Great Learning

TIME SERIES
BUSINESS REPORT

Submitted by:

N. Aishwarya
PGP-DSBA

July 2022

Table of Contents

List of Figures

List of Tables

Executive Summary – Time Series Forecasting:

Time series is defined as an ordered sequence of values of a variable at equally spaced time intervals. Based on the method, we obtain an understanding of the underlying factors that produced the observed data; this helps to fit a model and thereby forecast, monitor or even feedback and feedforward control.

Some of the applications of Time series include: Economic Forecasting, Sales Forecasting, Budgetary Analysis, Stock Market Analysis, Process and Quality Control etc.

Business problem 1 – Time Series Forecasting-Sparkling dataset

Problem Statement:

For this particular assignment, the data of different types of wine sales in the 20th century is to be analysed. Both of these data are from the same company but of different wines. As an analyst in the ABC Estate Wines, you are tasked to analyse and forecast Wine Sales in the 20th century.

Solution Approach:

The purpose of the solutioning exercise is to explore the dataset using time series techniques to arrive at the customer segmentation, thus enabling business strategies customized to them.

1.1. Read the data as an appropriate Time Series data and plot the data

Dataset Background:

- Monthly Sales Data of 'Sparkling' Wine manufactured by ABC Estate Wines starting from Jan 1980 to July 1995 is provided.
- As an analyst in the ABC Estate Wines, the task is to analyse and forecast Wine Sales in the 20th century.

Data Dictionary of the Dataset:

- The dataset 'Sparkling' contains two columns of data. The monthly time stamp from Jan 1980 to July 1995 and the sales corresponding to the wines.

1.1 Read the data as an appropriate Time Series data and plot the data.

Loaded required packages and read Monthly Sales of Sparkling wine dataset without using panda's date-time format.

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

Creating Time Stamps and adding it to the data frame to make it a Time Series Data:

```
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='M')
```

Add the time stamp to the original data-frame and set the time stamp as an index, also drop the Year Month column from the dataset.

	Sparkling
Time_Stamp	
1980-01-31	1686
1980-02-29	1591
1980-03-31	2304
1980-04-30	1712
1980-05-31	1471

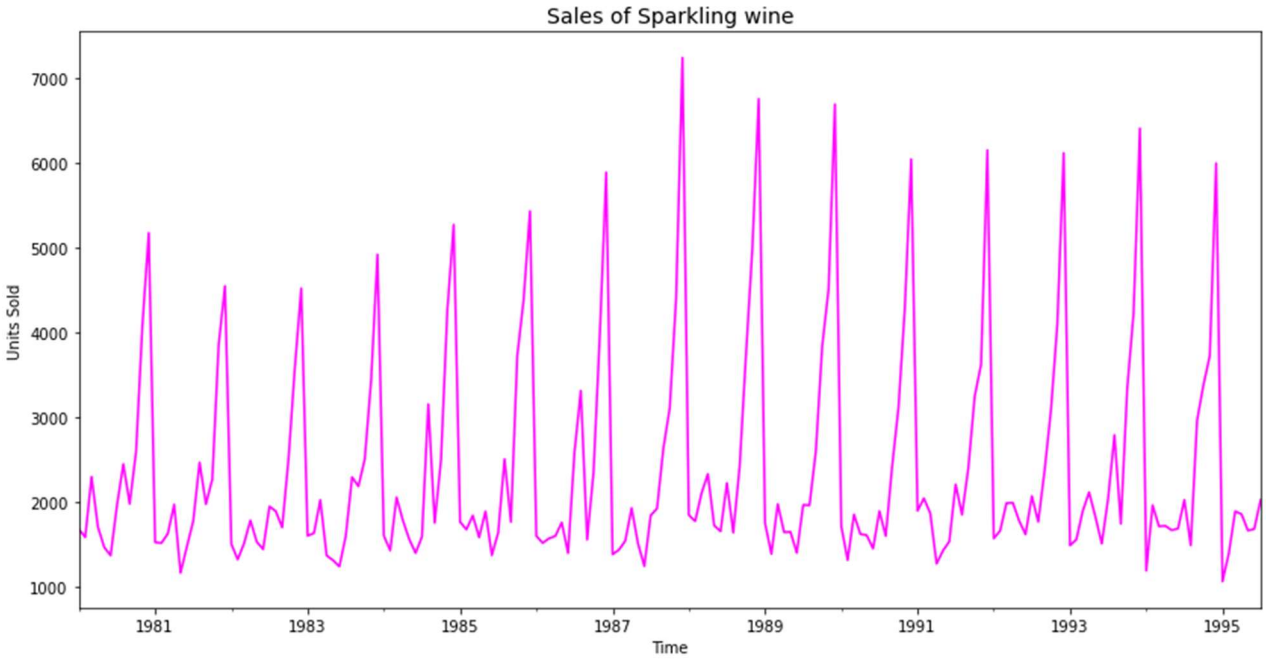
Figure 1: Head of the Time Series data

As it is a Time Series data, handling Null values is of utmost importance. The null values cannot be dropped as the Time series data need to contiguous, so they need to be properly imputed.

Inferences:

- All values are properly loaded for the dataset with the index as panda's date-time format.
- Sparkling time series data do not contain any missing values.

Plotting the Sparkling Time Series to understand the behavior of the data:



Inferences:

- The Sparkling wine dataset shows significant seasonality and doesn't show any consistent trend but has upward and downward slopes during the time period. Trend decreases initially, then increases and starts decreasing. Seasonality is also observed from the plot.
- Sparkling wine has been consistently favored over the years by customers.

1.2 Perform appropriate Exploratory Data Analysis to understand the data and also perform decomposition.

Solution approach:

Check the basic measures of descriptive statistics:

```
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
```

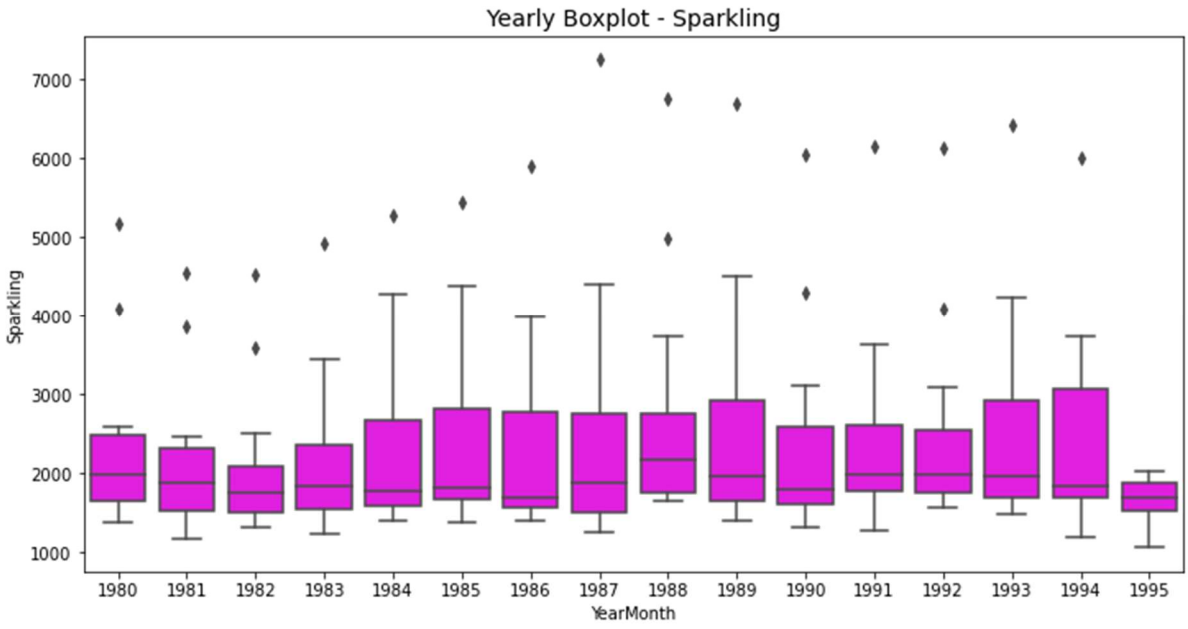
Figure 2: Data Description for Sparkling Dataset

The basic measures of descriptive statistics tell us how the Sales have varied across years. But for this measure of descriptive statistics, we have averaged over the whole data without taking the time component into account.

The descriptive summary of the data shows that:

- on an average 2402 units of Sparkling wines were sold each month on the given period of time.
- 50% of month's sales varied from 1605 units to 2549 units.
- Maximum sale reported in a month is 7242 units.

We will Plot a boxplot to understand the spread of sales across different years and within different months across years.



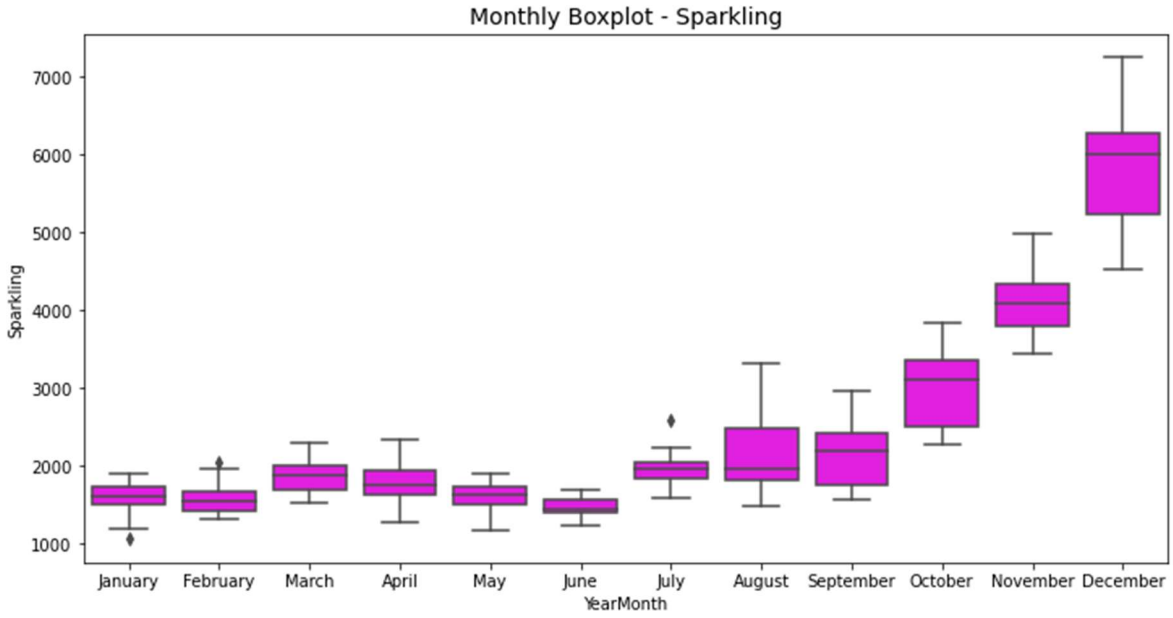
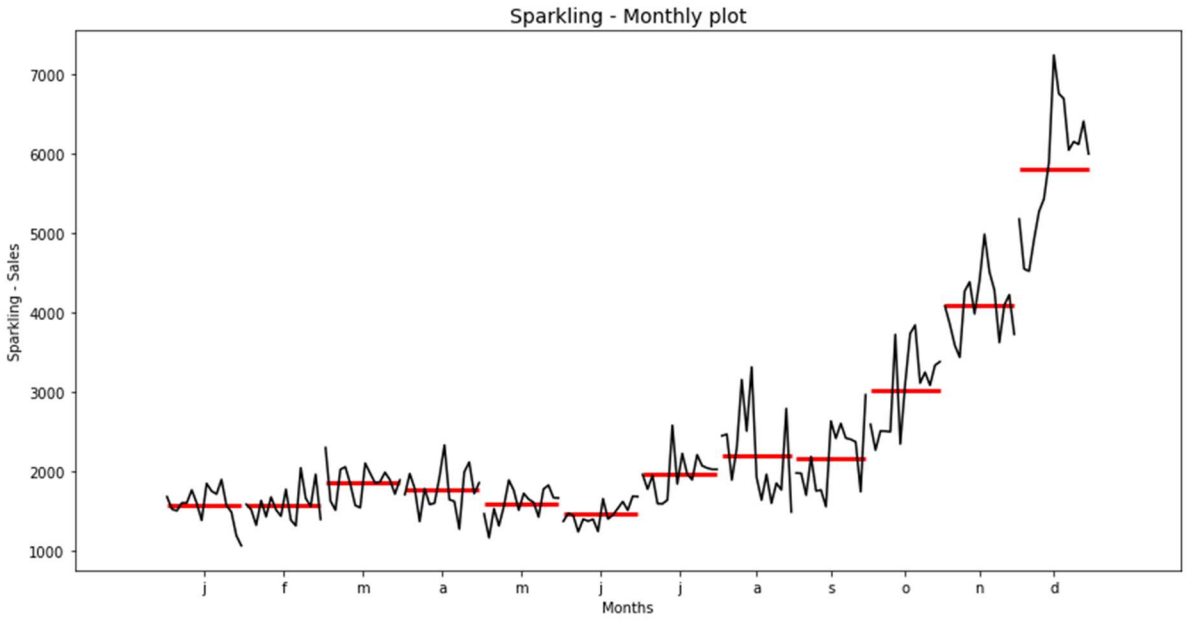


Figure 3: Yearly and Monthly Boxplot for all the years for Sparkling Dataset

Inferences:

- The yearly-boxplot, shows that the average sale of Sparkling has been more or less consistent across the period, at or a little below 2000 units.
- The outliers in the yearly-boxplot most probably represent the seasonal sale during the seasonal months.
- The monthly-box-plot shows a clear seasonality during the festive seasonal months of October, November and December, which peaks in December.

The given below plot shows us the behavior of the Time Series across various months. The red line is the median value.



Inferences:

- The monthly plot for Sparkling shows Mean and variation of units sold each month over the years. Sales in seasonal month's shows a higher variation than in the lean months.
- Sale in December with few points of mean below 6000, varies from 7400 to 4500 units over the years. Whereas sale in November varies from 3500 units to 5000 units and sale in October varies from 2500 to 4000 units.
- The lean months from January till September shows more or less a consistent sale around 2000 units.

Monthly Wine sales across years for Sparkling:

	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

Figure 4: Pivot table of sales across years

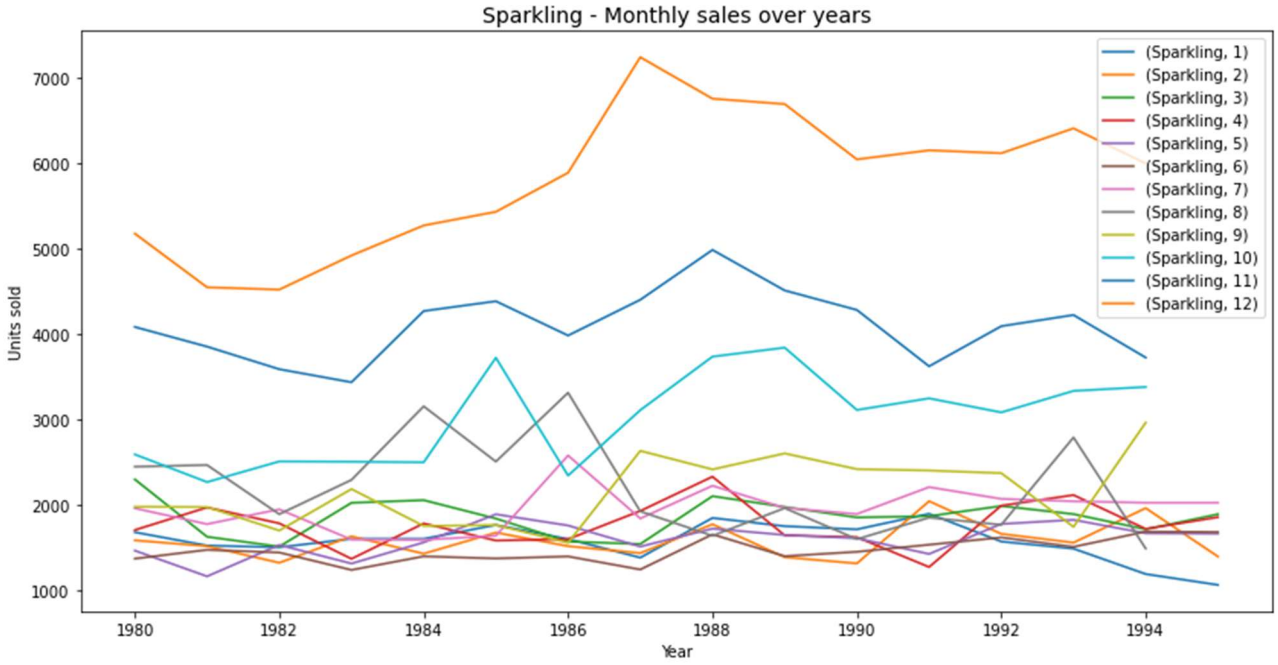


Figure 5: Plot of Monthly Wine sales across years for Sparkling

Observations:

- The plot of monthly sale over the years also shows the seasonality component of the time-series, with October, November and December selling exponentially higher volumes.
- The highest volume of Sparkling wines was sold in December, 1987 and the least of December sale was in 1981. Post 1987 December sales is around an average 6500 units, which was around 5000 in early 80's.
- The seasonal sale since 1990 has been more or less consistent around 6000 units in December, 4000 units in November and 3000 units in October.
- Sales for the months from January to July is seen to be consistent across the years, compared to the rest of the months

Plotting the average sales per month and the month-on-month percentage change of sales:

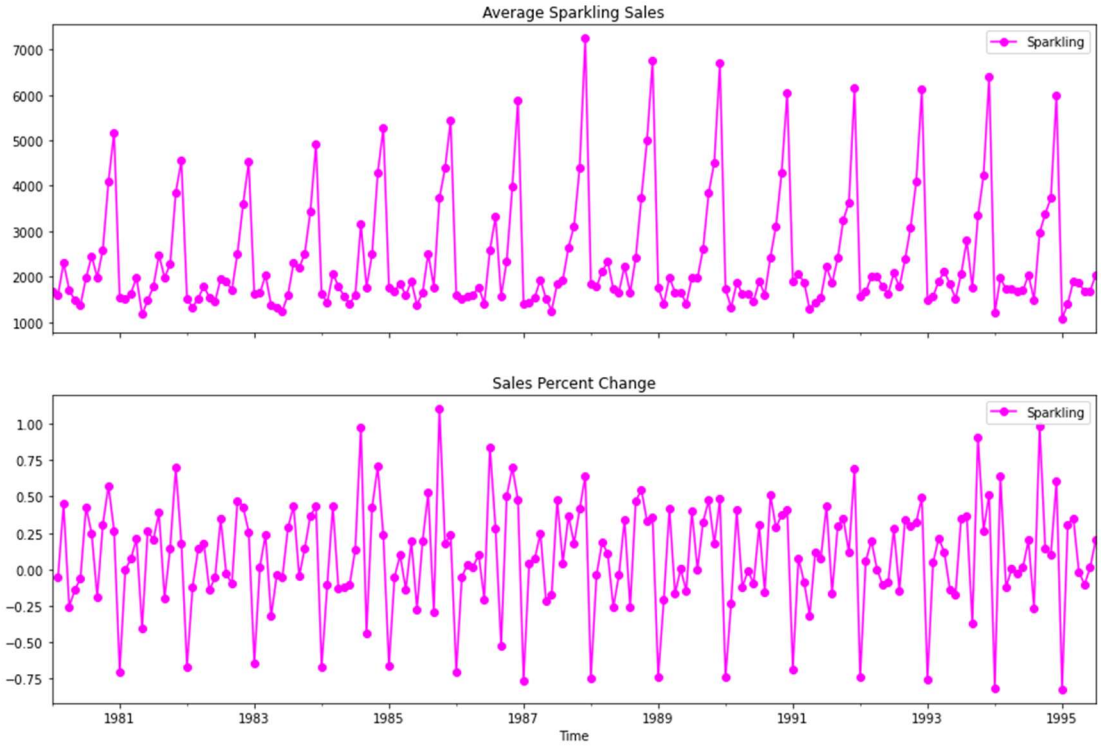


Figure 6: Average Sparkling sales and percent change

Decompose the Time Series and plot the different components:

If the seasonality and residual components are independent of the trend, then you have an additive series. If the seasonality and residual components are in fact dependent, meaning they fluctuate on trend, then you have a multiplicative series.

