CONTENTS

ABSTRACT

The human beings always want more profits without much risk from investment. The gold is the one having less risk and much profits. Now, the thing is that if we know in advance the gold price for next few years. Then, it'll be much easier for us to invest in gold. Now, we get overall idea about the gold investment and factors impacting gold price.

In this project, we'll find out the how the different factors impact gold prices. Also, we try to estimate the gold price by using linear regression techniques depending on different factors. After that, we use different time forecasting techniques for the forecasting future gold price. So, we'll get the most awaited answer for the question "What'll be the profit if I invested in Gold?". Also, we'll get the required time period for that much profit to earn. Let's find out the solutions to the questions in this report!!!

INTRODUCTION

Gold is the most popular in all precious metals as an investment. Investors generally buy gold as a way of diversifying risk, especially through the use of futures contracts and derivatives. The gold market is subject to speculation and volatility as are other markets. Compared to other precious metals used or investment, gold has been the most effective safe haven across a number of countries.

There are a many of precious metals, but gold is placed in high regard as an investment. Due to some influencing factors such as high liquidity and inflation-beating capacity, gold is one of the most preferred investments in India. Gold investment can be done in many forms like buying jewellery, coins, bars, gold exchange-traded funds, Gold funds, sovereign gold bond scheme, etc.

Though there are times when markets see a fall in the prices of gold but usually it doesn’t last for long and always makes a strong upturn. Once you have made your mind to invest in gold, you should decide the way of investing meticulously. If you want to know more about Gold Investment plans and other facts like different ways to buy and invest in gold, how to invest in gold online and much more, you are at the right place

Why invest in gold?

Gold is a unique asset, highly liquid, yet scarce. It's a luxury good as much as an investment. Gold is no one's liability and carries no risk. As such, it can play a fundamental role in an investment portfolio. Gold acts as a diversifier and a vehicle to mitigate losses in times of market stress. It can serve as a hedge against inflation and currency risk.

Key facts that investors should know:

- ✓ Gold is a mainstream asset driven by many factors, not just investment demand.
- ✓ Gold is one of the most effective diversifiers
- ✓ Gold provides competitive returns compared to other major financial assets
- ✓ Gold offers downside protection and positive performance
- ✓ Gold gives great returns in long term view

The combination of these factors means that adding gold to a portfolio can enhance risk-adjusted returns.

Factors impacting Gold price:

The prices of the gold are increasing and the price of the gold is affected by the various factors. The factors like exchange rate of US dollar with INR, Crude oil pricesand inflation rate impact the gold prices. Each of the factors is studied with the gold prices. The relationship between the factor and the gold prices is emphasized in past research papers. From the past, we had got some relationship between different factors and gold prices.

OBJECTIVES OF THE PROJECT

1. To analyse the patterns in gold price
2. To study and analyse the impact of different factors on gold price
3. To fit a forecasting model using time series techniques and predict the future prices

ABOUT DATA

We are going to analyse about the gold prices in this project. Obviously, we collected data from the secondary sources. The data for the gold prices taken from the source https://www.gold.org/goldhub/data/gold-prices. The data consists of gold prices from year 1980 to 2021. We have daily, monthly and yearly data. In this data, we have prices of gold in rupees per troy ounce. So, I had converted gold price to the rupees per 10 gram.

The data for USD-INR exchange downloaded from the sourcehttps://excelrates.com/historical-exchange-rates/USD-INR.
https://www.investing.com/currencies/usd-inr-historical-data. The data for inflation rates downloaded from https://www.macrotrends.net/countries/IND/india/inflation-rate-cpi.

STATISTICAL TOOLS AND TECHNIQUES USED

In this project, we first visualize the pattern in gold price over the years using different visualization techniques such as line chart, bar plot, etc. then we will find out the impact of different factors on gold prices. For this, we use the correlation and regression techniques to see the relationship between them and to predict future prices. Here, we use simple linear regression for predicting prices.

At the last, we will use time series forecasting techniques to forecast the future price of gold. The techniques such as exponential smoothing, ARIMA and SARIMA models to find out the best model for the future price prediction.

We are doing this project using the Python language.

Notations used – TS: time series

DATA ANALYSIS

❖ Data visualization –

We have the daily time series data of gold prices from 02-01-1979 till 14-05-2021 with corresponding gold price at each date. Data contains 11054 rows & 2 columns.

1. Let's look at the time series plot of gold prices over last 40 years.

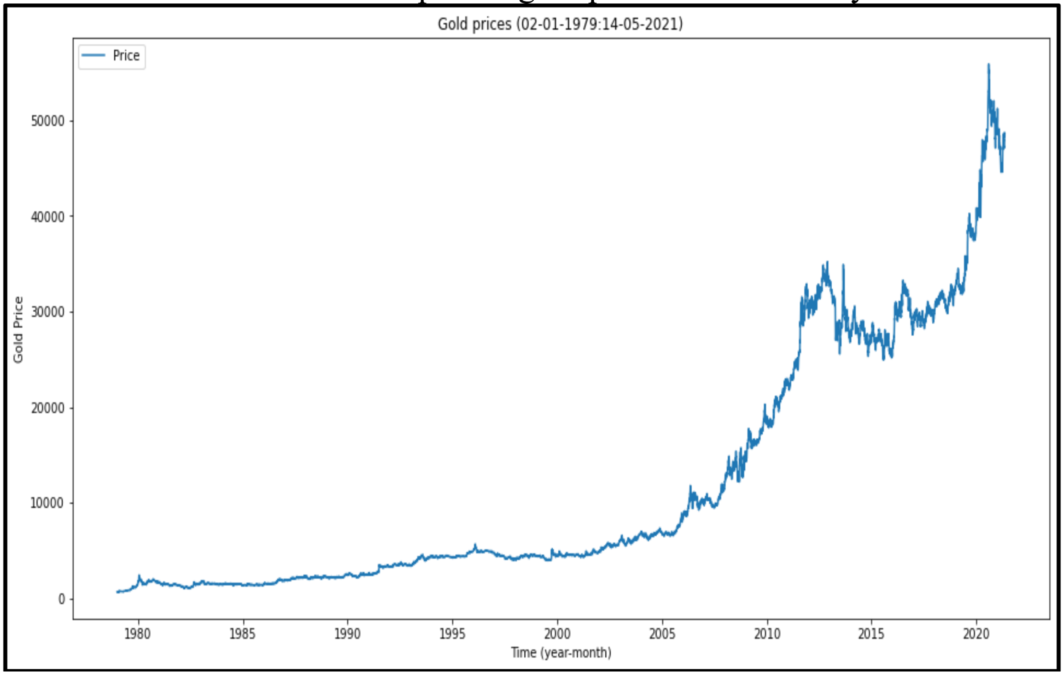

Fig. 1 – Time series plot of gold prices from 02-01-1979 to 14-05-2021

From the above plot, we can see that there is no sudden increase in gold price from 1979 till 2005. The prices are increasing slowly. But, after 2005 to 2012, there was boom in the gold prices due to nifty fall, excess liquidity, exchange accept gold as collateral and some more facts. Again, the gold price consolidates in the range of 20000 to 30000 till 2020. After that the gold price again breakout to next high of 56900 in just a year.So, the graph shows the presence of trend and seasonality.

2. Now, let's find out that the average gold price increase as year pass by.

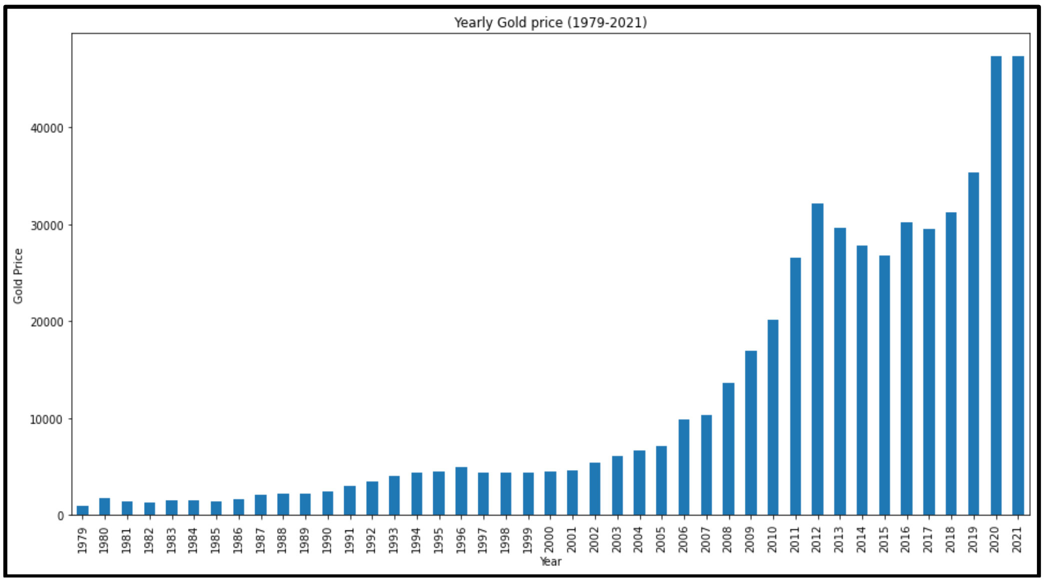

Fig. 2 Yearly average gold price from 1979 to 2021

The above bar plot of yearly gold price from 1979 to 2021 shows exponential growth in price over the years.

Now, we'll aggregate the time series to monthly time series to reduce the noise and make it more stable & hence it would be easier to fit a model. The plot for the aggregated time series is as follows -

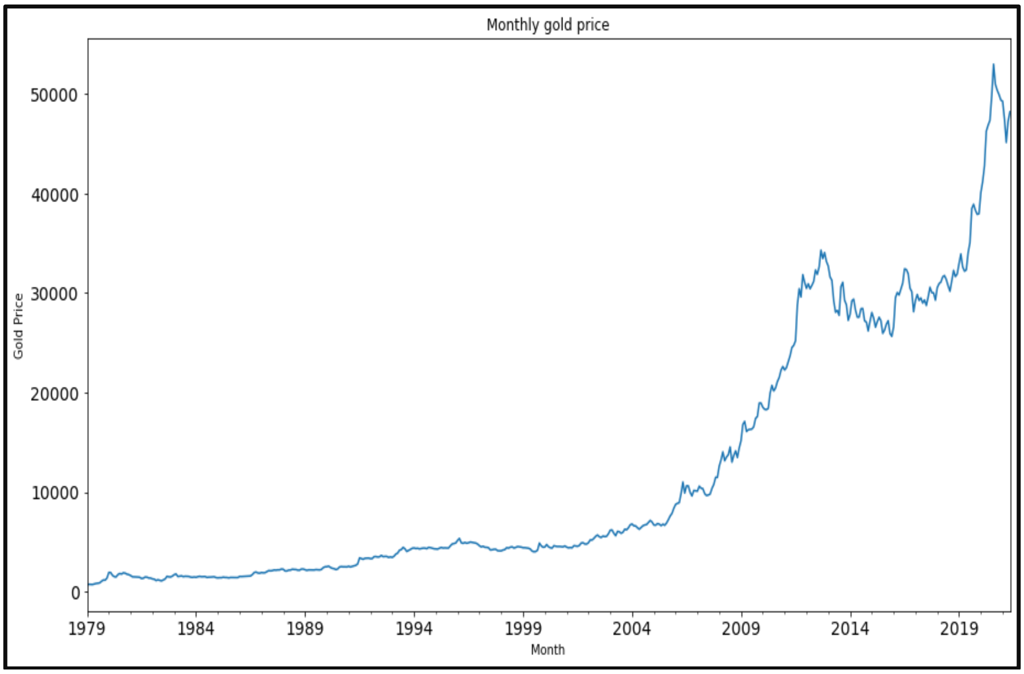

Fig. 4 – Monthly gold price (Jan 1979 – May 2021)

- **Splitting the data into training & validation part**

Now we will divide our data in train & validation. We will make a model on the train part & predict on the validation part to check the accuracy of our predictions.To divide the data into training & validation set, we will take last months as the validation data and rest for training data. We will take 76 months as trend will be the most in them. The starting date of the dataset is 31-01-1979 as we have seen in the exploration part & the end date is 31-05-2021.

