TIME SERIES FORECASTING

CONTENTS - FOR LABEL 'ROSE' A PRODUCT OF ABC ESTATE WINES

LIST OF FIGURES

LIST OF TABLES

1. PROBLEM DEFINITION

1.1. Context:

The 20th century laid the foundation for modern wine production, distribution, and appreciation.It established wine as a global commodity, with distinct reputations for Old World and New World producers.The rise of wine tourism, education, and critique systems created a thriving culture around wine.This century marked the transition of wine from a regional agricultural product to a global industry and cultural symbol.

The wine industry beyond the 20th century is characterized by adaptability and innovation. As the world faces environmental and economic challenges, the industry continues to thrive by embracing sustainability, leveraging technology, and appealing to the diverse tastes of a global consumer base. This era is marked by a balance between preserving age-old traditions and pioneering modern practices.

The sales of wine have undergone significant transformations over the years, influenced by changing consumer preferences, global market dynamics, and innovations in production and distribution.

1.2. Objective:

The primary objective is to analyze 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, capitalize on emerging market opportunities, and maintain a competitive edge in the wine industry.

2.DATA BACKGROUND AND CONTENTS

2.1.Purpose:

The purpose of collecting the data was to forecast the sales of the product **ROSE** from ABC Estate Wines. This forecasting aims to help the company enhance their strategies and develop new ideas to maintain a competitive position in the market.

2.2.Data Description and Dictionary:

The data provided is the sales of each month from 01-1980 to 07-1995 for the product **Rose.**

Data Dictionary

- **YearMonth:** Contains year and month
- **Rose:** Contains sales with respect to yearmonth column