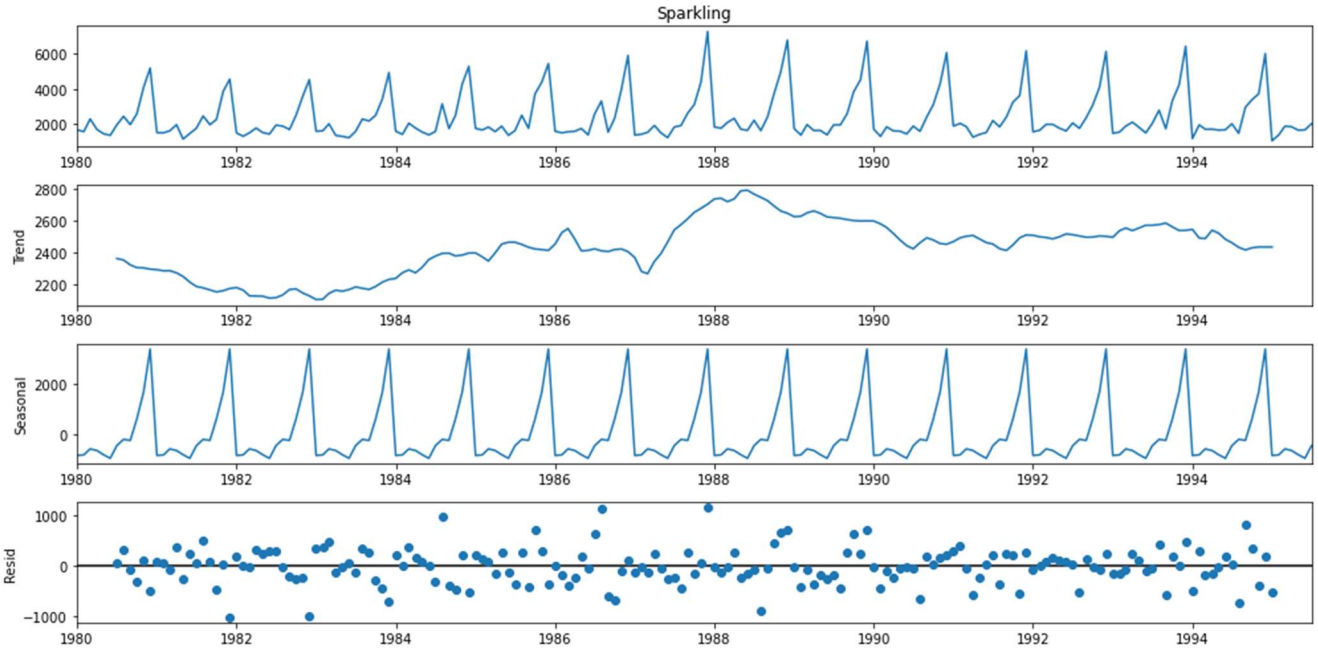


Figure 7: Decomposition of Sparkling Time Series with additive Seasonality

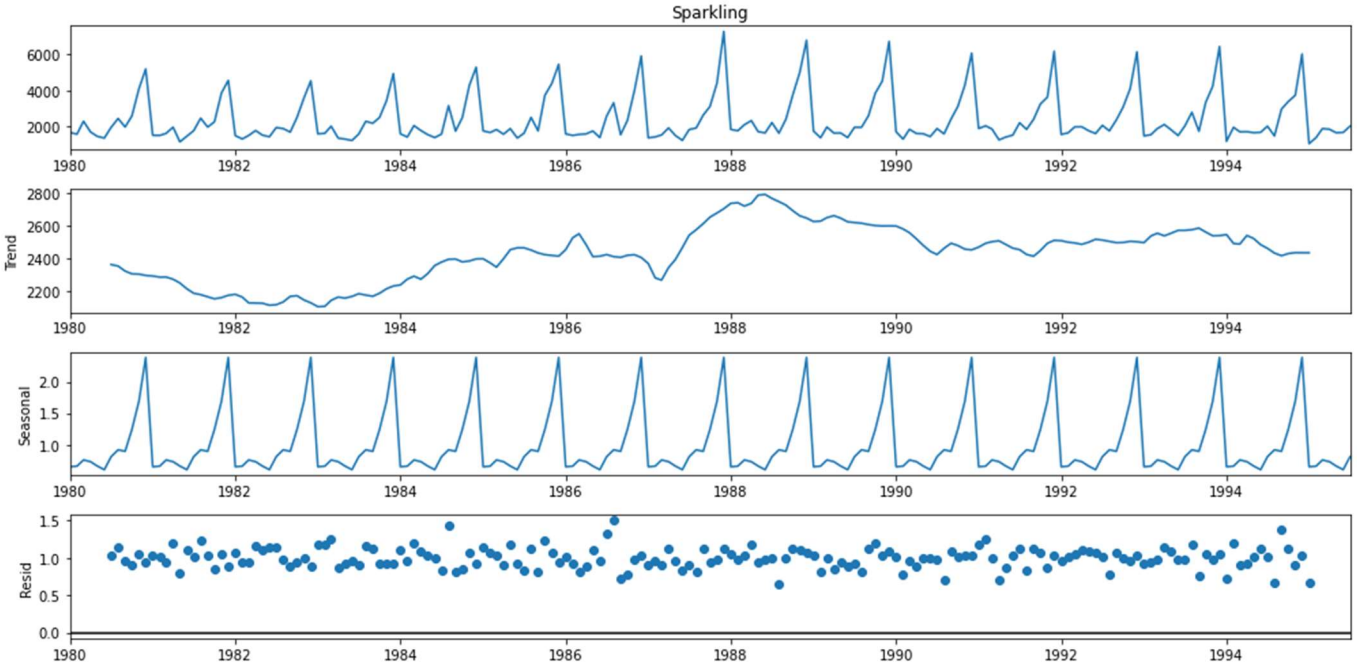


Figure 8: Decomposition of Sparkling Time Series with multiplicative Seasonality

Observations:

- As the altitude of the seasonal peaks in the observed plot is changing according to the change in trend, the time-series is assumed to be 'multiplicative'.
- The plot of the trend component does not show a consistent trend, but an intermediary period shows an upward trend which gets consistent on the late half of time-series.
- The additive model shows the seasonality with a variance of 3000 units and the multiplicative model shows a variance of 30%.
- The residual shows a pattern of high variability across the period of time-series, which is more or less consistent in both additive and multiplicative decompositions.
- The additive model shows a mean variance around 0 and the multiplicative model shows a variance around 10%.
- If the seasonality and residual components are independent of the trend, then you have an additive series. If the seasonality and residual components are in fact dependent, meaning they fluctuate on trend, then we have a multiplicative series.

1.3 Split the data into training and test. The test data should start in 1991.

The train and test datasets are created with year 1991 as starting year for test data.

```
train_spark = spark[spark.index.year < 1991]
test_spark = spark[spark.index.year >= 1991]
```

First and Last few rows of Training and Testing Data:

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

Figure 9: First and Last few rows of Training and Testing Data

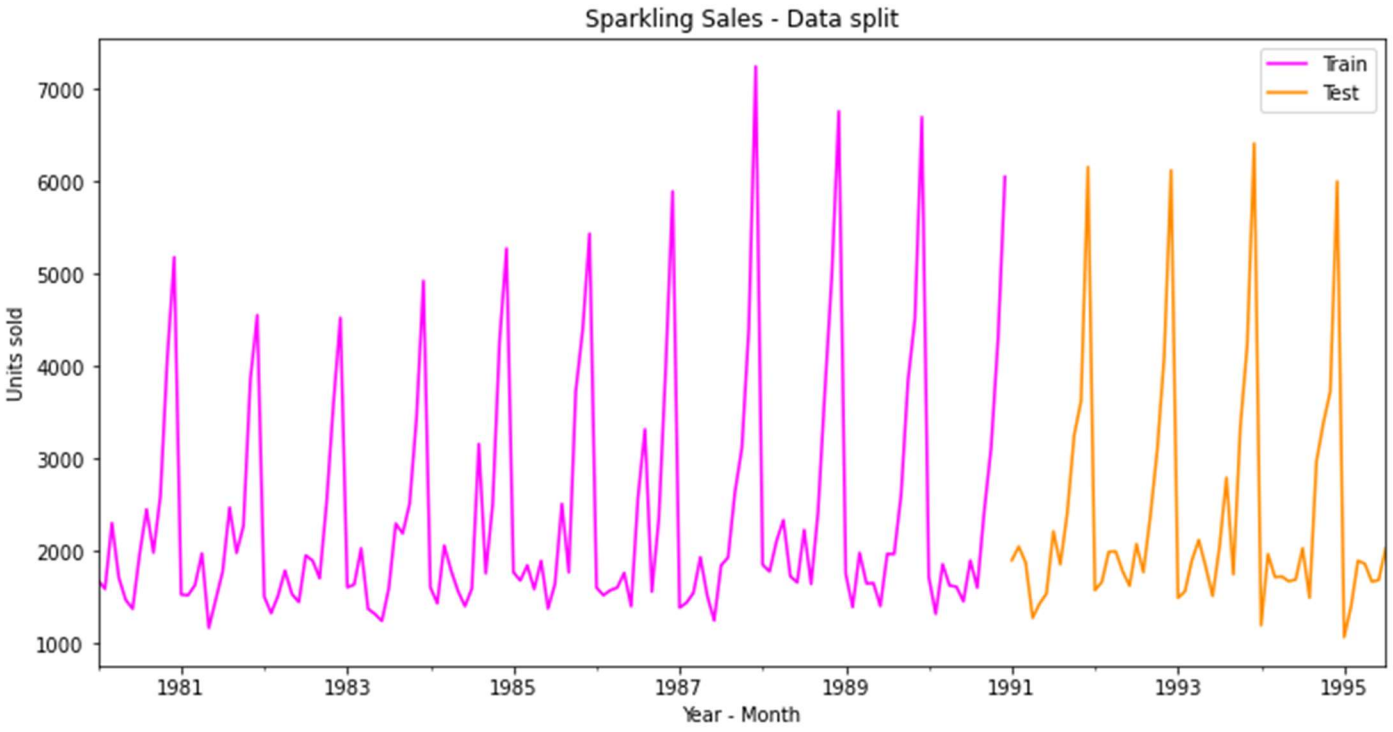


Figure 10: The Plot Sparkling Time Series as train and test

1.4 Build various exponential smoothing models on the training data and evaluate the model using RMSE on the test data. Other models such as regression, naïve forecast models, simple average models etc. should also be built on the training data and check the performance on the test data using RMSE.

Solution:

Model 1: Linear Regression

To regress the sale of Sparkling wines, numerical time instance order for both training and test set were generated and the values added to the respective datasets.

Training Time instance:

```
 [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,
65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95,
96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 12
1, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132]
Test Time instance
 [133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 15
7, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 1
82, 183, 184, 185, 186, 187]
```

We see that we have successfully generated the numerical time instance order for both the training and test set. Now we will add these values in the training and test set.

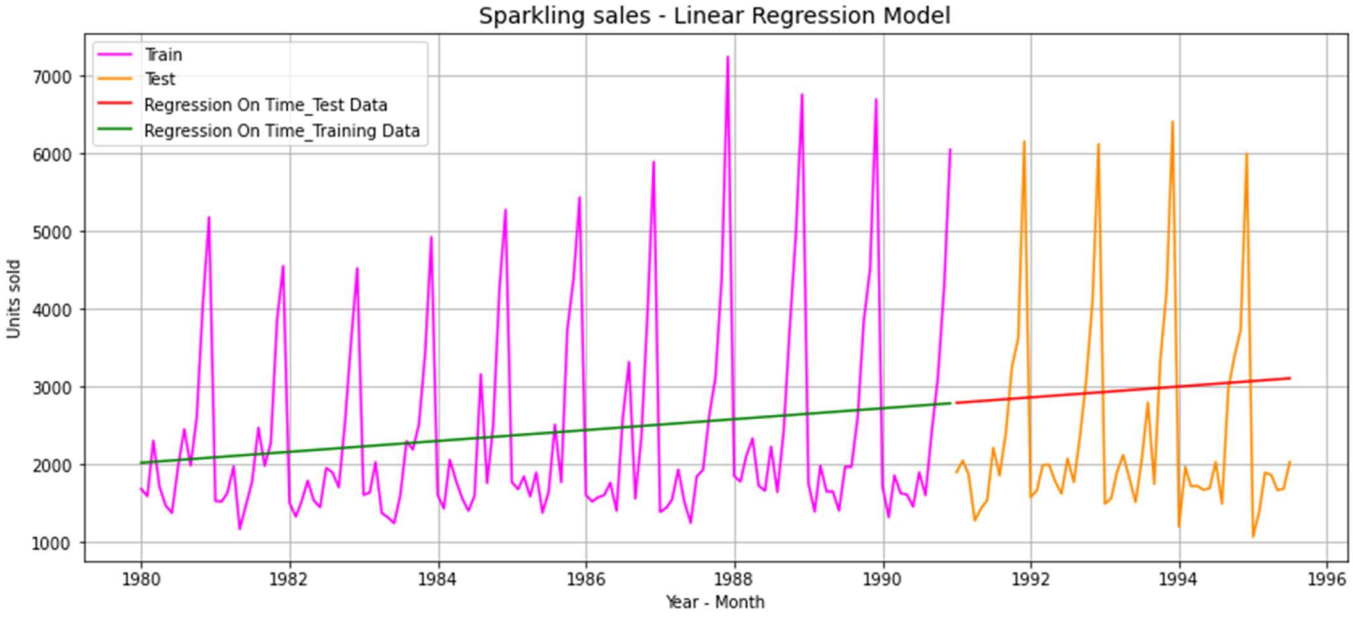


Figure 11 - Linear Regression model

Observations:

- The linear regression plots show a gradual upward trend in forecast of Sparkling wine, consistent with the observed trend which was not visually apparent.
- For Regression on Time forecast on the Test Data, RMSE is 1389.135.

Model 2: Naive forecast

In naive model, the prediction for tomorrow is the same as today and the prediction for day after tomorrow is tomorrow and since the prediction of tomorrow is same as today, therefore the prediction for day after tomorrow is also today.

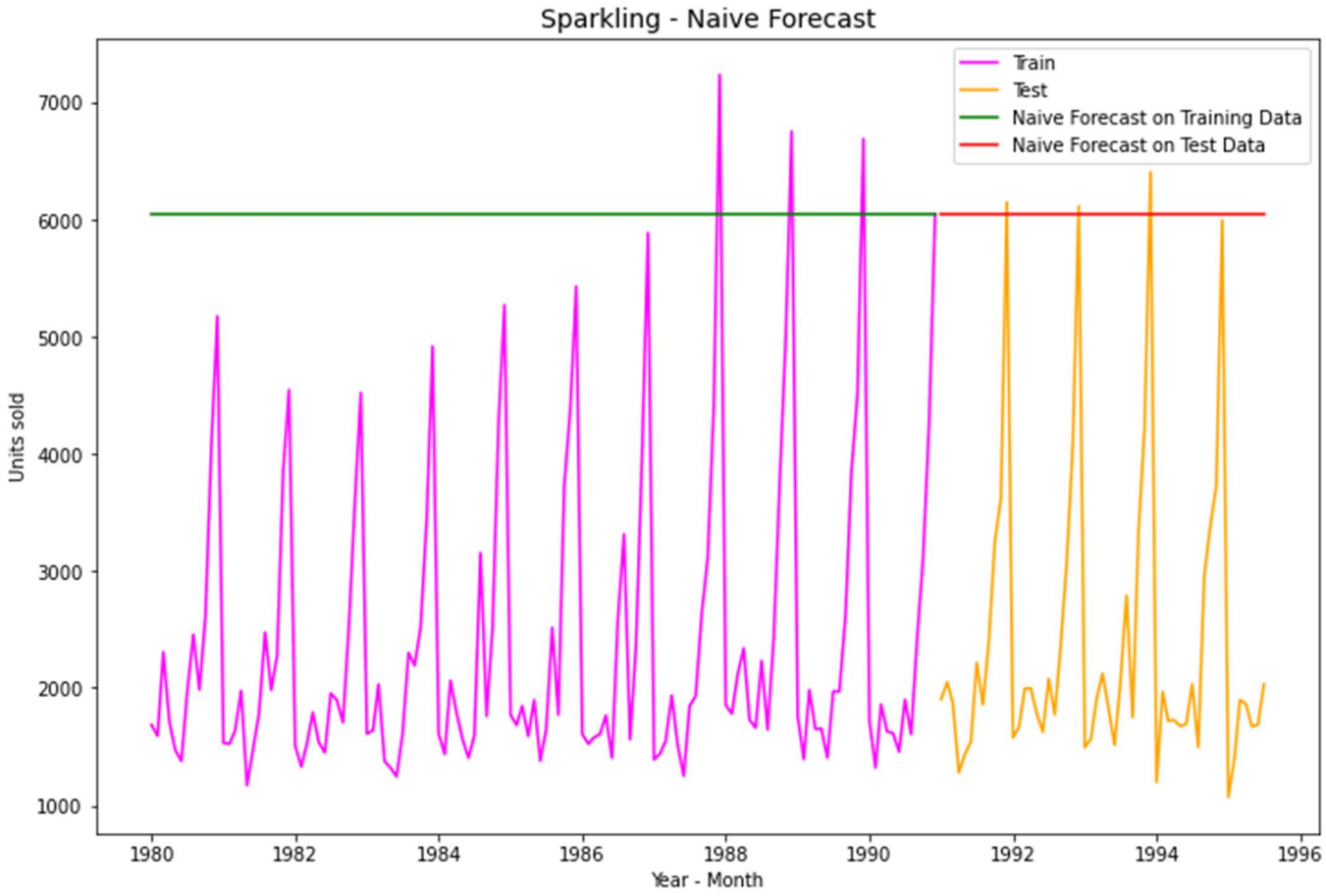


Figure 12 - Naïve forecast model

Observations:

- The model has taken the last value from the test set and fitted it on the rest of the train time period and used the same value to forecast the test set.
- For Naive forecast on the Test Data, RMSE is 3864.279
- The model does not capture the trend or seasonality for the given dataset.

Model 3: Simple Average

In the Simple Average model, the forecast is done using the mean of the time-series variable from the training set.

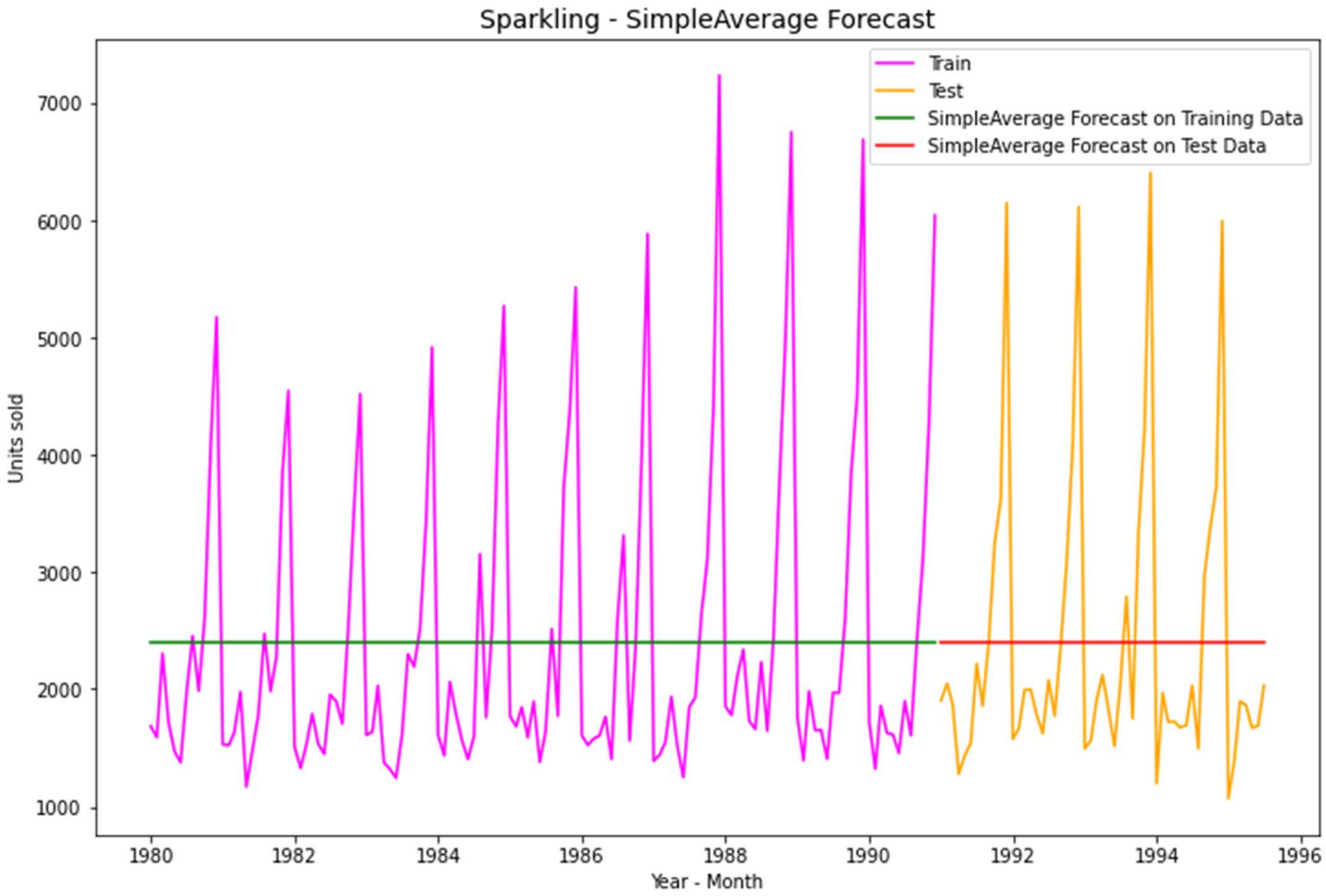


Figure 13 - Simple Average model

Observations:

- The model is not capable of either forecasting or able to capture the trend and seasonality present in the dataset.
- For Simple Average on the Test Data, RMSE is 1275.081

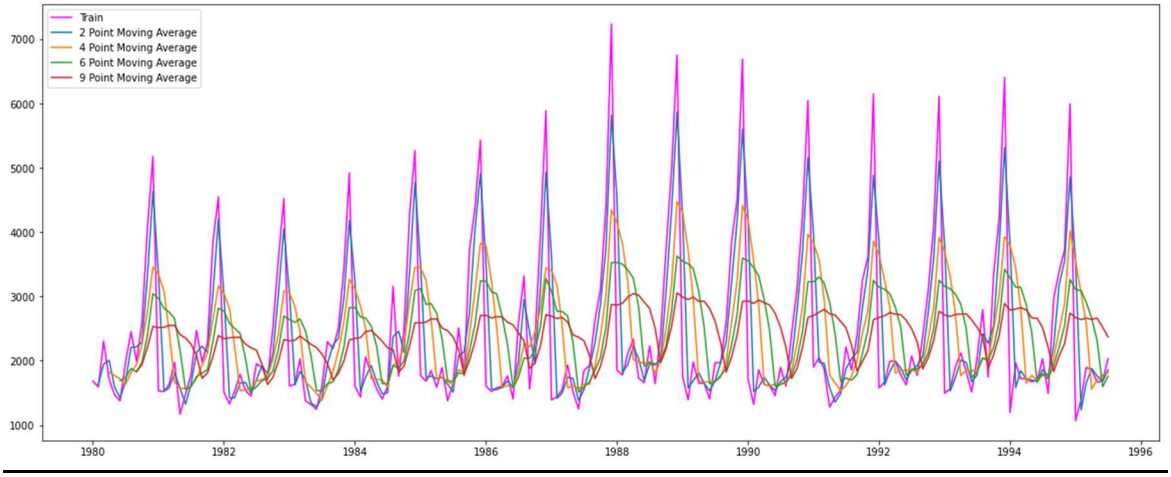
Model 4: Moving Average

- For the moving average model, we will calculate rolling means (or trailing moving averages) for different intervals. The best interval can be determined by the maximum accuracy.
- The moving average models are built for trailing 2 points, 4 points, 6 points and 9 points.

- For Sparkling dataset, the accuracy is found to be higher with the lower rolling point averages.
- In moving average forecasts the values can be fitted with a delay of n number of points.
- The best interval of moving average from the model is 2 points.

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

Plotting on Train Data:



Plotting on Test Data:

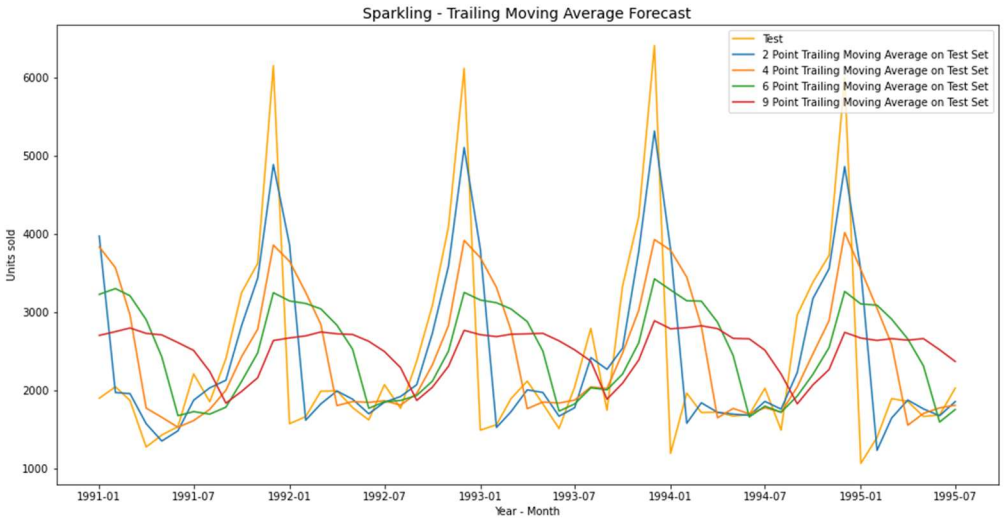


Figure 14: Moving Average on Train and Test data

Sparkling - Trailing Moving Average Forecast

- **RMSE VALUES :**

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315

RMSE for Test Data

Before we go on to build the various Exponential Smoothing models, let us plot all the models and compare the Time Series

Model Comparison

Model 5: Simple Exponential Smoothing

- This model will run without passing a value for alpha and used parameters: 'optimized=True, use brute=True'.
- The auto-fit model picked up alpha = 0.0496 as the smoothing parameter.
- Simple Exponential Smoothing is applied if the time-series has neither a trend nor seasonality, which is not the case with the given data.

