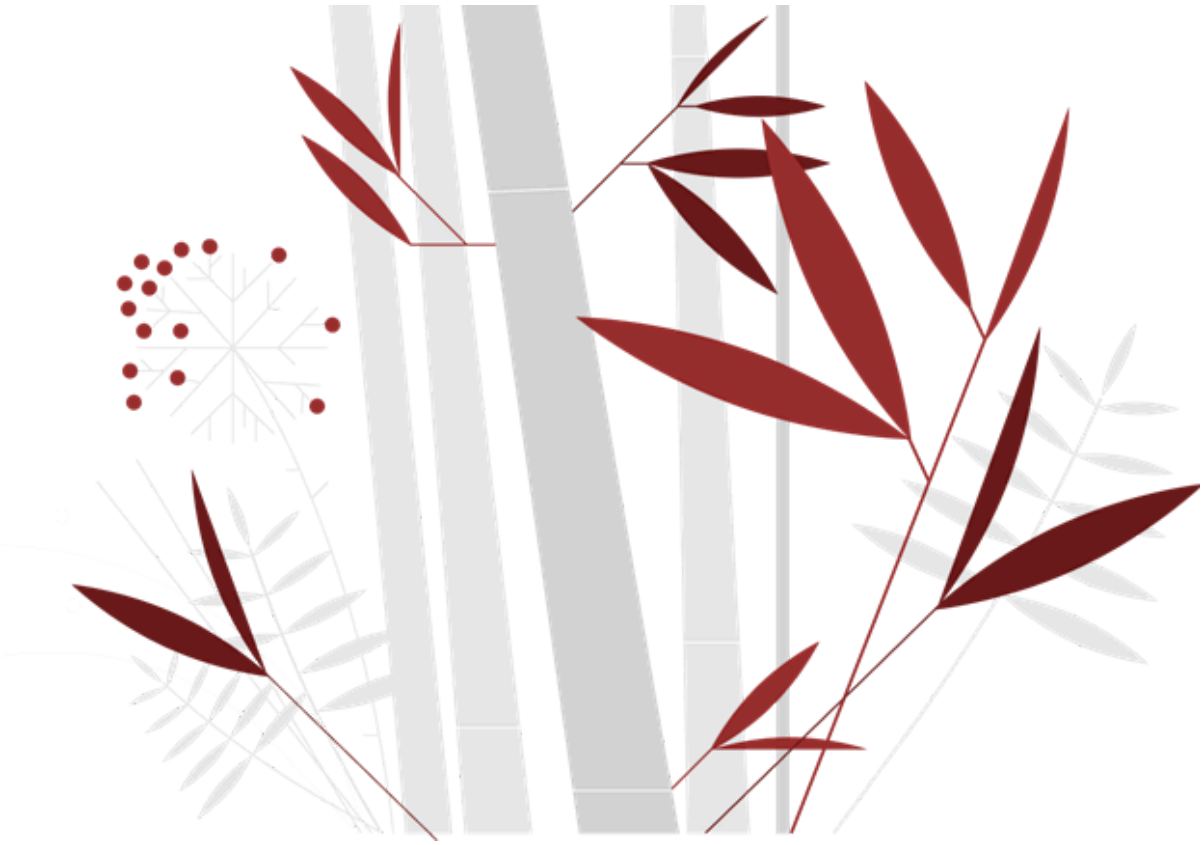
TIME SERIES FORECASTING (TSF) PROJECT-CODED

BY

Harsh Patel

15th September 2024



Problem-: ABC ESTATE WINES

As an analyst at ABC Estate Wines, we are presented with historical data encompassing the sales of different types of wines throughout the 20th century. These datasets originate from the same company but represent sales figures for distinct wine varieties. Our objective is to delve into the data, analyse trends, patterns, and factors influencing wine sales over the course of the century. By leveraging data analytics and forecasting techniques, we aim to gain actionable insights that can inform strategic decision-making and optimise sales strategies for the future.

Objective

The primary objective of this project is to analyse and forecast wine sales trends for the 20th century based on historical data provided by ABC Estate Wines. We aim to equip ABC Estate Wines with the necessary insights and foresight to enhance sales performance, capitalise on emerging market opportunities, and maintain a competitive edge in the wine industry.

[Note: Before we start any EDA and data Preprocessing we will load the necessary library files and load both the datasets named “Rose” and “Sparkling” using read_csv command.]

2. Exploratory Data Analysis and Data Pre-Processing for Rose Dataset:

We will build a Forecasting model for Rose flavour wine first and then for Sparkling flavour wine later. From the below Table-1 we can see the first and last five rows for the Rose dataset.

	YearMonth	Rose		YearMonth	Rose
0	1980-01	112.0	**182**	1995-03	45.0
1	1980-02	118.0	**183**	1995-04	52.0
2	1980-03	129.0	**184**	1995-05	28.0
3	1980-04	99.0	**185**	1995-06	40.0
4	1980-05	116.0	**186**	1995-07	62.0

Table-1 FIrst and last Five rows of Rose dataset.

Now, we will use the info function to get the number of variables and their data types.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 187 entries, 0 to 186
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   YearMonth  187 non-null    object 
 1   Rose       185 non-null    float64
dtypes: float64(1), object(1)
memory usage: 3.1+ KB
```

Table-2 Information of Rose dataset.

For Creating Time Series Data set we can do it in two ways:

Method-1: we will create a TImestamp and adding it to the data frame to convert it into a time series dataset and add that timestamp to the original data frame along with setting it as an index and then dropping the YearMonth column and we get the our Time series rose dataset as shown in Table-3 and 4.

```
DatetimeIndex(['1980-01-31', '1980-02-29', '1980-03-31', '1980-04-30',
               '1980-05-31', '1980-06-30', '1980-07-31', '1980-08-31',
               '1980-09-30', '1980-10-31',
               ...
               '1994-10-31', '1994-11-30', '1994-12-31', '1995-01-31',
               '1995-02-28', '1995-03-31', '1995-04-30', '1995-05-31',
               '1995-06-30', '1995-07-31'],
              dtype='datetime64[ns]', length=187, freq='ME')
```

Table-3 Timestamp creation for Rose dataset

	Rose
Time_stamp	
1980-01-31	112.0
1980-02-29	118.0
1980-03-31	129.0
1980-04-30	99.0
1980-05-31	116.0

Table-4 Method-1 Time series Rose dataset

Method-2: We read the original dataset as a Time series dataset by using the pandas function setting parse_date and squeeze as True as well as setting index_col as zero. And we get the first and last five rows of our new time series dataset for rose wine as shown in table-5.

First 5 rows for Rose Wine :

```
YearMonth
1980-01-01    112.0
1980-02-01    118.0
1980-03-01    129.0
1980-04-01     99.0
1980-05-01    116.0
Name: Rose, dtype: float64
```

Last 5 rows for Rose Wine :

```
YearMonth
1995-03-01    45.0
1995-04-01    52.0
1995-05-01    28.0
1995-06-01    40.0
1995-07-01    62.0
Name: Rose, dtype: float64
```

Table-5 Method-2 Time series Rose dataset

After that we found that there are two null/empty values in the rose dataset using is null function. And we found that there are two null/empty values and can be seen in fig-1. For these missing values we impute them using interpolation for polynomial order 2 and get the fig-2 and conclude that we have imputed the missing values.

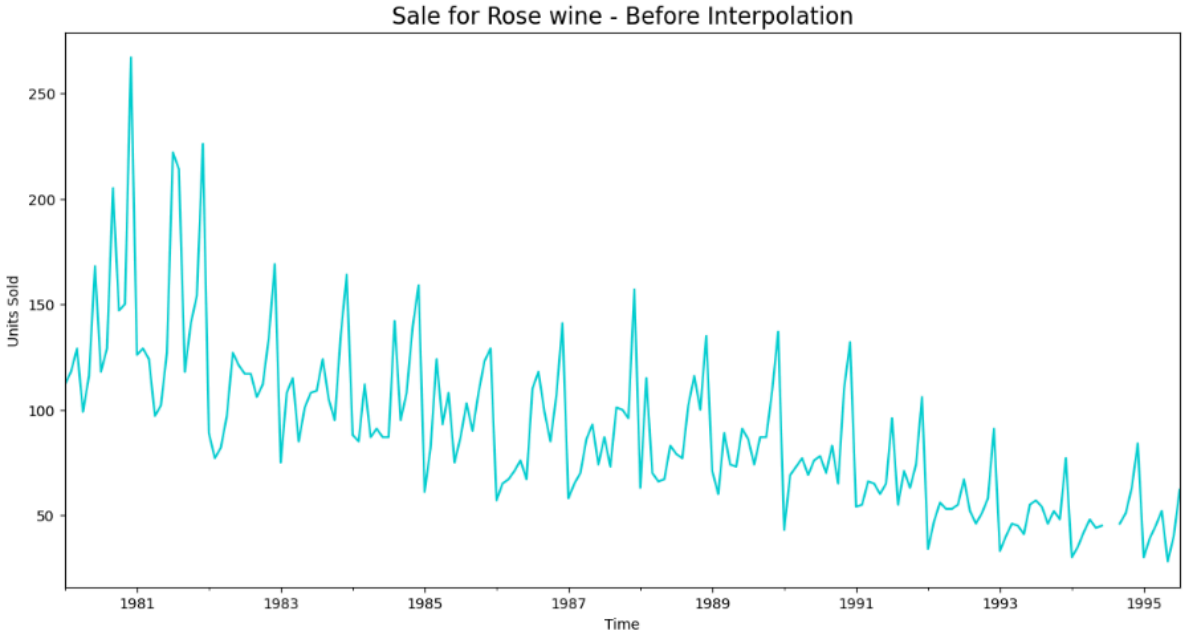


Fig-1 Before Imputation Rose dataset

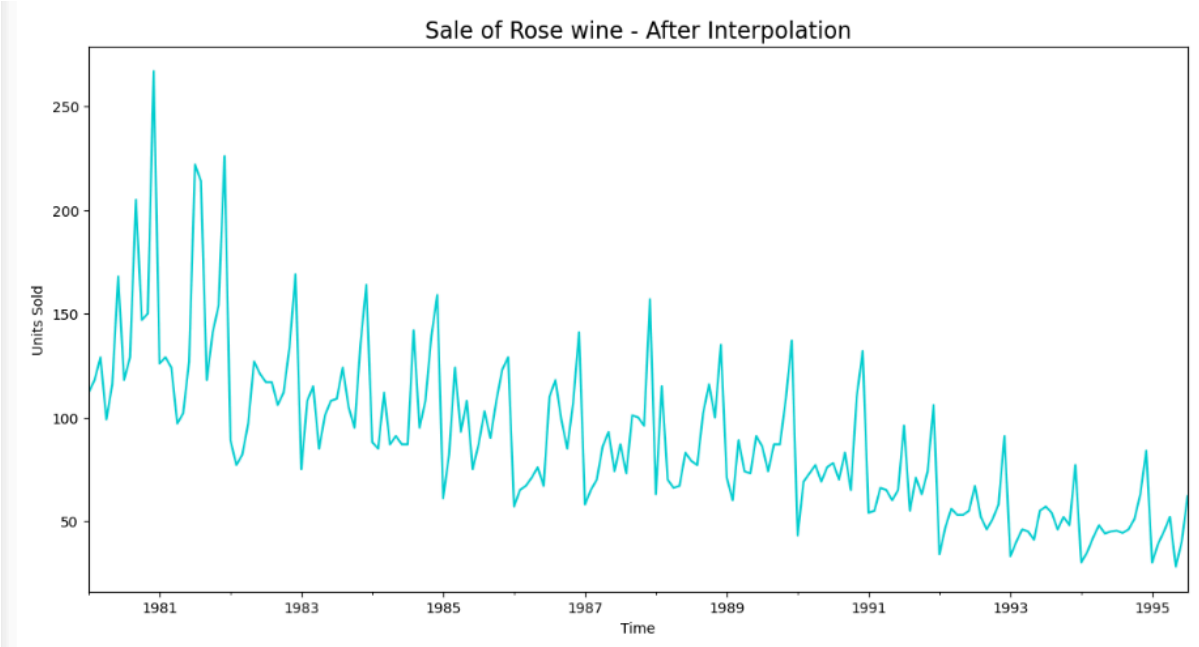


Fig-2 After Imputation Rose dataset

Now, we use the describe function on the rose data set and obtain the five required summary values like min, max, std, variance at 25,50 and 75 percent as well as count as shown in Table-6.