2.3.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
```

- The variables from the dataset are of object and float.
- The column rose have 2 missing values.

3.MISSING VALUE TREATMENT:

- The column Rose contains two missing values that need to be addressed.
- Linear interpolation has been applied to treat the missing values.
- The reason for choosing linear interpolation is that there were only two missing values. This method is well-suited in this case compared to other methods, such as taking the average of the nearest data points or the seasonal averages.

4.READING THE DATA AS APPROPRIATE TIME SERIES DATA:

- The column YearMonth and Rose are of object and float data type which needs to be converted to datatime and int.
- The Yearmonth is converted into datetime object and yearmonth column is set as index of the dataframe.The time index allows pandas to recognize the data as a time series.
- The column Rose is converted to int.
- Since time series data inherently relies on time as the organizing principle, many time series models depend on an indexed time column for accurate forecasting and analysis.

5.EDA

5.1.PLOTING TIME SERIES FOR UNDERSTANDING THE BEHAVIOUR OF THE DATA

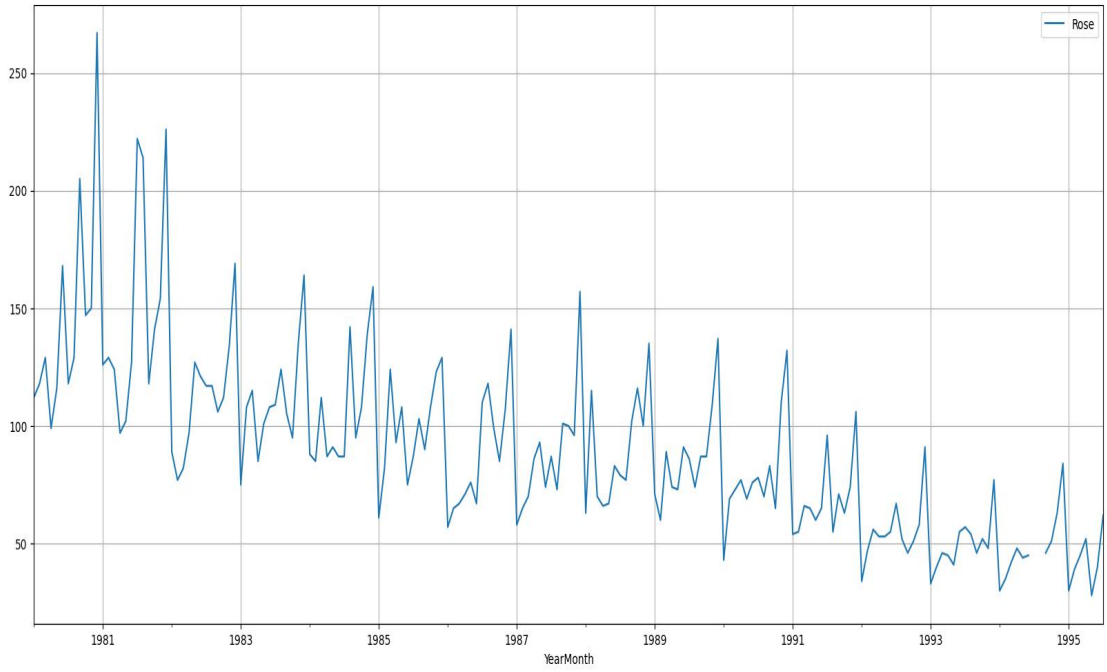


Fig 1:Time series

We can see that there is downward trend with no proper traces of seasonal pattern associated.

5.2.YEARLY BOXPLOT

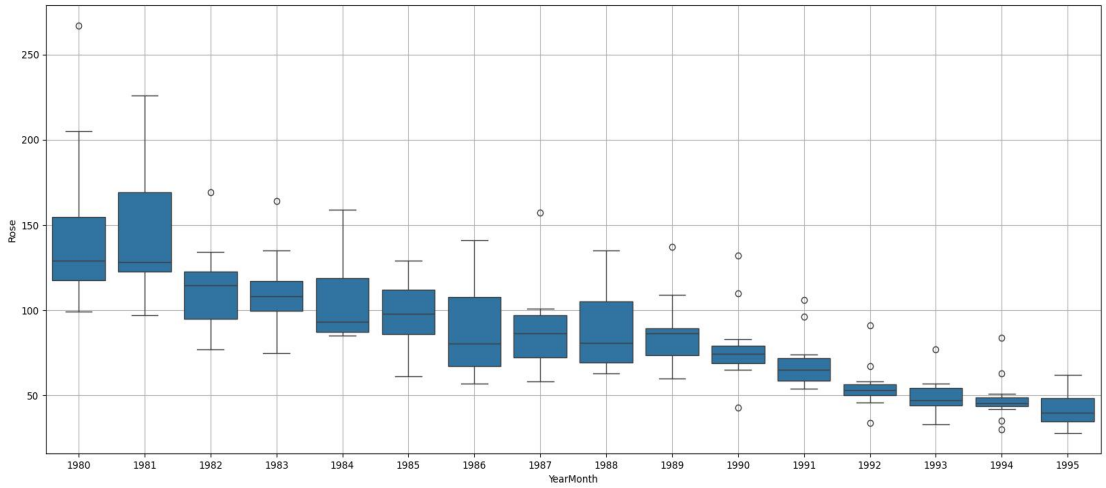


Fig 2:Yearly Boxplot

The sales for the year 1980 and 1981 were high and year by year we can see fall in the sales.

5.3.MONTHLY BOXPLOT

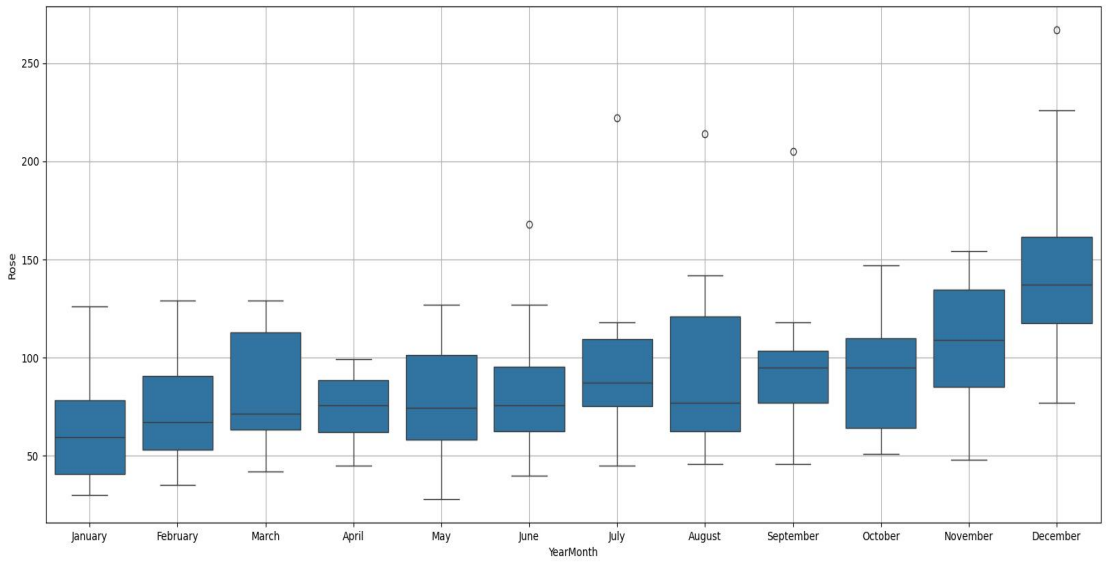


Fig 3:Monthly Boxplot

The sales for month November and December for every year were high,this might be of christmas and new year time.

5.4.SPREAD OF ACCIDENTS ACROSS DIFFERENT YEARS AND WITHIN DIFFERENT MONTHS ACROSS YEARS:

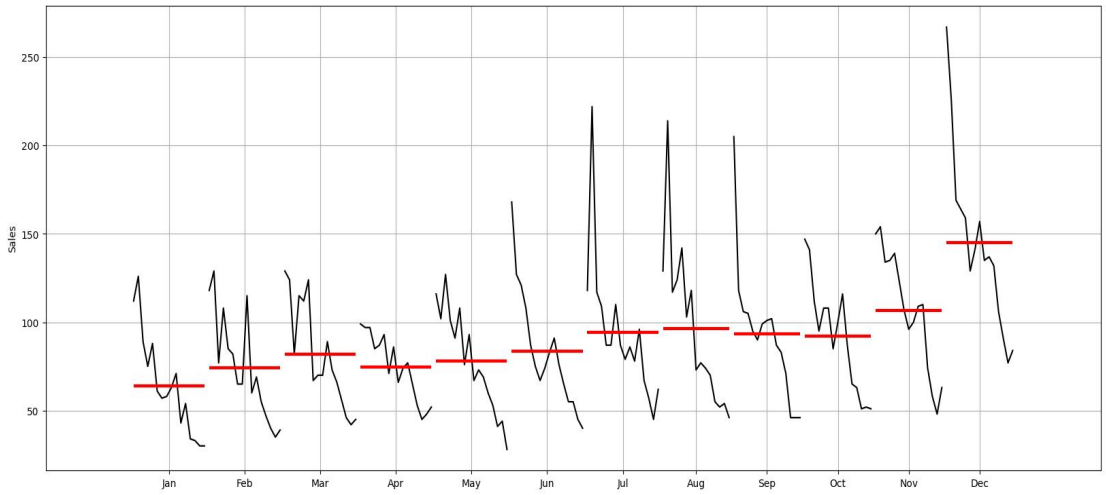


Fig 4:Spread of sales for each month over years

The red line indicates the median value ,the median value for november and december are high.

5.5.PIVOT TABLE:

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	118.0	129.0	205.0	147.0	150.0	267.0
1981	126.0	129.0	124.0	97.0	102.0	127.0	222.0	214.0	118.0	141.0	154.0	226.0
1982	89.0	77.0	82.0	97.0	127.0	121.0	117.0	117.0	106.0	112.0	134.0	169.0
1983	75.0	108.0	115.0	85.0	101.0	108.0	109.0	124.0	105.0	95.0	135.0	164.0
1984	88.0	85.0	112.0	87.0	91.0	87.0	87.0	142.0	95.0	108.0	139.0	159.0
1985	61.0	82.0	124.0	93.0	108.0	75.0	87.0	103.0	90.0	108.0	123.0	129.0
1986	57.0	65.0	67.0	71.0	76.0	67.0	110.0	118.0	99.0	85.0	107.0	141.0
1987	58.0	65.0	70.0	86.0	93.0	74.0	87.0	73.0	101.0	100.0	96.0	157.0
1988	63.0	115.0	70.0	66.0	67.0	83.0	79.0	77.0	102.0	116.0	100.0	135.0
1989	71.0	60.0	89.0	74.0	73.0	91.0	86.0	74.0	87.0	87.0	109.0	137.0
1990	43.0	69.0	73.0	77.0	69.0	76.0	78.0	70.0	83.0	65.0	110.0	132.0
1991	54.0	55.0	66.0	65.0	60.0	65.0	96.0	55.0	71.0	63.0	74.0	106.0
1992	34.0	47.0	56.0	53.0	53.0	55.0	67.0	52.0	46.0	51.0	58.0	91.0