- The forecasting using smoothing levels of alpha between 0 and 1 are as below, where the smoothing levels are passed manually.
- For alpha value closer to 1, forecasts follow the actual observation closely and closer to 0, forecasts are farther from actual and line gets smoothened.
- For Sparkling, test RMSE is found to be higher for values closer to zero, which is same as in Simple average forecast.
- By passing manual alpha values, alpha =0.025 gives a better RMSE compared to optimized RMSE value.

The output parameters for optimized model are:

```
{'smoothing_level': 0.049607360581862936,
 'smoothing_trend': nan,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 1818.535750008871,
 'initial_trend': nan,
 'initial_seasons': array([], dtype=float64),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Viewing the first five Predictions for Test Data:

YearMonth	Sparkling	predict
1991-01-01	1902	2724.932624
1991-02-01	2049	2724.932624
1991-03-01	1874	2724.932624
1991-04-01	1279	2724.932624
1991-05-01	1432	2724.932624

Plotting on both the Training and Test data:

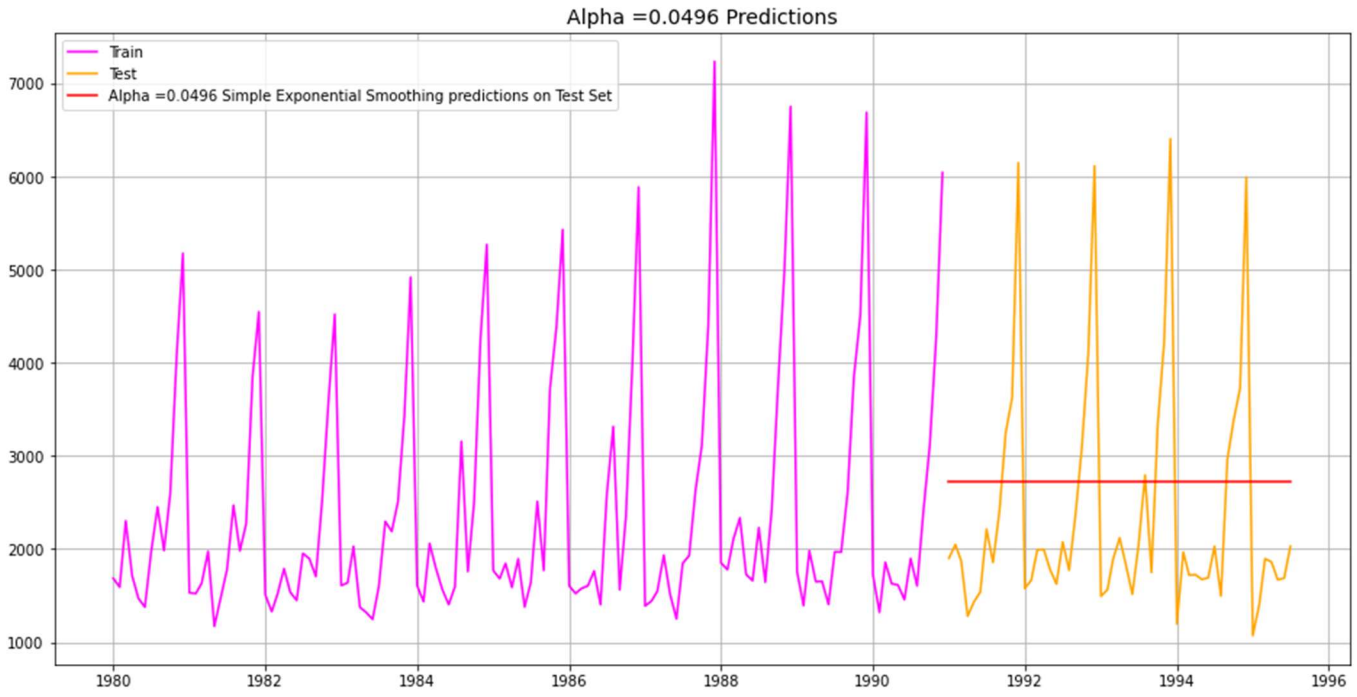


Figure 15: Simple Exponential Smoothing Model

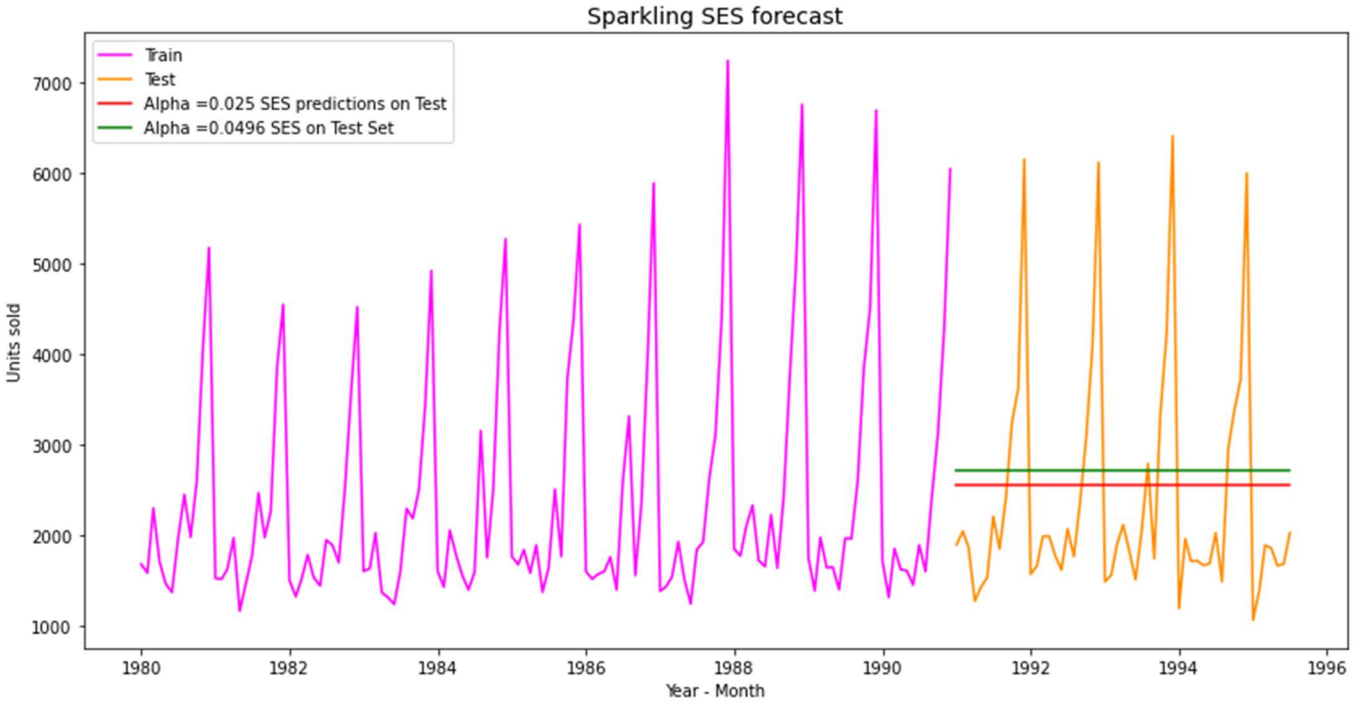
RMSE VALUES :

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576708
Auto_SARIMA_log(0, 1, 1)*(1, 0, 1, 12)	336.799059
Manual_ARIMA(0,1,0)	4779.154299
Manual_SARIMA#(3,1,1)*(1,1,2,12)	324.104370

Model Evaluation based on Iterations:

	Alpha Values	Train RMSE	Test RMSE
0	0.025	1322.084340	1286.248846
1	0.050	1318.429335	1316.411742
3	0.100	1333.873836	1375.393398
4	0.200	1356.042987	1595.206839
2	0.250	1359.701408	1755.488175

SES Optimized and Iterative Model



Model 6: Double Exponential Smoothing (Holt's Model):

- The Double Exponential Smoothing models is applicable when data has trend, but no seasonality. Sparkling data contain slight trend component and very significant seasonality.
- In first iteration, smoothing level (alpha) and trend (beta) are fitted to the model iteratively from values 0.1 to 1 and the best combination was chosen based on the RMSE values, which is as below with alpha 0.1 and beta 0.1 .
- On the second iteration the model was allowed to choose the optimized values using parameters 'optimized=True, use brute='True' .
- The auto-fit model retuned higher RMSE value compared to iterative alpha=0.1 and beta=0.1 RMSE value.

The output parameters for optimized model are:

```
{'smoothing_level': 0.6885714285714285,
 'smoothing_trend': 9.999999999999999e-05,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 1686.0,
 'initial_trend': -95.0,
 'initial_seasons': array([], dtype=float64),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Viewing the first five Predictions for Test Data:

	Sparkling	(predict_spark, 0.6885714285714285, 9.999999999999999e-05)
YearMonth		
1991-01-01	1902	5221.278699
1991-02-01	2049	5127.886554
1991-03-01	1874	5034.494409
1991-04-01	1279	4941.102264
1991-05-01	1432	4847.710119

Plotting on both the Training and Test data:

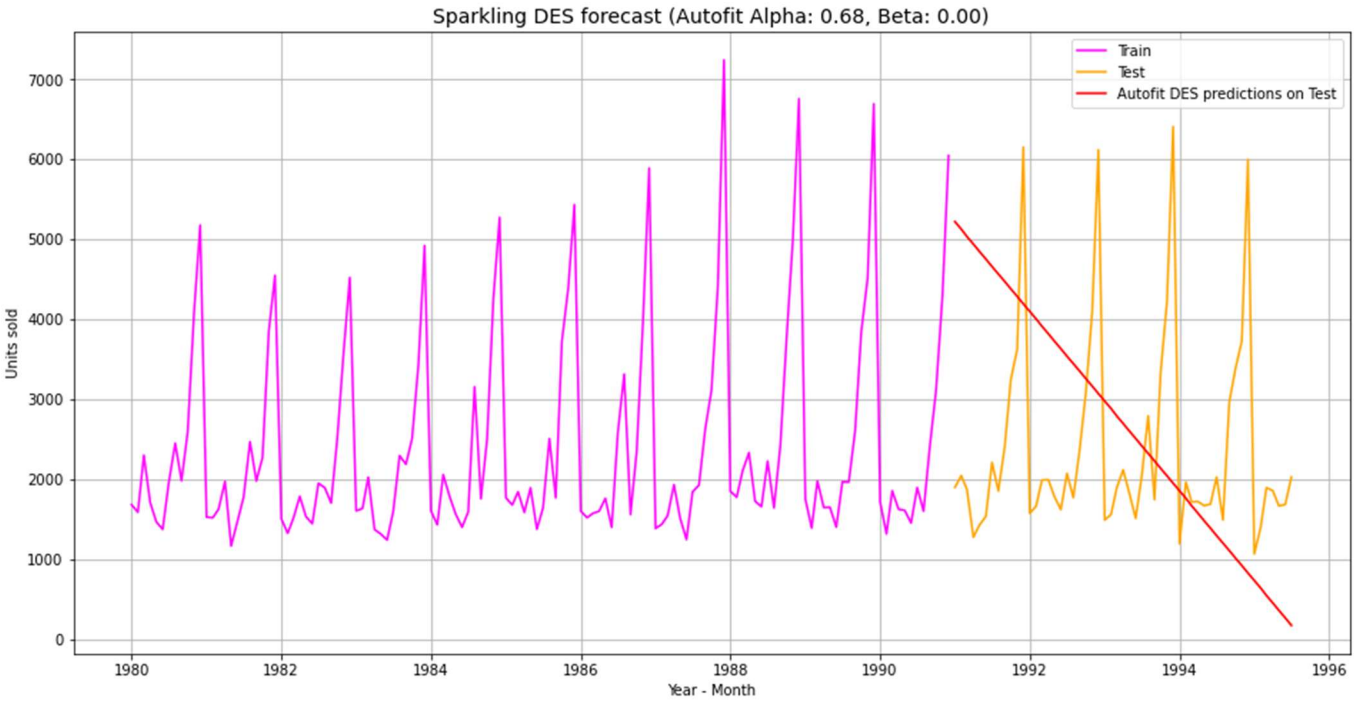


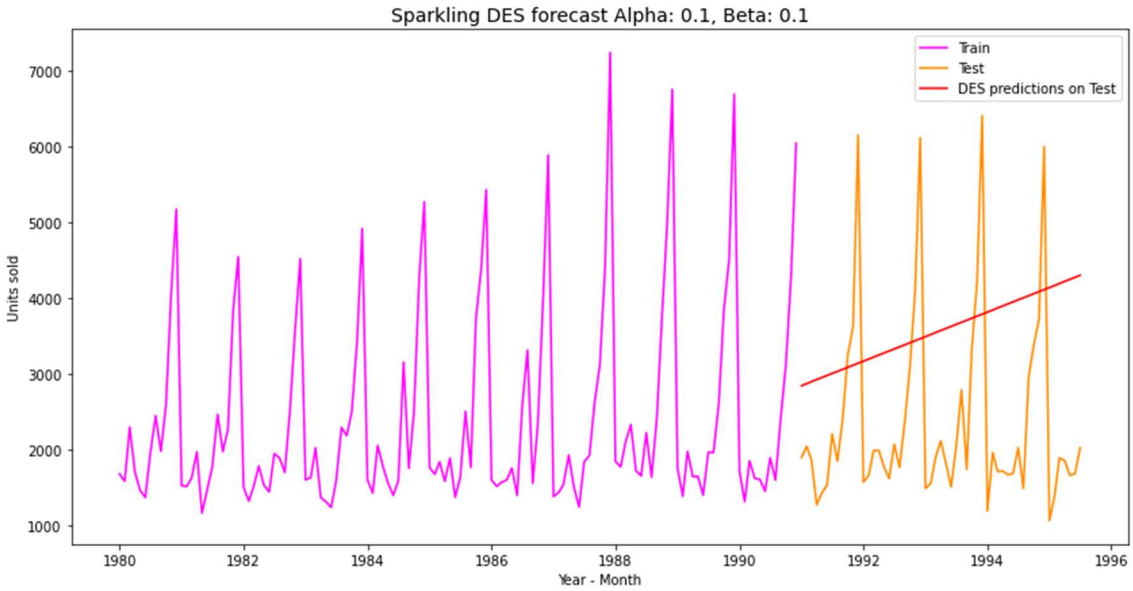
Figure 16: Double Exponential Smoothing Iterative Model

For Auto-fit Double Exponential Smoothing Model forecast on the Test Data, RMSE is 2007.24

Model Evaluation based on Iterations:

	Alpha Values	Beta Values	Train RMSE	Test RMSE
0	0.1	0.1	1382.52	1778.56
1	0.1	0.2	1413.60	2599.44
10	0.2	0.1	1418.04	3611.76
2	0.1	0.3	1445.76	4293.08
20	0.3	0.1	1431.17	5908.19

DES Iterative Model



- **RMSE VALUES :**

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000

Model 7: Triple Exponential Smoothing (Holt - Winter's Model)

- The Triple Exponential Smoothing models (Holt-Winter's Model) is applicable when data has both trend and seasonality. Sparkling data contain slight trend and significant seasonality .
- On first iteration, smoothing level (alpha), trend (beta) and seasonality (gamma) are fitted to the model iteratively from values 0.1 to 1 and the best combination was chosen based on the RMSE values, which is as below with alpha 0.4, beta 0.1 and gamma 0.3.
- On the second iteration the model was allowed to choose the optimized values using parameters 'optimized=True, use brute='True'.
- The auto-fit model retuned higher RMSE value compared to iterative alpha=0.4, beta=0.1 and gamma=0.3 RMSE value.

The output parameters for optimized model are:

```
{'smoothing_level': 0.111108139467838,
 'smoothing_trend': 0.06172875597197263,
 'smoothing_seasonal': 0.3950479631147446,
 'damping_trend': nan,
 'initial_level': 1639.9340657558994,
 'initial_trend': -12.22494561218149,
 'initial_seasons': array([1.06402008, 1.02352078, 1.40671876, 1.20165543, 0.97593
       0.97100155, 1.31897446, 1.69588922, 1.3895294 , 1.81476396,
       2.85150039, 3.62470528]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Plotting on both the Training and Test data:

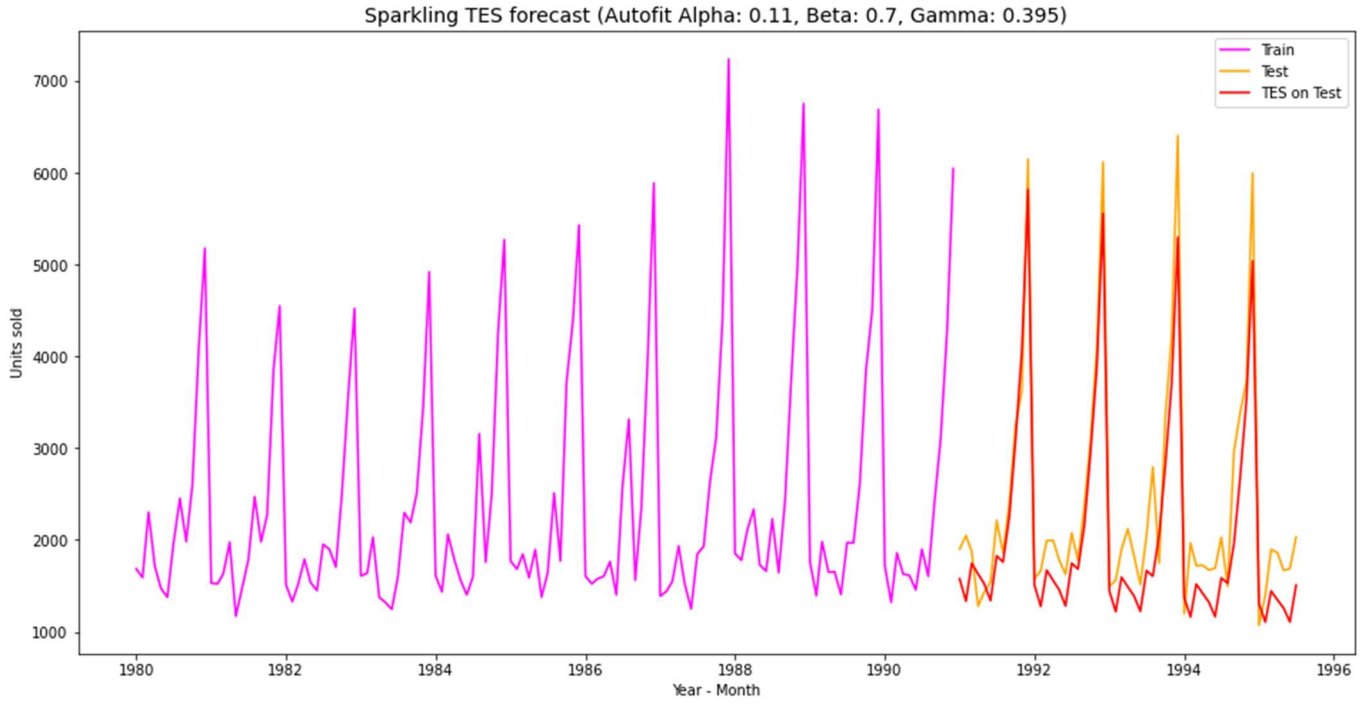


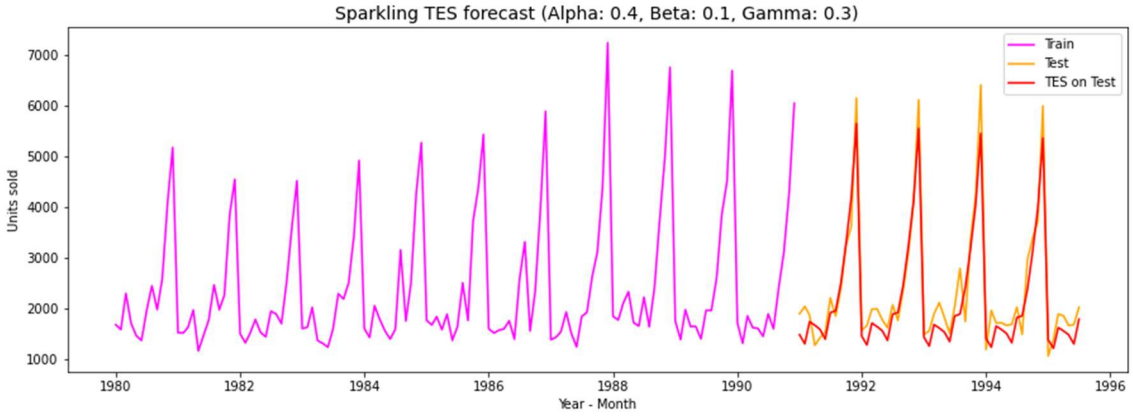
Figure 17: Triple Exponential Smoothing Optimized Model

For Auto-fit Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 469.768

Model Evaluation based on Iterations:

	Alpha Values	Beta Values	Train RMSE	Test RMSE	Gamma Values
240	0.4	0.1	387.990141	371.367690	0.3
321	0.5	0.1	409.863151	379.651522	0.4
320	0.5	0.1	401.756285	379.852675	0.3
64	0.1	0.9	435.461755	392.102406	0.3
176	0.3	0.3	404.513320	392.786198	0.3

TES Iterative Model



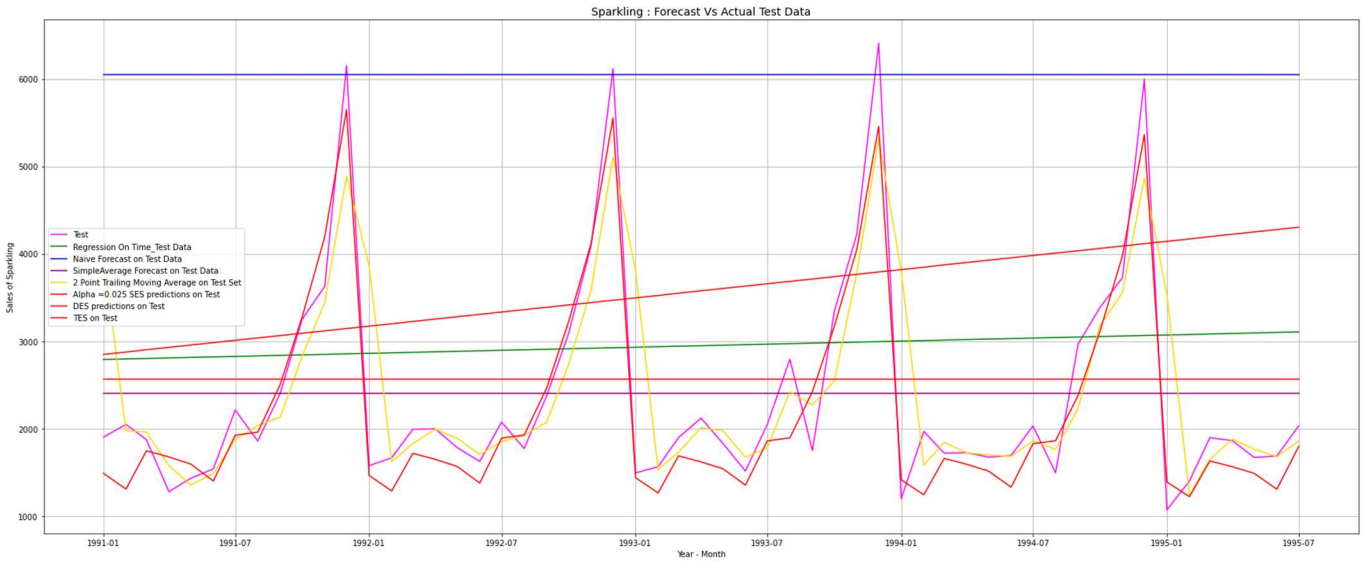
Model Comparison:

- **RMSE VALUES ON TEST DATA :**

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690

Table 1: Model comparison based on various parameters

Sparkling forecast v/s Actual Values



Inferences:

- From the comparison of accuracy values and the plot it can be inferred that Triple Exponential Smoothing is the best model, which has trend as well as seasonality components fitting well with the test data.
- 2-point trailing moving average model is also found to have fit well with a slight lag in test dataset.

1.5 Check for the stationarity of the data on which the model is being built on using appropriate statistical tests and also mention the hypothesis for the statistical test. If the data is found to be non-stationary, take appropriate steps to make it stationary. Check the new data for stationarity and comment.

Solution:

Stationarity should be checked at alpha = 0.05

- Augmented Dickey Fuller test is the statistical test to check the stationarity of a time series. The test determines the presence of unit root in the series to understand if the series is stationary or not.
- Null Hypothesis: The series has a unit root, that is series is non-stationary → Alternate Hypothesis: The series has no unit root, that is series is stationary.
- If we fail to reject the null hypothesis, it can say that the series is non-stationary and if we accept the null hypothesis, it can say that the series is stationary.
- The ADF test on the original Sparkling series retuned the below values, where p-value is greater than alpha .05 so we fail to reject the null hypothesis.

Original Time Series:

