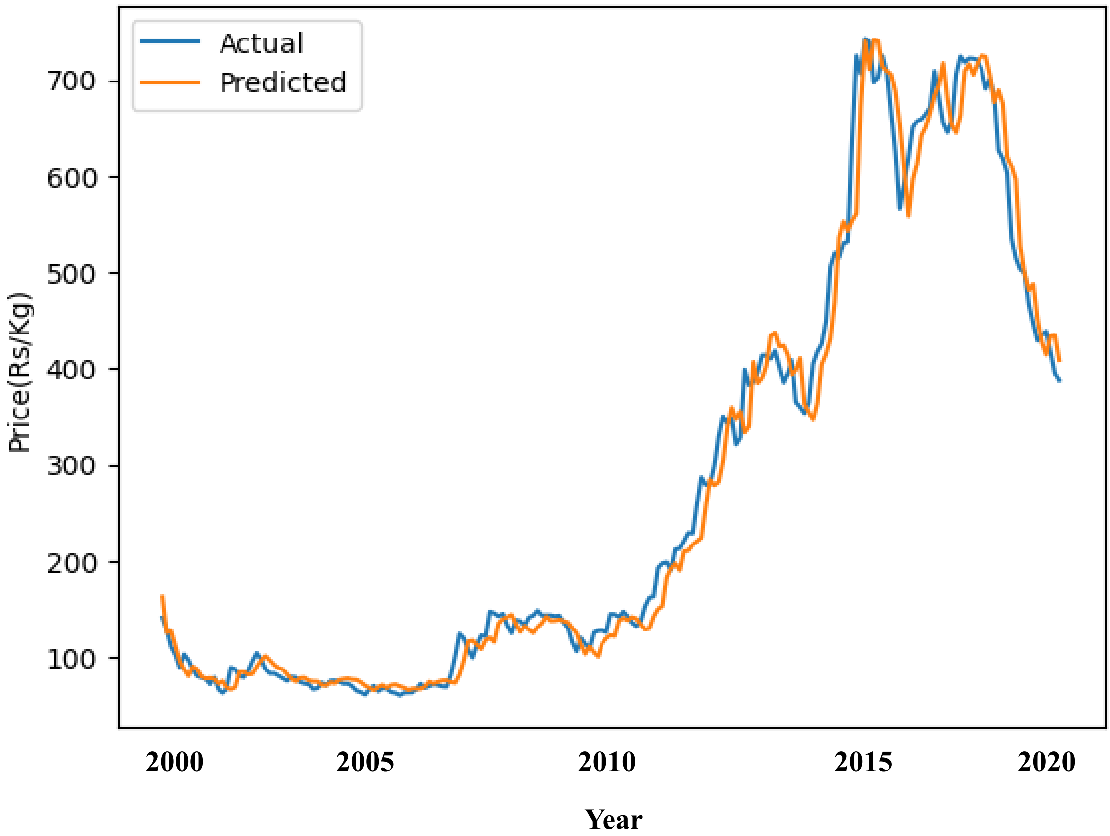


Figure 4.17 Actual and fitted plot for LSTM model for monthly price of garbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box ‘Q’ statistic and was found to be insignificant which is provided in Table 4.18. So, overall, we can say LSTM model shows a satisfactory result.

Table 4.18 Ljung-Box ‘Q’ statistic for residuals of LSTM model

Statistic	p-value
78.78	0.704^{NS}

NS: Non-significant

The residual plot from LSTM model is provided in Figure 4.18 and the residuals did not exhibit any specific pattern, they are scattered.

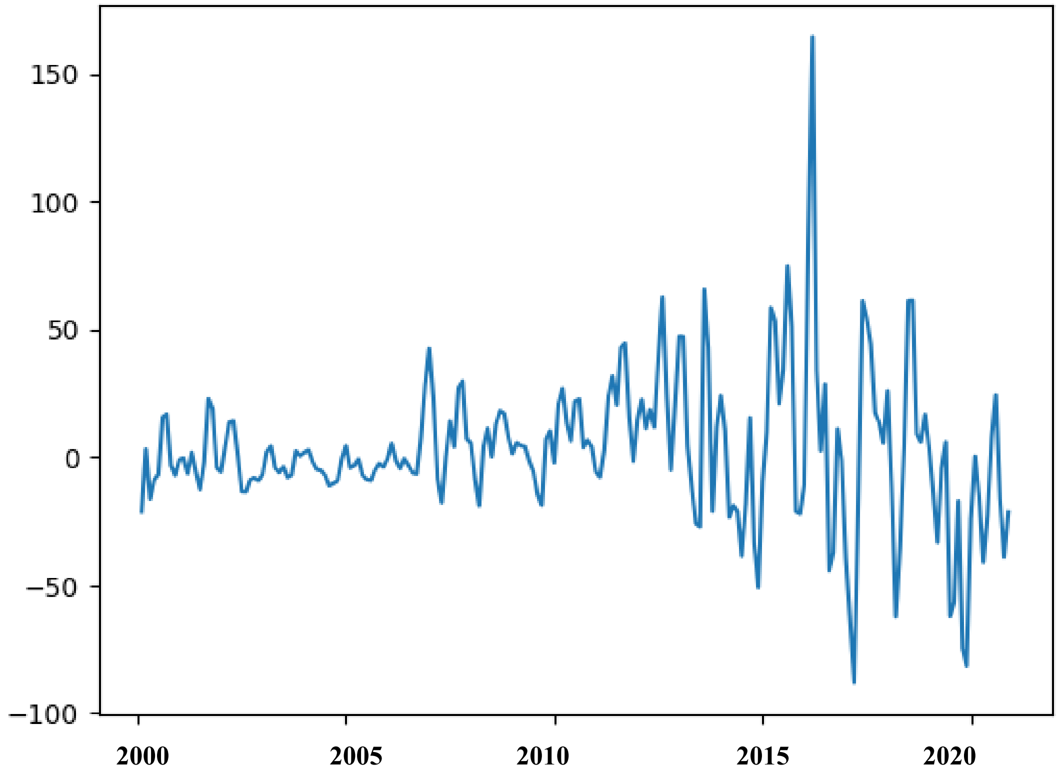


Figure 4.18 Residual plot for monthly price of garbled black pepper for LSTM model

4.1.5 Comparison of models

Forecast accuracy is closely related to validity of the results, although only in one direction. The results could be valid, but not particularly accurate. On the other hand, it is not possible to have an accurate forecast that is not valid. Therefore, accuracy is a measure of fitness of the forecast (Armstrong, 2009).

In the present study, a comparison of different models has been done in order to know the best model for forecasting monthly price of garbled black pepper in Kochi market. The comparison of all the 6 models were carried out based on the test MAPE and RMSE values which were considered to be least. According to the Table 4.19, the TDNN model with least MAPE and RMSE values was considered to be best among all the models considered.

Table 4.19 Comparison of time series forecasting models for monthly price of garbled black pepper

Model	MAPE	RMSE
Exponential trend model	39.9	106.47
SARIMA(2,1,2)(3,0,2)$_{12}$	4.71	16.19
GARCH(1,1)	4.63	17.56
HWMS	5.28	17.87
TDNN(6:2s:1l)	**4.29**	**13.92**
LSTM	7.98	28.50

4.1.5.1 Validation of forecasted monthly price for garbled black pepper using TDNN model

Below Table 4.20 and Table 4.21 shows the actual price (Rs/kg), forecasted price (Rs/kg) and their deviations (in per cent) for the monthly price of garbled black pepper for the next two years based on the TDNN model. The MAPE were found to be 4.19 and 4.86 respectively based on the actual and forecasted price of garbled black pepper using the TDNN model for the year 2021 and 2022.

Table 4.20 Forecasted monthly price for garbled black pepper from TDNN model for the year 2021

Month	Actual price (Rs/kg)	Forecast price (Rs/kg)	Deviations (in per cent)
January	346.11	349.2	0.89
February	346.67	359.58	3.72
March	375.04	386.12	2.95
April	402.09	406.41	1.07
May	396.84	410.9	3.54
June	421.26	422.38	0.26
July	418.92	426.23	1.74
August	415.78	433.15	4.17
September	418.45	440.13	5.18
October	440.52	453.33	2.90
November	515.54	460.36	10.70
December	536.38	466.01	13.11

Table 4.21 Forecasted monthly price for garbled black pepper from TDNN model for the year 2022

Month	Actual price (Rs/kg)	Forecast price (Rs/kg)	Deviations (in per cent)
January	510.5	470.20	7.89
February	518.14	475.00	8.32
March	531.95	479.16	9.92
April	534.86	483.36	9.62
May	523.85	487.40	6.95
June	508.95	489.01	3.91
July	509.48	495.33	2.77
August	516.88	498.36	3.58
September	520.78	508.33	2.39
October	521	510.12	2.08
November	523	520.24	0.52
December	525	523.36	0.31

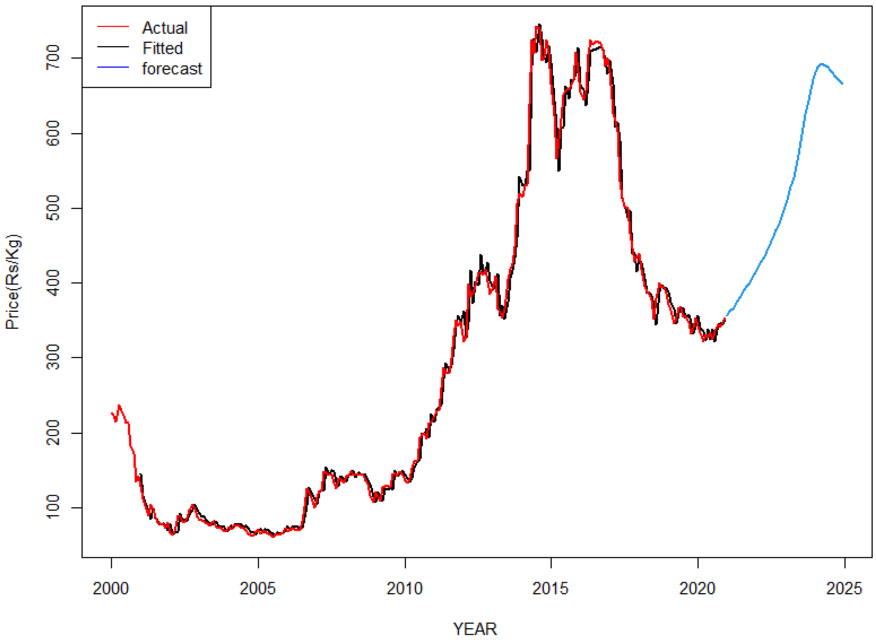


Figure 4.19 Actual, fitted and forecasted plot for TDNN model for monthly price of garbled black pepper

The plot of actual and fit values along with the forecasts for the monthly price of garbled black pepper price using TDNN model is given in Figure 4.19 and it was evident that actual and predicted prices are in agreement.

The TDNN model captures the pattern of the monthly price of garbled black pepper with the MAPE value 4.29. The residual plot from TDNN model is plotted in Figure 4.14 and was found to be scattered. Further, the residual ACF and PACF plots for the TDNN model given in Figure 4.15 exhibited that the majority of the spikes fall within the limits.

The LSTM model, being a type of recurrent neural network, would have been a better model if there are large data points. It stands out in handling sequences by effectively remembering and utilizing information over longer data points through its unique gating mechanism. This makes LSTMs well-suited for tasks like predicting time series, where capturing complex patterns and dependencies in the data is crucial. As the dataset grows, LSTMs prove advantageous over simpler models in retaining context and making more accurate predictions by considering the relationships between past inputs.

Thus, for monthly price of garbled black pepper price at Kochi market, TDNN model was selected as the best forecast model.

4.2 Analysis of monthly price of ungarbled black pepper in the Kochi market

The results obtained from the analysis of monthly price of ungarbled black pepper are presented below:

4.2.1 Pattern for monthly price data of ungarbled black pepper

The time plot for monthly average prices of ungarbled black pepper in the Kochi market from 2000 to 2020 were used to study the price pattern and it exhibited some notable patterns which is shown in Figure 4.20. Wide fluctuations could be observed in the price of ungarbled black pepper. These fluctuations reflect the intricate interplay of economic and market dynamics shaping the value of ungarbled black pepper over this period.

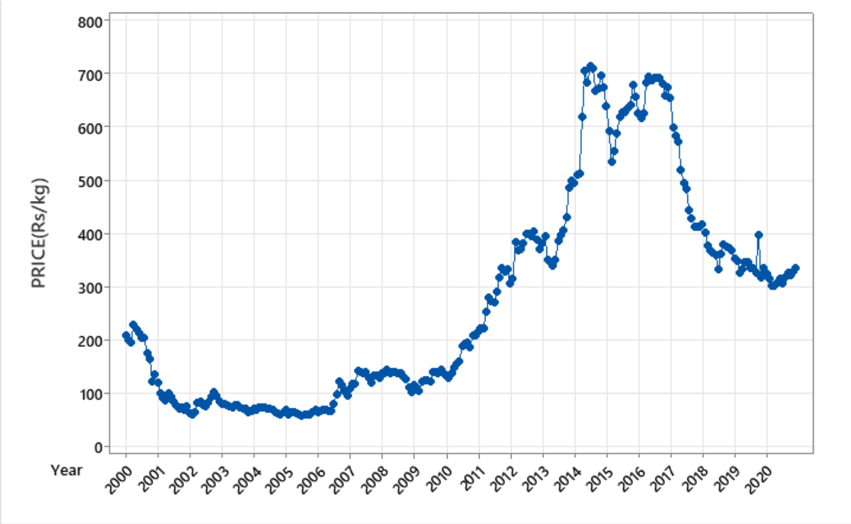


Figure 4.20 Price pattern for monthly price of ungarbled black pepper

4.2.2. Trend Analysis

Trend analysis was carried out for monthly price of ungarbled black pepper. Different functional forms like linear, quadratic, exponential, etc. were tried and the suitable model in each case was chosen based on the criteria like MAPE and RMSE values.

The trend equation fitted for monthly price of ungarbled black pepper along with the accuracy measures are provided in Table 4.22 along with the graphical plot of different trend models from Figure 4.21 to 4.23.

Table 4.22 Trend equations for monthly price of ungarbled black pepper

Functional Form	**Trend equation**	**MAPE**	**RMSE**
Linear Trend Model	$Y_t = 11.8 + 2.051 \times t$	52.5	122.12
Exponential Trend Model	$\boldsymbol{Y_t = 63.999 \times (1.00907^\wedge t)}$	**39.9**	**106.47**
Quadratic Trend Model	$Y_t = 2.6 + 2.392 \times t - 0.00135 \times t^2$	54.1	123.97

As observed from the table, lower MAPE value of 39.9 shows the adequacy of the exponential trend model in explaining the trend of monthly price of ungarbled black pepper.

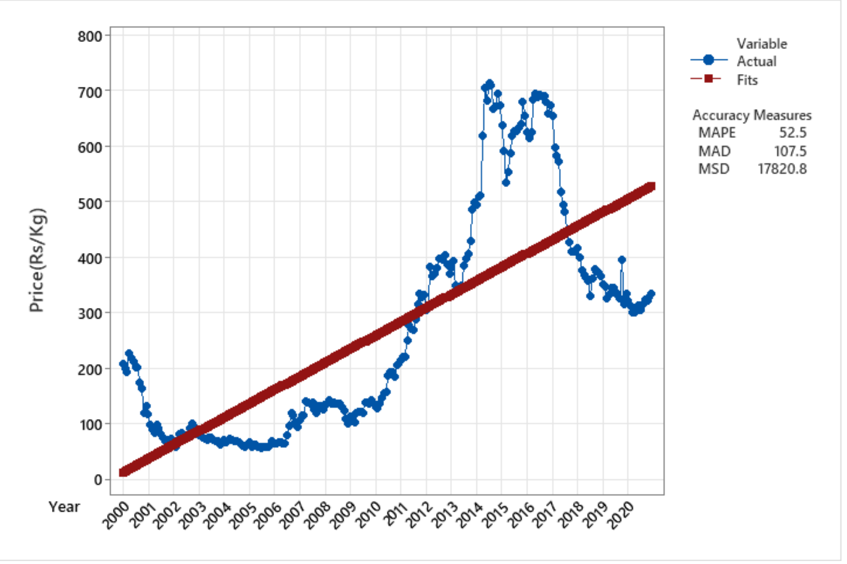


Figure 4.21 Linear trend plot for monthly price of ungarbled price of black pepper

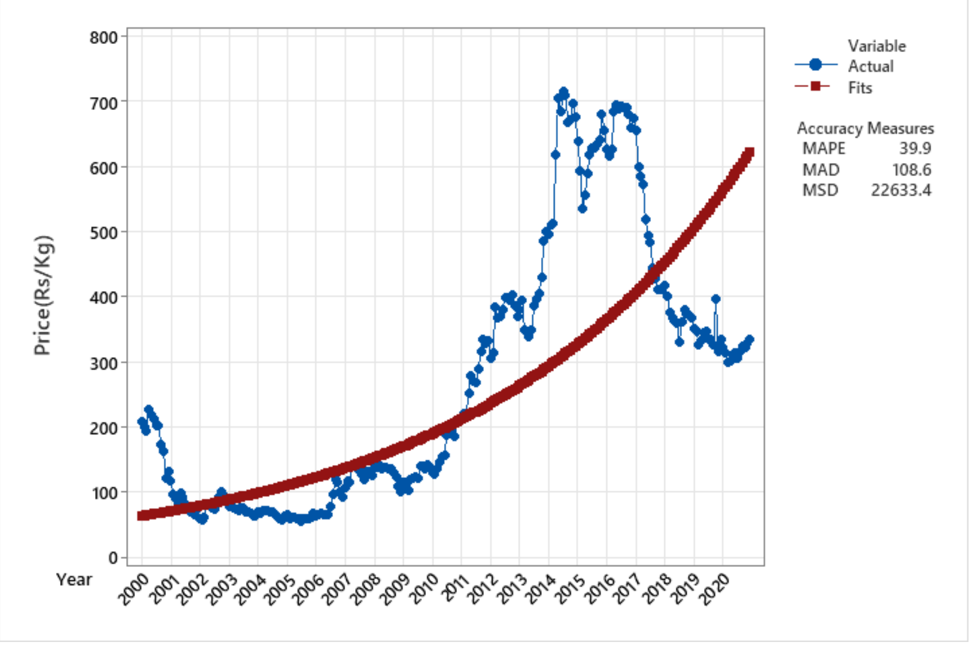


Figure 4.22 Exponential trend plot for monthly price of ungarbled black pepper

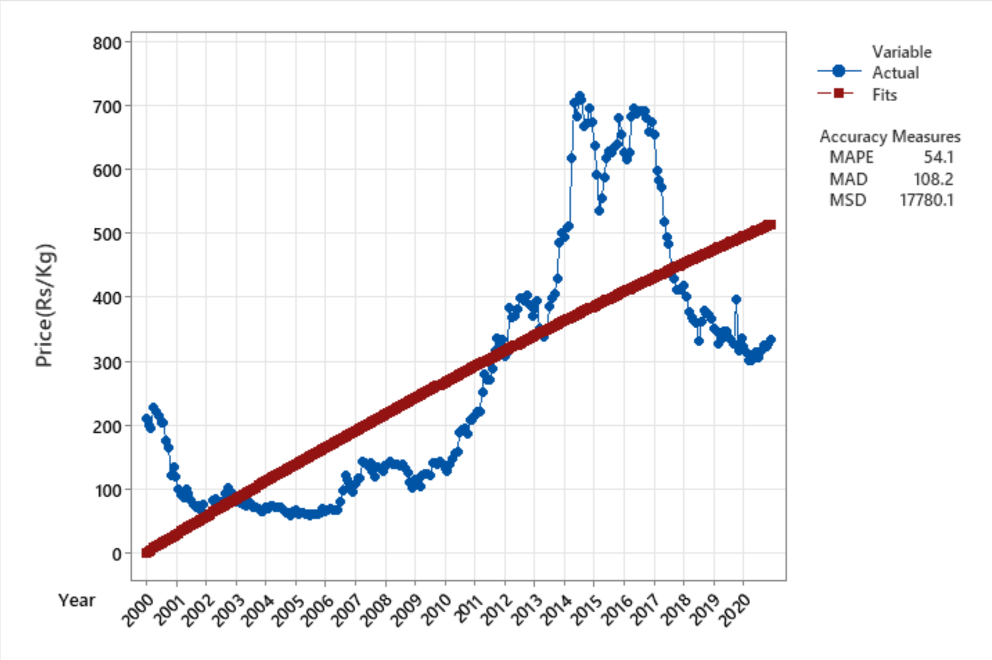


Figure 4.23 Quadratic trend plot for monthly price of ungarbled black pepper

4.2.3 Decomposition for monthly price of ungarbled black pepper

The time series data for monthly price of ungarbled black pepper is decomposed into four components, as described in time series analysis: trend, seasonal variation, cyclic variation, and irregular using a multiplicative model. The result is presented in Figure 4.24, which consists of four panels. In the first panel, the observed black pepper prices have been plotted. The second panel illustrates the trend in the prices, while the third and fourth panels depict the seasonal and random variations, respectively.

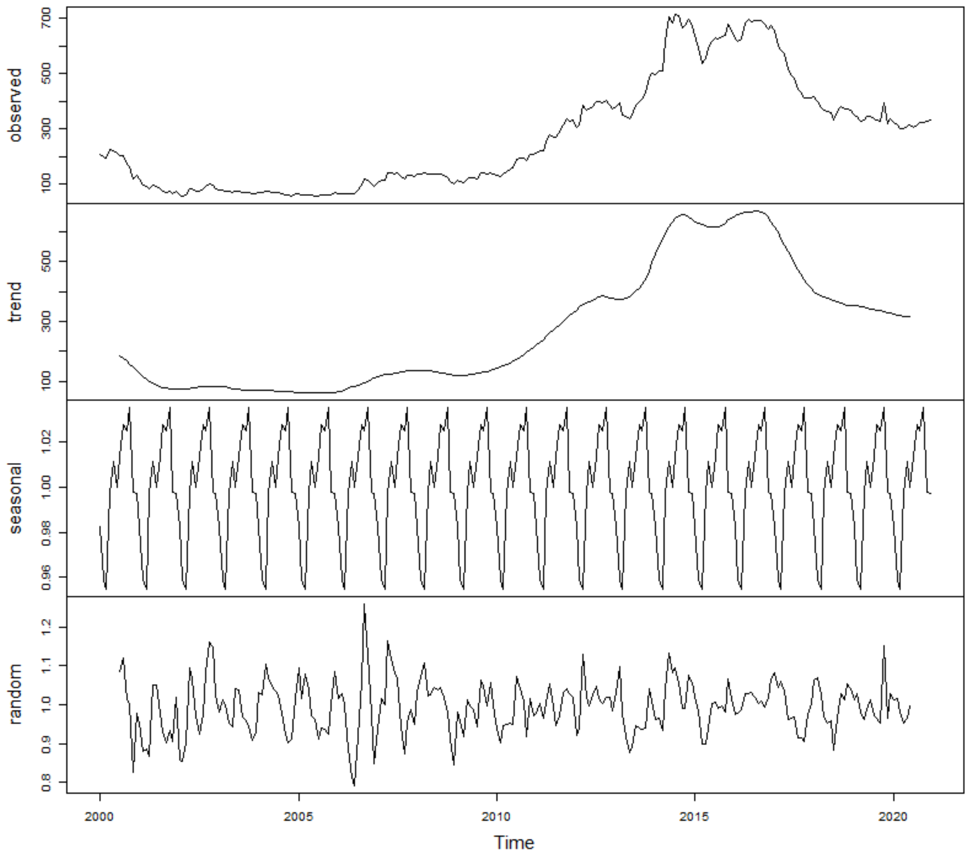


Figure 4.24 Decomposition for monthly price of ungarbled black pepper

4.2.3.1 Seasonal indices for monthly prices of ungarbled black pepper

Decomposition for the monthly prices of ungarbled black pepper during January 2000-December 2020 indicated the seasonality pattern in every year for the Kochi market. Hence, seasonal indices were computed for prices of black pepper for Kochi market, to get a clear picture of price pattern in the 12 months from January to December. Seasonal indices are provided in Table 4.23 and seasonal plot is shown in Figure 4.25.

Table 4.23 Seasonal indices for monthly price of ungarbled black pepper

Month	Seasonal indices (%) of black pepper price
January	98.26
February	97.40
March	94.21
April	100.86
May	100.69
June	100.22
July	100.90
August	100.90
September	100.27
October	102.73
November	100.70
December	101.11

The monthly seasonal indices for prices of black pepper from January to December indicate fluctuations in the market throughout the year. The months of January, February, and March exhibit lower seasonal indices, indicating relatively lower prices during this period. This corresponds with the flowering stage of black pepper in May and June, where the vines begin yielding in the third year but reach full bearing capacity in the sixth or seventh year.

The seasonal indices start to rise in April and continue to increase through May, June, and July, reaching a peak in October. This corresponds with the harvest season, which typically starts in November and extends to March. The higher indices in the months of October and November suggest elevated prices during the harvest period, aligning with the increased supply of black pepper in the market.

The indices for November, December, and January indicate that the prices remain relatively stable during the initial months of harvesting, which aligns with the information that harvesting alone occurs in November or December. The indices also reflect that from January to March, when both harvesting and arrivals in the market are taking place, the prices remain relatively stable or exhibit a slight increase, with indices for February and March being 100.70 per cent and 100.90 per cent respectively.

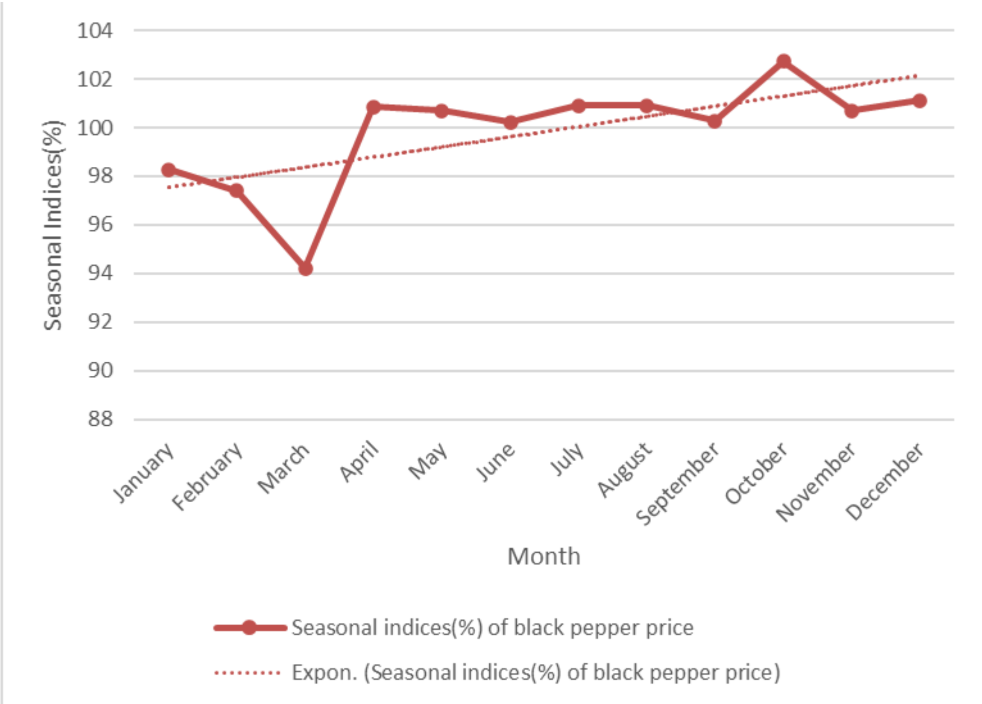


Figure 4.25 Seasonal plot for monthly price of ungarbled black pepper

4.2.3.2 Cyclical variation for monthly price of ungarbled black pepper

The cyclical pattern for monthly price of garbled black pepper is depicted in Figure 4.20. The first 11-year cycle was from 1983 to 1993 and the second cycle from 1993, showed some fluctuation near the peak values and reached the lowest value in 2004 (Sabu, 2015). The third cycle started from 2005 when the prices started gradually increasing until reaching a peak at Rs. 373.11, suggesting a period of increased demand or limited supply. Following this peak, the prices experience a downward trend, reaching a low of Rs.315.96, possibly influenced by factors such as increased production or changes in market dynamics.

The series continues with a subsequent increase in prices, reaching another peak at Rs.668.53. This indicates a new phase in the cycle, possibly influenced by improved market conditions or other external factors. The final data points show a decrease in prices, reaching Rs.398.05, suggesting a decline in market value.

The observed fluctuations in the price of ungarbled black pepper highlight the inherent volatility in the market, influenced by factors such as supply and demand dynamics, weather conditions affecting production, and broader economic factors.

4.2.3.3 Irregular variation for monthly price of ungarbled black pepper

The random effect is the residual effect after the trend, seasonal and cyclical effects have been removed from the original observations. As observed from the fourth panel of Figure 4.24, monthly price of ungarbled black pepper exhibited significant irregular variations during 2000 to 2020. They represent random effect such as demand and supply shocks on account of climatic aberrations or due to speculative factors.