1993	33.0	40.0	46.0	45.0	41.0	55.0	57.0	54.0	46.0	52.0	48.0	77.0
1994	30.0	35.0	42.0	48.0	44.0	45.0	45.0	46.0	46.0	51.0	63.0	84.0
1995	30.0	39.0	45.0	52.0	28.0	40.0	62.0	NaN	NaN	NaN	NaN	NaN

Table 1:Pivot Table

This pivot table gives clear indication of sales for each month of each year,this clearly shows sales dip every year.

5.6.MONTHLY SALES ACROSS YEARS:

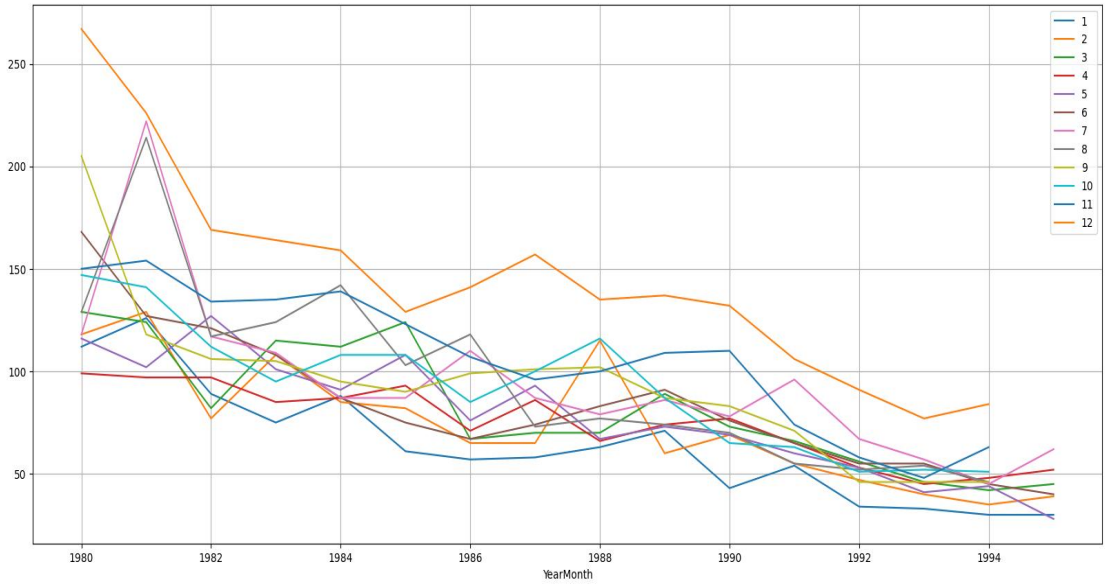


Fig 5:Monthly sales across years

The sales for december alone stands out.

5.7.EMPIRICAL CUMULATIVE DISTRIBUTION

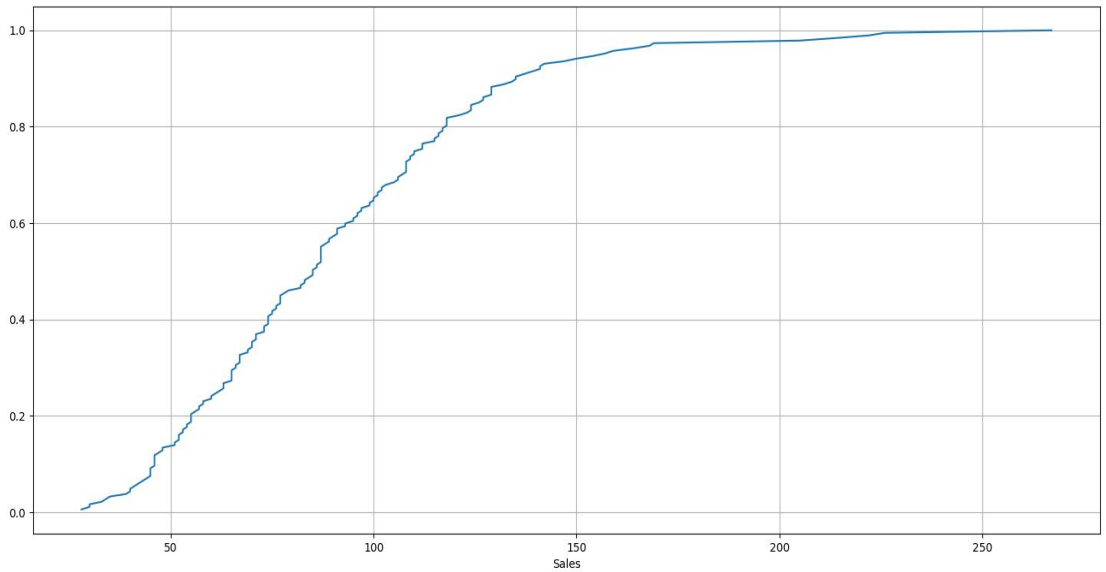


Fig 6:Empirical cumulative distribution

60% of the sales between the year 1980 to 1995 are below 100 and sales are not beyond 300.

5.8.AVERAGE SALES PER MONTH AND THE MONTH ON MONTH PERCENTAGE CHANGE OF SALES.

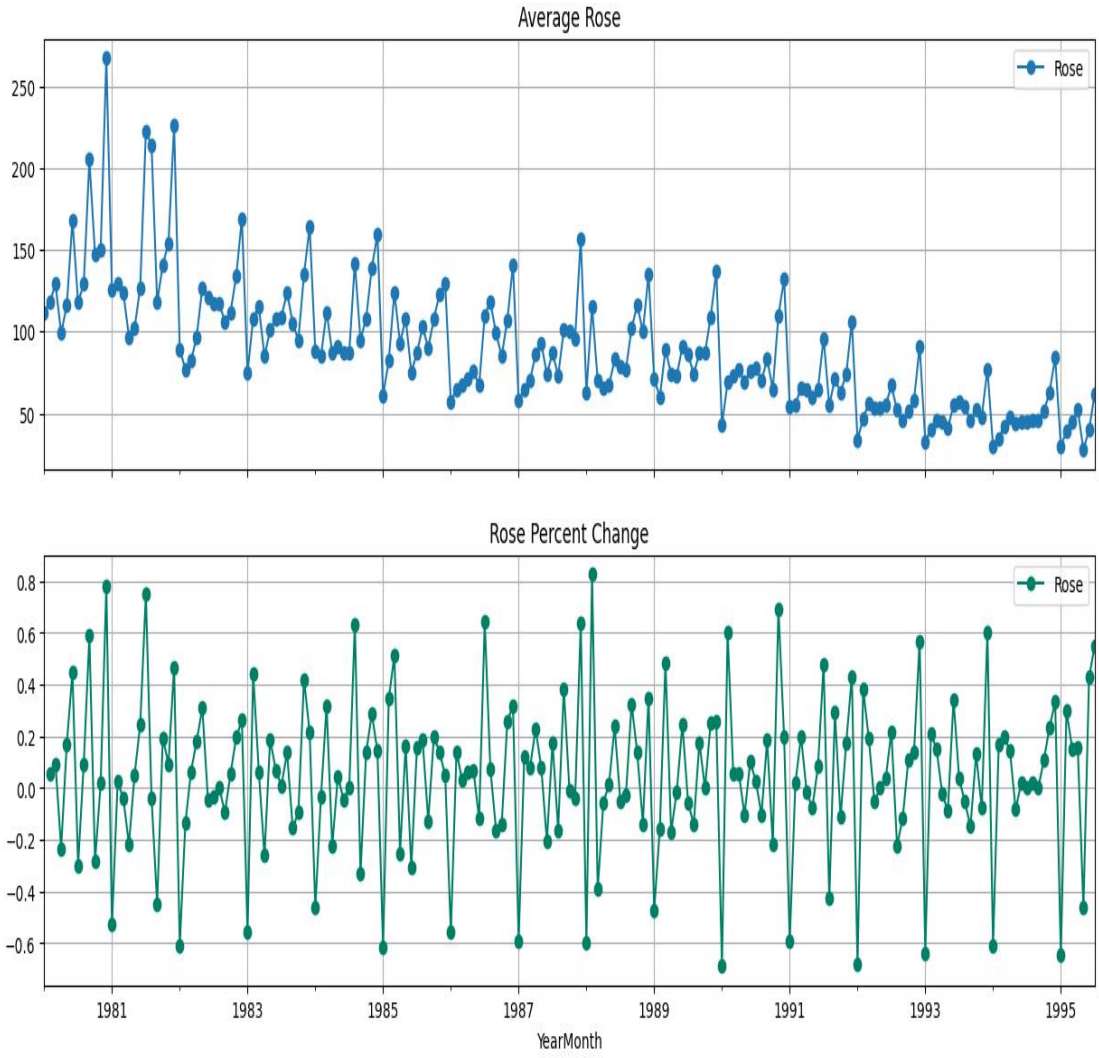


Fig 7:Average Sales and percentage change of sales per month

Every year the sales increases garadually month by month.The sales at the start of the year is higher compared to end of the year.

The sales for almost for every start of the year is about 60% less to previous year end.

6.DECOMPOSITION OF TIME SERIES

6.1.ADDITIVE

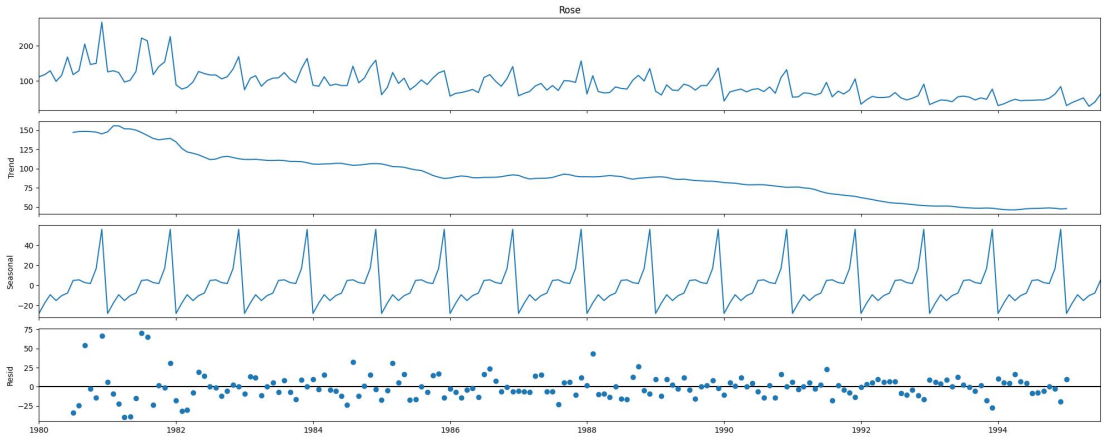


Fig 8:Decomposition-Additive

Which model better suits the data,the residual part shows which can be further used in the model building,looking at the residuals of additive shows the error are not normally distributed,not constant along zero.

6.2.MULTIPLICATIVE

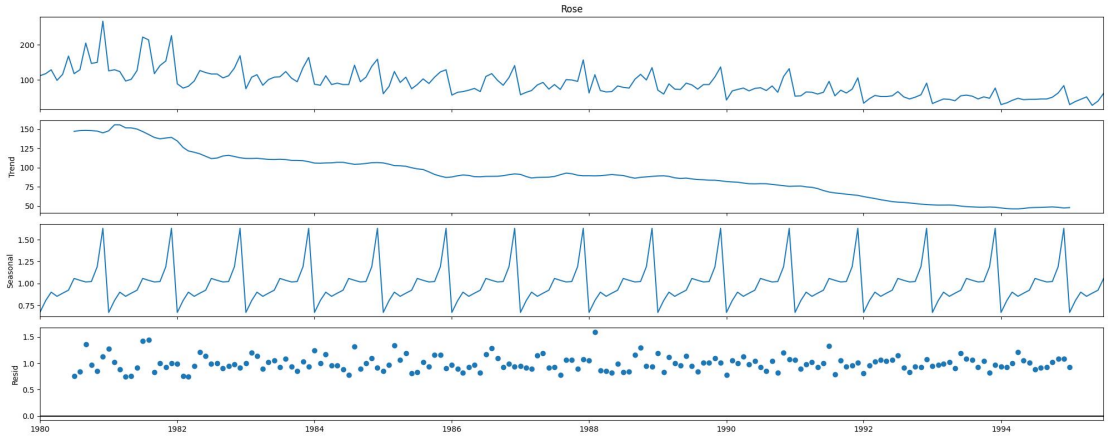


Fig 9:Decomposition-Multiplicative

Looking at the residuals of multiplicative shows the error are somewhat normally distributed,constant along one.Hence multiplicative better suits.

7.TRAIN AND TEST SPLIT:

- First 70% of the rows are selected as the train set.
- Remaining 30% of the rows are selected as test set.

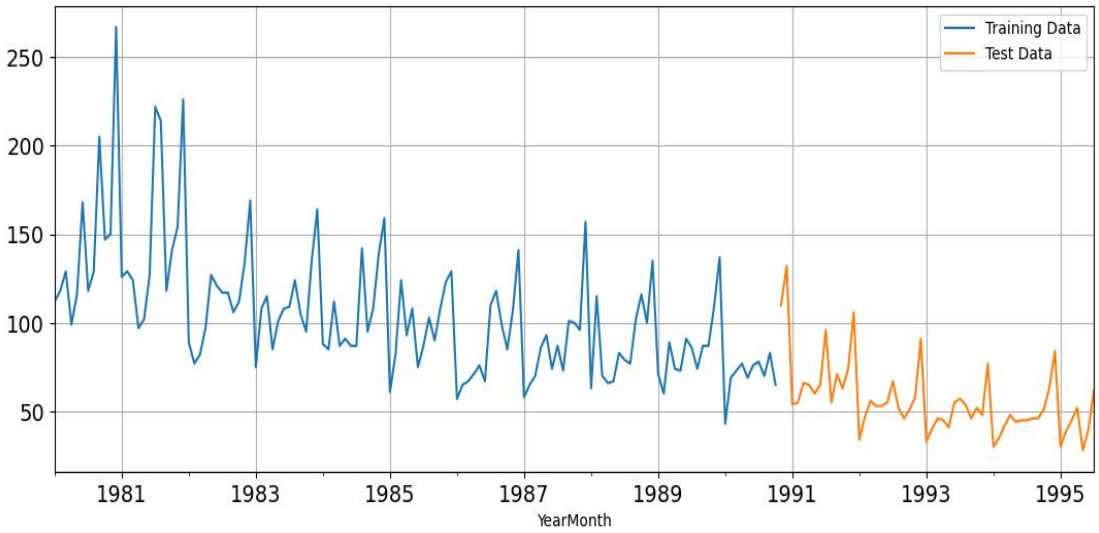


Fig 10:Train and Test split

8.MODEL BUILDING ON ORIGINAL DATA

8.1.LINEAR REGRESSION

- Time indices is created for the train and test dataset to align with the time series, with time instance for train data from 1 to 130 and test data from 131 to 187.
- Linear regression model requires independent and dependent variable,hence the time column created acts as the independent variable and the column Rose acts as the dependent variable.
- Now,let us check how the model has performed on the test data.

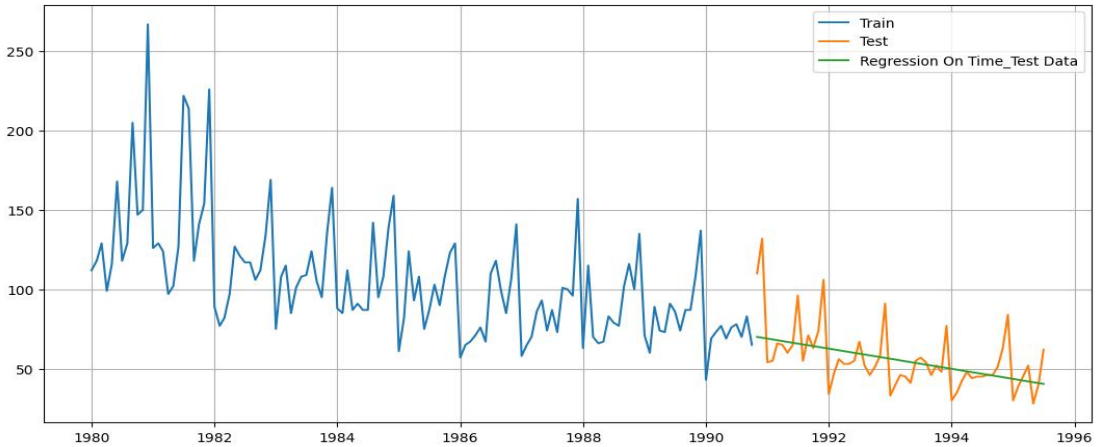


Fig 11:Linear Regression

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data by measuring the average magnitude of prediction errors.

	Test RMSE
RegressionOnTime	17.356199

- Let us continue with other forecasting model and check the score of RMSE.

8.2.SIMPLE AVERAGE:

- The simple average method takes out the average of the column Rose.
- The mean value will be repeated for each row in the newly created meanforecast column.
- Now,let us check how the model has performed on the test data.

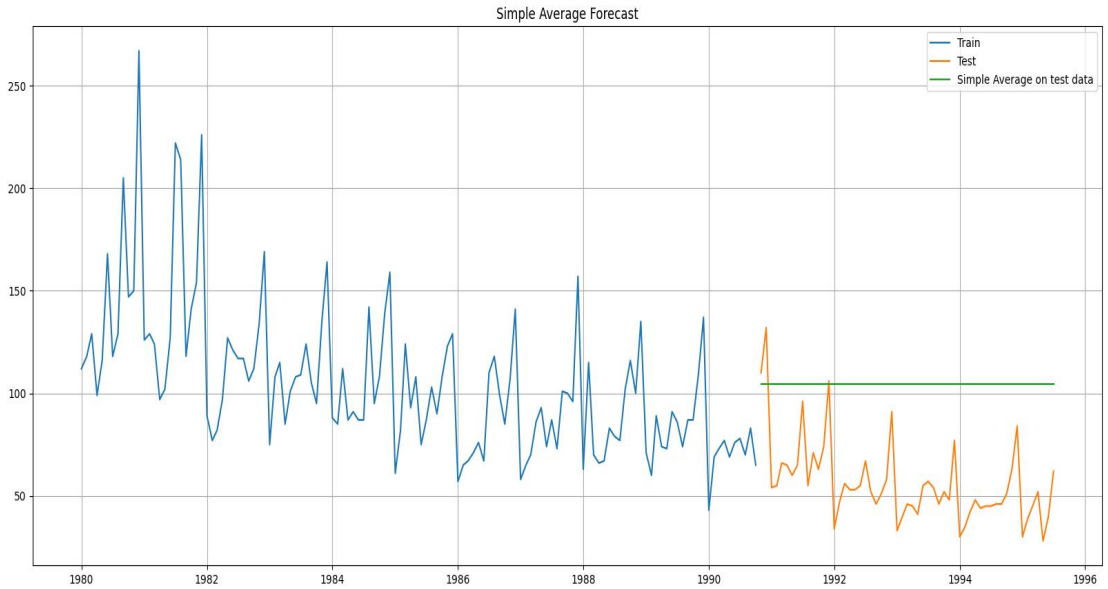


Fig 12:Simple Average