```
count    187.000000
mean      89.907184
std       39.246679
min       28.000000
25%       62.500000
50%       85.000000
75%      111.000000
max      267.000000
Name: Rose, dtype: float64
```

Table-6 Description of Rose dataset

Now we plot the boxplot for yearly sales of rose wine as seen in Fig-3.

Yearly Boxplot for Rose Dataset:

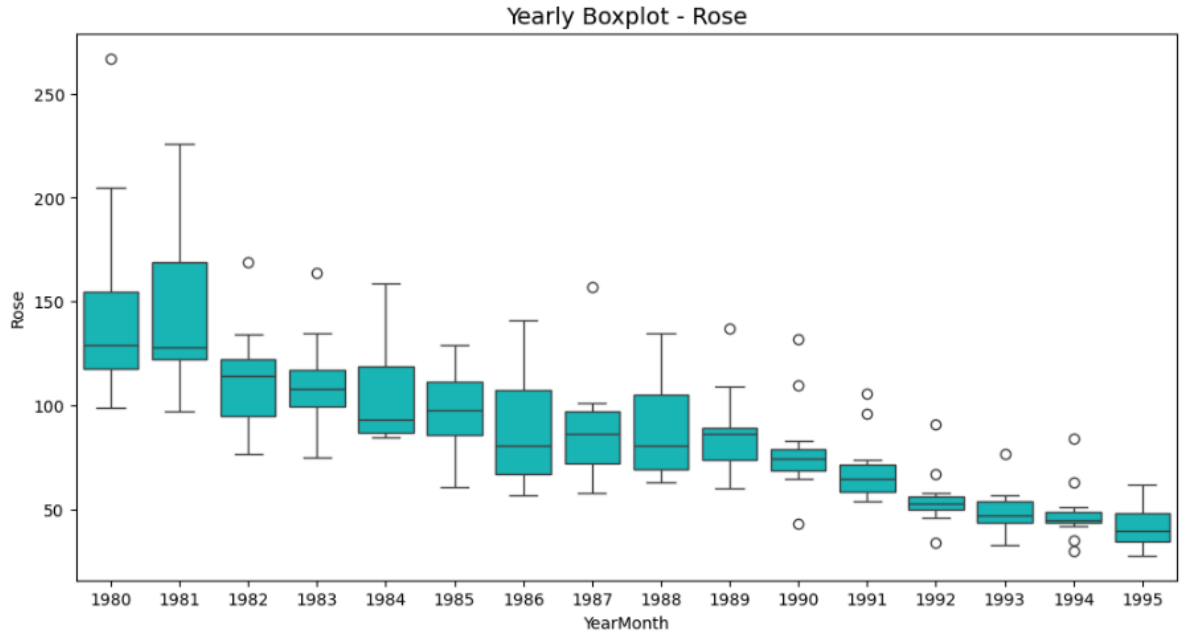


Fig-3 Boxplot Yearly sales for Rose dataset

We will also look at the boxplot of monthly sales of rose wine over the years combined as seen in Fig-4.

Monthly Boxplot for all the years for Rose Dataset:

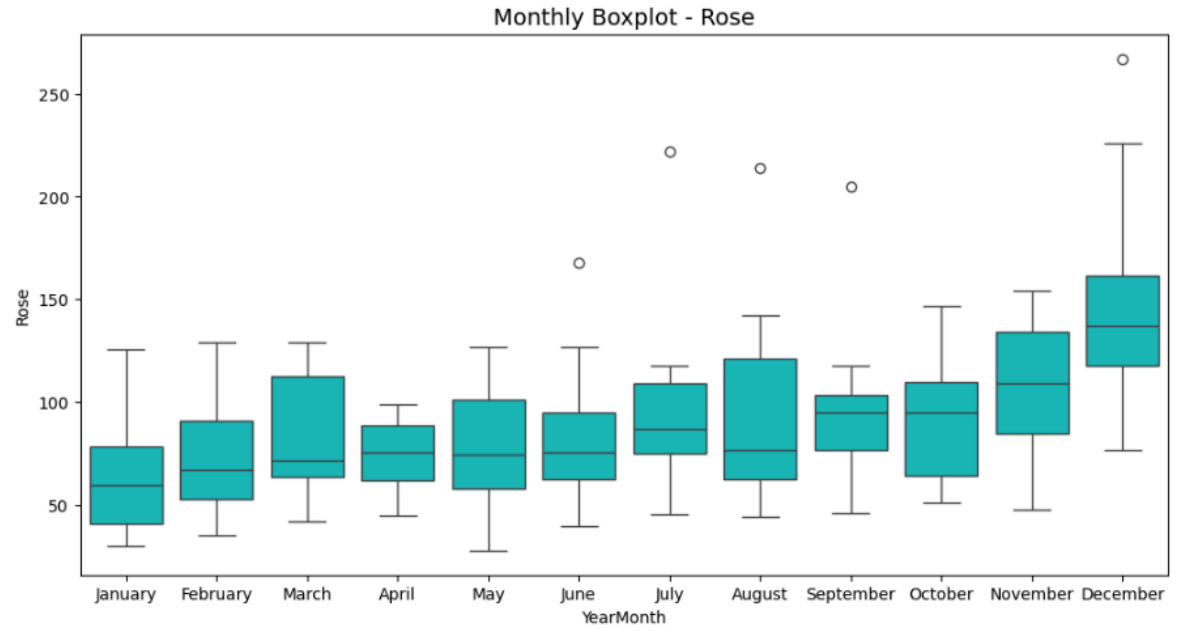


Fig-4 Boxplot Monthly sales for Rose dataset

As seen in below Fig-5 we can see the visualisation of mean and variance of monthly sales for all the years present in the dataset.

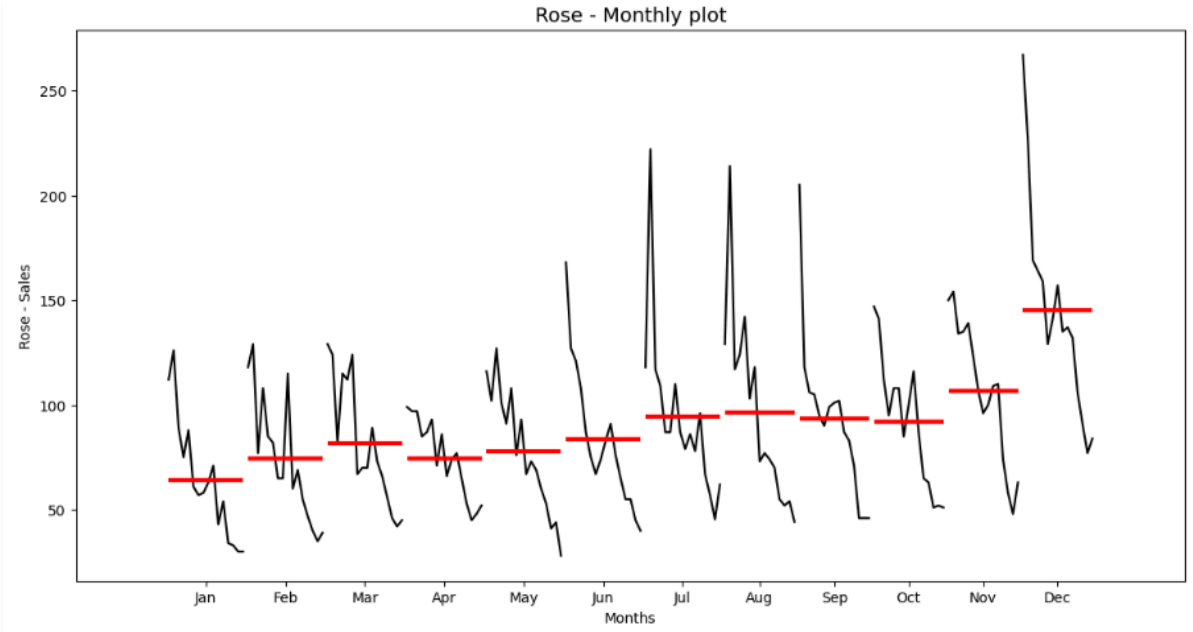


Fig-5 Mean and STD for Monthly sales of Rose dataset

Using pivot table function we can see the sales of rose wine in each month for each year as observed in the below Table-7 and its line chart can be looked at Fig-6.

												Rose
YearMonth	1	2	3	4	5	6	7	8	9	10	11	12
YearMonth												
1980	112.0	118.0	129.0	99.0	116.0	168.0	118.000000	129.000000	205.0	147.0	150.0	267.0
1981	126.0	129.0	124.0	97.0	102.0	127.0	222.000000	214.000000	118.0	141.0	154.0	226.0
1982	89.0	77.0	82.0	97.0	127.0	121.0	117.000000	117.000000	106.0	112.0	134.0	169.0
1983	75.0	108.0	115.0	85.0	101.0	108.0	109.000000	124.000000	105.0	95.0	135.0	164.0
1984	88.0	85.0	112.0	87.0	91.0	87.0	87.000000	142.000000	95.0	108.0	139.0	159.0
1985	61.0	82.0	124.0	93.0	108.0	75.0	87.000000	103.000000	90.0	108.0	123.0	129.0
1986	57.0	65.0	67.0	71.0	76.0	67.0	110.000000	118.000000	99.0	85.0	107.0	141.0
1987	58.0	65.0	70.0	86.0	93.0	74.0	87.000000	73.000000	101.0	100.0	96.0	157.0
1988	63.0	115.0	70.0	66.0	67.0	83.0	79.000000	77.000000	102.0	116.0	100.0	135.0
1989	71.0	60.0	89.0	74.0	73.0	91.0	86.000000	74.000000	87.0	87.0	109.0	137.0
1990	43.0	69.0	73.0	77.0	69.0	76.0	78.000000	70.000000	83.0	65.0	110.0	132.0
1991	54.0	55.0	66.0	65.0	60.0	65.0	96.000000	55.000000	71.0	63.0	74.0	106.0
1992	34.0	47.0	56.0	53.0	53.0	55.0	67.000000	52.000000	46.0	51.0	58.0	91.0
1993	33.0	40.0	46.0	45.0	41.0	55.0	57.000000	54.000000	46.0	52.0	48.0	77.0
1994	30.0	35.0	42.0	48.0	44.0	45.0	45.364189	44.279246	46.0	51.0	63.0	84.0
1995	30.0	39.0	45.0	52.0	28.0	40.0	62.000000	NaN	NaN	NaN	NaN	NaN

Table-7 Pivot table for Rose wine

Monthly Wine sales across years for Rose:

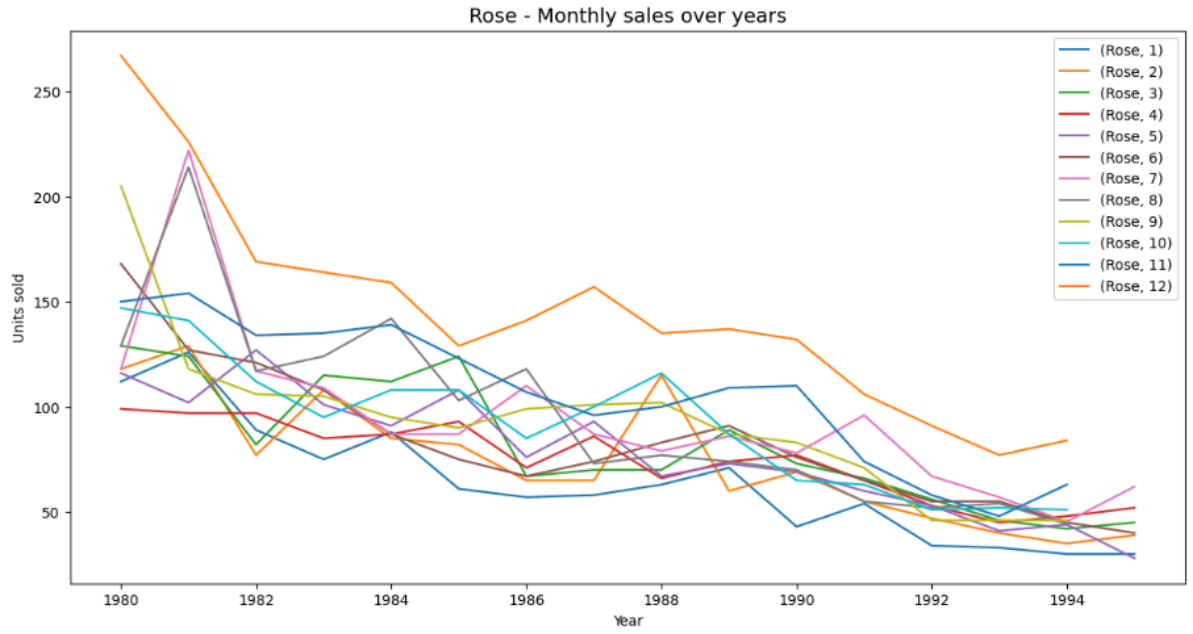


Fig-6 Line plot Sales for Rose Wine

Average rose sales and sales in percentage can be seen below FIg-7.

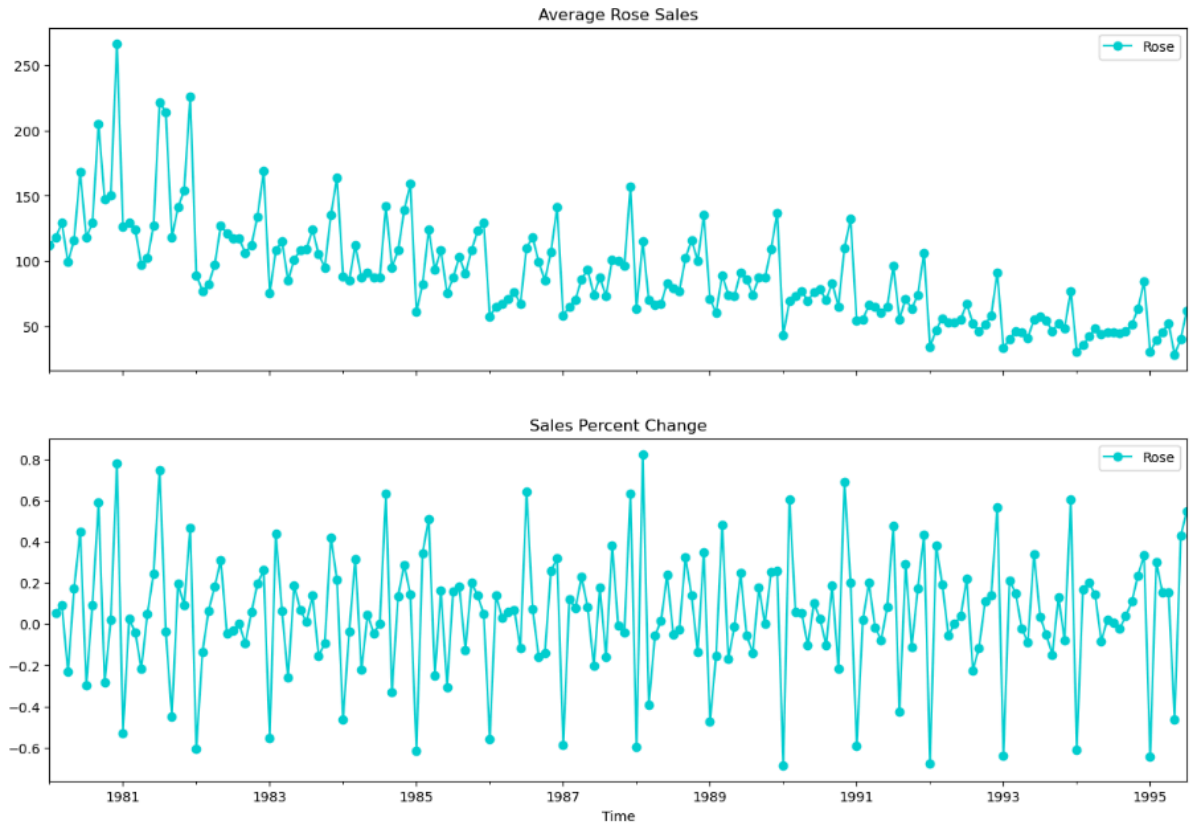


Fig-7 Line plot Average Sales for Rose Wine

Now, we will decompose the Rose time series into trends, seasonality and residuals using multiplicative decomposition model as we have observed from the above tables and figures that the dataset contains trend, seasonality and its decreasing in multiplicative manner and the decomposed model can be observed in Fig-8.

Decomposition of Rose Time Series with multiplicative Seasonality:

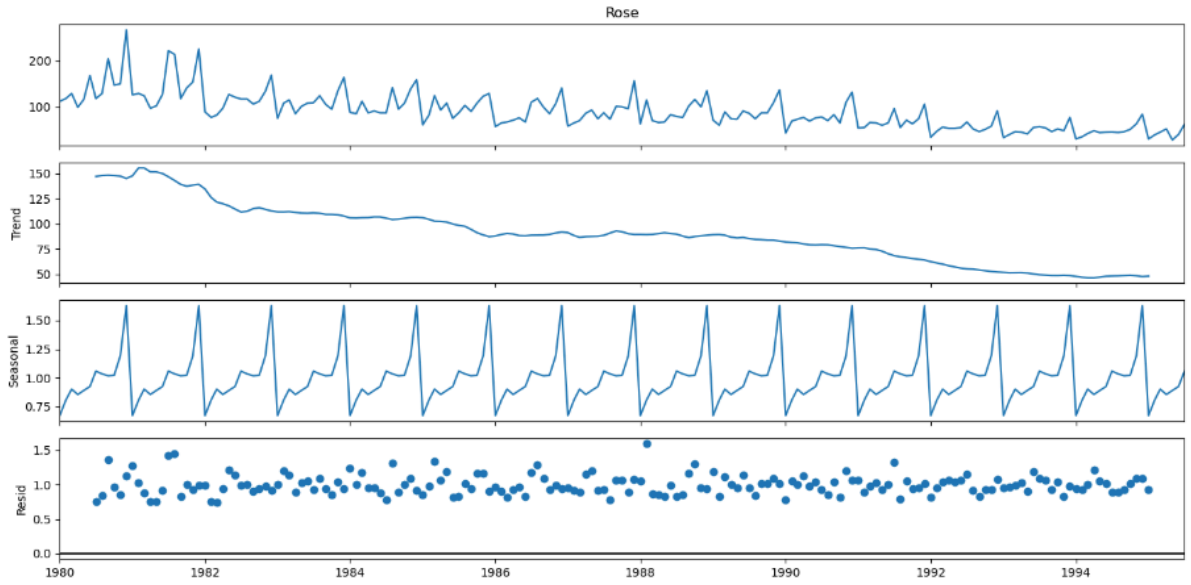


Fig-8 Decomposition of Rose dataset

Now, we will split the rose dataset into a test-train dataset and, since it's a time series model we will split the dataset in such a way that all the observation before the year 1991 is in training dataset and from 1991 year all observation shall be in test dataset. And as seen in Table-8 below we get the fist and last five rows of both the training and testing dataset.The split can be observed in the Fig-9 also.

First few rows of Training Data:

YearMonth	Rose
1980-01-01	112.0
1980-02-01	118.0
1980-03-01	129.0
1980-04-01	99.0
1980-05-01	116.0

Last few rows of Training Data:

YearMonth	Rose
1990-08-01	70.0
1990-09-01	83.0
1990-10-01	65.0
1990-11-01	110.0
1990-12-01	132.0

First few rows of Test Data:

YearMonth	Rose
1991-01-01	54.0
1991-02-01	55.0
1991-03-01	66.0
1991-04-01	65.0
1991-05-01	60.0

Last few rows of Test Data:

YearMonth	Rose
1995-03-01	45.0
1995-04-01	52.0
1995-05-01	28.0
1995-06-01	40.0
1995-07-01	62.0

Table-8 First and last five rows for Train and test dataset

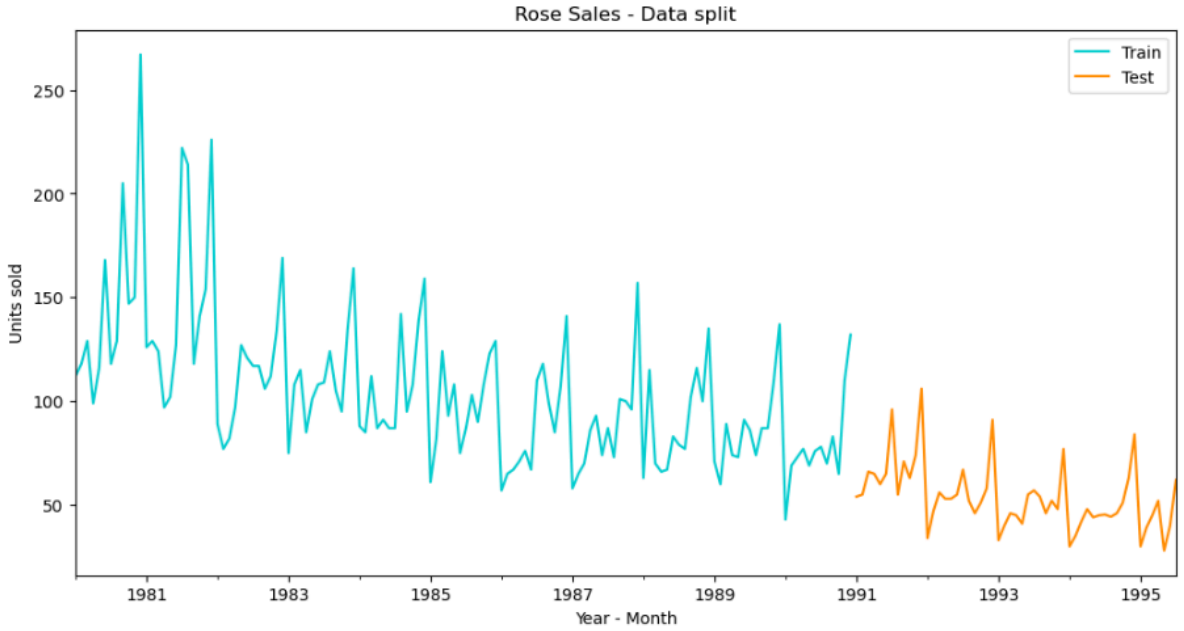


Fig-9 Split for Test-Train dataset

2. Model Building-Original Data:

Linear Regression Model:

Now, We built our first Time forecasting model using LinearRegression model and we obtained the model as seen in Fig-10.

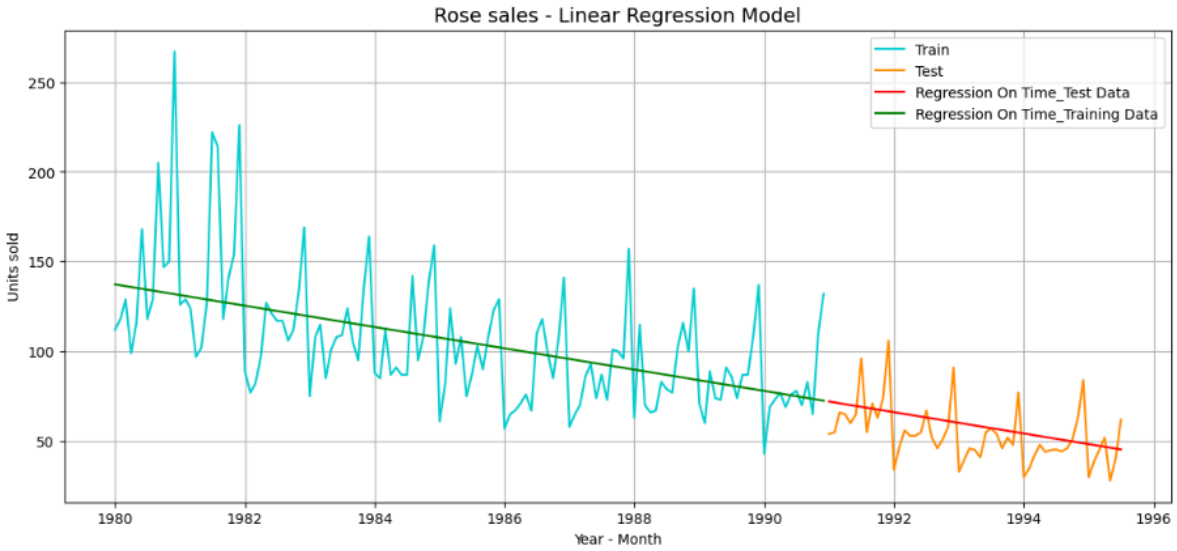


Fig-10 Linear Regression Model

We got the RMSE value on the test data as 15.278 and it has been stored in a dataframe named rose_resultDf.

Naive Model:

Now, we use Naive Bayes model for forecasting and obtained the below model as seen in FIg-11.

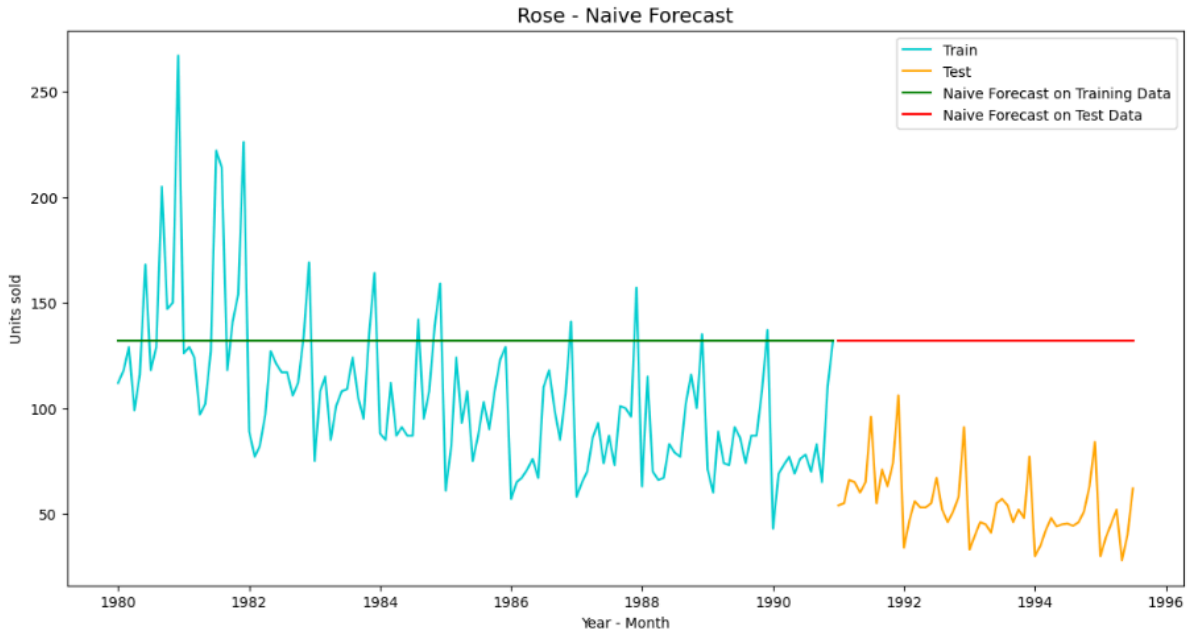


Fig-11 Naive Model

THe RMSE for Naive Model is 76.745 and has been stored in the rose result dataset.

Simple Average Model:

The third model we will make is using Simple Average and obtained the model as seen in Fig-12. THe RMSE for Simple Average is 15.774.

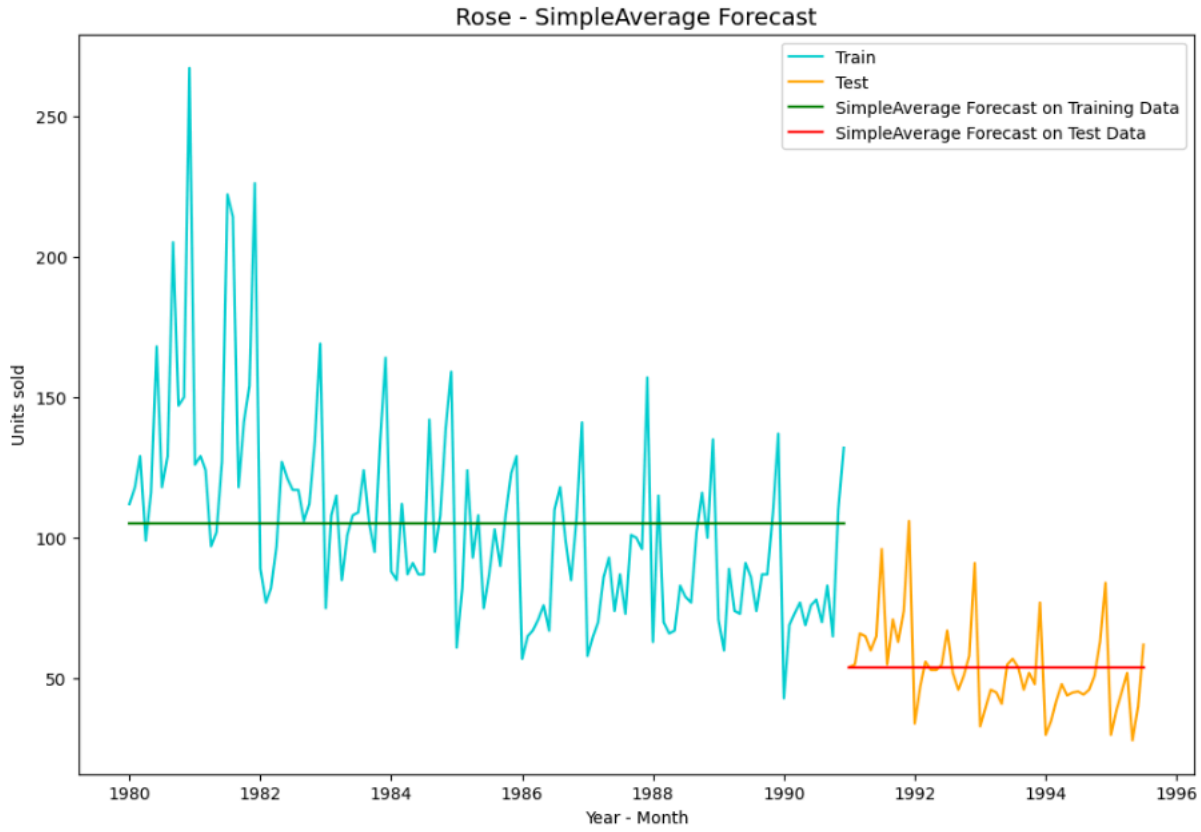


Fig-12 Simple Average model

Moving Average Model:

Using Moving Average we will calculate all the rolling means or trailing moving averages for different-different intervals. In this the best interval will be determined by maximum accuracy.We calculated and plotted the moving average for the rose dataset as seen in Table-9 and Fig-13.

YearMonth	Rose	Rose_Trailing_2	Rose_Trailing_4	Rose_Trailing_6	Rose_Trailing_9
1980-01-01	112.0	NaN	NaN	NaN	NaN
1980-02-01	118.0	115.0	NaN	NaN	NaN
1980-03-01	129.0	123.5	NaN	NaN	NaN
1980-04-01	99.0	114.0	114.5	NaN	NaN
1980-05-01	116.0	107.5	115.5	NaN	NaN

Table-9 Moving Average Values

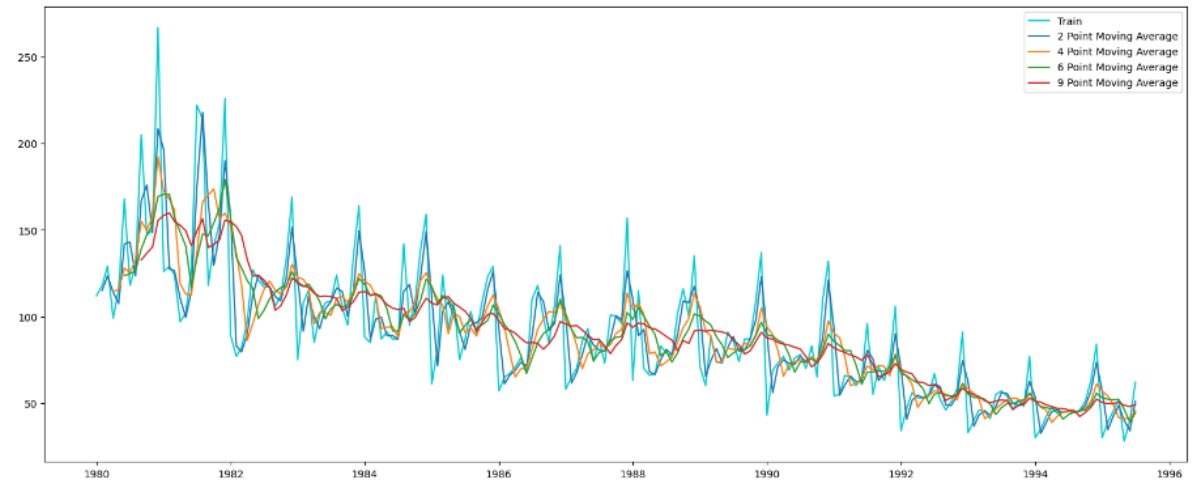


Fig-13 Moving Average model

After this we will apply all the moving averages on the test and train dataset and the line plot as shown in Fig-14. The RMSE values obtained for each rolling means are as follows:

For 2 point Moving Average Model forecast on the Test Data, rmse_rose is 11.530

For 4 point Moving Average Model forecast on the Test Data, rmse_rose is 14.458

For 6 point Moving Average Model forecast on the Test Data, rmse_rose is 14.573

For 9 point Moving Average Model forecast on the Test Data, rmse_rose is 14.733

Fig-14 Moving Average for Test-train dataset

Before moving forward we will plot a chart for all models to see what we have got up till now as seen in Fig-15. It can be said that we are creating better models but not accurate one's that can be used for forecasting.

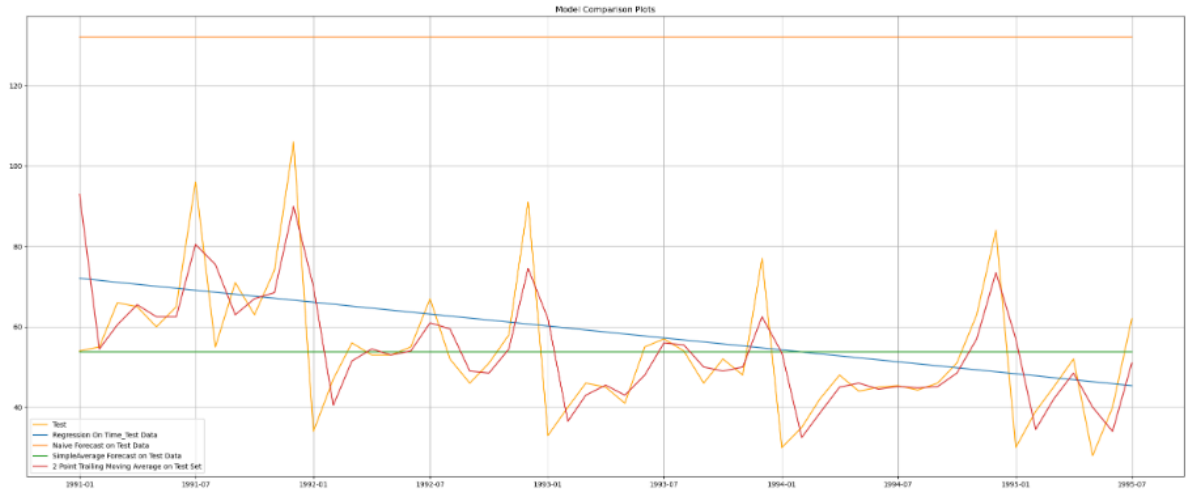


Fig-15 Line Chart for all Models

Simple Exponential Smoothing Model(Single,Double,Triple):

The next model we will create is using Simple Exponential Smoothing in Autofit/optimised and Manualfit/Iterative. The Fig-16 shows the autofit of SES and fig-17 shows the manual fit of SES.

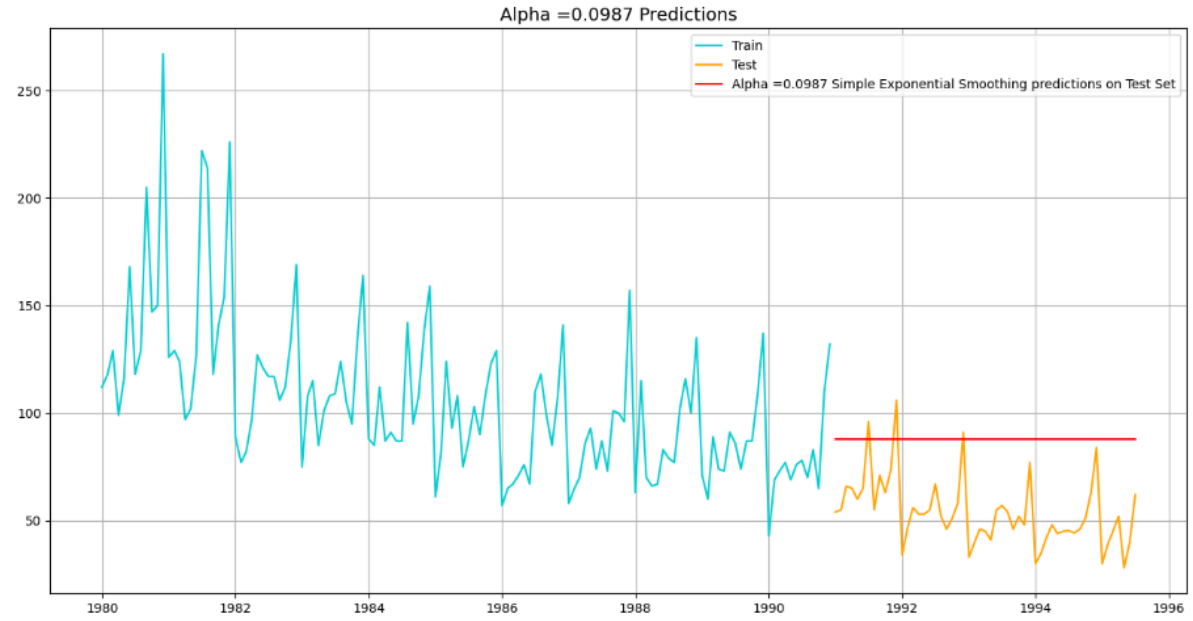


Fig-16 Optimised SES Model

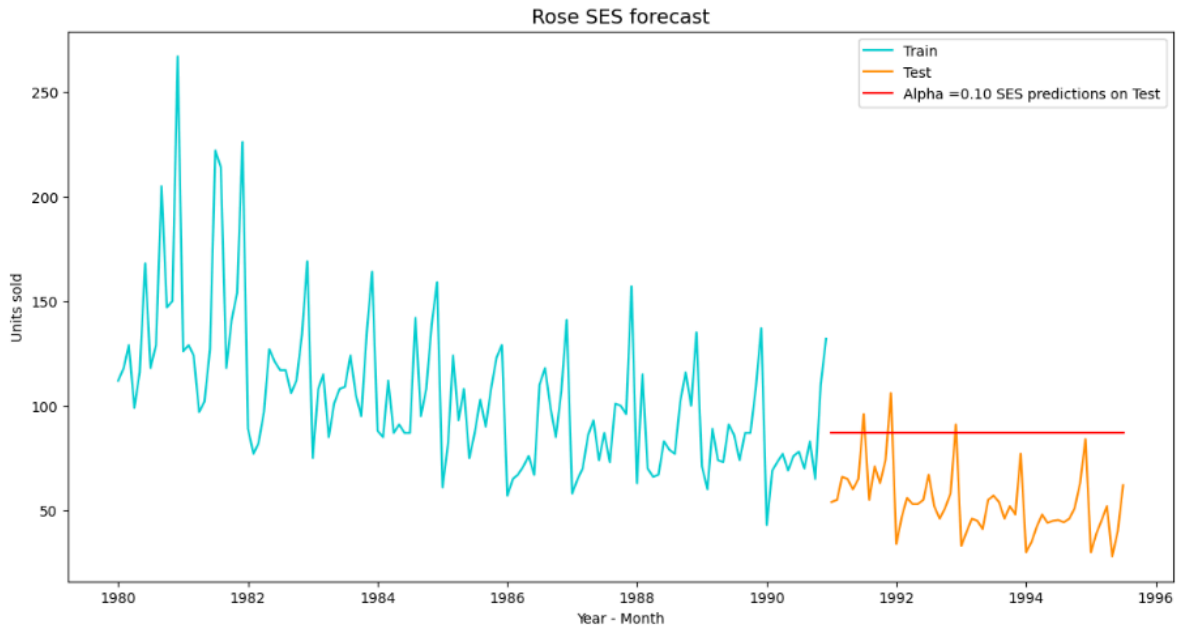


Fig-17 Iterative SES Model

Now we will do Double Exponential Smoothing in Autofit and Manual fit. From Fig-18 we can see DES in Autofit and Fig-19 shows the Manual fit for DES.

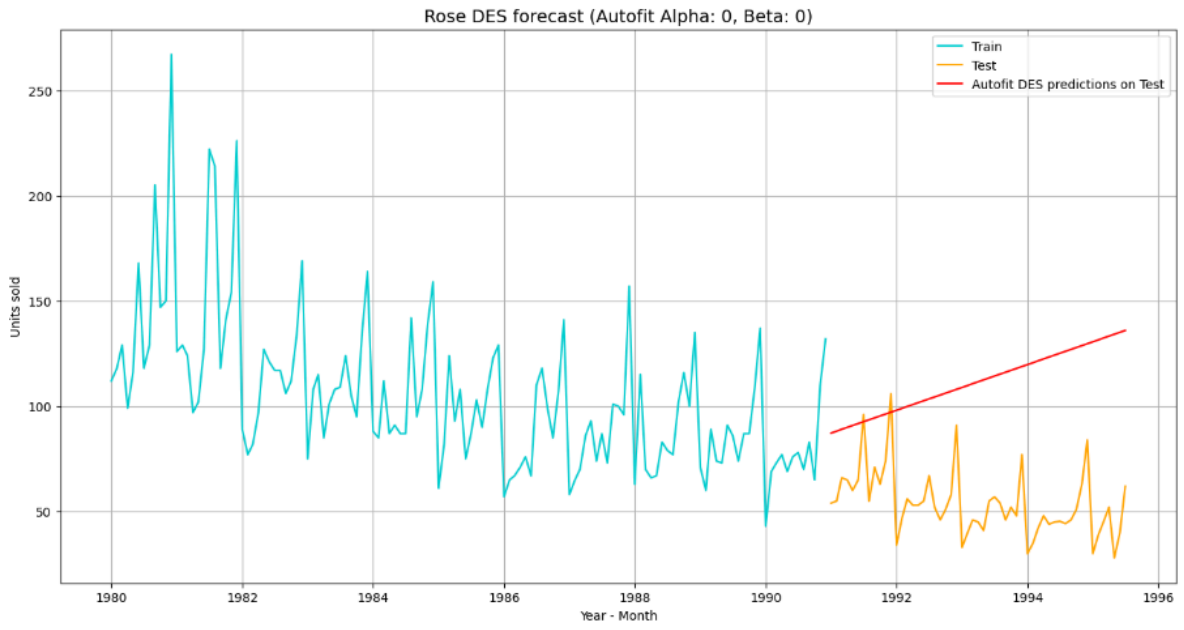


Fig-18 Optimised DES Model

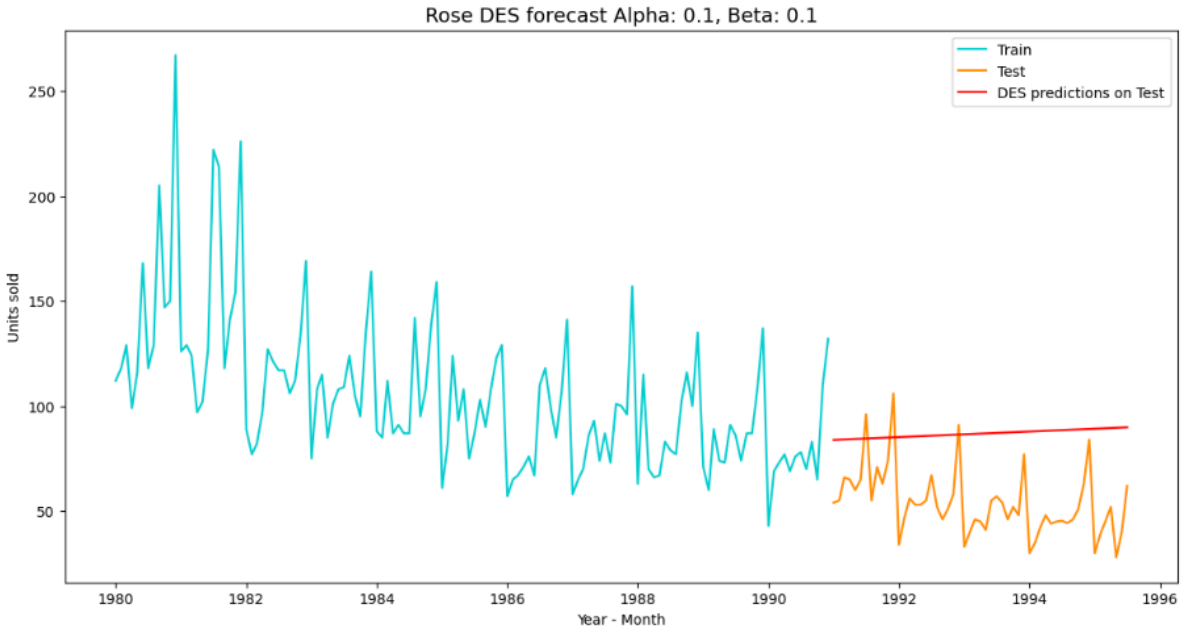


Fig-19 Iterative DES Model

Finally we will go for Triple Exponential Smoothing same as before for both Optimised and Iterative Models. THe Fig-20 shows the Autofit/optimised model of TES and Fig-21 shows the Iterative/Manualfit model of TES.

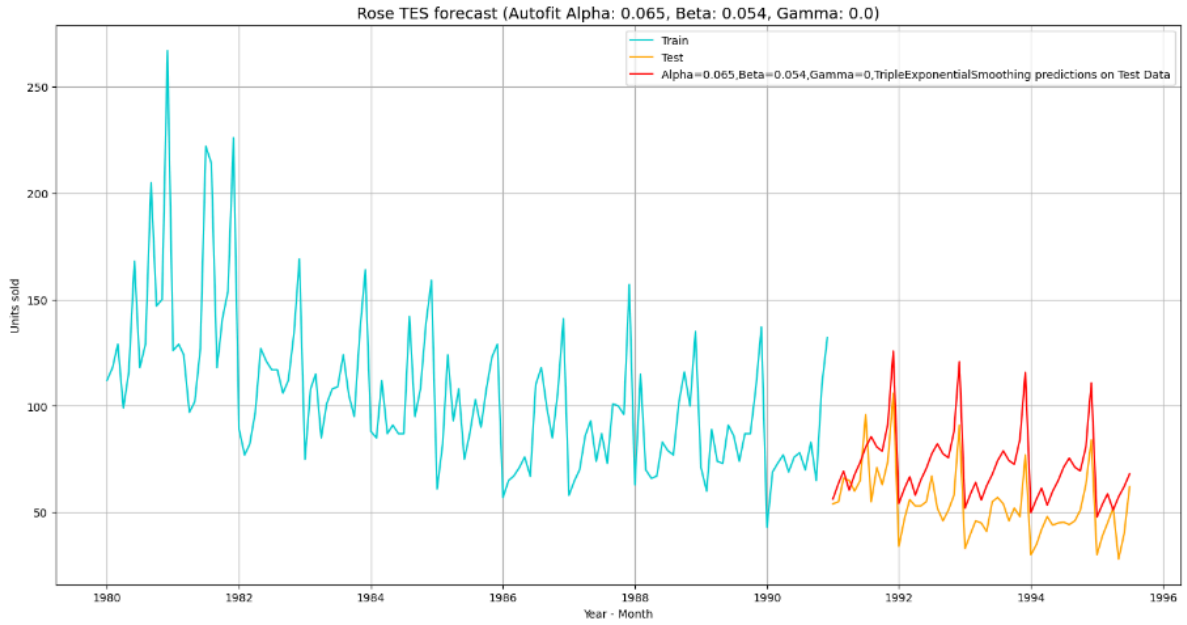


Fig-20 Optimised TES Model

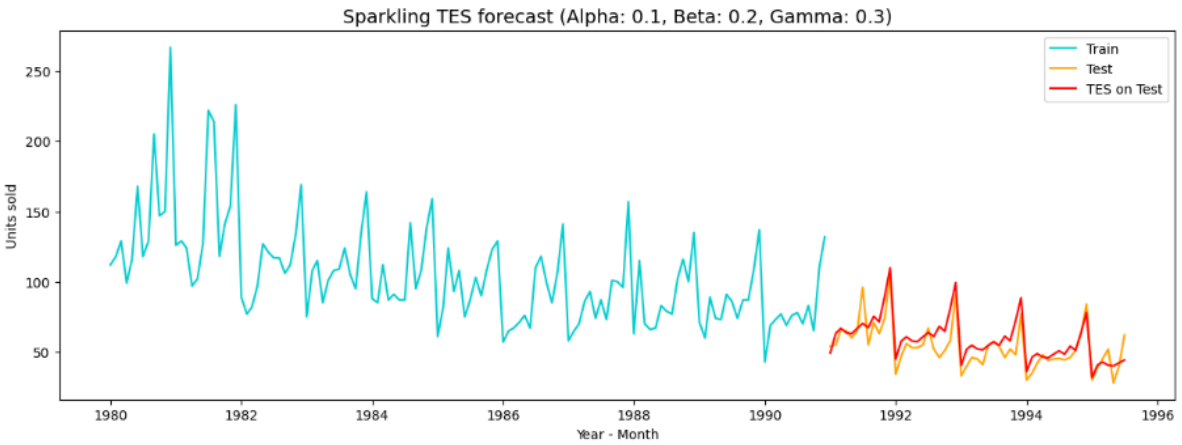


Fig- 21 Iterative TES Model

The Table-10 and Fig-22 shows the RMSE values in sorted order for the models we have used for forecasting up till now in the below:

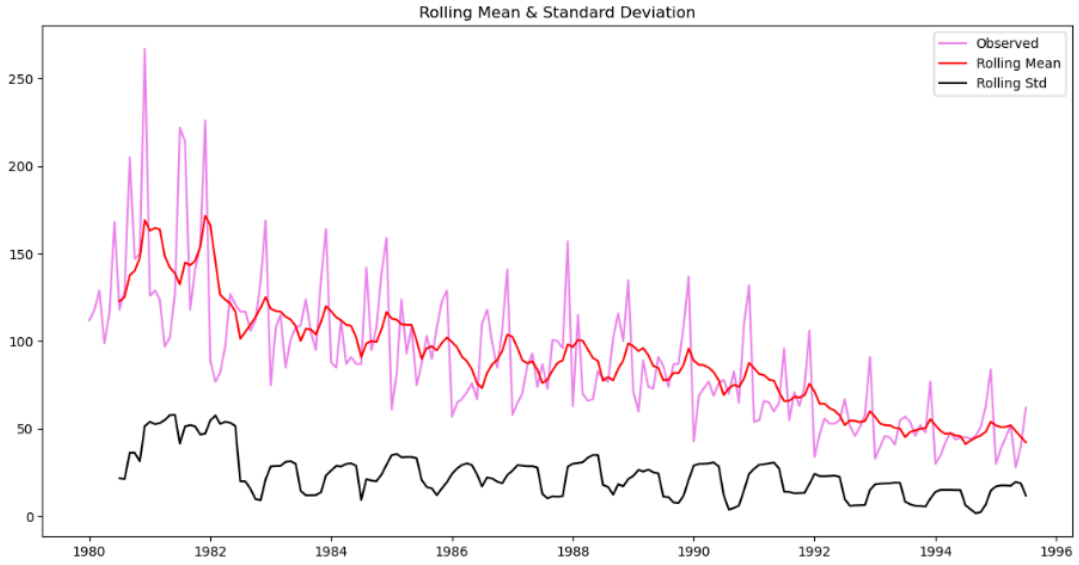
	Test RMSE
2 point TMA	11.530054
4 point TMA	14.458402
6 point TMA	14.572976
9 point TMA	14.732918
RegressionOnTime	15.278369
SimpleAverage	15.773671
Alpha=0.065,Beta=0.054,gamma=0.0 TES Optimized	20.194223
Alpha=0.1,Beta=0.2,gamma=0.3,TES_Iterative	23.656276
Alpha=0.10,SES_Iterative	36.856268
Alpha=0.1,Beta=0.1,DES_Iterative	36.950000
Alpha=0.0987, SES Optimized	37.620427
Alpha=0.0,Beta=0.0, DES Optimized	63.074706
NaiveModel	79.745697

Table-10 Sorted Test RMSE value

Fig-22 Forecast v/s Actual values of all models

3. Check For Stationarity:

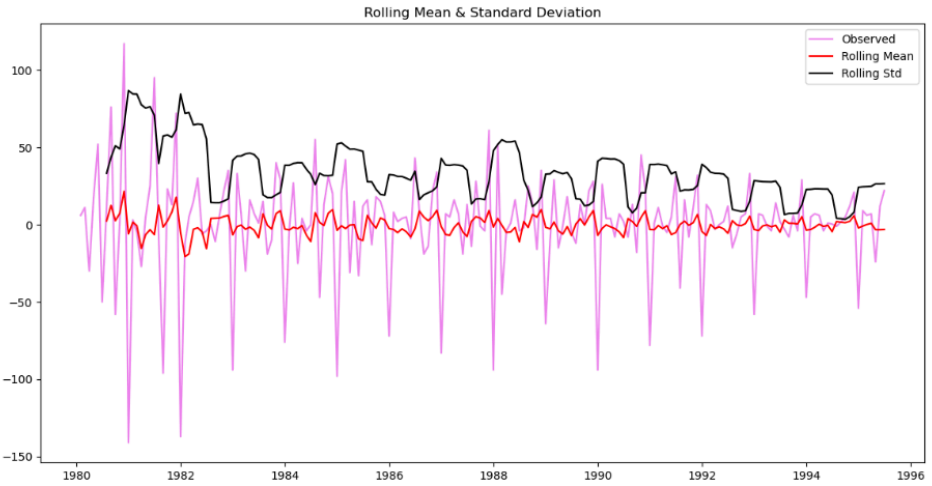
We will check whether the dataset is stationary or not. By applying Augmented Dickey Fuller test it has been found out that the rose dataset is non-stationary as seen in Fig-23.



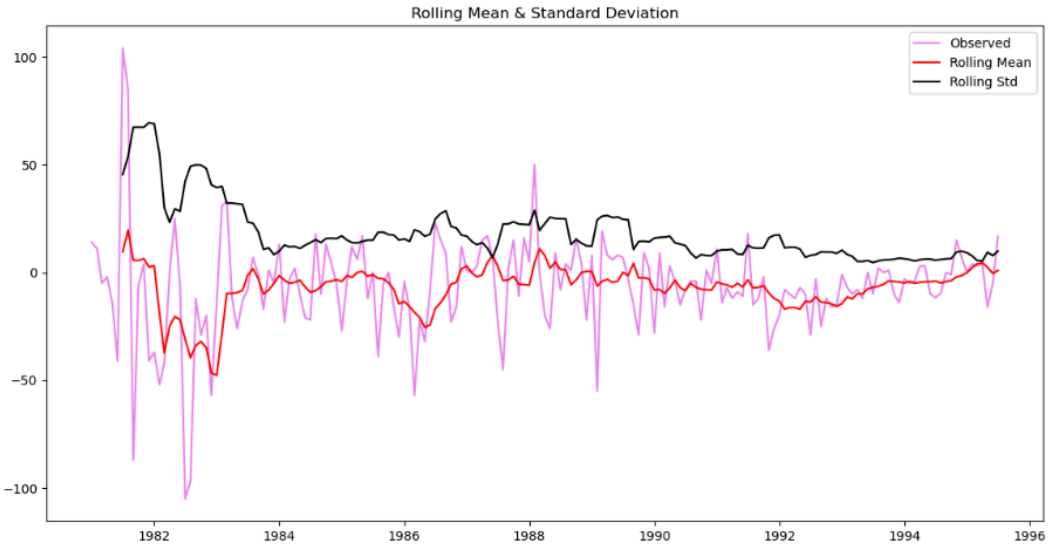

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.872615
p-value                          0.345051
#Lags Used                      13.000000
Number of Observations Used    173.000000
Critical Value (1%)             -3.468726
Critical Value (5%)             -2.878396
Critical Value (10%)            -2.575756
dtype: float64
```

Fig-23 Rolling mean and STD for Original Rose dataset

To make it a stationary dataset various combinations of differentials intervals are used as seen in Fig-24 and in Fig-25 we see the combinations for log transformations also.