4.2.4 Forecast for monthly price of ungarbled black pepper

A series of models comprising exponential smoothing model, ARIMA/SARIMA, ARCH/GARCH and ANN were fitted to forecast monthly price of ungarbled black pepper. The best model was selected based on the forecast accuracy measure: MAPE and RMSE. The results are provided in following subsections:

4.2.4.3 Exponential smoothing models for monthly price of ungarbled black pepper

Holt-Winters' Multiplicative Seasonal (HWMS) model was identified best among the different exponential smoothing models like SES, DES, HWAS and HWMS, for the monthly price of ungarbled black pepper based on criteria like agreement between observed and fitted price plots, MAPE and RMSE. The fit of the HWMS model of monthly price of ungarbled black pepper is provided in Figure 4.26 and it is evident that the actual and model fit values are in close agreement. The estimates of parameters of HWMS model are provided in Table 4.24. Various accuracy measures for HWMS model are provided in Table 4.25.

Table 4.24 Estimates of parameters for the HWMS model for monthly price of ungarbled black pepper

Parameter	Α	β	γ
Estimate	0.5821	0.1106	0.054

With these values for the parameters, HWMS model for black pepper price in Kochi market are as given below,

Level: L_t= 0.5821* $(\frac{Y_t}{S_{t-12}})$+.0.4179*(L_{t-1}– 14.2895)

Trend: b_t= 0.1106(L_t − L_{t-1})– 13.2895

Seasonality: S_t= S_{t-12}

Forecast: F_{t+m} = (L_t + b_{tm})S_{t-12+m}

The values of L_t, b_t and S_t were substituted in the forecast equation F_{t+m} to obtain the price.

Table 4.25 Model accuracy measures by HWMS model for monthly price of ungarbled black pepper

Accuracy measure	**Value**
RMSE	21.11
MAPE	**6.50**

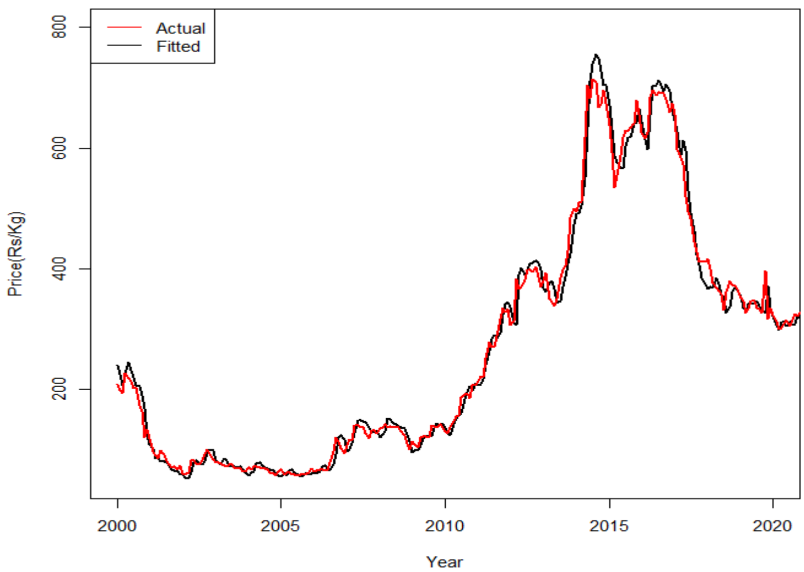


Figure 4.26 Actual and fitted plot for HWMS model for monthly price of ungarbled black pepper

The residual plot of monthly price of ungarbled black pepper from HWMS model is given in Figure 4.27 along with the ACF and PACF residual plots (Figure 4.28). It showed that most of the ACF and PACF values lie within the confidence limits indicating the adequacy of the model.

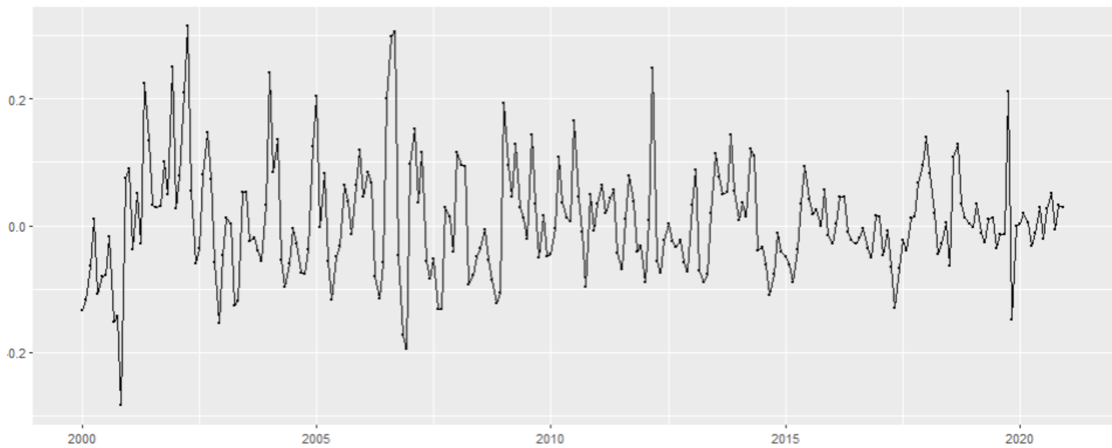


Figure 4.27 Residual plot for monthly price of ungarbled black pepper for HWMS model

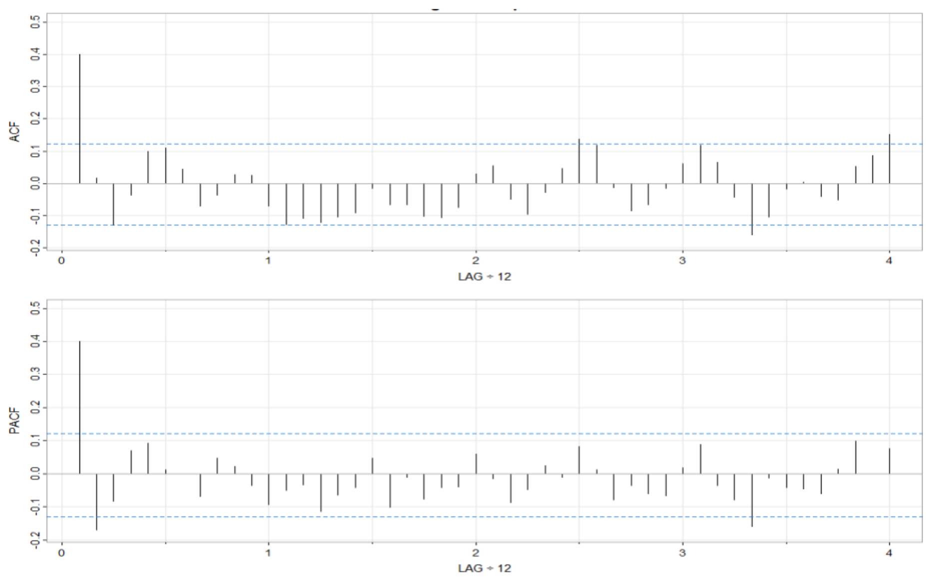


Figure 4.28 Residual ACF and PACF plots for monthly price of ungarbled black pepper for HWMS model

4.2.4.2 ARIMA model for monthly price of ungarbled black pepper

The time plot of monthly price for ungarbled black pepper indicates that it is non-stationary. Autocorrelations and partial autocorrelation were computed for monthly price of ungarbled black pepper which are provided in Table 4.27. Significance of autocorrelations upto 16 lags confirm the non stationarity of the data and partial autocorrelation value has very high value for lag 1.

Table 4.26 ACF and PACF values for monthly price of ungarbled black pepper

			Ljung-Box Statistic				
Lag	**Auto-correlation**	**SE**	**Value**	**Df**	**Probability (p)**	**Partial Auto-correlation**	**SE**
1	0.995	0.063	252.36	1	<0.001	0.995	0.063
2	0.988	0.063	502.24	2	<0.001	-0.162	0.063
3	0.981	0.063	749.45	3	<0.001	-0.014	0.063
4	0.973	0.063	993.62	4	<0.001	-0.073	0.064
5	0.964	0.063	1234.49	5	<0.001	-0.031	0.064
6	0.954	0.063	1471.49	6	<0.001	-0.106	0.064
7	0.943	0.063	1703.77	7	<0.001	-0.145	0.064
8	0.931	0.063	1931.29	8	<0.001	0.034	0.064
9	0.920	0.063	2154.21	9	<0.001	0.035	0.064
10	0.909	0.063	2372.69	10	<0.001	0.037	0.064
11	0.897	0.063	2586.58	11	<0.001	-0.038	0.064
12	0.886	0.063	2795.75	12	<0.001	-0.007	0.065
13	0.873	0.063	2999.96	13	<0.001	-0.045	0.065
14	0.861	0.063	3199.22	14	<0.001	-0.006	0.065
15	0.841	0.063	3393.55	15	<0.001	-0.027	0.065
16	0.837	0.063	3583.52	16	<0.001	0.117	0.065

$p<0.01$ indicates significance of autocorrelation

The ACF and PACF plots for monthly price of ungarbled black pepper is depicted in Figure 4.29. It is evident that in the ACF plot, spikes upto 16 lags fall above confidence limit and PACF plot showed spikes for a number of lags (1, 2 and 6) beyond confidence limits also indicated the non-stationarity of the time series.

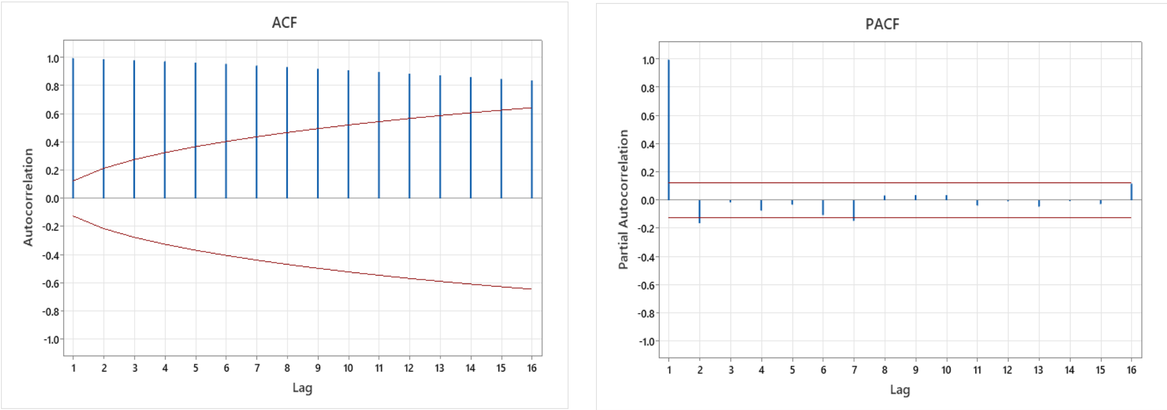


Figure 4.29 ACF and PACF plots for monthly price of ungarbled black pepper

The stationarity of the data was evaluated using ADF test. Following the initial differencing of the data, the ADF test was repeated. The significance of the ADF test statistic, as indicated in Table 4.27, affirmed the necessity of taking the first difference to achieve stationarity in the data.

Table 4.27 ADF test with critical values for monthly price of ungarbled black pepper

Ungarbled black pepper price	ADF test statistic	Probability(p)	Critical values
Actual	-2.0796	0.5423	0.106
First difference	-9.9913**	0.01	-2.32

** indicates significant at 5 per cent level($p<0.05$)

Among several ARIMA models tried, the tentative models were chosen based on the value of MAPE, AIC and RMSE. SARIMA (2,1,2) $(2,0,2)_{12}$ was chosen as the best forecast model for monthly price of ungarbled black pepper.

Table 4.28 Model accuracy measures by SARIMA (2,1,2) $(2,0,2)_{12}$ model for monthly price of ungarbled black pepper

Accuracy measures		
RMSE	**MAPE**	**AIC**
18.71	5.07	2209.02

The parameters of the model SARIMA$(2,1,2)(2,0,2)_{12}$ along with their tests of significance are provided in Table 4.29.

Table 4.29 SARIMA$(2,1,2)(2,0,2)_{12}$ model parameters for monthly price of ungarbled black pepper

Model Parameters	Estimate	SE	Z value	Probability
Non-seasonal difference	1			
AR Lag 2 (ϕ_2)	0.468**	0.259	1.809	0.034
MA Lag 2(θ_2)	-0.499**	0.219	-2.271	0.023
Seasonal difference	0			
AR Seasonal Lag 1(Φ_1)	-1.046***	0.250	-4.170	0.00
AR Seasonal Lag 1(Φ_2)	-0.81**	0.364	-2.222	0.026
MA Seasonal Lag 1(Θ_1)	1.077***	0.307	3.508	0.000
MA Seasonal Lag 2(Θ_2)	0.938**	0.474	1.978	0.047

**indicates significance at 5 per cent percentage level($p<0.05$)

***indicates significance at 1 per cent percentage level ($p<0.01$)

The general form of the SARIMA$(2,1,2)(2,0,2)_{12}$ model equation is as follows:

$$(1-\phi_2 B^2)(1-B)(1-\Phi_1 B^{12}- \Phi_2 B^{12})y_t = (1-\theta_2 B^2)(1-\Theta_1 B^{12}- \Theta_2 B^{12})\epsilon_t \ , y_t = \log_e Y_t$$

Where, $\phi_2 = 0.468$, $\Phi_1= -1.046$, $\Phi_2 =-0.81$, $\theta_2=-0.499$,

$\Theta_1=1.1077$, $\Theta_2=0.938$

The SARIMA$(2,1,2)(2,0,2)_{12}$ model equation is as follows:

$$(1-0.468B^2)(1-B)(1+1.046B^{12}+0.810B^{12})y_t=$$

$$(1+0.499B^2)(1-1.077^{12}- 0.938B^{12})\epsilon_t$$

The plot of actual and fit values for monthly prices of ungarbled black pepper using SARIMA$(2,1,2)(2,0,2)_{12}$ model is given in Figure 4.30.

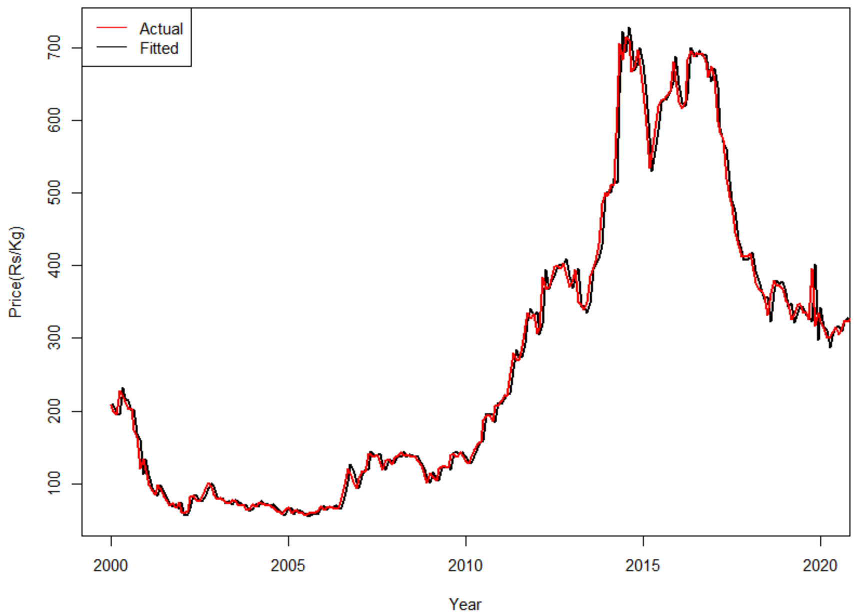


Figure 4.30 Actual and fitted plot for SARIMA(2,1,2) $(2,0,2)_{12}$ for monthly price of ungarbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic is provided in Table 4.30 and was found to be insignificant. So, overall, we can say SARIMA(2,1,2)$(2,0,2)_{12}$ model shown satisfactory result, among different ARIMA models.

Table 4.30 Ljung-Box 'Q' statistic for residuals of SARIMA(2,1,2)$(2,0,2)_{12}$ model

Statistic	p-value
16.918	**0.3909^{NS}**

NS: Non-significant

Residual ACF and PACF plots for the SARIMA(2,1,2)$(2,0,2)_{12}$ model fitted for monthly price of ungarbled black pepper price is provided in Figure 4.31. It could be seen that majority of the spikes in the ACF and PACF plots fall within the critical values. This shows the adequacy of SARIMA(2,1,2)$(2,0,2)_{12}$ model for forecasting monthly price of ungarbled black pepper price.

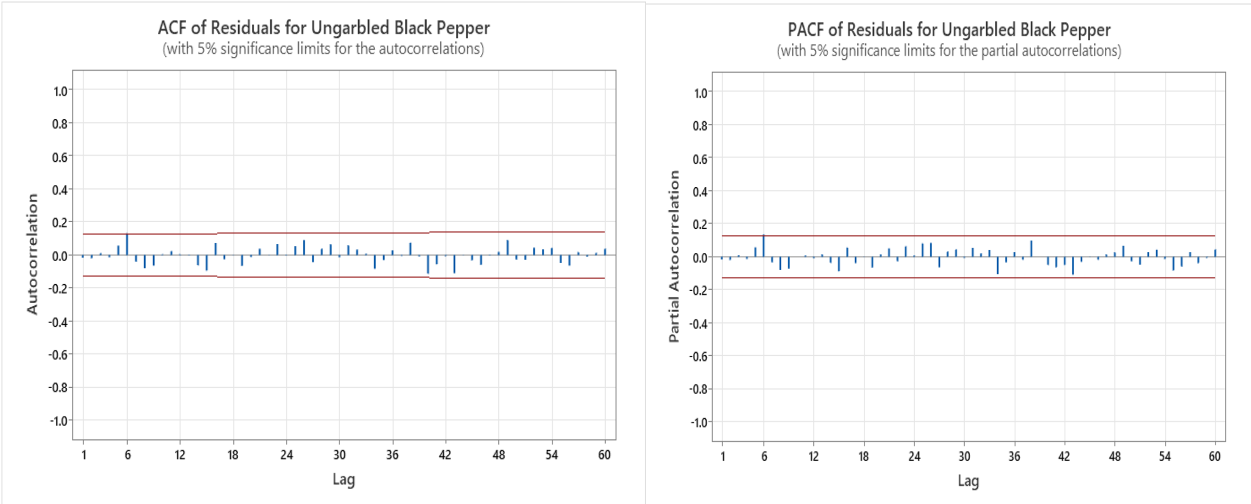


Figure 4.31 Residual ACF and PACF plots for monthly price of ungarbled black pepper for SARIMA(2,1,2) $(2,0,2)_{12}$

4.2.4.3 Volatility in monthly price of ungarbled black pepper

The monthly price of ungarbled black pepper for the period 2000-2020 was used to choose the ARCH family model using R software.

The Heteroscedasticity LM test was carried out to check the volatility or ARCH effect in the time-series. The results of the test are presented in Table 4.31, which reveals that there is an ARCH effect in the time series.

Table 4.31 Heteroscedasticity LM Test for first differenced

Statistic	p-value
20.849***	.000

*** Significant at 1 per cent level of significance

Several ARCH models were tested, but their coefficients were determined to be insignificant, highlighting the need for the implementation of the GARCH model.

4.2.4.3.1 GARCH(1,1) model for monthly price of ungarbled black pepper

The estimates of the GARCH (1,1) model fitted for the monthly price of ungarbled pepper is given in the Table 4.32. It showed that the constant term, ARCH and GARCH parameters are positive and significant indicating the volatility.

Table 4.32 Estimates of GARCH(1,1) model for monthly price of ungarbled black pepper

Parameter	Coefficient	Std error	t statistic	p-value
Constant term(α_0)	0.0010**	0.004	2.025	0.042
ARCH term(α_1)	0.2484**	0.0634	2.339	0.019
GARCH term(β_1)	0.6012***	0.1277	5.216	0.00

**Significant at 5 per cent level of significance

*** Significant at 1% level of significance

The GARCH(1,1) model is given by,

$$h_t = 0.0010 + 0.2484\varepsilon_{t-i}{}^2 + 0.6012\,\sigma_{t-j}^2$$

The time varying volatility includes a constant (0.0010), a component which depends on past errors ($0.2484\varepsilon_{t-i}{}^2$) and a component which depends on weighted average of past squared residuals ($0.6012\sigma_{t-j}^2$). In the Table 4.32, the p-value for the t-statistic of the first order coefficient (2.33) and the second order coefficient (5.21) suggests a significant GARCH (1,1) coefficient.

Table 4.33 Model accuracy measures by GARCH(1,1) model for monthly price of ungarbled black pepper

Accuracy measure	Value
RMSE	19.52
MAPE	5.63

Residual analysis was carried out to check the adequacy of the selected model. The Serial Correlation LM test for residuals is presented in Table 4.34. The large value of p (p=0.141>0.05) reveals that, there is no serial correlation in the residuals. The Ljung-Box test result is presented in Table 4.35. The large value of p (p=0.481 > 0.05) with respect to Ljung-Box 'Q' statistic indicates that the residuals are normally distributed.

Table 4.34 Serial Correlation LM test for residuals of GARCH(1,1) Model

Statistic	p-value
6.778	**0.141^{NS}**

NS: Non-significant

Table 4.35 Ljung-Box test for residuals of GARCH(1,1) Model

Statistic	p-value
49.85	**0.481^{NS}**

NS: Non-significant

4.2.4.4 ANN model for monthly price of ungarbled black pepper

Time delayed neural network (TDNN) model was fitted for monthly price of ungarbled black pepper. The best time lagged neural network with single hidden layer was found for each series by conducting experiments with the basic cross validation method. Out of a total of 36 neural network structures, a neural network model with six lagged observations as input nodes and three hidden nodes (6:3s:1l) performed better than other competing models in respect of forecasting accuracy measures. This means that most accurate price forecast for the given series is obtained when the price of six preceding months is used as inputs.

The selected TDNN model is described in Table 4.36 along with the forecasting accuracy measures for both training and testing set.

Table 4.36 Model accuracy measures by TDNN model for monthly price of ungarbled black pepper

Model	No. of parameters	MAPE		RMSE	
		Train	Test	Train	Test
6:3s:11	25	5.96	4.63	12.75	10.89

The actual monthly price along with the predicted values of monthly price of ungarbled black pepper using TDNN model is provided in Figure 4.32.

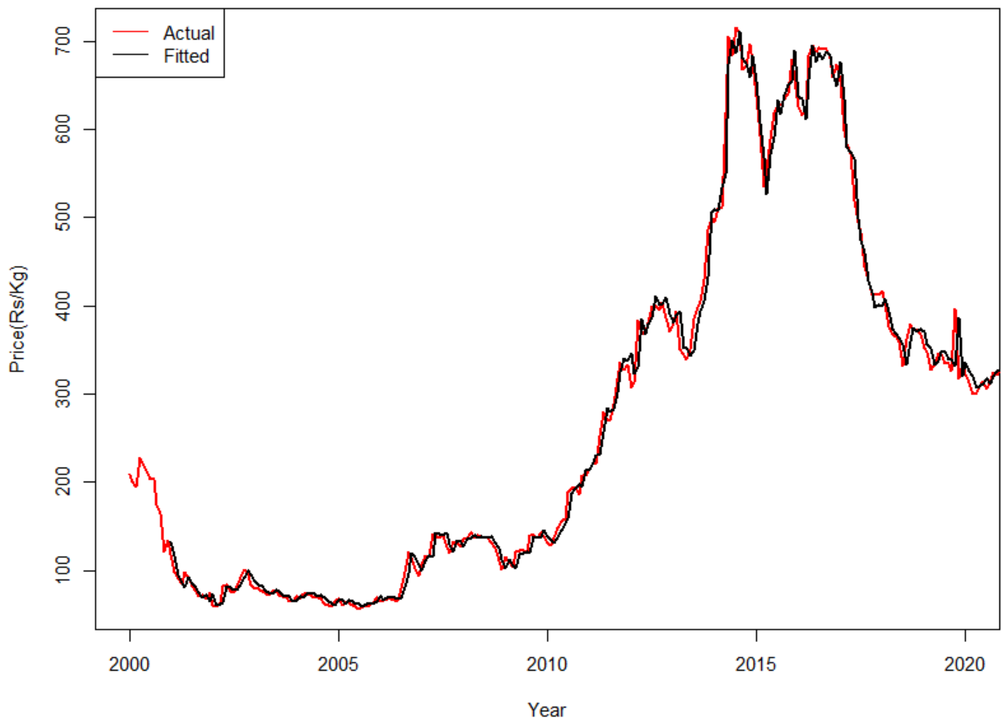


Figure 4.32 Actual and fitted plot for TDNN model for monthly price of ungarbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant provided in Table 4.37. So, overall, we can say TDNN model shows satisfactory result.

Table 4.37 Ljung-Box ‘Q’ statistic for residuals of TDNN model

Statistic	p-value
14.536	0.9336^{NS}

NS: Non-significant

The residual plot from TDNN model is provided in Figure 4.33 which did not exhibit any specific pattern. The residual ACF and PACF plot is provided in Figure 4.34, majority of the spikes in the residual ACF and PACF are within the critical values indicating the adequacy of the model.

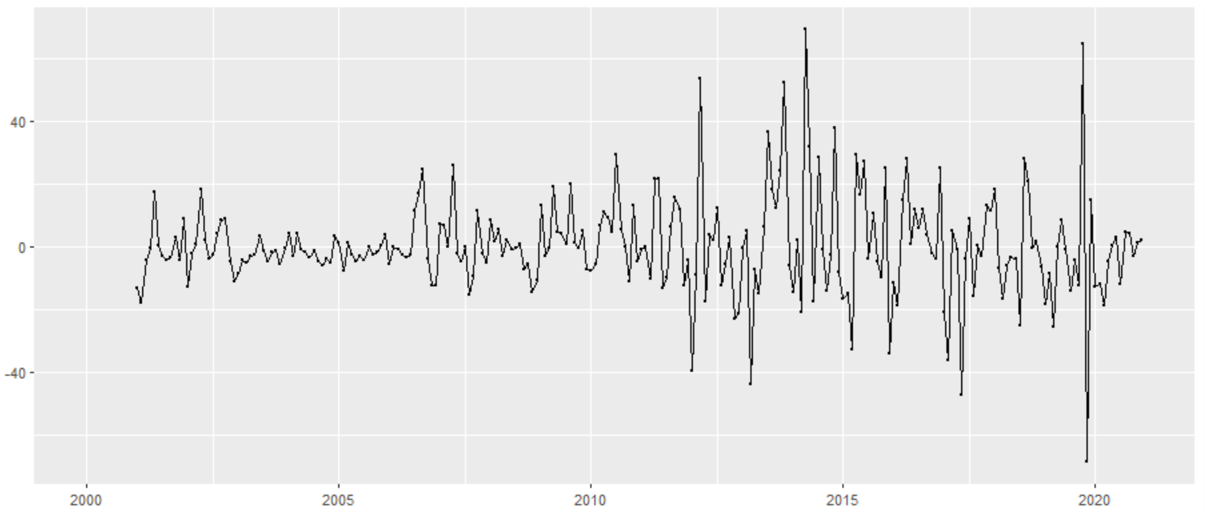


Figure 4.33 Residual plot for monthly price of ungarbled black pepper for TDNN model

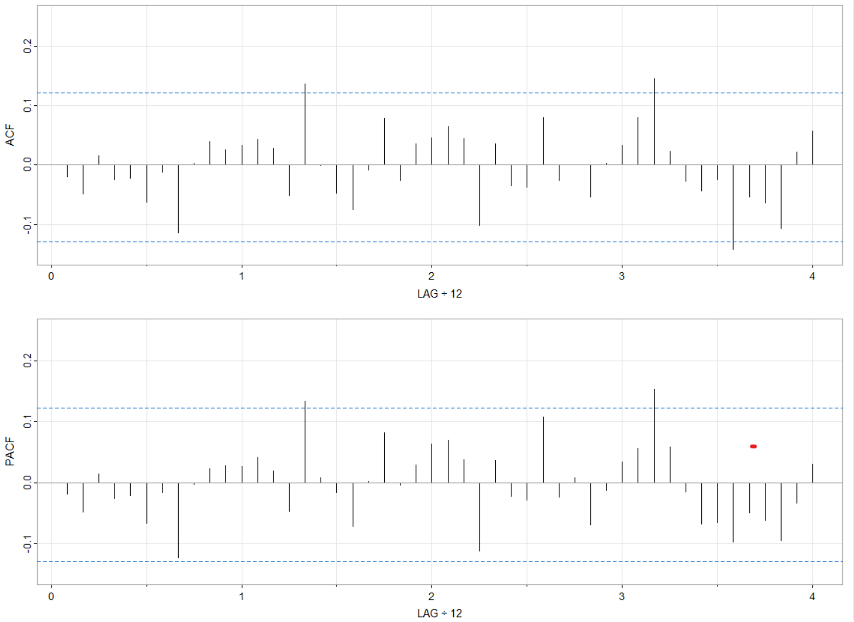


Figure 4.34 Residual ACF and PACF plots for monthly price of ungarbled black pepper for TDNN model

4.2.4.5 LSTM model for monthly price of ungarbled black pepper

LSTM model was fitted for monthly price of ungarbled black pepper price. The parameters used for the model was provided in Table 3.1. The LSTM model has been created with three layers by performing epoch of 50 times. During the training process, the model makes predictions on the training data, and the loss function is used to calculate the error between these predictions and the actual target values. The training loss is the average of these errors across all training samples. The goal of training is to adjust the weights and biases of the model to minimize this training loss. After each epoch, the weights of the model get updated and the new epoch works on those updated values that process continues in every epoch. In this study, MAE has been used as the measure to continue the epoch until MAE reaches a minimum which is shown in Figure 4.35.