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

	Test RMSE
SimpleAverage	52.412167

- Let us continue with other forecasting model and check the score of RMSE.

8.3.MOVING AVERAGE:

- Let us calculate the moving average by rolling() which is used to create a rolling window of a specified size over the data. The mean() function then computes the average of the values within that window.
- Moving average is done for different window sizes (2, 4, 6, and 9) and stores the results in new columns.
- Now,let us check how the model has performed on the test data.

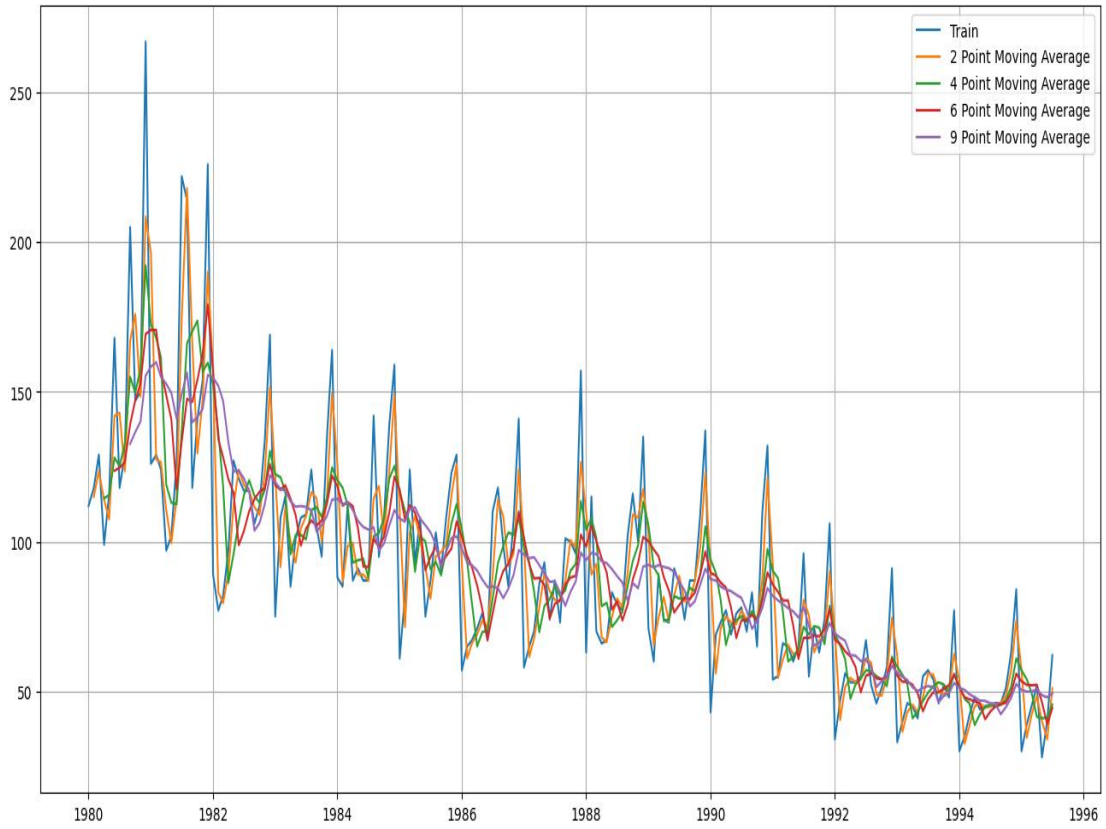


Fig 13:Moving Average

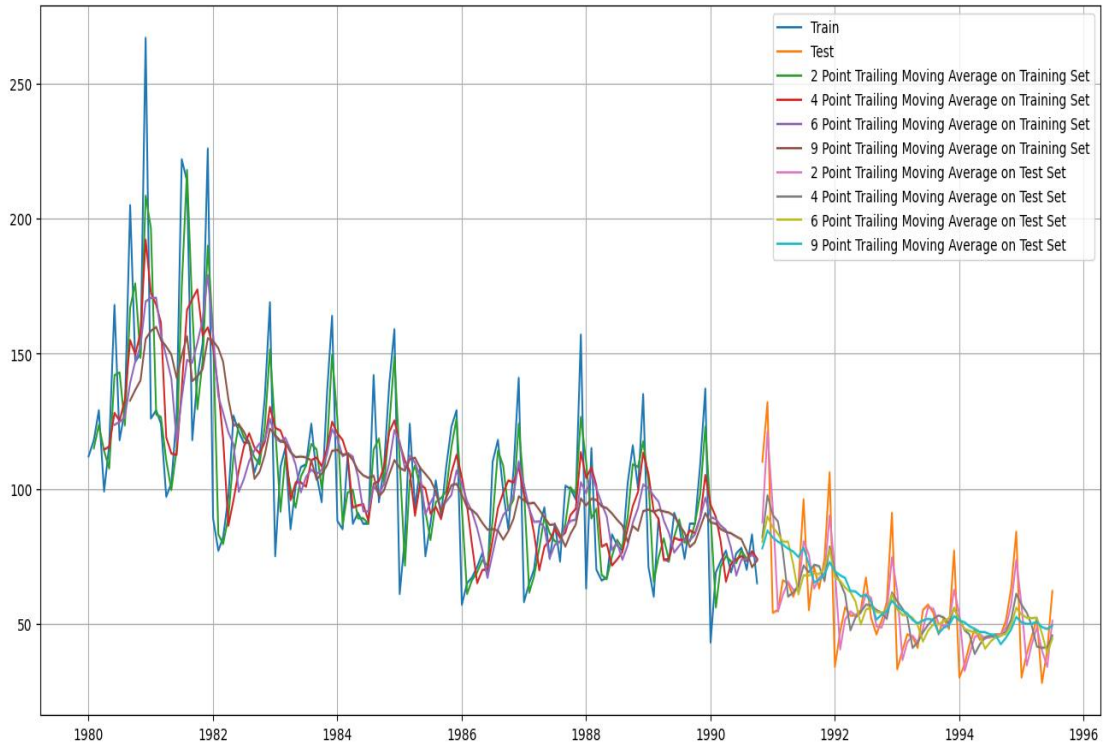


Fig 14:Moving Average on train and test set

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

For 2 point Moving Average Model forecast on the Training Data, RMSE is 11.801
For 4 point Moving Average Model forecast on the Training Data, RMSE is 15.367
For 6 point Moving Average Model forecast on the Training Data, RMSE is 15.863
For 9 point Moving Average Model forecast on the Training Data, RMSE is 16.342

- Let us continue with other forecasting model and check the score of RMSE.

8.4.SINGLE EXPONENTIONAL SMOOTHING

- Simple Exponential Smoothing (SES) model is initialized for time series forecasting on the Rose column of the SES_train DataFrame.

- SES is a forecasting method that applies weights to past observations, with the most recent observations receiving the highest weights.

- SES uses a smoothing parameter (often denoted as α, alpha) that ranges from 0 to 1. This parameter determines the weight given to the most recent observation compared to the previous forecast.

- If α is close to 1, the model will put more weight on recent observations, making it more responsive to changes.
- If α is close to 0, the model will rely more on historical averages, making it less sensitive to recent fluctuations.
- The variable model_SES_autofit contain a fitted SES model object obtained from calling the fit() method on a SimpleExpSmoothing model.
- The expression model_SES_autofit.params is used to access the estimated parameters of a fitted Simple Exponential Smoothing (SES) model from the statsmodels library.
- The params attribute retrieves the parameters of the fitted SES model. In SES, the key parameter is the smoothing level (often denoted as alpha, α).

{'smoothing_level': 0.12777740554752187,
'smoothing_trend': nan,
'smoothing_seasonal': nan,
'damping_trend': nan,
'initial_level': 112.0,
'initial_trend': nan,
'initial_seasons': array([], dtype=float64),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the SES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

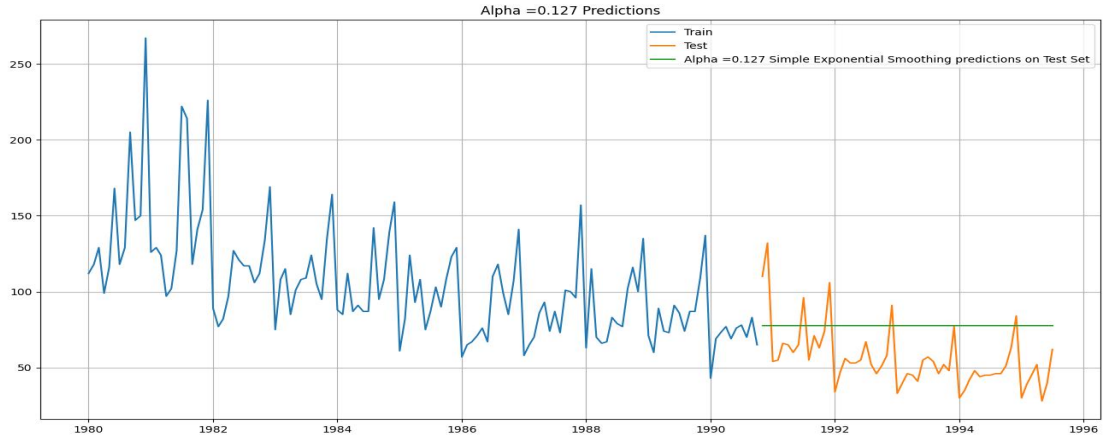


Fig 15:Single Exponential Smoothing

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

For Alpha =0.127 Simple Exponential Smoothing Model forecast on the Test Data, R MSE is 29.224

- Let us also perform manual alpha values to check the performance at different alpha values.