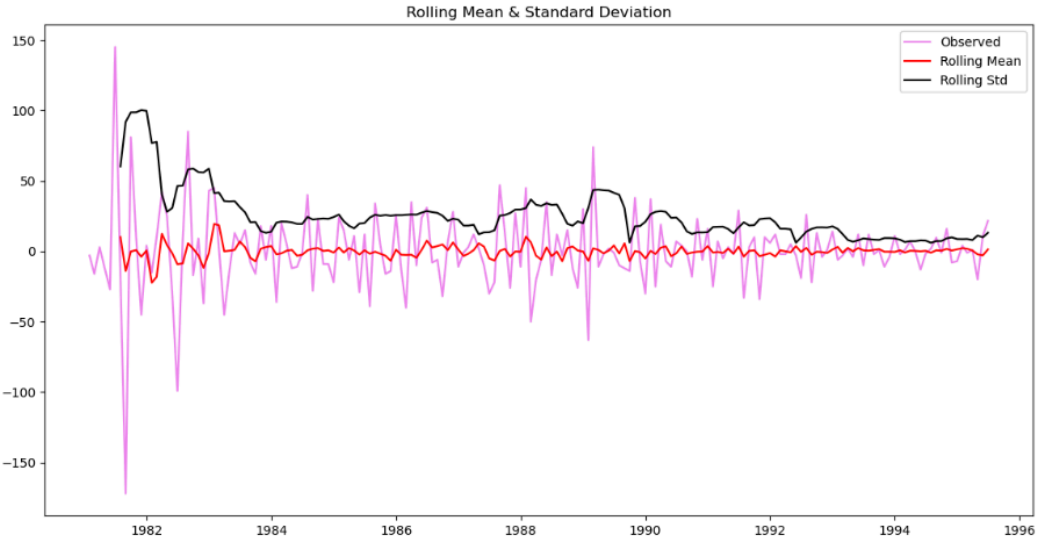



```
Results of Dickey-Fuller Test:
Test Statistic                -8.044081e+00
p-value                        1.814191e-12
#Lags Used                     1.200000e+01
Number of Observations Used    1.730000e+02
Critical Value (1%)           -3.468726e+00
Critical Value (5%)           -2.878396e+00
Critical Value (10%)          -2.575756e+00
dtype: float64
```

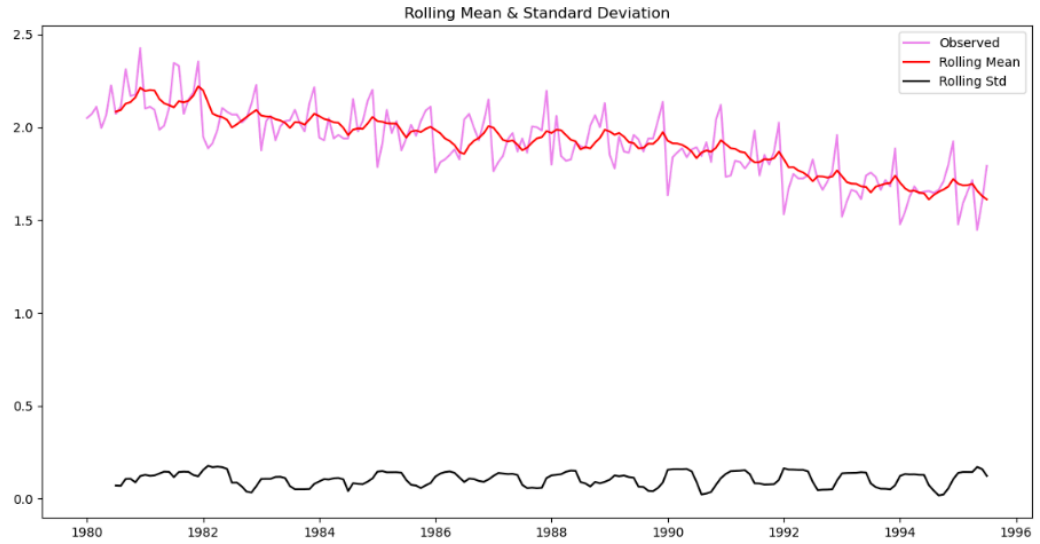



```
Results of Dickey-Fuller Test:
Test Statistic                  -4.257265
p-value                          0.000526
#Lags Used                      11.000000
Number of Observations Used    163.000000
Critical Value (1%)             -3.471119
Critical Value (5%)             -2.879441
Critical Value (10%)            -2.576314
dtype: float64
```

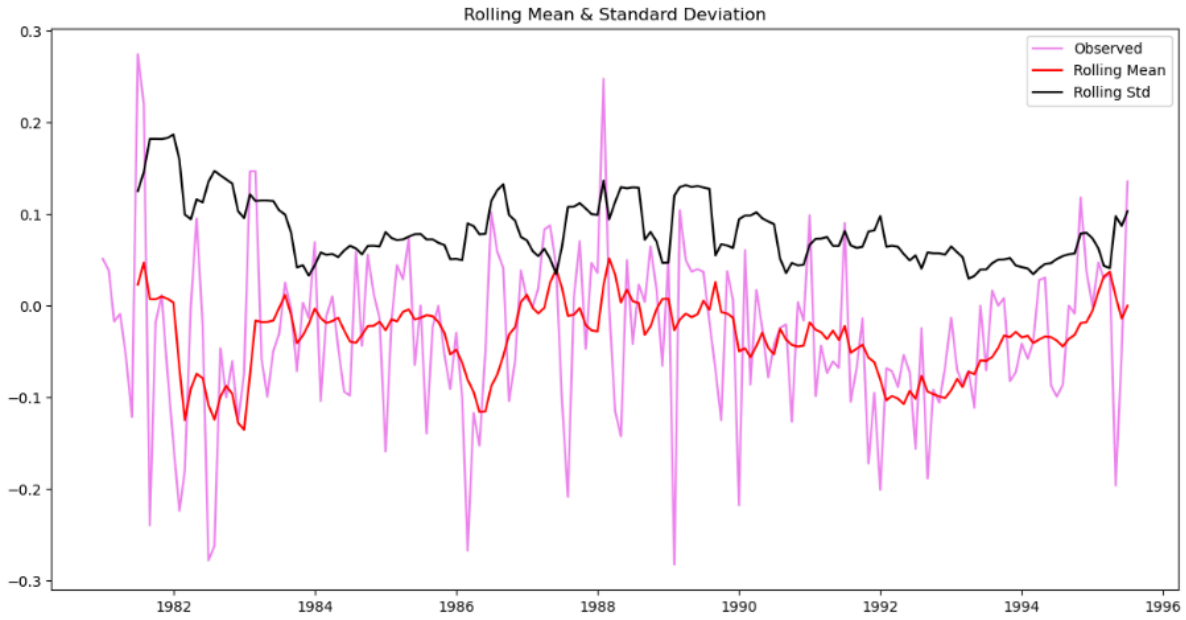
Fig-24 Rolling mean and STD without log transformation




```
Results of Dickey-Fuller Test:
Test Statistic                  -4.605791
p-value                          0.000126
#Lags Used                      11.000000
Number of Observations Used    162.000000
Critical Value (1%)             -3.471374
Critical Value (5%)             -2.879552
Critical Value (10%)            -2.576373
dtype: float64
```



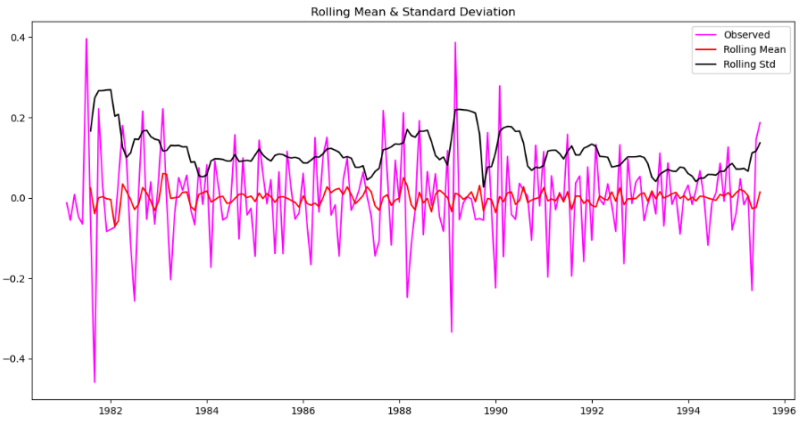

```
Results of Dickey-Fuller Test:
Test Statistic                  -0.412363
p-value                          0.908014
#Lags Used                      12.000000
Number of Observations Used    174.000000
Critical Value (1%)             -3.468502
Critical Value (5%)             -2.878298
Critical Value (10%)            -2.575704
dtype: float64
```




```
Results of Dickey-Fuller Test:
Test Statistic                  -3.934772
p-value                          0.001793
#Lags Used                      11.000000
Number of Observations Used    163.000000
Critical Value (1%)             -3.471119
Critical Value (5%)             -2.879441
Critical Value (10%)            -2.576314
dtype: float64
```

Fig-25 Rolling mean and STD with log transformation

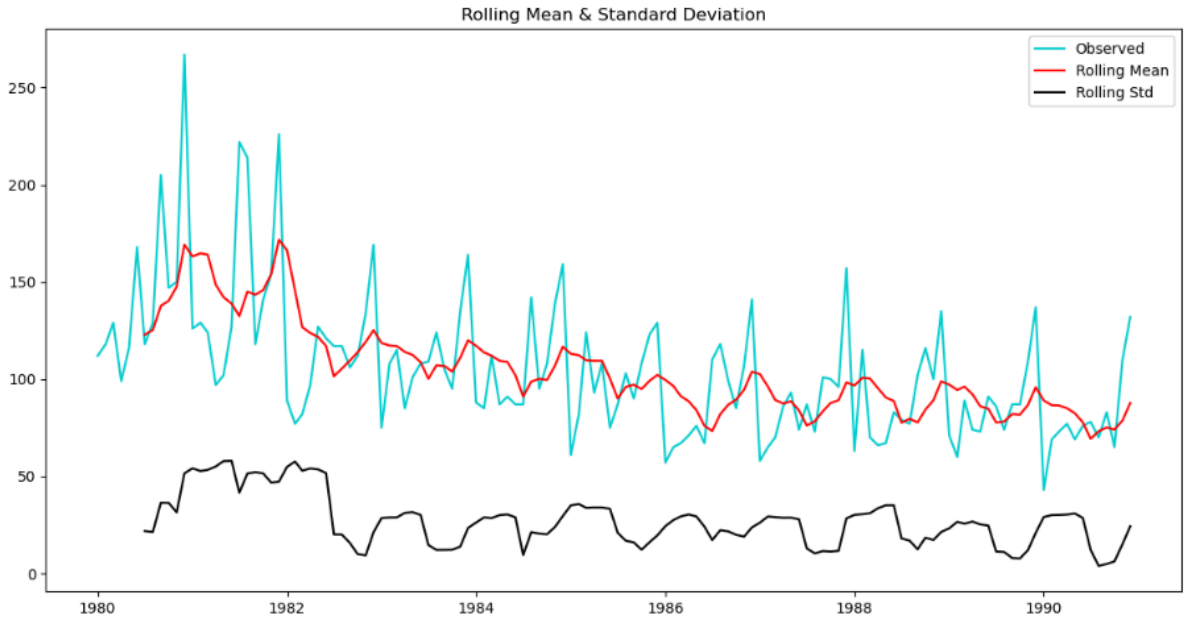
As per Fig-26 we can see that ADF is also done with log transformation of train data with differing seasonal order of 12.




```
Results of Dickey-Fuller Test:
Test Statistic                  -4.628179
p-value                          0.000115
#Lags Used                      11.000000
Number of Observations Used    162.000000
Critical Value (1%)             -3.471374
Critical Value (5%)             -2.879552
Critical Value (10%)            -2.576373
dtype: float64
```

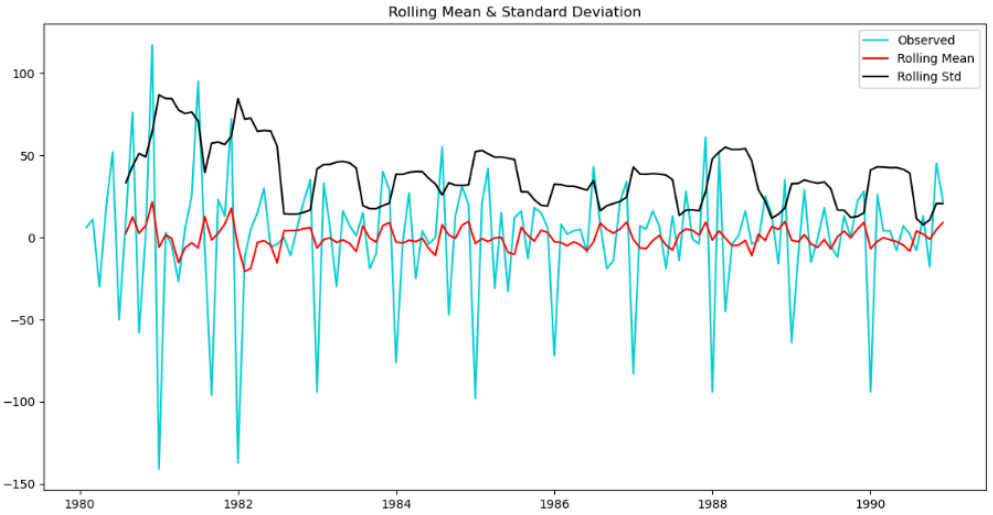
Fig-26 Rolling mean and STD with log transformation and Differential =12

And obtained that with differencing order one when applied is the most stationary one out of all of them and can be compared with the train dataset rolling mean and standard deviation as seen in Fig-27 and Fig-28 respectively.




```
Results of Dickey-Fuller Test:
Test Statistic                  -2.164250
p-value                          0.219476
#Lags Used                      13.000000
Number of Observations Used    118.000000
Critical Value (1%)             -3.487022
Critical Value (5%)             -2.886363
Critical Value (10%)            -2.580009
dtype: float64
```

Fig-27 Rolling mean and STD for train dataset




```
Results of Dickey-Fuller Test:
Test Statistic                -6.592372e+00
p-value                        7.061944e-09
#Lags Used                     1.200000e+01
Number of Observations Used    1.180000e+02
Critical Value (1%)           -3.487022e+00
Critical Value (5%)           -2.886363e+00
Critical Value (10%)          -2.580009e+00
dtype: float64
```

Fig-28 Rolling mean and STD with stationary train dataset

4. Model Building- Stationary Data:

Now, we will find and plot the Autocorrelation and partial Autocorrelation for both the original rose dataset and the training dataset as seen below in Fig-29 and Fig-30 respectively.

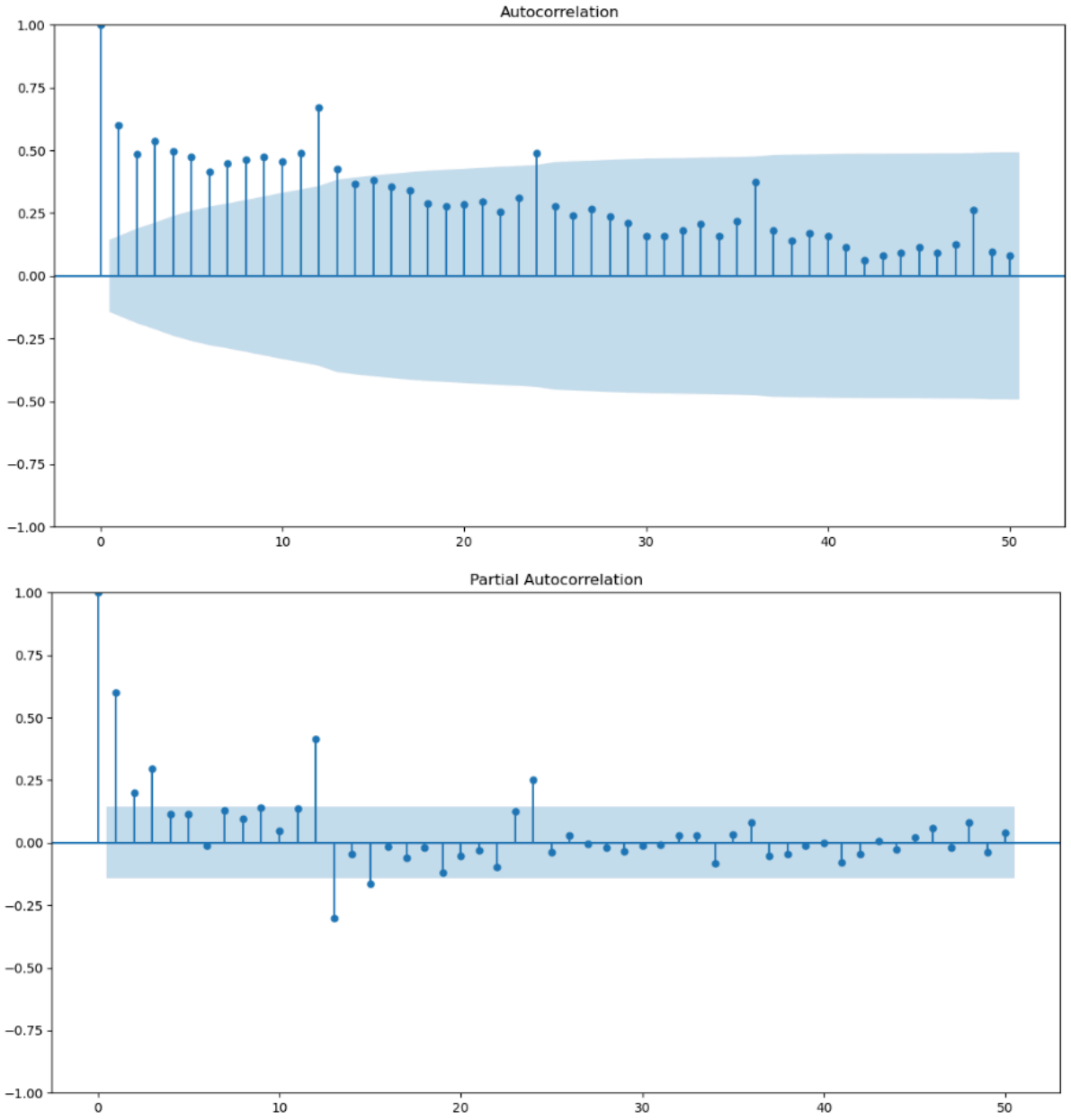


Fig-29 ACF and PACF for Train Dataset

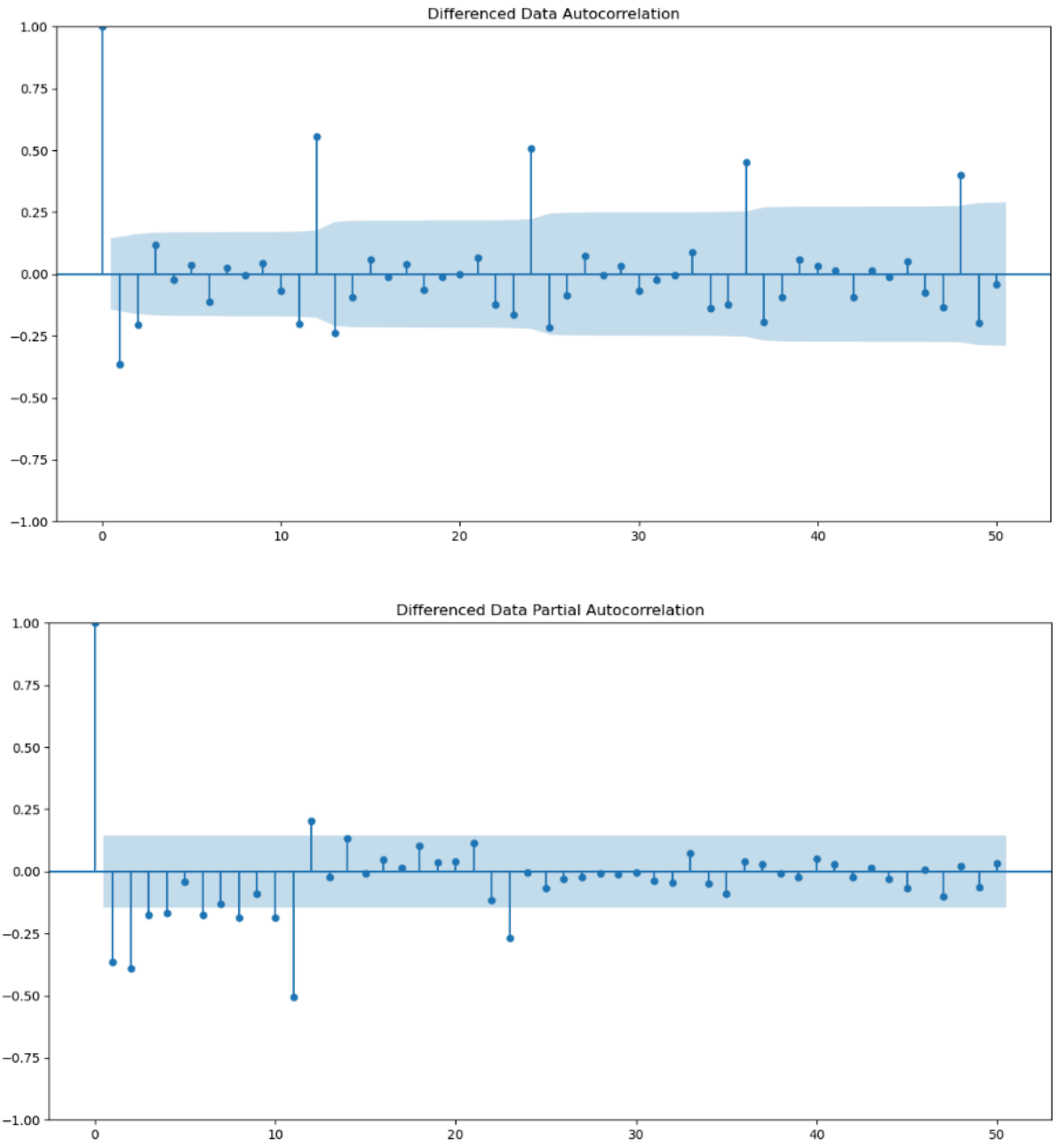


Fig-30 ACF and PACF for Stationary Train Dataset

Auto ARIMA Model:

Next, we will build the ARIMA and SARIMA model for Autofit . For the Autofit model in ARIMA the parameters are selected using AIC(Akaike Information Criteria) as shown in Table-11 below.We used this model on the test data and obtained the RMSE value as 37.334.

SARIMAX Results

Dep. Variable:	Rose	No. Observations:	132
Model:	ARIMA(0, 1, 2)	Log Likelihood	-636.836
Date:	Sat, 14 Sep 2024	AIC	1279.672
Time:	17:07:37	BIC	1288.297
Sample:	01-01-1980 - 12-01-1990	HQIC	1283.176
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	-0.6970	0.072	-9.689	0.000	-0.838	-0.556
ma.L2	-0.2042	0.073	-2.794	0.005	-0.347	-0.061
sigma2	965.8407	88.305	10.938	0.000	792.766	1138.915

Ljung-Box (L1) (Q):	0.14	Jarque-Bera (JB):	39.24
Prob(Q):	0.71	Prob(JB):	0.00
Heteroskedasticity (H):	0.36	Skew:	0.82
Prob(H) (two-sided):	0.00	Kurtosis:	5.13

Table-11 Auto ARIMA model Build

Auto SARIMA Model:

Now, we will build the SARIMA model in autofit using AIC and obtain the below Table-12.

SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(0, 1, 2)x(2, 1, 2, 12)	Log Likelihood	-380.485
Date:	Sat, 14 Sep 2024	AIC	774.969
Time:	17:08:20	BIC	792.622
Sample:	0 - 132	HQIC	782.094
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	-0.9524	0.184	-5.166	0.000	-1.314	-0.591
ma.L2	-0.0764	0.126	-0.605	0.545	-0.324	0.171
ar.S.L12	0.0480	0.177	0.271	0.786	-0.299	0.395
ar.S.L24	-0.0419	0.028	-1.513	0.130	-0.096	0.012
ma.S.L12	-0.7526	0.301	-2.503	0.012	-1.342	-0.163
ma.S.L24	-0.0721	0.204	-0.354	0.723	-0.472	0.327
sigma2	187.8696	45.278	4.149	0.000	99.126	276.613

Ljung-Box (L1) (Q):	0.06	Jarque-Bera (JB):	4.86
Prob(Q):	0.81	Prob(JB):	0.09
Heteroskedasticity (H):	0.91	Skew:	0.41
Prob(H) (two-sided):	0.79	Kurtosis:	3.77

Table-12 Auto SARIMA model Build

Fig-31 shows the diagnostics plot for Auto SARIMA model and tested on test data and obtained the predicted values as shown in Table-13.

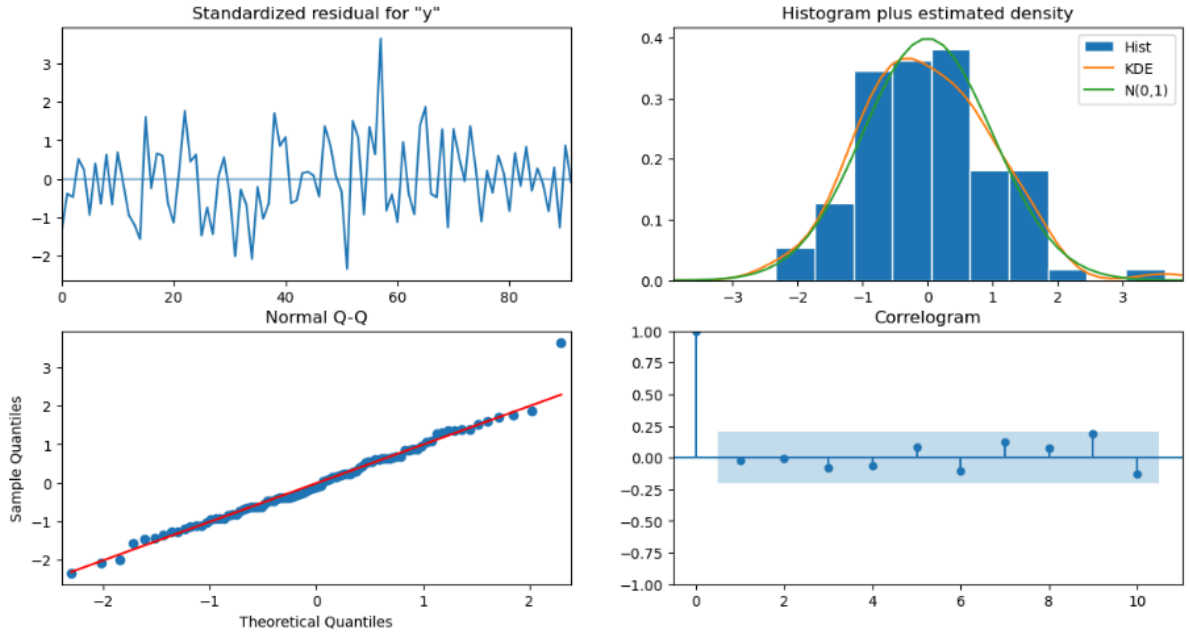


Fig-31 Diagnostic plot for Auto SARIMA

YearMonth	Rose	rose_forecasted
1991-01-01	54.0	44.213524
1991-02-01	55.0	62.326767
1991-03-01	66.0	67.313318
1991-04-01	65.0	63.161053
1991-05-01	60.0	66.474277

Table-13 Predicted values as per Auto SARIMA

We will plot the graph and obtain as seen in Fig-32 how well the Auto SARIMA did on test data and obtained the RMSE value as 16.528.

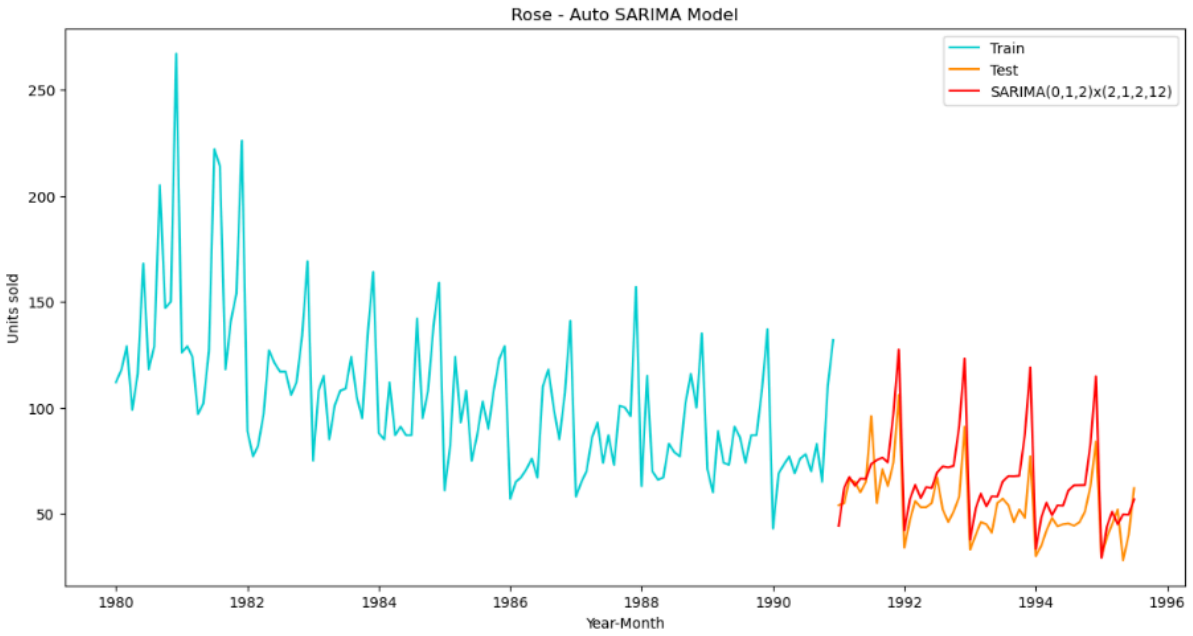
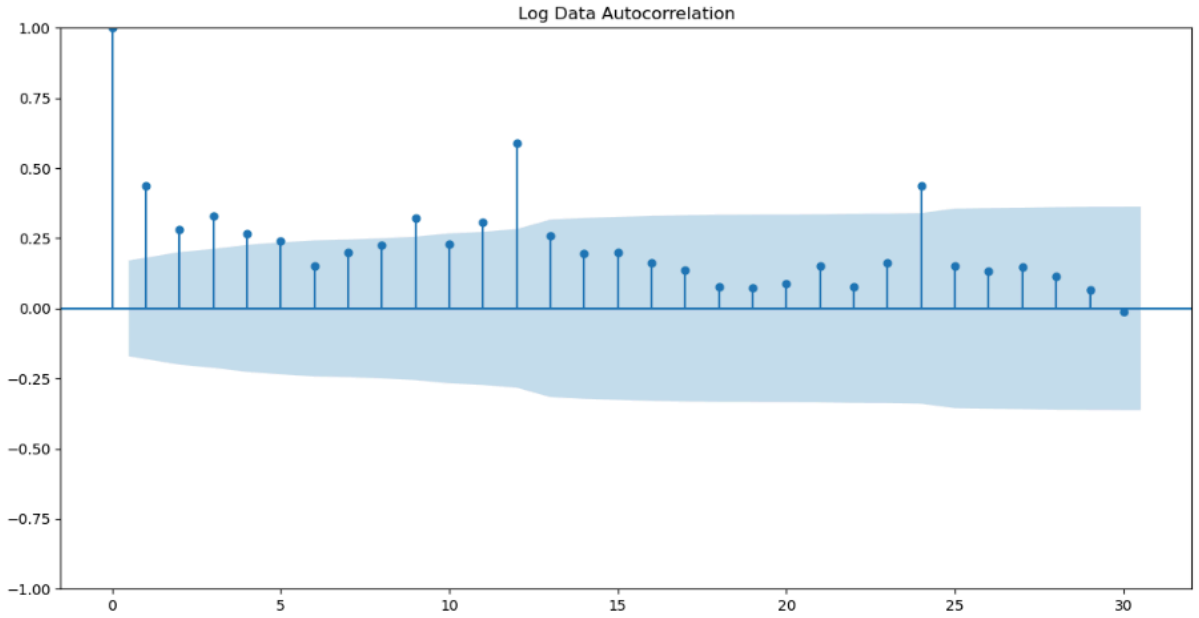
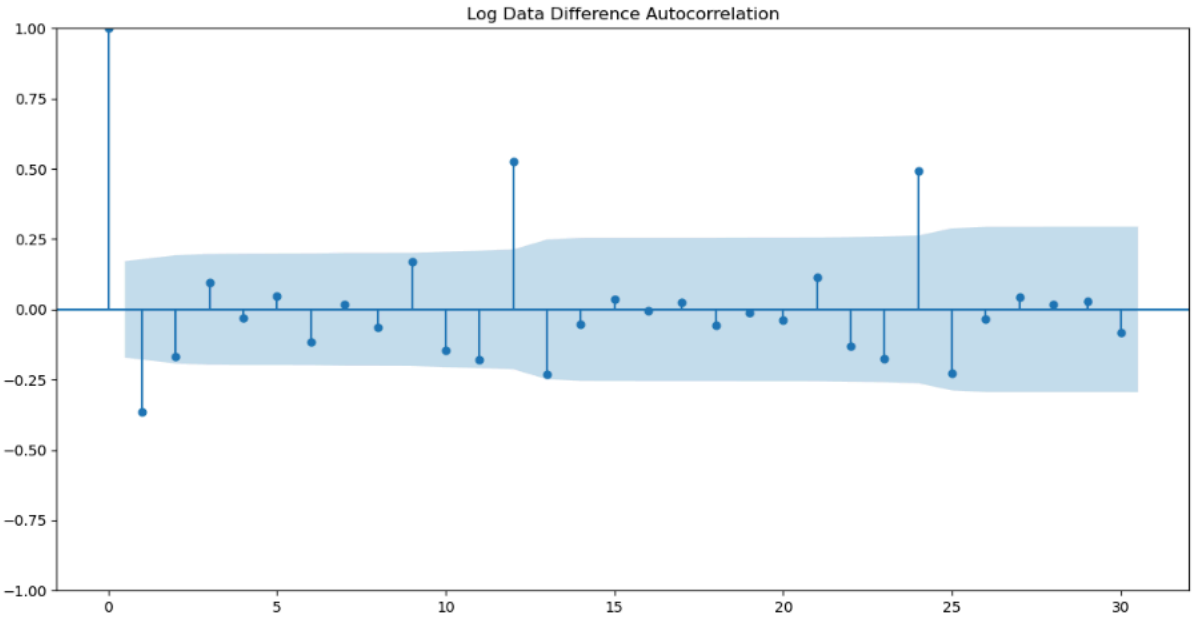


Fig-32 Time series as per Auto SARIMA model

Auto SARIMA Log Transformed Model:

For the Auto SARIMA model we find the ACF and PACF for the log transformed train dataset and obtain the plot as observed in fig-33.