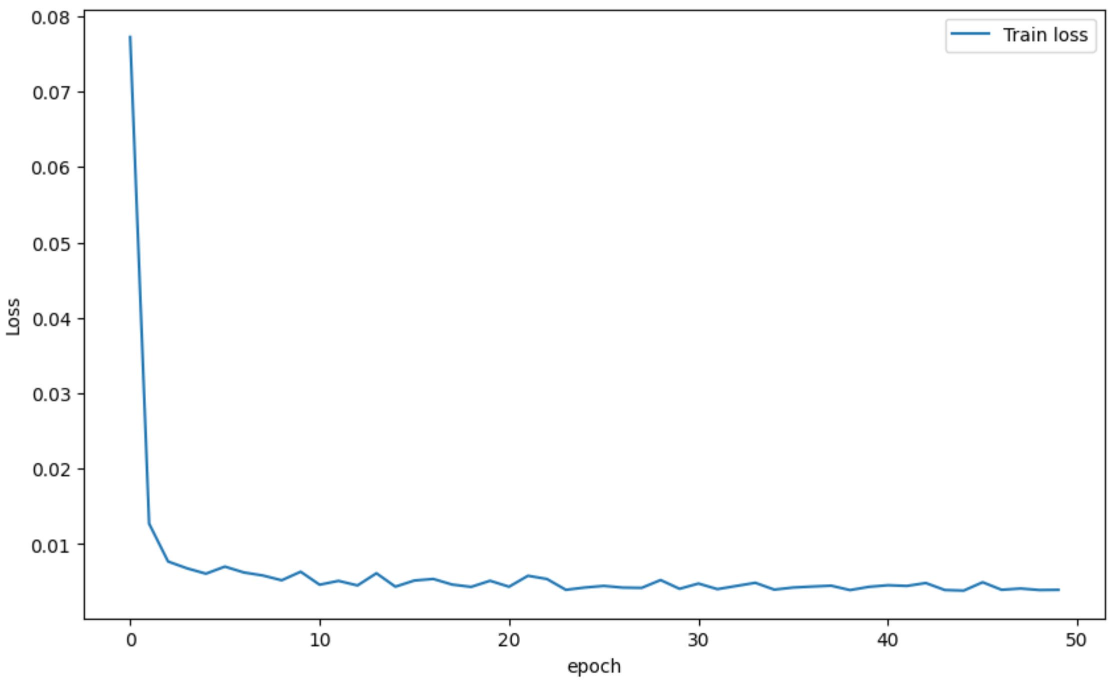


Figure 4.35 Training loss for LSTM model for monthly price of ungarbled black pepper

Various accuracy measures for LSTM model are provided in Table 4.38.

Table 4.38 Model accuracy measures by LSTM model of monthly price of ungarbled black pepper

Accuracy measure	Value
RMSE	28.49
MAPE	**8.24**

The actual price along with the predicted values for monthly price of ungarbled black pepper using LSTM model is provided in Figure 4.36.

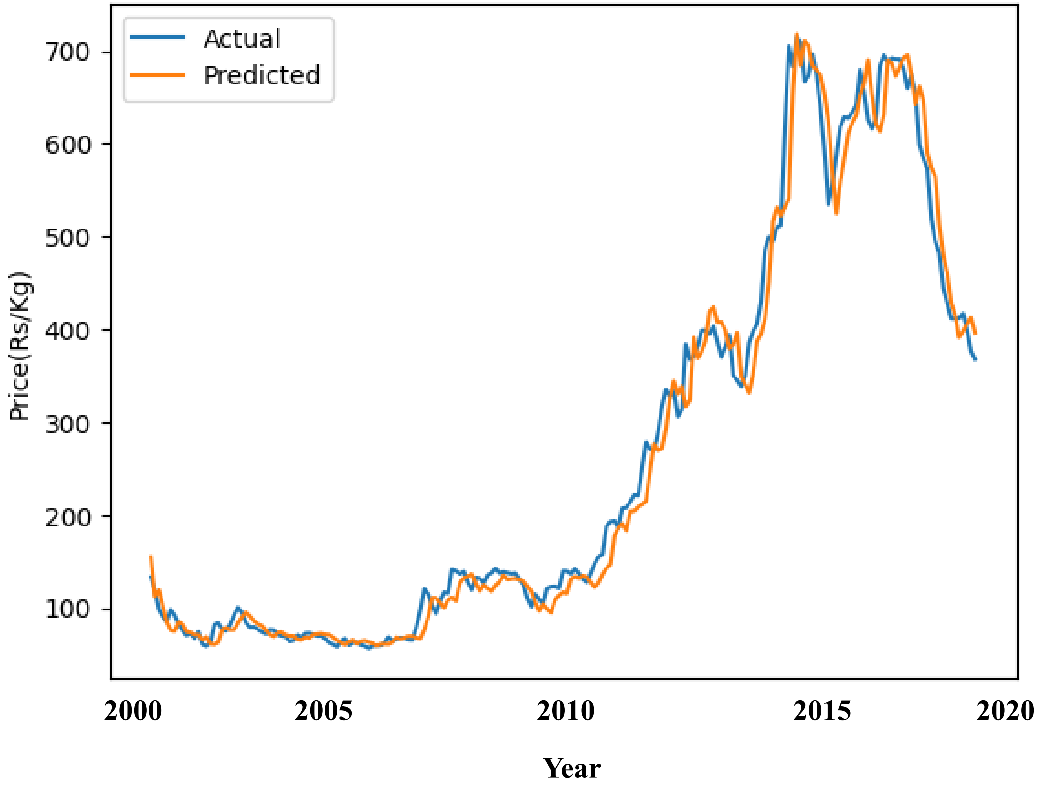


Figure 4.36 Actual and fitted plots for LSTM model for monthly price of ungarbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant. So, we can say LSTM model shows satisfactory result.

Table 4.39 Ljung-Box 'Q' statistic for residuals of LSTM model

Statistic	p-value
78.61	0.435^{NS}

NS: Non-significant

The residual plot from LSTM model is provided in Figure 4.37 and the residuals did not exhibit any specific pattern, they are scattered.

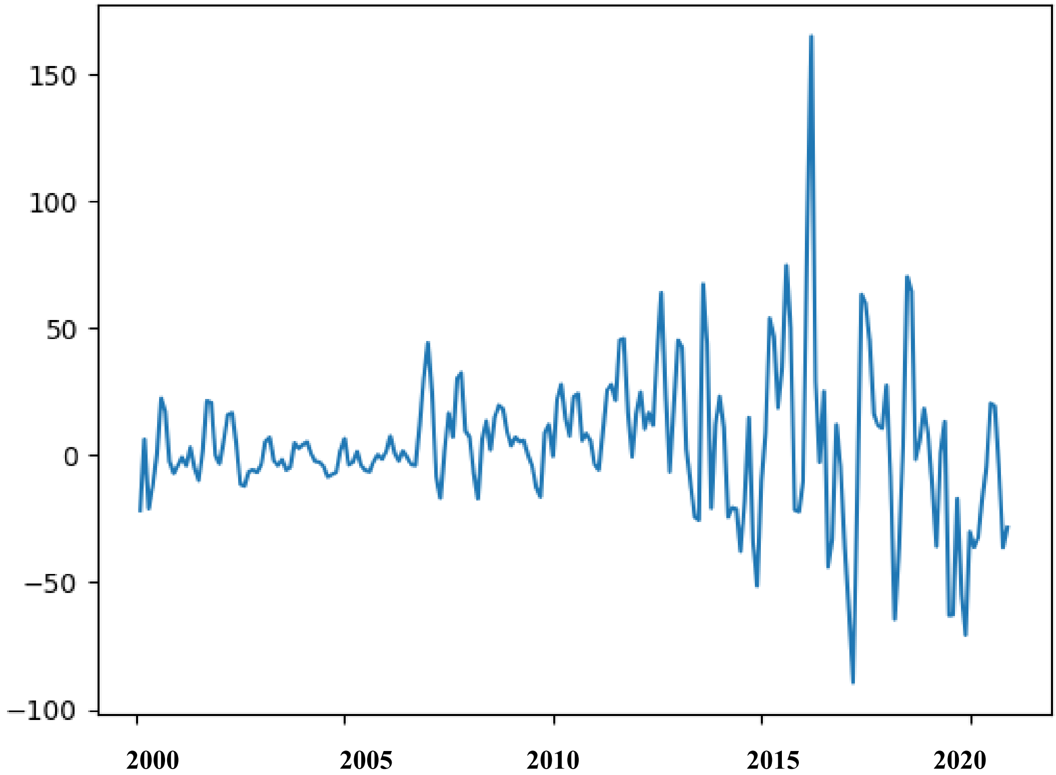


Figure 4.37 Residual plot for monthly price of ungarbled black pepper for LSTM model

4.2.5 Comparison of models

In present study, a comparison of different models has been done in order to know the best model for forecasting monthly price of ungarbled black pepper in Kochi market. The comparison of all the 6 models were carried out based on the test MAPE and RMSE values which were considered to be least. According to the Table 4.40, the TDNN model with least MAPE and RMSE values was considered to be best among all the models considered.

Table 4.40 Comparison of time series forecasting models for monthly price of ungarbled black pepper

Model	MAPE	RMSE
Exponential trend model	39.9	102.47
SARIMA(2,1,2)(2,0,2)12	5.07	18.71
GARCH(1,1)	5.63	19.52
HWMS	6.50	21.11
TDNN(6:3s:1l)	**4.63**	**10.89**
LSTM	8.24	28.49

4.2.5.1 Validation of forecasted monthly price for ungarbled black pepper using TDNN model

Below Table 4.41 and Table 4.42 shows the actual price (Rs/kg), forecasted price (Rs/kg) and their deviations (in per cent) for the monthly price of garbled black pepper for the next two years based on the TDNN model. The MAPE were found to be 4.09 and 5.05 respectively based on the actual and forecasted price of garbled black pepper using the TDNN model for the year 2021 and 2022.

Table 4.41 Forecasted monthly price for ungarbled black pepper from TDNN model for the year 2021

Month	Actual price(Rs/kg)	Forecast price(Rs/kg)	Deviations (in per cent)
January	327	336.96	3.04
February	326.75	345.58	5.76
March	353.5	375.34	6.17
April	381	389.44	2.21
May	385	392.47	1.94
June	401.5	398.21	0.81
July	398.8	400.75	0.48
August	395	402.14	1.80
September	398	406.32	2.09
October	402	412.22	2.54
November	492.5	445.43	9.55
December	515.6	450.17	12.69

Table 4.42 Forecasted monthly price for ungarbled black pepper from TDNN model for the year 2022

Month	Actual price (Rs/kg)	Forecast price (Rs/kg)	Deviations (in per cent)
January	499.5	467.28	6.45
February	498.75	470.84	5.59
March	512.25	476.01	7.07
April	513	485.18	5.42
May	504.25	488.72	3.07
June	489	490.49	0.30
July	488	492.49	0.93
August	495	498.63	0.73
September	498.2	521.97	4.77
October	491	525.12	6.94
November	488	531.63	8.94
December	494	545.29	10.38

The plot of actual and fit values along with the forecasts for the monthly ungarbled black pepper price using TDNN model is given in Figure 4.36 and it was evident that actual and predicted prices are in agreement.

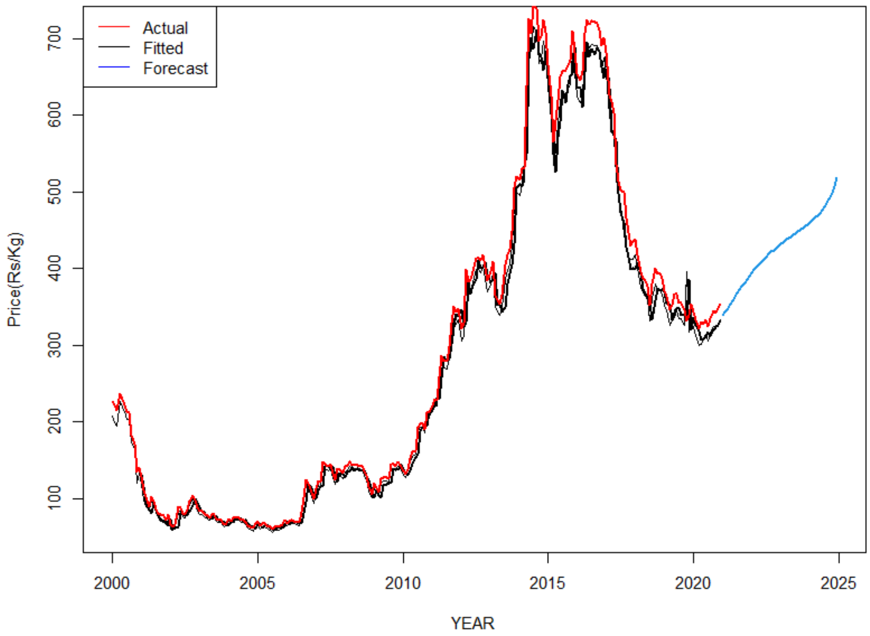


Figure 4.38 Actual, fitted and forecasted plot for TDNN model for monthly price of ungarbled black pepper

The TDNN model captures the pattern for the monthly price of ungarbled black pepper with the MAPE value 4.63. The residual plot from TDNN model is plotted in Figure 4.33 and was found to be scattered. Further, residual ACF and PACF plots for the TDNN model in Figure 4.34, exhibited that the majority of the fall within the critical values which shows the adequacy of the selected TDNN model.

4.3 Analysis of weekly price of garbled black pepper in the Kochi market

The results obtained from the analysis of weekly price data for garbled black pepper are presented below:

4.3.1 Pattern for weekly price data of garbled black pepper

The time plot for weekly average prices of garbled black pepper in the Kochi market from 2000 to 2020 were used to study the price pattern and it exhibited some notable patterns which is depicted in Figure 4.39.

It's essential to note that while these observations provide a broad overview of the price trends, the specific factors influencing pepper prices are multifaceted and can be influenced by various economic, agricultural, and geopolitical factors. Detailed analysis and external information about market dynamics would be required for a more in-depth understanding of the observed patterns.

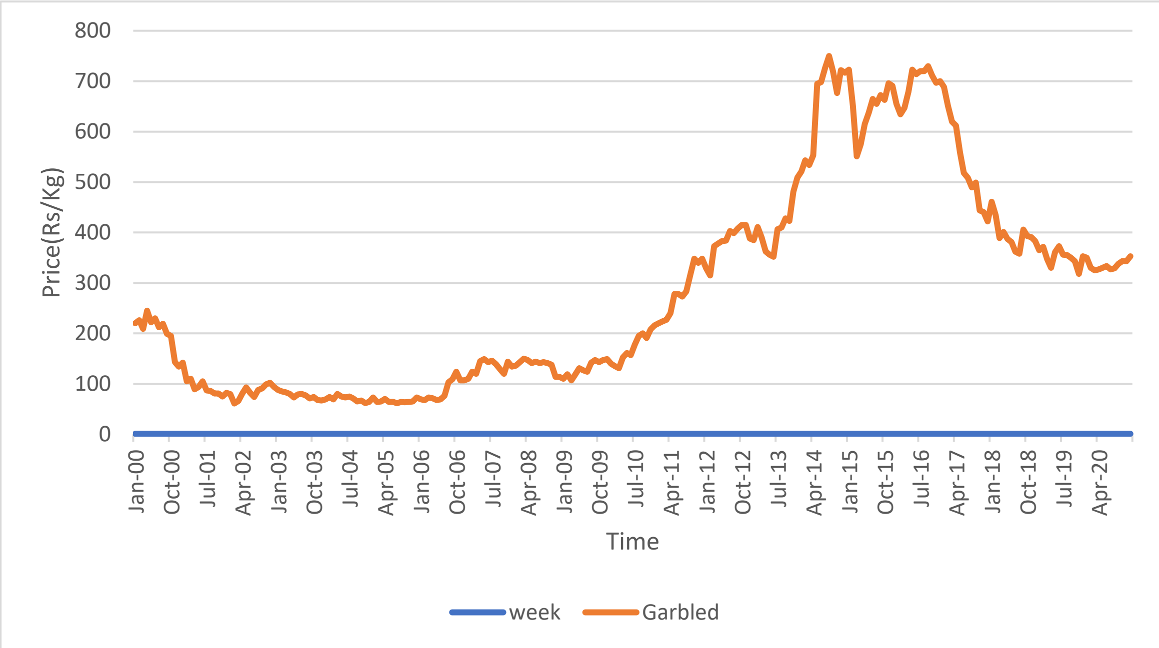


Figure 4.39 Price pattern for weekly price of garbled black pepper

4.3.2 Trend Analysis

Trend analysis was carried out for weekly average price of garbled black pepper. Different functional forms like linear, quadratic, exponential, etc. were tried and the suitable model in each case was chosen based on the MAPE and RMSE values.

The trend equation fitted for weekly price of garbled black pepper and their accuracy measures are provided in Table 4.43. Graphical plots revealing the pattern of the series based on the different trend models tried are plotted in Figure 4.40 – 4.42.

Table 4.43 Trend equations for weekly price of garbled black pepper

Functional Form	Trend equation	MAPE	RMSE
Linear Trend Model	$Y_t = 13.65 + 0.4947 \times t$	52.7	122.34
Exponential Trend Model	$\boldsymbol{Y_t = 67.4153 \times (1.00208\text{^}t)}$	**40.3**	**106.98**
Quadratic Trend Model	$Y_t = 2.3 + 0.5571 \times t - 0.000057 \times t^2$	53.9	123.72

As observed from the table, lower MAPE value of 40.3 and RMSE value of 106.98 shows the adequacy of the exponential trend model in explaining the trend of weekly price of garbled black pepper price.

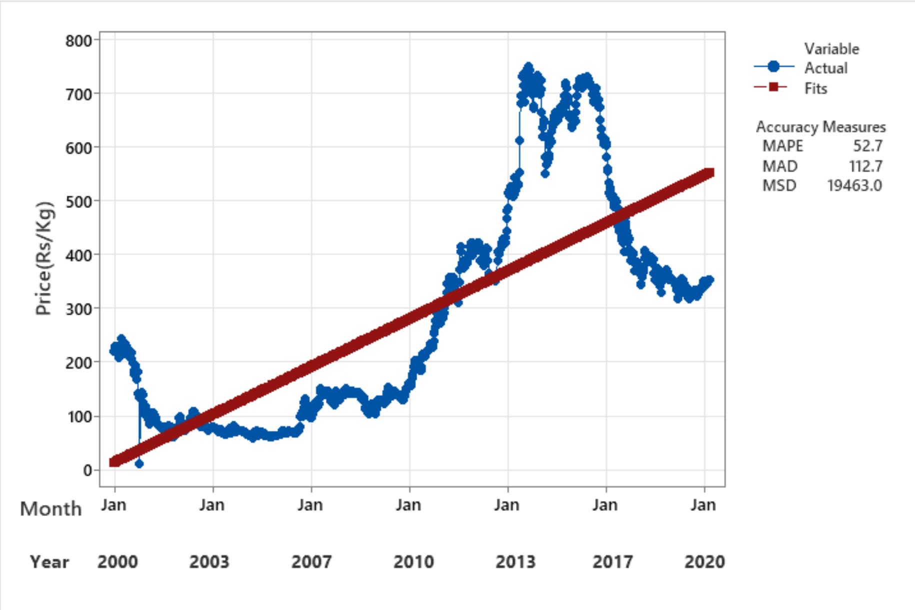


Figure 4.40 Linear trend plot for weekly price of garbled black pepper

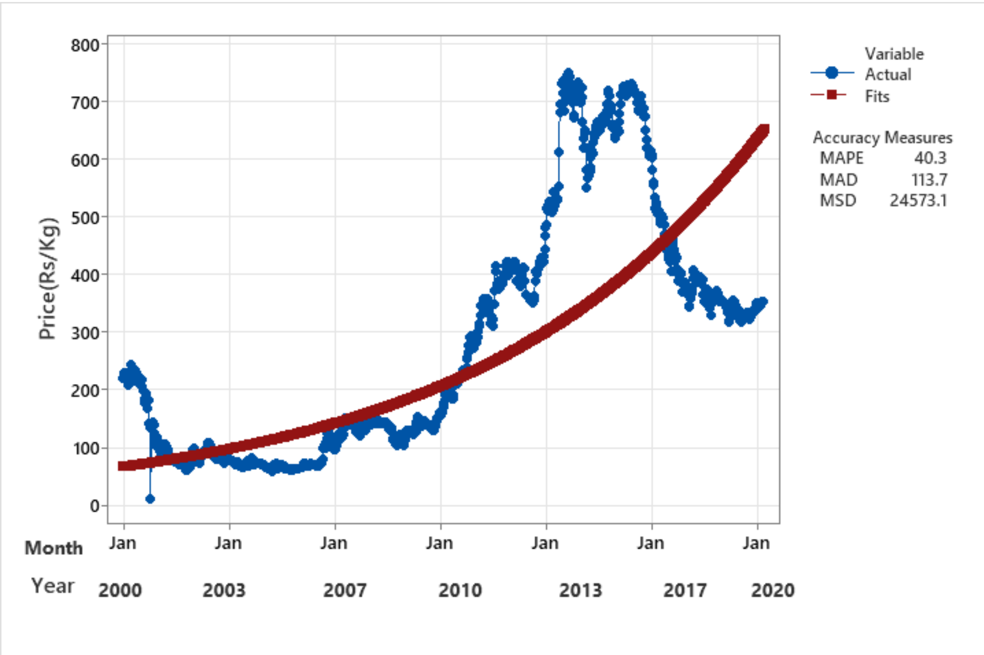


Figure 4.41 Exponential trend plot for weekly price of garbled black pepper

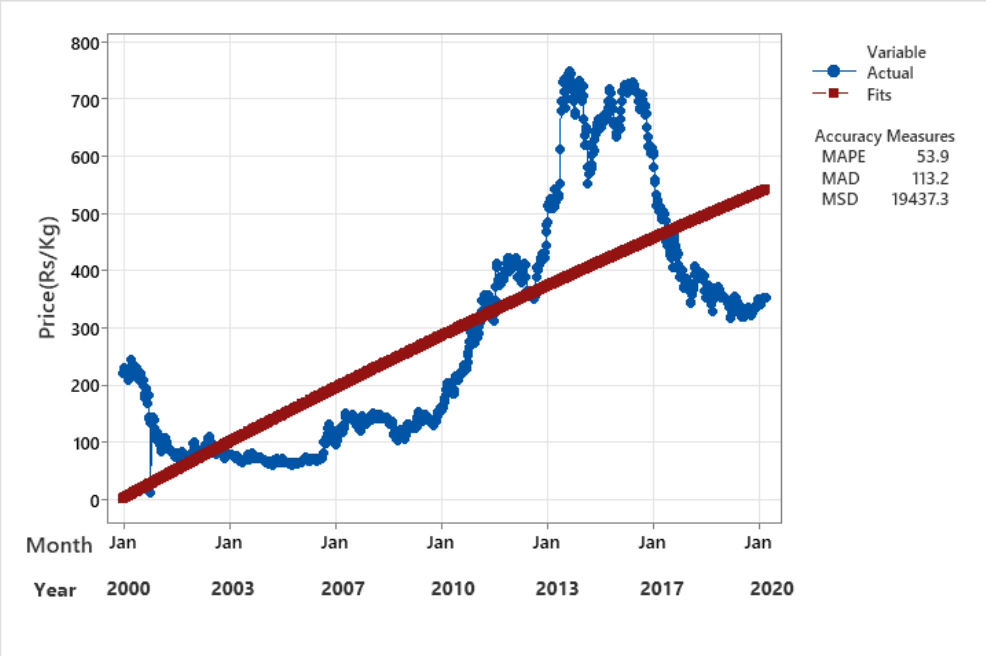


Figure 4.42 Quadratic trend model for weekly price of garbled black pepper

4.3.3 Decomposition for weekly price of garbled black pepper

The weekly price of garbled black pepper is decomposed into four components, as described in time series analysis: trend, seasonal variation, cyclic variation, and irregular variation using a multiplicative model. The result is presented in Figure 4.43, which consists of four panels. In the first panel, the observed weekly price of garbled black pepper has been plotted. The second panel illustrates the trend in the prices, while the third and fourth panels depict the seasonal and random variations, respectively. Notably, the weekly price pattern of garbled black pepper dataset reflects a dynamic economic landscape with periods of growth, decline, and high volatility.

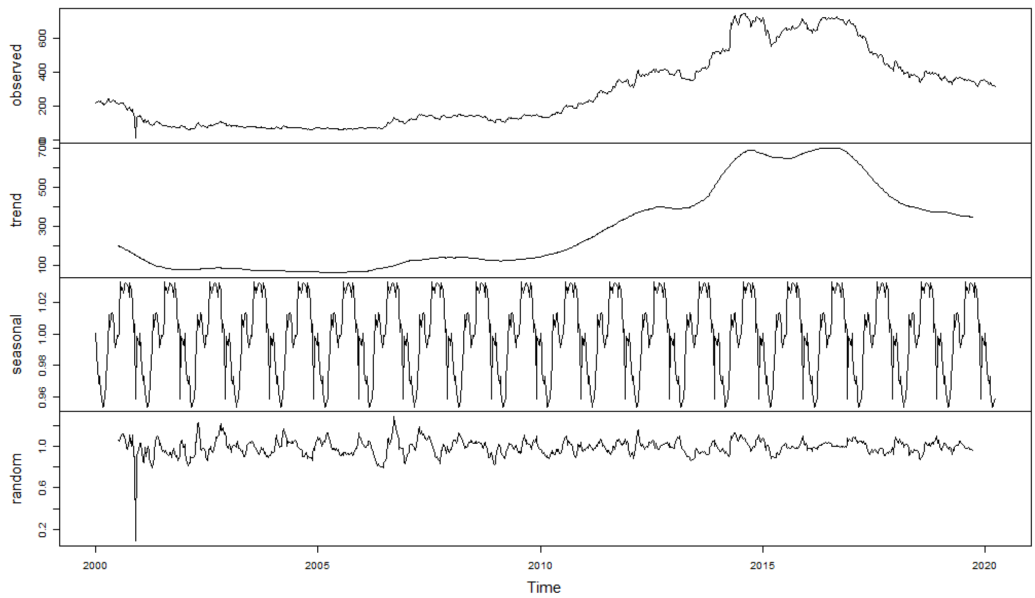


Figure 4.43 Decomposition for weekly price of garbled black pepper

4.3.3.1 Seasonal indices for weekly price of garbled black pepper

Decomposition of the time series data of weekly price for garbled black pepper during January 2000-December 2020 indicated the seasonality pattern in every year. Seasonal indices were calculated for 52 weeks (January to December) from the weekly price data of garbled black pepper from January, 2000 to December, 2020 to understand the seasonal behaviour of prices.

Seasonal indices are calculated by ratio to moving average for weekly price of garbled black pepper and corresponding seasonal plot is shown in Figure 4.44.