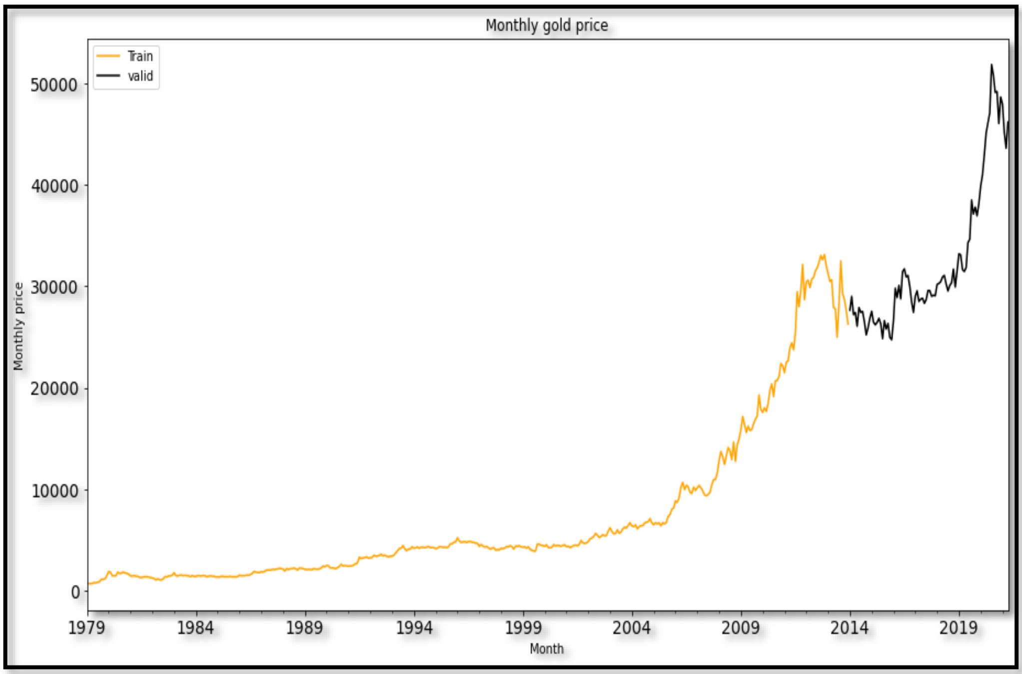

Fig. 5 – Training and validation part on time series plot

The above plot shows the training and validation parts of time series. The yellow colour line shows the training part and black colour line shows the validation part.

Now, we will look at various models now to forecast the TS. Methods for the forecasting are:

i. Naive Approach
ii. Moving Average
iii. Simple Exponential Smoothing
iv. Halt's Linear Trend Model

❖ Time Series forecasting–

- **Naive Approach-**
  In this forecasting technique, we assume that the next expected point is equal to last observed point. So, we can expect a straight horizontal line as the prediction.

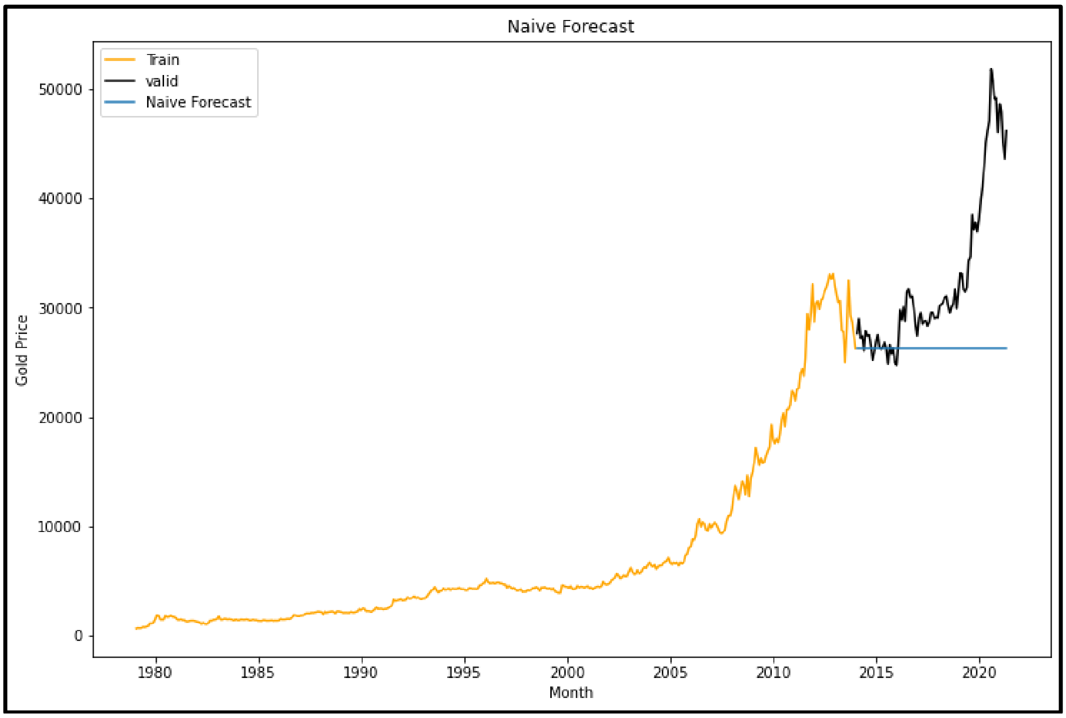

Fig. 6 – Naive Approach forecasting

The above plot shows the gold price forecasting using naive approach method. We can see that it is not good fit on the validation part.

The value of residual mean sum of square by Naive approach = 9560.3325

The RMSE also indicates that the Naïve approach is not best for predicting. So, we will use moving average method.

- **Moving Average method**–
  In this technique we will take the average of the gold price for last few time periods only. Here, we take average of last 12 months.

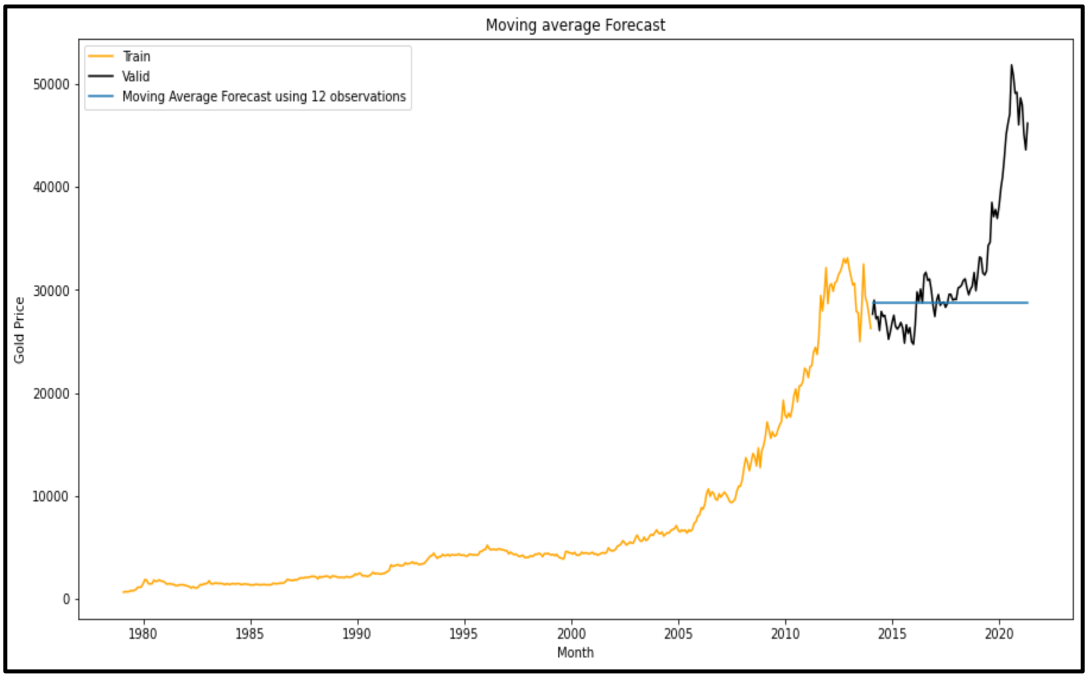

Fig. 7 – Moving average method forecasting

The above plot shows the gold price forecasting using moving average method. We can see that it is not good fit on the validation part.

The value of residual mean sum of square by Moving average method = 8124.6771

The RMSE also indicates that the moving average method is not best for predicting. So, we will use simple exponential smoothing.

- **Simple Exponential smoothing** –
  In this technique, we assign larger weights to more recent observations than to observations from the distant past. The weights decrease exponentially as observations come from further in the past, the smallest weights are associated with the oldest observations.

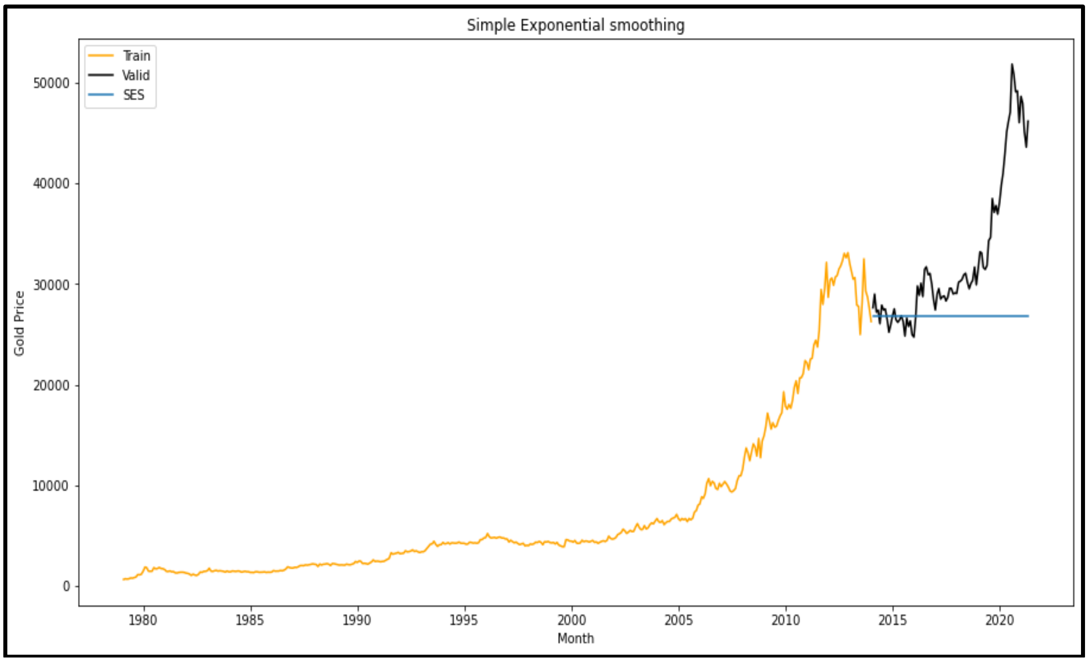

Fig. 8 – Simple Exponential smoothing forecasting

The above plot shows the gold price forecasting using simple exponential smoothing method. We can see that it is not good fit on the validation part.