	Alpha Values	Train RMSE	Test RMSE
8	0.9	37.507371	22.496993
7	0.8	36.330954	23.212945
6	0.7	35.288467	23.895092
5	0.6	34.372651	24.529970
4	0.5	33.578304	25.109760
3	0.4	32.893017	25.657916
2	0.3	32.292266	26.310496
1	0.2	31.779467	27.482273
0	0.1	31.643829	30.310717

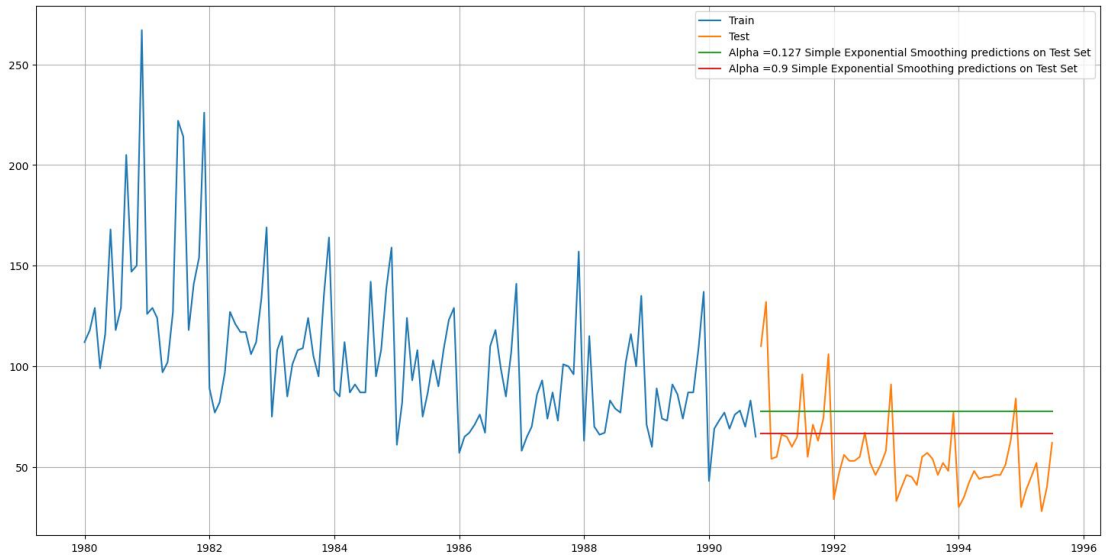


Fig 16:Single Exponential Smoothing

- The manual processed alpha at 0.9 has better RMSE scores compared to auto alpha at 0.127

8.5.DOUBLE EXPONENTIONAL SMOOTHING

- We have fitted a Double Exponential Smoothing (DES) model using the Exponential Smoothing class from the statsmodels library,with an additive trend model with no seasonal pattern in the model.

- We have fitted a Double Exponential Smoothing model to the time series data while optimizing the smoothing parameters automatically.

- The params attribute retrieves the parameters of the fitted DES model. In DES, the key parameter is the smoothing level and smoothing trend (often denoted as alpha α and Beta).

{'smoothing_level': 1.4901161193847656e-08,
'smoothing_trend': 2.070482310011032e-10,
'smoothing_seasonal': nan,
'damping_trend': nan,
'initial_level': 139.35304597867423,
'initial_trend': -0.5291021253561838,
'initial_seasons': array([], dtype=float64),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the DES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

For Alpha =0.0 and Beta =0.0 Simple Exponential Smoothing Model forecast on t he Test Data, RMSE is 29.224

- Let us also perform manual alpha and beta values to check the performance at different alpha and beta values.

	Alpha Values	Beta Values	Train RMSE	Test RMSE
18	0.5	0.5	39.637227	17.547521
13	0.4	0.8	41.359445	17.595903
0	0.3	0.3	35.778091	17.726168
16	0.5	0.3	37.417219	17.750342
24	0.6	0.3	38.377546	18.071278

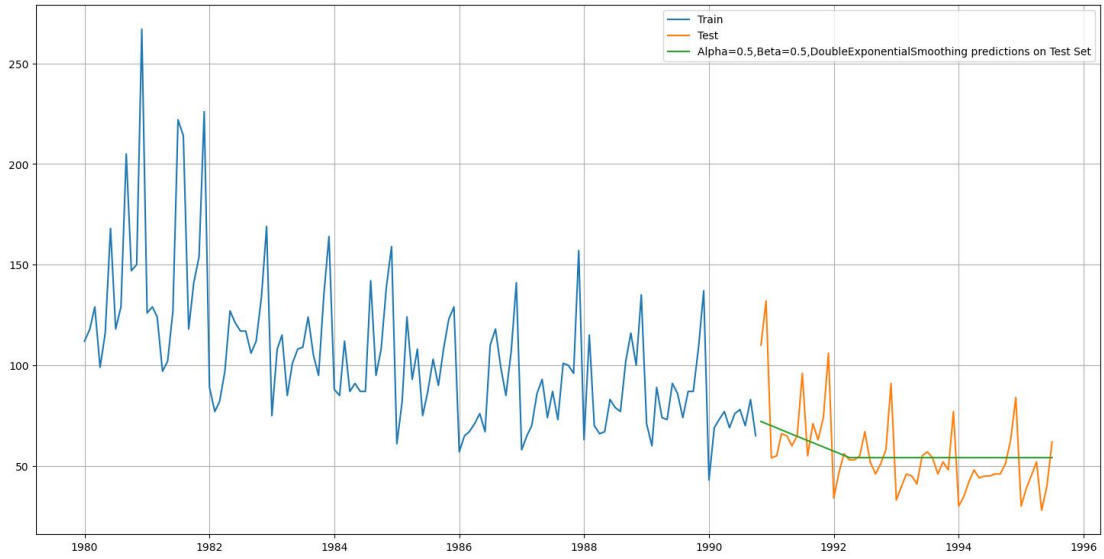


Fig 17:Double Exponential Smoothing

The manual processed RMSE at alpha and beta =0.5 has better score compared to auto alpha and beta =0.0

8.6.TRIPLE EXPONENTIONAL SMOOTHING

- We have fitted a Triple Exponential Smoothing (DES) model using

the Exponential Smoothing class from the statsmodels library,with an additive trend model with seasonal pattern as multiplicative.

- We have fitted a Triple Exponential Smoothing model to the time series data while optimizing the smoothing parameters automatically.

- The params attribute retrieves the parameters of the fitted TES model. In TES, the key parameter is the smoothing level , smoothing trend and smoothing seasonal (often denoted as alpha α ,Beta and gamma).

{'smoothing_level': 0.10149537680635393,
'smoothing_trend': 5.35227985694191e-07,
'smoothing_seasonal': 4.753443133120085e-05,
'damping_trend': nan,
'initial_level': 127.34117641758287,
'initial_trend': -0.5145410559544459,
'initial_seasons': array([0.86107985, 0.9733365 , 1.06416698, 0.9333589
, 1.04767686,
1.12846211, 1.24235129, 1.32871582, 1.24379656, 1.22358422,