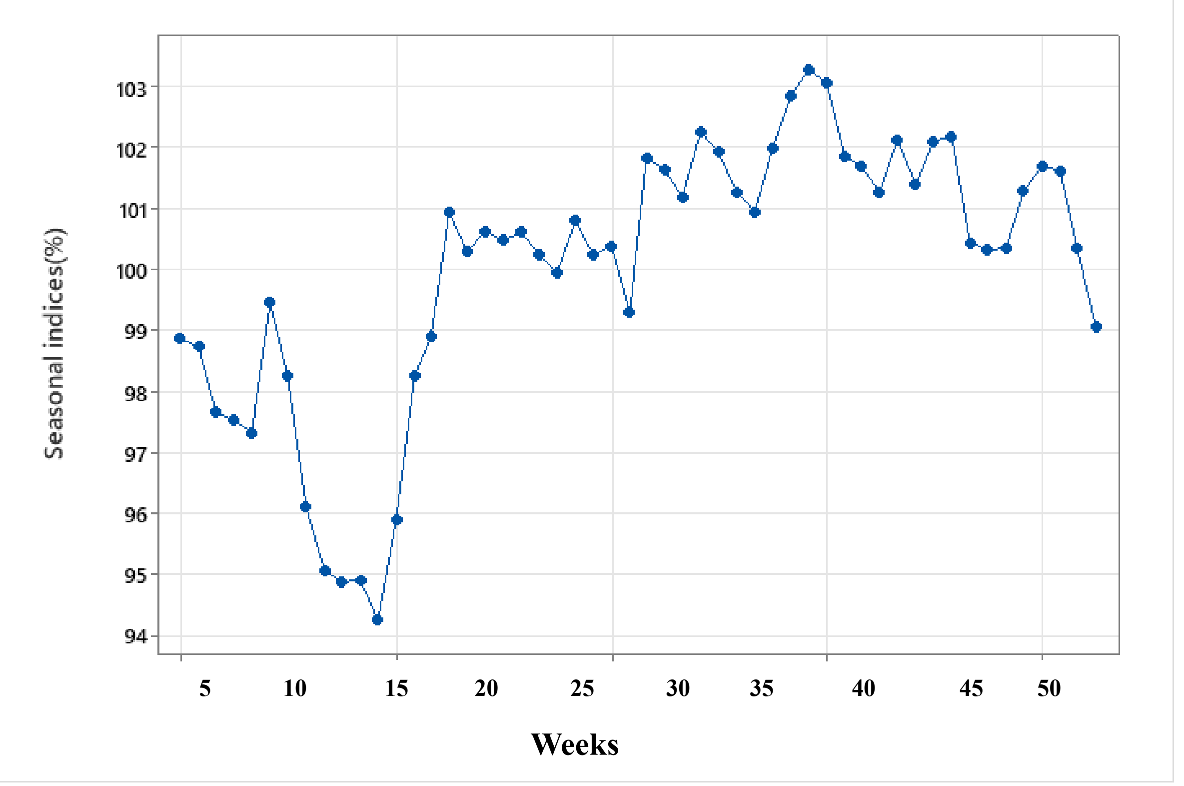


Figure 4.44 Seasonal plot for weekly price of garbled black pepper

The weekly seasonal indices for black pepper prices reveal a clear pattern of seasonality over the course of a year. Each week is associated with a specific seasonal index, indicating the relative strength of the observed seasonality compared to the average. The monthly seasonality, as described earlier, illustrates lower prices in January, February, and March, corresponding to the flowering stage of black pepper. This aligns with the beginning of the weekly observations, where first week of May shows a slightly below-average seasonal effect. As we progress through the weeks, the indices fluctuate, mirroring the variations in the seasonal patterns described on a monthly scale. For instance, third week of September and first week of November exhibit higher indices indicating stronger seasonality during these periods which corresponds to the harvest season. Conversely, first week of January show declining indices, reaching a low in third week of March reflecting a dip in seasonality, akin to the lower prices noted in the monthly analysis. This interconnection between the monthly and weekly seasonality indices provides a comprehensive understanding of how black pepper prices follow distinct patterns throughout the year, driven by the seasonal nature of its production.

4.3.3.2 Cyclical variation for weekly price of garbled black pepper

Price cycles indicate deviations in price levels from the average trend due to the recurring patterns of economic booms and recessions. These cycles, which span several years and exhibit various periodicities, are particularly evident in the production and pricing of black pepper.

The cyclical pattern for weekly price of garbled black pepper is depicted in Figure 4.39. The first 11 year cycle was from 1983 to 1993 and the second cycle from 1993, showed some fluctuation near the peak values and reached the lowest value in 2004 (Sabu, 2015). The third cycle started from 2005 when the prices started gradually increasing until reaching a peak in 2010. This upward trend is part of the cyclical variation, reflecting a period of growth and possibly influenced by factors such as increased demand, favourable weather conditions, or changes in market dynamics.

Subsequently, the prices experience a sharp spike in 2011indicating a peak in the cycle. This could be attributed to specific factors like a surge in demand, reduced supply, or other market forces. Following the peak, there is a noticeable downward trend in prices. This decline represents the trough of the cycle, possibly influenced by factors such as increased production, changes in consumer preferences, or economic conditions.

The cyclical variation continues as prices start to rise again in 2020. This upward movement suggests a new phase of the cycle, possibly influenced by factors such as improved market conditions, changes in agricultural practices, or global economic trends.

4.4.3.3 Irregular variation for weekly price of garbled black pepper

The random effect is the residual effect after the trend, seasonal and cyclical effects have been removed from the original observations. As observed from the fourth panel of Figure 4.43, weekly price of garbled black pepper exhibited significant irregular variations during 2000 to 2020. They represent random effect such as demand and supply shocks on account of climatic aberrations or due to speculative factors.

4.3.4 Forecast for weekly price of garbled black pepper

A series of models comprising exponential smoothing model, ARIMA/SARIMA, ARCH/GARCH and ANN were fitted to forecast weekly price of garbled black pepper. The best model was selected based on the forecast accuracy measure: MAPE and RMSE. The results are provided in following subsections:

4.3.4.1 Exponential smoothing model for weekly price of garbled black pepper

Holt-Winters' Multiplicative Seasonal (HWMS) model was identified best among the different exponential smoothing models like SES, DES, HWAS and HWMS, for the weekly price of garbled black pepper based on criteria like agreement between observed and fitted price plots, MAPE and RMSE. The fit of the HWMS model of weekly price of garbled black pepper is provided in Figure 4.45 and was found that the actual and model fit values are in close agreement. The estimates of parameters of HWMS model are provided in Table 4.44. Various accuracy measures for HWMS model are provided in Table 4.45.

Table 4.44 Estimates of parameters for the HWMS model for weekly price of garbled black pepper

Parameter	Α	Β	γ
Estimate	0.999	0.0114	0.00

With these values for the parameters, HWMS model for weekly price of black pepper price are as given below,

Level: L_t= 0.999* ($\frac{Y_t}{S_{t-12}}$)+.001*(L_{t-1}– 0.8426)

Trend: b_t = 0.0114($L_t - L_{t-1}$) – 1.8426

Seasonality: S_t= S_{t-12}

Forecast: $F_{t+m} = (L_t + b_t m)S_{t-12+m}$

The values of L_t, b_t and S_t were substituted in the forecast equation F_{t+m} to obtain the price.

Table 4.45 Model accuracy measures by HWMS model for weekly price of garbled black pepper

Accuracy measure	Value
RMSE	10.03
MAPE	3.28

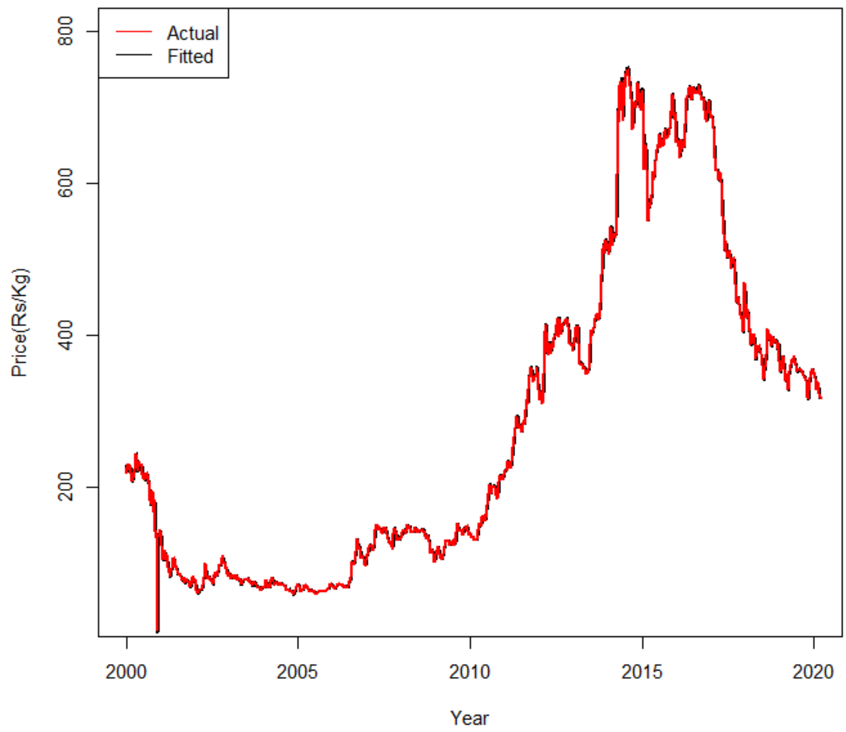


Figure 4.45 Actual and fitted plot for HWMS model of weekly price of garbled black pepper

The residual plot for black pepper price from HWMS model is given in Figure 4.46 along with the ACF and PACF residual plots (Figure 4.47). It showed that most of the ACF and PACF values lie within the confidence limits.

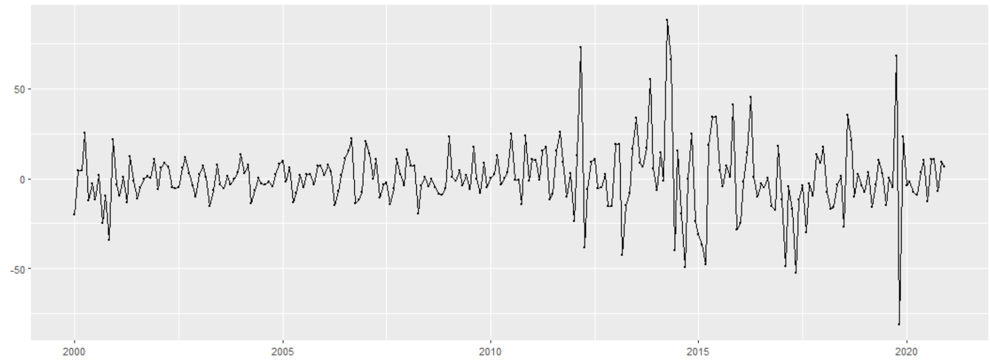


Figure 4.46 Residual plot for weekly price of garbled black pepper price for HWMS model

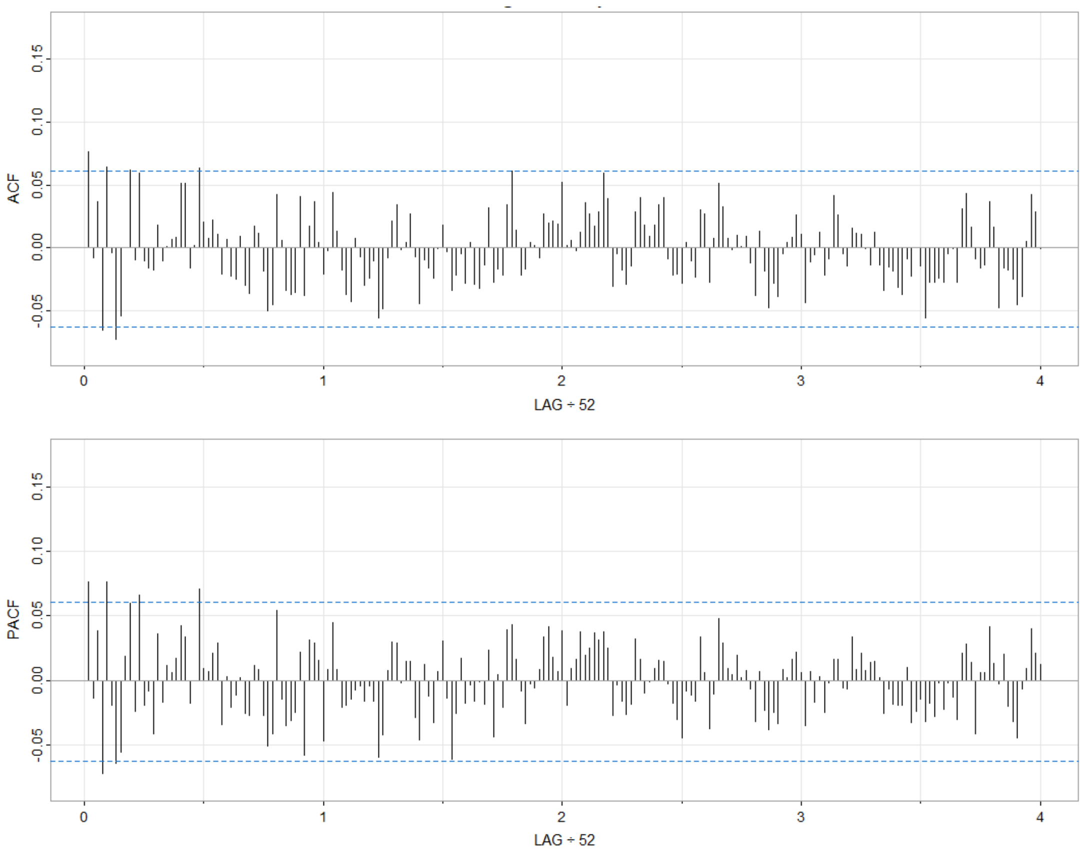


Figure 4.47 Residual ACF and PACF plots for weekly price of garbled black pepper for HWMS model

4.3.4.2 ARIMA model for weekly price of garbled black pepper

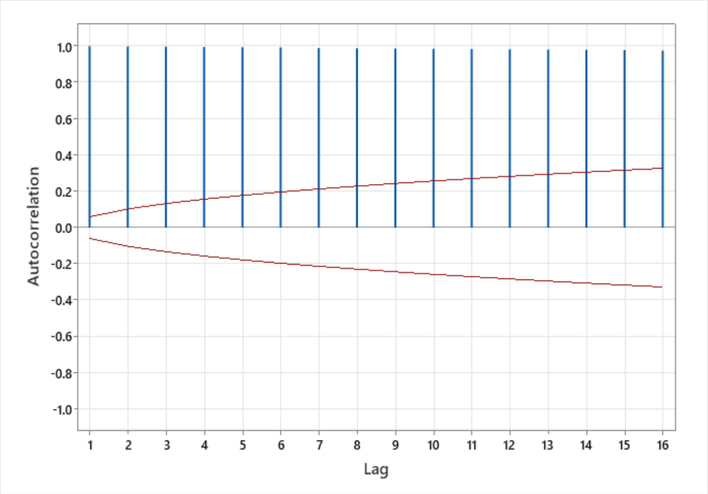
The time plot of weekly price for garbled black pepper indicates that it is non-stationary. Autocorrelations and partial autocorrelation were computed for weekly price of garbled black pepper which are provided in Table 4.46. Significance of autocorrelations upto 16 lags confirm the non stationarity of the data and partial autocorrelation value has very high value for lag 1.

Table 4.46 ACF and PACF values for weekly price of garbled black pepper

			Ljung-Box Statistic				
Lag	**Auto-correlation**	**SE**	**Value**	**Df**	**Probability (p)**	**Partial Auto-correlation**	**SE**
1	0.998	0.029	1195.27	1	<0.001	0.998	0.029
2	0.997	0.029	2387.28	2	<0.001	-0.074	0.029
3	0.995	0.029	3575.99	3	<0.001	0.001	0.029
4	0.993	0.029	4761.06	4	<0.001	-0.047	0.029
5	0.991	0.029	5942.69	5	<0.001	0.032	0.029
6	0.989	0.029	7120.61	6	<0.001	-0.043	0.029
7	0.987	0.029	8294.84	7	<0.001	0.011	0.029
8	0.985	0.029	9465.64	8	<0.001	0.029	0.029
9	0.983	0.029	10633.18	9	<0.001	0.023	0.029
10	0.982	0.029	11797.43	10	<0.001	-0.012	0.029
11	0.980	0.029	12958.11	11	<0.001	-0.040	0.029
12	0.978	0.029	14115.20	12	<0.001	0.002	0.029
13	0.976	0.029	15268.36	13	<0.001	-0.049	0.029
14	0.974	0.029	16417.59	14	<0.001	0.006	0.029
15	0.972	0.029	17562.95	15	<0.001	0.008	0.029
16	0.970	0.029	18704.40	16	<0.001	0.001	0.029

p<0.01 indicates significance of autocorrelation

The ACF and PACF plots for weekly price of garbled black pepper is depicted in Figure 4.48. It is evident that in the ACF plot, spikes upto 16 lags fall above confidence limit and PACF plot showed spikes for a number of lags (1, 2 and 6) beyond confidence limits also indicated the non-stationarity of the time series.



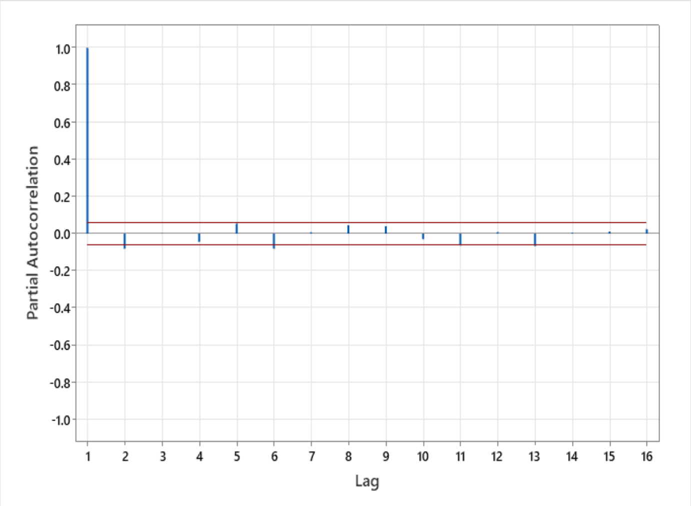


Figure 4.48 ACF and PACF plots for weekly price of garbled black pepper

The stationarity of the data was evaluated using ADF test. Following the initial differencing of the data, the ADF test was repeated. The significance of the ADF test statistic, as indicated in Table 4.47, affirmed the necessity of taking the first difference to achieve stationarity in the data.

Table 4.47 ADF test with critical values for weekly price of garbled black pepper

Garbled black pepper price	ADF test statistic	Probability(p)	Critical values
Actual	-1.0651	0.9282	1.46
First difference	-21.844**	0.01	-2.32

** indicates significant at 5%level(p<0.05)

Among several ARIMA models tried, the tentative models were chosen based on the value of MAPE, AIC and RMSE. SARIMA (2,1,2) $(1,0,0)_{52}$ was chosen as the best forecast model for weekly price of garbled black pepper.

Table 4.48 Tentatively identified SARIMA(p,d,q)$(P,D,Q)_{52}$ models for weekly price of garbled black pepper

Tentative models	MAPE	AIC	RMSE
SARIMA(2,1,2)$(0,0,1)_{52}$	3.23	7830.95	9.97
SARIMA(2,1,2)$(1,0,0)_{52}$	**3.23**	**7828**	**9.96**

The parameters of the model SARIMA(2,1,2)$(1,0,0)_{52}$ along with their tests of significance are provided in Table 4.49.

Table 4.49 SARIMA(2,1,2)$(1,0,0)_{52}$ model parameters for weekly price of garbled black pepper

Model Parameters	Estimate	SE	Z value	Probability
Non-seasonal difference	1			
AR Lag 1 (ϕ_1)	0.328***	0.109	2.993	0.0027
AR Lag 2 (ϕ_2)	0.639***	0.102	6.225	0.00
MA Lag 1(θ_1)	-0.238**	0.100	-2.368	0.0178
MA Lag 2(θ_2)	-0.704***	0.091	-7.730	0.00
Seasonal difference	0			
AR Seasonal Lag 1(Φ_1)	-0.025**	0.038	-0.6704	0.0380

**indicates significance at 5% percentage level($p<0.05$)

***indicates significance at 1% percentage level ($p<0.01$)

The general form of the SARIMA(2,1,2)$(1,0,0)_{52}$ model equation is as follows:

$(1-\phi_1 B-\phi_2 B^2)(1-B)(1-\Phi_1 B^{52})\ y_t=(1-\theta_1 B-\theta_2 B^2)\ \epsilon_t$,$y_t=\log_e Y_t$

Where, $\phi_1 = 0.328$, $\phi_2 = 0.639$, $\Phi_1= -0.025$, $\theta_1=-0.238$, $\theta_2=-0704$

The SARIMA(2,1,2)(1,0,0)$_{52}$ model equation is as follows:

$$(1-0.328B-0.639B^2)(1-B)(1+0.025B^{52})y_t=(1+0.238B+0.704B^2)\epsilon_t$$

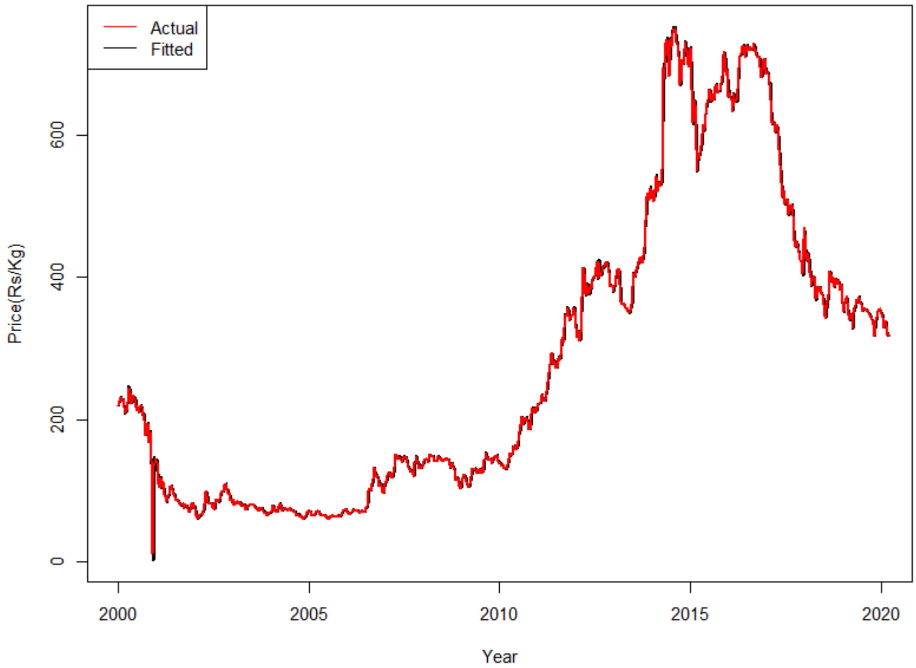


Figure 4.49 Actual and fitted plot for SARIMA(2,1,2) (1,0,0)$_{52}$ for weekly price of garbled black pepper

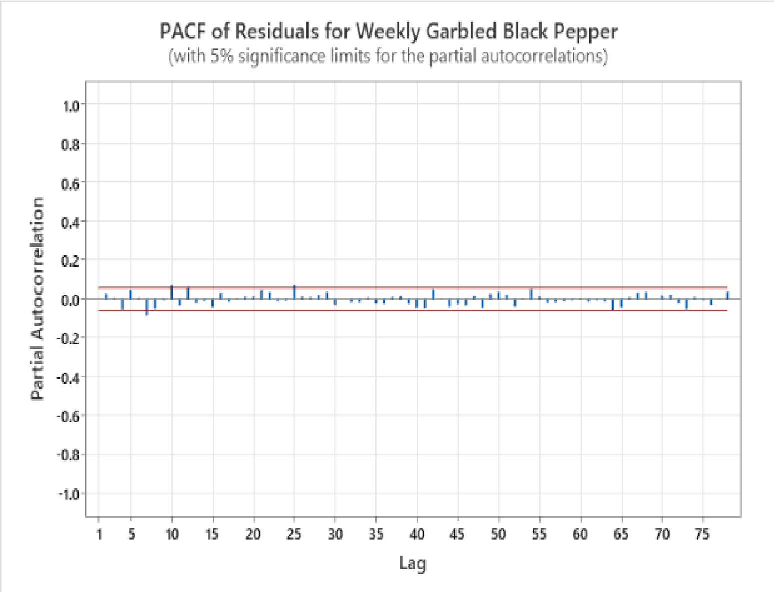
The plot of actual and fit values for weekly price of garbled black pepper using SARIMA(2,1,2)(1,0,0)$_{52}$ model is given in Figure 4.49.

The adequacy of the model was also tested using the value of Box-Pierce Q statistics and was found to be insignificant. So, overall we can say SARIMA(2,1,2)(1,0,0)$_{52}$ model show satisfactory result, among different ARIMA models.

Table 4.50 Ljung-Box 'Q' statistic for residuals of SARIMA(2,1,2)(1,0,0)$_{52}$ model

Statistic	p-value
93.023	0.6502^{NS}

NS: Non-significant



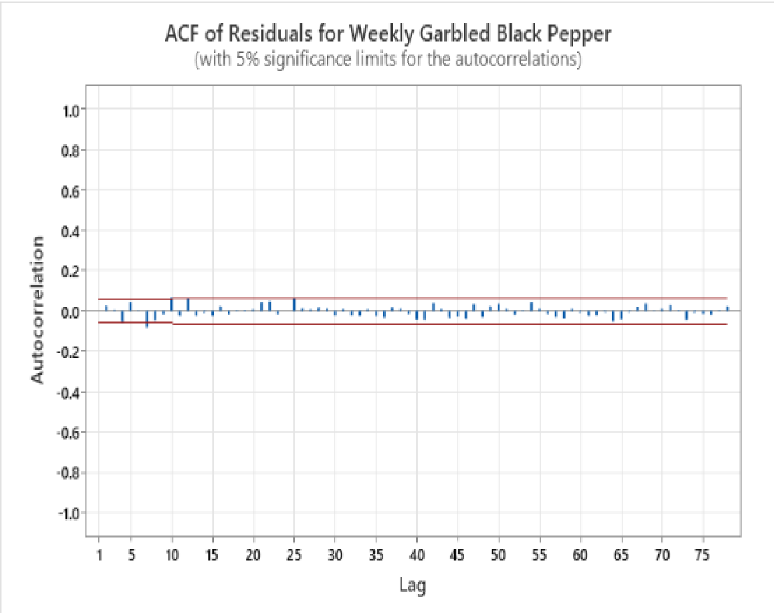


Figure 4.50 Residual ACF and PACF plots for weekly price of garbled black pepper for SARIMA(2,1,2) $(1,0,0)_{52}$

Residual ACF and PACF plots for the SARIMA(2,1,2)$(1,0,0)_{52}$ model fitted for weekly price of garbled black pepper is provided in Figure 4.50. It could be seen that majority of the spikes in the ACF and PACF plots fall within the critical values, thus indicating the adequacy of SARIMA(2,1,2)$(1,0,0)_{52}$ model for forecasting weekly price of garbled black pepper.

4.3.4.2 Volatility in weekly price of garbled black pepper

The weekly price of garbled black pepper for the period 2000-2020 was used to choose the ARCH family models using R software.

The Heteroscedasticity LM test was carried out to check the volatility or ARCH effect in the time-series. The result of the test is presented in Table 4.51 which reveals that, there is an ARCH effect in the time series,

Table 4.51 Heteroscedasticity LM Test for first differenced

Statistic	p-value
484.36***	0.000

*** Significant at 1 per cent level of significance

Several ARCH models were tested, but their coefficients were found to be insignificant, highlighting the need for the implementation of the GARCH model.

4.3.4.2.1 GARCH(1,1) model for weekly price of garbled black pepper

The estimates of the GARCH (1,1) model fitted for the weekly price of garbled pepper is given in the Table 4.52. As observed, that the constant term, ARCH and GARCH parameters are positive and significant indicating the volatility.

Table 4.52 Estimates of GARCH(1,1) model for weekly price of garbled black pepper

Parameter	Coefficient	Std. error	t value	p-value
Constant term(α_0)	0.0021***	0.004	4.273	0.00
ARCH term(α_1)	0.9266***	0.051	11.146	0.00
GARCH term(β_1)	0.0302***	0.082	30.751	0.00

***Significant at 1% level of significance

The GARCH(1,1) model is given by,

$$h_t = 0.0021+0.9266\varepsilon_{t-i}{}^2+0.0302\sigma_{t-j}^2$$

The time varying volatility includes a constant (0.0021), a component which depends on past errors ($0.9266\varepsilon_{t-i}{}^2$) and a component which depends on weighted average of past squared residuals ($0.0302\sigma_{t-j}^2$) . In the Table 4.53, the p-value for the t-statistic of the first order coefficient (11.14) and the second order coefficient (30.75) suggests a significant GARCH (1,1) coefficient

Table 4.53 Model accuracy measures by GARCH(1,1) model for weekly price of garbled black pepper

Accuracy measure	Value
RMSE	10.00
MAPE	3.25

Residual analysis was carried out to check the adequacy of the selected model. The Serial Correlation LM test for residuals is presented in Table 4.54. The large value of p ($p=0.999>0.05$) reveals that there is no serial correlation in the residuals. The Ljung-Box test of residuals is presented in Table 4.55. The large value of p ($p=0.963 > 0.05$) with respect to Ljung-Box 'Q' statistic indicates that the residuals are normally distributed.

Table 4.54 Serial Correlation LM test for residuals of GARCH(1,1) Model

Statistic	p-value
5.11	**0.999^{NS}**

NS: Non-significant

Table 4.55 Ljung-Box test for residuals of GARCH(1,1) Model

Statistic	p-value
17.3	**0.963**^{NS}

NS: Non-significant

4.3.4.4 ANN model for weekly price of garbled black pepper

Time delayed neural network (TDNN) model was fitted for weekly price of garbled black pepper .The best time lagged neural network with single hidden layer was found for each series by conducting experiments with the basic cross validation method. Out of a total of 112 neural network structures, a neural network model with thirteen lagged observations as input nodes and eight hidden layers (13:8s:1l) performed better than other competing models in respect of forecasting accuracy measures. This means that most accurate price forecast for the given series is obtained when the price of thirteen preceding weeks is used as inputs.