The value of residual mean sum of square by Moving average method = 9220.7856

The RMSE also indicates that the simple exponential smoothing method is not best for predicting. So, we will use double exponential smoothing (Holt linear model).

- **Triple Exponential smoothing (Holt Winter method)** –
  Datasets which show a similar set of pattern after fixed intervals of a time period suffer from seasonality.The above models don't take into account the seasonality of the dataset while forecasting. Hence, we need a method that takes into account both trend & seasonality to forecast future prices.
  One such algorithm that we can use in such a scenario is Holt's Winter method. The idea behind Holt's Winter is to apply exponential smoothing to the seasonal components in the addition to level & trend.

  Let's fit the model on training dataset & validate it using the validation dataset.

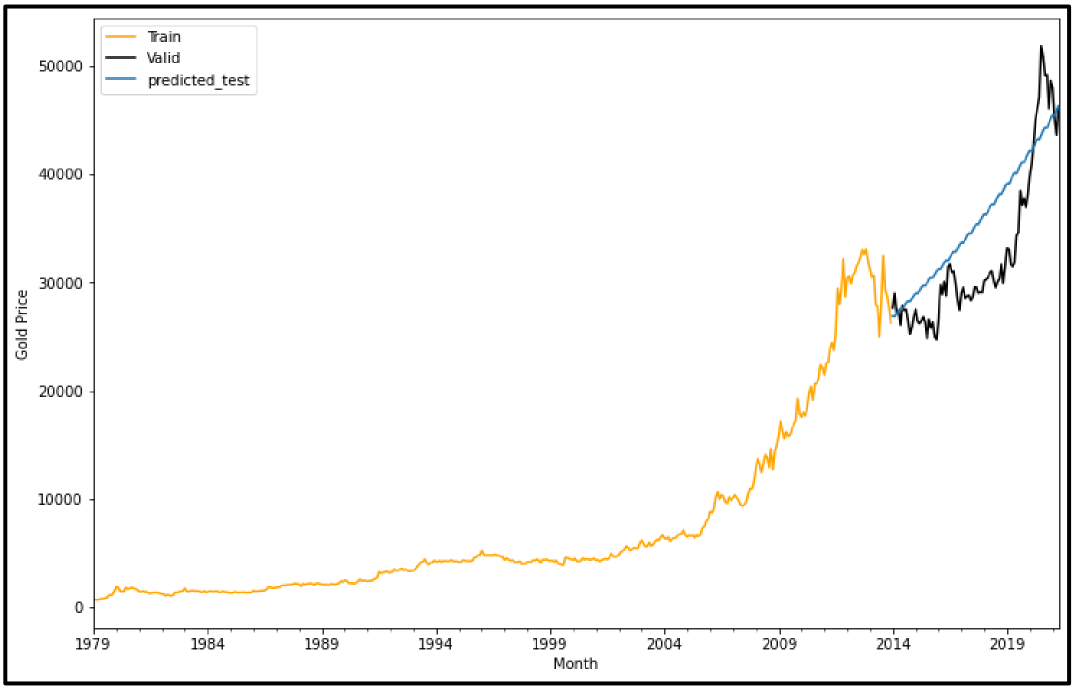

Fig. 9 – Holt Winter model forecasting

The above plot shows the gold price forecasting using triple exponential smoothing method. We can see that it seems good fit on the validation part, but not best fit.

The value of residual mean sum of square by Moving average method = 4727.8236

The RMSE also indicates that the Triple exponential smoothing method is not best for predicting. So, we will use ARIMA and SARIMA model for prediction.

- **ARIMA Model –**
  - ✓ ARIMA stands for Auto Regression Integrated Moving Average. It is specified by 3 ordered parameters (p,d,q).
  - ✓ Here p is the order of the autoregressive model (number of time lags)
  - ✓ d is the degree of differencing (number of times the data have had past values subtracted)
  - ✓ q is the order of moving average model.
  - ✓ The ARIMA forecasting for a stationary TS is nothing but a linear (like a linear regression) equation.

There are 3 basic criterion for a series to be classified as stationary series:

- ✓ The mean of the TS should not be a function of time. It should be constant.
- ✓ The variance of the TS should not be a function of time.

- ✓ The covariance of the i^{th} item & the $(i+m)^{th}$ term should not be a function of time.

We make the series stationary to make the variables independent. Variables can be dependent in various ways, but can only be independent in one way. So, we will get more information when they are independent. Hence, the TS must be stationary.

If the timeseries is not stationary, firstly we have to make it stationary. For doing so, we need to remove the trend & seasonality from the data.

Now, we have to make sure that the TS is stationary. If the TS is not stationary, we'll make it stationary.

For checking stationarity –

- We use Dickey Fuller test to check the stationarity of the series.
- The intuition behind this is that it determines how strongly a TS is defined by trend.
- The null hypothesis of the test is the TS is not stationary (has some time-dependent structure).
- The alternate hypothesis(rejecting the null hypothesis) is that the TS is stationary.
- If the 'Test statistic' is less than the 'Critical value', We can reject the null hypothesis & say that the series is stationary or we can use p-value, if p-value less than level of significance (α), we reject the null hypothesis.

From the rolling mean and rolling standard deviation, we have following graph for the original data –

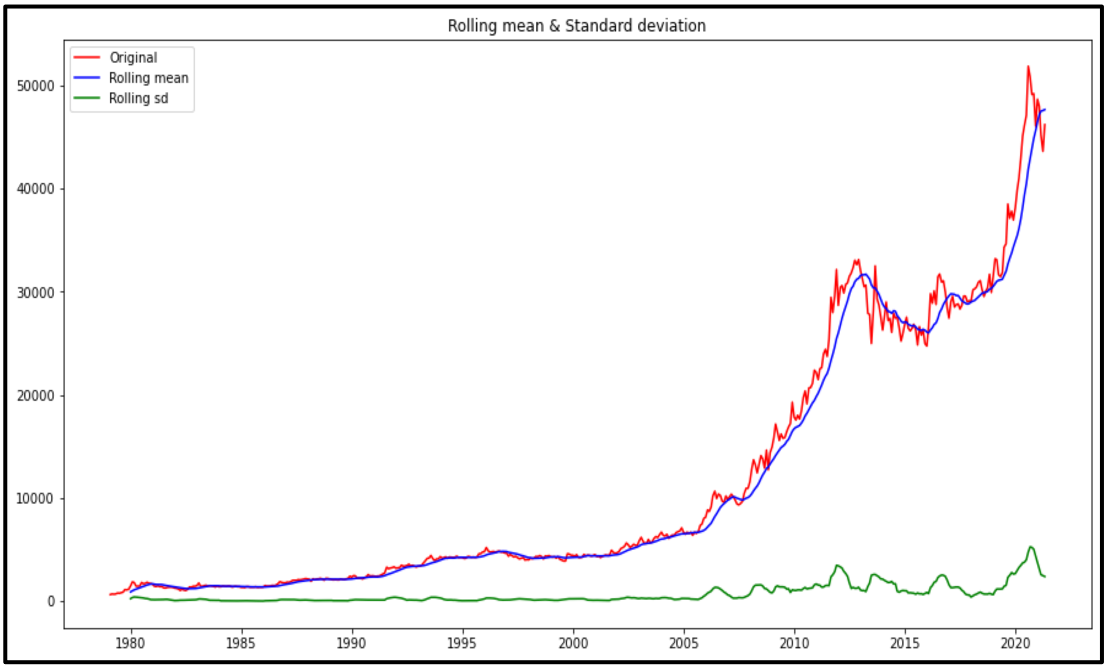

Fig. 10– Rolling mean &Rolling standard deviation plot

From the above plot, we can see that there is presence of stationarity. Still, we confirm it by using the results of Dickey Fuller test.

The result of Dickey-Fuller test is –

Results of Dickey-Fuller test	
Test Statistic	1.897364
p-value	0.998524
#Lags_used	8
Numberofobservationsused	499
Critical Value (1%)	-3.44352
Critical Value (5%)	-2.86735
Critical Value (10%)	-2.56986

Here, the p-value is 0.998524 which is greater than 0.05, hence we do not reject H_o. so, the original data is not stationary.

Now, we will try to removetrend in the data.

A trend exists when there is a long-term increase or decrease in the data. It does not have to be linear.

We see an increasing trend in the data so we can apply transformation which penalizes higher values more than smaller ones, we use log transformation.

We will take the log of gold prices for each year and then plot.

Now, we will plot the transformed series.

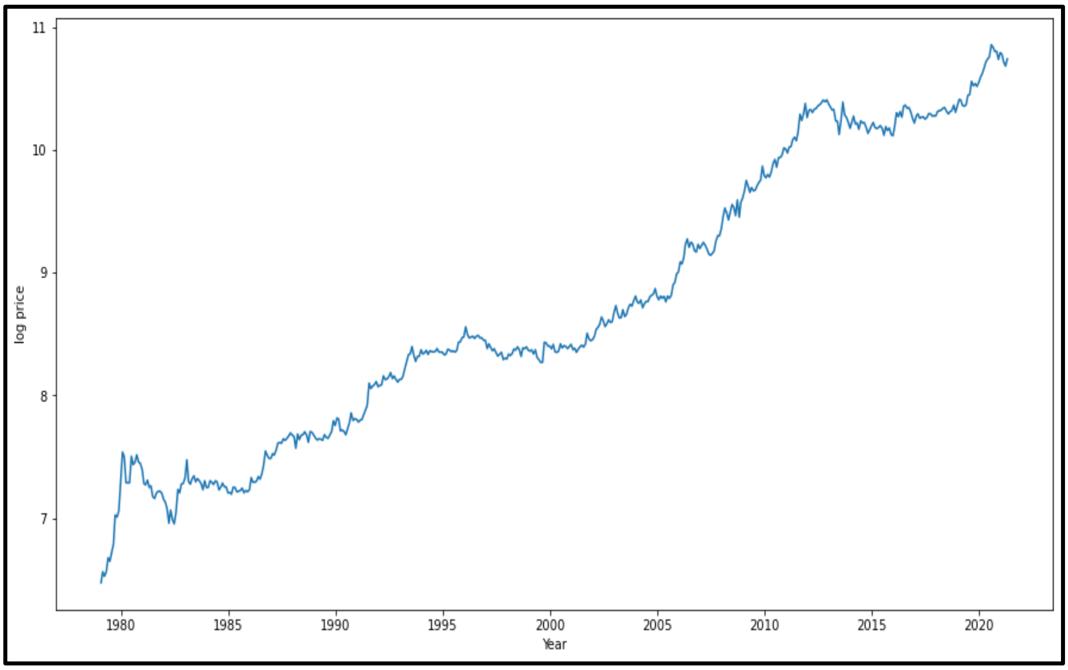

Fig. 11 – Transformed series plot

So we can observe an increasing trend. Now we will remove this increasing trend to make our TS stationary. Let's now stabilize the mean of the TS which is also a requirement for a stationary time series. Differencing can help to make the series stable & eliminate the trend.