```
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
```

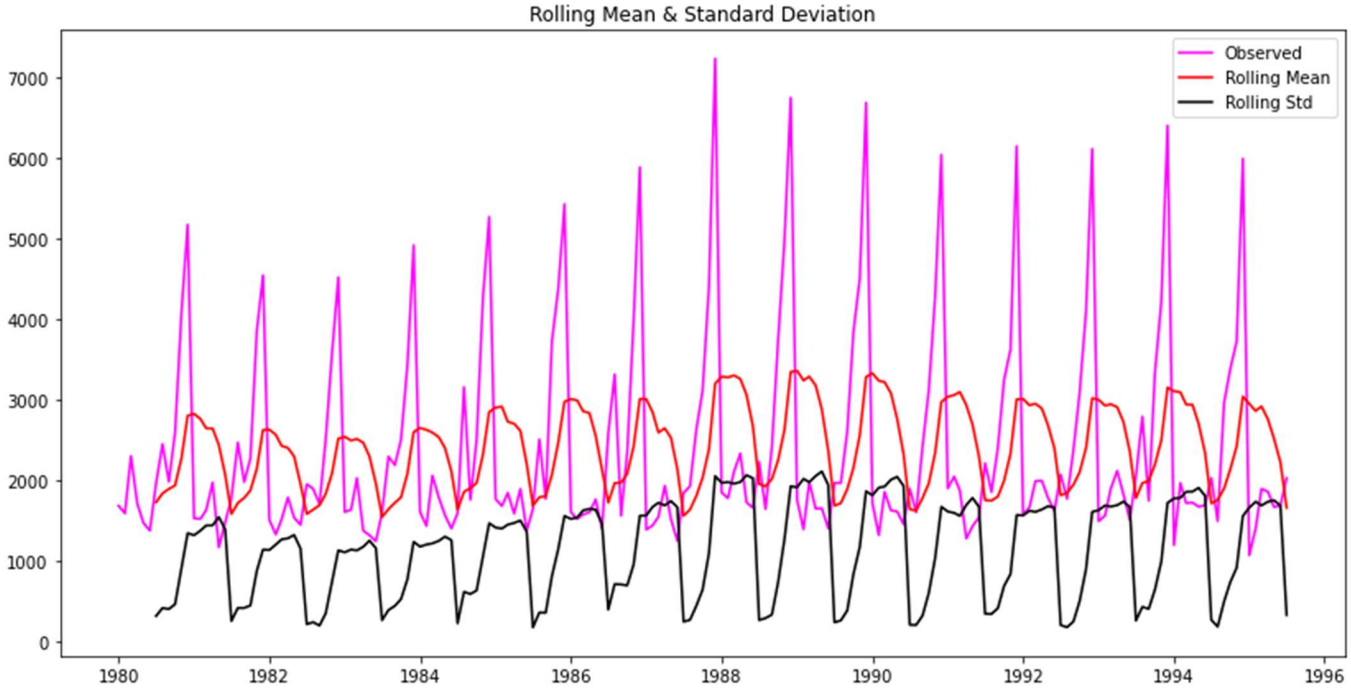


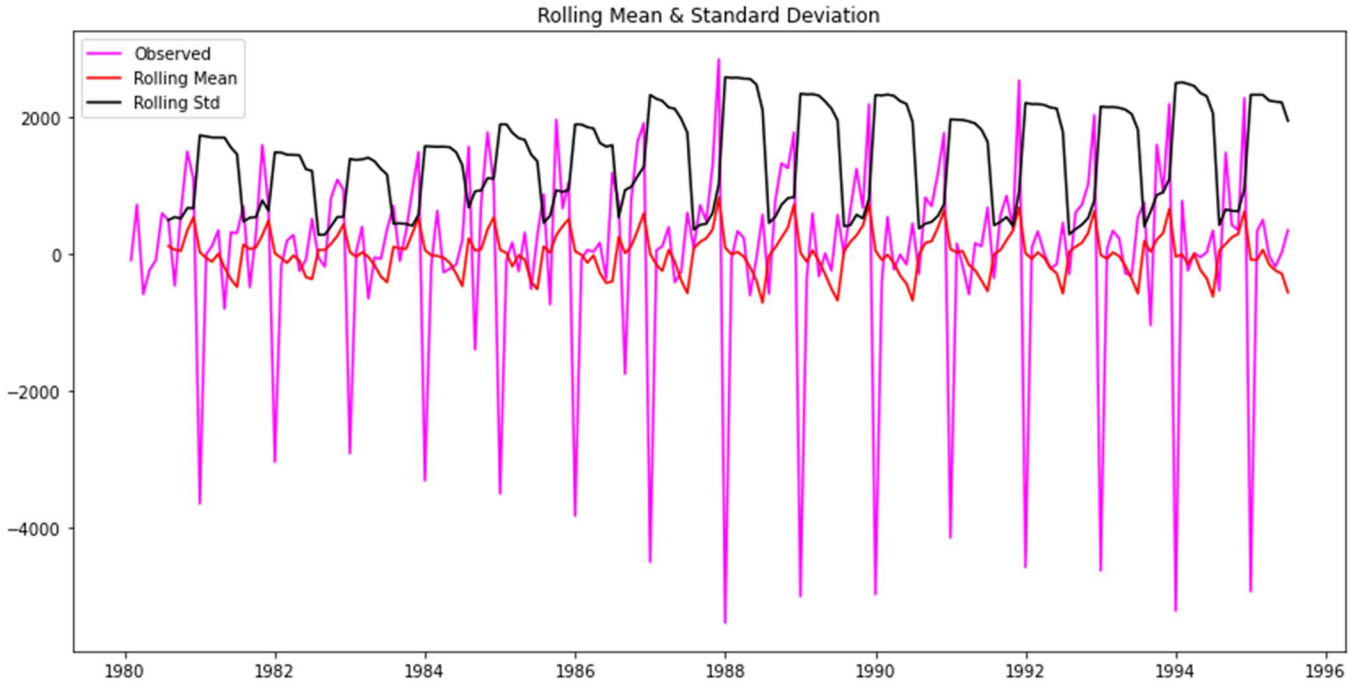
Figure 18: ADF Test on Original Series

Observations:

- Differencing of order one is applied on the Sparkling series as below and tested for stationarity. At an order of differencing 1, the series is found to be stationary as below.
- The rolling means and standard deviation is also plotted to understand the component of seasonality and to ascertain if it's multiplicative or additive in character.
- The altitude of rolling mean and std dev is seen changing according to change in slope, which indicates multiplicity.
- The ADF test is also done in this exercise with logarithmic transformation of the train data and differencing of seasonal order (12), to understand if removing the multiplicity of the seasonal component will have an impact on the accuracy of model.

ADF test after differencing d=1

```
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
```



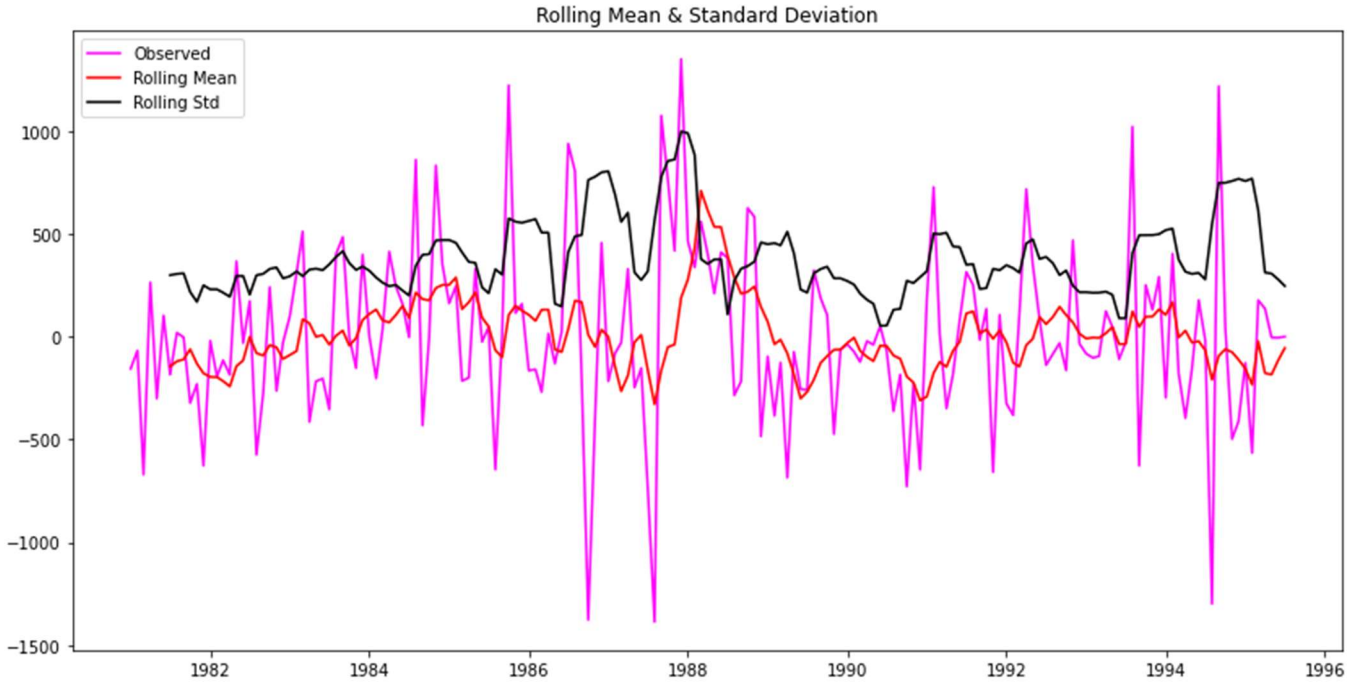
We see that at alpha= 0.05 the Time Series is indeed stationary when d=1

If the series is non-stationary, standardized the Time Series by taking a difference of the Time Series. Then we can use this particular differenced series to train the ARIMA/SARIMA models. Also, we can look at other kinds of transformations as part of making the time series stationary like taking logarithms.

Checking Seasonality Time Series:

```
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
```

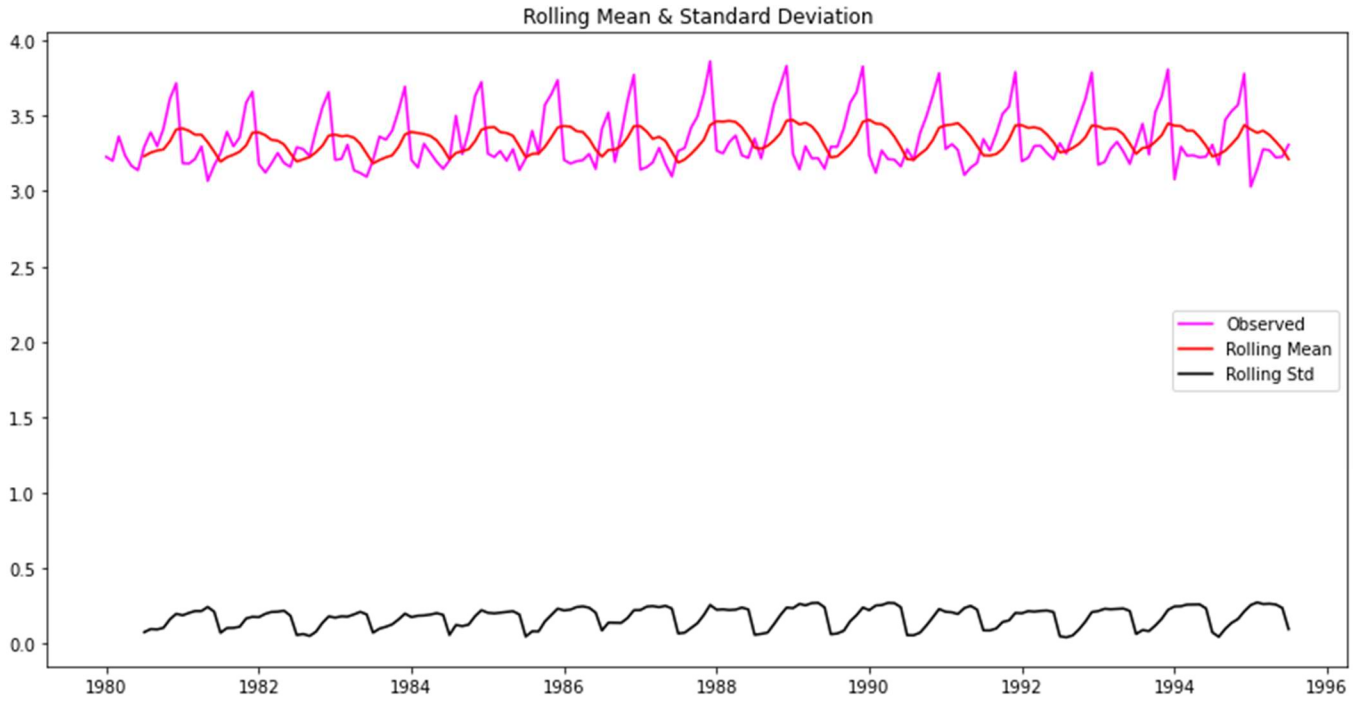
ADF test for Seasonality



We see that at alpha = 0.05 the Time Series is indeed stationary. But seasonality is multiplicative.

Taking Log series:

```
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
```



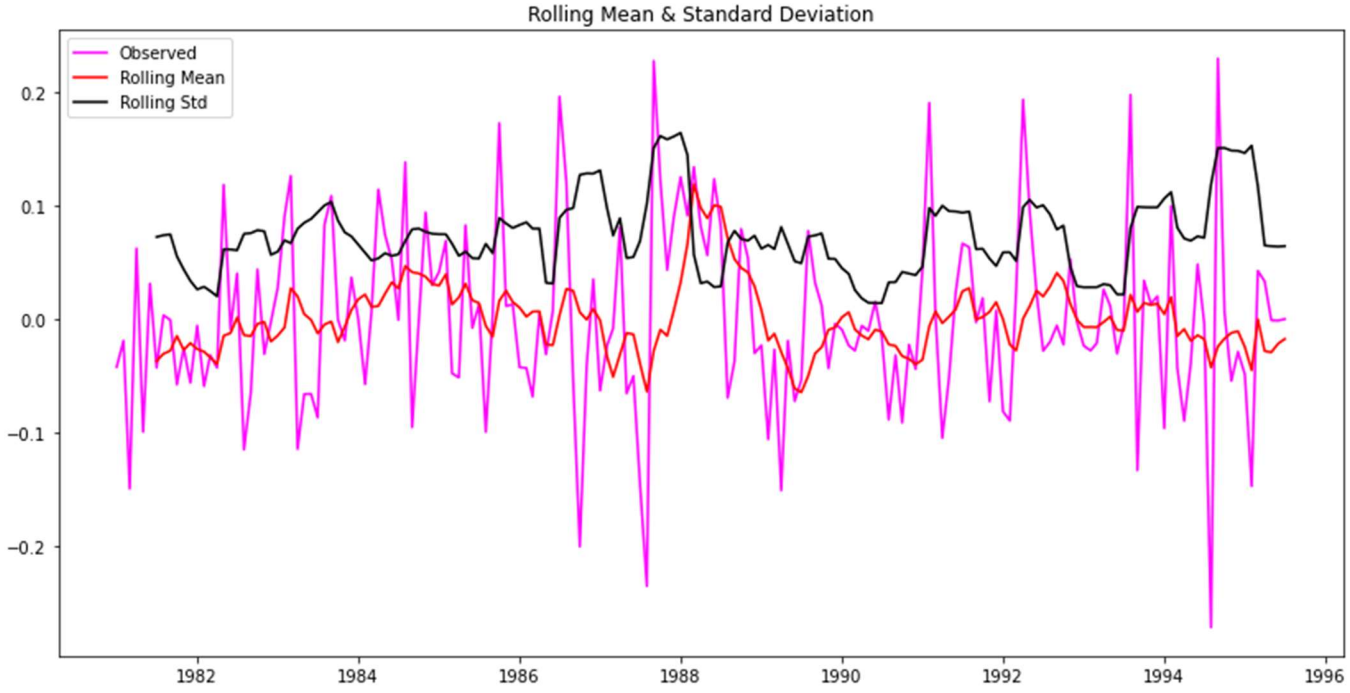
Inference:

- Seasonality is now additive but non-stationary

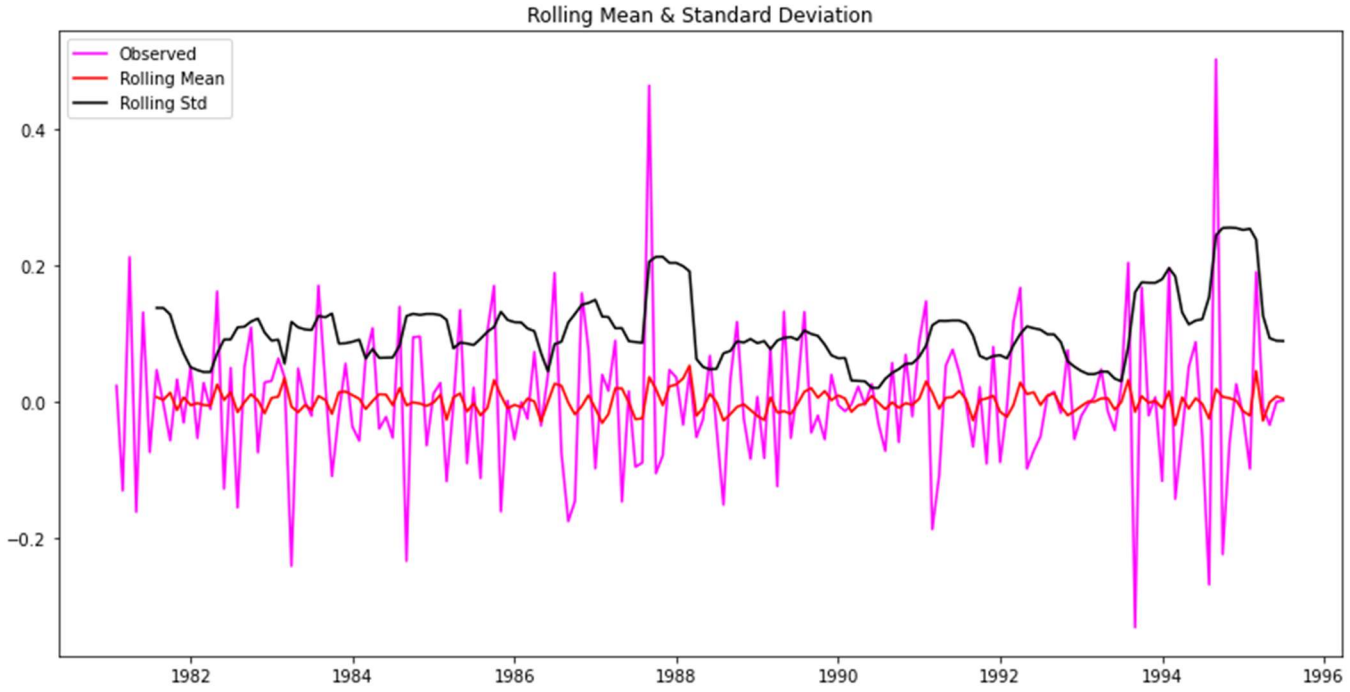
Difference of log series:

```
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
```

ADF Test on log series after differencing