The selected TDNN model is described in Table 4.56 along with the forecasting accuracy measures for both training and testing set.

Table 4.56 Model accuracy measures by TDNN model for weekly price of garbled black pepper

Model	No. of parameters	MAPE		RMSE	
		Train	Test	Train	Test
13:8s:1l	121	3.95	1.99	8.49	6.78

The actual weekly price along with the predicted values of weekly price of garbled black pepper using TDNN model is provided in Figure 4.51.

Figure 4.51 Actual and fitted plots for TDNN model for weekly price of garbled black pepper

The adequacy of the model was also evaluated using the value of Ljung-Box ‘Q’ statistic and was found to be insignificant which is shown in Table 4.57. So, overall we can say TDNN model shows satisfactory result.

Table 4.57 Ljung-Box ‘Q’ statistic for residuals of TDNN model

Statistic	p-value
159.57	**0.3783^{NS}**

NS: Non-significant

The residual plot from TDNN model is provided in Figure 4.52 which did not exhibit any specific pattern. The residual ACF and PACF plot is provided in Figure 4.53, majority of the spikes in the residual ACF and PACF are within the critical values indicating the adequacy of the model.

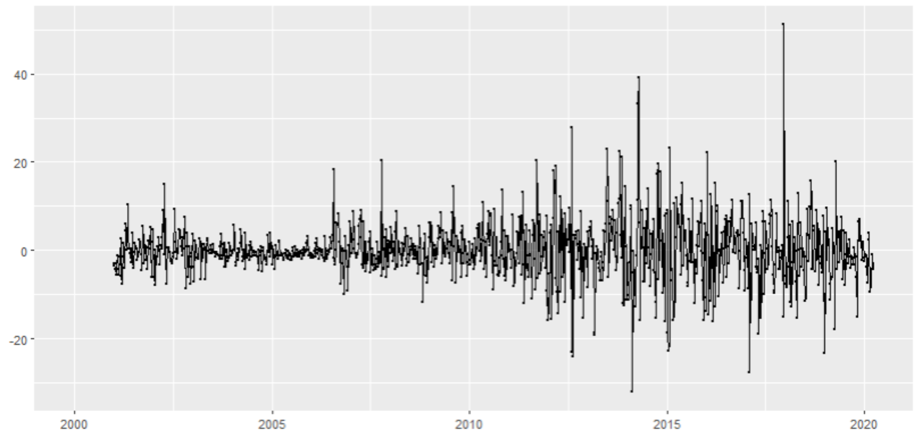


Figure 4.52 Residual plot for weekly price of garbled black pepper for TDNN model

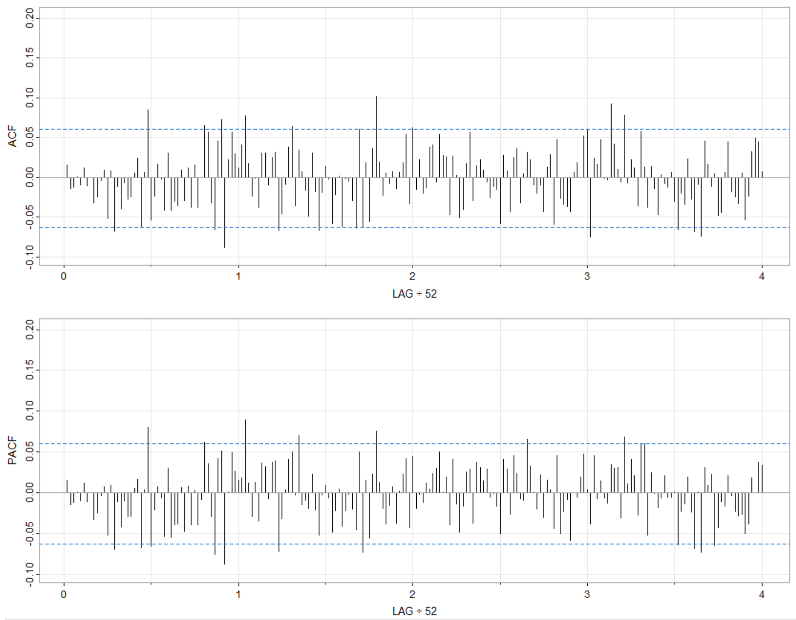


Figure 4.53 Residual ACF and PACF plot for weekly price of garbled black pepper for TDNN model

4.3.4.5 LSTM model for weekly price of garbled black pepper

LSTM model was fitted for weekly price of garbled black pepper. The parameters used for the model was provided in Table 3.1. The LSTM model has been created with three layers by performing epoch of 50 times. During the training process, the model makes predictions on the training data, and the loss function is used to calculate the error between these predictions and the actual target values. The training loss is the average of these errors across all training samples. The goal of training is to adjust the weights and biases of the model to minimize this training loss. After each epoch, the weights of the model get updated and the new epoch works on those updated values that process continues in every epoch. In this study, MAE has been used as a measure to continue the epoch until MAE reaches a minimum which is shown in Figure 4.54.

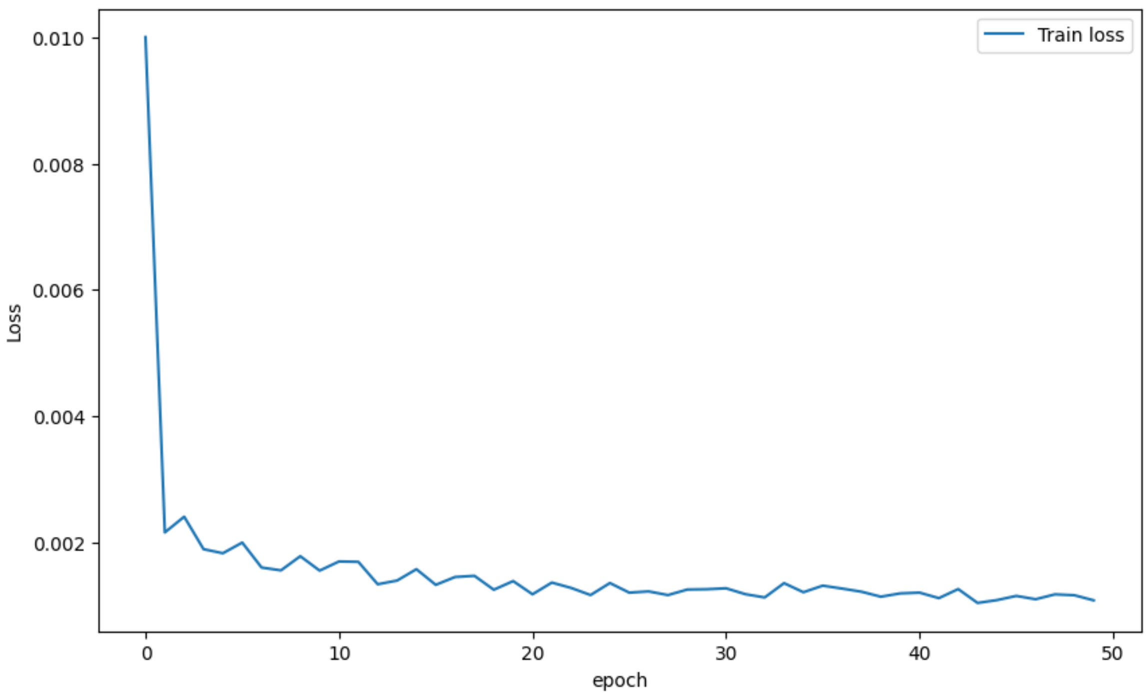


Figure 4.54 Training loss for LSTM model for weekly price of garbled black pepper

Various accuracy measures for LSTM model are provided in Table 4.58.

Table 4.58 Model accuracy measures by LSTM model for weekly price of garbled black pepper

Accuracy measure	Value
RMSE	16.36
MAPE	5.57

The actual price along with the predicted values for weekly price of garbled black pepper using LSTM model is provided in Figure 4.55.

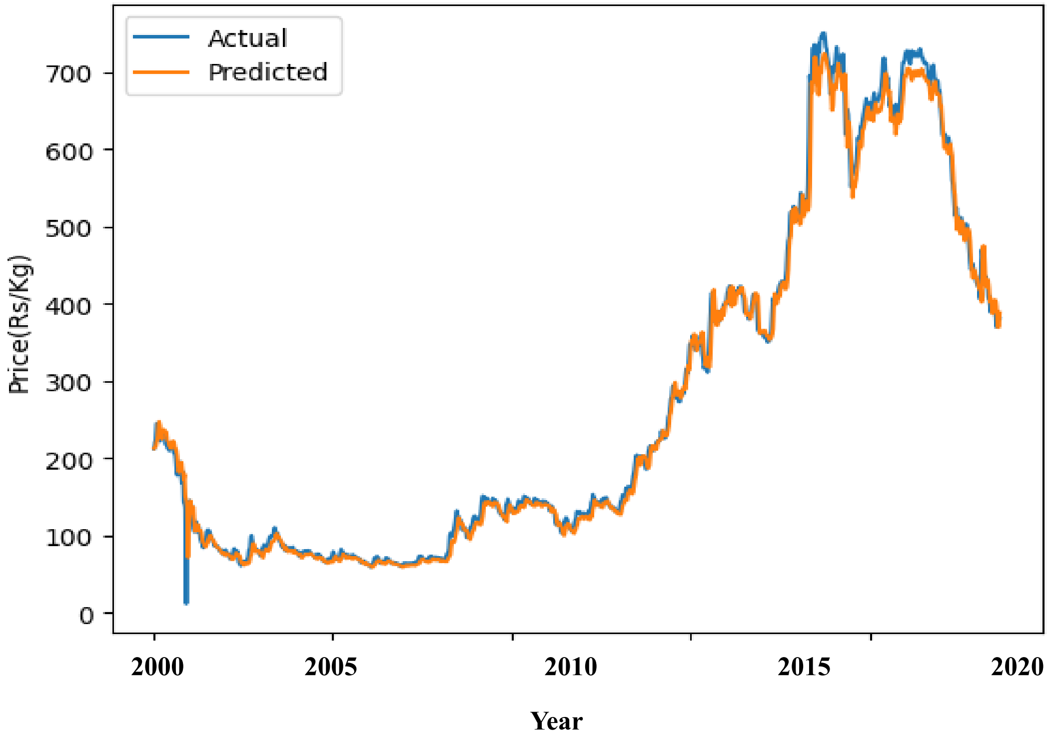


Figure 4.55 Actual and fitted plot for LSTM model for weekly price of garbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant which is shown in Table 4.59. So overall, we can say LSTM model shows a satisfactory result.

Table 4.59 Ljung-Box 'Q' statistic for residuals of LSTM model

Statistic	p-value
624.52	0.610^{NS}

NS: Non-significant

The residual plot from LSTM model is provided in Figure 4.56 and the residuals did not exhibit any specific pattern, they are scattered.

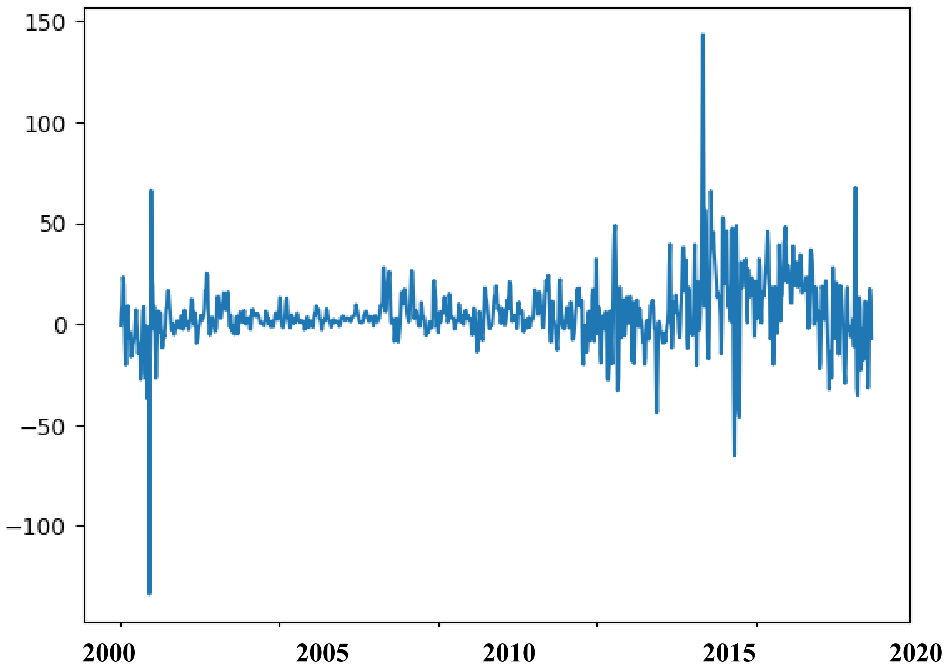


Figure 4.56 Residual plot for weekly price of garbled black pepper for LSTM model

4.3.5. Comparison of models

In present study, a comparison of different models has been done in order to know the best model for forecasting price of weekly price of garbled black pepper in Kochi market. The comparison of all the 6 models were carried out based on the test MAPE and RMSE values which were considered to be least. According to the Table 4.60, the TDNN model with least MAPE and RMSE values was considered to be best among all the models considered.

Table 4.60 Comparison of time series forecasting models for weekly price of garbled black pepper

Model	MAPE	RMSE
Exponential trend model	40.3	106.98
SARIMA(2,1,2)(1,0,0)$_{52}$	3.23	9.96
GARCH(1,1)	3.25	10.00
HWMS	3.28	10.03
TDNN(13:8s:1l)	**1.99**	**6.74**
LSTM	5.57	16.36

The plot of actual and fit values along with the forecasts for the weekly price of garbled black pepper using TDNN model is given in Figure 4.57. The figure showed that the actual and predicted prices are in agreement.

The MAPE value are found to be 3.11 and 3.16 respectively based on the actual and forecasted weekly price of garbled black pepper using the TDNN model for the year 2021 and 2022.

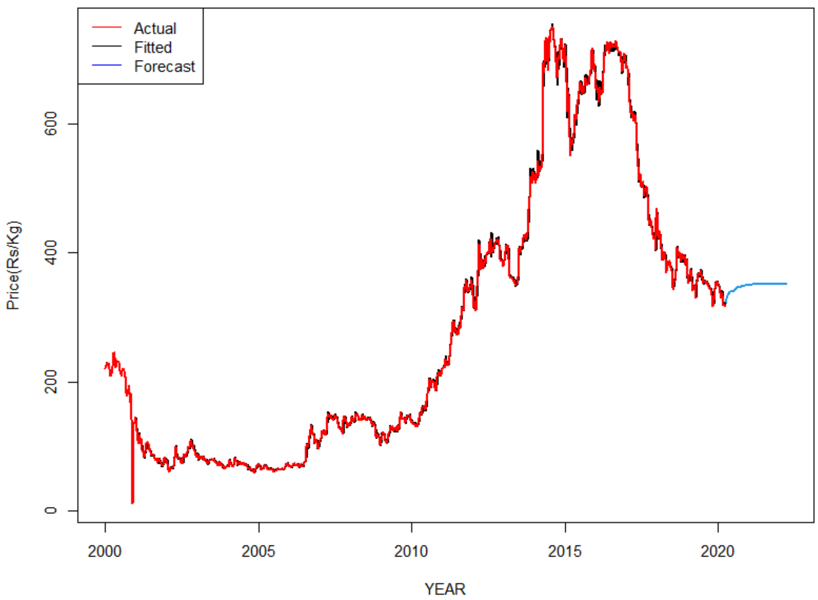


Figure 4.57 Actual, predicted and forecasted plot for TDNN model of weekly price of garbled black pepper

The TDNN model captures the pattern of the weekly price of garbled black pepper. The MAPE value was 1.99. The residual plot from TDNN model is plotted in Table 4.52 and was found to be scattered. Further, residual ACF and PACF plots for the TDNN model in Figure 4.53, exhibited that the majority of the spikes fall within the critical values which shows the adequacy of TDNN model.

The LSTM model, being a type of recurrent neural network, would have been a better model if data points increase. It stands out in handling sequences by effectively remembering and utilizing information over longer data points through its unique gating mechanism. This makes LSTMs well-suited for tasks like predicting time series, where capturing complex patterns and dependencies in the data is crucial. As the dataset grows, LSTMs prove advantageous over simpler models in retaining context and making more accurate predictions by considering the relationships between past inputs. Thus, for weekly price of garbled black pepper price at Kochi market, TDNN model was selected as the best forecast model.

4.4. Analysis of weekly price of ungarbled black pepper in the Kochi market

The results obtained from the analysis of weekly ungarbled black pepper price data for Kochi market is presented below:

4.4.1 Pattern for weekly price data of ungarbled black pepper

The time plot for weekly average prices of ungarbled black pepper in the Kochi market from 2000 to 2020 were used to study the price pattern and it exhibited some notable patterns which is depicted in Figure 4.58.

It's essential to note that while these observations provide a broad overview of the price trends, the specific factors influencing pepper prices are multifaceted and can be influenced by various economic, agricultural, and geopolitical factors.

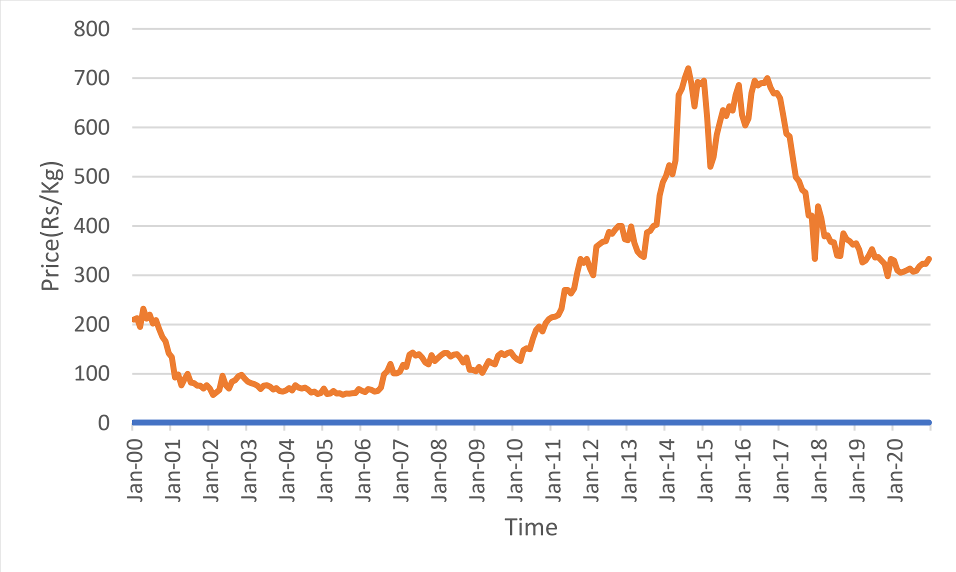


Figure 4.58 Price pattern for weekly price of ungarbled black pepper

4.4.2. Trend Analysis

Trend analysis was carried out for weekly average price of ungarbled black pepper. Different functional forms like linear, quadratic, exponential, etc. were tried and the suitable model in each case was chosen based on the MAPE and RMSE values.

The trend equation fitted for weekly price of ungarbled black pepper along and their accuracy measures are provided in Table 4.61. Graphical plots revealing the pattern of the series based on the different trend models tried are plotted in Figure 4.59 – 4.61.

Table 4.61 Trend equations for weekly price of ungarbled black pepper

Functional Form	Trend equation	MAPE	RMSE
Linear Trend Model	$Y_t = 14.01 + 0.4704 \times t$	53.1	122.80
Exponential Trend Model	$\boldsymbol{Y_t = 64.6177 \times (1.00207\text{^}t)}$	**40.4**	**107.11**
Quadratic Trend Model	$Y_t = 0.1 + 0.5466 \times t - 0.000070 \times t^2$	54.6	123.84

As observed from the table, lower MAPE value of 40.4 and RMSE value of 107.11, shows the adequacy of the exponential trend model in explaining the trend of the black pepper price.

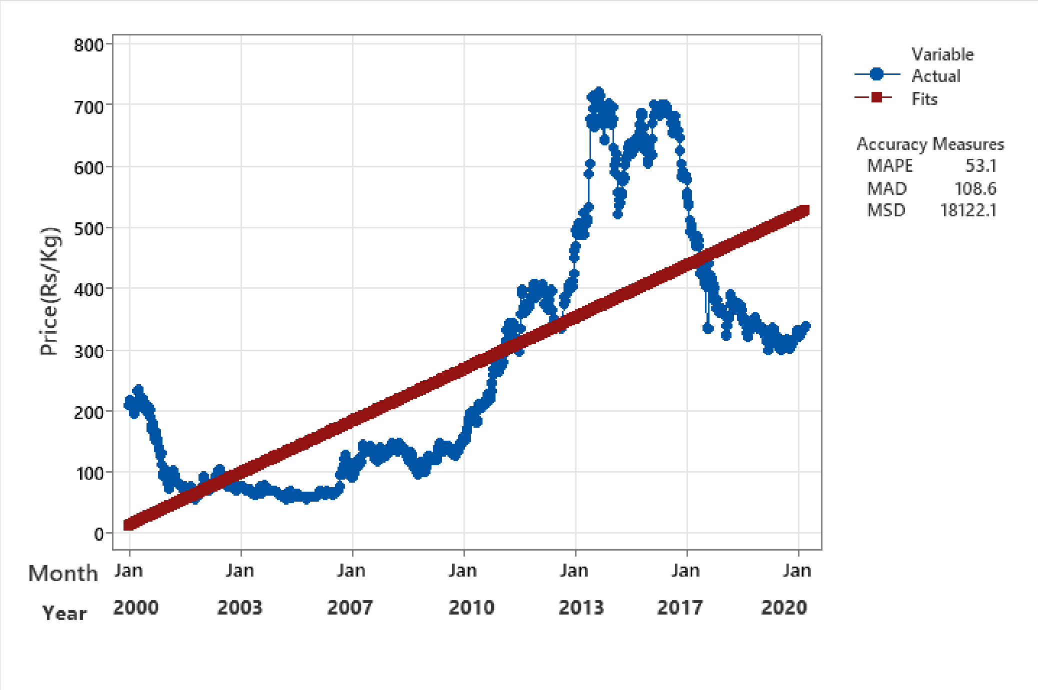


Figure 4.59 Linear trend plot for weekly price of ungarbled black pepper

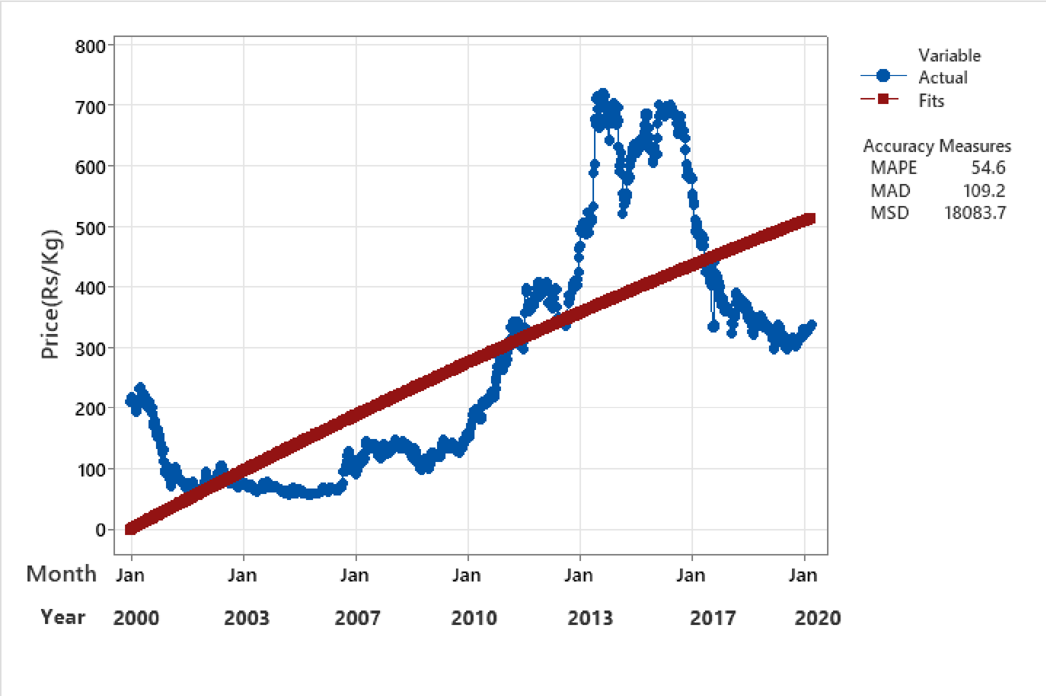


Figure 4.60 Exponential trend plot for weekly price of ungarbled black pepper

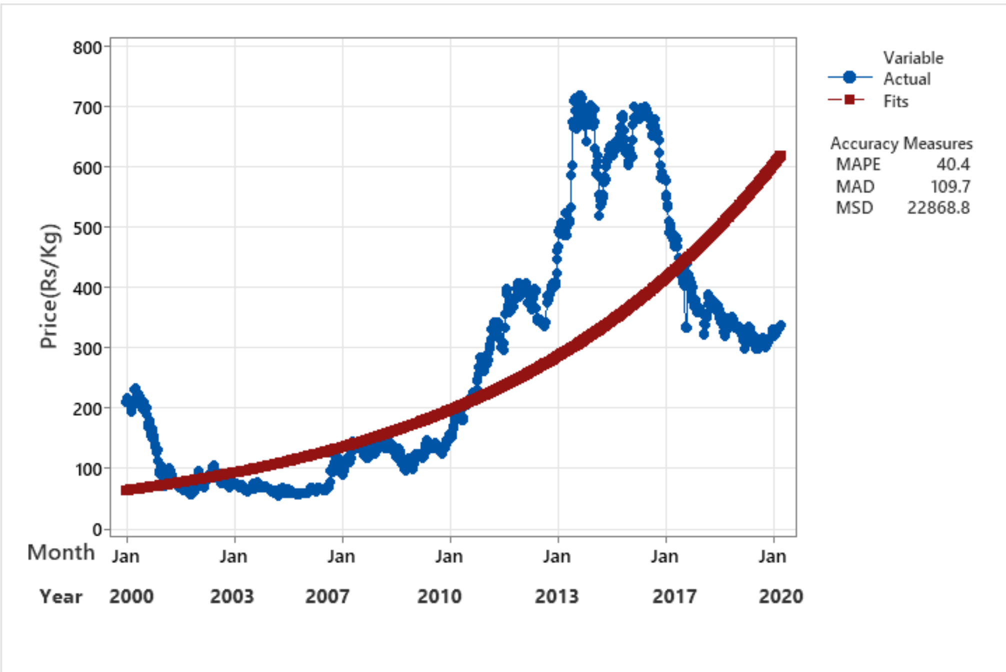


Figure 4.61 Quadratic trend model for weekly price of ungarbled black pepper

4.4.3 Decomposition for weekly price of ungarbled black pepper

The weekly price of ungarbled black pepper is decomposed and is depicted in Figure 4.62. In the first panel, the observed weekly price of ungarbled black pepper has been plotted. The second panel illustrates the trend in the prices, while the third and fourth panels depict the seasonal and random variations, respectively. Notably, the weekly price pattern of garbled black pepper dataset reflects a dynamic economic landscape with periods of growth, decline, and high volatility.

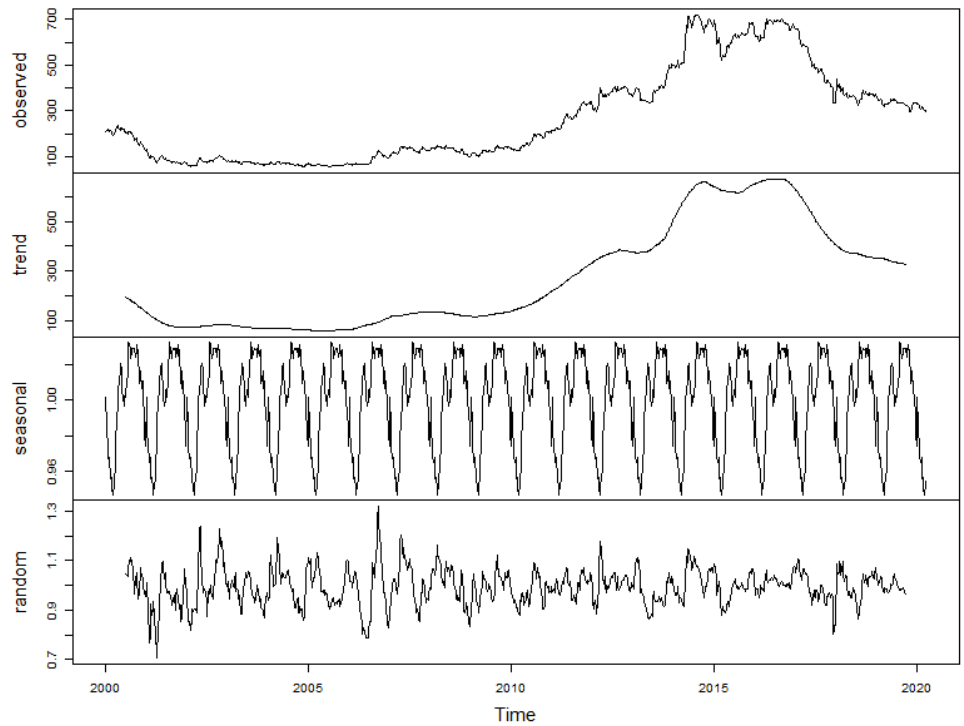


Figure 4.62 Decomposition for weekly price of ungarbled black pepper

4.4.3.1 Seasonal indices for weekly price of ungarbled black pepper

Decomposition of the weekly price of ungarbled black pepper during January 2000-December 2020 indicated the same seasonality pattern in every year for the Kochi market. Seasonal indices were calculated for 52 weeks (January to December) from the weekly price data of ungarbled black from January, 2000 to December, 2020 to understand the seasonal behaviour of prices.