1.40693323, 1.93996332]),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the TES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

For Alpha=0.1015,Beta=0.0,Gamma=0.0, Triple Exponential Smoothing Model f orecast on the Test Data, RMSE is 9.325

- Let us also perform manual alpha and beta values to check the performance at different alpha ,beta values and gamma values.

	Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE
25	0.3	0.6	0.4	27.743621	15.503837
140	0.5	0.4	0.7	37.430299	18.500881
135	0.5	0.3	1.0	47.353331	20.994935
78	0.4	0.4	0.9	43.001123	23.709751
177	0.5	0.9	0.4	41.232290	25.873746

The auto processed at alpha = 0.1015 and beta =0.0 and gamma= 0.0 has better score compared to manual.

Lets us compare the RMSE scores for the models so far.

	Test RMSE
Alpha=0.1015,Beta=0.0,Gamma=0.0,TripleExponentialSmoothing	9.324844
2pointTrailingMovingAverage	11.801167
4pointTrailingMovingAverage	15.366538
Alpha=0.3,Beta=0.6,Gamma=0.4,TripleExponentialSmoothing	15.503837
6pointTrailingMovingAverage	15.863483
9pointTrailingMovingAverage	16.342156
RegressionOnTime	17.356199
Alpha=0.5,Beta=0.5,DoubleExponentialSmoothing	17.547521
Alpha=0.9,SimpleExponentialSmoothing	22.496993
Alpha=0.127,SimpleExponentialSmoothing	29.223811
SimpleAverage	52.412167

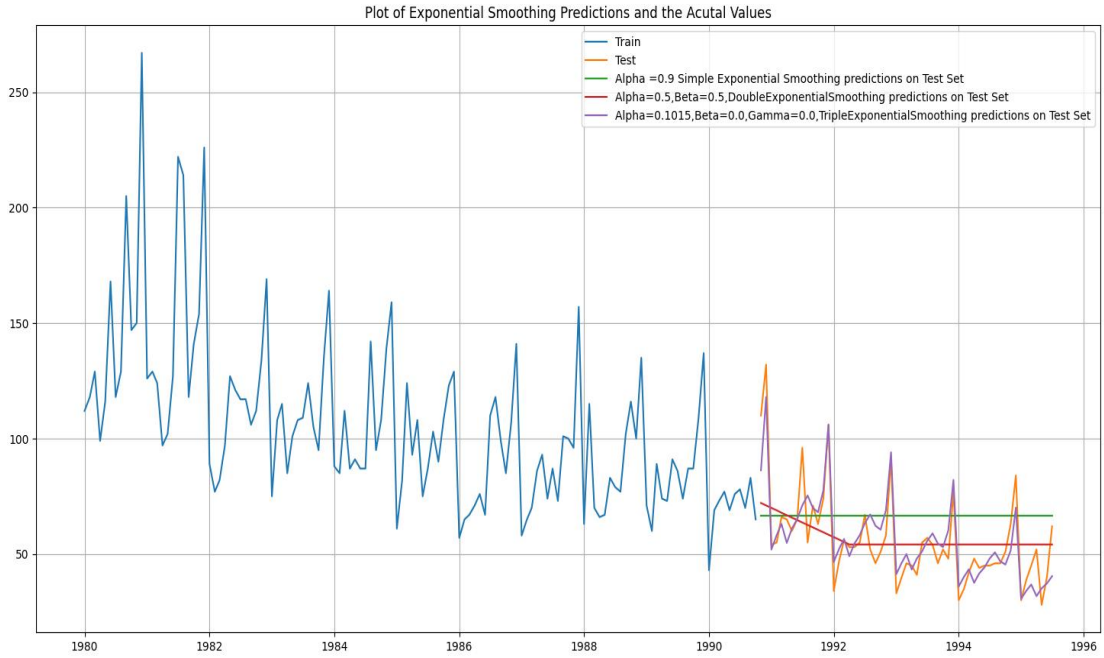


Fig 18:Triple Exponential Smoothing

Further,let us perform the ARIMA and SARIMA model,for that the these model assumes that the entire data is stationary.Let us perform stationarity check to continue with ARIMA and SARIMA.

9.CHECK FOR STATIONARITY

To check the stationarity of the data lets us perform Augmented Dickey-Fuller test.

Results of Dickey-Fuller Test:
Test Statistic -1.877785
p-value 0.342583
#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

- We check the p-value to know whether the data is stationary or not.
- We formulate hypothesis for the result.

Null Hypothesis:The data is non-stationary.

Alternate Hypothesis: The data is stationary.

- Since the p-value id greater than 0.05 we fail to reject the null hypothesis thus indicating data is non-stationary.Further we perform one order differencing and lest check the result.

Results of Dickey-Fuller Test:
Test Statistic -8.044360e+00
p-value 1.811239e-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

- Since the p-value id less than 0.05 reject the null hypothesis thus indicating data is stationary.

10.MODEL BUILDING ON STATIONARY DATA

10.1.ACF AND PACF PLOT

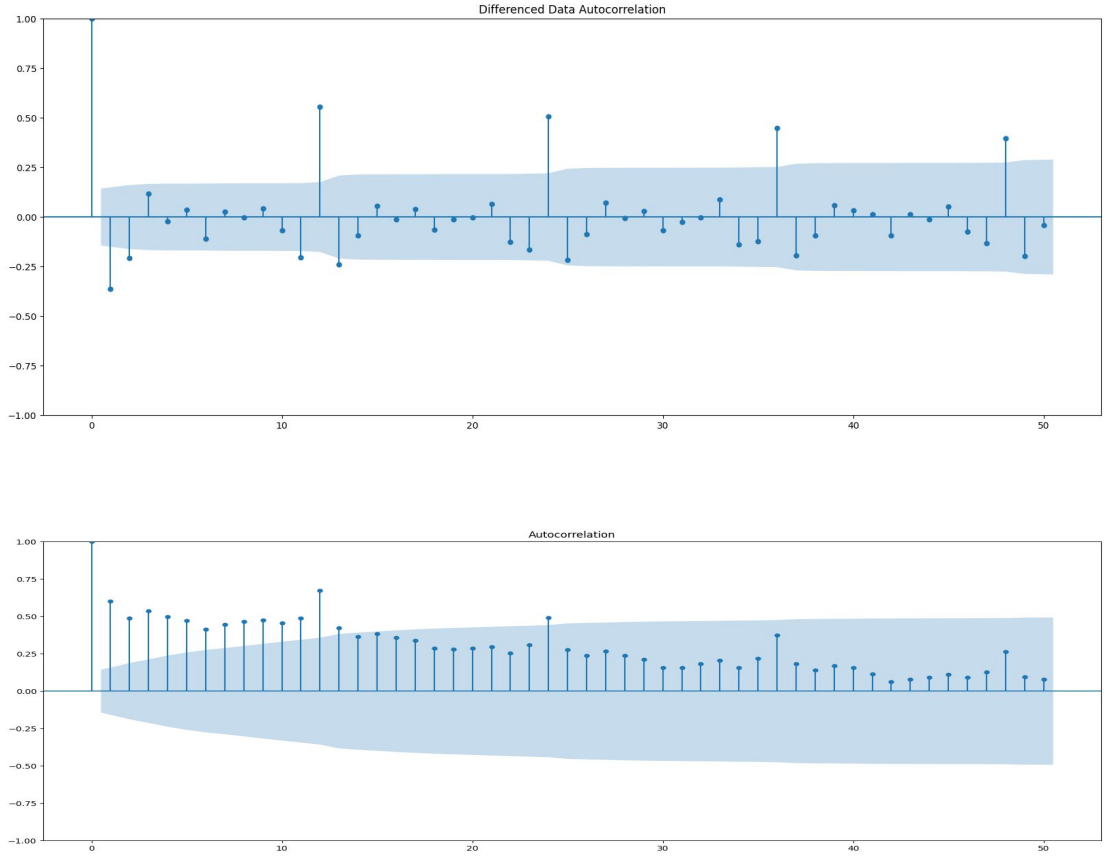
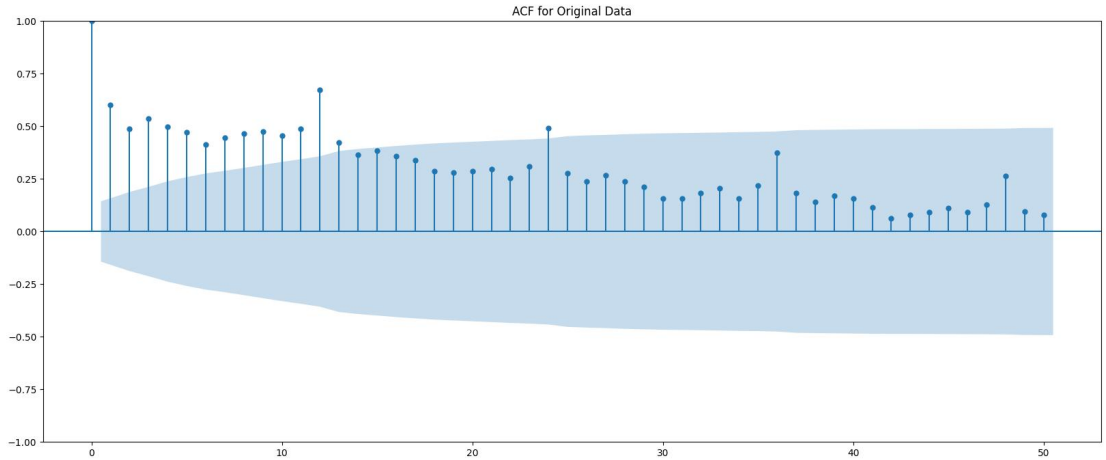
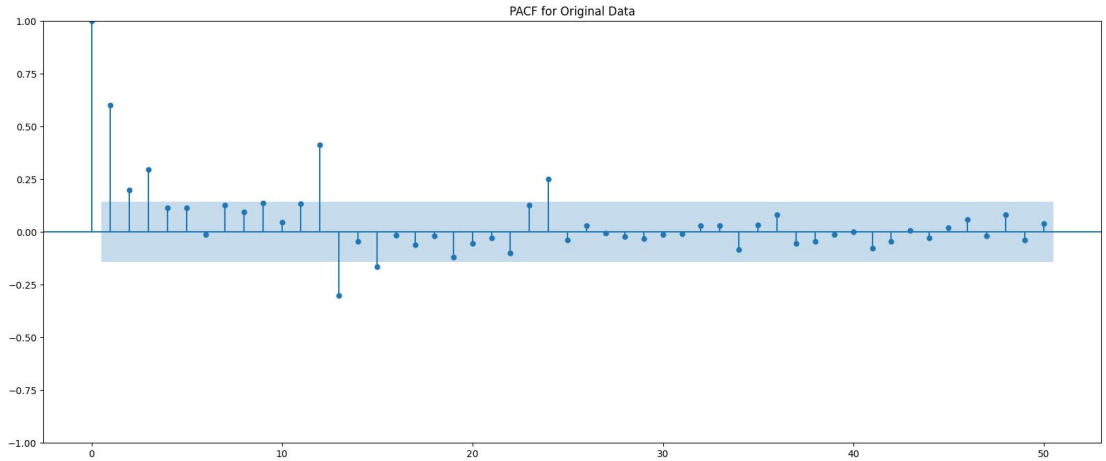
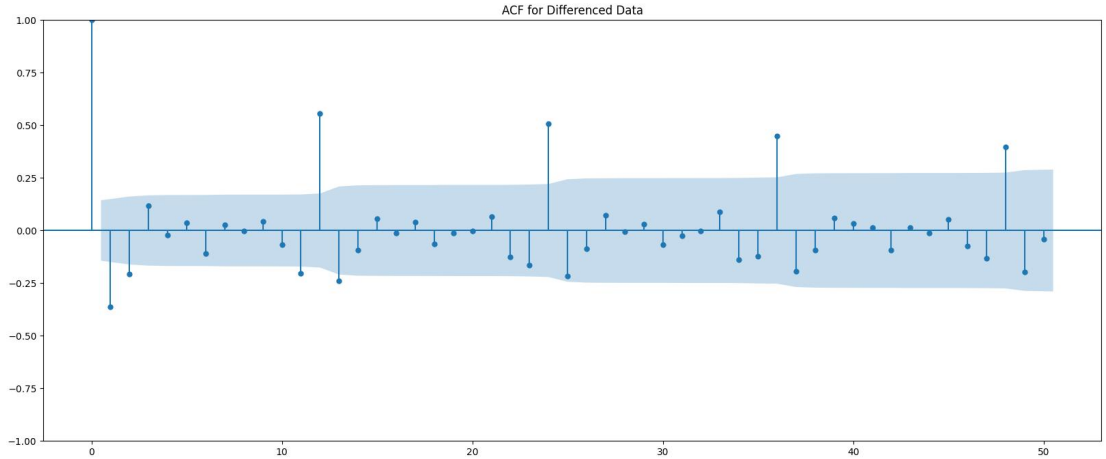


Fig 19:ACF and PACF plot

10.2.FINDING AR AND MA VALUES USING PACF AND ACF PLOT







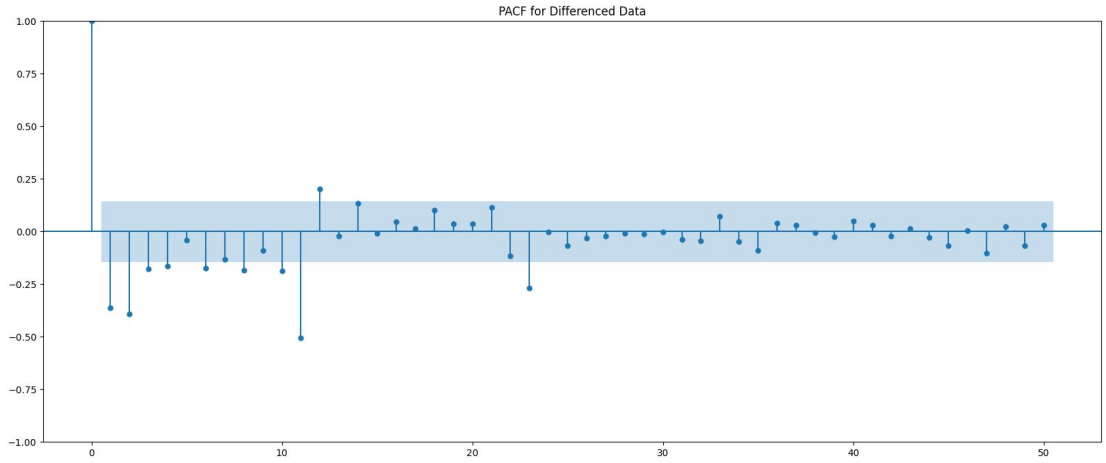


Fig 20:ACF and PACF plot on original and differenced data

- By anlayzing the ACF and PACF plot we have extracted the AR and MA as 5 and 3 with which we can perform the manual ARIMA and manual SARIMA.

10.3.ARIMA MODEL-AUTO