Log Data Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30


Log Data Difference Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30

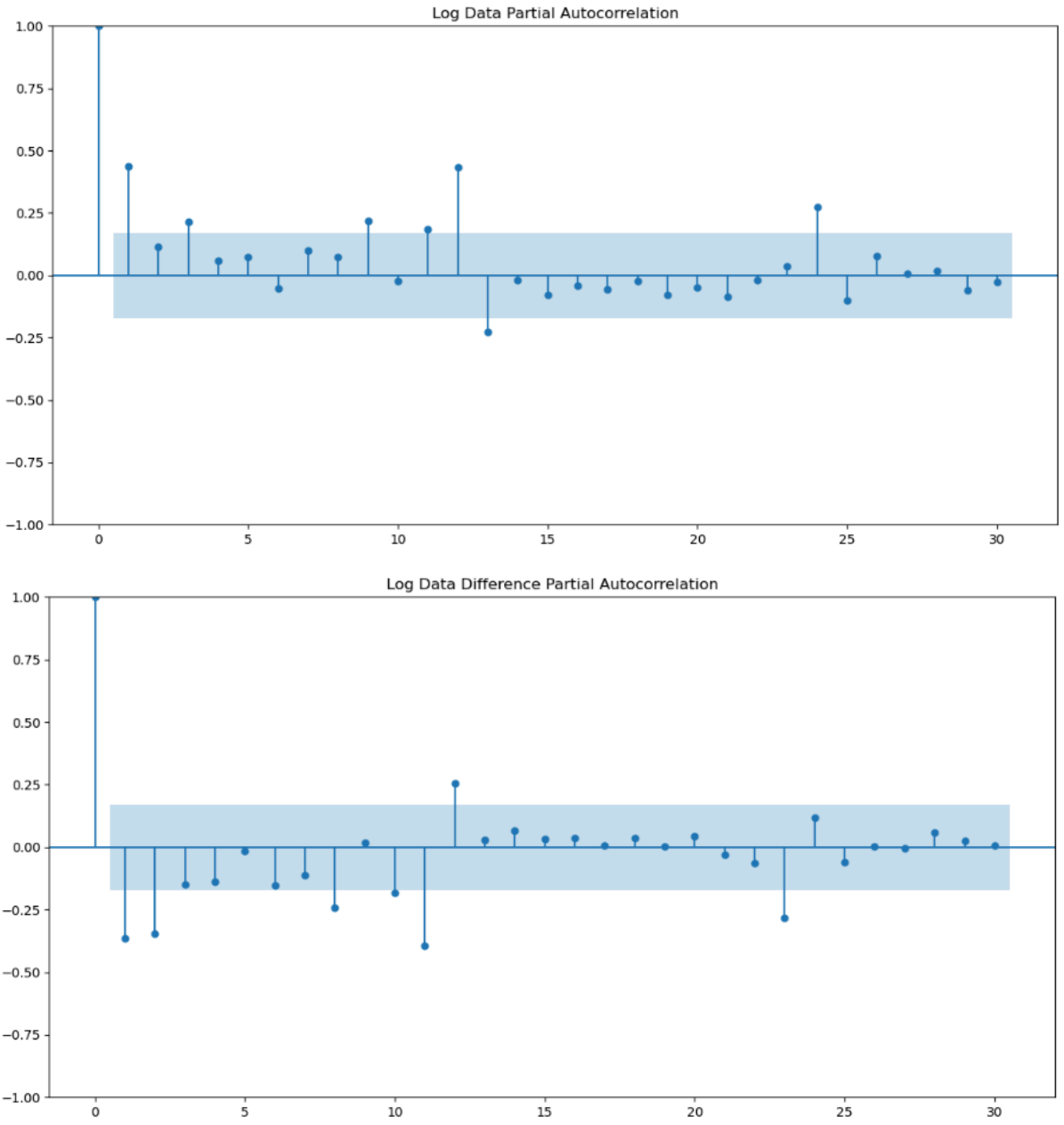


Fig-33 ACF and PACF for log transformed Auto SARIMA

Now, we build the Auto SARIMA model for this log transformed dataset and obtained the Table-14 as seen in below. And its diagnostic plot can be seen in fig-34.

SARIMAX Results

Dep. Variable:	Rose	No. Observations:	132
Model:	SARIMAX(0, 1, 1)x(1, 0, 1, 12)	Log Likelihood	127.538
Date:	Sat, 14 Sep 2024	AIC	-247.076
Time:	17:09:48	BIC	-236.028
Sample:	01-01-1980 - 12-01-1990	HQIC	-242.591
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	-1.0652	0.058	-18.391	0.000	-1.179	-0.952
ar.S.L12	0.9555	0.028	33.785	0.000	0.900	1.011
ma.S.L12	-0.8305	0.151	-5.497	0.000	-1.127	-0.534
sigma2	0.0051	0.001	5.146	0.000	0.003	0.007

Ljung-Box (L1) (Q):	1.31	Jarque-Bera (JB):	0.98
Prob(Q):	0.25	Prob(JB):	0.61
Heteroskedasticity (H):	0.80	Skew:	0.18
Prob(H) (two-sided):	0.50	Kurtosis:	3.26

Table-14 Log transformed Auto SARIMA model Build

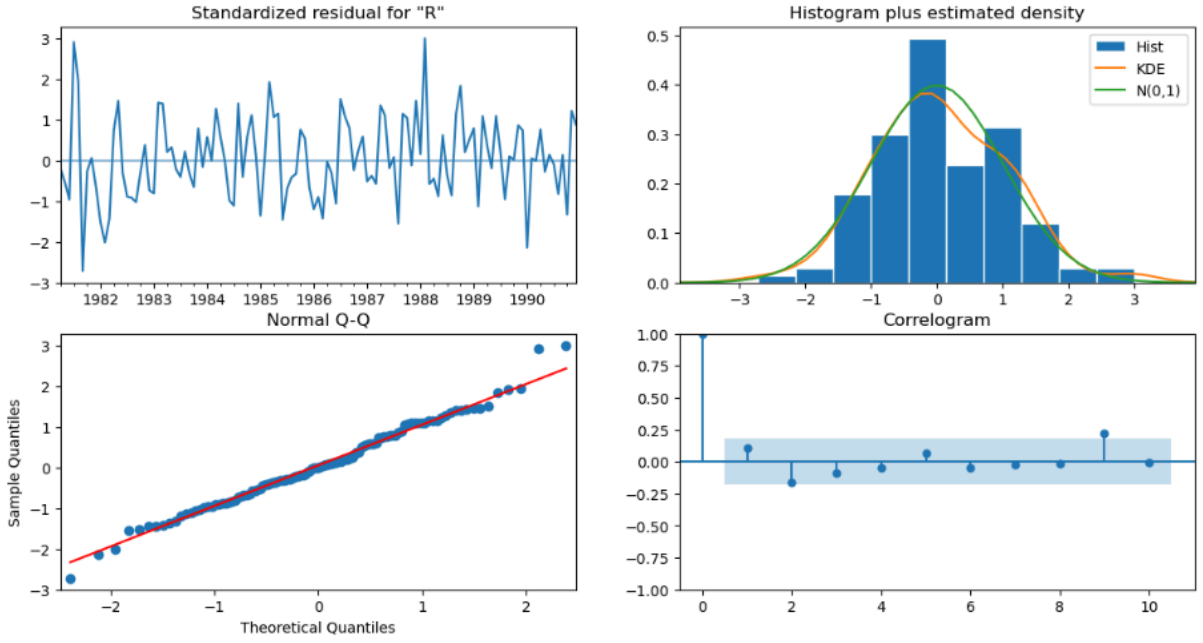


Fig-34 Diagnostic plot for log transformed Auto SARIMA

After the model is trained we apply the log transformation on the test dataset and use this SARIMA dataset to forecast the test data and can be seen in fig-35.The RMSE value obtained for this model is 17.918.

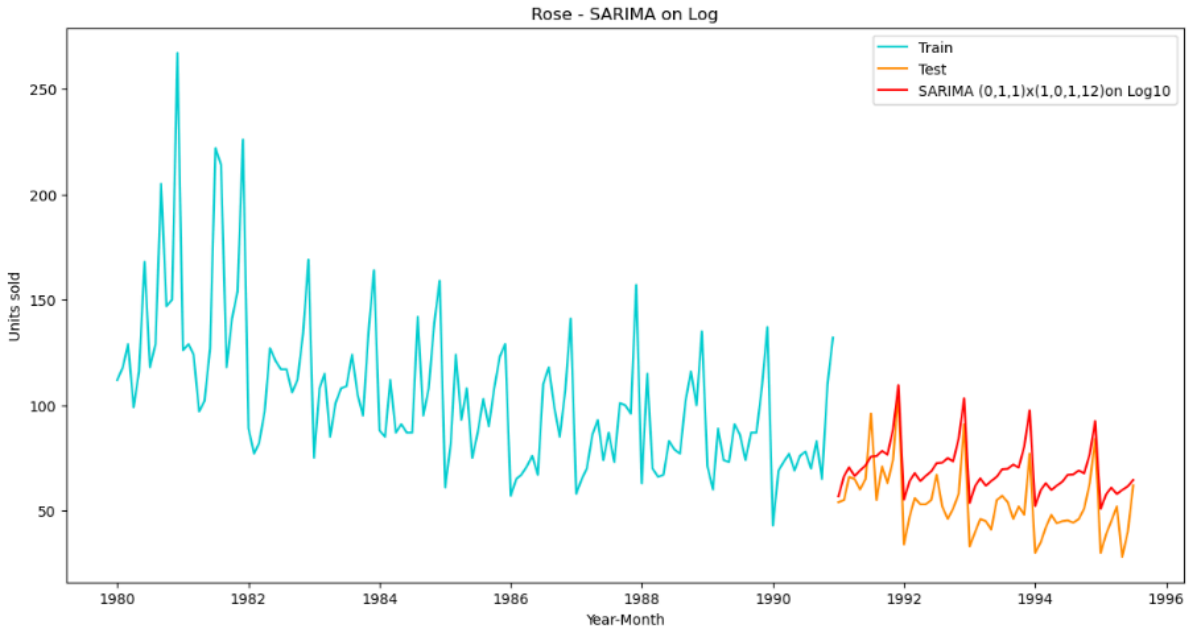


Fig-35 Time series as per log transformed Auto SARIMA model

Manual ARIMA Model:

Now, we will build the Manual ARIMA model using order(0,1,0) and obtain the Table-15.We predict test data values and obtain the value for RMSE is 79.746. For this model we can see the ACF and PACF in Fig-36.

SARIMAX Results

Dep. Variable:	Rose	No. Observations:	132
Model:	ARIMA(0, 1, 0)	Log Likelihood	-665.577
Date:	Sat, 14 Sep 2024	AIC	1333.155
Time:	17:09:50	BIC	1336.030
Sample:	01-01-1980	HQIC	1334.323
	- 12-01-1990		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
sigma2	1515.6738	122.418	12.381	0.000	1275.740	1755.608

Ljung-Box (L1) (Q):	17.11	Jarque-Bera (JB):	59.55
Prob(Q):	0.00	Prob(JB):	0.00
Heteroskedasticity (H):	0.38	Skew:	-0.95
Prob(H) (two-sided):	0.00	Kurtosis:	5.70

Table-15 Manual ARIMA model Build

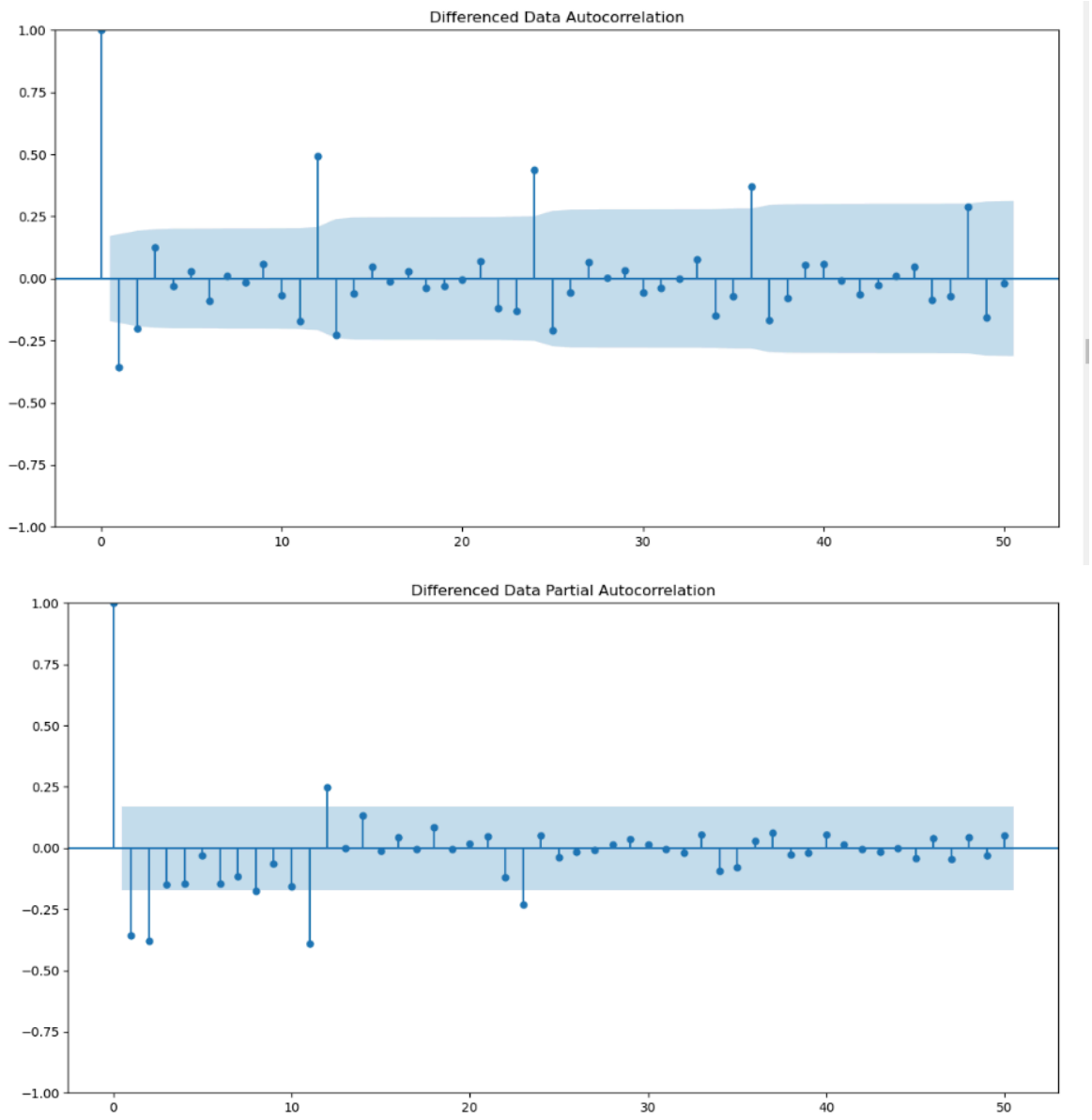


Fig-36 ACF and PACF for Manual ARIMA

Manual SARIMA Model:

Next, we will build the Manual SARIMA model for (4,1,2)*(0,1,2,12) and obtain Table-16. and the forecast the model on test data to obtain a graph as seen in FIg-37.

SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(4, 1, 2)x(0, 1, 2, 12)	Log Likelihood	-384.369
Date:	Sat, 14 Sep 2024	AIC	786.737
Time:	17:09:52	BIC	809.433
Sample:	0	HQIC	795.898
	- 132		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.8967	0.132	-6.814	0.000	-1.155	-0.639
ar.L2	0.0165	0.171	0.097	0.923	-0.319	0.352
ar.L3	-0.1132	0.174	-0.650	0.515	-0.454	0.228
ar.L4	-0.1598	0.116	-1.380	0.168	-0.387	0.067
ma.L1	0.1508	0.174	0.866	0.387	-0.191	0.492
ma.L2	-0.8492	0.164	-5.166	0.000	-1.171	-0.527
ma.S.L12	-0.3907	0.102	-3.848	0.000	-0.590	-0.192
ma.S.L24	-0.0887	0.091	-0.977	0.329	-0.267	0.089
sigma2	238.9649	0.001	2.02e+05	0.000	238.963	238.967

Ljung-Box (L1) (Q):	0.06	Jarque-Bera (JB):	0.01
Prob(Q):	0.80	Prob(JB):	0.99
Heteroskedasticity (H):	0.76	Skew:	-0.01
Prob(H) (two-sided):	0.46	Kurtosis:	3.06

Table-16 Manual SARIMA model Build

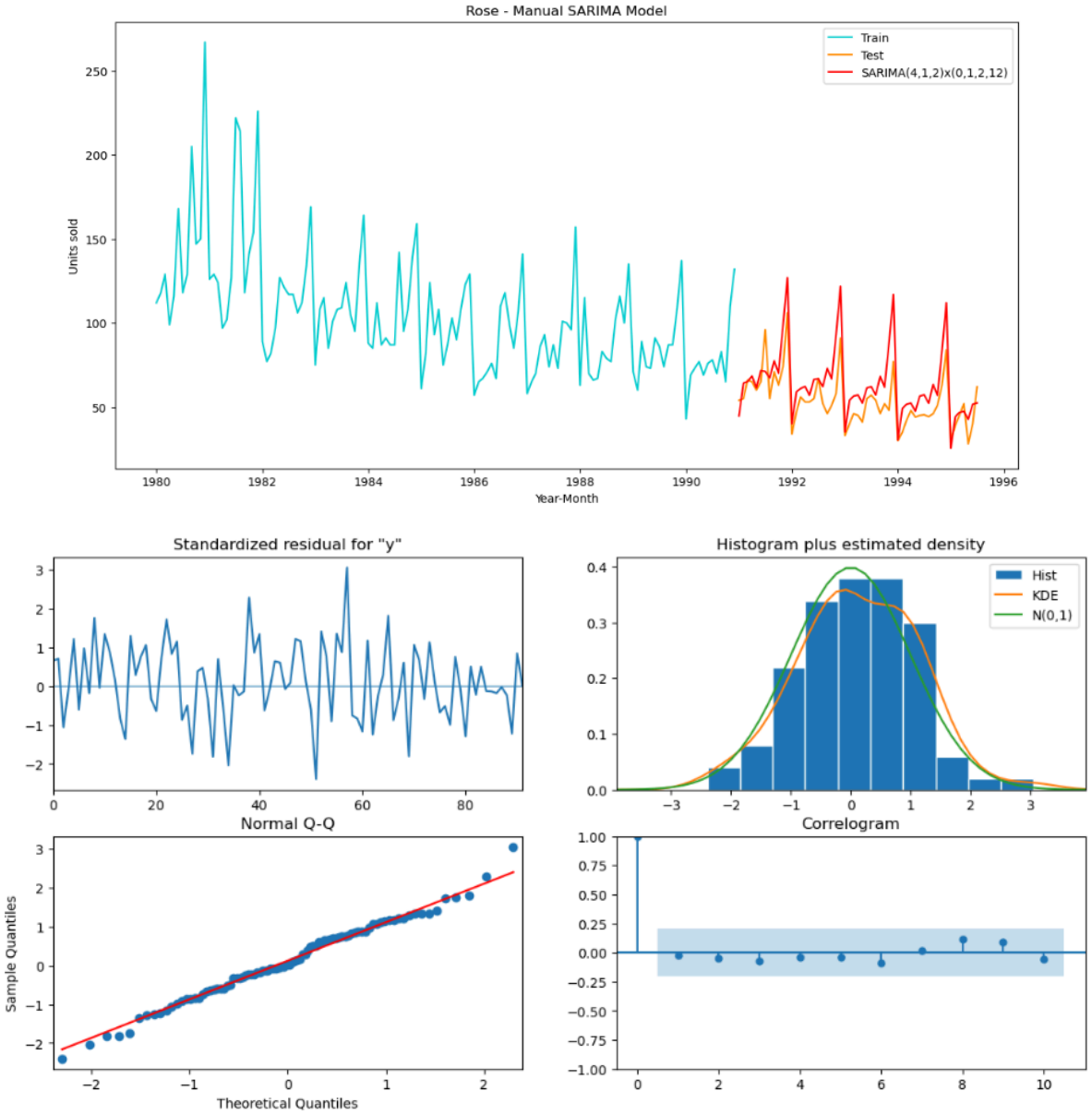
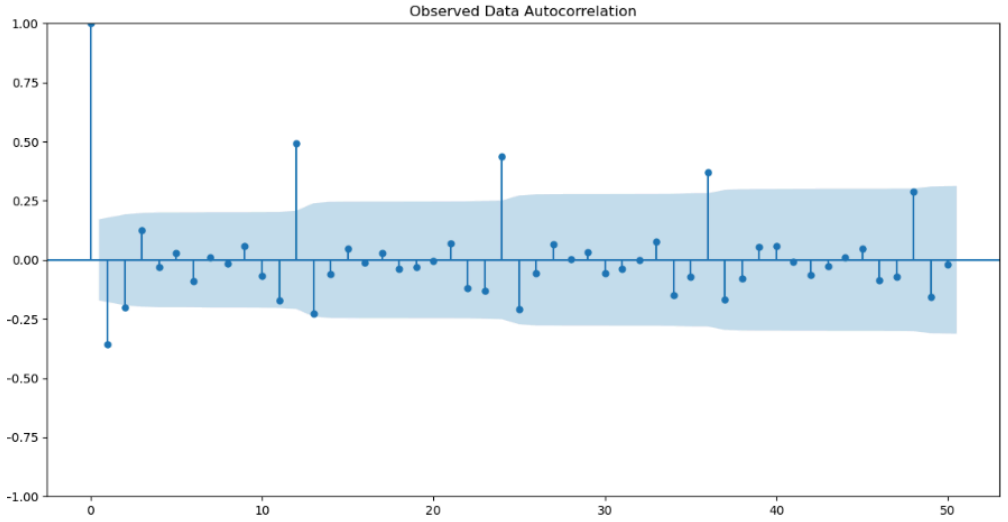
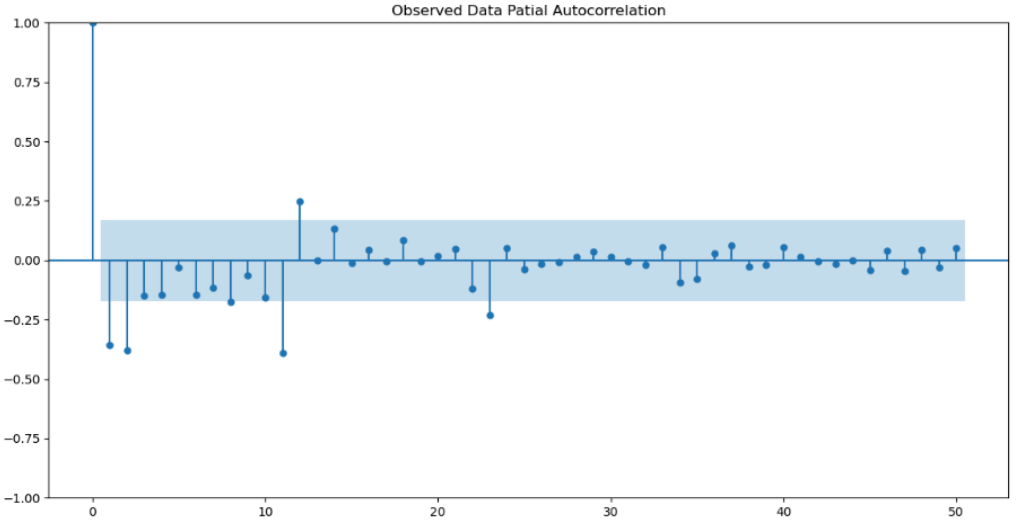
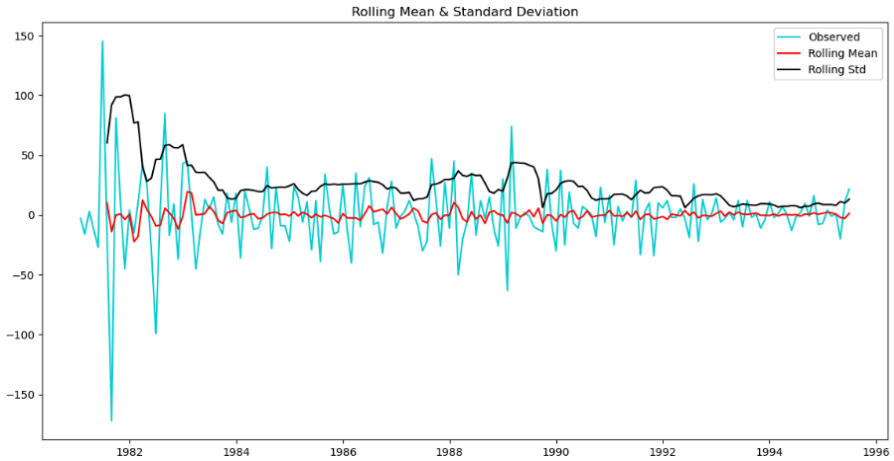


Fig-37 aTIme series for Manual SARIMA

For this model the ACF and PACF are seen in Fig-38. The RMSE value obtained from this model is15.389.








```
Results of Dickey-Fuller Test:
Test Statistic                  -4.605791
p-value                          0.000126
#Lags Used                      11.000000
Number of Observations Used    162.000000
Critical Value (1%)             -3.471374
Critical Value (5%)             -2.879552
Critical Value (10%)            -2.576373
dtype: float64
```

Fig-38 ACF and PACF for Manual SARIMA

5. Compare the performance of the models:

Now as seen in the below Table-17 we can say that the best model is the TES(Triple Exponential Smoothing or Holt Winter's) model.

	Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE
112	0.2	0.5	0.3	23.656276	9.880143
8	0.1	0.2	0.3	20.871304	9.896558
177	0.3	0.3	0.4	24.588120	10.154653
185	0.3	0.4	0.4	25.599445	10.356184
9	0.1	0.2	0.4	21.613205	10.380317

Table-17 TES(Holt Winters) Model as Best model

Fig-39 shows the Forecasted values on the whole original dataset and predicts the future values for the next 12 months.The Fig-40 shows the 12 months that are forecasted for the rose dataset.And Table-18 shows the forecasted values along with its statistics summary.

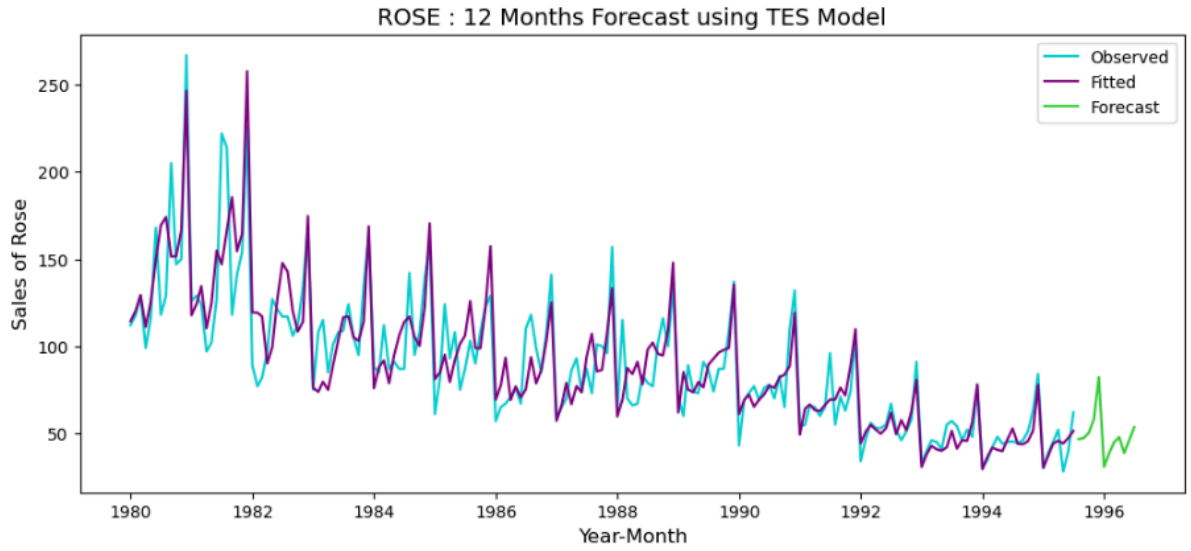


Fig-39 Original vs Forecast graph for Best model

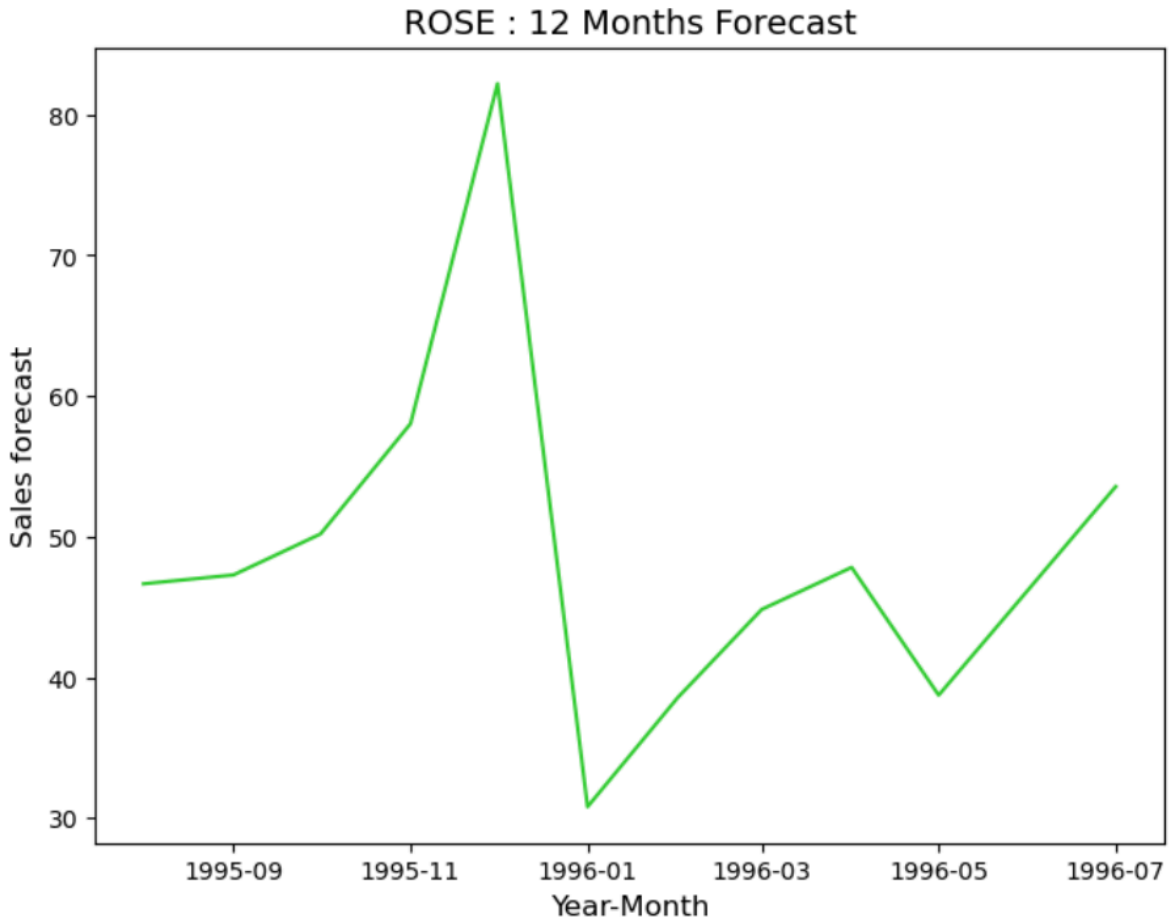


Fig-40 12 Months Forecast Line plot

1995-08-01	46.645790		
1995-09-01	47.277865		
1995-10-01	50.192393		
1995-11-01	58.032966		
1995-12-01	82.211767	count	12.000000
1996-01-01	30.793144	mean	48.739064
1996-02-01	38.536058	std	12.747211
1996-03-01	44.822234	min	30.793144
1996-04-01	47.814473	25%	43.298672
1996-05-01	38.727986	50%	46.961827
1996-06-01	46.255071	75%	51.034051
1996-07-01	53.559025	max	82.211767
Freq: MS, dtype: float64		dtype: float64	

Table-18 shows the Future forecast results and Summary statistics.