Seasonal indices are calculated by ratio to moving average for weekly price of ungarbled black pepper and corresponding seasonal plot is shown in Figure 4.63.

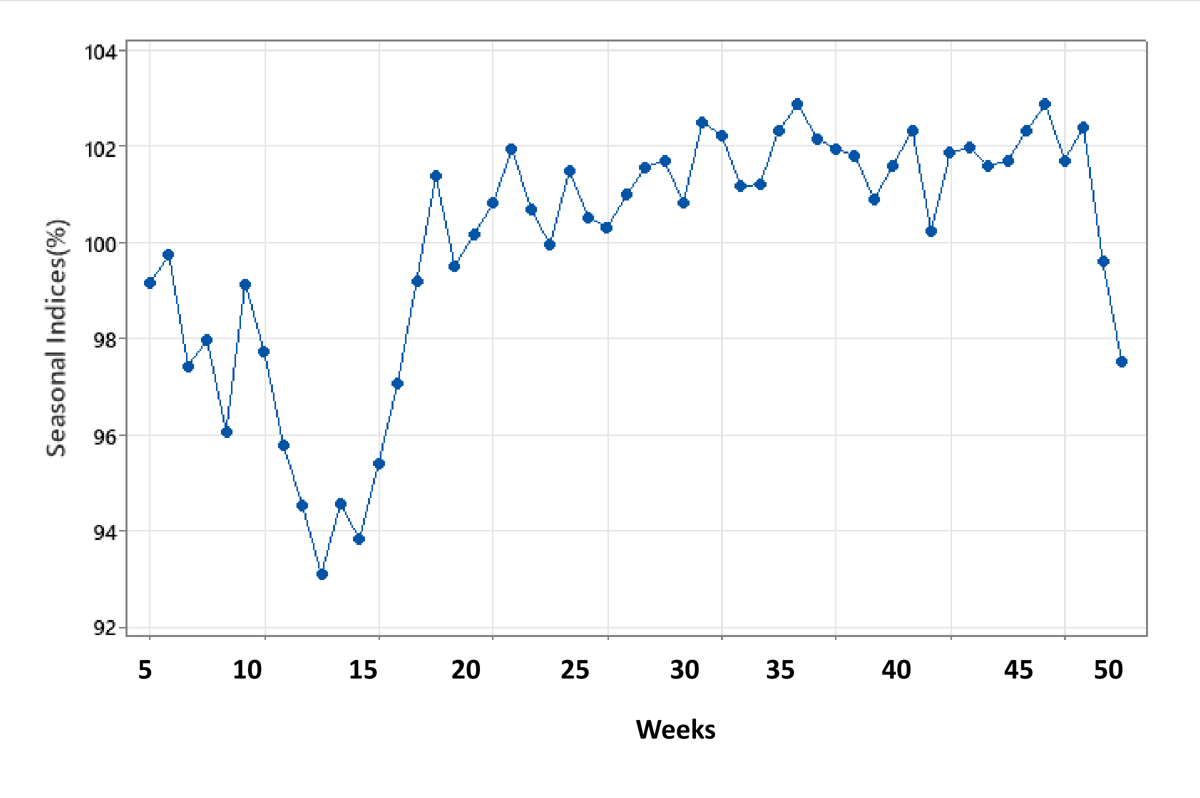


Figure 4.63 Seasonal plot for weekly price of ungarbled black pepper

The weekly seasonal indices for price of ungarbled black pepper reveal a clear pattern of seasonality over the course of a year. These indices indicate the percentage variation in prices compared to an average. The data exhibits a recurring pattern with fluctuations, ranging from 93.11 per cent to 102.89 per cent. First week of April to first week of August show a relatively higher range, suggesting potential periods of increased demand or other influential factors affecting black pepper prices. The lowest price index is recorded in first week of February at 93.11 per cent, while the highest occurs in first week of 102.89 per cent. This information is crucial for stakeholders in the black pepper market, providing insights into the cyclical nature of pricing trends, which can aid in strategic planning, decision-making, and risk management within the market.

4.4.3.2 Cyclical variation for weekly price of ungarbled black pepper

The cyclical pattern for weekly price of ungarbled black pepper is depicted in Figure 4.58. The first 11-year cycle was from 1983 to 1993 and the second cycle from 1993, showed some fluctuation near the peak values and reached the lowest value in 2004 (Sabu, 2015). The third cycle started from 2005 when the prices started gradually increasing until reaching a peak in 2010. This upward trend is part of the cyclical variation, reflecting a period of growth and possibly influenced by factors such as increased demand, favourable weather conditions, or changes in market dynamics.

Subsequently, the prices experience a sharp spike in 2011 indicating a peak in the cycle. This could be attributed to specific factors like a surge in demand, reduced supply, or other market forces. Following the peak, there is a noticeable downward trend in prices. This decline represents the trough of the cycle, possibly influenced by factors such as increased production, changes in consumer preferences, or economic conditions.

The cyclical variation continues as prices start to rise again in 2020. This upward movement suggests a new phase of the cycle, possibly influenced by factors such as improved market conditions, changes in agricultural practices, or global economic trends.

4.4.3.3 Irregular variation for weekly price of ungarbled black pepper

The random effect is the residual effect after the trend, seasonal and cyclical effects have been removed from the original observations. As observed from the fourth panel of Figure 4.62, weekly price of ungarbled black pepper exhibited significant irregular variations during 2000 to 2020. They represent random effect such as demand and supply shocks on account of climatic aberrations or due to speculative factors.

4.4.4. Forecast for weekly price of ungarbled black pepper

A series of models comprising exponential smoothing model, ARIMA/SARIMA, ARCH/GARCH and ANN were fitted to forecast weekly price of ungarbled black pepper. The best model was selected based on the forecast accuracy measure: MAPE and RMSE. The results are provided in following subsections:

4.4.4.3 Exponential smoothing model

Holt-Winters' Multiplicative Seasonal (HWMS) model was identified best among the different exponential smoothing models like SES, DES, HWAS and HWMS, for the weekly price of ungarbled black pepper based on criteria like agreement between observed and fitted price plots, MAPE and RMSE. The fit of the HWMS model for weekly price of ungarbled black pepper is provided in Figure 4.64 and the that the actual and model fit values are in close agreement. The estimates of parameters of HWMS model are provided in Table 4.62. Various accuracy measures of HWMS model is given in Table 4.63.

Table 4.62 Estimates of parameters of the HWMS model for weekly price of ungarbled black pepper

Parameter	α	B	γ
Estimate	0.9999	0.0137	0.00

With these values for the parameters, HWMS model for weekly price of ungarbled black pepper are as given below,

Level: L_t= 0.999* $(\frac{Y_t}{S_{t-12}})$+.0884*(L_{t-1}– 0.4835)

Trend: b_t = 0.0137($L_t - L_{t-1}$) - 1.4835

Seasonality: S_t= S_{t-12}

Forecast: $F_{t+m} = (L_t + b_t m)S_{t-12+m}$

The values of L_t, b_t and S_t were substituted in the forecast equation $F_{t+m}$to obtain the price.

Table 4.63 Model accuracy measures by HWMS model for weekly price of ungarbled black pepper

Accuracy measure	Value
RMSE	9.30
MAPE	**2.41**

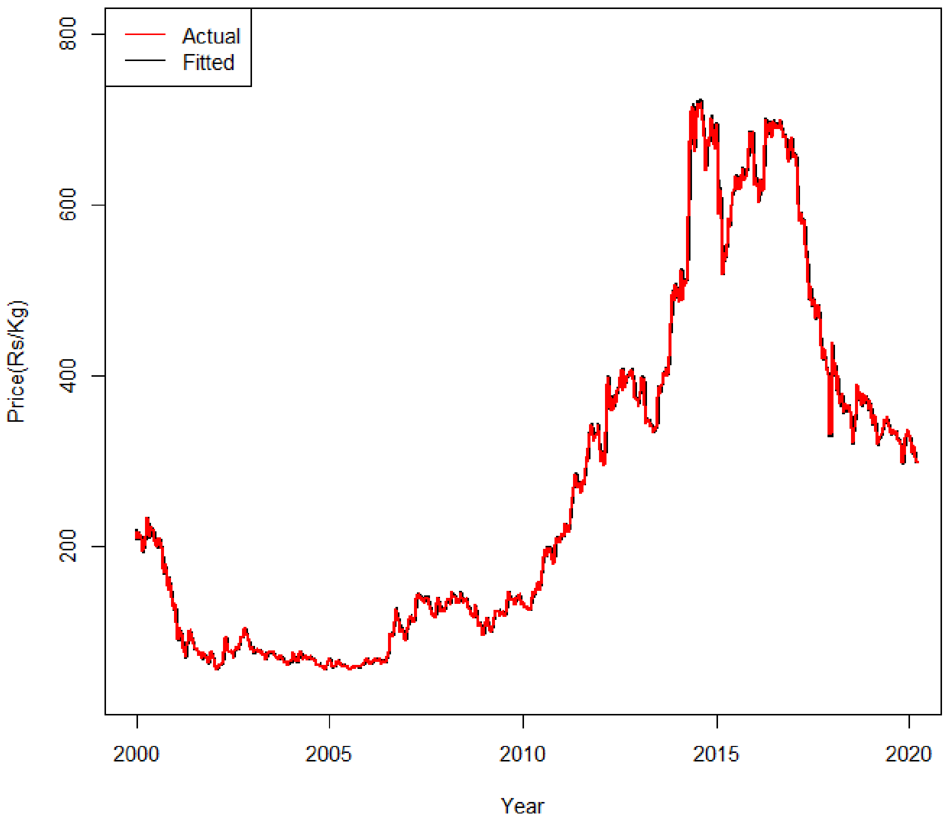


Figure 4.64 Actual and fitted plot for HWMS model for weekly price of ungarbled black pepper

The residual plot for weekly price of ungarbled black pepper from HWMS model is given in Figure 4.65 along with the ACF residual plot (Figure 4.66). It showed that the most of the ACF values lie outside the confidence limits indicating the adequacy of the model.

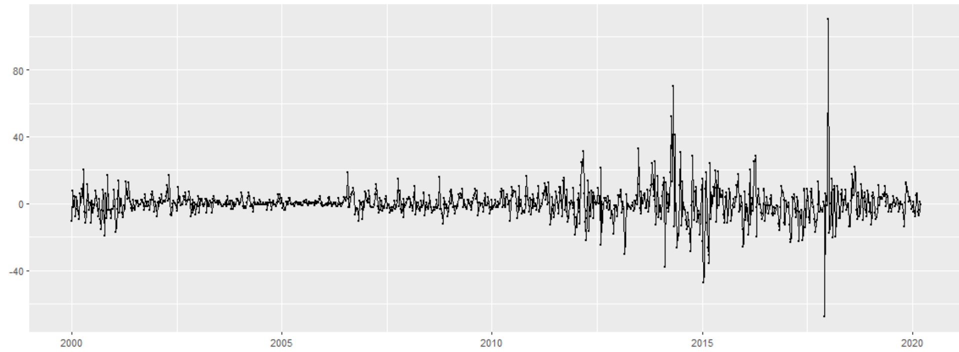


Figure 4.65 Residual plot for weekly price of ungarbled black pepper for HWMS model

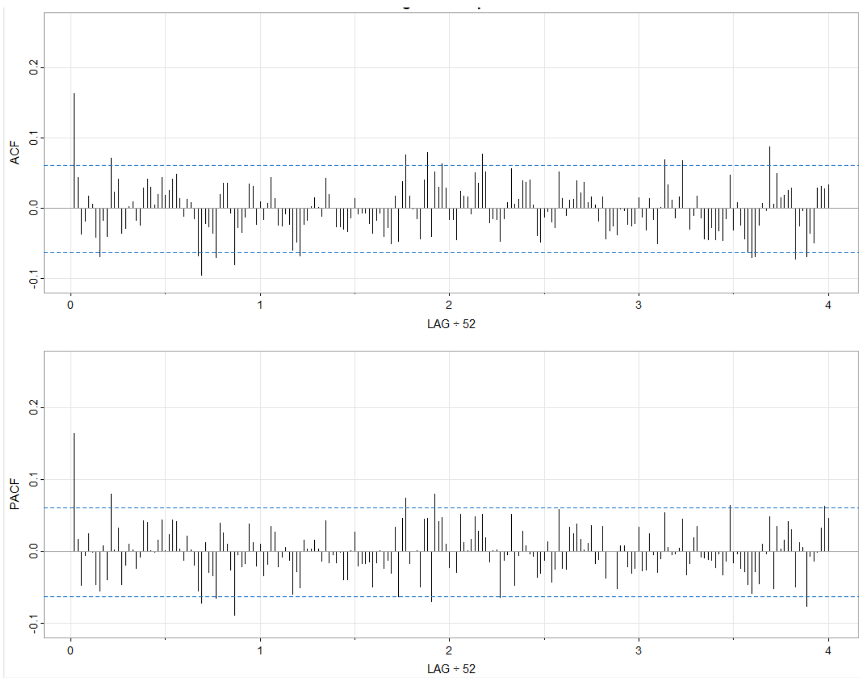


Figure 4.66 Residual ACF and PACF plots for weekly price of ungarbled black pepper for HWMS model

4.4.4.2 ARIMA model for weekly price of ungarbled black pepper

The time plot for weekly price for garbled black pepper indicates that it is non-stationary. Autocorrelations and partial autocorrelation were computed for weekly price of garbled black pepper are provided in Table 4.64. Significance of autocorrelations upto 16 lags confirm the non stationarity of the data and partial autocorrelation value has very high value for lag 1.

Table 4.64 ACF and PACF values for weekly price of ungarbled black pepper

			Ljung-Box Statistic				
Lag	**Auto-correlation**	**SE**	**Value**	**Df**	**Probability (p)**	**Partial Auto-correlation**	**SE**
1	0.998	0.029	252.83	1	<0.001	0.998	0.029
2	0.996	0.029	503.37	2	<0.001	-0.117	0.029
3	0.994	0.029	751.53	3	<0.001	-0.018	0.029
4	0.992	0.029	996.83	4	<0.001	0.001	0.029
5	0.990	0.029	1239.02	5	<0.001	-0.010	0.029
6	0.988	0.029	1477.49	6	<0.001	0.002	0.029
7	0.986	0.029	1711.45	7	<0.001	-0.018	0.029
8	0.983	0.029	1940.72	8	<0.001	0.003	0.029
9	0.981	0.029	2165.48	9	<0.001	0.023	0.029
10	0.979	0.029	2385.75	10	<0.001	-0.007	0.029
11	0.977	0.029	2601.38	11	<0.001	-0.017	0.029
12	0.975	0.029	2812.29	12	<0.001	-0.054	0.029
13	0.972	0.029	3018.19	13	<0.001	-0.014	0.029
14	0.970	0.029	3219.14	14	<0.001	0.032	0.029
15	0.968	0.029	3415.49	15	<0.001	0.022	0.029
16	0.965	0.029	3607.24	16	<0.001	-0.002	0.029

$p<0.01$ indicates significance of autocorrelation

The ACF and PACF plots for weekly price of garbled black pepper are depicted in Figure 4.67. It was evident that in the ACF plot, spikes upto 16 lags fall above confidence limit and PACF plot showed spikes for a number of lags (1, 2, 4 and 12) beyond confidence limits also indicated the non-stationarity of the time series.

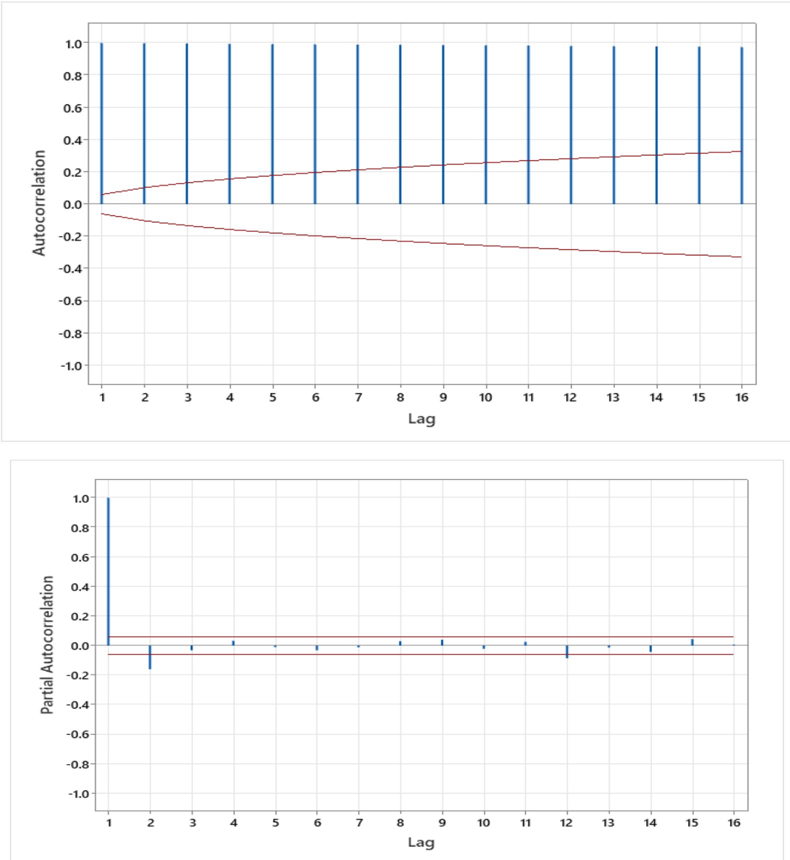


Figure 4.67 ACF and PACF plots for weekly price of ungarbled black pepper

The stationarity of the data was evaluated using ADF test. Following the initial differencing of the data, the ADF test was repeated. The significance of the ADF test statistic, as indicated in Table 4.64, affirmed the necessity of taking the first difference to achieve stationarity in the data.

Table 4.65 ADF test with critical values for weekly price of ungarbled black pepper

Ungarbled Black pepper price	ADF test statistic	Probability(p)	Critical values
Actual	-0.9812	0.9417	1.56
First difference	-20.117**	0.01	-2.32

** indicates significant at 5 per cent level(p<0.05)

Among several ARIMA models tried, the tentative models were chosen based on the value of MAPE, AIC and RMSE. SARIMA (1,1,1) $(1,0,1)_{52}$ was chosen as the best forecast model for weekly price of ungarbled black pepper.

Table 4.66 Model accuracy measures by SARIMA (1,1,1) $(1,0,1)_{52}$ model for weekly price of ungarbled black pepper

Model	MAPE	AIC	RMSE
SARIMA(1,1,1)$(1,0,1)_{52}$	**2.36**	**7648.35**	**9.15**

The parameters of the model SARIMA(1,1,1)$(1,0,1)_{52}$ along with their tests of significance are provided in Table 4.66.

Table 4.67 SARIMA(1,1,1)$(1,0,1)_{52}$ model parameters for weekly price of ungarbled black pepper

Model Parameters	Estimate	SE	Z value	Probability
Non-seasonal difference	1			
AR Lag 1 (ϕ_1)	0.296***	0.002	100.419	0.00
MA Lag 1(θ_1)	-0.116***	0.001	-60.624	0.00
Seasonal difference	0			
AR Seasonal Lag 1(Φ_1)	-0.075***	0.003	-21.747	0.00
MA Seasonal Lag 1(Θ_1)	0.105***	0.001	55.163	0.00

*** indicates significant at 1 per cent level(p<0.01)

The general form of the SARIMA(1,1,1)$(1,0,1)_{52}$ model equation is as follows:

$$(1-\phi_1 B)(1-B)(1-\Phi_1 B^{52})y_t=(1-\theta_1 B)(1-\Theta_1 B^{52})\epsilon_t \quad ,y_t=\log_e Y_t$$

Where, ϕ_1=0.296,Φ_1= -0.075 , θ_1= -0.116,Θ_1=0.105

The SARIMA(1,1,1)$(1,0,1)_{12}$ model equation is as follows:

$$(1-0.296B)(1-B)(1+0.075B^{52})y_t=(1+0.116B)(1-0.105^{52})\epsilon_t$$

The plot of actual and fit values for weekly price of garbled black pepper using SARIMA(1,1,1)(1,0,1)52 model is given in Figure 4.68.

Figure 4.68 Actual and fitted plot for SARIMA(1,1,1) (1,0,1)$_{52}$ for weekly price of ungarbled black pepper

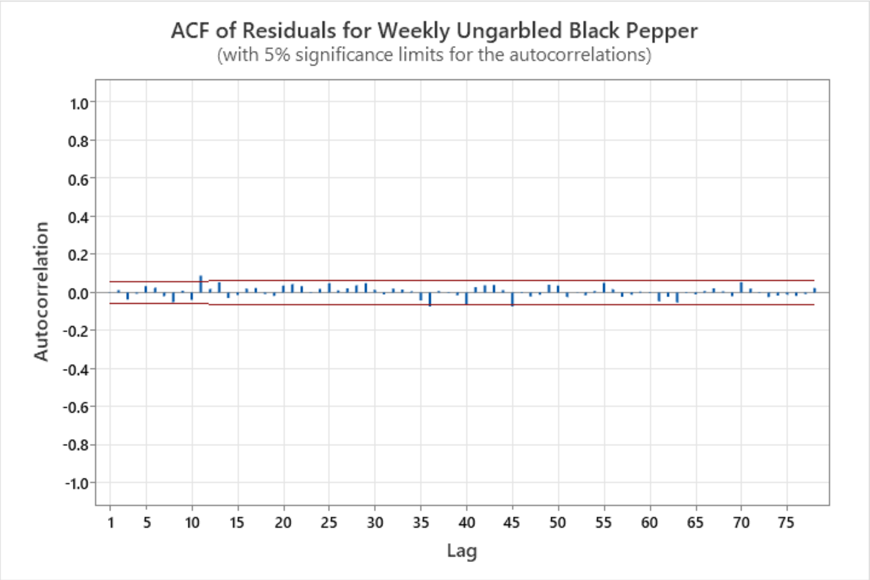
The adequacy of the model was also tested using the value of Box-Pierce Q statistics and was found to be insignificant. So, overall we can say SARIMA(1,1,1)(1,0,1)$_{52}$ model show satisfactory result, among different ARIMA models.

Table 4.68 Ljung-Box 'Q' statistic for residuals of SARIMA(2,1,2)(1,0,1)$_{52}$ model

Statistic	p-value
128.71	0.4574^{NS}

NS: Non-significant

Residual ACF and PACF plots for the SARIMA(1,1,1)(1,0,1)$_{52}$ model fitted for weekly price of garbled black pepper is provided in Figure 4.69.It could be seen that majority of the spikes in the ACF and PACF plots fall within the critical values, thus indicating the adequacy of SARIMA(1,1,1)(1,0,1)$_{52}$ model for forecasting weekly price of ungarbled black pepper .



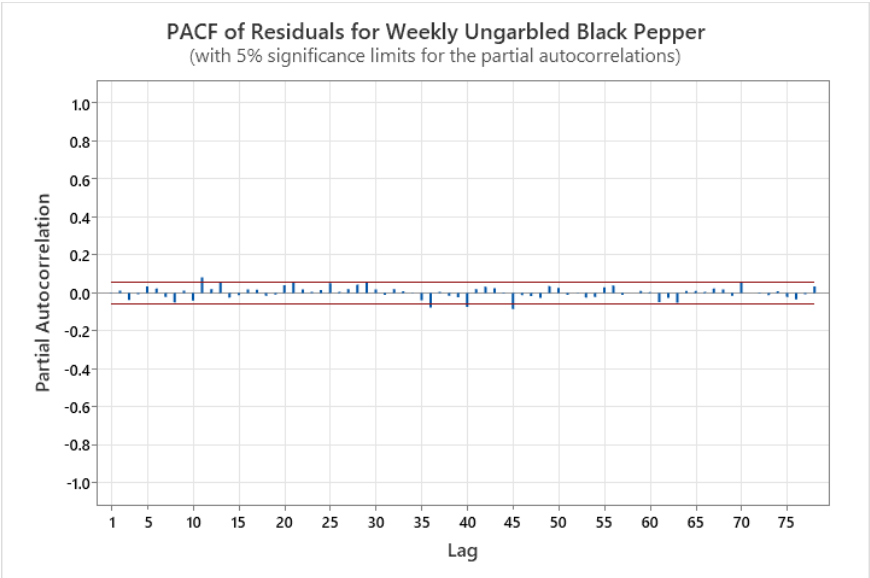


Figure 4.69 Residual ACF and PACF plots for weekly price of ungarbled black pepper for SARIMA(1,1,1) (1,0,1)$_{52}$

4.4.4.2 Volatility in weekly price of ungarbled black pepper

The weekly price of ungarbled black pepper for the period 2000-2020 were used to choose the ARCH family model using R software.

The Heteroscedasticity LM test was carried out to check the volatility or ARCH effect in the time-series. The result of the test is presented in Table 4.69, which reveals that there is an ARCH effect in the time series.

Table 4.69 Heteroscedasticity LM Test for first differenced

Statistic	p-value
467.26***	0.000

N – No. of observations

***Significant at 1 per cent level of significance

Several ARCH models were tested, but their coefficients were determined to be insignificant, highlighting the need for the implementation of the GARCH model.

4.4.4.2.1 GARCH(1,1) model for weekly price of ungarbled black pepper

The estimates of the GARCH (1,1) model fitted for the weekly ungarbled pepper price is given in the Table 4.70. As observed, the constant term, ARCH and GARCH parameters are positive and significant.

Table 4.70 Estimates of GARCH(1,1) model for weekly price of ungarbled black pepper

Parameter	Coefficient	Std. error	t statistic	p-value
Constant term(α_0)	0.0104***	0.009	3.436	0.0065
ARCH term(α_1)	0.1659***	0.034	7.859	0.00
GARCH term(β_1)	0.7632***	0.050	15.196	0.00

***indicates significant at 1 per cent level($p<0.01$)

The GARCH(1,1) model is given by,

h_t= 0.0104+0.1659$\varepsilon_{t-i}{}^2$+0.7632σ_{t-j}^2

The time varying volatility includes a constant (0.0014), a component which depends on past errors (0.1659$\varepsilon_{t-i}{}^2$) and a component which depends on weighted average of past squared residuals (0.7632σ_{t-j}^2). In the Table 4.70, the p-value for the t-statistic of the first order coefficient (7.85) and the second order coefficient (15.19) suggests a significant GARCH (1,1) coefficient.

Table 4.71 Model accuracy measures by GARCH(1,1) model for weekly price of ungarbled black pepper

Accuracy measure	**Value**
RMSE	9.24
MAPE	**2.40**

Residual analysis was carried out to check the adequacy of the selected model. The Serial Correlation LM test of residuals is presented in Table 4.72. The large value of p (p=0.723>0.05) reveals that, there is no serial correlation in the residuals. The Ljung-Box test results is presented in Table 4.73. The large value of p (p=0.389 > 0.05) with respect to Ljung-Box 'Q' statistic indicates that, the residuals are normally distributed.

Table 4.72 Serial Correlation LM test for residuals of GARCH(1,1) Model

Statistic	**p-value**
27.20	**0.732^{NS}**

NS: Non-significant

Table 4.73 Ljung-Box test for residuals of GARCH(1,1) Model

Statistic	p-value
18.29	**0.389^{NS}**

NS: Non-significant

4.4.4.4 ANN model for weekly price of ungarbled black pepper

Time delayed neural network (TDNN) model was fitted for weekly price of ungarbled black pepper. The best time lagged neural network with single hidden layer was found for each series by conducting experiments with the basic cross validation method. Out of a total of 96 neural network structures, a neural network model with twelve lagged observations as input nodes and seven hidden nodes (12:7s:1l) performed better than other competing models in respect of forecasting accuracy measures. This means that most accurate price forecast for the given series is obtained when the price of twelve preceding weeks is used as inputs.

The selected TDNN model is described in Table 4.74 along with the forecasting accuracy measures for both training and testing set.

Table 4.74 Model accuracy measures by TDNN model for weekly price of black pepper

Model	No. of parameters	MAPE		RMSE	
		Train	Test	Train	Test
12:7s:1l	92	4.01	2.22	8.96	7.42

The actual weekly price along with the predicted values of weekly price of ungarbled black pepper at Kochi using TDNN model is provided in Figure 4.70.