```
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
```



ACF PLOTS:

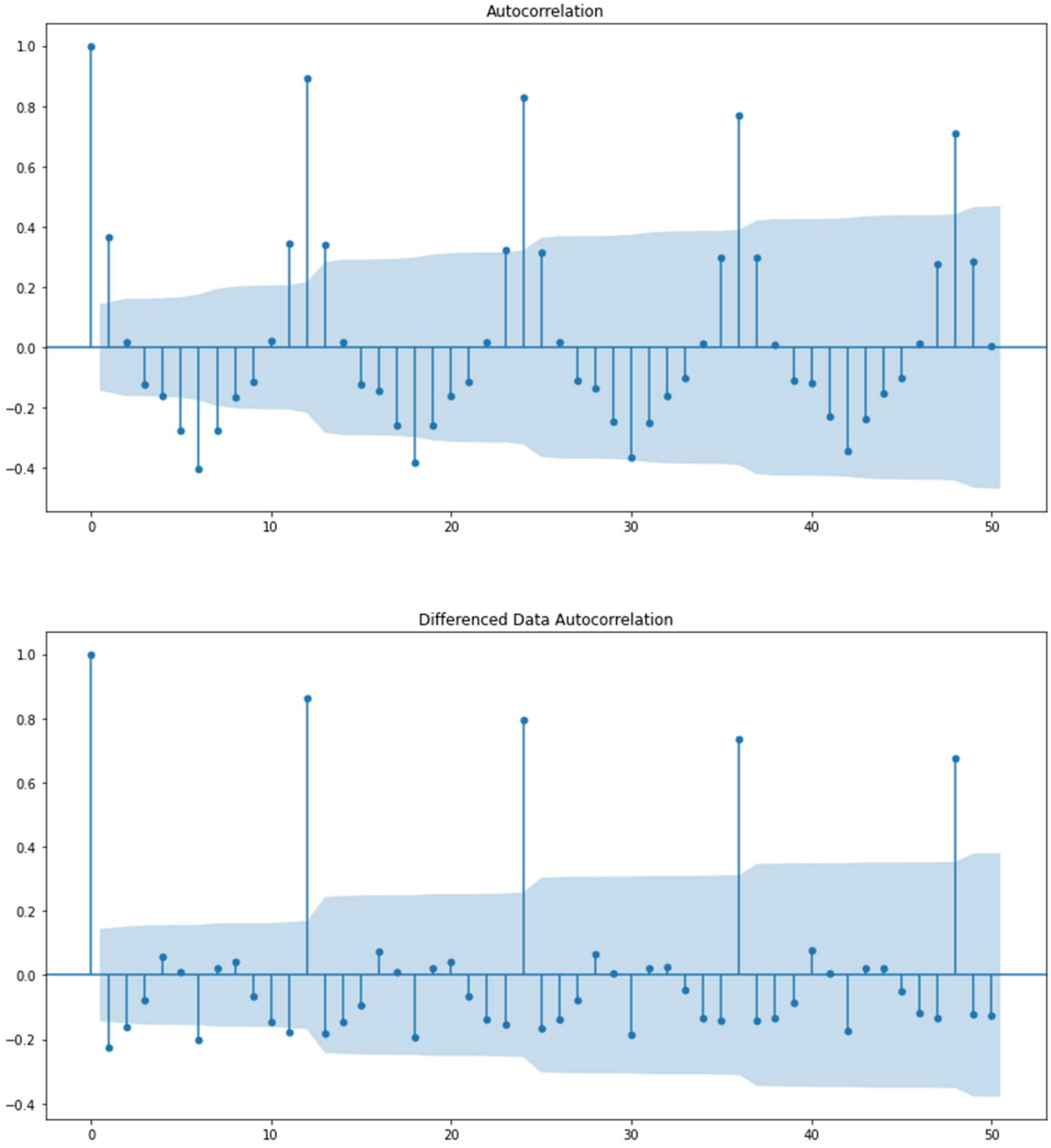


Figure 19: Autocorrelation and Differenced Data Autocorrelation of Sparkling dataset

PACF PLOTS:

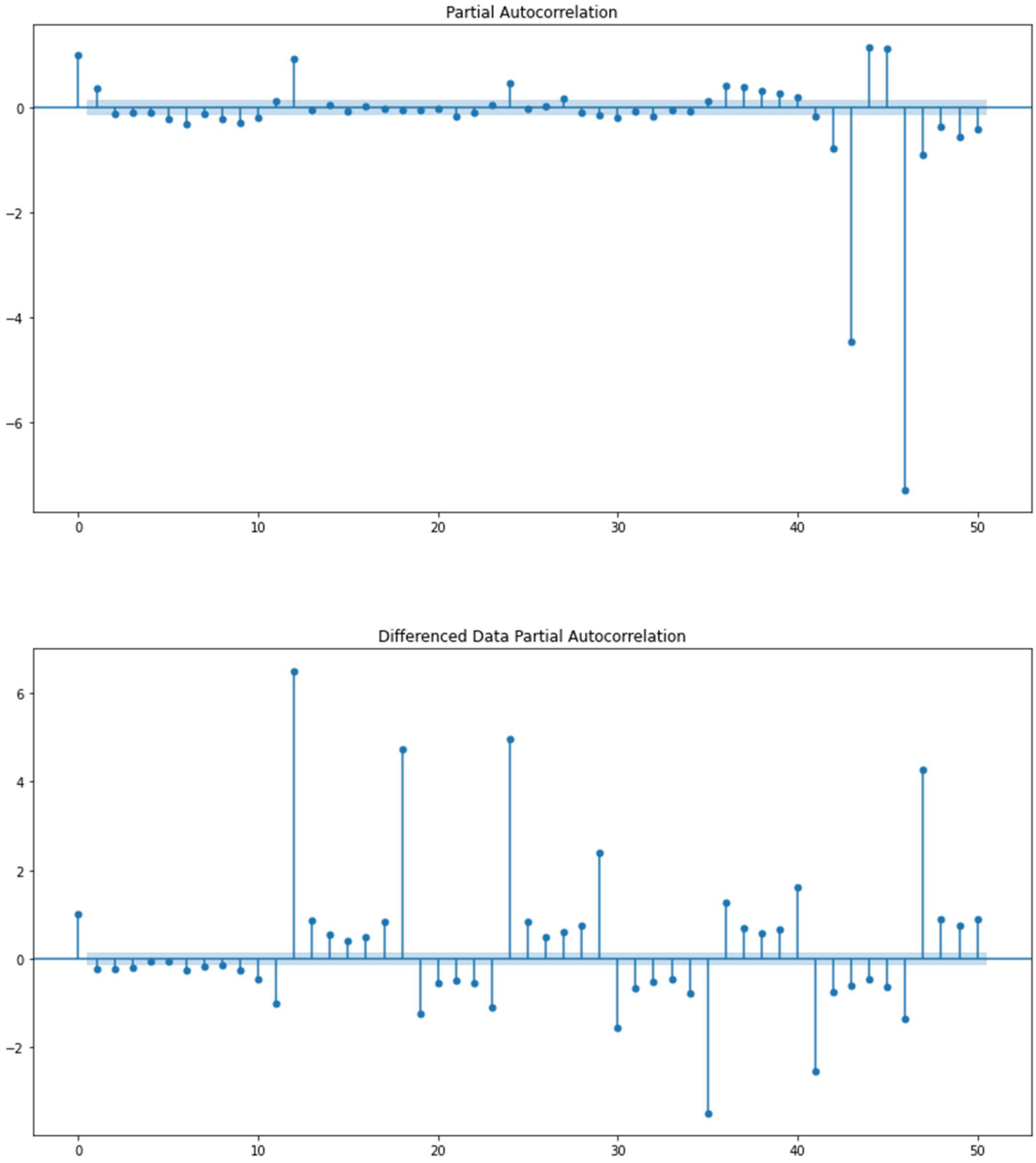
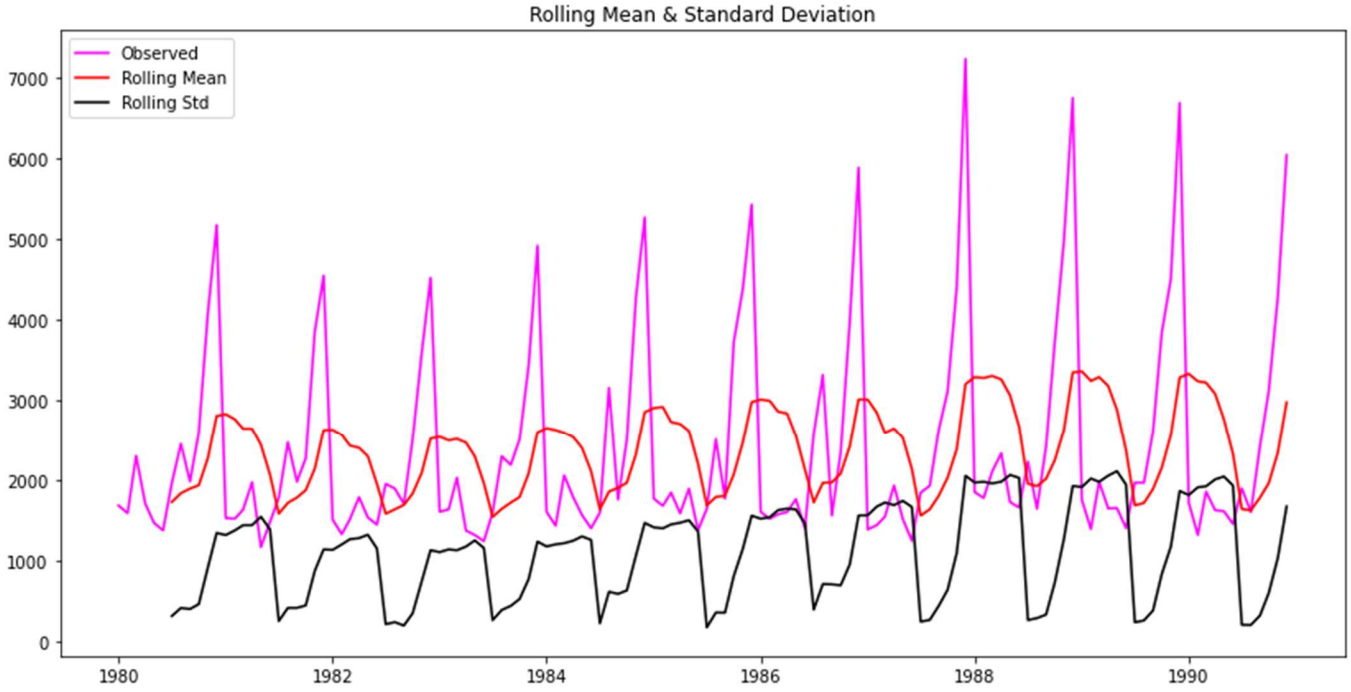


Figure 20: Partial Autocorrelation and Differenced Data Partial Autocorrelation of Sparkling dataset

From the above plots, we can say that there seems to be a seasonality in the data.

Check for stationarity of the Training Data Time Series:

```
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
```



ADF Test on Train Data

Inferences:

- We see that at 5% significant level the Time Series is non-stationary.
- Let us take a difference of order 1 and check whether the Time Series is stationary or not.

```
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
```

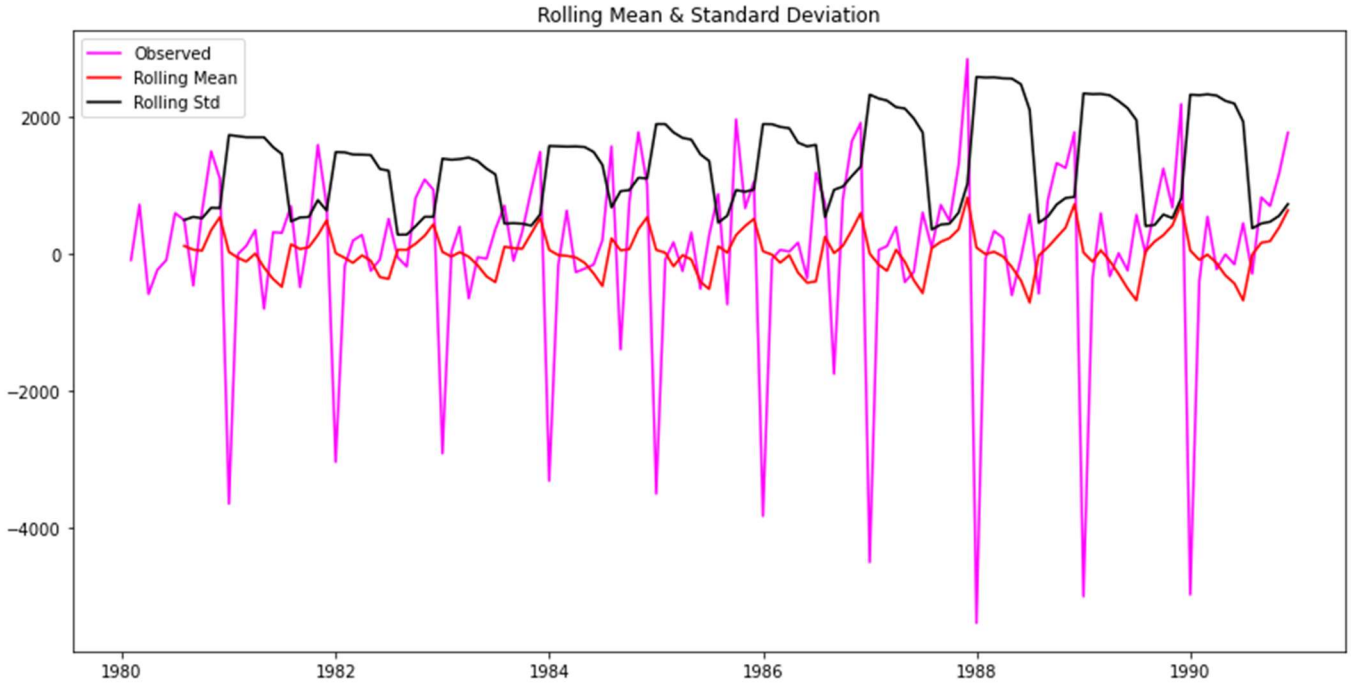


Figure 21: Stationarity of Training Data Time Series before and after differencing

1.6. Build an automated version of the ARIMA/SARIMA model in which the parameters are selected using the lowest Akaike Information Criteria (AIC) on the training data and evaluate this model on the test data using RMSE.

Solution:

Auto-ARIMA Model:

The following loop helps us in getting a combination of different parameters of p and q in the range of 0 and 2. We have kept the value of d as 1 as we need to take a difference of the series to make it stationary.
Some parameter combinations for the Model...

```
Model: (0, 1, 1)
Model: (0, 1, 2)
Model: (1, 1, 0)
Model: (1, 1, 1)
Model: (1, 1, 2)
Model: (2, 1, 0)
Model: (2, 1, 1)
Model: (2, 1, 2)
```

AIC values are sorted in the ascending order to get the parameters for the minimum AIC value.

	param	AIC
8	(2, 1, 2)	2210.626049
7	(2, 1, 1)	2232.360490
2	(0, 1, 2)	2232.783098
5	(1, 1, 2)	2233.597647
4	(1, 1, 1)	2235.013945
6	(2, 1, 0)	2262.035600
1	(0, 1, 1)	2264.906439
3	(1, 1, 0)	2268.528061
0	(0, 1, 0)	2269.582796

ARIMA Model Results

Dep. Variable:	D.Sparkling	No. Observations:	131
Model:	ARIMA(2, 1, 2)	Log Likelihood	-1099.313
Method:	css-mle	S.D. of innovations	1013.755
Date:	Wed, 13 Jul 2022	AIC	2210.626
Time:	09:23:47	BIC	2227.877
Sample:	02-01-1980 - 12-01-1990	HQIC	2217.636

	coef	std err	z	P>|z|	[0.025	0.975]
const	5.5845	0.519	10.753	0.000	4.567	6.602
ar.L1.D.Sparkling	1.2698	0.075	17.040	0.000	1.124	1.416
ar.L2.D.Sparkling	-0.5601	0.074	-7.617	0.000	-0.704	-0.416
ma.L1.D.Sparkling	-1.9957	0.043	-46.821	0.000	-2.079	-1.912
ma.L2.D.Sparkling	0.9957	0.043	23.291	0.000	0.912	1.079

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	1.1335	-0.7074j	1.3361	-0.0888
AR.2	1.1335	+0.7074j	1.3361	0.0888
MA.1	1.0000	+0.0000j	1.0000	0.0000
MA.2	1.0043	+0.0000j	1.0043	0.0000

Figure 22: Auto ARIMA Model Summary Report

Observations:

- ARIMA model was built with optimized model and found the least AIC value =2210.62 at (2, 1, 2).
- As the Sparkling series of data contain seasonality component, ARIMA model do not perform well. The RMSE value for this Auto- ARIMA model is 1374.037

RMSE VALUES ON TEST DATA :

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009

Auto-SARIMA Model:

From ACF and PACF plots We can see that there is a seasonality of 12. We will run our auto SARIMA models by setting seasonality 12.

- The model was built on train data with seasonality 12 and with different optimal parameters (p, d, q)x(P, D, Q) parameters, the lowest AIC is 1382.35 was obtained at (1, 1, 2)x(0, 1, 2, 12). The model was built with the above parameters.

Examples of some parameter combinations for Model...