6. Actionable Insights & Recommendations:

Based on the provided data and analysis, here are some actionable insights and recommendations for ABC Estate Wine to address the low demand for Rose wine and capitalise on seasonal trends:

1. Capitalise on the Holiday Season

- **Increase Inventory**: Stock up on Rose wine in anticipation of the rising sales and December peak. Ensure sufficient inventory to meet the expected demand of 82 units in December.
- **Pre-Holiday Marketing Push (Aug-Oct)**: Launch targeted marketing campaigns and special offers to attract new customers, particularly first-time wine drinkers and those open to different brands. Highlight the festive appeal of Rose wine.

2. Address Long-Term Decline

- **Data Analysis**: Conduct further data analysis to understand the reasons behind the long-term decline in sales since 1980. Identify any shifts in consumer preferences, market trends, or competitive actions that may have contributed to this decline.
- **Customer Feedback**: Gather feedback from existing customers to understand their perceptions and preferences regarding Rose wine. Use this information to inform strategic decisions.

3. Rebranding & Innovation

- **Consider Rebranding**: Explore rebranding the existing Rose wine with a fresh image. This could include new packaging, a new label design, or even a collaboration with a new winemaker to create a unique blend.
- **Product Innovation**: Introduce new variants or limited-edition releases of Rose wine to generate excitement and attract attention. Experiment with different flavors or ageing techniques to differentiate the product.

4. Marketing & Promotions

- **Seasonal Promotions**: Offer special promotions and discounts during the holiday season to boost sales. Consider bundling Rose wine with other popular products or creating holiday gift sets.
- **Social Media Campaigns**: Leverage social media platforms to create buzz around Rose wine. Share engaging content, such as recipes, pairing suggestions, and customer testimonials, to increase visibility and attract new customers.

5. Evaluate Sales Performance Post-Holiday Season

- **Assess Sales Trends**: After the December peak, evaluate the overall sales trend. If there is a positive trend, continue with the existing Rose wine variant and marketing strategies.
- **Adjust Strategies**: If sales do not improve, consider adjusting marketing strategies or exploring new distribution channels to reach a broader audience.

By implementing these recommendations, ABC Estate Wine can address the low demand for Rose wine, capitalise on seasonal spikes, and work towards reversing the long-term decline in sales.

NOW WE DO THE SAME FOR THE SPARKLING DATASET

7. Exploratory Data Analysis and Data Pre-Processing for Sparkling Dataset:

From the below Table-19 we can see the first and last five rows for the Sparkling dataset.

	YearMonth	Sparkling
0	1980-01	1686
1	1980-02	1591
2	1980-03	2304
3	1980-04	1712
4	1980-05	1471

	YearMonth	Sparkling
182	1995-03	1897
183	1995-04	1862
184	1995-05	1670
185	1995-06	1688
186	1995-07	2031

Table-19 FIrst and last Five rows of Sparkling dataset.

Now, we will use the info function to get the number of variables and their data types.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 187 entries, 0 to 186
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   YearMonth  187 non-null    object
 1   Sparkling  187 non-null    int64
dtypes: int64(1), object(1)
memory usage: 3.1+ KB
```

Table-20 Information of Sparkling dataset.

Using Method-2 we read the original dataset as a Time series dataset by using the pandas function setting parse_date and squeeze as True as well as setting index_col as zero. And we get the first and last five rows of our new time series dataset for rose wine as shown in table-21.

```
DatetimeIndex(['1980-01-31', '1980-02-29', '1980-03-31', '1980-04-30',
               '1980-05-31', '1980-06-30', '1980-07-31', '1980-08-31',
               '1980-09-30', '1980-10-31',
               ...
               '1994-10-31', '1994-11-30', '1994-12-31', '1995-01-31',
               '1995-02-28', '1995-03-31', '1995-04-30', '1995-05-31',
               '1995-06-30', '1995-07-31'],
              dtype='datetime64[ns]', length=187, freq='ME')
```

Table-21 Timestamp creation for sparkling dataset

After that we found that there are two null/empty values in the rose dataset using is null function. And we found that there are zero null/empty values.

Now, we use the describe function on the rose data set and obtain the five required summary values like min, max, std, variance at 25,50 and 75 percent as well as count as shown in Table-22.

Data Description for Sparkling Dataset:

```
[399]:

count     187.000000
mean     2402.417112
std      1295.111540
min      1070.000000
25%      1605.000000
50%      1874.000000
75%      2549.000000
max      7242.000000
Name: Sparkling, dtype: float64
```

Table-22 Description of sparkling dataset

Now we plot the boxplot for yearly sales of rose wine as seen in Fig-41.

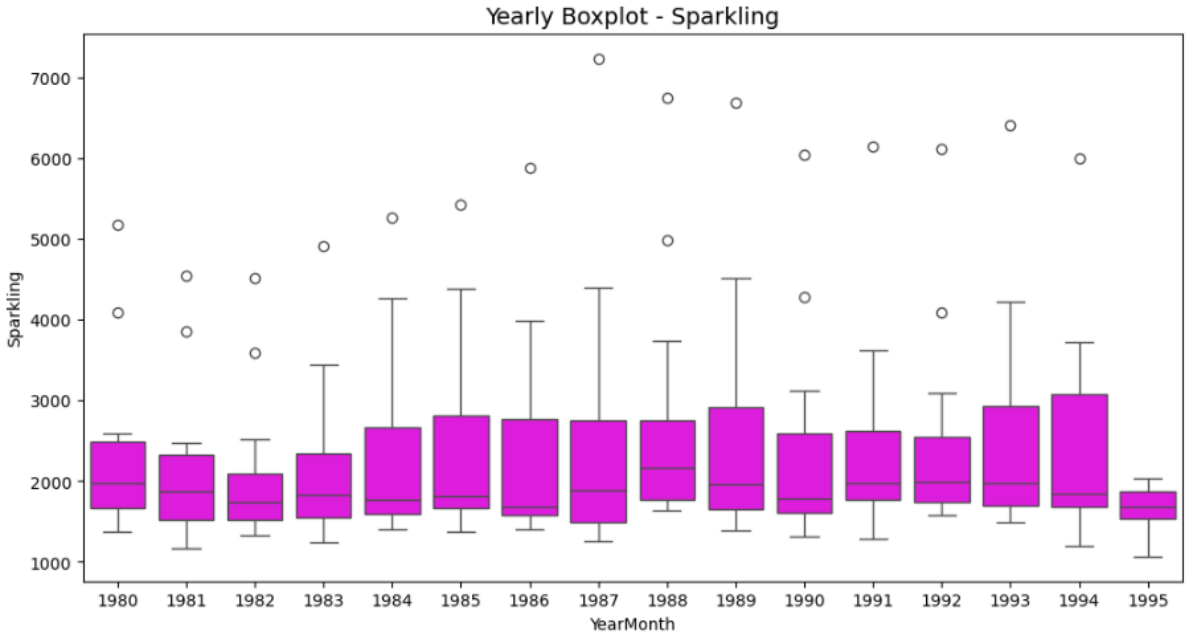


Fig-41 Boxplot Yearly sales for sparkling dataset

We will also look at the boxplot of monthly sales of rose wine over the years combined as seen in Fig-42.

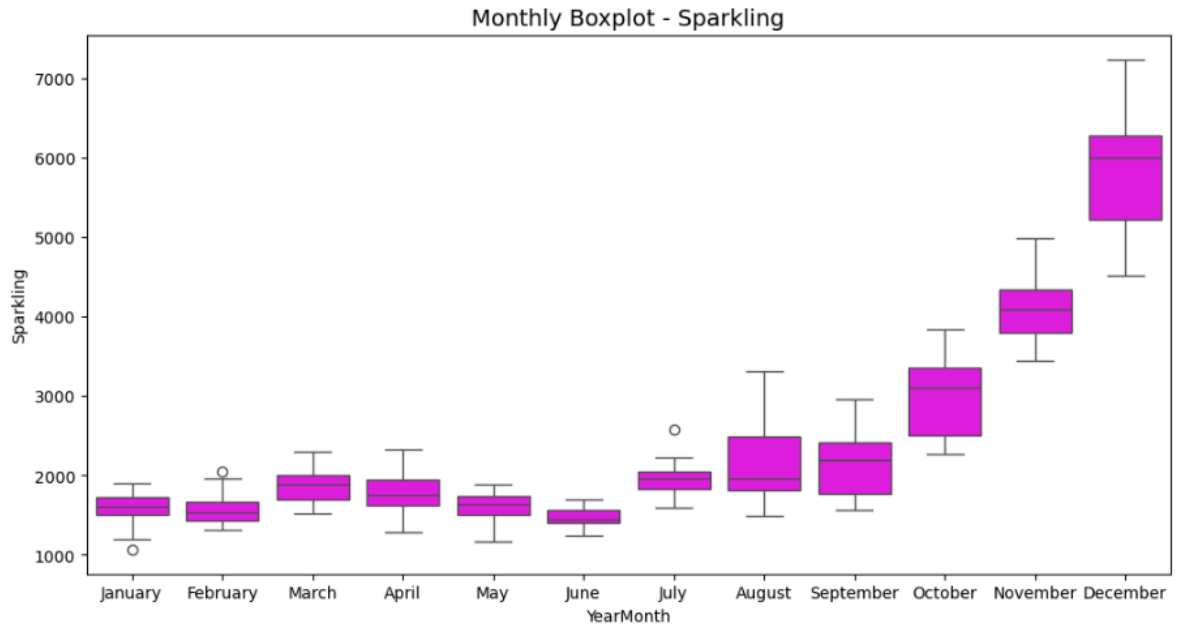


Fig-42 Boxplot Monthly sales for sparkling dataset

As seen in below Fig-43 we can see the visualisation of mean and variance of monthly sales for all the years present in the dataset.

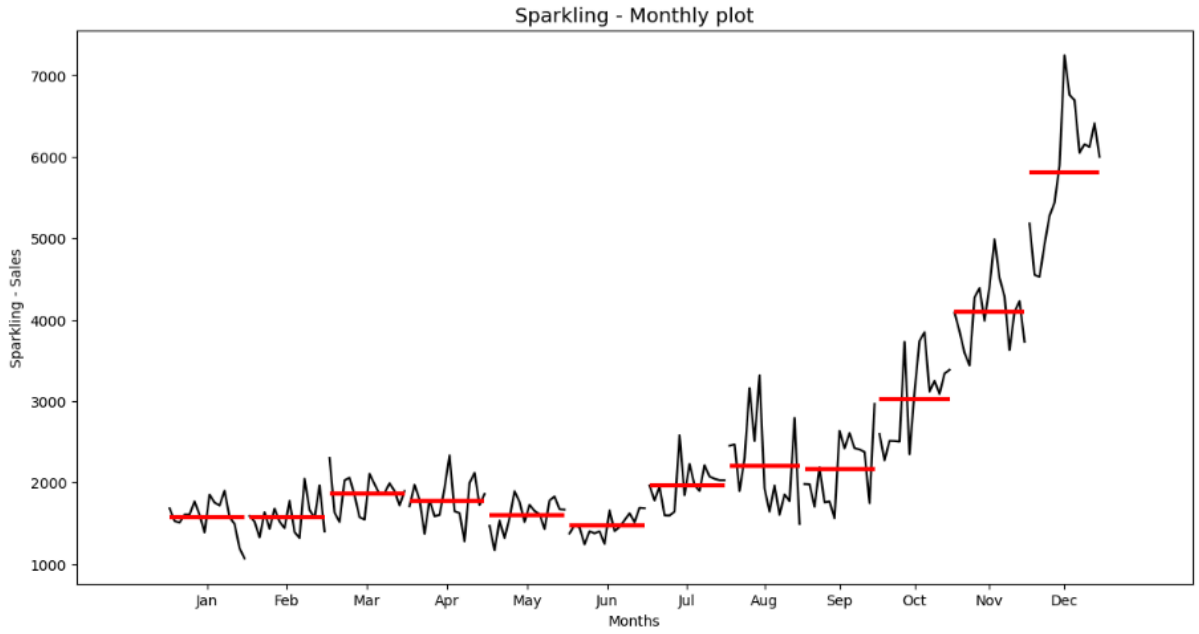


Fig-43 Mean and STD for Monthly sales of sparkling dataset

Using pivot table function we can see the sales of rose wine in each month for each year as observed in the below Table-23 and its line chart can be looked at Fig-44.

	Sparkling											
YearMonth	1	2	3	4	5	6	7	8	9	10	11	12
YearMonth												
1980	1686.0	1591.0	2304.0	1712.0	1471.0	1377.0	1966.0	2453.0	1984.0	2596.0	4087.0	5179.0
1981	1530.0	1523.0	1633.0	1976.0	1170.0	1480.0	1781.0	2472.0	1981.0	2273.0	3857.0	4551.0
1982	1510.0	1329.0	1518.0	1790.0	1537.0	1449.0	1954.0	1897.0	1706.0	2514.0	3593.0	4524.0
1983	1609.0	1638.0	2030.0	1375.0	1320.0	1245.0	1600.0	2298.0	2191.0	2511.0	3440.0	4923.0
1984	1609.0	1435.0	2061.0	1789.0	1567.0	1404.0	1597.0	3159.0	1759.0	2504.0	4273.0	5274.0
1985	1771.0	1682.0	1846.0	1589.0	1896.0	1379.0	1645.0	2512.0	1771.0	3727.0	4388.0	5434.0
1986	1606.0	1523.0	1577.0	1605.0	1765.0	1403.0	2584.0	3318.0	1562.0	2349.0	3987.0	5891.0
1987	1389.0	1442.0	1548.0	1935.0	1518.0	1250.0	1847.0	1930.0	2638.0	3114.0	4405.0	7242.0
1988	1853.0	1779.0	2108.0	2336.0	1728.0	1661.0	2230.0	1645.0	2421.0	3740.0	4988.0	6757.0
1989	1757.0	1394.0	1982.0	1650.0	1654.0	1406.0	1971.0	1968.0	2608.0	3845.0	4514.0	6694.0
1990	1720.0	1321.0	1859.0	1628.0	1615.0	1457.0	1899.0	1605.0	2424.0	3116.0	4286.0	6047.0
1991	1902.0	2049.0	1874.0	1279.0	1432.0	1540.0	2214.0	1857.0	2408.0	3252.0	3627.0	6153.0
1992	1577.0	1667.0	1993.0	1997.0	1783.0	1625.0	2076.0	1773.0	2377.0	3088.0	4096.0	6119.0
1993	1494.0	1564.0	1898.0	2121.0	1831.0	1515.0	2048.0	2795.0	1749.0	3339.0	4227.0	6410.0
1994	1197.0	1968.0	1720.0	1725.0	1674.0	1693.0	2031.0	1495.0	2968.0	3385.0	3729.0	5999.0
1995	1070.0	1402.0	1897.0	1862.0	1670.0	1688.0	2031.0	NaN	NaN	NaN	NaN	NaN

Table-23 Pivot table for sparkling wine

Monthly Wine sales across years for Sparkling:

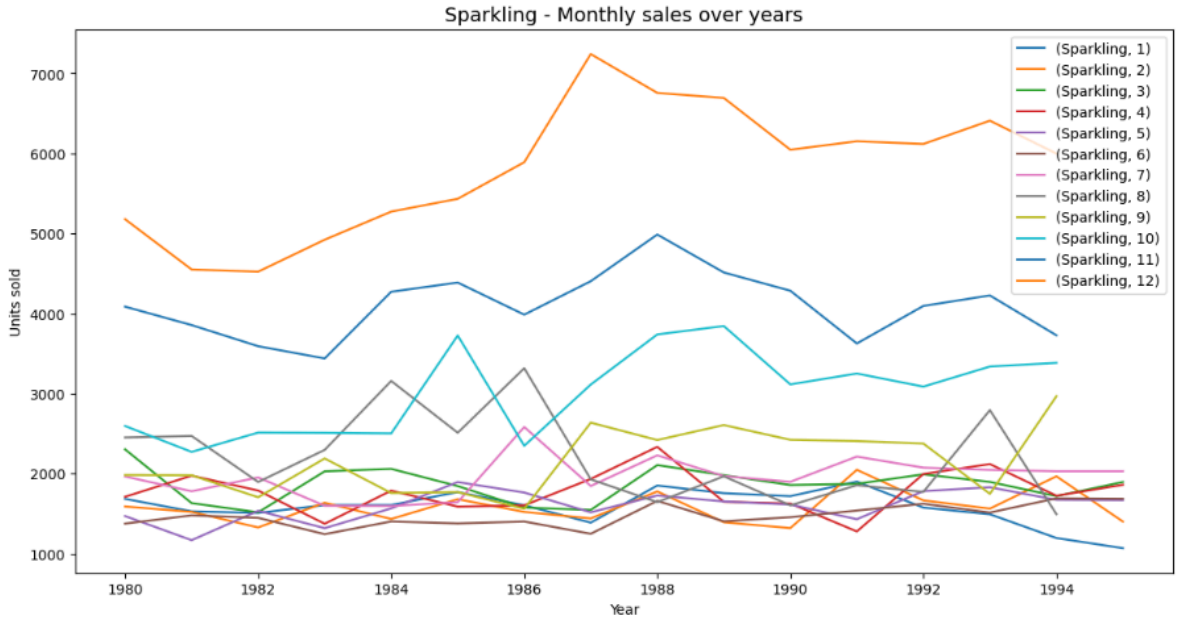


Fig-44 Line plot Sales for sparkling Wine

Average rose sales and sales in percentage can be seen below FIg-45.

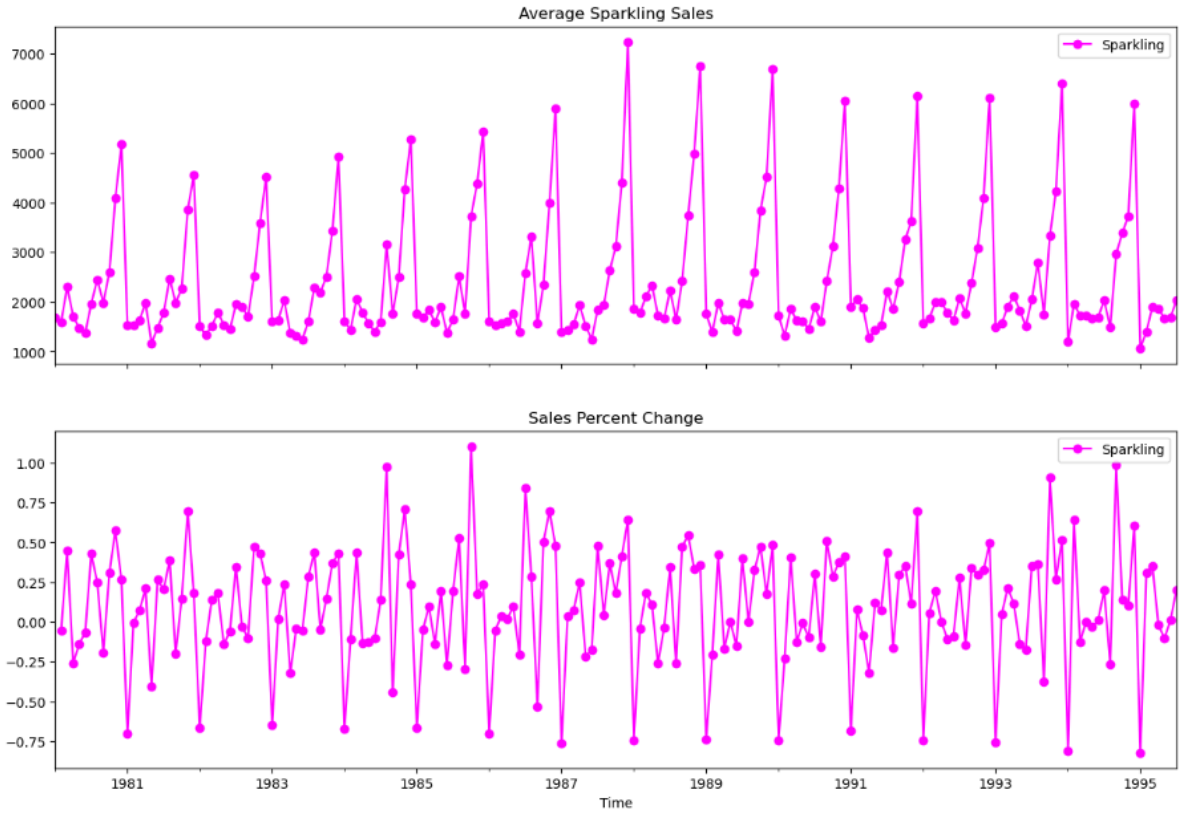
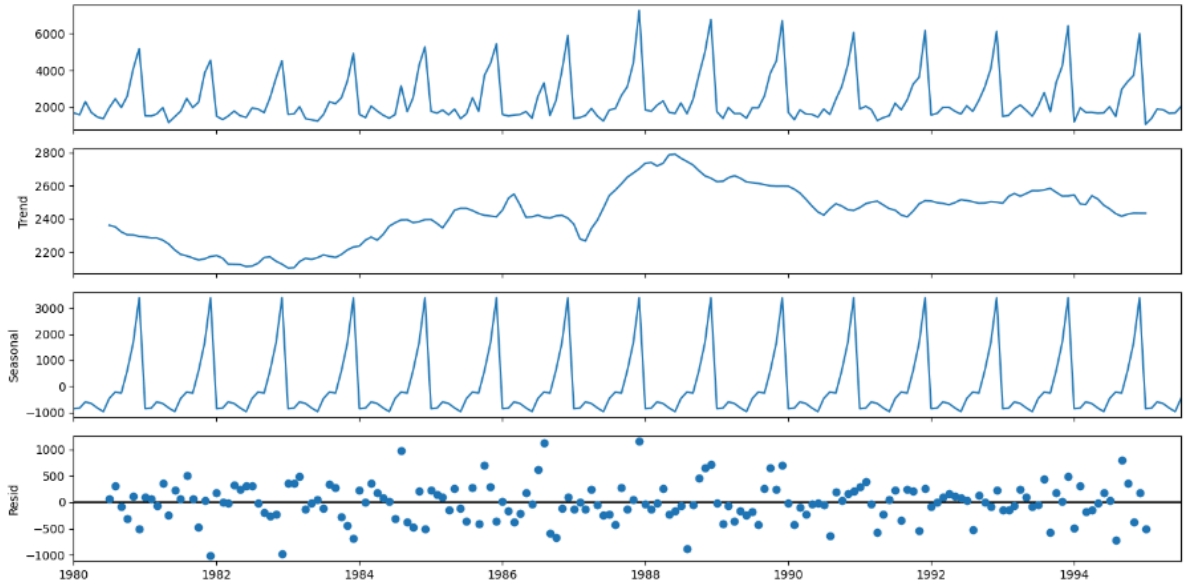


Fig-45 Line plot Average Sales for sparkling Wine

Now, we will decompose the Rose time series into trends, seasonality and residuals using multiplicative decomposition model as we have observed from the above tables and figures that the dataset contains trend, seasonality and its decreasing in multiplicative manner and the decomposed model can be observed in Fig-46.

Decomposition of Sparkling Time Series with additive Seasonality:



Decomposition of Sparkling Time Series with multiplicative Seasonality:

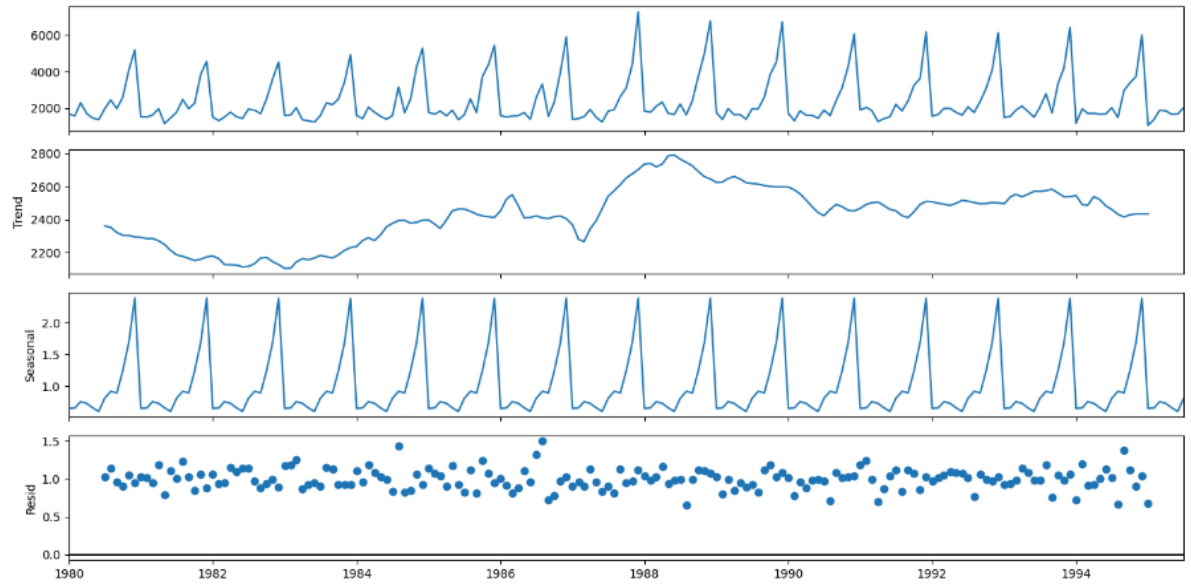


Fig-46 Decomposition of sparkling dataset

Now, we will split the rose dataset into a test-train dataset and, since it's a time series model we will split the dataset in such a way that all the observation before the year 1991 is in training dataset and from 1991 year all observation shall be in test dataset. And as seen in Table-24 below we get the fist and last five rows of both the training and testing dataset.The split can be observed in the Fig-47 also.

First few rows of Training Data:

YearMonth	Sparkling
1980-01-01	1686
1980-02-01	1591
1980-03-01	2304
1980-04-01	1712
1980-05-01	1471

Last few rows of Training Data:

YearMonth	Sparkling
1990-08-01	1605
1990-09-01	2424
1990-10-01	3116
1990-11-01	4286
1990-12-01	6047

First few rows of Test Data:

YearMonth	Sparkling
1991-01-01	1902
1991-02-01	2049
1991-03-01	1874
1991-04-01	1279
1991-05-01	1432

Last few rows of Test Data:

YearMonth	Sparkling
1995-03-01	1897
1995-04-01	1862
1995-05-01	1670
1995-06-01	1688
1995-07-01	2031

Table-24 First and last five rows for Train and test dataset

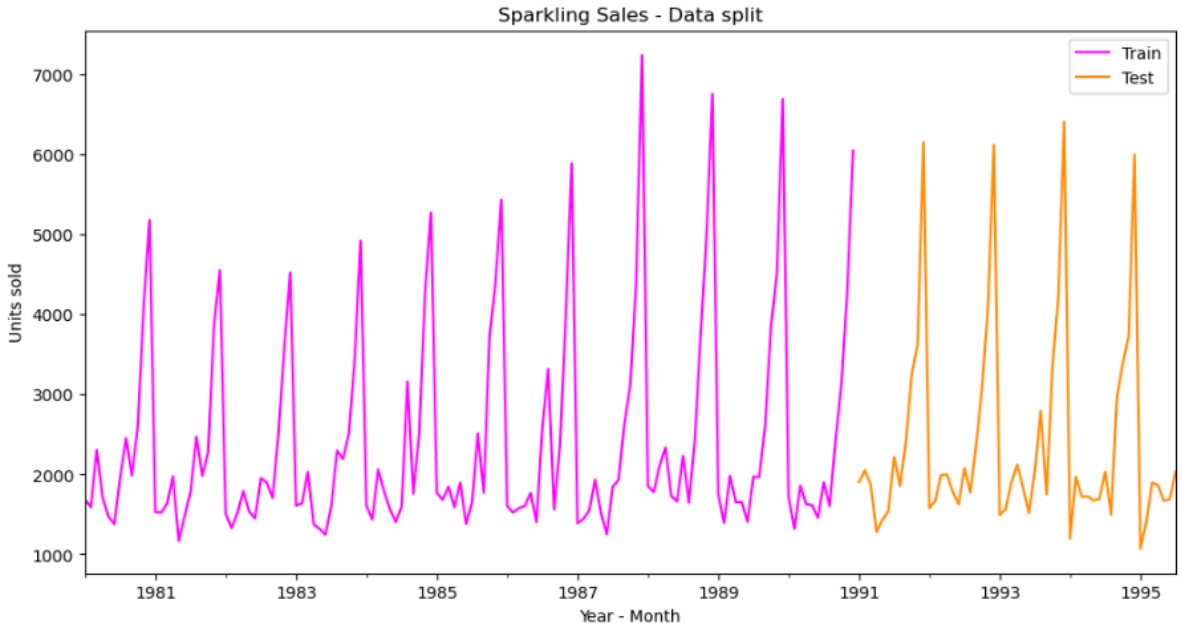


Fig-47 Split for Test-Train dataset

8. Model Building - Original Data:

Linear Regression Model:

Now, We built our first Time forecasting model using LinearRegression model and we obtained the model as seen in Fig-48.

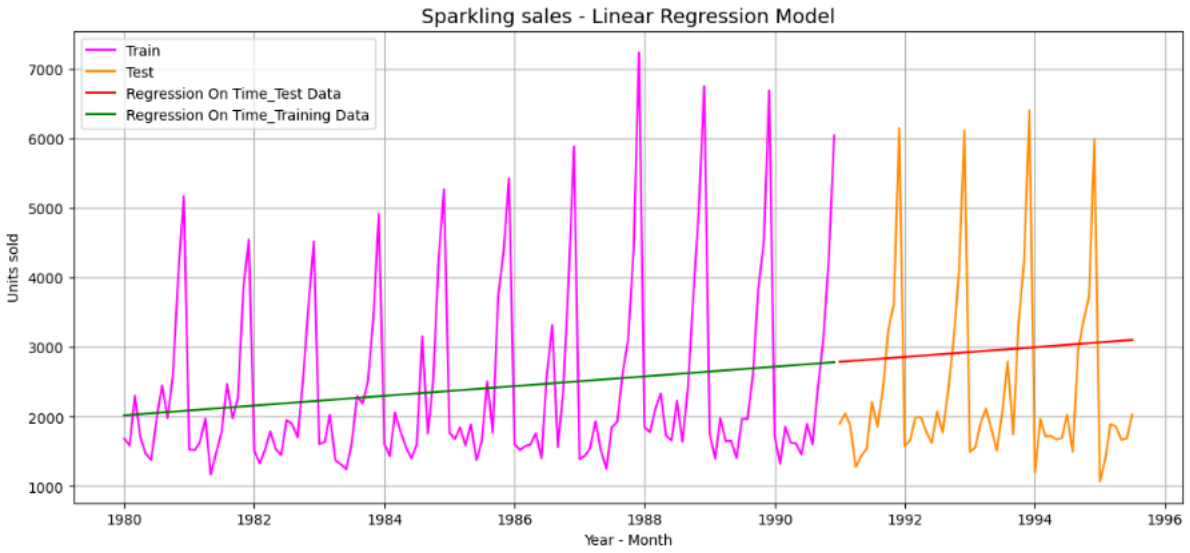


Fig-48 Linear Regression Model

We got the RMSE value on the test data as 15.278 and it has been stored in a dataframe named rose_resultDf.

Naive Model:

Now, we use Naive Bayes model for forecasting and obtained the below model as seen in FIg-49.

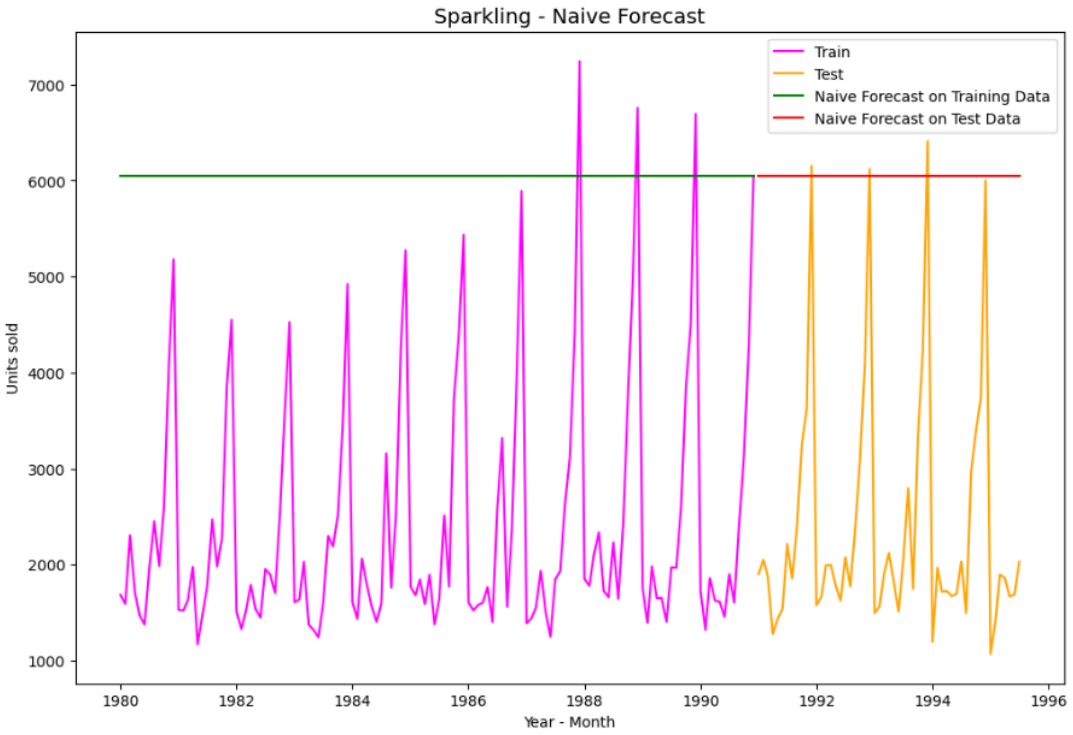


Fig-49 Naive Model

THe RMSE for Naive Model is 76.745 and has been stored in the rose result dataset.

Simple Average Model:

The third model we will make is using Simple Average and obtained the model as seen in Fig-50. THe RMSE for Simple Average is 15.774.

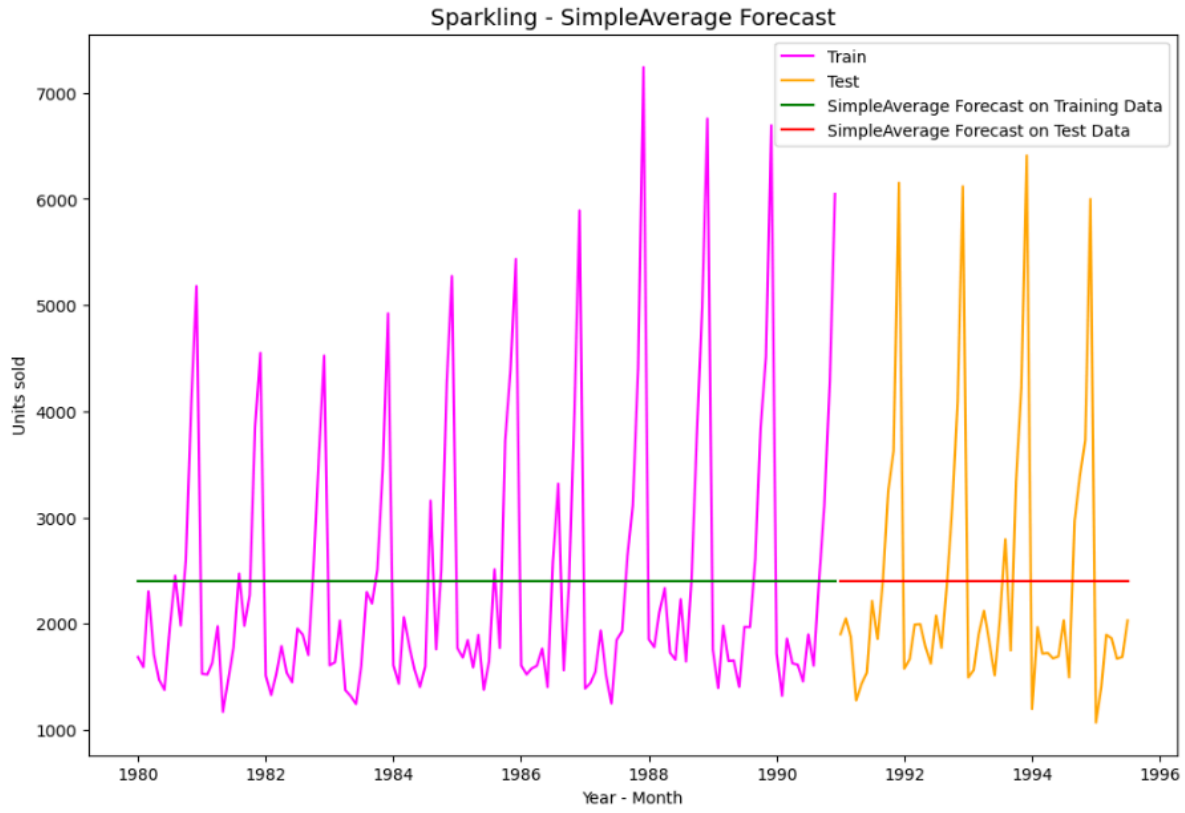


Fig-50 Simple Average model

Moving Average Model:

Using Moving Average we will calculate all the rolling means or trailing moving averages for different-different intervals. In this the best interval will be determined by maximum accuracy.We calculated and plotted the moving average for the rose dataset as seen in Table-25 and Fig-51.

YearMonth	Sparkling	Spark_Trailing_2	Spark_Trailing_4	Spark_Trailing_6	Spark_Trailing_9
1980-01-01	1686	NaN	NaN	NaN	NaN
1980-02-01	1591	1638.5	NaN	NaN	NaN
1980-03-01	2304	1947.5	NaN	NaN	NaN
1980-04-01	1712	2008.0	1823.25	NaN	NaN
1980-05-01	1471	1591.5	1769.50	NaN	NaN

Table-25 Moving Average Values

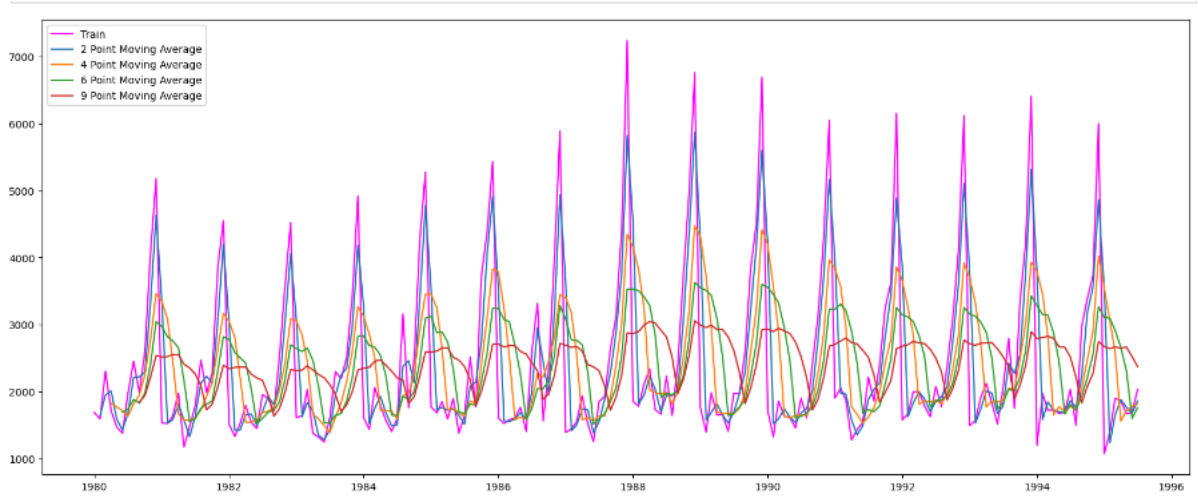


Fig-51 Moving Average model

After this we will apply all the moving averages on the test and train dataset and the line plot as shown in Fig-52. The RMSE values obtained for each rolling means are as follows:

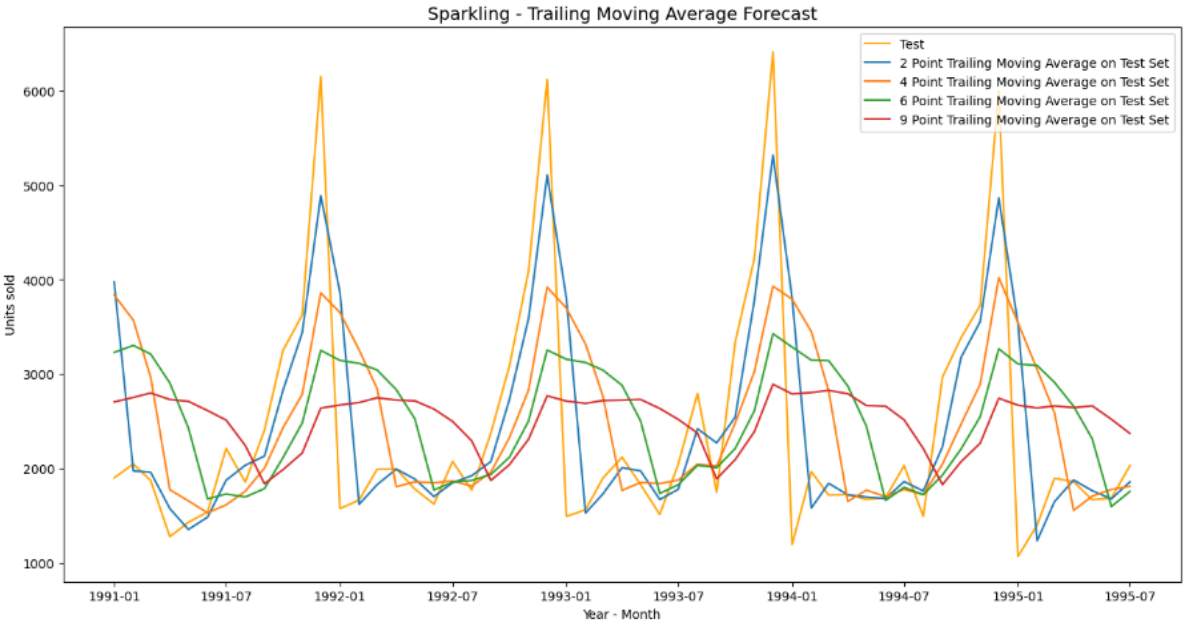


Fig-52 Moving Average for Test-train dataset

Before moving forward we will plot a chart for all models to see what we have got up till now as seen in Fig-53. It can be said that we are creating better models but not accurate one's that can be used for forecasting.

Fig-53 Line Chart for all Models

Simple Exponential Smoothing Model(Single,Double,Triple):

The next model we will create is using Simple Exponential Smoothing in Autofit/optimised and Manualfit/Iterative. The Fig-54 shows the autofit of SES and fig-55 shows the manual fit of SES.

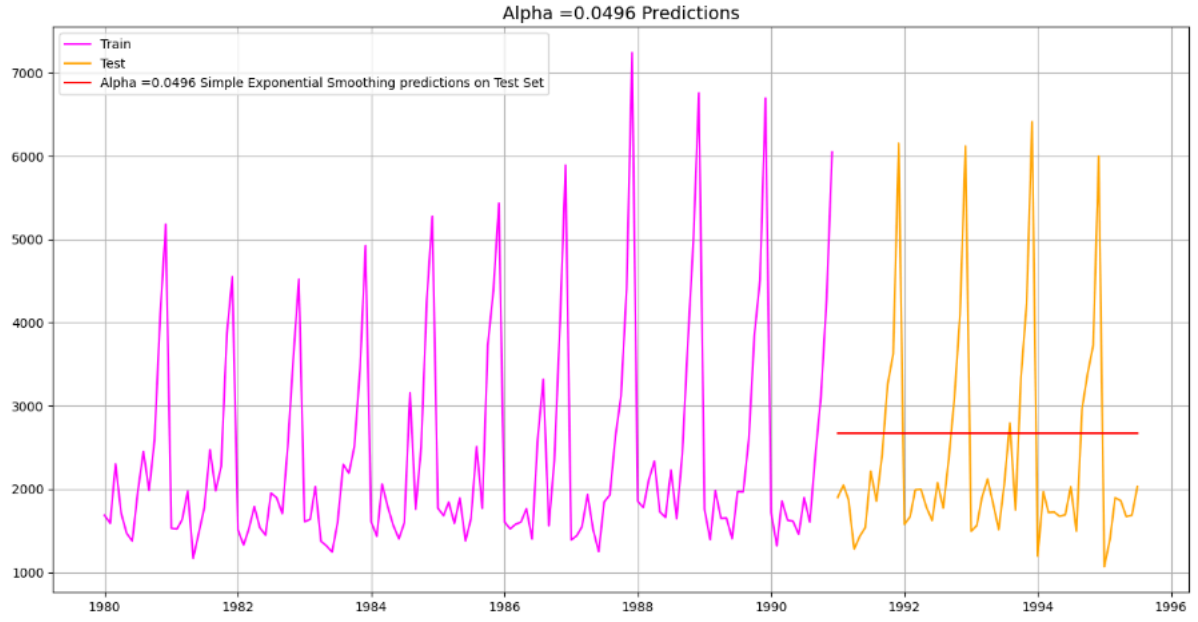


Fig-54 Optimised SES Model

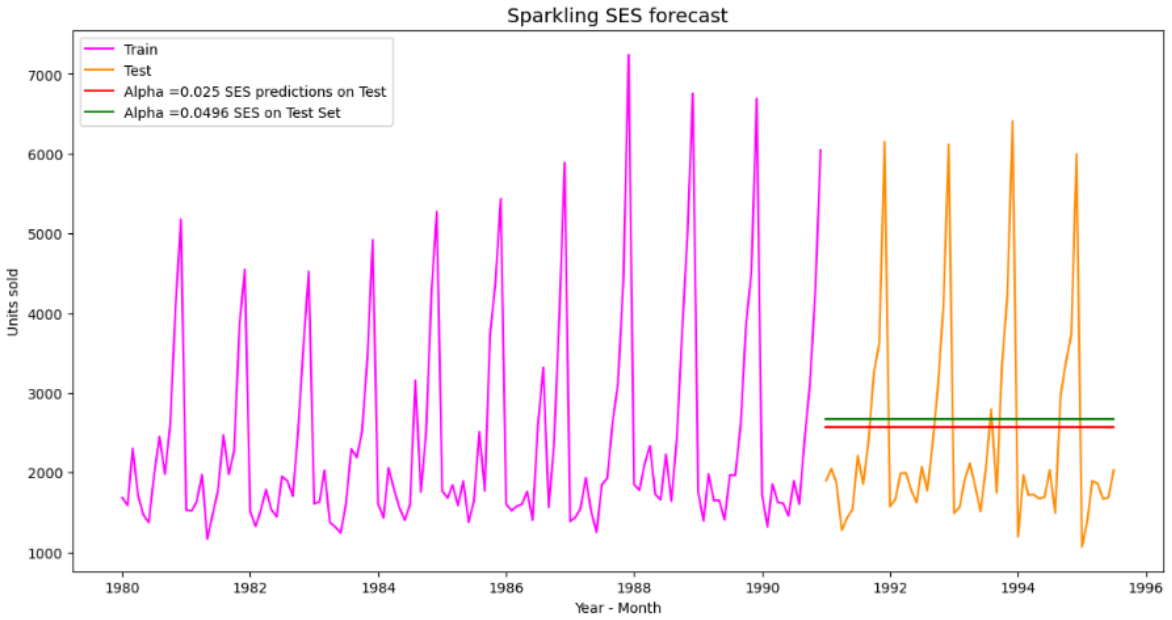


Fig-55 Iterative SES Model

Now we will do Double Exponential Smoothing in Autofit and Manual fit. From Fig-56 we can see DES in Autofit and Fig-57 shows the Manual fit for DES.

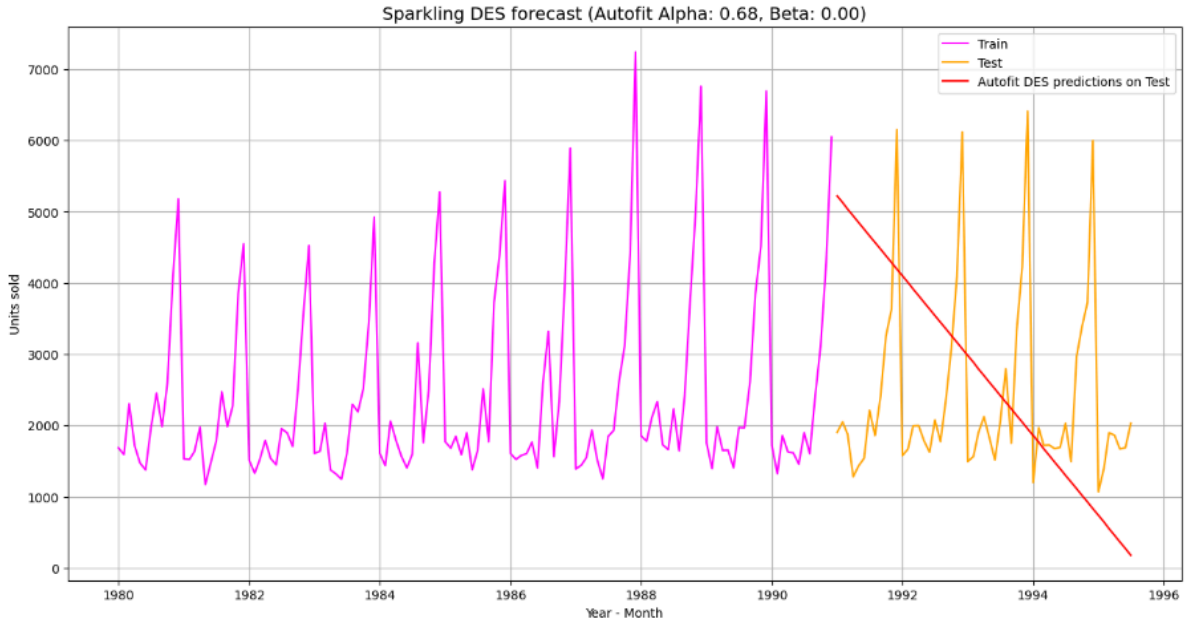


Fig-56 Optimised DES Model

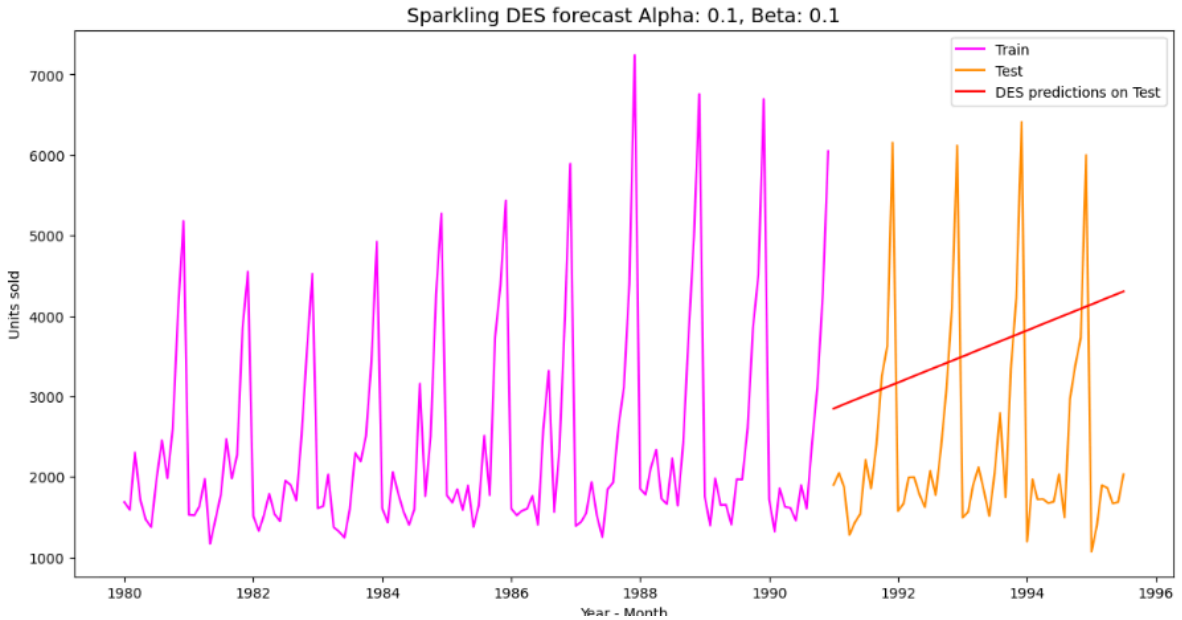


Fig-57 Iterative DES Model

Finally we will go for Triple Exponential Smoothing same as before for both Optimised and Iterative Models. The Fig-58 shows the Autofit/optimised model of TES and Fig-59 shows the Iterative/Manualfit model of TES.

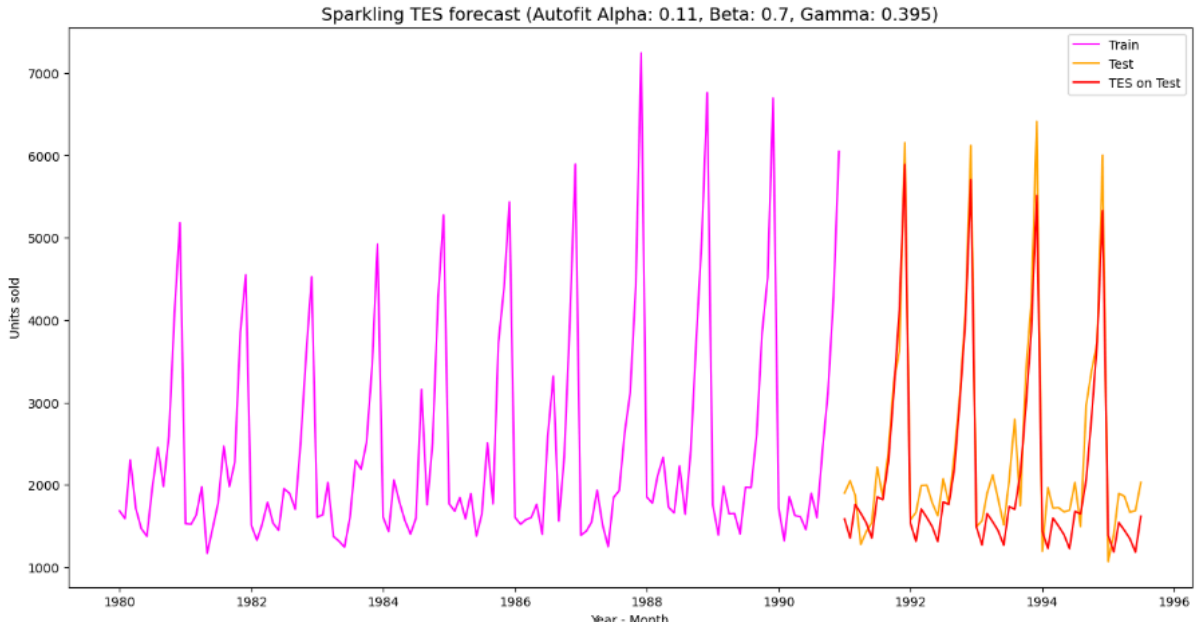


Fig-58 Optimised TES Model

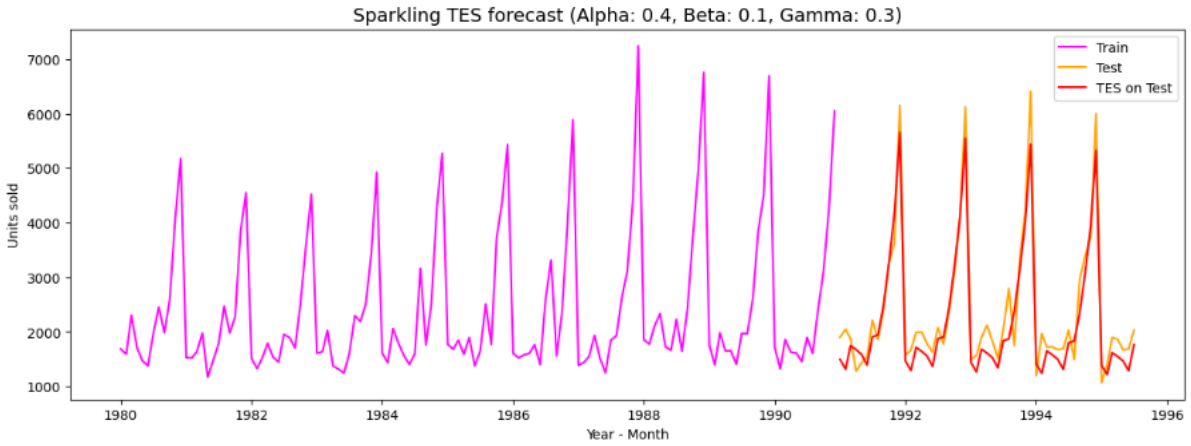


Fig-59 Iterative TES Model

The Table-26 and Fig-60 shows the RMSE values in sorted order for the models we have used for forecasting up till now in the below:

	Test RMSE
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	396.598057
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	404.286809
2 point TMA	813.400684
4 point TMA	1156.589694
SimpleAverage	1275.081804
6 point TMA	1283.927428
Alpha=0.025,SES iterative	1286.248846
Alpha=0.0496, SES Optimized	1304.927405
9 point TMA	1346.278315
RegressionOnTime	1389.135175
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
NaiveModel	3864.279352

Table-26 Sorted Test RMSE value

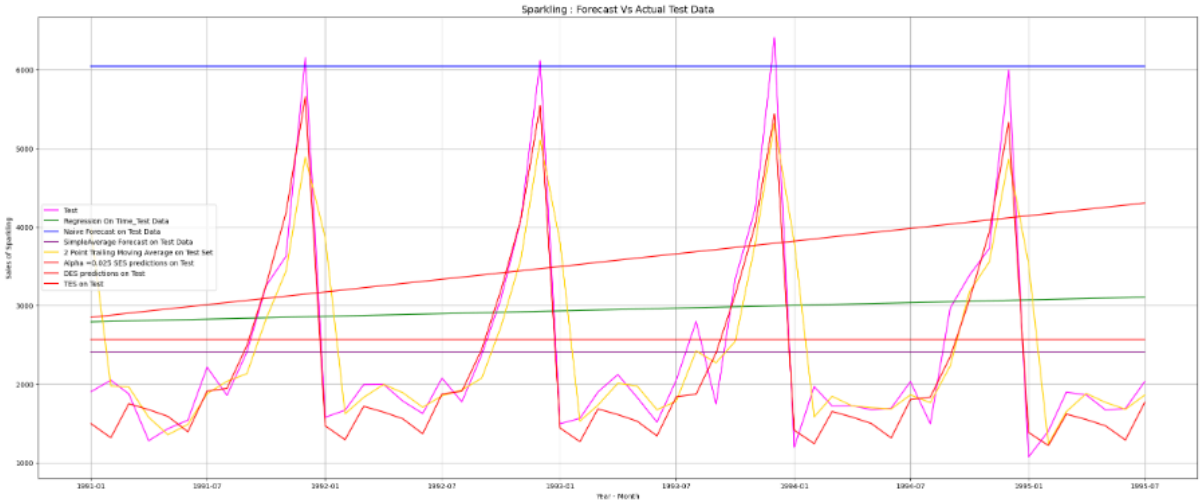
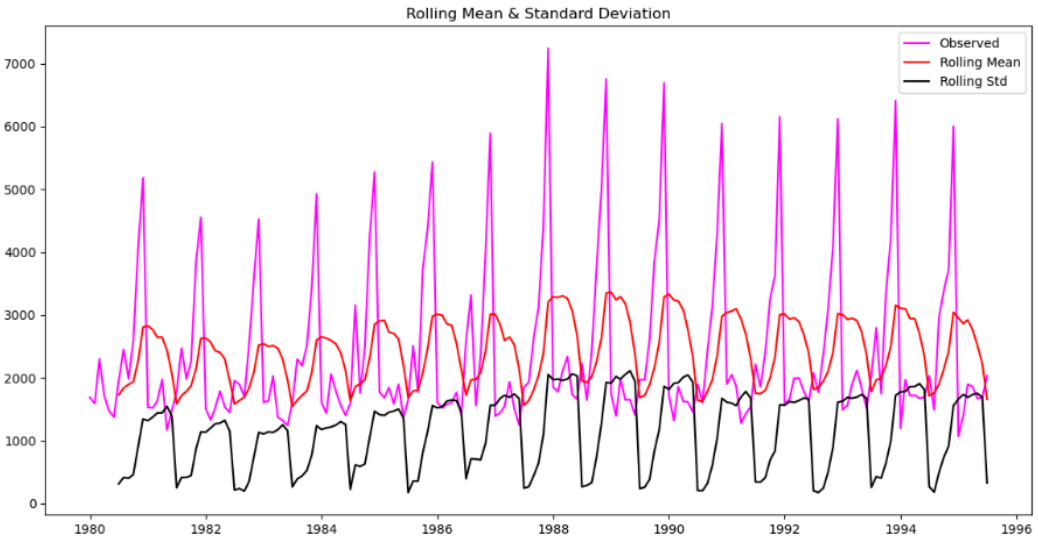


Fig-60 Forecast v/s Actual values of all models

9. Check for Stationarity:

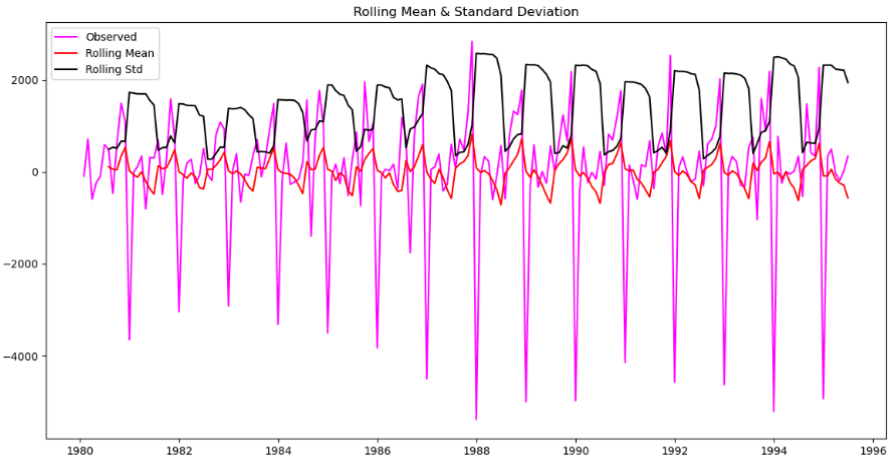
We will check whether the dataset is stationary or not. By applying Augmented Dickey Fuller test it has been found out that the rose dataset is non-stationary as seen in Fig-61.



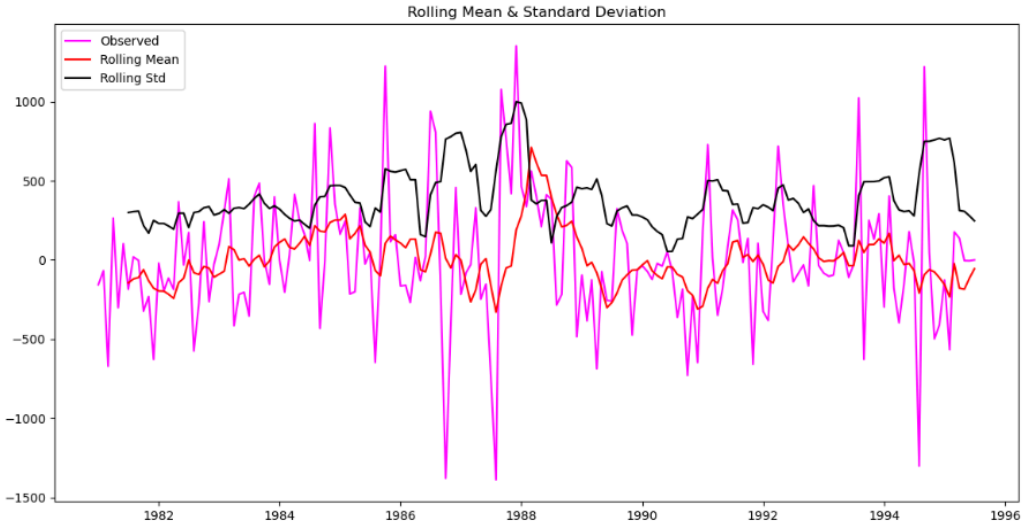

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.360497
p-value                          0.601061
#Lags Used                      11.000000
Number of Observations Used    175.000000
Critical Value (1%)             -3.468280
Critical Value (5%)             -2.878202
Critical Value (10%)            -2.575653
dtype: float64
```

Fig-61 Rolling mean and STD for Original sparkling dataset

To make it a stationary dataset various combinations of differentials intervals are used as seen in Fig-62 and Fig-63. In Fig-62 the Graphs are obtained without log transformation and one of them has a difference of one order. While in Fig-63 it is the same as fig-62 but with log transformation.