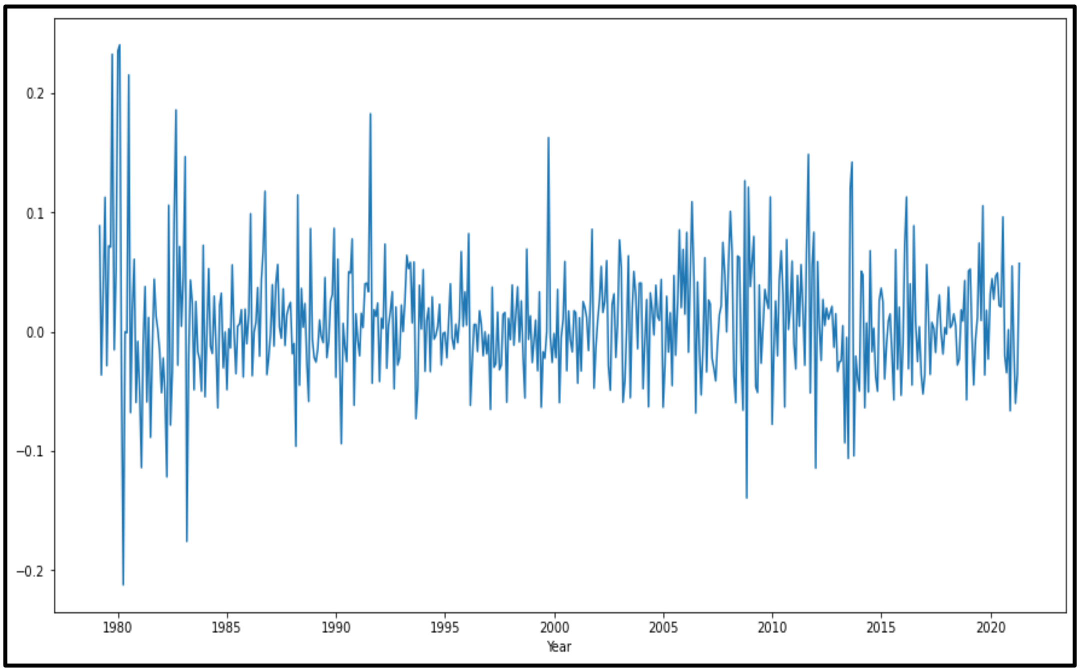

Fig. 12 – Differenced series plot

Now, let us check it stationarity.

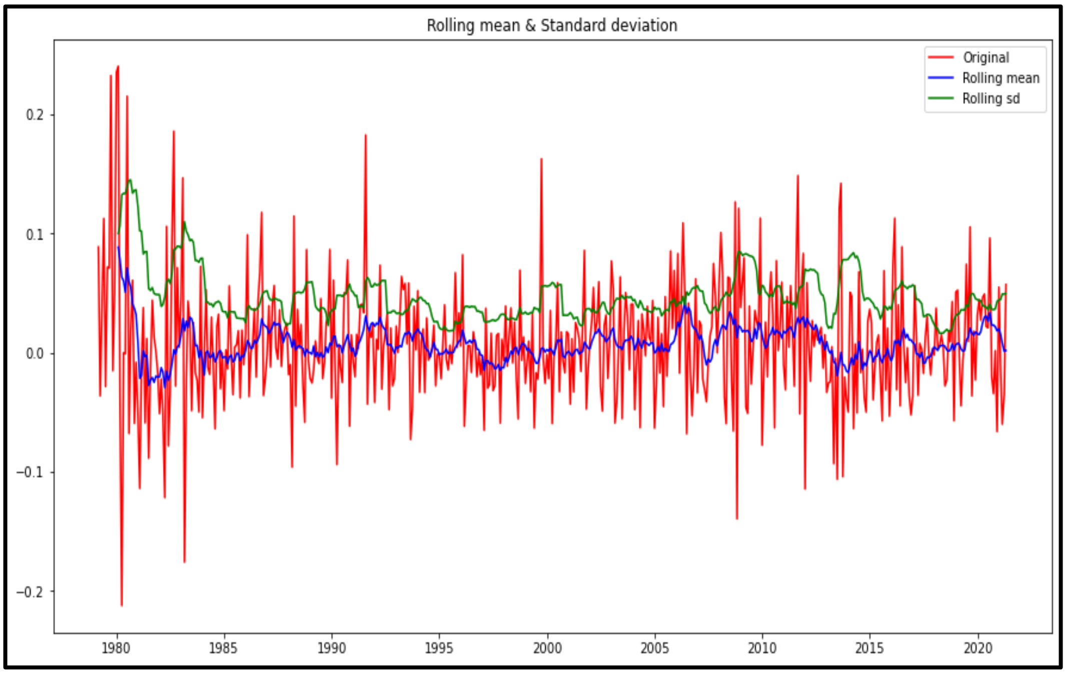

Fig. 13 – Rolling mean &Rolling standard deviation plot

From the above plot, we can see that there is presence of stationarity. Still, we confirm it by using the results of Dickey Fuller test.

The result of Dickey-Fuller test is –

Teststatistic	-17.49
p-value	0.00
#Lags_used	1.00
Numberofobservationsused	505.00
Critical Value (1%)	-3.44
Critical Value (5%)	-2.87
Critical Value (10%)	-2.57

Here, the p-value is 0.00 which is less than 0.05, hence we reject H_o. so, the original data is stationary.

Let's choose the best model –

Performing stepwise search to minimize aic

ARIMA(0,1,0) intercept : AIC=-1553.584

ARIMA(1,1,0) intercept : AIC=-1554.085

ARIMA(0,1,1) intercept : AIC=-1554.427

ARIMA(0,1,0) : AIC=-1542.528

ARIMA(1,1,1) intercept : AIC=-1552.960

ARIMA(0,1,2) intercept : AIC=-1553.777

ARIMA(1,1,2) intercept : AIC=-1550.502

ARIMA(0,1,1) : AIC=-1541.565

Best model: ARIMA(0,1,1)

After fitting the ARIMA(0,1,1) model, we get the plot as follows –

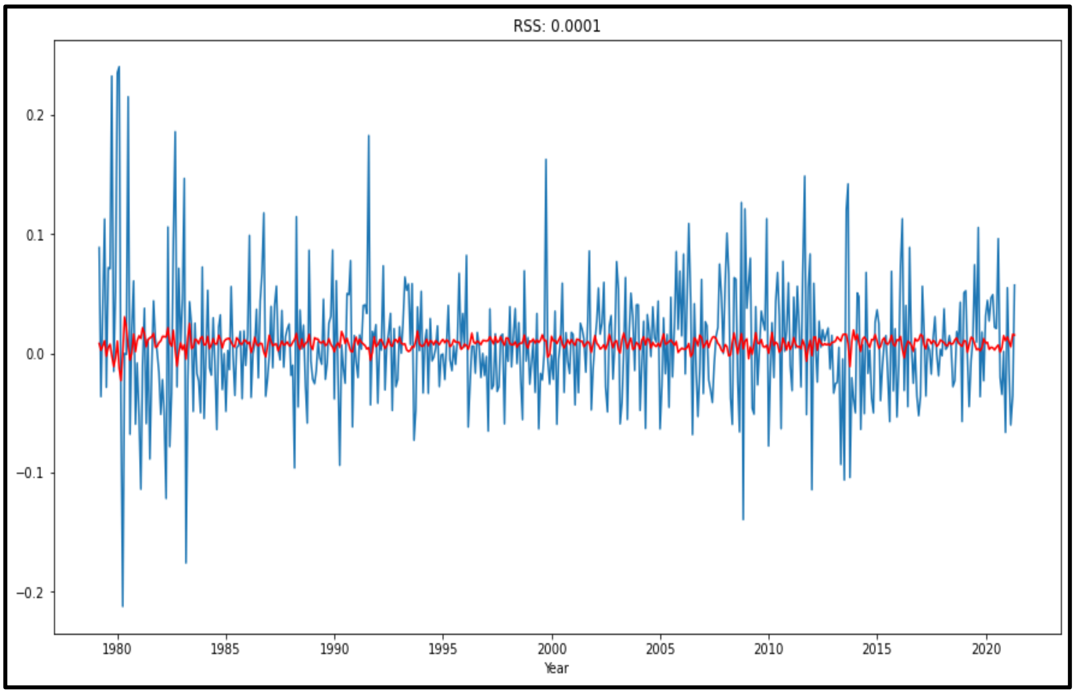

Fig. 17 ARIMA model plot

Here, Residual sum of square for the ARIMA model is 0.0001.

Model summary for ARIMA(0,1,1) is –

ARIMA Model Results

coefstd err z P>|z| [0.025 0.975]

----

const 0.0084 0.002 3.952 0.000 0.004 0.013

ma.L1.D.price -0.0795 0.047 -1.702 0.089 -0.171 0.012

Roots

==

Real Imaginary Modulus Frequency

MA.1 12.5770 +0.0000j 12.5770 0.0000

Let's define a function which can be used to change the scale of the model to the original scale. So, we can analyse that how well the model fitted.