```
Model: (0, 1, 1)(0, 1, 1, 12)
Model: (0, 1, 2)(0, 1, 2, 12)
Model: (1, 1, 0)(1, 1, 0, 12)
Model: (1, 1, 1)(1, 1, 1, 12)
Model: (1, 1, 2)(1, 1, 2, 12)
Model: (2, 1, 0)(2, 1, 0, 12)
Model: (2, 1, 1)(2, 1, 1, 12)
Model: (2, 1, 2)(2, 1, 2, 12)
```

AIC Values with different parameters sorted in ascending order:

	param	seasonal	AIC
47	(1, 1, 2)	(0, 1, 2, 12)	1382.347780
20	(0, 1, 2)	(0, 1, 2, 12)	1382.484254
50	(1, 1, 2)	(1, 1, 2, 12)	1384.137874
74	(2, 1, 2)	(0, 1, 2, 12)	1384.317618
23	(0, 1, 2)	(1, 1, 2, 12)	1384.398867

Model Evaluation of SARIMA Model:

SARIMAX Results

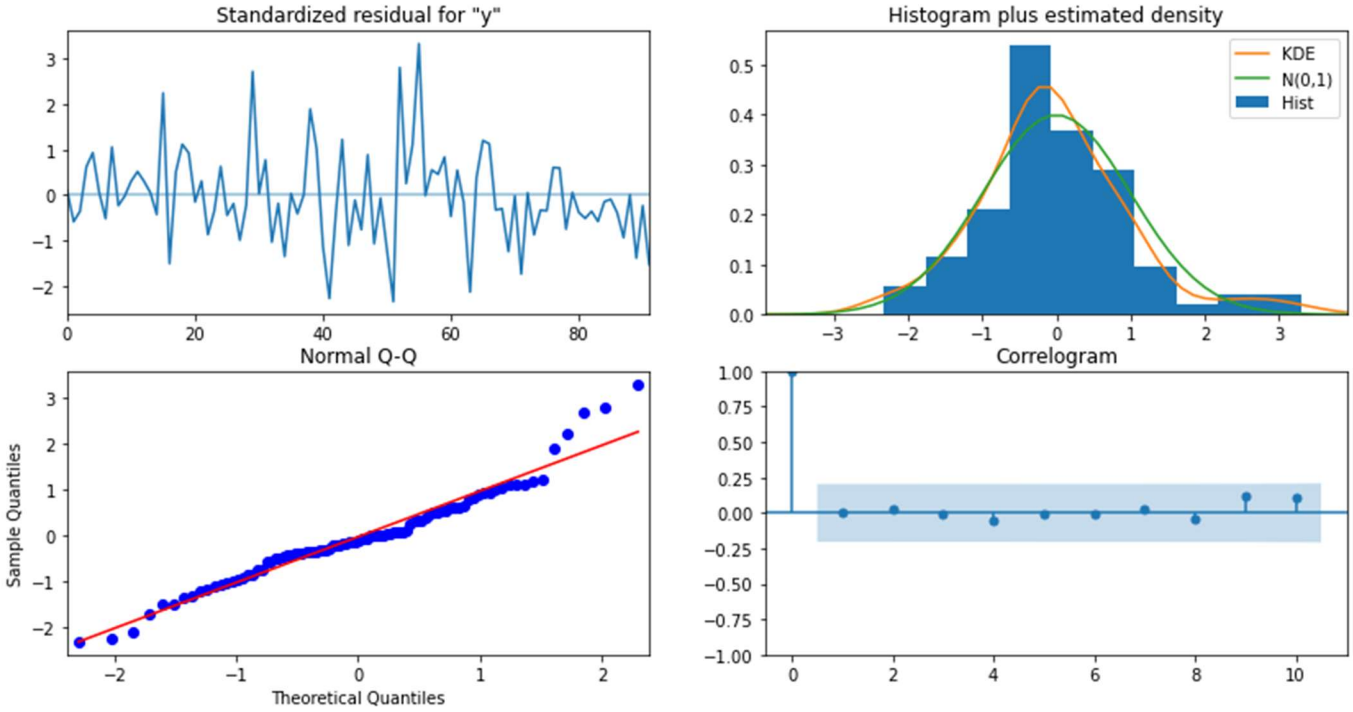
Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(1, 1, 2)x(0, 1, 2, 12)	Log Likelihood	-685.174
Date:	Wed, 13 Jul 2022	AIC	1382.348
Time:	09:34:39	BIC	1397.479
Sample:	0	HQIC	1388.455
	- 132		
Covariance Type:	opg		

	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

Figure 23: SARIMA Model Result

Diagnostic-plot



Observations:

- The diagnostics plot of the model was derived and the standardized residuals are found to follow a mean of zero, and the histogram shows the residuals follow a normal distribution.
- The Normal Q-Q plot also shows that the quantiles come from a normal distribution as the point forms roughly a straight line.
- The correlogram shows the autocorrelation of the residuals and there are no significant lags above the confidence index.
- The RMSE values of the automated SARIMA model is 382.58

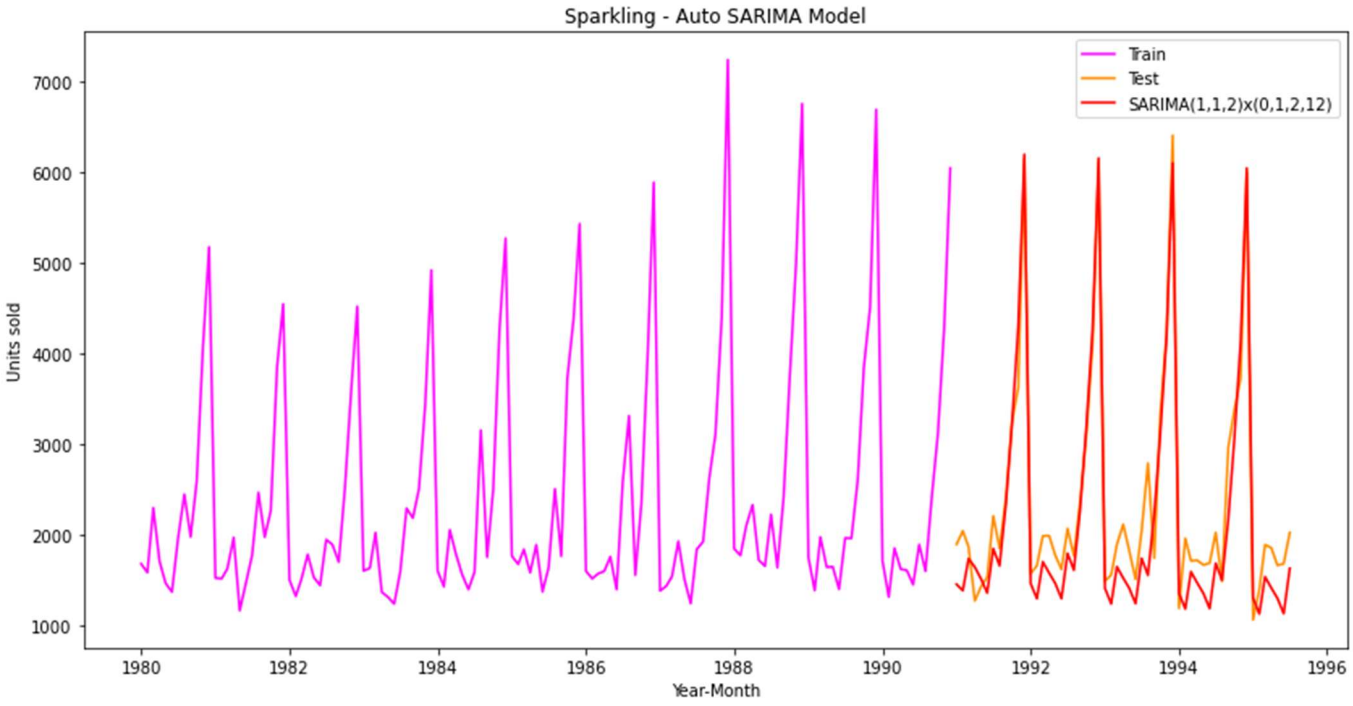
Prediction on the Test Set :

y	mean	mean_se	mean_ci_lower	mean_ci_upper
0	1460.244632	412.922772	650.930870	2269.558393
1	1392.437181	429.721304	550.198901	2234.675460
2	1743.201708	430.065870	900.288092	2586.115323
3	1650.066942	433.930031	799.579709	2500.554174
4	1522.656037	434.242918	671.555558	2373.756516

Extracting the predicted and true values of our time series we get:

YearMonth	Sparkling	spark_forecasted
1991-01-01	1902	1460.244632
1991-02-01	2049	1392.437181
1991-03-01	1874	1743.201708
1991-04-01	1279	1650.066942
1991-05-01	1432	1522.656037

Plot of Actual v/s Forecasted Result on test data



The model was built on log transformed train data and with seasonality 12 and with different optimal parameters (p, d, q)x(P, D, Q) parameters, the lowest AIC is 382.577 was obtained at

(0, 1, 1)*(1, 0, 1, 12). The model was built with the above parameters.

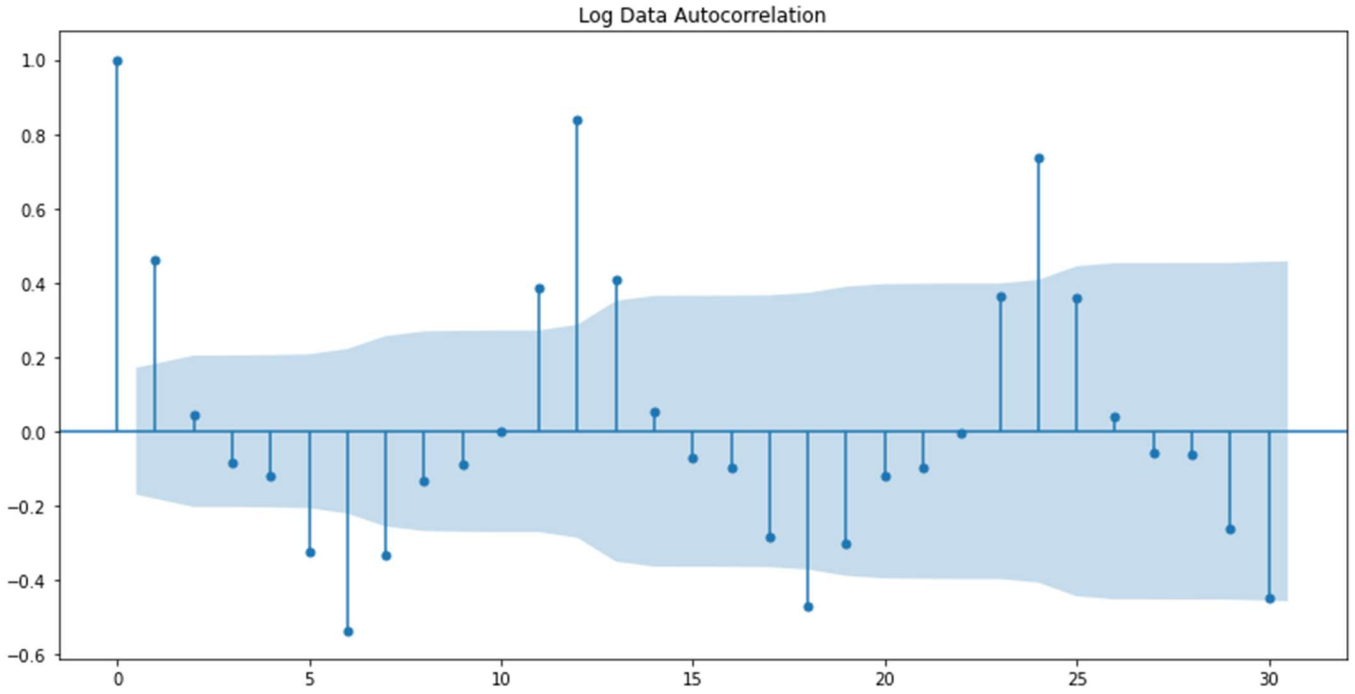
RMSE VALUES ON TEST DATA :

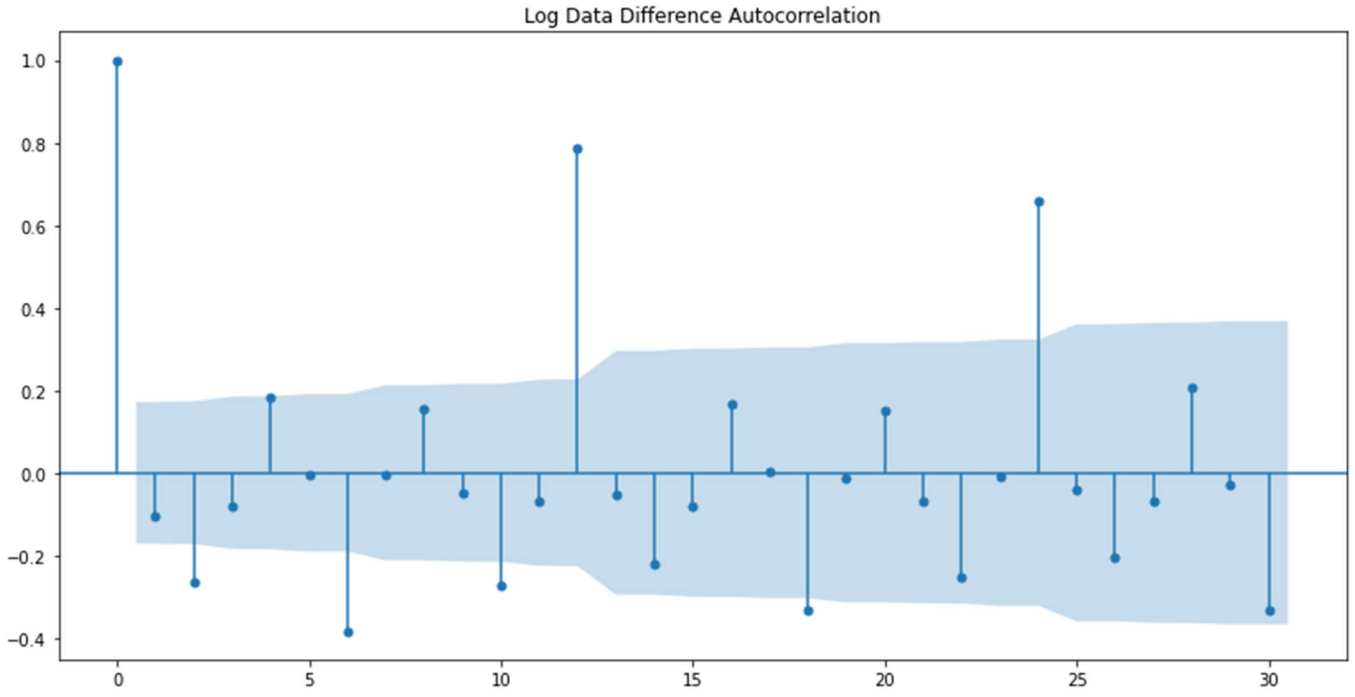
	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576708

AUTO SARIMA on Log Series:

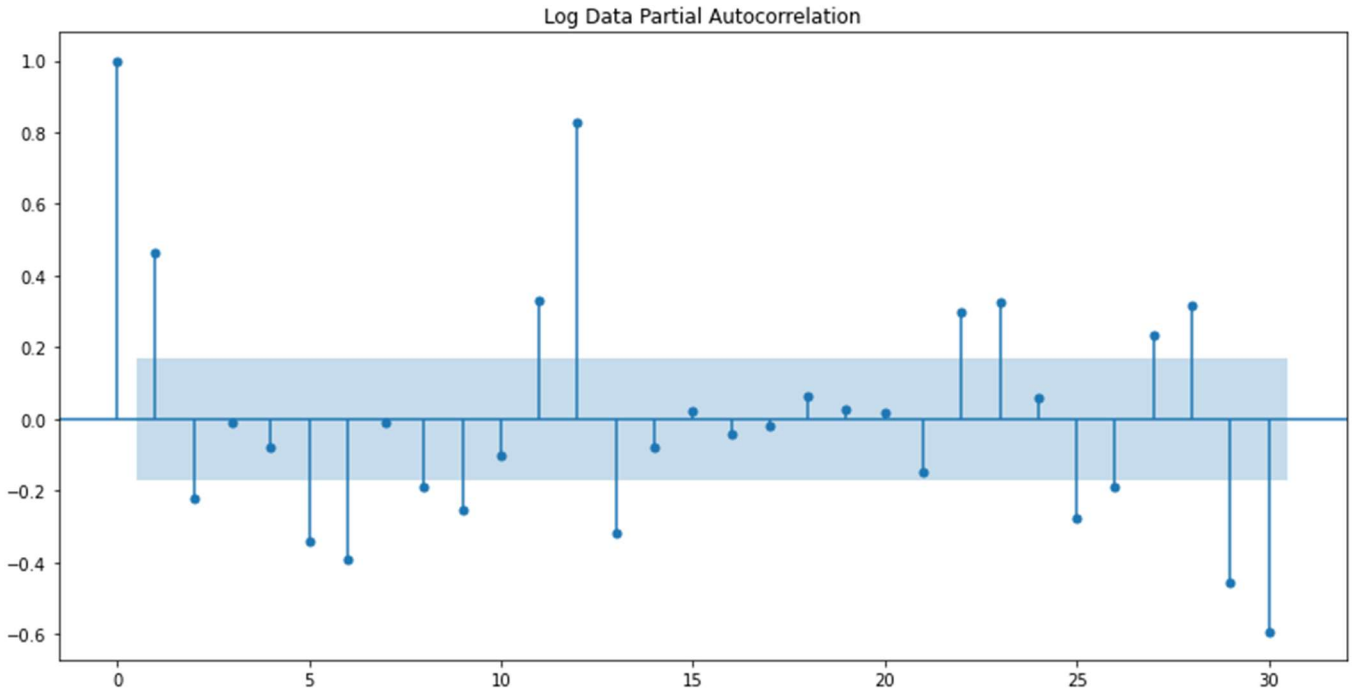
- The model was built on log transformed train data and with seasonality 12 and with different optimal parameters (p, d, q)x(P, D, Q) parameters, the lowest AIC is 336.799 was obtained at (0, 1, 1)*(1, 0, 1, 12).The model was built with the above parameters.

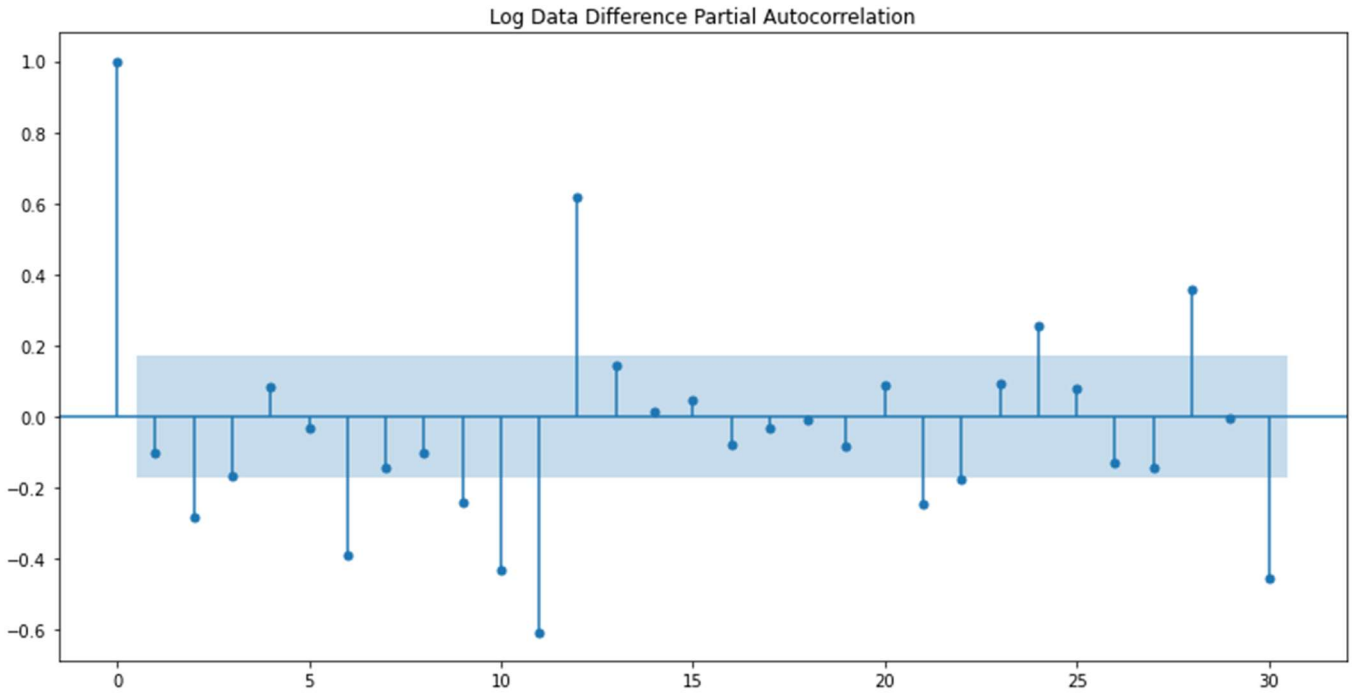
ACF PLOTS:


Log Data Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
5
10
15
20
25
30


Log Data Difference Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
5
10
15
20
25
30

PACF PLOTS:


Log Data Partial Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
5
10
15
20
25
30


Log Data Difference Partial Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
5
10
15
20
25
30

Examples of some parameter combinations for Model...

```
Model: (0, 1, 1)(0, 0, 1, 12)
Model: (0, 1, 2)(0, 0, 2, 12)
Model: (1, 1, 0)(0, 1, 0, 12)
Model: (1, 1, 1)(0, 1, 1, 12)
Model: (1, 1, 2)(0, 1, 2, 12)
Model: (2, 1, 0)(1, 0, 0, 12)
Model: (2, 1, 1)(1, 0, 1, 12)
Model: (2, 1, 2)(1, 0, 2, 12)
```

AIC Values:

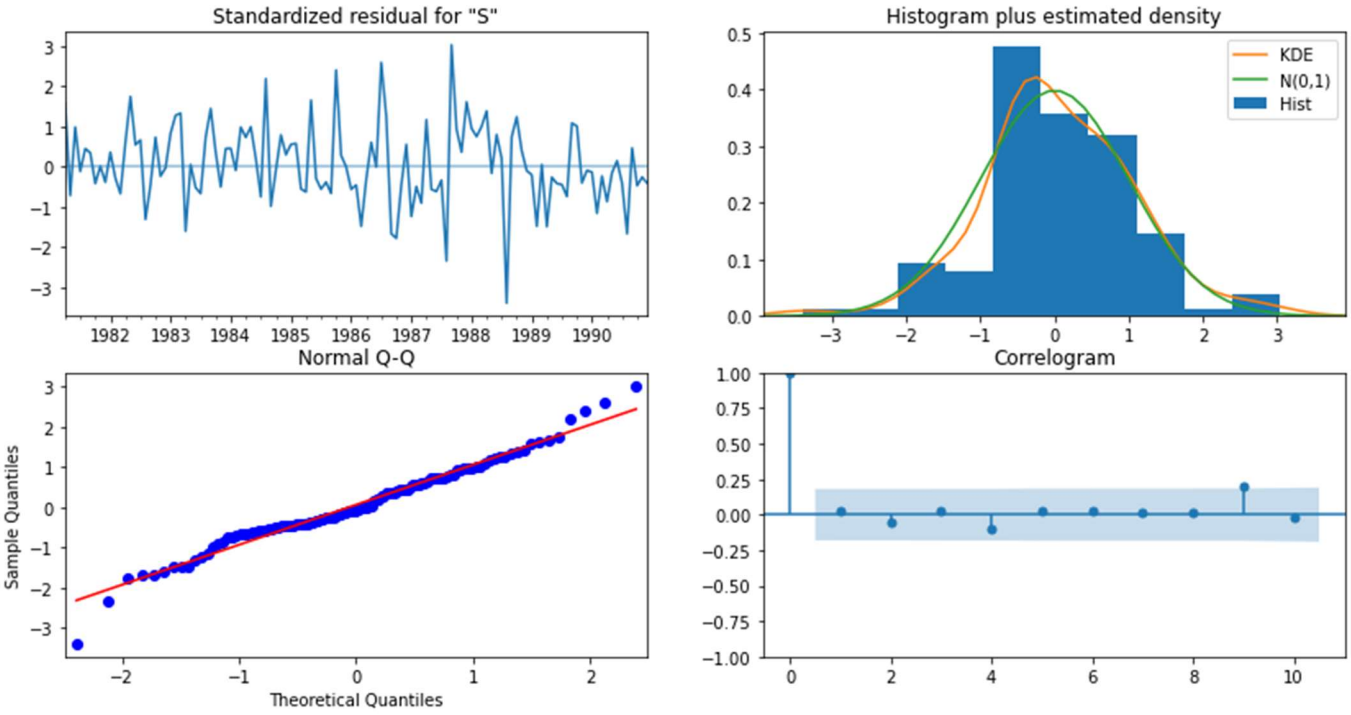
	param	seasonal	AIC
25	(0, 1, 1)	(1, 0, 1, 12)	-284.472032
79	(1, 1, 1)	(1, 0, 1, 12)	-282.517330
43	(0, 1, 2)	(1, 0, 1, 12)	-281.567996
97	(1, 1, 2)	(1, 0, 1, 12)	-279.611701
133	(2, 1, 1)	(1, 0, 1, 12)	-278.288232

Log Series SARIMA Model Result:

```
                                 SARIMAX Results
==========================================================================================
Dep. Variable:                           Sparkling   No. Observations:                  132
Model:             SARIMAX(0, 1, 1)x(1, 0, 1, 12)   Log Likelihood                 146.236
Date:                            Wed, 13 Jul 2022   AIC                           -284.472
Time:                                    11:59:30   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    -19.863      0.000      -0.985      -0.808
ar.S.L12       1.0112      0.020     49.871      0.000       0.971       1.051
ma.S.L12      -0.6489      0.075     -8.629      0.000      -0.796      -0.502
sigma2         0.0045      0.001      7.842      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
===================================================================================
```

Figure 24: Log Series SARIMA Model Summary

Diagnostic Plot:



Diagnostic-Plot

Observations:

- The diagnostics plot of the model was derived and the standardized residuals are found to follow a mean of zero, and the histogram shows the residuals follow a normal distribution.
- The Normal Q-Q plot also shows that the quantiles come from a normal distribution as the point forms roughly a straight line.
- The correlogram shows the autocorrelation of the residuals and there are no significant lags above the confidence index.
- From the above model summary, it can be inferred that MA.L1, AR.L.S12, MA.L.S12 terms has the highest absolute weightage.
- From the p-values it can be inferred that terms MA.L1, AR.L.S12, MA.L.S12 are significant terms, as their values are below 0.05. The RMSE values of the automated SARIMA of log series model is 336.799

Prediction on SARIMA Log Series:

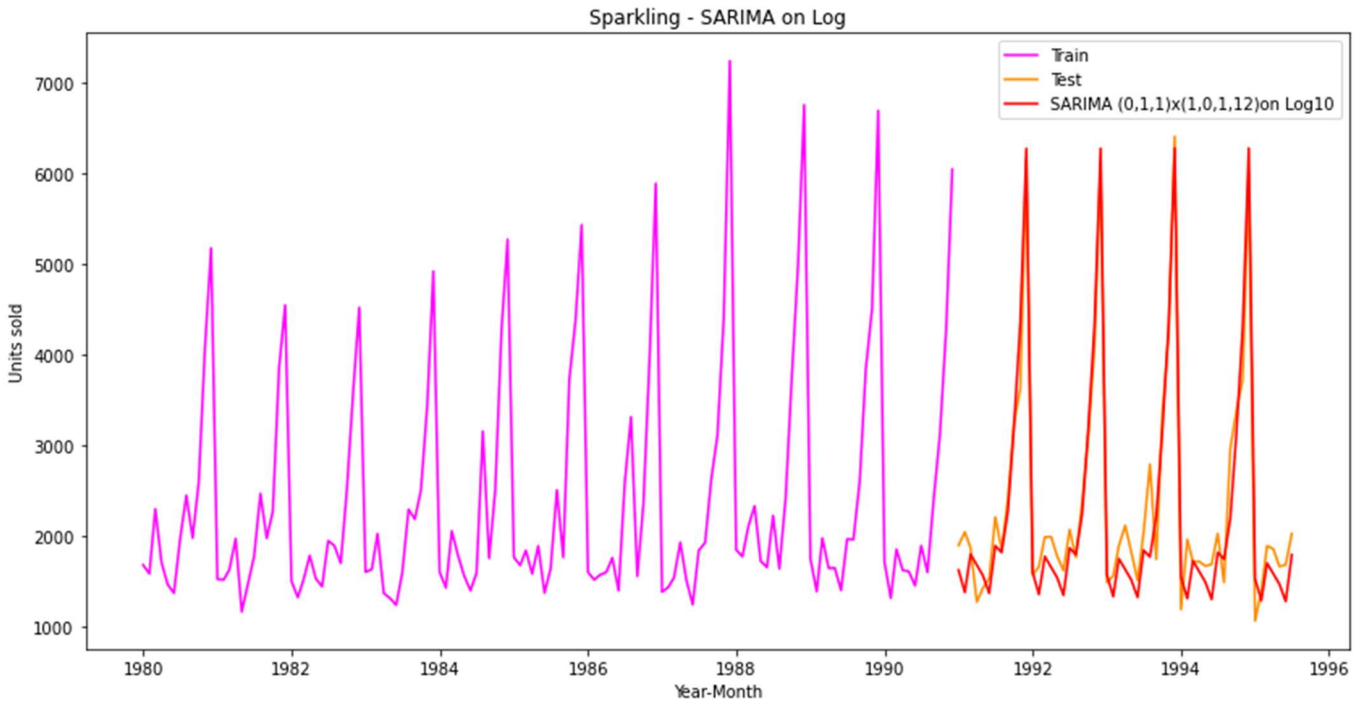
Sparkling	mean	mean_se	mean_ci_lower	mean_ci_upper
1991-01-01	3.212033	0.067108	3.080503	3.343562
1991-02-01	3.141308	0.067465	3.009080	3.273537
1991-03-01	3.256287	0.067821	3.123361	3.389213
1991-04-01	3.226733	0.068174	3.093114	3.360353
1991-05-01	3.195789	0.068527	3.061479	3.330099

Extracting the predicted and true values of our time series:

Forecasted Result on test data

YearMonth	Sparkling	spark_forecasted	spark_forecasted_log
1991-01-01	1902	1460.244632	1629.418664
1991-02-01	2049	1392.437181	1384.549093
1991-03-01	1874	1743.201708	1804.208809
1991-04-01	1279	1650.066942	1685.516569
1991-05-01	1432	1522.656037	1569.599978

Plot of Actual v/s Forecasted Result on test data



RMSE Values on test set:

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576708
Auto_SARIMA_log(0, 1, 1)*(1, 0, 1, 12)	336.799059

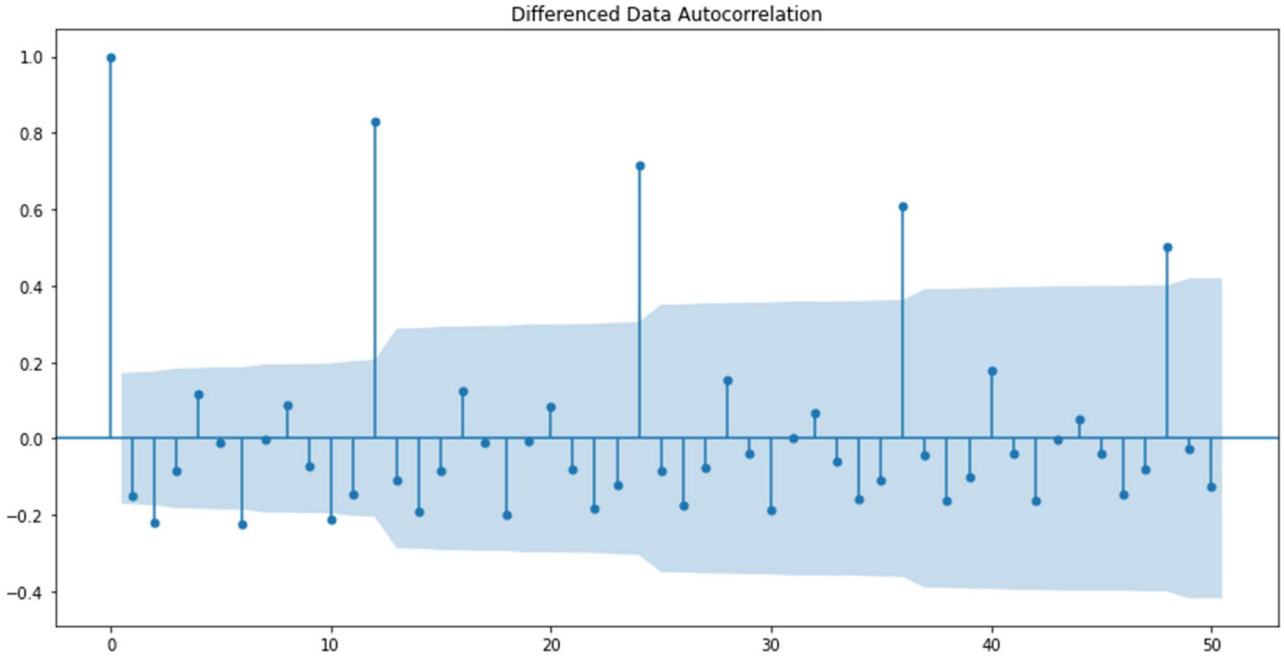
1.7 Build ARIMA/SARIMA models based on the cut-off points of ACF and PACF on the training data and evaluate this model on the test data using RMSE.

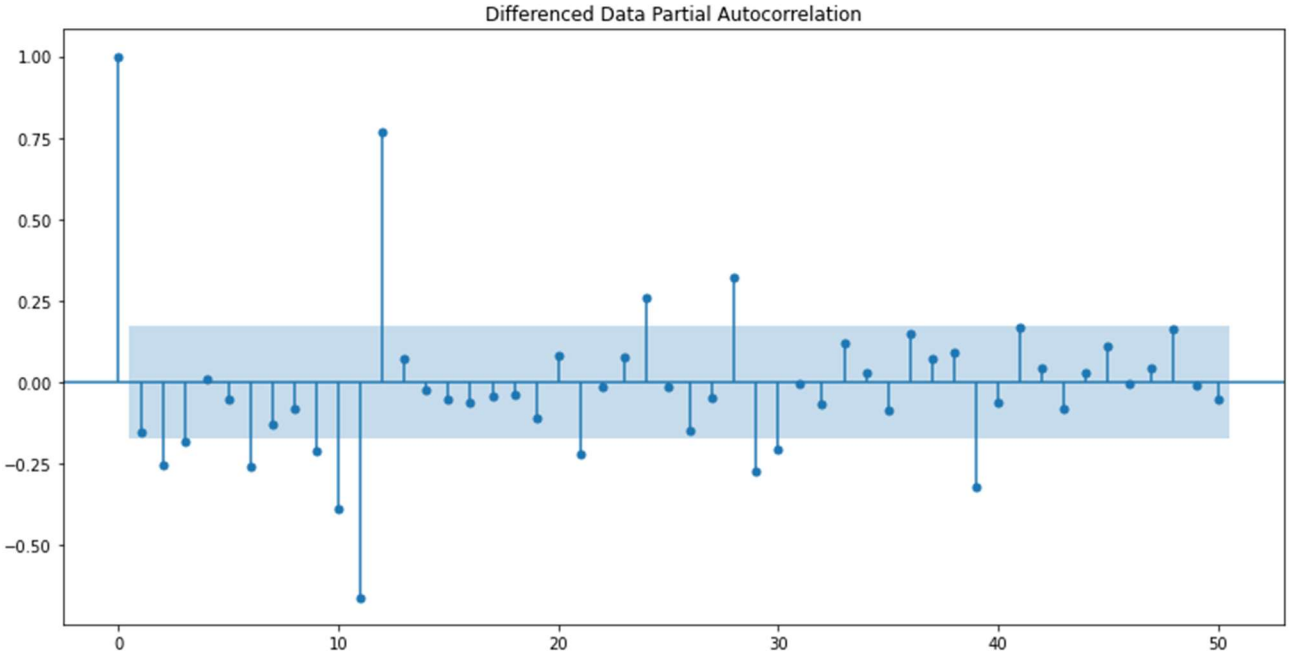
Solution:

Manual ARIMA:

We Will Look at the ACF and the PACF plots once more.

ACF and PACF Plots





- Here, we have taken alpha=0.05.
- The Auto-Regressive parameter in an ARIMA model is 'p' which comes from the significant lag before which the PACF plot cuts-off to 0.
- The Moving-Average parameter in an ARIMA model is 'q' which comes from the significant lag before the ACF plot cuts-off to 0.
- By looking at above plots, we can say that both the PACF and ACF plot cuts-off at lag 0.
- When we see that both the AR(p) and the MA(q) model are of order 0, we have to convert the input variable into a 'float64' type variable else Python might throw an error.

ARIMA Model Results:

```
                             ARIMA Model Results
==============================================================================
Dep. Variable:             D.Sparkling   No. Observations:                  131
Model:                 ARIMA(0, 1, 0)   Log Likelihood               -1132.791
Method:                           css   S.D. of innovations           1377.911
Date:                Wed, 13 Jul 2022   AIC                           2269.583
Time:                        11:59:32   BIC                           2275.333
Sample:                    02-01-1980   HQIC                          2271.919
                         - 12-01-1990
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const         33.2901    120.389      0.277      0.782    -202.667     269.248
==============================================================================
```

The RMSE value of manual ARIMA model is 4780. Since the ARIMA model do not capture the seasonality, this model does not perform well.

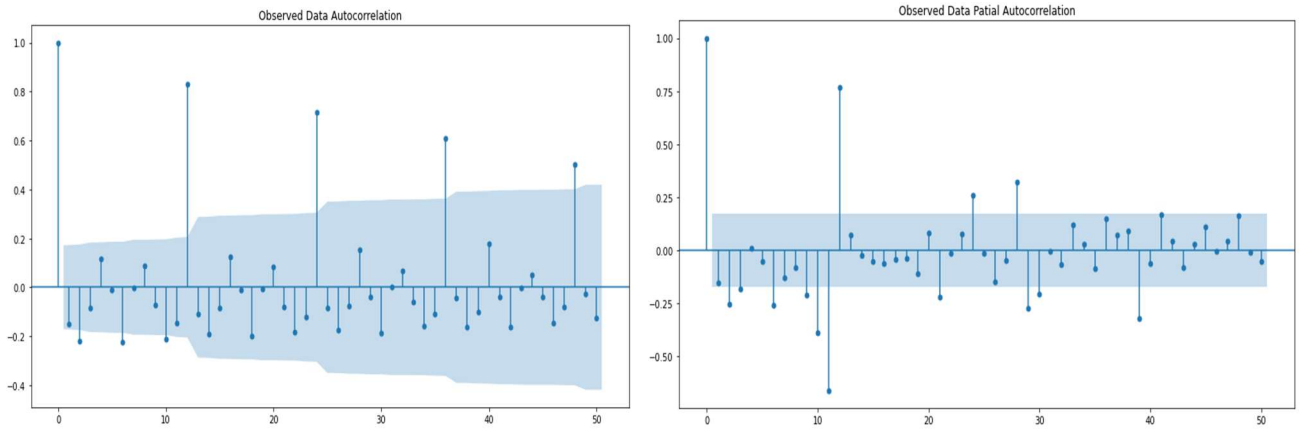
The data has some seasonality so we should build a SARIMA model to get better accuracy.

RMSE Values on test set:

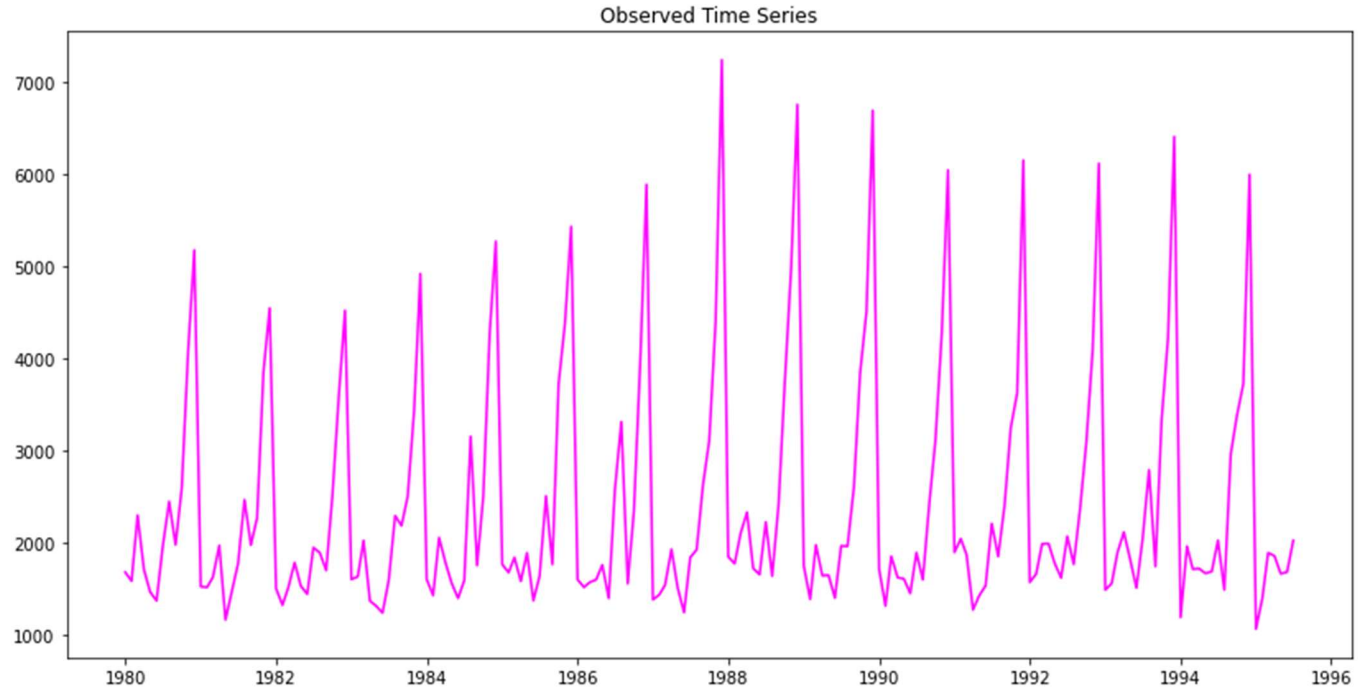
	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576708
Auto_SARIMA_log(0, 1, 1)*(1, 0, 1, 12)	336.799059
Manual_ARIMA(0,1,0)	4779.154299
Manual_SARIMA#(3,1,1)*(1,1,2,12)	324.104370
Manual_ARIMA(0,1,0)	4779.154299

Manual SARIMA Model:

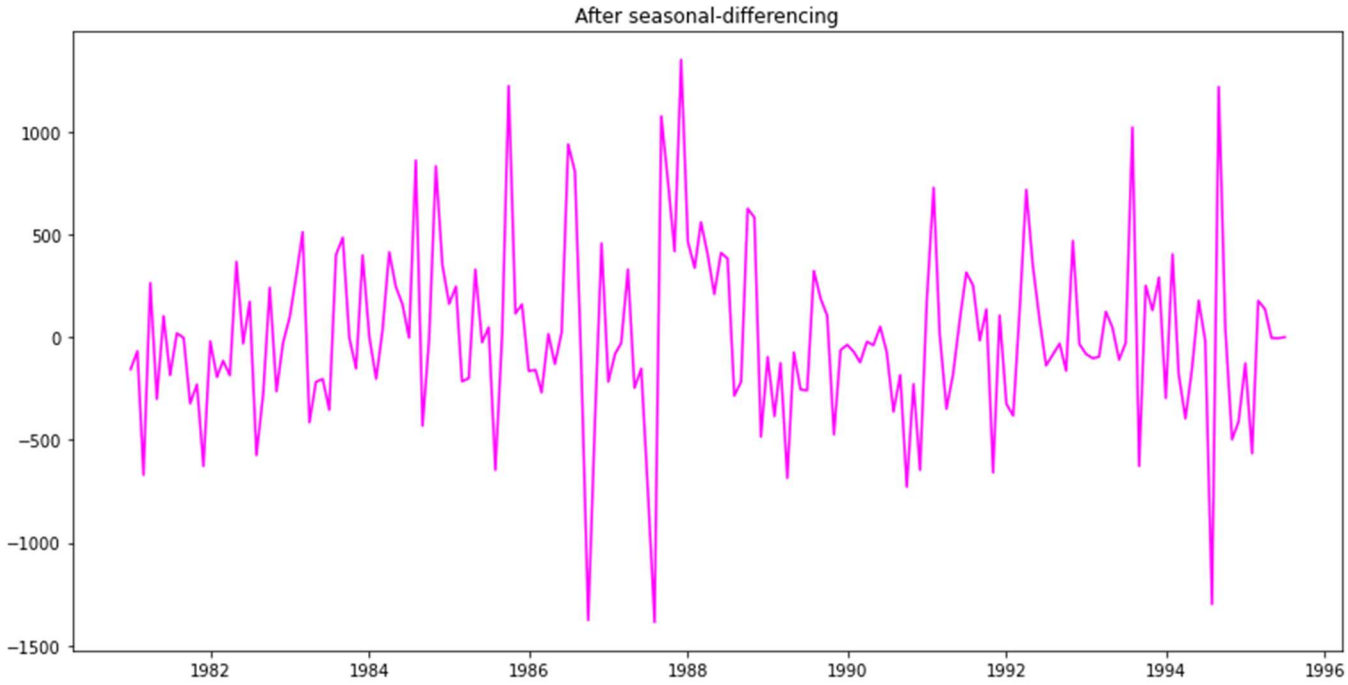
ACF and PACF Plots



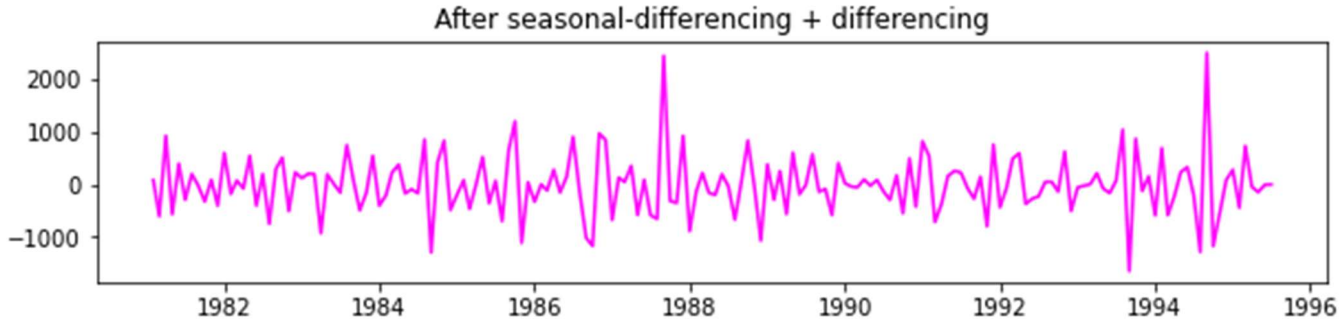
From the ACF plot of the observed/ train data, it can be inferred that at seasonal interval of 12, the plot is not quickly tapering off. So, a seasonal differencing of 12 has to be taken



We see that there is marginal trend and but have significant seasonality. So, now we take a seasonal differencing and check the series.



The marginal trend in the data is still seen.

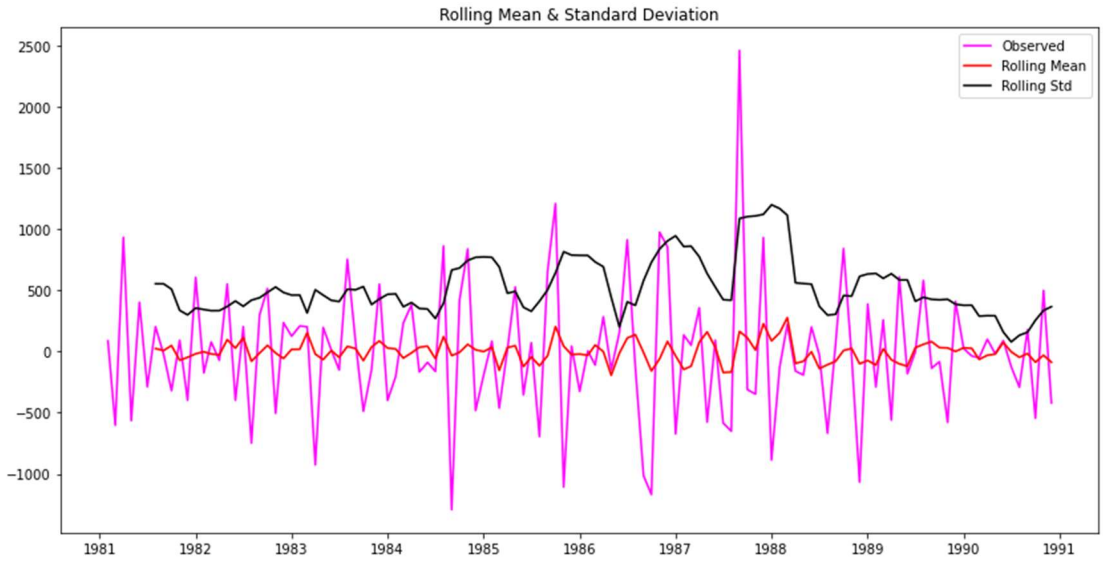


Inferences:

- Now we see that there is almost no trend present in the data. Seasonality is only present in the data.
- From the plots above an apparent slight trend is still existing after differencing of seasonal order of 12. With a further differencing of order one, no trend is present.
- An ADF test need to be done to check the stationarity after the above differencing. With a p-value below alpha 0.05 and test statistic below critical values, it can be confirmed that the data is stationary.

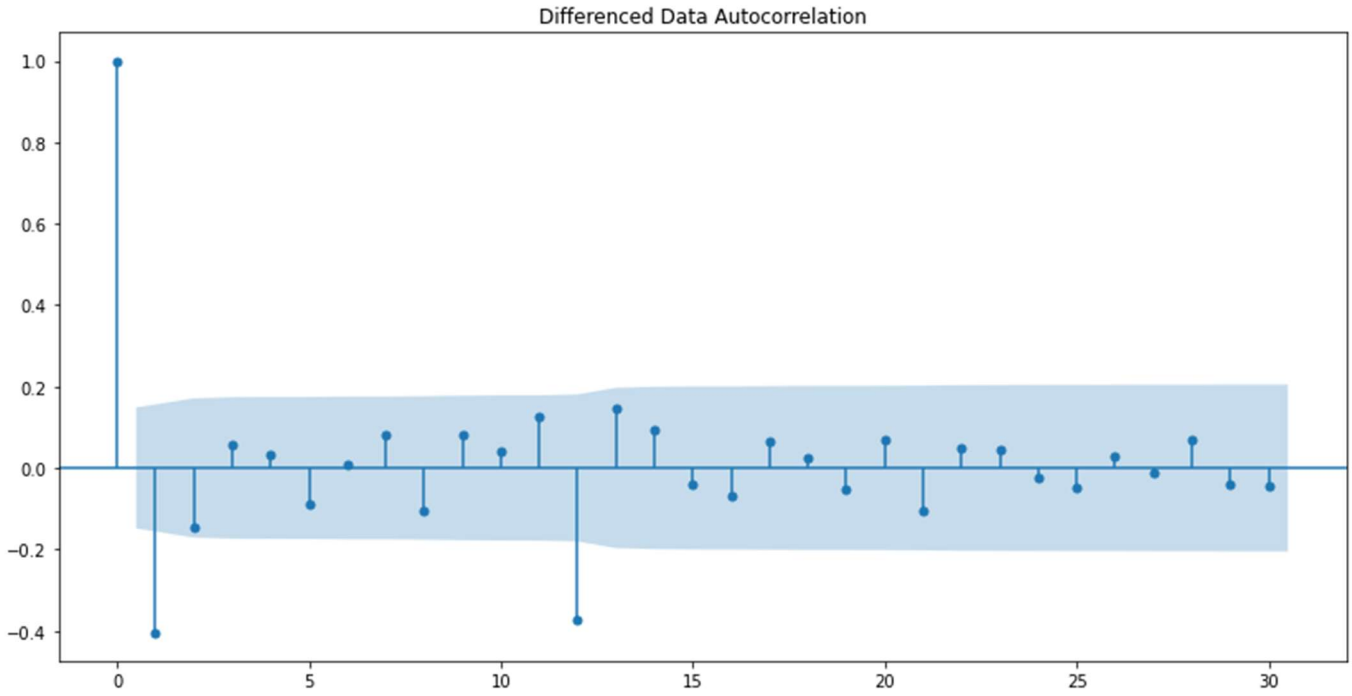
We will Check the stationarity of the above series before fitting the SARIMA model.

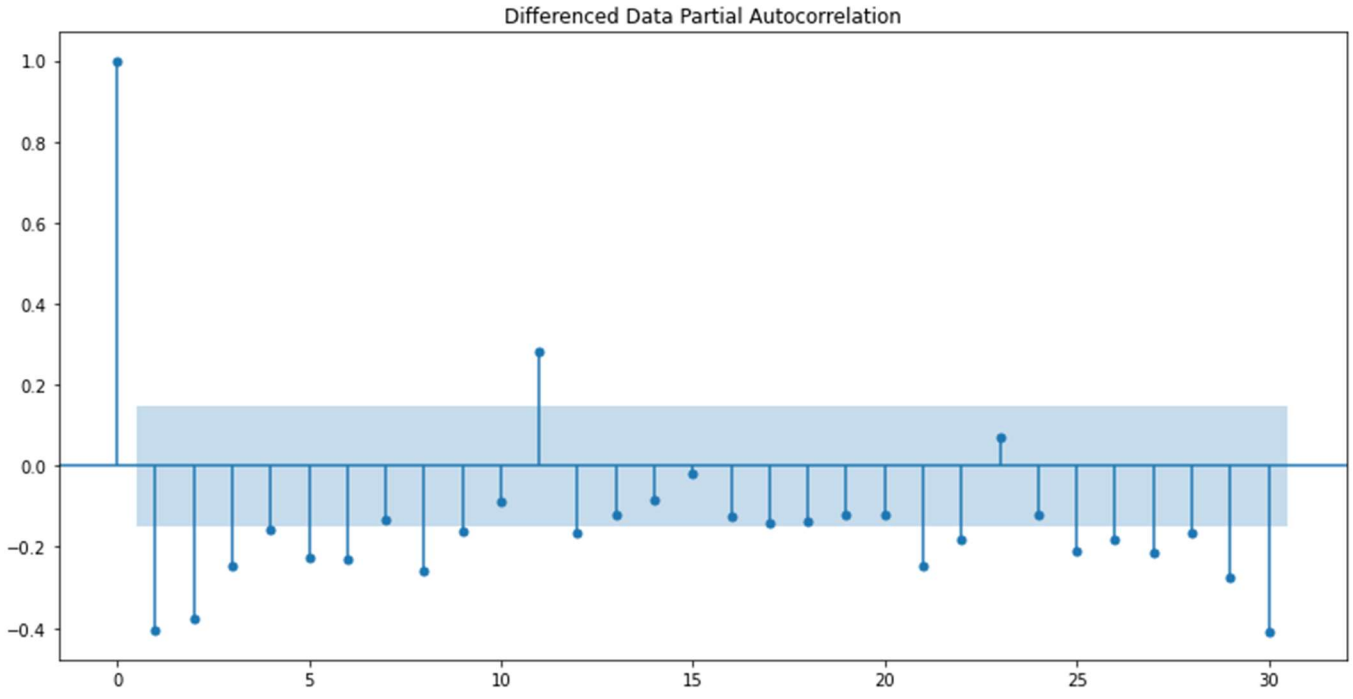
```
Results of Dickey-Fuller Test:
Test Statistic                  -3.342905
p-value                          0.013066
#Lags Used                      10.000000
Number of Observations Used    108.000000
Critical Value (1%)             -3.492401
Critical Value (5%)             -2.888697
Critical Value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