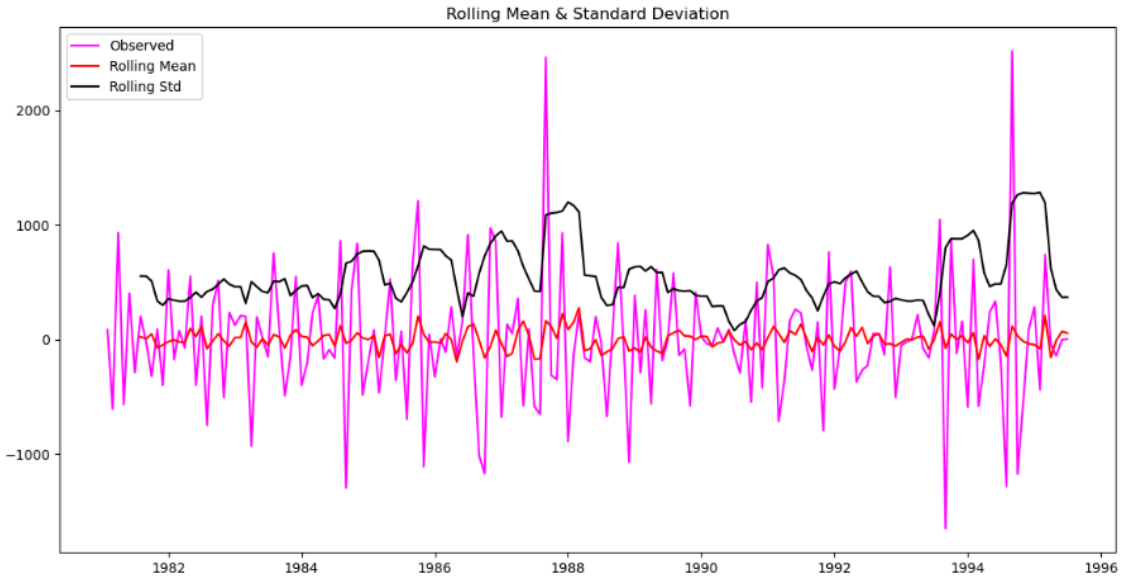



```
Results of Dickey-Fuller Test:
Test Statistic                -45.050301
p-value                         0.000000
#Lags Used                     10.000000
Number of Observations Used    175.000000
Critical Value (1%)            -3.468280
Critical Value (5%)            -2.878202
Critical Value (10%)           -2.575653
dtype: float64
```

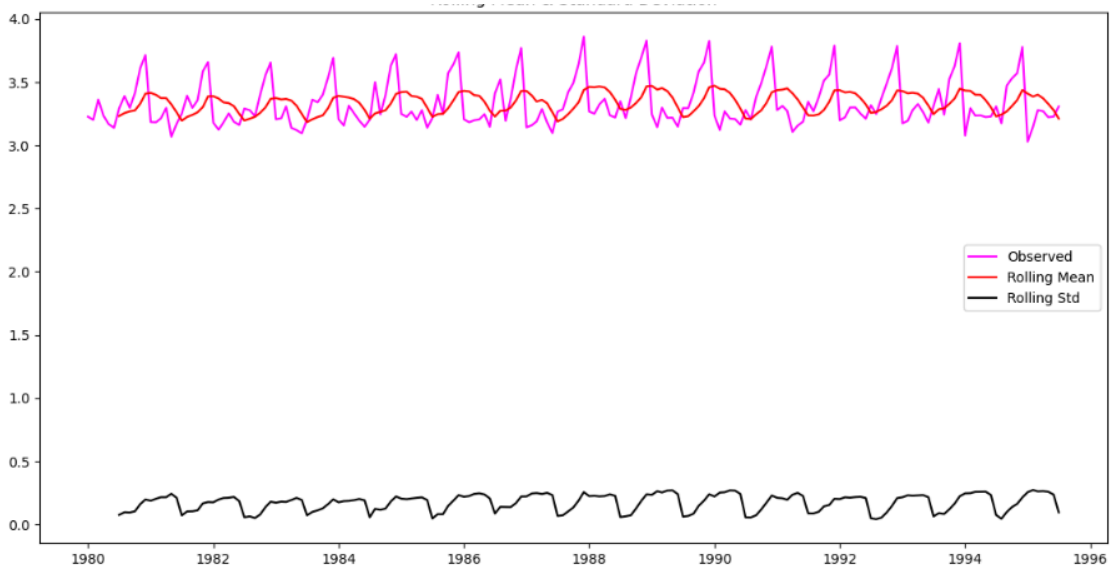



```
Results of Dickey-Fuller Test:
Test Statistic                -4.460165
p-value                        0.000232
#Lags Used                    11.000000
Number of Observations Used   163.000000
Critical Value (1%)           -3.471119
Critical Value (5%)           -2.879441
Critical Value (10%)          -2.576314
dtype: float64
```

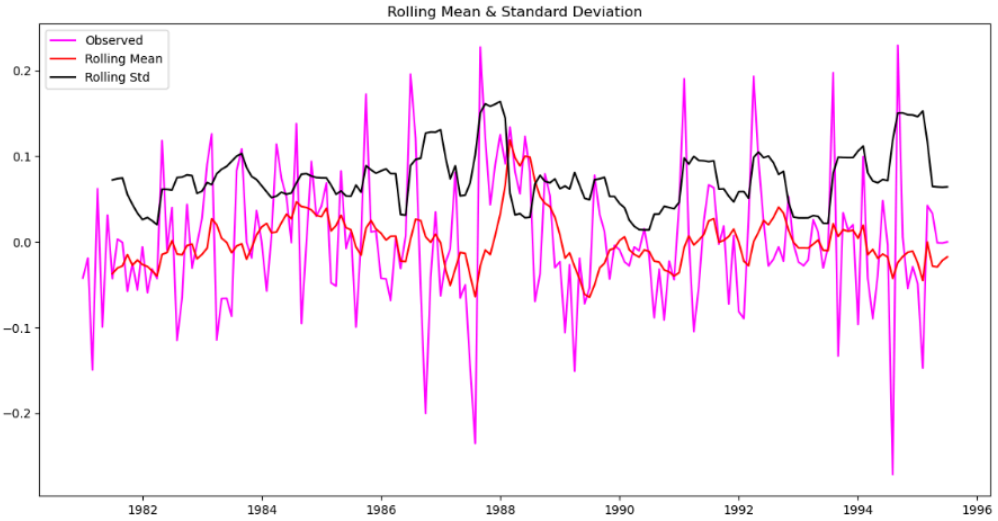
Fig-62 Rolling mean and STD without log transformation




```
Results of Dickey-Fuller Test:
Test Statistic                  -5.113533
p-value                          0.000013
#Lags Used                      11.000000
Number of Observations Used    162.000000
Critical Value (1%)             -3.471374
Critical Value (5%)             -2.879552
Critical Value (10%)            -2.576373
dtype: float64
```



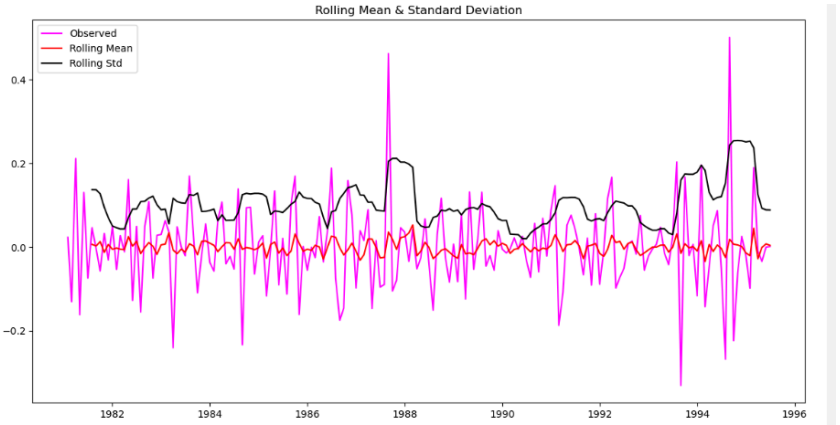

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.749630
p-value                          0.405740
#Lags Used                      11.000000
Number of Observations Used    175.000000
Critical Value (1%)             -3.468280
Critical Value (5%)             -2.878202
Critical Value (10%)            -2.575653
dtype: float64
```




```
Results of Dickey-Fuller Test:
Test Statistic                  -5.183811
p-value                          0.000009
#Lags Used                      11.000000
Number of Observations Used    163.000000
Critical Value (1%)             -3.471119
Critical Value (5%)             -2.879441
Critical Value (10%)            -2.576314
dtype: float64
```

Fig-63 Rolling mean and STD with log transformation

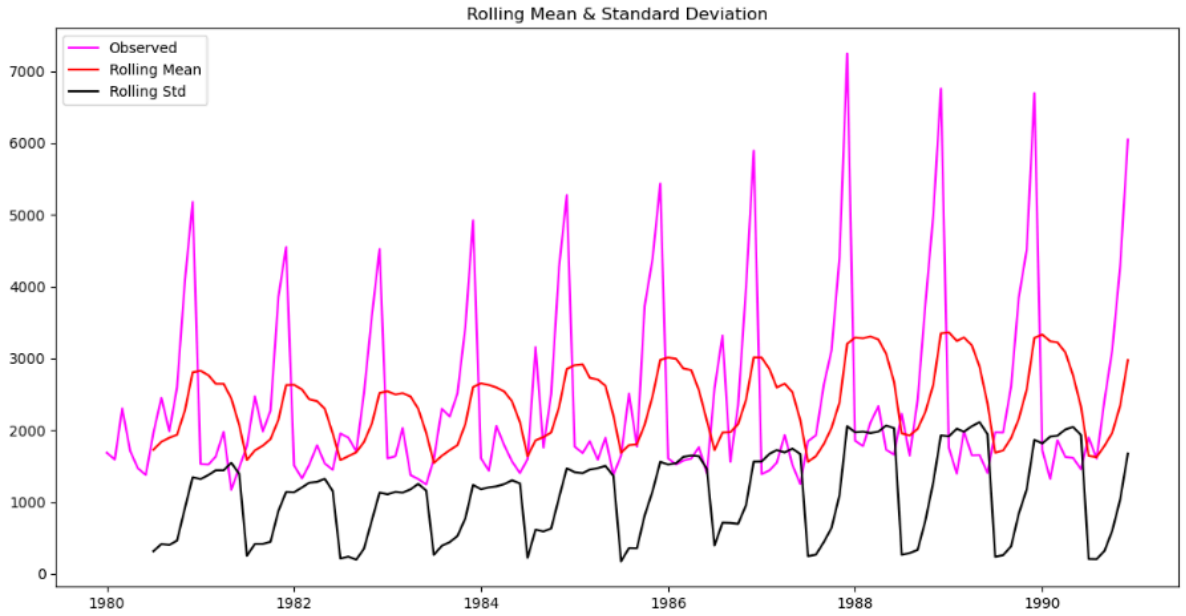
As per Fig-64 we can see that ADF is also done with log transformation of train data with differing seasonal order of 12.




```
Results of Dickey-Fuller Test:
Test Statistic                  -5.254601
p-value                          0.000007
#Lags Used                      12.000000
Number of Observations Used    161.000000
Critical Value (1%)             -3.471633
Critical Value (5%)             -2.879665
Critical Value (10%)            -2.576434
dtype: float64
```

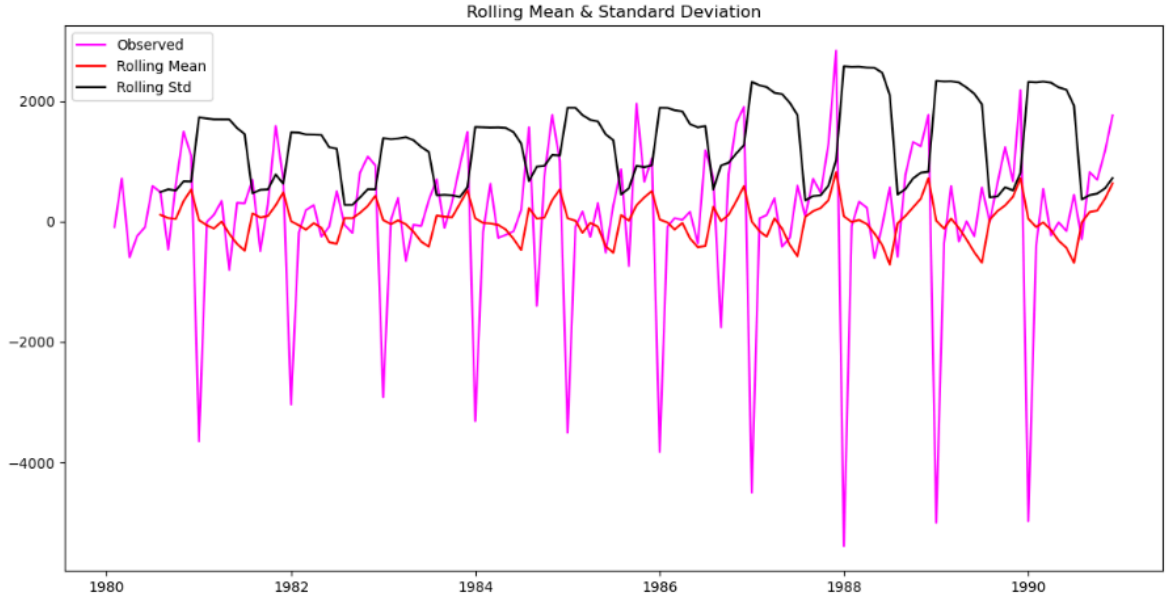
Fig-64 Rolling mean and STD with log transformation and Differential =12

And obtained that with differencing order one when applied is the most stationary one out of all of them and can be compared with the original dataset rolling mean and standard deviation as seen in Fig-65 and Fig-66 respectively.




```
Results of Dickey-Fuller Test:
Test Statistic                  -1.208926
p-value                          0.669744
#Lags Used                      12.000000
Number of Observations Used    119.000000
Critical Value (1%)             -3.486535
Critical Value (5%)             -2.886151
Critical Value (10%)            -2.579896
dtype: float64
```

Fig-65 Rolling mean and STD for train dataset