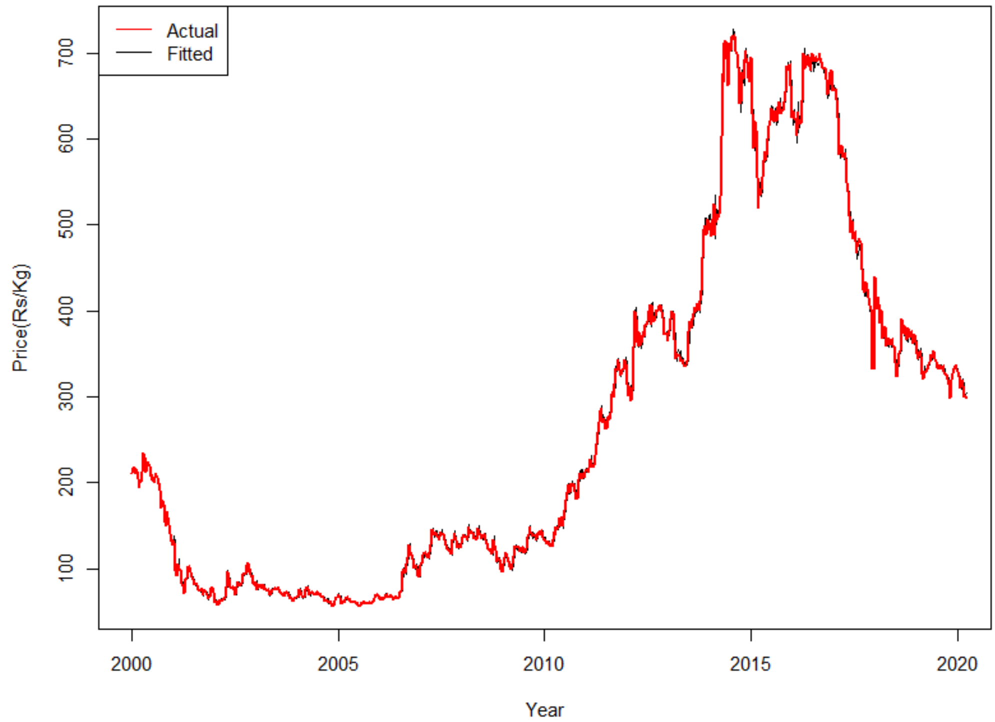


Figure 4.70 Actual and fitted plot for TDNN model of weekly price of ungarbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant provided in Table 4.75. So, overall, we can say TDNN model shows satisfactory result.

Table 4.75 Ljung-Box 'Q' statistic for residuals of TDNN model

Statistic	p-value
133.69	0.2648^{NS}

NS: Non-significant

The residual plot from TDNN model is provided in Figure 4.71 which did not exhibit any specific pattern. The residual ACF and PACF plot is provided in Figure 4.72, majority of the spikes in the residual ACF and PACF are within the critical values indicating the adequacy of the model.

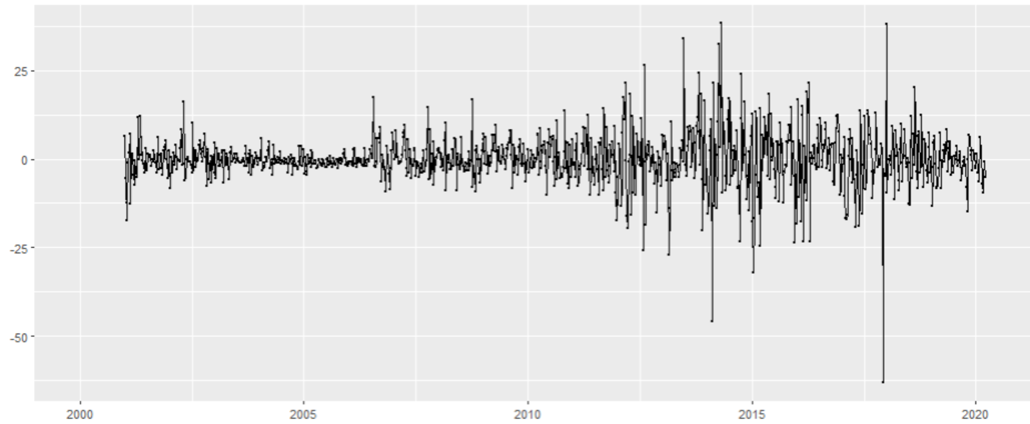


Figure 4.71 Residual plot for weekly price of ungarbled black pepper of TDNN model

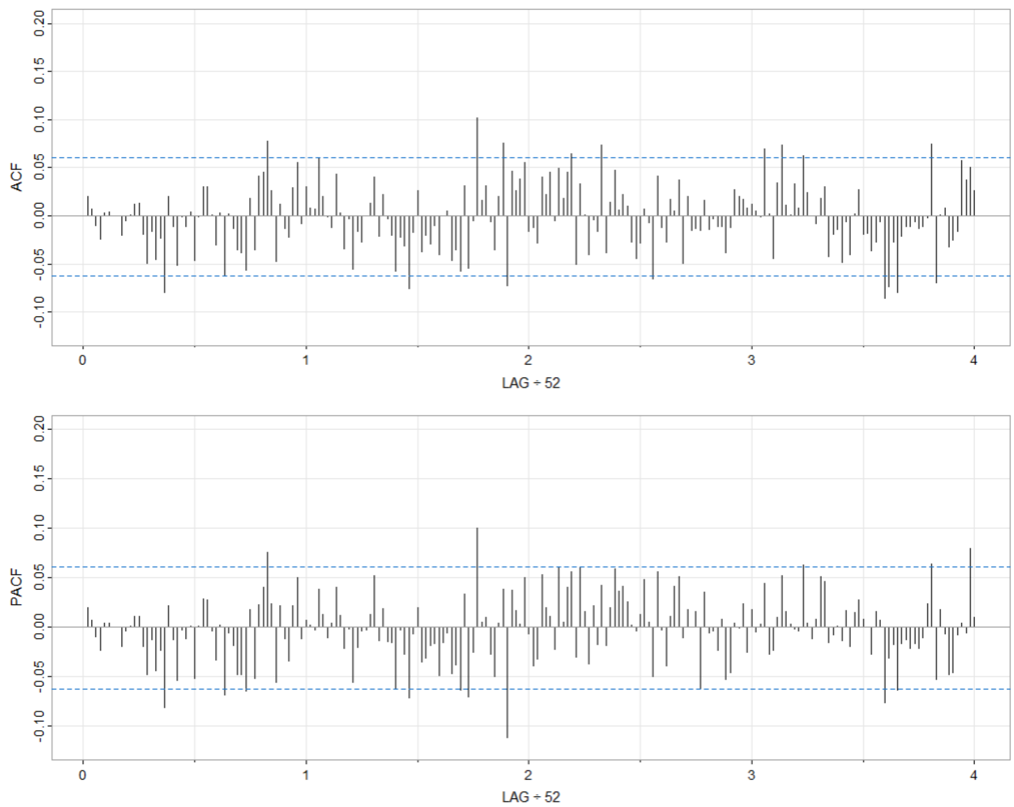


Figure 4.72 Residual ACF and PACF plots for weekly price of ungarbled black pepper for TDNN model

4.4.4.5 LSTM model for weekly price of ungarbled black pepper

LSTM model was fitted for weekly price of ungarbled black pepper. The parameters used for the model was provided in Table 3.1. The LSTM model has been created with three layers by performing epoch of 50 times. During the training process, the model makes predictions on the training data, and the loss function is used to calculate the error between these predictions and the actual target values. The training loss is the average of these errors across all training samples. The goal of training is to adjust the weights and biases of the model to minimize this training loss. After each epoch, the weights of the model get updated and the new epoch works on those updated values that process continues in every epoch. In this study, MAE has been used as the measure to continue the epoch until MAE reaches a minimum which is shown in Figure 4.73.

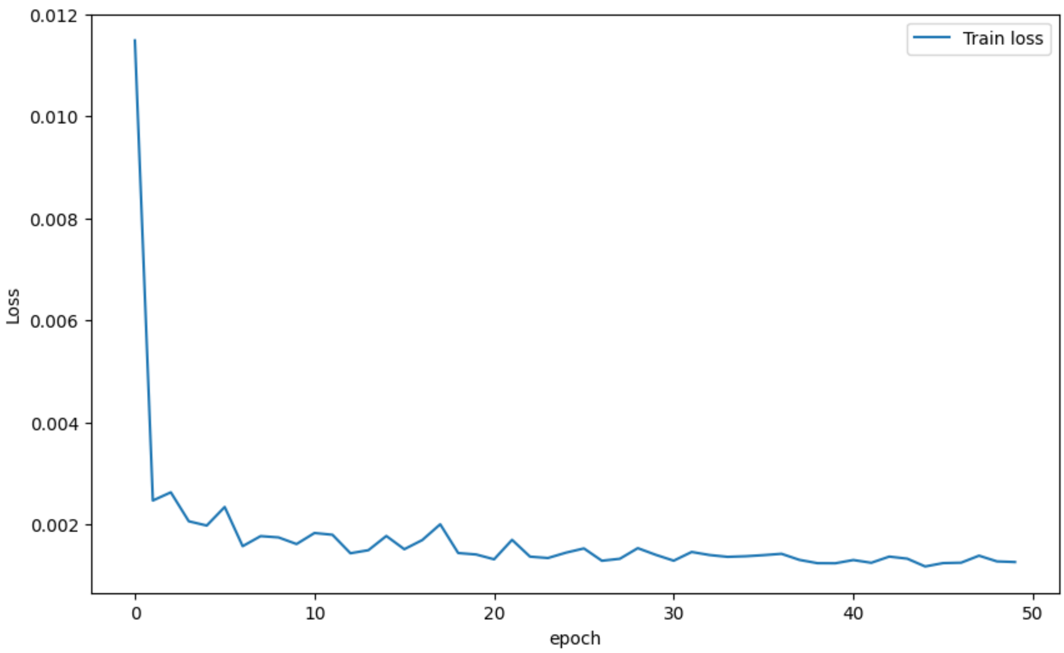


Figure 4.73 Training loss for LSTM model of weekly price of ungarbled black pepper

Various accuracy measures for LSTM model are provided in Table 4.76.

Table 4.76 Model accuracy measures by LSTM model for weekly price of ungarbled black pepper

Accuracy measure	Value
RMSE	16.69
MAPE	6.57

The actual price along with the predicted values for weekly price of ungarbled black pepper using LSTM model is provided in Figure. 4.18.

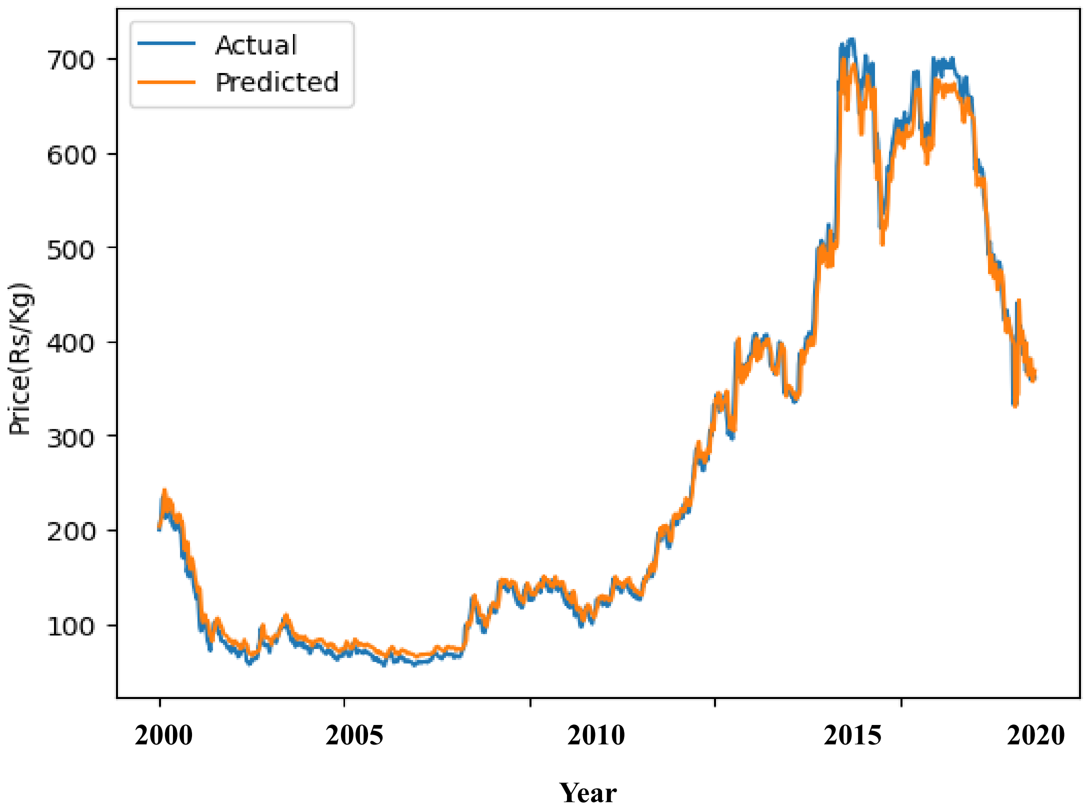


Figure 4.74 Actual and fitted plot for LSTM model for weekly price of ungarbled black pepper

The adequacy of the model was also tested using the value of Ljung-Box 'Q' statistic and was found to be insignificant which is shown in Table 4.77. So, overall we can say LSTM model shown satisfactory result.

Table 4.77 Ljung-Box ‘Q’ statistic for residuals of LSTM model

Statistic	p-value
1502.02	0.700^{NS}

NS: Non-significant

The residual plot from LSTM model is provided in Figure 4.75 and the residuals did not exhibit any specific pattern and are scattered.

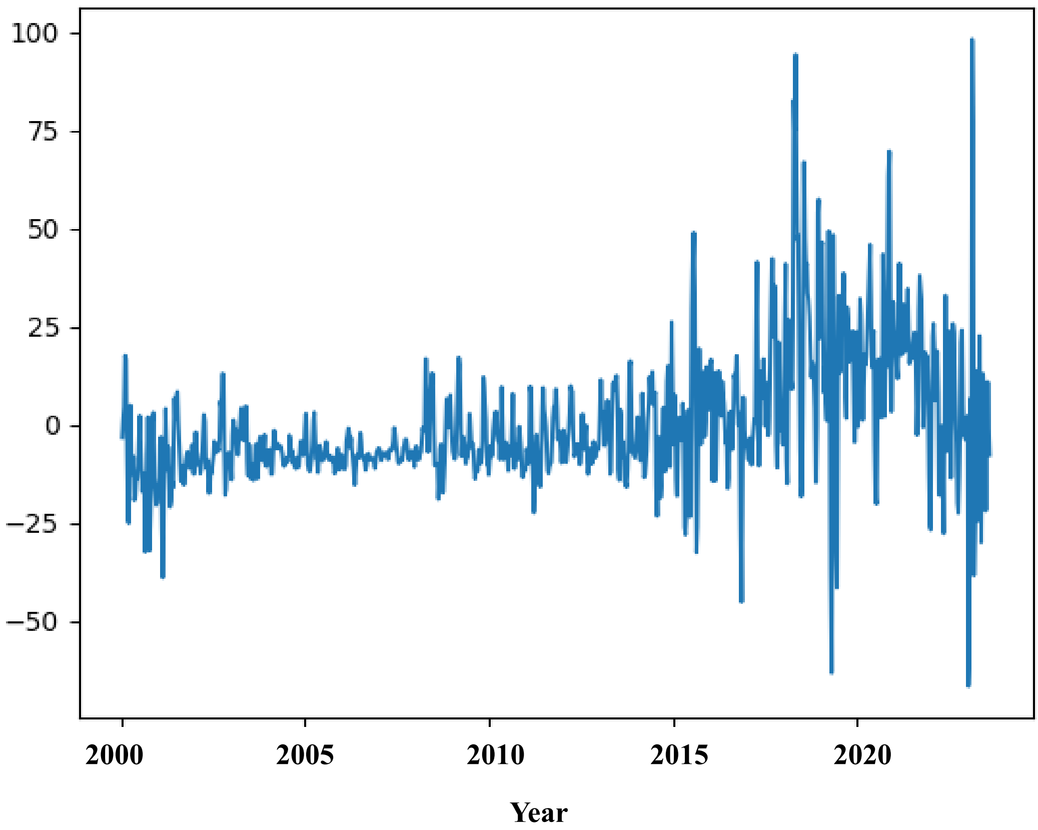


Figure 4.75 Residual plot for weekly price of ungarbled black pepper for LSTM model

4.4.5. Comparison of models

In present study, a comparison of different models has been done in order to know the best model for forecasting weekly price of ungarbled black pepper in Kochi market. The comparison of all the 6 models were carried out based on the test MAPE and RMSE values which were considered to be least. According to the Table 4.78, the TDNN model with least MAPE and RMSE values was considered to be best among all the models considered.

Table 4.78 Comparison of time series forecasting models for weekly price of ungarbled black pepper

Model	MAPE	RMSE
Exponential trend model	40.4	107.11
SARIMA(2,1,2)(1,0,0)	2.36	9.15
GARCH(1,1)	2.40	9.24
HWMS	2.41	9.30
TDNN(12:7s:1l)	**2.22**	**7.42**
LSTM	6.57	10.69

The plot of actual and fit values along with the forecasts for the weekly price of ungarbled black pepper using TDNN model is given in Figure 4.76. The figure showed that the actual and predicted prices are in agreement.

The MAPE and RMSE are found to be 3.36 and 3.58 respectively based on the actual and forecasted weekly price of ungarbled black pepper using the TDNN model for the year 2021 and 2022.

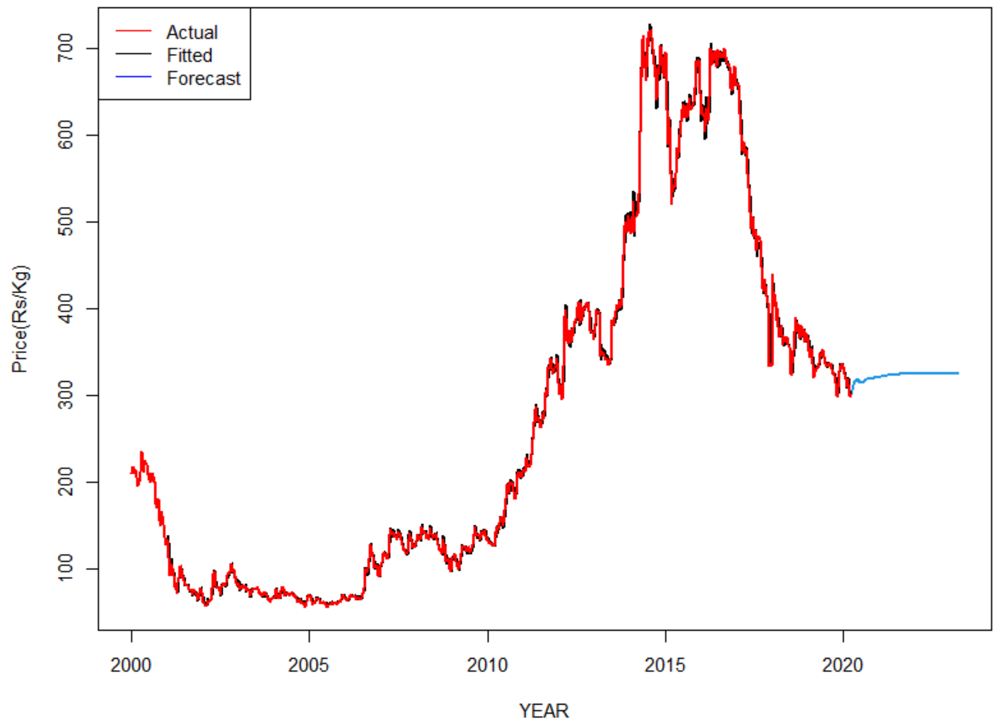


Figure 4.76 Actual, predicted and forecasted plots for TDNN model for weekly price of ungarbled black pepper

The TDNN model captures the pattern of the weekly price of ungarbled black pepper. The MAPE value was 2.22. The residual plot from TDNN model is plotted in Figure 4.71 and was found to be scattered. Further, residual ACF and PACF plots for the TDNN model in Figure 4.72, exhibited that the majority of the spikes fall within the critical values which shows the adequacy of TDNN model. Thus, for weekly price of ungarbled black pepper at Kochi market TDNN model was selected as the best forecast model.

The study observed that the price trends of both garbled and ungarbled black pepper varied in a similar pattern from 2000 to 2020. However, from the year 2014 to 2015, price of garbled black pepper has been increasing. This could be attributed to various factors such as added processing cost involved in garbling, superior quality and consumer appeal of garbled pepper, potential supply shortages and perceived value gain through extra processing steps involved. These factors collectively contribute to a higher price for garbled black pepper, reflecting its premium quality and demand in comparison to ungarbled pepper.

In the present study, both monthly and weekly price of garbled and ungarbled black pepper was considered in order to study how the price behaviour in terms of pattern, trend, seasonality etc. has been across months and weeks for the time duration considered (2000 -2020).

Further the study revealed that the forecasting models for weekly price of both garbled and ungarbled black pepper demonstrated greater accuracy in terms of measures like MAPE and RMSE than the monthly price forecasting models. The probable reason for this could be that the weekly price which is at a granular level gives a clear picture of the pattern of price movements or its volatility on a weekly basis than the monthly price. Based on the various forecasting models fitted for the monthly and weekly price data in the study, TDNN model emerged as the best model to forecast the price of both garbled and ungarbled black pepper in Kochi market of Kerala.

Summary and Conclusion

5. SUMMARY AND CONCLUSION

This chapter provides a summary of the research study titled "A comparative analysis of price forecasting models for black pepper." The motive of the study was to assess various time series models in predicting monthly and weekly price of both garbled and ungarbled black pepper based on data from Kochi market and to recommend appropriate forecasting models.

The study utilized a database comprising time series data encompassing average prices of monthly and weekly both garbled and ungarbled black pepper at the Kochi market. The data, spanning from 2000 to 2020, was sourced from the Spices Board in Kochi.

Time plots for the monthly and weekly prices of garbled and ungarbled black pepper were used to delineate price trends. In order to analyze the time series data of black pepper prices, a multiplicative model was utilized for decomposition. This decomposition process separated the price data into distinct TS components.

Various conventional time series models, including exponential smoothing models, ARIMA, SARIMA, ARCH, and GARCH, were fitted to the price series in order to arrive at the best forecasting model using the R software.

In addition to the traditional method of price forecasting, machine learning techniques like Artificial Neural Network (ANN), and Recurrent Neural Network (RNN) were applied to the data. An ANN is a mathematical model that aims to replicate the functionality and architecture of biological neural networks. In order to effectively handle temporal price series data in neural networks, it is essential to incorporate a form of short-term memory to impart dynamic capabilities which is possible with the Time-delay Neural Network Model (TDNN). TDNN which is a major type of ANN, was employed for this temporal data as it uses time delays at the input layer of the network to build a short-term memory model for forecasting the prices of black pepper. An artificial neural network with a recurrent topology is called a recurrent neural network. Long Short-Term Memory (LSTM) neural network is a specialized type of RNN, as it can learn patterns with long dependencies and is adept at detecting complex patterns.

The price data was split into training and testing data in the ratio of 80:20. The best forecasting model was chosen based on the lower values of the RMSE and MAPE. The predictability performance of the selected model was also evaluated using MAPE.

The examination of the price pattern of black pepper uncovered substantial price fluctuations within the Kochi market. To gain a broad understanding of the price trends, various models, including linear, exponential, and quadratic, were applied. Among the tested models, the exponential model emerged as the most suitable fit for both monthly and weekly prices of both garbled and ungarbled black pepper.

Both monthly and weekly prices exhibited an ascending trend along with significant seasonal variations. Seasonal indices for the months spanning from January to December were computed using the ratio to moving average method for both garbled and ungarbled black pepper prices, reflecting a consistent seasonality pattern in both data sets.

Cyclical variations identified two distinct cycles: the initial 11-year cycle spanning from 1983 to 1993 and the subsequent cycle commencing in 1993, which exhibited fluctuations near peak values and reached its lowest point in 2004. The third cycle, commencing in 2005, demonstrated an increasing trend, reaching its peak in 2014 before exhibiting a subsequent decline. Notably, the third cycle displayed an extended duration in the boom phase, spanning almost nine years, and is currently in the early stages of the slump phase.

Exponential smoothing is a specific type of moving average technique employed in time series analysis to generate smoothed data for forecasting. The HWMS model was observed as the finest among the different exponential smoothing models for this time series data on prices of garbled and ungarbled black pepper. Box and Jenkins introduced a pragmatic three-stage approach for identifying a suitable model called ARIMA, where a good model entails the minimum number of estimated parameters required to effectively capture the patterns in the provided data. SARIMA$(2,1,2)(3,0,2)_{12}$, SARIMA$(2,1,2)(2,0,2)_{12}$, SARIMA$(2,1,2)$ $(1,0,0)_{52}$ and

SARIMA(1,1,1) $(1,0,1)_{52}$ were identified best among several ARIMA models tried for monthly and weekly garbled and ungarbled black pepper respectively.

The fluctuation of commodity prices over a period of time refers to the term price volatility. As the study uses the time series data of black pepper which consist of volatility, the ARCH family models can be employed. The GARCH (1,1) was considered best among the different ARCH family models for this price series data. ANN is a mathematical model that aims to replicate the functionality and architecture of biological neural networks. Time-delay Neural Network (TDNN) which is a major type of ANN was employed for this temporal data as it uses time delays at the input layer of the network to build a short-term memory model for forecasting the prices of black pepper.TDNN (6:2s:1l), TDNN(6:3s:1l), TDNN (13:8s:1l) and TDNN (12:7s:1l) models were found to be the pinnacle artificial neural network model with lower MAPE values 4.29, 4.63, 1.99 and 2.22 in the case of monthly and weekly prices of garbled and ungarbled black pepper respectively. The results revealed that the TDNN model showed superiority in price forecasting of black pepper in all the cases when compared with all other models.

Thus, the TDNN model was used to forecast the prices of black pepper from January 2021 to December 2022. The MAPE value between the actual and forecasted prices for 2021 and 2022 was 4.19 and 4.86 respectively for monthly price of garbled black pepper, while for monthly price of ungarbled black pepper it was 4.09 and 5.05 respectively. The MAPE value between the actual and forecasted prices for 2021 and 2022 was 3.11 and 3.16 respectively for weekly price of garbled black pepper, while for weekly price of ungarbled black pepper it was 3.36 and 3.58 respectively.

Conclusion

The analysis suggested that the TDNN model proves to be a reliable forecasting tool for predicting the price movements of black pepper in the Kochi market. The robustness of the TDNN model offers a plethora of opportunities for understanding the future price pattern of black pepper, which enables various stakeholders, such as producers and traders, to adapt to price fluctuations and for policymakers to ensure market stability. The TDNN model's reliability in forecasting black pepper prices not only enhances market transparency, fostering overall market efficiency in India.

Future line of work

- The study can be extended to various agricultural commodities other than black pepper.
- Inclusion of exogenous variables influencing the price of black pepper can be taken into consideration, thus enabling the study of price pattern in the series in conjunction with the movement in exogenous variable.
- In addition, various time series decomposition techniques such as Empirical Mode Decomposition (EMD), Wavelet etc. can be applied for price of commodities.
- Further, machine learning techniques like Extreme Machine Learning (ELM) and Wavelet-ANN models can be developed for similar series.

References

REFERENCES

Adanaliglu, H. and Yercanm, M. 2012. An analysis of tomato prices at wholesale level in Turkey: An application of SARIMA model [on-line]. Available: http://www.custoseagronegocioonline.com.br. pdf (31 March 2023).

Adebiyi, A.A., Adewumi, A.O., and Ayo, C.K. 2014. Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction. *J. of Appl. Math*.

Aggarwal, C. C. 2018. *Neural Networks and Deep Learning*. Springer Cham Publishers.

Ajjan, N., Selvam, S., Shivakumar, K.M., Anitha, S., Muruganathi, D., and Malarvizhi, A. 2009a. A study on the Application of Artificial Neural Networks (ANN) in commodity price forecasting. Centre for Agricultural and Rural Development Studies, Tamil Nadu Agricultural University, Coimbatore, 63p.

Armstrong, J.S. 2009. Selecting Forecasting Methods, *Principles of Forecasting : A Handbook for Researchers and Practitioners*, M.A: Kluwer Academic Publishers.

Bharathi, R. 2009. Price behavior of Mulberry silk cocoon in Ramnagar and Siddlaghatta market – A statistical analysis. M. Sc. Thesis, Univ. Agric. Sci., Dharwad, Karnataka (India).

Bhavani, D., Srikala, T. and Ananda. 2015. Price volatility in major chilli markets of India. Indian J. Econ. Dev. 3(3): 194-198.

Bollerslev, T. 1986. Generalized autoregressive conditional heteroskedasticity. *J. of Econ.* 31:307-327.

Booranawong, T. and Booranawong, A. 2017. Simple and double exponential smoothing methods with designed input data for forecasting a seasonal time series: In an application for lime prices in Thailand. *Suranaree J. Sci. Technol.* 24(3): 301-310.