Checking the ACF and the PACF plots for the new modified Time Series:

ACF and PACF Plots

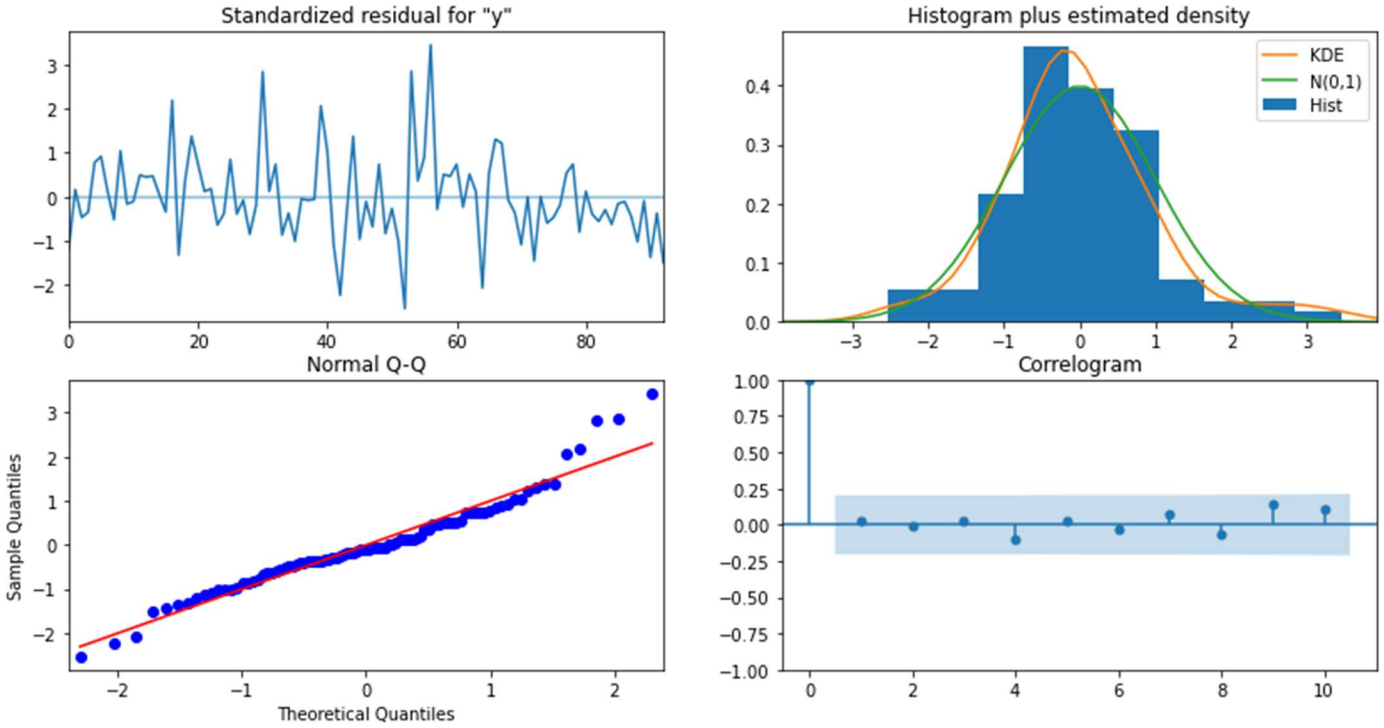




Inferences:

- Here we have taken alpha = 0.05 and seasonal period as 12.
- From the PACF plot it can be seen that till 3rd lag it's significant before cut-off, so AR term 'p = 3' is chosen. At seasonal lag of 12, it almost cuts off, so seasonal AR 'P = 1'
- From ACF plot it can be seen that lag 1 is significant before it cuts off, so MA term 'q =1' is selected and at seasonal lag of 12, a significant lag is apparent, so kept seasonal MA term 'Q = 1' initially.
- The seasonal MA term 'Q' was later optimized to 2, by validating model performance, as the data might be under-differenced.
- The final selected terms for SARIMA model is (3, 1, 1)*(1,1,2,12).
- The diagnostics plot of the model was derived and the standardized residuals are found to follow a mean of zero, and the histogram shows the residuals follow a normal distribution.
- The Normal Q-Q plot also shows that the quantiles come from a normal distribution as the point forms roughly a straight line.
- The correlogram shows the autocorrelation of the residuals and there are no significant lags above the confidence index.
- The RMSE values of the automated SARIMA model is 324.10

Diagnostic Plots:



Manual SARIMA Model

SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(3, 1, 1)x(1, 1, [1, 2], 12)	Log Likelihood	-693.697
Date:	Thu, 14 Jul 2022	AIC	1403.394
Time:	11:49:50	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

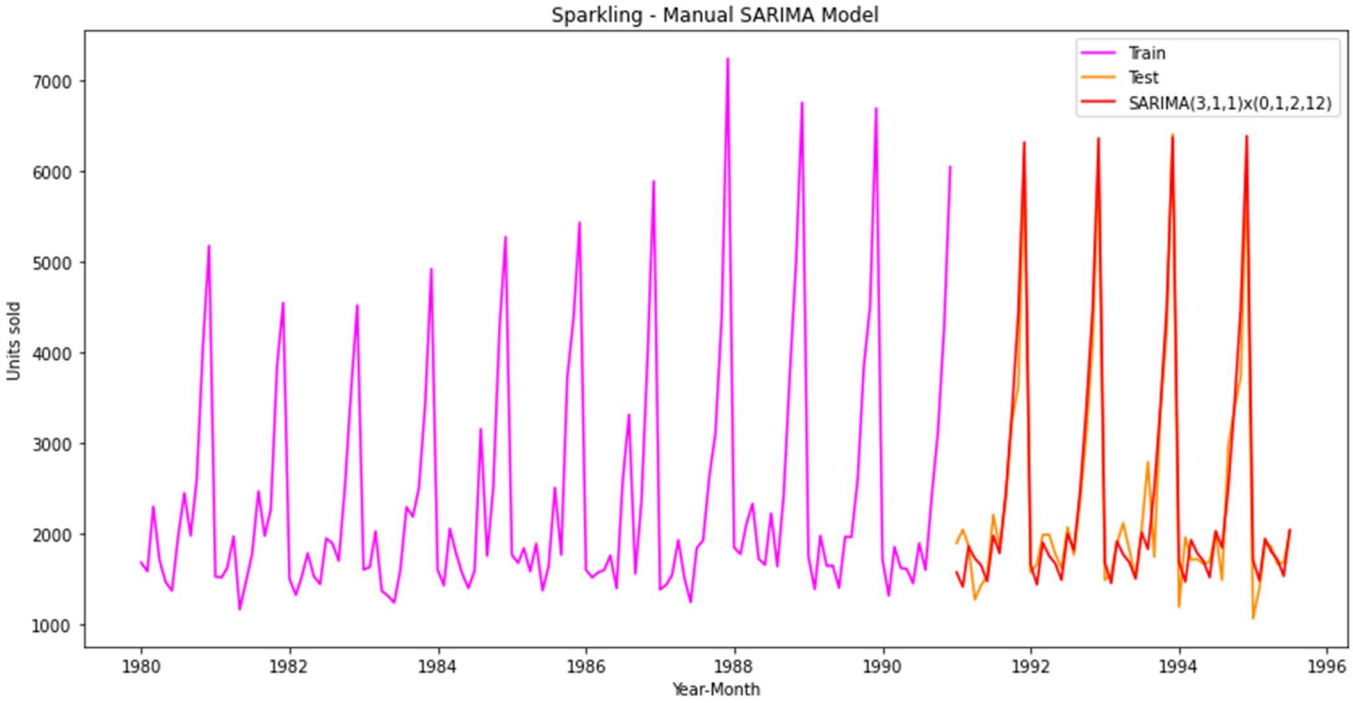
Forecasted Values:

	y	mean	mean_se	mean_ci_lower	mean_ci_upper
0		1579.905527	416.593917	763.396454	2396.414600
1		1419.146615	429.113122	578.100351	2260.192878
2		1868.135387	429.103765	1027.107463	2709.163312
3		1731.464728	430.972636	886.773883	2576.155573
4		1659.814781	431.905708	813.295148	2506.334414

Extracting the predicted and true values of our time series:

YearMonth	Sparkling	spark_forecasted	spark_forecasted_log	manual_spark_forecasted
1991-01-01	1902	1460.244632	1629.418664	1579.905527
1991-02-01	2049	1392.437181	1384.549093	1419.146615
1991-03-01	1874	1743.201708	1804.208809	1868.135387
1991-04-01	1279	1650.066942	1685.516569	1731.464728
1991-05-01	1432	1522.656037	1569.599978	1659.814781

Actual Plot v/s Forecast Result on test data



1.8 Build a table with all the models built along with their corresponding parameters and the respective RMSE values on the test data.

Solution:

	Test RMSE
RegressionOnTime	1389.135175
NaiveModel	3864.279352
SimpleAverage	1275.081804
2 point TMA	813.400684
4 point TMA	1156.589694
6 point TMA	1283.927428
9 point TMA	1346.278315
Alpha=0.0496, SES Optimized	1316.035487
Alpha=0.025,SES iterative	1286.248846
Alpha=0.68,Beta=0.0, DES Optimized	2007.238526
Alpha=0.1,Beta=0.1,DES iterative	1778.560000
Alpha=0.11,Beta=0.7,gamma=0.395 TES Optimized	469.767970
Alpha=0.4,Beta=0.1,gamma=0.3,TES iterative	371.367690
Auto_ARIMA(2,1,2)	1374.037009
Auto_SARIMA(1, 1, 2)*(0, 1, 2, 12)	382.576708
Auto_SARIMA_log(0, 1, 1)*(1, 0, 1, 12)	336.799059
Manual_ARIMA(0,1,0)	4779.154299
Manual_SARIMA#(3,1,1)*(1,1,2,12)	324.104370

Table 2: RMSE values of various models

Inference:

Manual SARIMA (3,1,1)*(1,1,2,12) is found to be the best model, followed by Auto_SARIMA model.

1.9 Based on the model-building exercise, build the most optimum model(s) on the complete data and predict 12 months into the future with appropriate confidence intervals/bands.

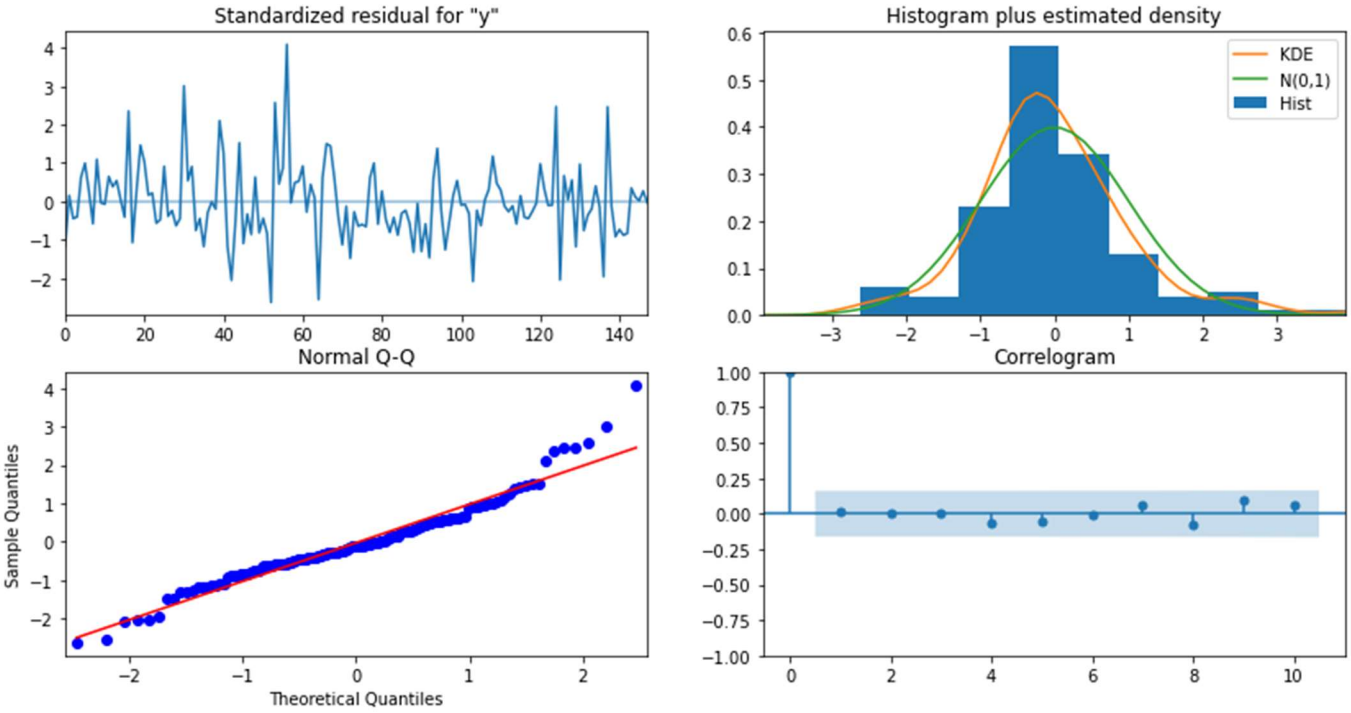
Solution:

- Based on the overall model evaluation and comparison, Manual SARIMA is selected for final prediction into 12 months in future.
- Manual SARIMA model with optimal parameters (3,1,1)*(1,1,2,12) is found to be the best model in terms of accuracy scored against the full data.
- The model predicts an upward trend and continuation of the seasonal surge in sales in the upcoming 12 months. According to the model the seasonal sale will be more

than that of the previous year.

Building a Manual SARIMA on the entire dataset:

Diagnostic Plots



SARIMAX Results

Dep. Variable:	y	No. Observations:	187
Model:	SARIMAX(3, 1, 1)x(1, 1, [1, 2], 12)	Log Likelihood	-1094.342
Date:	Thu, 14 Jul 2022	AIC	2204.685
Time:	11:49:54	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

Figure 25: Model SARIMA Model Summary

Forecast for the next 12 months using this model:

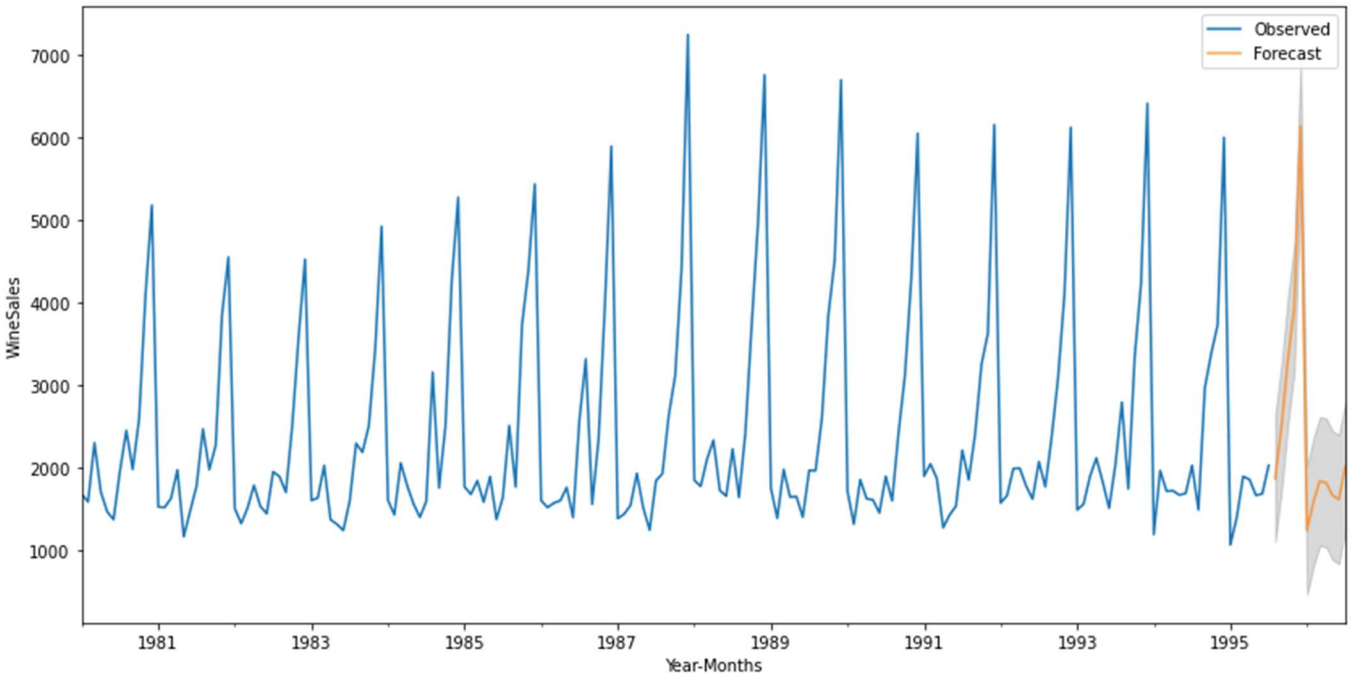
For Manual-SARIMA Model forecast on the Entire Data, RMSE is 547.591

	y	mean	mean_se	mean_ci_lower	mean_ci_upper
0		1870.888582	390.915200	1104.708870	2637.068294
1		2489.623605	395.293855	1714.861887	3264.385323
2		3299.650017	395.322883	2524.831404	4074.468629
3		3934.056613	396.282374	3157.357433	4710.755793
4		6135.396030	396.768722	5357.743625	6913.048436

Manual SARIMA Forecast values

Plot of forecast along with the confidence band:

Actual Plot v/s Future Forecast Result



1.10 Comment on the model thus built and report your findings and suggest the measures that the company should be taking for future sales.

Solution:

Forecasted Values:

```
1995-08-31    1870.888582
1995-09-30    2489.623605
1995-10-31    3299.650017
1995-11-30    3934.056613
1995-12-31    6135.396030
1996-01-31    1245.727182
1996-02-29    1584.643750
1996-03-31    1840.705253
1996-04-30    1823.847823
1996-05-31    1668.706094
1996-06-30    1620.472484
1996-07-31    2020.534846
Freq: M, Name: mean, dtype: float64
```

Summary Statistics:

```
count      12.000000
mean     2461.187690
std      1391.118211
min      1245.727182
25%      1656.647692
50%      1855.796918
75%      2692.130208
max      6135.396030
Name: mean, dtype: float64
```

Inferences:

- From 1981 to 1983 trend is decreasing , 1983 to 1988 increasing, then decreasing throughout 1995.
- Highest Sale on 1987, whereas lowest sale on 1995.
- The model forecasts sale of 29534 units of Sparkling wine in 12 months into future. Which is an average sale of 2462 units per month.
- The seasonal sale in December 1995 will hit a maximum of 6136 units, before it drops to the lowest sale in January 1996; at 1246 units .
- It is evident from monthly plot that sales have been increased from September to December. Stock has to be more during these time frame. January recorded the lowest sale which is right after the month of December of previous years.
- The wine company is recommended to ramp up their procurement and production line in accordance with the above forecasts for the third quarter of 1995 (October, November and December), which is a total of 13,370 units of sparkling wine is expected to be sold.
- Since the dataset has seasonality, SARIMA model would be best suited model. Same was evident through RMSE value. SARIMA has the lowest RMSE value. SARIMA model was applied on full data.
- Sales for next 12 month is predicted with confidence interval. Sales are varying drastically across the month, since shelf life of the wine is more, company should do production on median values which close to 2000. Which will meet peak demand without additional resource during the month of highest sales.
- The forecast also indicates that the year-on-year sale of sparkling wine is not showing an upward trend. The winery must adopt innovative marketing skills to improve the sale compared to previous years.