```
Results of Dickey-Fuller Test:
Test Statistic                -8.005007e+00
p-value                        2.280104e-12
#Lags Used                     1.100000e+01
Number of Observations Used    1.190000e+02
Critical Value (1%)           -3.486535e+00
Critical Value (5%)           -2.886151e+00
Critical Value (10%)          -2.579896e+00
dtype: float64
```

Fig-66 Rolling mean and STD with stationary train dataset

10. Model Building - Stationary Data:

Now, we will find and plot the Autocorrelation and partial Autocorrelation for both the original rose dataset and the training dataset as seen below in Fig-67 and Fig-68 respectively.

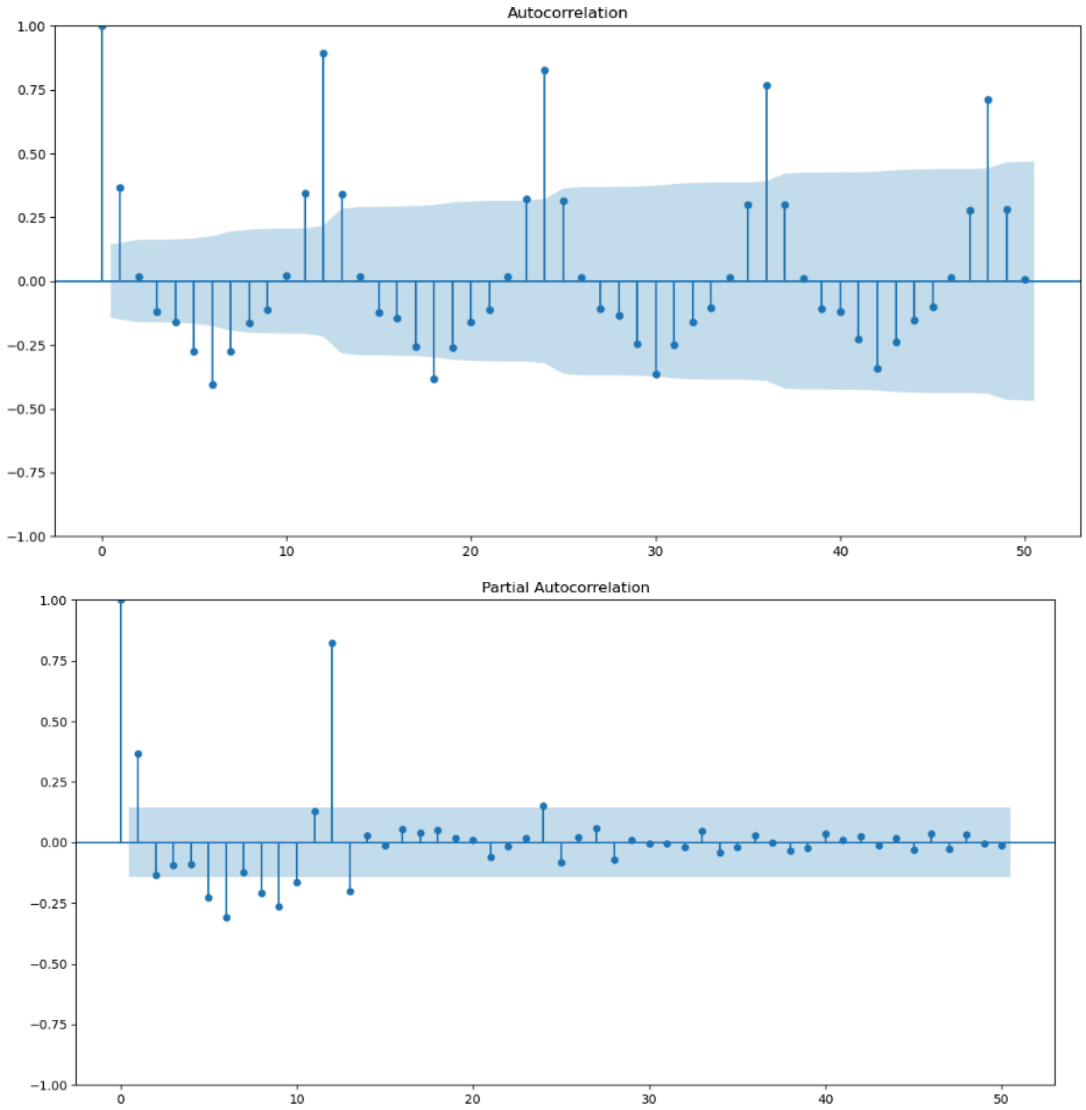


Fig-67 ACF and PACF for Train Dataset

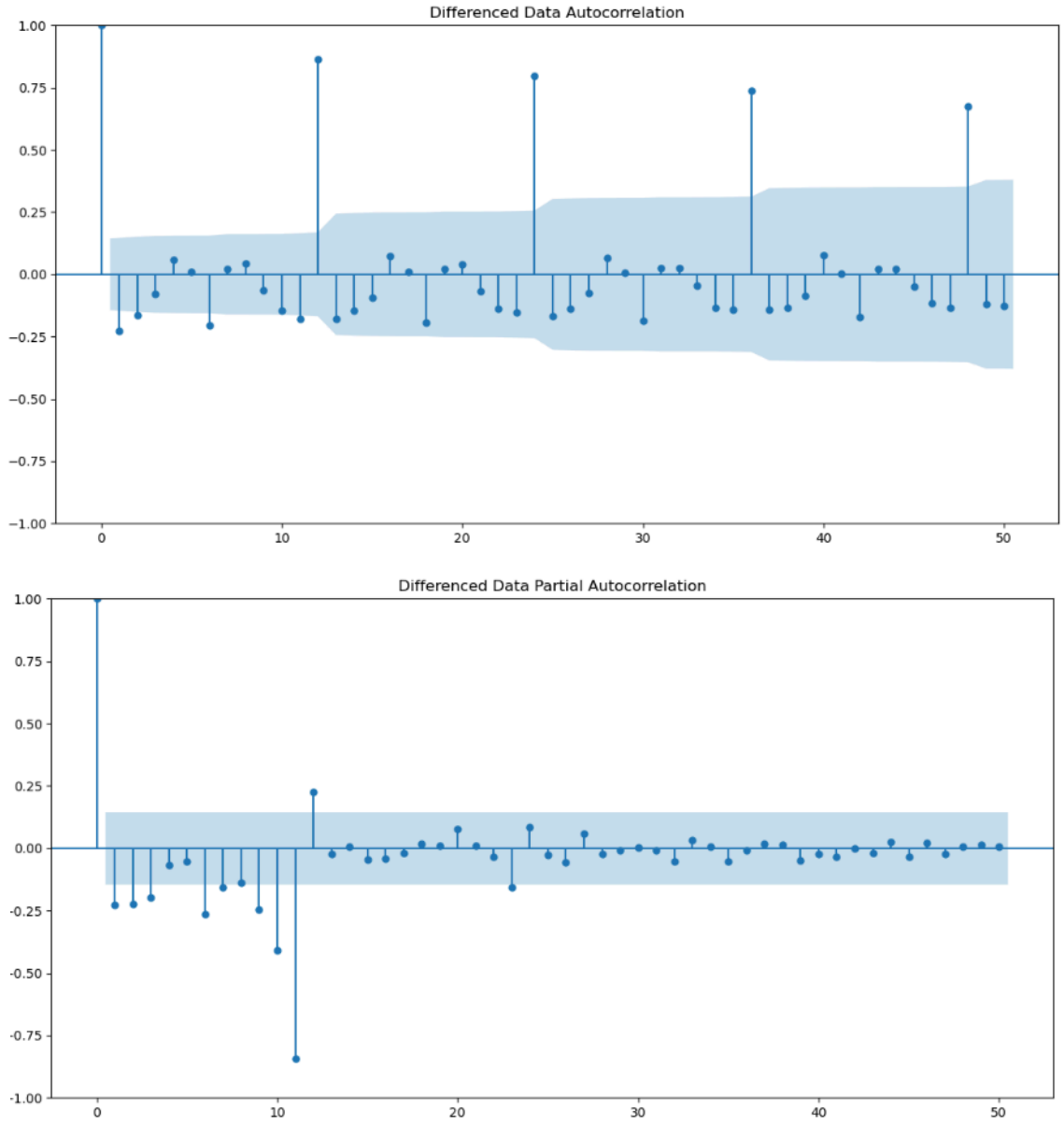


Fig-68 ACF and PACF for Stationary Train Dataset

Auto ARIMA Model:

Next, we will build the ARIMA and SARIMA model for Autofit . For the Autofit model in ARIMA the parameters are selected using AIC(Akaike Information Criteria) as shown in Table-27 below.We used this model on the test data and obtained the RMSE value as 37.334.

SARIMAX Results

Dep. Variable:	Sparkling	No. Observations:	132
Model:	ARIMA(2, 1, 2)	Log Likelihood	-1101.755
Date:	Sat, 14 Sep 2024	AIC	2213.509
Time:	20:23:40	BIC	2227.885
Sample:	01-01-1980	HQIC	2219.351
	- 12-01-1990		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	1.3121	0.046	28.781	0.000	1.223	1.401
ar.L2	-0.5593	0.072	-7.739	0.000	-0.701	-0.418
ma.L1	-1.9917	0.109	-18.214	0.000	-2.206	-1.777
ma.L2	0.9999	0.110	9.108	0.000	0.785	1.215
sigma2	1.099e+06	2e-07	5.51e+12	0.000	1.1e+06	1.1e+06

Ljung-Box (L1) (Q):	0.19	Jarque-Bera (JB):	14.46
Prob(Q):	0.67	Prob(JB):	0.00
Heteroskedasticity (H):	2.43	Skew:	0.61
Prob(H) (two-sided):	0.00	Kurtosis:	4.08

Table-27 Auto ARIMA model Build

Auto SARIMA Model:

Now, we will build the SARIMA model in autofit using AIC and obtain the below Table-28.

SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(1, 1, 2)x(0, 1, 2, 12)	Log Likelihood	-685.174
Date:	Sat, 14 Sep 2024	AIC	1382.348
Time:	20:28:35	BIC	1397.479
Sample:	0	HQIC	1388.455
	- 132		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.5507	0.287	-1.922	0.055	-1.112	0.011
ma.L1	-0.1612	0.235	-0.687	0.492	-0.621	0.299
ma.L2	-0.7218	0.175	-4.132	0.000	-1.064	-0.379
ma.S.L12	-0.4062	0.092	-4.401	0.000	-0.587	-0.225
ma.S.L24	-0.0274	0.138	-0.198	0.843	-0.298	0.243
sigma2	1.705e+05	2.45e+04	6.956	0.000	1.22e+05	2.19e+05

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	13.48
Prob(Q):	0.95	Prob(JB):	0.00
Heteroskedasticity (H):	0.89	Skew:	0.60
Prob(H) (two-sided):	0.75	Kurtosis:	4.44

Table-28 Auto SARIMA model Build

Fig-69 shows the diagnostics plot for Auto SARIMA model and tested on test data and obtained the predicted values as shown in Table-29.

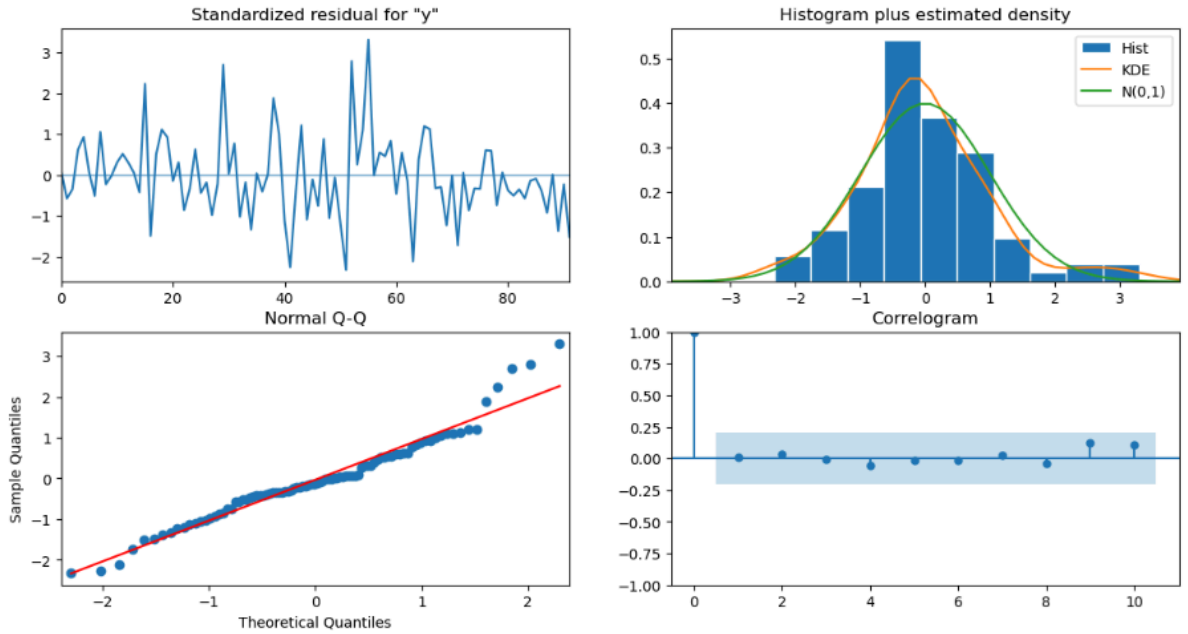


Fig-69 Diagnostic plot for Auto SARIMA

YearMonth	Sparkling	spark_forecasted
1991-01-01	1902	1460.244613
1991-02-01	2049	1392.437155
1991-03-01	1874	1743.201695
1991-04-01	1279	1650.066914
1991-05-01	1432	1522.656022

Table-29 Predicted values as per Auto SARIMA

We will plot the graph and obtain as seen in Fig-70 how well the Auto SARIMA did on test data and obtained the RMSE value as 16.528.

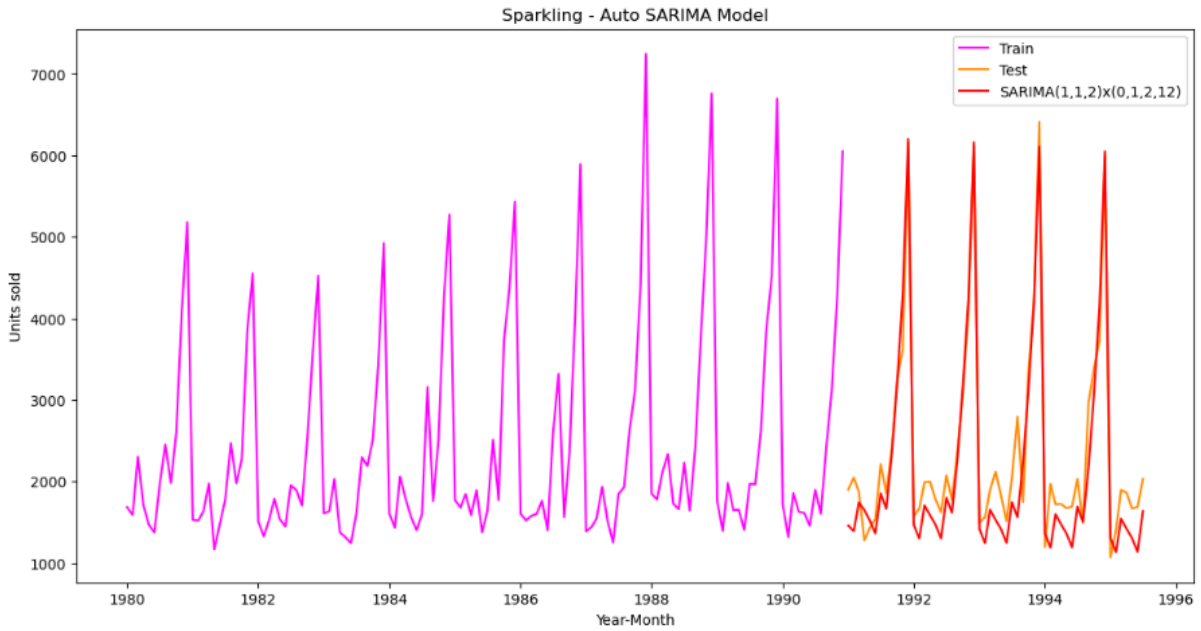
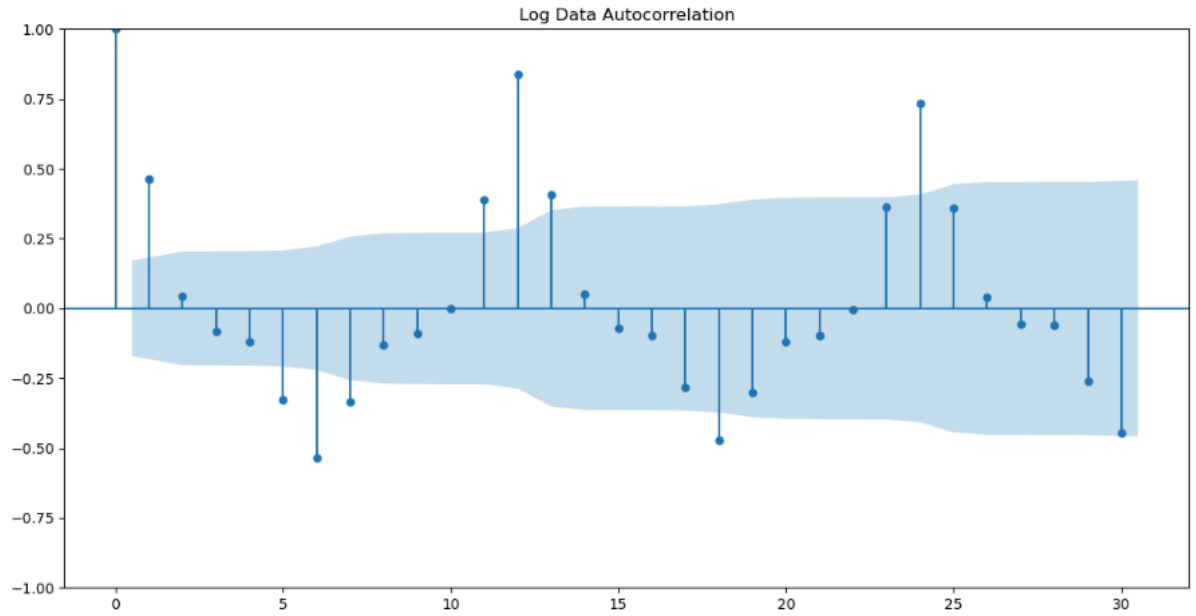
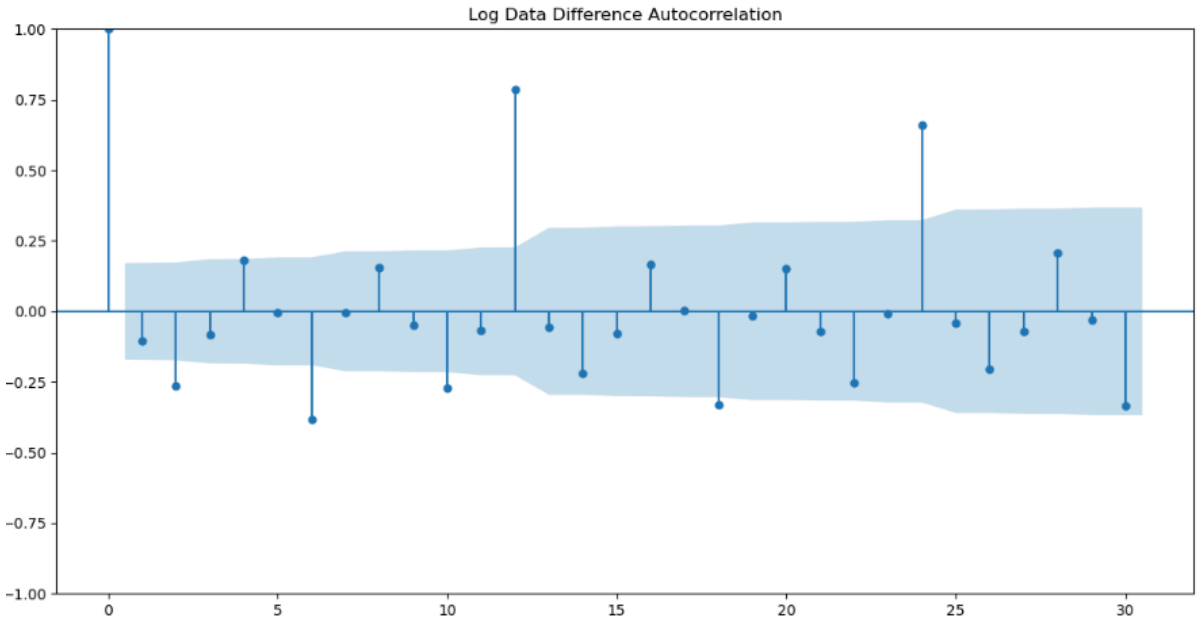


Fig-70 Time series as per Auto SARIMA model

Auto SARIMA Log Transformed Model:

For the Auto SARIMA model we find the ACF and PACF for the log transformed train dataset and obtain the plot as observed in fig-71.


Log Data Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30


Log Data Difference Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30

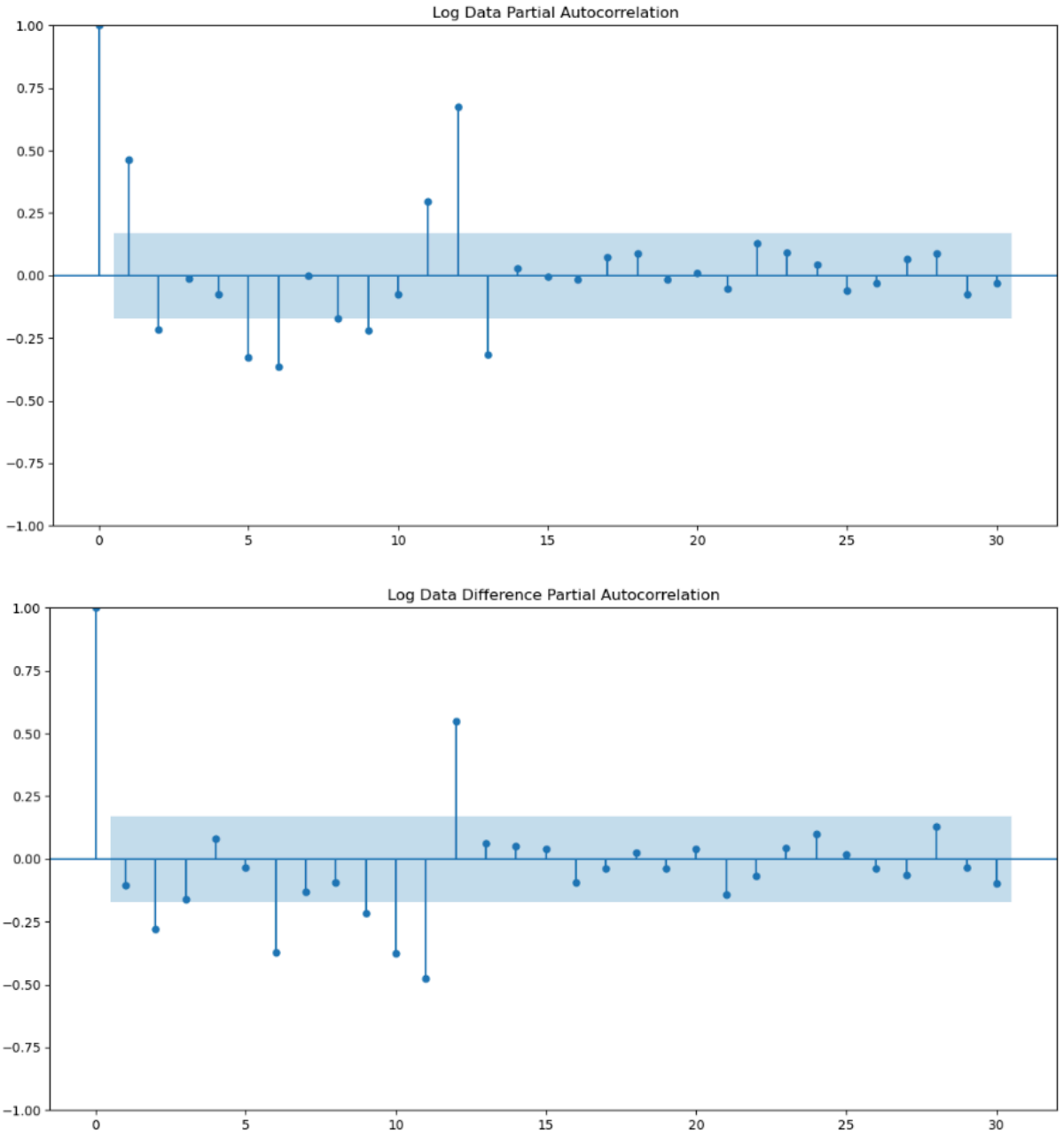


Fig-71 ACF and PACF for log transformed Auto SARIMA

Now, we build the Auto SARIMA model for this log transformed dataset and obtained the Table-30 as seen in below. And its diagnostic plot can be seen in fig-72.

SARIMAX Results

Dep. Variable:	Sparkling	No. Observations:	132
Model:	SARIMAX(0, 1, 1)x(1, 0, 1, 12)	Log Likelihood	146.236
Date:	Sat, 14 Sep 2024	AIC	-284.472
Time:	20:35:42	BIC	-273.423
Sample:	01-01-1980	HQIC	-279.986
	- 12-01-1990		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	-0.8966	0.045	-19.861	0.000	-0.985	-0.808
ar.S.L12	1.0112	0.020	49.872	0.000	0.971	1.051
ma.S.L12	-0.6490	0.075	-8.629	0.000	-0.796	-0.502
sigma2	0.0045	0.001	7.841	0.000	0.003	0.006

Ljung-Box (L1) (Q):	0.11	Jarque-Bera (JB):	5.26
Prob(Q):	0.74	Prob(JB):	0.07
Heteroskedasticity (H):	1.43	Skew:	-0.00
Prob(H) (two-sided):	0.27	Kurtosis:	4.04

Table-30 Log transformed Auto SARIMA model Build

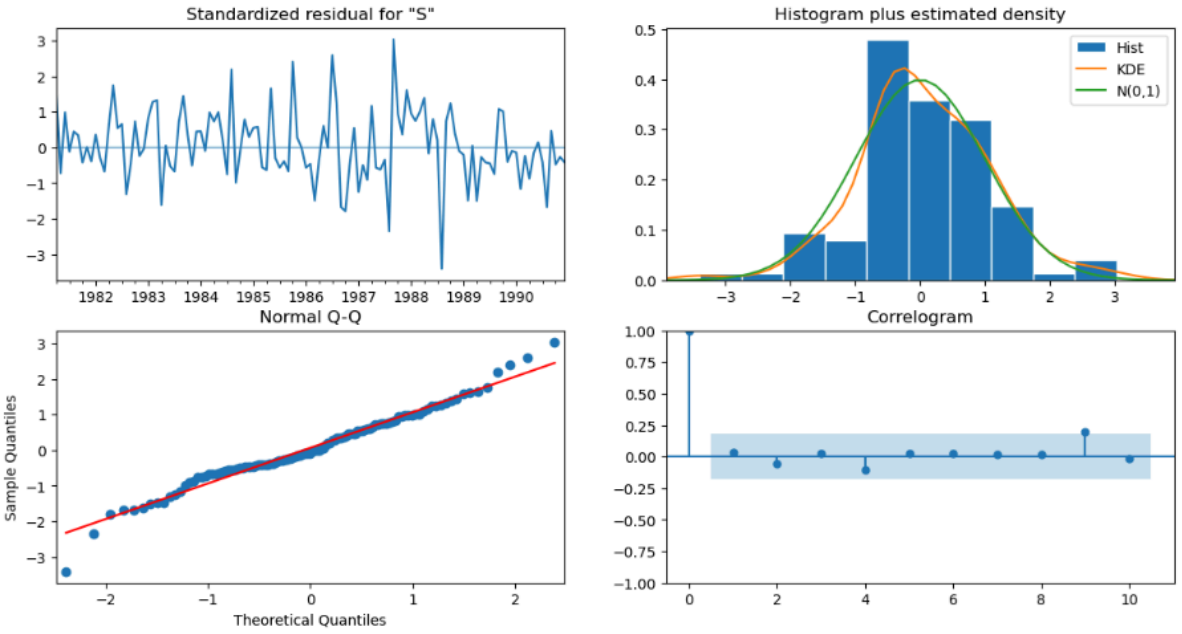


Fig-72 Diagnostic plot for log transformed Auto SARIMA

After the model is trained we apply the log transformation on the test dataset and use this SARIMA dataset to forecast the test data and can be seen in fig-73.The RMSE value obtained for this model is 17.918.

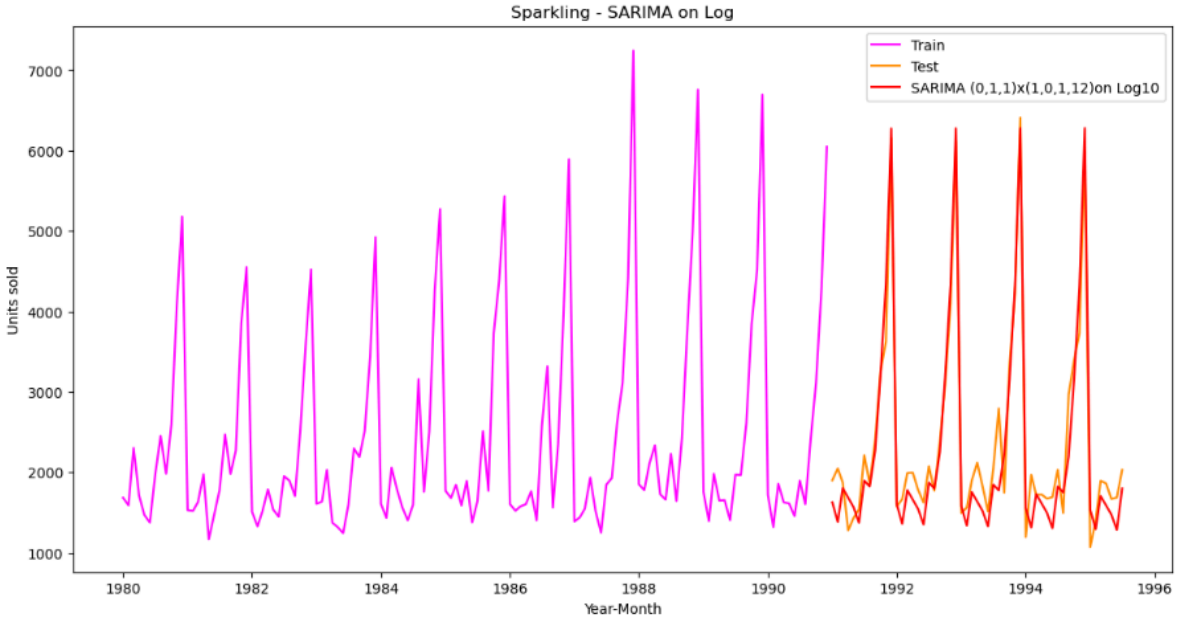


Fig-73 Time series as per log transformed Auto SARIMA model

Manual ARIMA model:

Now, we will build the Manual ARIMA model using order(0,1,0) and obtain the Table-31.We predict test data values and obtain the value for RMSE is 79.746. For this model we can see the ACF and PACF in Fig-74.

SARIMAX Results

Dep. Variable:	Sparkling	No. Observations:	132
Model:	ARIMA(0, 1, 0)	Log Likelihood	-1132.832
Date:	Sat, 14 Sep 2024	AIC	2267.663
Time:	20:36:34	BIC	2270.538
Sample:	01-01-1980	HQIC	2268.831
	- 12-01-1990		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
sigma2	1.885e+06	1.29e+05	14.658	0.000	1.63e+06	2.14e+06

Ljung-Box (L1) (Q):	3.07	Jarque-Bera (JB):	198.83
Prob(Q):	0.08	Prob(JB):	0.00
Heteroskedasticity (H):	2.46	Skew:	-1.92
Prob(H) (two-sided):	0.00	Kurtosis:	7.65

Table-31 Manual ARIMA model Build

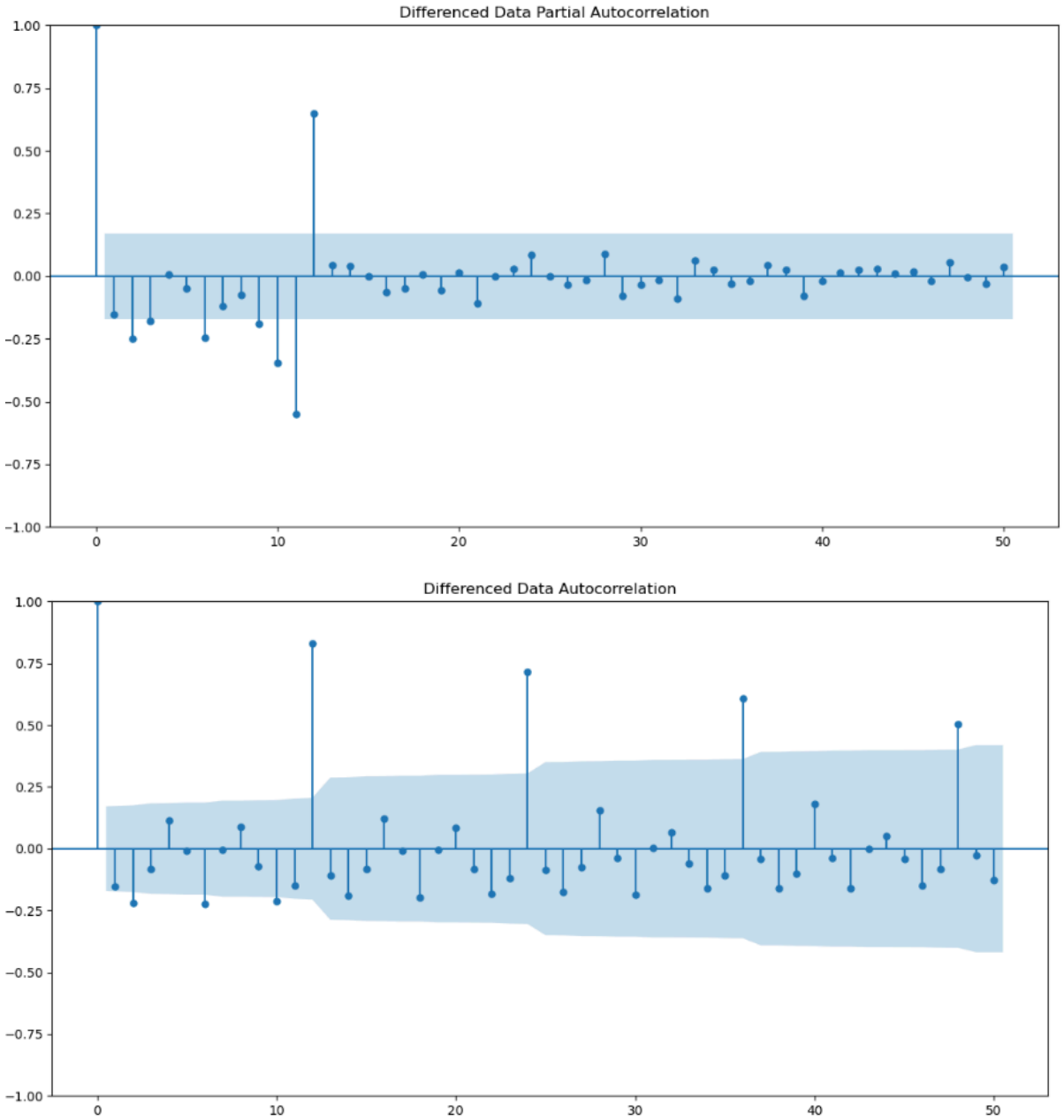


Fig-74 ACF and PACF for Manual ARIMA

Manual SARIMA model:

Next, we will build the Manual SARIMA model for (4,1,2)*(0,1,2,12) and obtain Table-32. and forecast the model on test data to obtain a graph as seen in FIg-75.

SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(3, 1, 1)x(1, 1, [1, 2], 12)	Log Likelihood	-693.697
Date:	Sat, 14 Sep 2024	AIC	1403.394
Time:	20:39:09	BIC	1423.654
Sample:	0	HQIC	1411.574
	- 132		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.2229	0.130	1.713	0.087	-0.032	0.478
ar.L2	-0.0798	0.131	-0.607	0.544	-0.337	0.178
ar.L3	0.0921	0.122	0.756	0.450	-0.147	0.331
ma.L1	-1.0241	0.094	-10.925	0.000	-1.208	-0.840
ar.S.L12	-0.1992	0.866	-0.230	0.818	-1.897	1.499
ma.S.L12	-0.2109	0.881	-0.239	0.811	-1.938	1.516
ma.S.L24	-0.1299	0.381	-0.341	0.733	-0.877	0.617
sigma2	1.654e+05	2.62e+04	6.302	0.000	1.14e+05	2.17e+05

Ljung-Box (L1) (Q):	0.04	Jarque-Bera (JB):	19.66
Prob(Q):	0.83	Prob(JB):	0.00
Heteroskedasticity (H):	0.81	Skew:	0.69
Prob(H) (two-sided):	0.56	Kurtosis:	4.78

Table-32 Manual SARIMA model Build

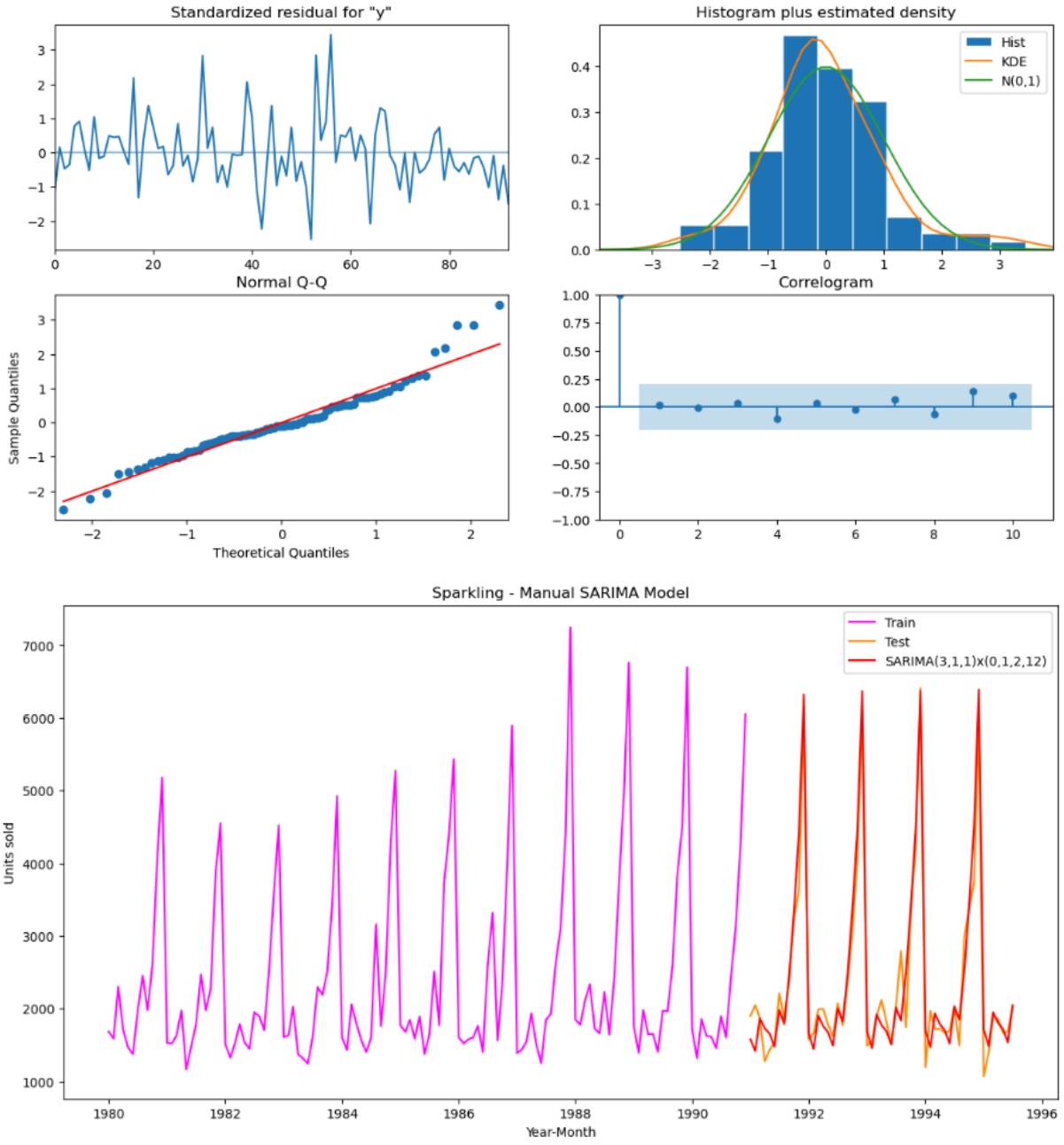
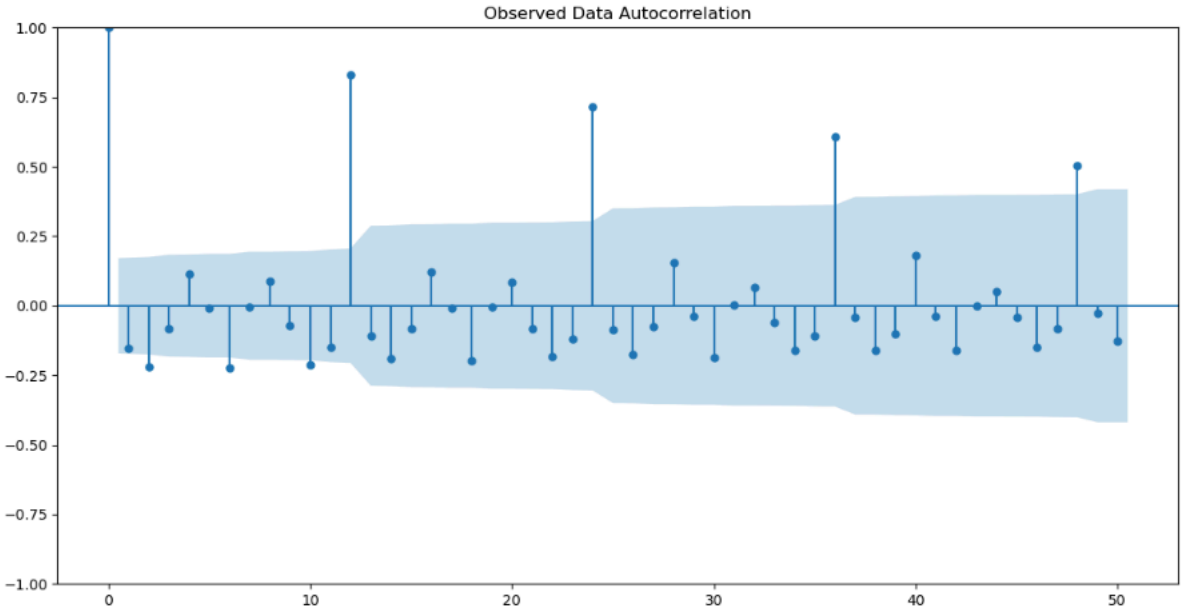
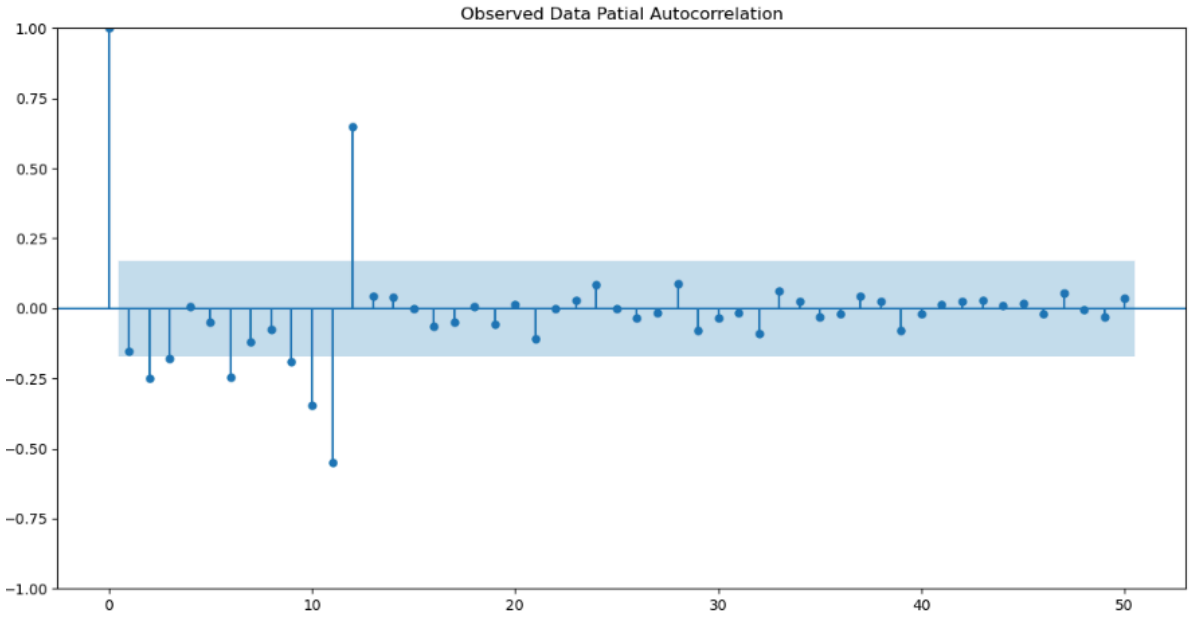
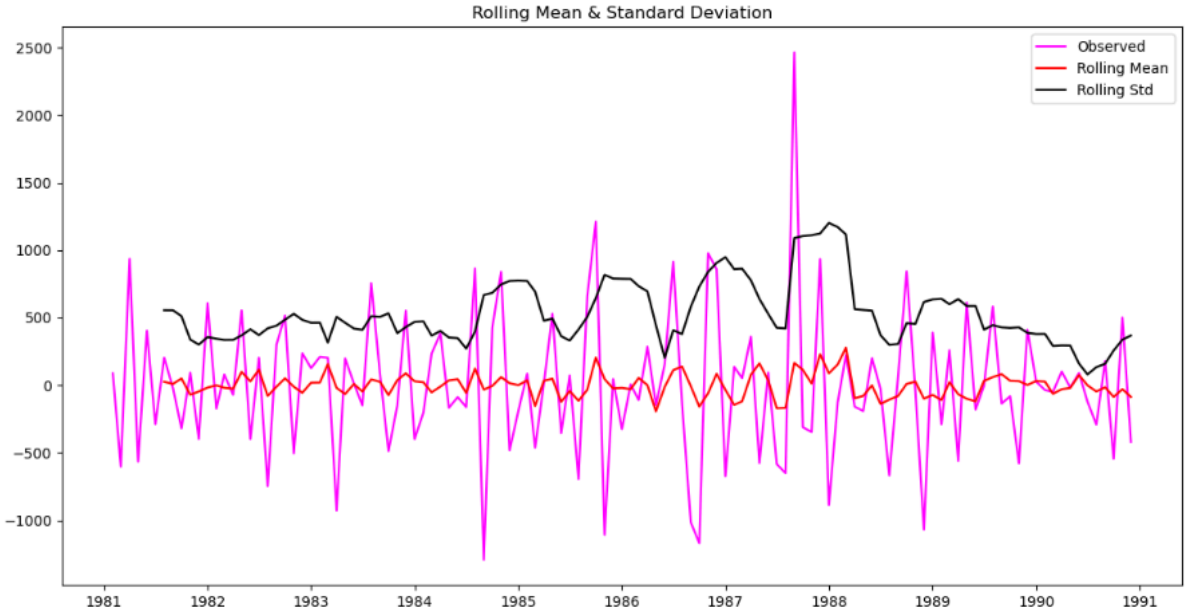


Fig-75 TIme series for Manual SARIMA

For this model the ACF and PACF are seen in Fig-76. The RMSE value obtained from this model is15.389.


Observed Data Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50


Observed Data Patial Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50




```
Results of Dickey-Fuller Test:
Test Statistic                  -3.342905
p-value                          0.013066
#Lags Used                      10.000000
Number of Observations Used    108.000000
Critical Value (1%)             -3.492401
Critical Value (5%)             -2.888697
Critical Value (10%)            -2.581255
dtype: float64
```

Fig-76 ACF and PACF for Manual SARIMA

11. Compare the performance of the models:

Now as seen in the below Table-33 we can say that the best model is the Manual SARIMA model.

	Test RMSE
Manual_SARIMA#(3,1,1)*(1,1,2,12)	324.107217
Auto_SARIMA_log(0, 1, 1)*(1, 0, 1, 12)	336.810981
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576735
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	396.598057
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	404.286809
2 point TMA	813.400684
4 point TMA	1156.589694
SimpleAverage	1275.081804
6 point TMA	1283.927428
Alpha=0.025,SES iterative	1286.248846
Auto_ARIMA(2,1,2)	1299.980426
Alpha=0.0496, SES Optimized	1304.927405
9 point TMA	1346.278315
RegressionOnTime	1389.135175
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Manual_ARIMA(0,1,0)	3864.279352
NaiveModel	3864.279352

SARIMAX Results

Dep. Variable:	y	No. Observations:	187
Model:	SARIMAX(3, 1, 1)x(1, 1, [1, 2], 12)	Log Likelihood	-1094.342
Date:	Sat, 14 Sep 2024	AIC	2204.685
Time:	20:41:08	BIC	2228.662
Sample:	0	HQIC	2214.427
	- 187		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.1159	0.086	1.349	0.177	-0.052	0.284
ar.L2	-0.0639	0.100	-0.636	0.525	-0.261	0.133
ar.L3	0.0473	0.091	0.521	0.603	-0.131	0.225
ma.L1	-0.9658	0.036	-26.792	0.000	-1.036	-0.895
ar.S.L12	-0.1973	0.706	-0.279	0.780	-1.581	1.186
ma.S.L12	-0.3455	0.717	-0.482	0.630	-1.751	1.060
ma.S.L24	-0.1219	0.398	-0.306	0.759	-0.902	0.658
sigma2	1.528e+05	1.53e+04	10.019	0.000	1.23e+05	1.83e+05

Ljung-Box (L1) (Q):	0.02	Jarque-Bera (JB):	42.29
Prob(Q):	0.90	Prob(JB):	0.00
Heteroskedasticity (H):	0.77	Skew:	0.71
Prob(H) (two-sided):	0.37	Kurtosis:	5.20

Table-33 Manual SARIMA is the best Model

Fig-77 shows the Forecasted values on the whole original dataset and predicts the future values for the next 12 months.The Fig-78 shows the 12 months that are forecasted for the rose dataset.

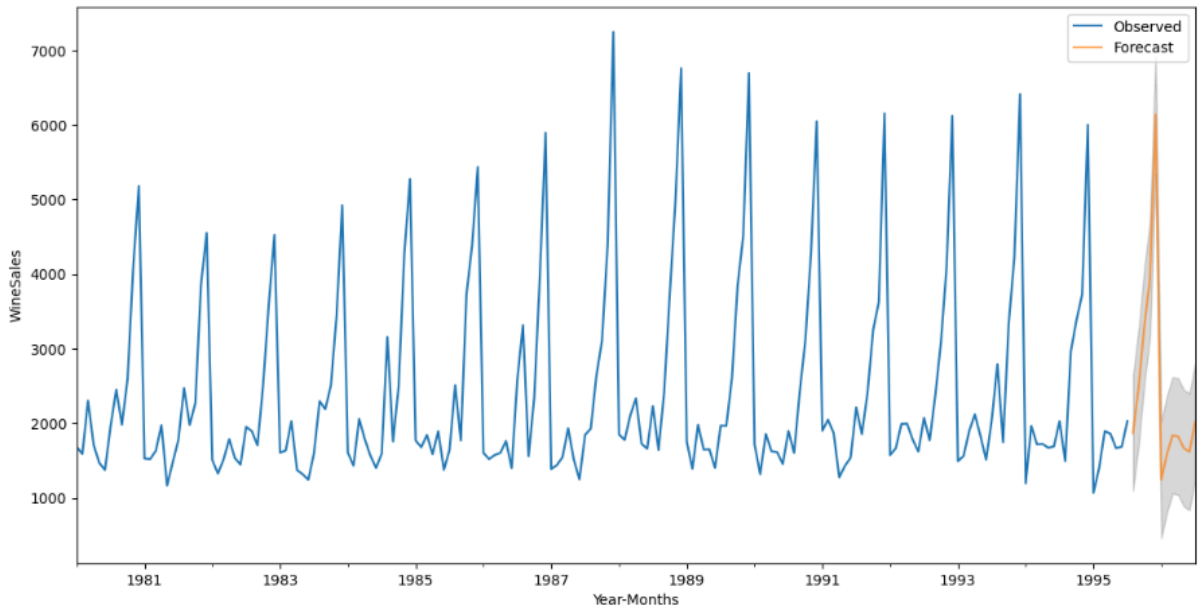


Fig-77 Forecast for the next 12 months sales of Sparkling wine

The Statistical Summary for the forecasted values are given below:

```
count      12.000000
mean     2461.187701
std      1391.118211
min      1245.727215
25%      1656.647698
50%      1855.796933
75%      2692.130206
max      6135.396044
Name: mean, dtype: float64
```

The Predicted values are Given Below:

```
1995-08-31    1870.888601
1995-09-30    2489.623603
1995-10-31    3299.650018
1995-11-30    3934.056636
1995-12-31    6135.396044
1996-01-31    1245.727215
1996-02-29    1584.643760
1996-03-31    1840.705265
1996-04-30    1823.847831
1996-05-31    1668.706103
1996-06-30    1620.472485
1996-07-31    2020.534856
Freq: ME, Name: mean, dtype: float64
```

Table-34 shows the Future forecast results and Summary statistics

12. Actionable Insights & Recommendations:

Based on the provided data and analysis, here are some actionable insights and recommendations for ABC Estate Wine to address the flat sales trend of Sparkling wine and capitalise on seasonal fluctuations:

1. Capitalise on the Holiday Season

- Increase Inventory: Build stock in anticipation of rising sales and the December peak. Ensure sufficient inventory to meet the expected demand of 6136 units in December.
- Targeted Advertising (Oct-Dec): Launch focused advertising campaigns during the holiday season to leverage the existing buying trend and potentially boost sales further. Highlight the celebratory nature of Sparkling wine.

2. Address Flat Sales Trend

- Deep Sales Dive (Jan-Mar): Utilise the first quarter slowdown to conduct a thorough analysis of year-over-year sales data to understand the stagnant sales trend. Identify potential areas for improvement outside the holiday season.
- Customer Feedback: Gather feedback from existing customers to understand their perceptions and preferences regarding Sparkling wine. Use this information to inform strategic decisions.

3. Product Innovation & Marketing

- Celebration-Themed Design: Consider introducing a special, lower-priced bottle design specifically for celebratory purposes (e.g., bottle designed for popping). This can attract customers looking for festive options.
- Summer Marketing: Develop marketing campaigns promoting Sparkling wine for summer gatherings (e.g., picnics, barbecues) to capitalise on the sales recovery period (Jul-Aug). Highlight the versatility of Sparkling wine for various occasions.

4. Maintain Steady Inventory

- Consistent Inventory Levels: While December brings a significant sales increase, maintaining a consistent inventory level throughout the year might help capture potential customers who enjoy Sparkling wine outside the holiday season. This ensures availability and can help smooth out sales fluctuations.

5. Evaluate Sales Performance Post-Holiday Season

- Assess Sales Trends: After the December peak, evaluate the overall sales trend. If there is a positive trend, continue with the existing Sparkling wine variant and marketing strategies.
- Adjust Strategies: If sales do not improve, consider adjusting marketing strategies or exploring new distribution channels to reach a broader audience.

By implementing these recommendations, ABC Estate Wine can address the flat sales trend of Sparkling wine, capitalise on seasonal spikes, and work towards improving overall sales performance.