After retransforming data, we get the following plot -

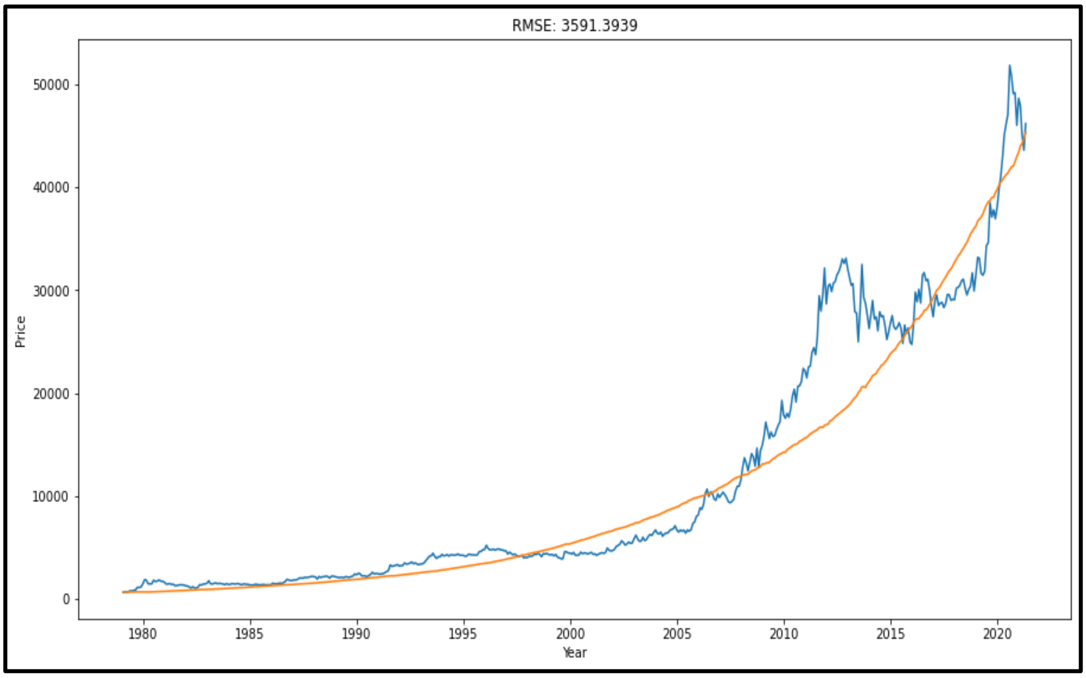

Fig. 18Plot of ARIMA model

The root mean square error for ARIMA model is 3591.5939. It's looking good fit for the gold price data.

Since, there is presence of seasonality in the time series data. We'll try to minimize the RMSE by fitting SARIMA model. Let's see how it performs

SARIMA model -

An extension to ARIMA that supports the direct modelling of the seasonal component of the series is called SARIMA. The Seasonal Autoregressive Integrated Moving Average, or SARIMA, method for time series forecasting with univariate data containing trends and seasonality.

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component.It adds three new hyperparameters to specify the autoregression (AR), differencing (I) and moving average (MA) for the seasonal component of the series, as well as an additional parameter for the period of the seasonality.

Trend Elements

There are three trend elements that require configuration.
They are the same as the ARIMA model; specifically:
p: Trend autoregression order.
d: Trend difference order.
q: Trend moving average order.

Seasonal Elements

There are four seasonal elements that are not part of ARIMA that must be configured; they are:
P: Seasonal autoregressive order.
D: Seasonal difference order.
Q: Seasonal moving average order.
m: The number of time steps for a single seasonal period.

Here, the m parameter influences the P, D, and Q parameters. For data, an m of 12 for monthly data suggests a yearly seasonal cycle.

A P=1 would make use of the first seasonally offset observation in the model.

Similarly, a D of 1 would calculate a first order seasonal difference and a Q=1 would use a first order errors in the model.

Let's find out best model of SARIMA to fit on the data.

SARIMA(0, 0, 0)x(0, 0, 0, 12)12 - AIC:-1541.3185

SARIMA(0, 0, 0)x(0, 0, 1, 12)12 - AIC:-1567.4511

SARIMA(0, 0, 0)x(1, 0, 0, 12)12 - AIC:-1570.1183

SARIMA(0, 0, 0)x(1, 0, 1, 12)12 - AIC:-1567.0707

SARIMA(0, 0, 1)x(0, 0, 0, 12)12 - AIC:-1536.6342

SARIMA(0, 0, 1)x(0, 0, 1, 12)12 - AIC:-1588.6116

SARIMA(0, 0, 1)x(1, 0, 0, 12)12 - AIC:-1574.5853

SARIMA(0, 0, 1)x(1, 0, 1, 12)12 - AIC:-1589.3916

SARIMA(1, 0, 0)x(0, 0, 0, 12)12 - AIC:-1540.2896

SARIMA(1, 0, 0)x(0, 0, 1, 12)12 - AIC:-1571.8638

SARIMA(1, 0, 0)x(1, 0, 0, 12)12 - AIC:-1570.7009

SARIMA(1, 0, 0)x(1, 0, 1, 12)12 - AIC:-1571.7506

SARIMA(1, 0, 1)x(0, 0, 0, 12)12 - AIC:-1534.6348

SARIMA(1, 0, 1)x(0, 0, 1, 12)12 - AIC:-1586.6226

SARIMA(1, 0, 1)x(1, 0, 0, 12)12 - AIC:-1573.007

SARIMA(1, 0, 1)x(1, 0, 1, 12)12 - AIC:-1587.5379

From the above output, we get best model of SARIMA with parameters (1, 0, 1)x(1, 0, 1, 12)12. Now, we use this model to see the how it fits the data.

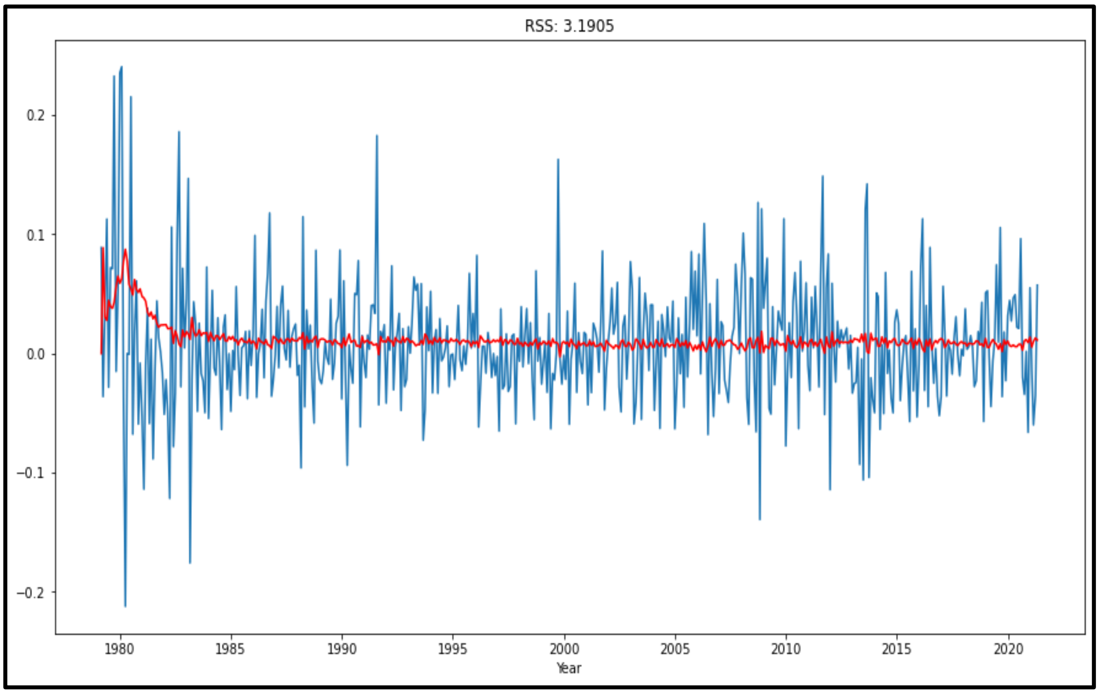

Fig 19 SARIMA model plot

Here, Residual sum of square for the SARIMA model is 3.1905.

Model summary for SARIMA(1,0,1)(1,0,1)(12) is –

SARIMAX Results

	coefstd err		z	P>|z|	[0.025	0.975]
ar.L1	-0.1480	0.576	-0.257	0.797	-1.276	0.980
ar.L2	-0.0703	0.055	-1.272	0.203	-0.179	0.038
ma.L1	0.0787	0.574	0.137	0.891	-1.047	1.204
ar.S.L12	-0.0248	0.045	-0.553	0.580	-0.113	0.063
ma.S.L12	-0.9635	0.045	-21.469	0.000	-1.051	-0.876
sigma2	0.0027	0.000	16.755	0.000	0.002	0.003

Let's define a function which can be used to change the scale of the model to the original scale. So, we can analyse that how well the model fitted.

After retransforming data, we get the following plot –

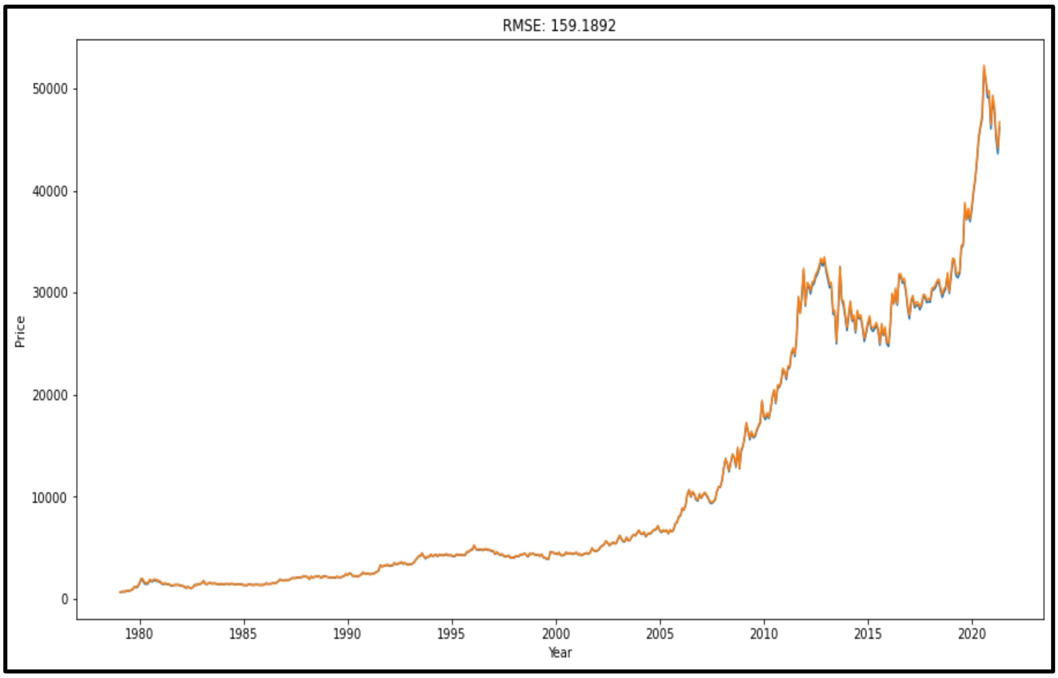

Fig. 20 Plot of SARIMA model

The root mean square error for SARIMA model is 159.1892. It's best fit for the gold price data.Now, we will use SARIMA model to forecast the prices for next 5 years. Let's see is it worth to invest in gold or not.