Carter, C. and Rausser, G. C. 1981. Commodity futures market efficiency in the soybean complex. *Giannini Foundation of Agric. Econ*.139(2): 30-48.

Chaudhari, D. J. and Tingre, A. S. 2014.Use of ARIMA modeling for forecasting green gram prices for Maharashtra. *J. Food Legumes*. 27(2): 136-139.

Choudhury, K., Jha, G. K., Das, P., and Chaturvedi, K. K. 2019. Forecasting potato price using ensemble artificial neural networks. *Indian J. of Ext. Educ*. 55(1) : 73-77.

Chi, Y. N, and Chi, O. 2021. Modeling and forecasting of monthly global price of bananas using seasonal ARIMA and multilayer perceptron neural network. *Econometrics. Ekonometria. Adv. in Appl. Data Analysis*.25(3): 22-61.

Cermak, M., Malec, K. and Maitah, M. 2017. Price Volatility Modelling – Wheat: GARCH Model Application [on-line]. Agris on-line Papers in Eco. Informatics. 9:15-24. [18 Nov. 2023].

Croxton, F. E., Cowden D. J., Klein, S. Applied general statistics. Prentice Hall of India, New Delhi, India, 1979, 754p.

Deshmukh, S.S., and Paramasivam, R. 2016. Forecasting of milk production in India with ARIMA and VAR time series models. *Asian J. Dairy and Food Res*. 35(1) 20-30.

Dudhat, A. S. 2017. A statistical analysis on price volatility and market integration in major domestic groundnut markets. PhD (Ag) thesis, Junagadh Agricultural University.

Elman, J.L., 1990. Finding structure in time. *Cognitive sci. 14*(2). pp.179-211.

Gan-qiong, L., Shi-wei, L., and Zhe-min L. 2010. Short term price forecasting for Agro-product using Artificial Neural Networks. *Agric. Agric. Sci. Procedia.* 12(1): 278-287.

Gu, Y.H., Jin, D., Yin, H., Zheng, R., Piao, X., Yoo, S.J. 2022. Forecasting Agricultural Commodity Prices Using Dual Input Attention LSTM. *Agric.* 12 (256).

Guha, B. and Bandyopadhyay, G. 2016. Gold Price Forecasting using ARIMA Model.*J. Adv. Manag. Sci.* 4(2): 117-121.

Gutierrez, E.J.C.; Pedro, S.E.; Lopez, L.R. and Pulido, Calvo .2004. Comparison between traditional methods and artificial neural networks for ammonia concentration forecasting in an eel (Anguilla anguilla L.) intensive rearing system. *Aqua. Engg.* 31(3/4): 183-203.

Haykin, S. 1999. Artificial neural networks (2nd ed.). Pearson Education.

Huang, W.R.; Xu, B. and Chan, H.A. 2004. Forecasting flows in Apalachicola River using neural networks. *Hyd. Pro.* 18(13): 2545-2564.

Huertas Garcia, R. and Rodriguez, B.M. 2007. Forecasting international tourist demand using Holt-Winters. *Papers de Turisme, Spain* .41: 75-90.

Indiastat. 2023. Indiastat. home page [on-line]. Available: https://www.indiastat.com.[28 Nov. 2023].

Indraji, K.N. 2016. Price forecast models for coconut and coconut oil. M.Sc.(Ag) thesis, Kerala Agricultural University, Thrissur.

International Pepper Community. 2023. International Pepper Community. Home page [on-line]. Available: https://www.ipcnet.org/ [28 Nov.2023].

Jalikatti, N. K., Patil, B. L., Basavaraj, H., Kunal, L. B., Yeledhalli, R.A., and Kataraki, P. A. 2013. Spatial and temporal variations in arrivals and prices of onion in northern Karnataka- An econometric analysis. *Karnataka J. Agric. Sci.* 26(4): 565-566.

Jha, G.K., Sinha, K. 2013. Agricultural Price Forecasting Using Neural Network Model: An Innovative Information Delivery System. *Agril. Econ. Res. Rev.* 26 (2) : 229 – 239.

Joy, D. 2021. Time series modelling and forecasting of tea prices in India.M.Sc.(Ag) thesis, Kerala Agricultural University, Thrissur.

Jyothi, T. 2011. Price behaviour of Turmeric in India. M.Sc. (Ag) thesis, Kerala Agricultural University, Thrissur. 152p.

Kurumatani, K. 2020. Time series forecasting of agricultural product prices based on recurrent neural networks and its evaluation method. *SN Appl. Sci.* 2 : 1434.

Kachroo. M.M., Wani. M.H., Baba S.H., Nazir .N., Lone F.A.,Shaheen. F.A., and Bhat, A. 2021. Analysis of trend and seasonality in prices and arrivals of selected agricultural commodities in India. *J. of Plant Dev. Sci.* 13(8): 575-581.

Krishnakutty, C. N. 2014. Forecasting of teak prices in Kerala State, India, using autoregressive integrated moving average models. *Indian J. For.* 24(2): 119-122.

Kuruvila, A., Bastine, L. C., and Joseph, J. 2012. Assessing the implications of international price volatility and transmission on Indian agricultural prices in the context of trade liberalization. SANEI Working Paper Series No. 12-01. South Asia Network of Economic Research Institutes, Dhaka.

Kumar, M. T. L., Suresh, H. S., and Munirajappa, R. 2011. Holt-Winters Exponential smoothing and Seasonal ARIMA time series technique for forecasting of onion price in Bangalore market. Mysore J. Agric. Sci. 45(3): 602-607.

Kumar, M. T. L., Munirajappa, R., and Suresh, H. S. 2011. Application of Seasonal ARIMA model for forecasting monthly prices of potato in Bangalore market. Mysore J. Agric. Sci. 45(4): 778-782.

Kumaraswamy, N. and Sekar, C. 2014. Growth Performance and Price Forecasting of Potato - An Economic Analysis. Trends Biosci. 7(20): 3315-3320.

Kumari, P., Goswami, V., N. H, and Pundir, R.S. (2023) Recurrent neural network architecture for forecasting banana prices in Gujarat, India. *Plos One.*18(6): 27-56.

Kumari, P., Parmar, D.J., Kumar, S.M., Mahera, A.B., Lad, Y.A. 2022.Comparison of Statistical Models for Prediction Area, Production and Yield of Citrus in ujarat. *Biol. Forum.* 14(2) : 540- 545.

Mani, P., Nagesh Kumar, D., & Jayaraman, V. 2005. Comparison of ARIMA and rtificial neural networks models for river flow forecasting. *J. of Hydrology*.30(1) :229-246.

Manoj K. and Madhu A. (2014). An application of time series ARIMA forecasting model for predicting sugarcane production in India. *Stud. in Business and Econ*. 9(1):81-94.

Mahto, A., Afshar, M., Biswas, R., Ahmed, J., and Shah, I., 2021. Short-Term Forecasting of Agriculture Commodities in Context of Indian Market for

Sustainable Agriculture by Using the Artificial Neural Network. *J. of Food. Qual.* 13 (1).

Makridakis, S. and Hibbon, M. 1979. Accuracy of forecasting : An empirical investigation. *J. of the Roy. Stat. Soc. A*. 41(2): 97-145.

Menon, P. P., Kuruvila, K. M., Madhusoodhan, K. J., and Thomas, J. 2006. Autoregressive models for forecasting cardamom price. *J. Plant. Crops.* 34(3) : 679-680.

Nabi, D. M. and Shahabuddin, M. Y. 1998. A statistical forecast of rubber prices *.J. Nat. Rubb. Res.* 3(2): 69-80.

Naveena, K., Rathod, Santosha, Shukla, Garima and Yogish, K.J. 2014. Forecasting of coconut production in India: A suitable time series model. *Int. J. Agric. Engg.* 7(1) : 190-193.

Naveena, K. 2016. Statistical modelling for forecasting of price and export of Indian coffee. Ph.D. Thesis, Banaras Hindu University, Varanasi, Uttar Pradesh, India.

Naveena, K., Singh, S., Rathod, S., and Singh, A. 2017. Hybrid ARIMA-ANN modelling for forecasting the price of Robusta coffee in India. *Int. J. of Cur. Microbiology and Applied Sci*. 6(7):17–26.

Prathima, C.M. (2018). Statistical modelling of arrivals and prices of tender coconut in Maddur market. M.Sc. Thesis. University of Agricultural Sciences, Bengaluru, Karnataka.

Rangoda, B. D. P., Abeywickrama, L. M. and Fernado, M. T. N. 2004. An analysis of different forecasting models for prices of coconut products in Sri Lanka. *Proceedings of the Third Academic Sessions*. Coconut Research Institute, Lunuwila. Pp 1-16.

Rathod, S., Singh, K.N., Patil, S.G., Naik, R.H., and Ray, M. 2018. Modeling and forecasting of oilseed production of India through artificial intelligence techniques. *Indian J. of Agric. Sci.* 88(1): 22-27.

Rathod, S., Chitikela, G., Bandumula, N., Ondrasek, G., Ravichandran, S., and Sundaram, R.M. 2022. Modeling and Forecasting of Rice Prices in India during the COVID-19 Lockdown Using Machine Learning Approaches. *Agron.* 10(2): 12-21.

Reeja, V. 2011. Price behaviour of natural rubber in India. M.Sc. (Ag) thesis, Kerala Agricultural University, Thrissur, 76p.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1986. Learning representations by back-propagating errors. *Nat. 323*(6088). pp.533-536.

Sabu, S. S., 2015. Price volatility of black pepper and its implications in Kerala. M.Sc. (Ag) thesis, Kerala Agricultural University, Thrissur, 109p.

Sabu, S. S., Kuruvila, A. and Manojkumar, K. 2019. Price behaviour of black pepper in Indian and International markets: a comparative analysis. *J. of Spices and Aromat. Crops.* 28(1): 27-33.

Sangsefidi, S. J., Moghadasi, R., Yazdani, S., & Mohamadi Nejad, A. 2015. Forecasting the prices of agricultural products in Iran with ARIMA and ARCH Models. *Int. J. of Advanced and Appl. Sci.* 2(11) : 54-57.

Sharma, H., Burrak, S.S. 2015. Moth Bean Price Forecasting in Churu Market of Rajasthan: A Comparison of ARIMA and ANN Models. *Agric. Res. J.* 52 (3) :96-100.

Spiegel, R. M. 1992. Theory and problems of statistics (2nd Ed.). Mc Graw-Hill International, Singapore. 504p

Sreekanth, P. D., Geetanjali, H., Sreedevi, P. D., Ahmed, S., Kumar, R. N. and Kamla Jayanthi, P. D. 2009. Forecasting groundwater level using artificial neural networks. *Curr. Sci.* 96(7): 933.

Sutradhar, T., Hazari, S. and Nagpure, S. C. 2019. Effect of seasonal and cyclical variation in prices of natural rubber in India. *Int. J. Sci. Environ*. 8(3):696- 701.

Tajdini, A., Tavakkoli, A., Latibari, A. J., Roohnia, M., and Pourmousa, S. 2014. Statistical modelling to forecast the wood -based panels consumption in Iran. *Int.J. Biosci*. 4(12):1- 11.

Tsay, L.S., Analysis of Financial Time Series, 2nd ed., Hoboken, N.J: Wiley, 2005.

Vasanthakumar, T., Sudheendra, M., and Mallikarjuna, H. B. 2005. Forecasting of price of teak (Tectona grandis Linn.) – By exponential smoothing analysis. *Environ. Ecol.* 32(2): 566-570.

Vijay, N., and Mishra, G.C. 2018. Time Series Forecasting Using ARIMA and ANN Models for Production of Pearl Millet (BAJRA) Crop of Karnataka, India. *Int. J. of Curr. Microbiol. and App. Sci.* 7(12): 880-889.

Yang, A. J., Michael, S., Haigh, B., and David J. L. 2001. Agricultural liberalization policy and commodity price volatility: a GARCH application. Appl. Econ. Lett. 8(9): 593-598.

Zhang, G., Patuwo, B. E., Hu, M. Y. 1998. Forecasting with artificial neural networks: The state of the Art. *Int. J. of Forecast*. 14: 35-62.

Appendices

APPENDIX I

R codes :

#Loading the packages

```
library(tseries)
library(forecast)
library(ggplot2)
library(stats)
install.packages("dygraphs")
library(dygraphs)
library(forecast)
library(e1071)
library(tseries)
library(tidyverse)
library(fNonlinear)
library(lmtest)
library(quantmod)
library(astsa)
library(rugarch)
library(tseries)
library(FinTS)
```

#defining dataset

```
data = MONTHLY
data
garbled = data$GARBLED
garbled
ts=ts(garbled,start=c(2000,1),end=c(2020,12),frequency = 12)
ts
summary(ts)
```

#decomposing data

```
ddata <-decompose(ts,"multiplicative")
ddata
plot(ddata)
plot(ddata$trend)
plot(ddata$seasonal)
plot(ddata$random)
```

#adf test

```
s<-adf.test(ts)
adf.test(ts, k=1)
```

#first difference

```
ts.1=diff(ts)
ts.1
adf.test(ts.1,k=1)
```

#ARIMA

```
m1=Arima(ts,order=c(2,1,2),seasonal=list(order=c(3,0,2)))
summary(m1)
coeftest(m1)
forecast<-forecast(m1)
forecast
fitted2=m1$fitted
fitted2
```

```
#plotting observed, fitted and forecasted value
ts
str(ts)
```

```
fit1=Arima(ts,order=c(2,1,2),seasonal=list(order=c(3,0,2)))
summary(fit1)
prd=fitted(fit1)
prd
f1<-forecast(fit1,h=48)
f1
plot( prd, type = "l", col = "blue", xlab = "Year", ylab =
"Price(Rs/Kg)",xlim=c(2000,2020))
lines( ts, col = "red",lwd= 2)
lines(f1$mean,col="green",lty=2)
plot(f1,xlab="YEAR",ylab="Price(Rs/Kg)")
lines( prd, col = "black",lwd=2)
lines(ts,col="purple")
legend("topleft",legend=c("Actual","Fitted"),col=c("red","black"),lty=1)
```

#RESIDUAL

```
checkresiduals(fit1)
Box.test(residuals(fit1),lag=10,type="Ljung-Box",fitdf=1)
accuracy(fit1$residuals)
acf(fit1$residuals)
pacf(fit1$residuals)
```

#HWMS

```
garbled.exp<-hw(ts,seasonal= c("multiplicative"), h=48)
garbled.exp
autoplot(forecast(garbled.exp))
```

#assessing model

```
summary(garbled.exp)
checkresiduals(garbled.exp)

# forecast the next 5 quarters
garbled.f4 <- forecast(garbled.exp, h = 48)
garbled.f4
```

#RESIDUAL

```
checkresiduals(garbled.exp)
Box.test(residuals(garbled.exp),lag=10,type="Ljung-Box",fitdf=1)
accuracy(fit1$residuals)
acf(fit1$residuals)
pacf(fit1$residuals)
acf2(garbled.exp$residuals)
```

#plotting

```
fit2=hw(ts,seasonal= c("multiplicative"),h=100)
summary(fit2)
prd2=fitted(fit2)

prd2
f2<-forecast(fit2,h=48)
f2
plot( prd2, type = "l", col = "black",lwd=2, xlab = "Year", ylab =
"Price(Rs/Kg)",xlim=c(2000,2020),ylim=c(50,800))
lines( ts, col = "red",lwd=2)
lines(f2$mean,col="green",lty=2)
plot(f2,xlab="YEAR",ylab="Price(Rs/Kg)",ylim=c(100,800))
```

```
lines( prd2, col = "red")

lines(ts,col="purple")

legend("topleft",legend=c("Actual","Fitted"),col=c("red","black"),lty=1)


#ARCH AND GARCH

#ARCH

install.packages("rugarch")

library(rugarch)

data = Pepper_Prices_garbled_and_ungarbled_monthly

data

garbled = data$GARBLED

garbled

ts=ts(garbled,start=c(2000,1),end=c(2020,12),frequency = 12)

ts

returns <- diff(log(ts))  # Compute log returns

# Fit ARCH model

arch_model <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder =
c(1, 0)), mean.model = list(armaOrder = c(0, 0)))

arch_model

arch_fit <- ugarchfit(spec = arch_model, data = ts)

arch_fit

# Print the model summary

print(arch_fit)

# Fit GARCH(p, q) model

garch_model <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder =
c(1, 1)), mean.model = list(armaOrder = c(0, 0)))

garch_fit <- ugarchfit(spec = garch_model, data = returns)

garch_fit
```

 # TDNN Model

```
library(dplyr)
library(tseries)
library(forecast)
library(FinTS)
library(stats)
library(MLmetrics)
library(readxl)

#GARBLED MONTHLY
monthlygarb = Pepper_Prices_garbled_and_ungarbled_monthly
attach(monthlygarb)
monthlygarb = monthlygarb$GARBLED[1:252]
len <- length(monthlygarb)
monthlygarb #selected till december 2020
Data=ts(monthlygarb, start = c(2000,1), frequency = 12)  #frequency changed to 12
Data
TimedelayNN<-function(Data,input.lag,Neuron,train_size)
{
  n1 <- length(Data)-train_size
  TrI <- Data[1:n1]
  TeI <- Data[-c(1:n1)]
  {
    RMSE1<-NULL;MAE1<-NULL;MAPE1<-NULL;RR<-NULL;RAN<-
NULL;RMSE2<-NULL
    MAE2<-NULL;MAPE2<-NULL;Homo<-NULL;Auco<-NULL;Auco1<-
NULL;Homo1<-NULL;SW1<-NULL;SW<-NULL
    for (i in 1:input.lag) {
      for (n in 1:Neuron) {
        set.seed(12)
```

```
f=forecast::nnetar(TrI,p=i,size=n,scale.inputs = T,repeats = 10)
fore=forecast::forecast(TrI,h=train_size,model=f)
Model=c(i,n)
g=input.lag+1
l=length(TrI)
r=tseries::runs.test(factor(sign(na.omit(f$residuals))))
h=FinTS::ArchTest(f$residuals,lags = 12)
b=Box.test(f$residuals,lag=12,type = c("Ljung-Box"))
s=shapiro.test(na.omit(f$residuals))
Tr_RMSE=MLmetrics::RMSE(f$fitted[g:l],TrI[g:l])
Tr_MAE=MLmetrics::MAE(f$fitted[g:l],TrI[g:l])
Tr_MAPE=MLmetrics::MAPE(f$fitted[g:l],TrI[g:l])*100
Te_RMSE=MLmetrics::RMSE(fore$mean,TeI)
Te_MAE=MLmetrics::MAE(fore$mean,TeI)
Te_MAPE=MLmetrics::MAPE(fore$mean,TeI)*100
ABS=Tr_RMSE/Te_RMSE
Auco<-rbind(Auco,b$p.value)
Auco1<-rbind(Auco1,b$statistic)
RAN<-rbind(RAN,r$p.value)
Homo<-rbind(Homo,h$p.value)
Homo1<-rbind(Homo1,h$statistic)
SW1<-rbind(SW1,s$statistic)
SW<-rbind(SW,s$p.value)
RR=rbind(RR,Model)
RMSE1<-rbind(RMSE1,Tr_RMSE)
MAE1<-rbind(MAE1,Tr_MAE)
MAPE1<-rbind(MAPE1,Tr_MAPE)
RMSE2<-rbind(RMSE2,Te_RMSE)
MAE2<-rbind(MAE2,Te_MAE)
```

```
      MAPE2<-rbind(MAPE2,Te_MAPE)

     }
   }
 }

TT=cbind(RR,RMSE1,MAE1,MAPE1,RMSE2,MAE2,MAPE2,Auco1,Auco,Homo1,
Homo,SW1,SW)
  return(TT)
}
train_size = len * 0.8
train_size = as.integer(train_size)
Last<-TimedelayNN(Data, input.lag =6, Neuron = 6, train_size)
meanvalues<-as.data.frame(Last)
meanvalues
model=forecast::nnetar(Data,p=6,size=2,scale.inputs = T,repeats = 20)
summary(model)
model
fore1=forecast::forecast(Data,h=24,model=model)
model
fore1
```

Python Code for LSTM model:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import SimpleRNN,LSTM
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
```

```
df1=df['GARBLED']
#import libraries
import pandas as pd
import numpy as np
import tensorflow as tf
import os
df=pd.read_excel("/content/Pepper Prices (garbled and
ungarbled)monthly DATA - Copy.xlsx")
df['YEAR'] = pd.to_datetime(df['YEAR'])
df.set_index('YEAR',inplace=True)
df.head()
train_size = int(len(df1)*0.8)
test_size = len(df1) - train_size
train = df1.iloc[:train_size]
test = df1.iloc[train_size:len(df1)]
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler(feature_range=(-1,1))
train1 = scaler.fit_transform(train.values.reshape(-1, 1))
def create_dataset(train, look_back=1):
  dataX, dataY = [], []
  for i in range(len(train)-look_back-1):
    a = train[i:(i+look_back), 0]
    dataX.append(a)
    dataY.append(train[i + look_back, 0])
  return np.array(dataX), np.array(dataY)
look_back = 10
trainX, trainY = create_dataset(train1, look_back)
trainX = np.reshape(trainX, (trainX.shape[0], 1,
trainX.shape[1]))
# Create LSTM model
def create_LSTM(units):
    model = Sequential()
    # Input layer
    model.add(LSTM(units = 100, return_sequences =
True,input_shape = [trainX.shape[1], trainX.shape[2]]))

    # Hidden layer
    model.add(LSTM(units = 100))
    model.add(Dense(units = 1))
    #Compile model
    model.compile(loss='mean_squared_error', optimizer='adam')
    return model
model_LSTM = create_LSTM(32)
def fit_model(model):
    history = model.fit(trainX, trainY, epochs = 50,
                        batch_size = 1, verbose=2)
```

```
    return history
model.summary(model_LSTM)
plot_loss (history_LSTM, 'LSTM')
def prediction(model):
    prediction = model.predict(trainX)
    prediction = scaler.inverse_transform(prediction)
    return prediction
prediction_LSTM = prediction(model_LSTM)
from sklearn.metrics import mean_squared_error,
mean_absolute_percentage_error,mean_absolute_error
train_rmse = np.sqrt(mean_squared_error(train[look_back+1:],
prediction_LSTM))
train_rmsetrainPredict1  = pd.DataFrame(prediction_LSTM)

t1=trainPredict1['Prediction']
t2 = train
t3 = t2
t4=t3.reset_index(drop=True)
t5=t4.iloc[look_back+1:]
t5.index=np.arange(1,len(t1)+1)
t1.index = np.arange(1,len(t1)+1)
res=t5-t1
plt.plot(res)
import statistics as stats
import statsmodels.api as sm
sm.stats.acorr_ljungbox(res, lags=[12], return_df=True)
def mape(actual, predicted):
    actual, predicted = np.array(t5), np.array(t1)
    return np.mean(np.abs((actual - predicted) / actual)) * 100
actual = t5
predicted = t1
mape(actual, predicted)
MSE = np.square(np.subtract(actual,predicted)).mean()
import math
RMSE = math.sqrt(MSE)
print("Root Mean Square Error:\n")
print(RMSE)
# Plot the predictions and actual values
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# Set specific years as x-axis ticks
plt.xticks(dates, [date.year for date in dates])
plt.plot(actual, label='Actual')
plt.plot(predicted, label='Predicted')
```

```
plt.legend()
plt.xlabel('Year')
plt.ylabel('Price(Rs/Kg)')
plt.show()
```

A COMPARATIVE ANALYSIS OF PRICE FORECASTING MODELS FOR BLACK PEPPER

By

AKSHAYA AJITH

(2021-19-004)

ABSTRACT OF THE THESIS

Submitted in partial fulfillment of the

requirement for the degree of

Master of Science in Agriculture

(AGRICULTURAL STATISTICS)

Faculty of Agriculture

Kerala Agricultural University

DEPARTMENT OF AGRICULTURAL STATISTICS

COLLEGE OF AGRICULTURE

VELLANIKKARA, THRISSUR - 680656

KERALA, INDIA

2024

ABSTRACT

Black pepper, the 'king of spices' is one of the most popular and widely consumed spices which shares a place on most dinner tables with salt. India is the third largest producer in the world (International Pepper Community, 2023), also a significant consumer and exporter of black pepper, with Kerala and Karnataka producing the majority of the nation's output. Kerala ranked second in terms of black pepper acreage (76,160 ha), and production (33290 metric tonnes) but seventh in terms of productivity (0.44 metric tonnes/ha) (GOI, 2023). Historically, the market value of pepper contributed to the development of the city of Kochi as a centre of international commerce. Kochi has the first exclusive pepper exchange in India which was established by the Indian Pepper and Spice Traders Association, IPSTA and the exchange was well regulated by the traditional players here, without any default on supply or delivery of the commodity and volatility.

As per latest records of Spices Board, the price of black pepper surpassed Rs.500/Kg which was the maximum price in the past years since 2022 in Kochi market and this indicated that black pepper prices are highly volatile. Being a perennial crop, the large variation in prices of black pepper within a year is a major problem faced by farmers as well as consumers. Hence, analysis of time series data of prices of black pepper is of prime importance.

In this context, the present study was undertaken to evaluate different time series models for prices of black pepper and to suggest suitable forecast models for Kochi market. Time series data on monthly and weekly average prices of garbled and ungarbled black pepper at Kochi market from January 2000 to December 2020, collected from Spices Board, Kochi formed the database for the study. Analysis of price pattern revealed that wide fluctuation exists in the prices of black pepper in Kochi market. In order to have a general idea about trend of prices of black pepper, models like linear, exponential and quadratic, were fitted. From among several models tried, exponential model was found to be best fit for the monthly and weekly prices of both garbled and ungarbled black pepper. To examine the time series data for the price of black pepper, a multiplicative model was employed for decomposition. Seasonal indices for the 12 months from January to December was calculated for both garbled

and ungarbled black pepper prices as the seasonal variation were present in monthly and weekly data.

Different traditional time series models such as exponential smoothing models (single, double, Holt-Winters' multiplicative models (HWMS)), Auto Regressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA), Auto Regressive Conditional Heteroskedasticity (ARCH) and Generalized Auto Regressive Conditional Heteroskedasticity (GARCH) were applied to the price data using R software. The Augmented Dickey-Fuller test and Heteroscedasticity Lagrange's Multiplier test were used to test the stationarity and volatility of the time-series respectively.

In addition to the traditional method of price forecasting, machine learning techniques like Artificial Neural Network (ANN), and Recurrent Neural Network (RNN) were applied to the data. An artificial neural network (ANN) is a mathematical model that aims to replicate the functionality and architecture of biological neural networks. Time-delay Neural Network (TDNN) which is a major type of ANN was employed for this temporal data as it uses time delays at the input layer of the network to build a short-term memory model for forecasting the prices of black pepper. An artificial neural network with a recurrent topology is called a recurrent neural network. Long Short-Term Memory (LSTM) neural network, a specialized type of RNN, as it possesses the capability to learn patterns with long dependencies and is adept at detecting complex patterns. The price data was split into training and testing data in the ratio of 80:20. The best forecasting model was determined based on the lower values of the Root Mean Square (RMSE) and Mean Absolute Percentage Error (MAPE). The predictability performance of the selected model was also evaluated using MAPE.

The HWMS model was observed as the finest among the different exponential smoothing models for this time series data on prices of garbled and ungarbled black pepper. SARIMA(2,1,2)$(3,0,2)_{12}$, SARIMA(2,1,2)$(2,0,2)_{12}$, SARIMA(2,1,2) $(1,0,0)_{52}$ and SARIMA(1,1,1) $(1,0,1)_{52}$ were identified best among several ARIMA models tried for monthly and weekly garbled and ungarbled black pepper respectively. The GARCH (1,1) was considered best among the different ARCH family models for this price series data. TDNN (6:2s:1l), TDNN(6:3s:1l), TDNN (13:8s:1l) and TDNN (12:7s:1l) models were found to be the pinnacle artificial neural network model with lower MAPE values 4.29, 4.63, 1.99 and 2.22 in the case of monthly and weekly prices of garbled and ungarbled black pepper respectively. The results revealed that the TDNN model showed superiority in price forecasting of black pepper in all the cases when compared with all other models.

Thus, the TDNN model was used to forecast the prices from January 2021 to December 2022. The MAPE value between the actual and forecasted prices for 2021 and 2022 was 4.19 and 4.86 respectively for monthly price of garbled black pepper, while for monthly price of ungarbled black pepper it was 4.09 and 5.05 respectively. The MAPE value between the actual and forecasted prices for 2021 and 2022 was 3.11 and 3.16 respectively for weekly price of garbled black pepper, while for weekly price of ungarbled black pepper it was 3.36 and 3.58 respectively.

In conclusion, the analysis suggested that the TDNN model proves to be a reliable forecasting tool for predicting prices of black pepper in the Kochi market. The robustness of the TDNN model offers a plethora of opportunities for understanding the future price pattern of black pepper which enables, various stakeholders such as producers and traders to adapt with the price fluctuations and for policymakers to ensure market stability. The TDNN model's reliability in forecasting black pepper prices not only enhances market transparency fostering overall market efficiency in India.