Month	Price
31-05-2021	46557.84
30-06-2021	46606.76
30-07-2021	47068.16
31-08-2021	48339.88
30-09-2021	49142.88
29-10-2021	48895.2
30-11-2021	49523.16
31-12-2021	49449.93
31-01-2022	50660.71
28-02-2022	51032.61
31-03-2022	50453.65
29-04-2022	50891.86

31-05-2022	51435.96
30-06-2022	51674.23
29-07-2022	52265.94
31-08-2022	53609.07
30-09-2022	54434.23
31-10-2022	54169.02
30-11-2022	54756.97
30-12-2022	54752.96
31-01-2023	56040.18
28-02-2023	56357.33
31-03-2023	55684.99
28-04-2023	56236.61
31-05-2023	56834.16
30-06-2023	57092.38
31-07-2023	57743.93
31-08-2023	59229.73
29-09-2023	60143.2
31-10-2023	59849.92
30-11-2023	60502.49
29-12-2023	60495.94
31-01-2024	61919.63
29-02-2024	62272.65
29-03-2024	61530.64
30-04-2024	62138.3
31-05-2024	62798.67
28-06-2024	63084.12
31-07-2024	63804.11
30-08-2024	65445.79
30-09-2024	66455.08
31-10-2024	66131.03

29-11-2024	66852
31-12-2024	66844.82
31-01-2025	68417.89
28-02-2025	68807.88
31-03-2025	67987.97
30-04-2025	68659.46
30-05-2025	69389.12
30-06-2025	69704.53
31-07-2025	70500.08
29-08-2025	72314.05
30-09-2025	73429.26
31-10-2025	73071.2
28-11-2025	73867.83
31-12-2025	73859.9
30-01-2026	75598.06
27-02-2026	76028.98
31-03-2026	75123.03
30-04-2026	75864.99

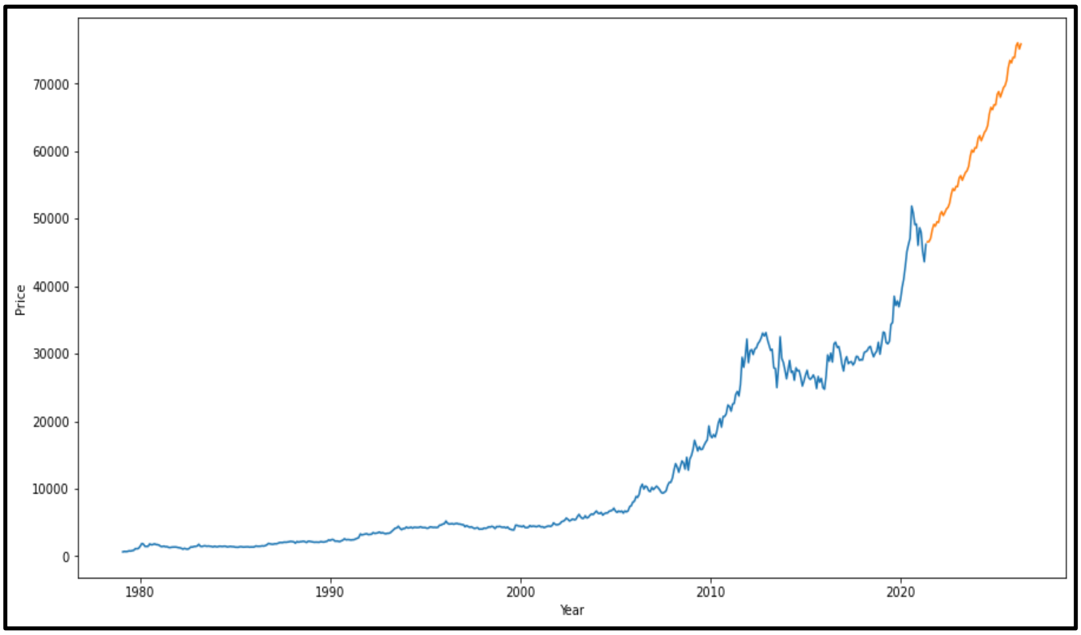

Fig. 21 plot of forecasting gold price for next 5 years

Here, the blue line shows the original gold prices till April 2021 & orange line represents the forecasted future gold prices for next 5 years (60 months) up to April 2026.

The time series plot shows there is increasing trend (uptrend) in the next 5 years. This shows that it is worth to invest in gold looking at long term prospective.

Residual diagnosis:

We will first plot the residual to see that the variance is constant or not.

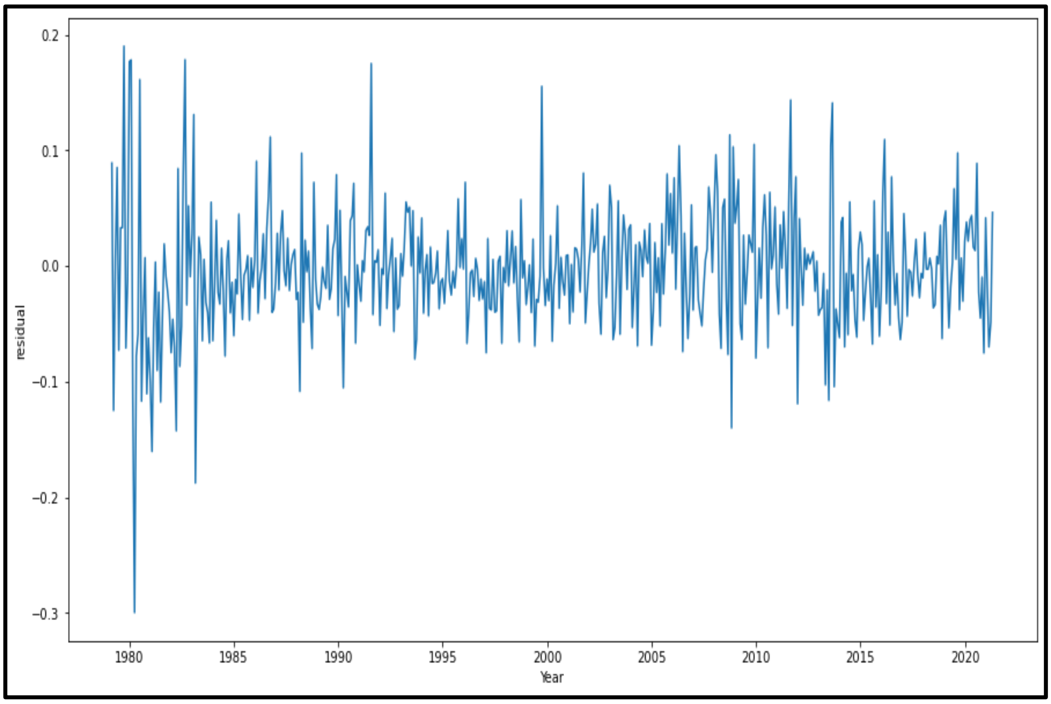

Fig.22 Residual plot

From the above residual plot, we can see that the residuals are constant over the time. This shows that the variance is constant.

Now, we plot the histogram of residuals to check the normality assumptions.

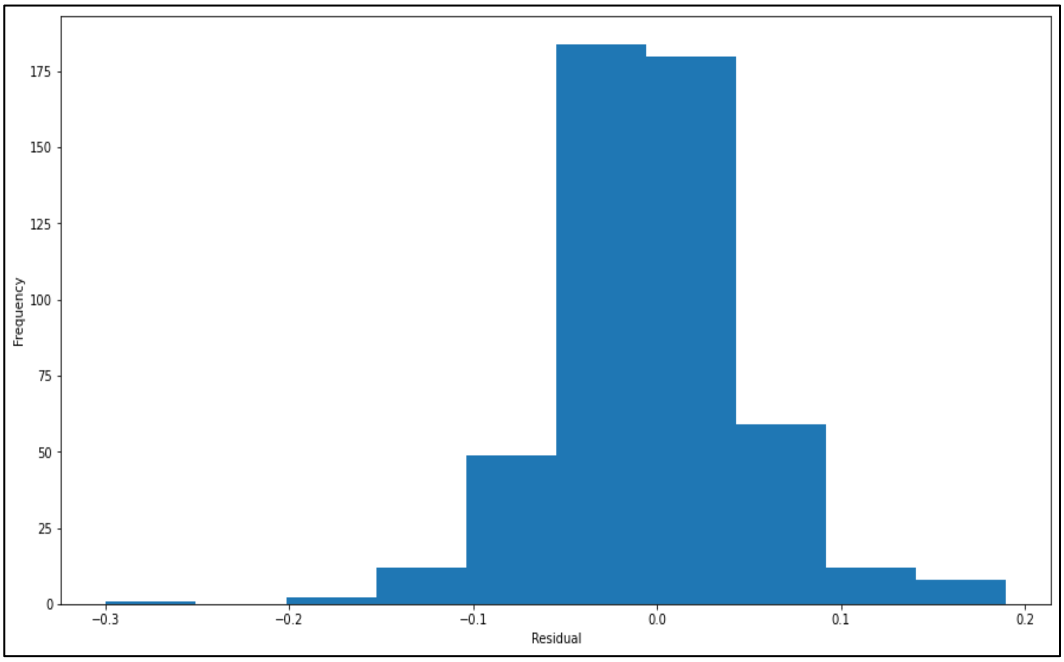

Fig.23 Histogram of residuals

The above the histogram of residuals shows that the residuals are approximately normally distributed. The mean of residual is -0.00352 which is nearly equal to zero.So, we can say that the residuals are normally distributed with mean 0 and constant variance σ_e^2.

Factors impacting gold prices -

Now, let's find out the impact of different factors on Gold prices. First, we will see the how **Inflation rates** affect the gold prices.

Gold has always been considered a good hedge against inflation. Rising inflation rates typically appreciates gold prices. Traditional theory implies that the relative price of consumer goods and of such real assets as land and gold should not be permanently affected by the rate of inflation. A change in the general rate of inflation should, in equilibrium, cause an equal change in the rate of inflation for each asset price While calculating the price of gold there are two inflation rates. One is Gold internal inflation rate, which is change in its production from its mines. Other is monetary inflation. The price of gold over the medium to long term is determined by its inflation rate relative to that of the currency you want to measure it with. With most fiat currency inflation rates, running substantially higher than gold's inflation rate it is easy to see why the gold price will continue to increase over time, and why it has consistently increased over time. This is not about to change regardless of short-term volatility.

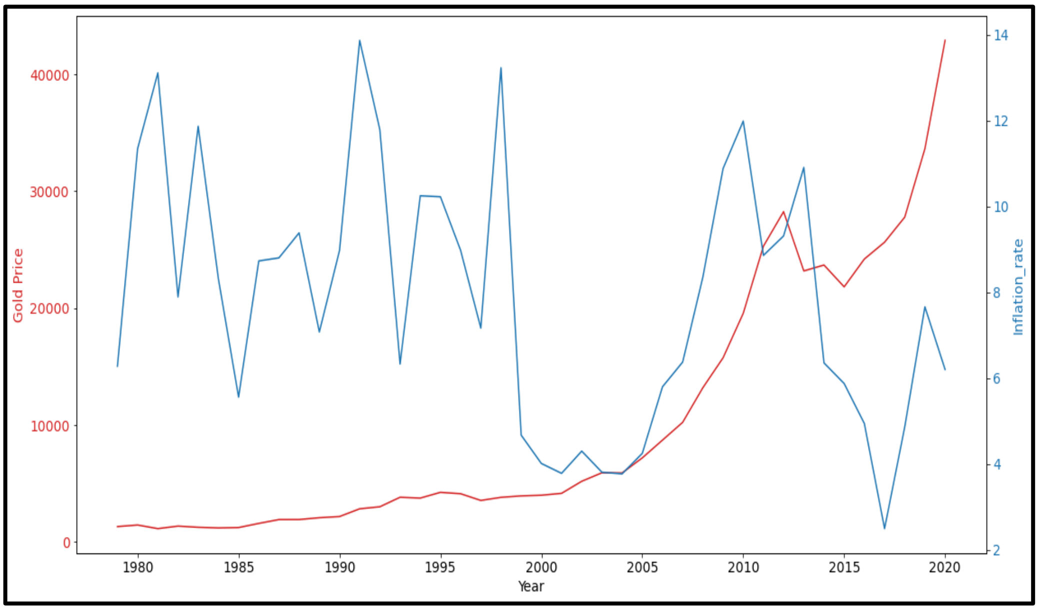

Fig. 24 Trend analysis of Inflation rates and gold prices

The scatter plot for the inflation rate vs gold prices is as follows –

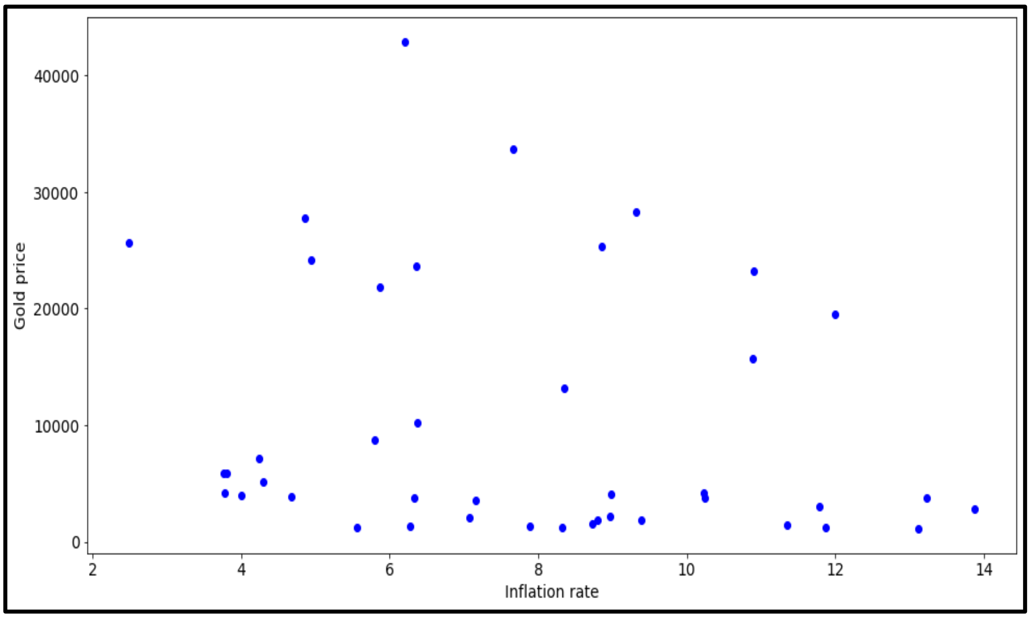

Fig. 25 Scatter plot (Inflation rate vs Gold price)

The scatter plot shows that there is somewhat linear relationship between inflation rate and gold prices. It seems that the increase inflation rate, there is decrease in gold prices. So, there is negative correlation between inflation rate and gold prices.

The correlation coefficient between inflation rate and gold prices is -0.1844. This indicates that there is low negative correlation between them.

For checking the significance of correlation coefficient, we use the Pearson's correlation coefficient.

Our hypothesis are as follows:

H_o: There is no correlation between inflation rate and gold prices, i.e. r = 0. VS

H_a: There is significant correlation between inflation rate and gold prices, i.e. $r \neq 0$.

From the python, we get the output as follows –

Correlation coefficient: -0.2592 p-value: 0.2424

We know that, if p-value is less than α, then we reject H_o.

Here, p-value is 0.2424 which greater than 0.05. Hence, we do not reject H_o. This implies that there is no significant correlation between inflation rate and gold prices.

Now, we will see the impact of **USD-INR** prices on gold prices.

It is an important question that is there any correlation between gold prices and the value of US DOLLAR. Now the answer depends upon situation and changes with change in global economic scenario. Nowthere is an inverse relationship between gold prices and US Dollar. Before 1950 US $ was also considered as the inflation hedge. But this is not true now. So in the past it can observe the positive correlation between gold prices and US $. As a tool of hedge now gold is demanded more than the US $. When the price of gold depreciates the investors outside US will benefited because the dollar price of the gold will increase. Investor can shift away from the dollar denominated assets to gold. Past experiences also that gold has been used as a hedge against currency risk.

Fig. 26 Trend analysis of USD INR and gold prices

The scatter plot for the USD INR vs gold prices is as follows –

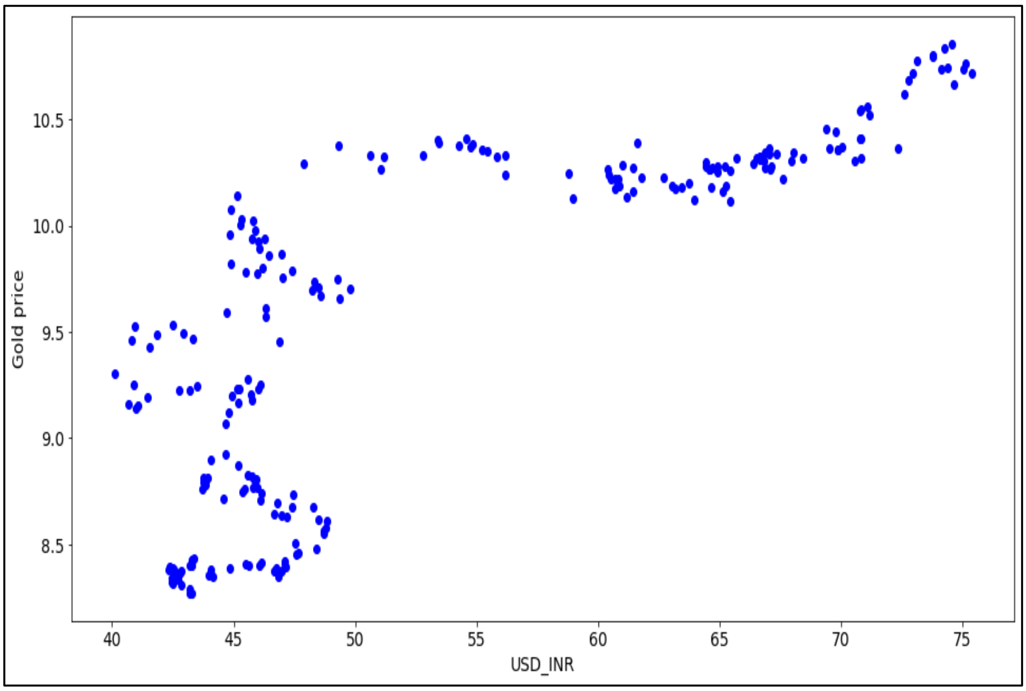

Fig. 27 Scatter plot USD INR vs Gold price)

The scatter plot shows that there is positive linear relationship between USD INR and gold prices. It seems that the increase USD INR, there is increase in gold prices. So, there is positive correlation between USD INR and gold prices.

The correlation coefficient between USD INR and gold prices is **0.8691.** This indicates that there is strong positive correlation between them.

For checking the significance of correlation coefficient, we use the Pearson's correlation coefficient.

Our hypothesis are as follows:

H_o: There is no correlation between USD INR and gold prices, i.e. r = 0. VS

H_a: There is significant correlation between USD INR and gold prices, i.e. $r \neq 0$.

From the python, we get the output as follows –

Correlation coefficient: 0.8691 p-value: 5.87×10^{-81}

We know that, if p-value is less than α, then we reject H_o.

Here, p-value is 5.87×10^{-81} which less than 0.05. Hence, we reject H_o. This implies that there is significant correlation between USD INR and gold prices.

Let's find out the relationship between Nifty 50 and gold prices.

The scatter plot for the Nifty price vs gold prices is as follows –

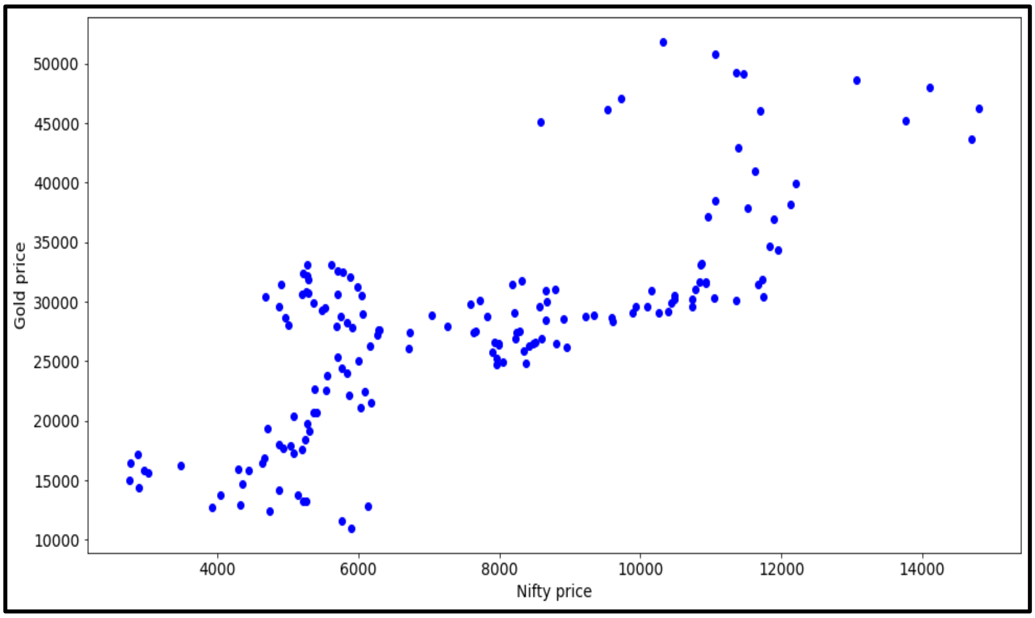

Fig. 29 Scatter plot for Nifty price vs gold price

The scatter plot shows that there is positive linear relationship between Nifty price and gold prices. It seems that the increase Nifty, there is increase in gold prices. So, there is positive correlation between Nifty and gold prices.

The correlation coefficient between Nifty and gold prices is 0.7465. This indicates that there is strong positive correlation between them.For checking the significance of correlation coefficient, we use the Pearson's correlation coefficient.

Our hypothesis are as follows:

H_o: There is no correlation between USD INR and gold prices, i.e. r = 0. VS

H_a: There is significant correlation between USD INR and gold prices, i.e. $r \neq 0$.

From the python, we get the output as follows –

Correlation coefficient: 0.7465 p-value: 4.12×10^{-30}

We know that, if p-value is less than α, then we reject H_o.

Here, p-value is 4.12×10^{-30} which less than 0.05. Hence, we reject H_o. This implies that there is significant correlation between Nifty and gold prices.

MAJOR FINDINGS

1. From the gold price plot, we can see that there is no sudden increase in gold price from 1979 till 2005. The prices are increasing slowly. But, after 2005 to 2012, there was boom in the gold prices due to nifty fall, excess liquidity, exchange accept gold as collateral and some more facts. Again, the gold price consolidates in the range of 20000 to 30000 till 2020. After that the gold price again breakout to next high of 56900 in just a year.
2. We find out the best model for forecasting of gold prices which is SARIMA (1,0,1)12.We had forecasted gold prices for next 5 years.

- **Factors impacting gold prices -**

1. The scatter plot shows that there is somewhat linear relationship between inflation rate and gold prices. It seems that the increase inflation rate, there is decrease in gold prices. So, there is negative correlation between inflation rate and gold prices. The correlation coefficient between inflation rate and gold prices is -0.1844. This indicates that there is low negative correlation between them.
2. The scatter plot shows that there is positive linear relationship between USD INR and gold prices. It seems that the increase USD INR, there is increase in gold prices. So, there is positive correlation between USD INR and gold prices. The correlation coefficient between USD INR and gold prices is **0.8691.** This indicates that there is strong positive correlation between them.
3. The scatter plot shows that there is positive linear relationship between Nifty price and gold prices. It seems that the increase Nifty, there is increase in gold prices. So, there is positive correlation between Nifty and gold prices. There is strong correlation between nifty price and gold price, but if we see the trend the nifty is inversely proportional to gold price.

REFERENCES

- https://en.wikipedia.org/wiki/Gold_as_an_investment
- https://www.machinelearningplus.com/time-series/time-series-analysis-python/
- Times series analysis – James D. Hamilton
- https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4.htm
- www.gold.org
- Basic statistics – B L Agarwal
- A study on impact of select factors on the price of Gold – Dr Sindhu

APPENDIX

Python code -

```
# importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima_model import ARMA
# Loading data
data = pd.read_csv("C:/Users/ASUS/Desktop/Monthly Gold data.csv",
index_col=[0],parse_dates=[0])
# droppingna values
data1 = data.dropna()
# shape of data
data1.shape
# column names
data1.columns
# changing column name
data1.columns = ['price']
# first 5 rows of data
data1.head()
# Descriptive statistics of data
data1.describe()
# plotting the graph of data
plt.xlabel("Date")
plt.ylabel("price")
plt.plot(data1)
# splitting of dataset
train=data1[:410]
valid=data1[410:]
# plotting the time series train test data set
train.price.plot(figsize=(15,8), title = 'Monthly gold price', fontsize= 14, label='Train',
color = 'orange')
valid.price.plot(figsize=(15,8), title = 'Monthly gold price', fontsize=14, label = 'valid',
color='black')
plt.xlabel('Month')
plt.ylabel('Monthly price')
plt.legend(loc='best')
plt.show()
# naive forecast
```

```
dd = np.asanyarray(train.price)
y_hat = valid.copy()
y_hat['naive'] = dd[len(dd)-1]
plt.figure(figsize=(12,8))
plt.plot(train.index, train['price'], label='Train', color = "orange")
plt.plot(valid.index, valid['price'], label = 'valid', color = "black")
plt.plot(y_hat.index, y_hat['naive'], label = 'Naive Forecast')
plt.legend(loc='best')
plt.title("Naive Forecast")
plt.xlabel("Month")
plt.ylabel("Gold Price")
plt.show()
# Checking RMSE
from sklearn.metrics import mean_squared_error
from math import sqrt
rms = sqrt(mean_squared_error(valid.price, y_hat.naive))
print("Residual sum of square is:",rms)
# moving average forecast
y_hat_avg = valid.copy()
y_hat_avg['moving_avg_forecast'] = train['price'].rolling(12).mean().iloc[-1] # average of
last 12 observations
plt.figure(figsize=(15,8))
plt.plot(train['price'], label='Train', color = "orange")
plt.plot(valid['price'], label='Valid', color = "black")
plt.plot(y_hat_avg['moving_avg_forecast'], label='Moving Average Forecast using 12
observations')
plt.title("Moving average Forecast")
plt.xlabel("Month")
plt.ylabel("Gold Price")
plt.legend(loc='best')
plt.show()
# checking RMSE
from sklearn.metrics import mean_squared_error
from math import sqrt
rms = sqrt(mean_squared_error(valid.price, y_hat_avg.moving_avg_forecast))
print("Residual mean sum of square is:",rms)
# Simple exponential smoothing
from statsmodels.tsa.holtwinters import ExponentialSmoothing
hwmodel=ExponentialSmoothing(train.price,trend='mul', seasonal='mul',
seasonal_periods=4).fit()
test_pred=hwmodel.forecast(88)
test_pred
# plotting the predcited
train['price'].plot(legend=True, label='Train', figsize=(12,8), color = "orange")
```

```
valid['price'].plot(legend=True, label='Valid', color = "black")
test_pred.plot(legend=True, label='predicted_test')
plt.title("")
plt.xlabel("Month")
plt.ylabel("Gold Price")
plt.show()
# checking MSE
from sklearn.metrics import mean_squared_error
np.sqrt(mean_squared_error(test_pred, valid['price']))
# Double exponential smoothing
from statsmodels.tsa.api import Holt
y_hat_avg = valid.copy()
fit1 = Holt(np.asarray(train['price'])).fit(smoothing_level = 0.55, smoothing_slope = 0.15)
y_hat_avg['Holt_linear'] = fit1.forecast(len(valid))
plt.figure(figsize=(16,8))
plt.plot(train['price'], label='Train', color = "orange")
plt.plot(valid['price'], label='Valid', color = "black")
plt.plot(y_hat_avg['Holt_linear'], label='Holt_linear')
plt.title("Holt Linear model")
plt.xlabel("Month")
plt.ylabel("Gold Price")
plt.legend(loc='best')
plt.show()
# Defining stationarity function
from statsmodels.tsa.stattools import adfuller
def test_stationarity(timeseries):

   #determining rolling statistics
rolmean = timeseries.rolling(window=12).mean()
rolstd = timeseries.rolling(window=12).std()

   #plot rolling statistics
plt.figure(figsize=(15,8))
orig = plt.plot(timeseries,color = 'red',label='Original')
   mean = plt.plot(rolmean, color = 'blue', label = 'Rolling mean')
   std = plt.plot(rolstd, color = 'green', label = 'Rolling sd')
plt.legend(loc = 'best')
plt.title('Rolling mean & Standard deviation')
plt.show(block=False)

   #Perform Dickey-Fuller test
   from statsmodels.tsa.stattools import adfuller

print('Results of Dickey-Fuller test')
```

```
dftest = adfuller(timeseries,autolag='AIC')

dfoutput = pd.Series(dftest[0:4], index=['Test_statistic','p-value','#Lags_used','Number_of_observations_used'])
   for key,value in dftest[4].items():
dfoutput['Critical Value (%s)'%key] = value
   print(dfoutput)
# testing stationarity of data
test_stationarity(data1)

# log transformation

data_log = np.log(data1)

# plotting transformed data
plt.figure(figsize=(15,8))
plt.plot(data_log)
plt.xlabel("Year")
plt.ylabel("log price")
# Checking stationarity
from statsmodels.tsa.seasonal import seasonal_decompose
decomposition = seasonal_decompose(data_log, period=12)

trend = decomposition.trend
seasonal = decomposition.seasonal
residual = decomposition.resid

plt.figure(figsize=(15,8))
plt.subplot(411)
plt.plot(data_log, label = "Original")
plt.legend(loc = "best")
plt.subplot(412)
plt.plot(trend, label="trend")
plt.legend(loc='best')
plt.subplot(413)
plt.plot(seasonal,label='Seasonality')
plt.legend(loc='best')
plt.subplot(414)
plt.plot(residual, label='Residual')
plt.legend(loc='best')
plt.tight_layout()
# Differencing
DiffTS = data_log - data_log.shift()
plt.figure(figsize=(15,8))
plt.plot(DiffTS)
```

```
# Best ARIMA
p_values=range(0,2)
q_values=range(0,2)
d_values=range(0,1)
import pmdarima as pm
stepwise_fit = pm.auto_arima(data_log,
start_p=0,d=1,start_q=0,
max_p=5,max_q=7,
                    trace=True,suppress_warnings=True)

# Fitting best ARIMA model
from statsmodels.tsa.arima_model import ARIMA
#ARIMA model
model = ARIMA(data_log, order = (2,1,1))
results_ARIMA = model.fit(disp=-1)
plt.figure(figsize=(15,8))
plt.plot(DiffTS)
plt.xlabel("Year")
plt.plot(results_ARIMA.fittedvalues, color = 'red')
plt.title('RSS: %.4f'% sum(results_ARIMA.fittedvalues-DiffTS['price'])**2)
print("plotting ARIMA model")

# Retransforming
ARIMA_diff_predictions = pd.Series(results_ARIMA.fittedvalues, copy=True)
print(ARIMA_diff_predictions)
predictions_ARIMA_diff_cumsum = ARIMA_diff_predictions.cumsum().shift().fillna(0)
print (predictions_ARIMA_diff_cumsum)
predictions_ARIMA_log = pd.Series(np.ones(data1.shape[0])*np.log(data1['price'])[0],
index=data1.index)
predictions_ARIMA_log =
predictions_ARIMA_log.add(predictions_ARIMA_diff_cumsum,fill_value=0)
print(predictions_ARIMA_log)
predictions_ARIMA = np.exp(predictions_ARIMA_log)
# plotting forecasted values
plt.figure(figsize=(15,8))
plt.plot(data1)
plt.plot(predictions_ARIMA)
plt.xlabel("Year")
plt.ylabel("Price")
plt.title('RMSE: %.4f'% np.sqrt(sum((predictions_ARIMA-
data1['price'])**2)/len(data1)))
# ignore warnings
import itertools
import warnings # To ignore warnings
```

```
warnings.filterwarnings("ignore")
# importing library for SARIMA model
import statsmodels.api as sm
# Fitting best SARIMA model
p = range(0, 2)
d = range(1)
q = range(0, 2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d, q))]
print('Examples of parameter combinations for Seasonal ARIMA...')
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[1]))
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[2]))

for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(DiffTS.price,
                                            order=param,
seasonal_order=param_seasonal,
enforce_stationarity=False,
enforce_invertibility=False)
            results = mod.fit()
            print('SARIMA{}x{}12 - AIC:{}'.format(param, param_seasonal, results.aic))
        except:
            continue
# fitting SARIMA model
mod = sm.tsa.statespace.SARIMAX(log_data, trend='n', order=(1,1,1), seasonal_order=(0,0,0,12))
results_SARIMA = mod.fit()
plt.figure(figsize=(15,8))
plt.plot(log_data)
plt.xlabel("Year")
plt.plot(results_SARIMA.fittedvalues, color = 'red')
plt.title('RSS: %.4f'% sum(results_SARIMA.fittedvalues-DiffTS['price'])**2)
print("plotting SARIMA model")
# forecasting values
x1=results_ARIMA.forecast(steps=60)
predictions_ARIMA = np.exp(x)
#plotting forecasted values
plt.figure(figsize=(15,8))
plt.plot(data1['price'])
plt.plot(predictions_SARIMA)
plt.xlabel("Year")
plt.ylabel("Price")
```

****