- Itertools is imported , which provides functions that create iterators for efficient looping.
- p and q are defined as the range from 0 to 2 (inclusive), which means it can take the values 0, 1, or 2.
- d is defined as a range from 1 to 2 (inclusive), which means it can only take the value 1.
- Each combinations of p,d,q are shown below.For the ARIMA model the p,d,q represents the parameters for the non-seasonal components,where:

    - p: The number of lag observations included in the model, also known as the autoregressive (AR) term.
    - d: The number of times that the raw observations are differenced to make the time series stationary.
    - q: The size of the moving average (MA) window, representing the lagged forecast errors in the prediction model.

- These parameters are crucial in defining the structure of the ARIMA model and tailoring it to capture the underlying patterns in the data effectively.

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)

An empty DataFrame named ARIMA_AIC is created with two columns: param and A IC.

'param': Will store the (p,d,q)parameter combinations for the ARIMA model using brute force approach.

'AIC': Will store the corresponding Akaike Information Criterion (AIC) values for each model. AIC is used to evaluate the goodness of fit of statistical models, with lower values indicating better fit.

	param	AIC
2	(0, 1, 2)	1259.247780
5	(1, 1, 2)	1259.473205
4	(1, 1, 1)	1260.036763
7	(2, 1, 1)	1261.014076
1	(0, 1, 1)	1261.327444
8	(2, 1, 2)	1261.472001
6	(2, 1, 0)	1278.135281
3	(1, 1, 0)	1297.077294
0	(0, 1, 0)	1313.175861

- With this the ARIMA model is fitted to a time series where the data is indexed by date.
- The summary provides insights into the model's fit and helps assess the chosen parameters' adequacy.
- Now,let us check how the model has performed on the test data for p=0,d=1,q=2.

	RMSE
ARIMA(0,1,2)	30.903931

Let us perfrom SARIMA model further and check the RMSE score.

10.4.SARIMA MODEL-AUTO

List of comprehension SARIMA parameter combinations are generated by including a fixed seasonal period of 6.The seasonal period of 6 was determined from looking at the seasonal part of the decomposition.

Varible pdq created which represents combinations of parameters for the non-seasonal ARIMA part of the model.

Variable model_pdq extends these combinations to include seasonal parameters for SARIMA.

Each combinations of p,d,q and P,D,Q are shown below.For the SARIMA model the p,d,q represents the parameters for the non-seasonal components ans P,D,Q represents the seasonal components where:

- p: The number of lag observations included in the model, also known as the autoregressive (AR) term.
- d: The number of times that the raw observations are differenced to make the time series stationary.
- q: The size of the moving average (MA) window, representing the lagged forecast errors in the prediction model.

P - Seasonal AutoRegressive (SAR) Order:

- The number of lagged seasonal terms to include in the model.
- It accounts for relationships between observations at the same position in different seasonal cycles (e.g., sales in December this year vs. December last year).

D- Seasonal Differencing:

- The number of times the data needs to be differenced to remove seasonal trends and achieve stationarity.
- If the data exhibits a recurring seasonal pattern, applying a seasonal differencing step (e.g., subtracting values from 12 months ago) can help remove it.

Q - Seasonal Moving Average (SMA) Order:

- The number of lagged forecast errors to include in the seasonal model.

- This captures the influence of past seasonal prediction errors on the current seasonal period.

- These parameters are crucial in defining the structure of the SARIMA model and tailoring it to capture the underlying patterns in the data effectively.

Some parameter combinations for the Model
Model: (0, 1, 1)(0, 0, 1, 6)
Model: (0, 1, 2)(0, 0, 2, 6)
Model: (1, 1, 0)(1, 0, 0, 6)
Model: (1, 1, 1)(1, 0, 1, 6)
Model: (1, 1, 2)(1, 0, 2, 6)
Model: (2, 1, 0)(2, 0, 0, 6)
Model: (2, 1, 1)(2, 0, 1, 6)
Model: (2, 1, 2)(2, 0, 2, 6)

An empty DataFrame named ARIMA_AIC is created with two columns: param , AIC and seasonal.

'param': Will store the non-seasonal parameters (p,d,q) parameter combinations for the SARIMA model using Brute-force approach.

'seasonal': Will store the seasonal parameters (P,D,Q,S)parameter combinations for the SARIMA model.

'AIC': Will store the corresponding Akaike Information Criterion (AIC) values for each model. AIC is used to evaluate the goodness of fit of statistical models, with lower values indicating better fit.

	param	seasonal	AIC
26	(0, 1, 2)	(2, 0, 2, 6)	1025.823325
80	(2, 1, 2)	(2, 0, 2, 6)	1029.198820
53	(1, 1, 2)	(2, 0, 2, 6)	1031.313974
71	(2, 1, 1)	(2, 0, 2, 6)	1034.131058
44	(1, 1, 1)	(2, 0, 2, 6)	1034.920837

- With this the SARIMA model is fitted to a time series where the data is indexed by date.
- The summary provides insights into the model's fit and helps assess the chosen parameters' adequacy.

- Now,let us check how the model has performed on the test data for p=0,d=1,q=2 and P=2,D=0,Q=2,S=6.

RMSE Score of 25.766391011931212 for SARIMA.

10.5.ARIMA MODEL-MANUAL

With the AR=5 and MA=3 taken from PACF and ACF plot and d=1 of differencing order lets us perform ARIMA AND SARIMA

The RMSE score was 22.136673648163573.

10.6.SARIMA MODEL-MANUAL

With the AR=5 and MA=3 taken from PACF and ACF plot and P=0 ,Q=1 and D=0 by observing the PACF and ACF model.

The RMSE score was 29.49283058446138

11. COMPARING PERFROMANCE OF ALL MODELS:

	Test RMSE
Alpha=0.1015,Beta=0.0,Gamma=0.0,TripleExponentialSmoothing	9.324844
2pointTrailingMovingAverage	11.801167
4pointTrailingMovingAverage	15.366538
Alpha=0.3,Beta=0.6,Gamma=0.4,TripleExponentialSmoothing	15.503837
6pointTrailingMovingAverage	15.863483
9pointTrailingMovingAverage	16.342156
RegressionOnTime	17.356199
Alpha=0.5,Beta=0.5,DoubleExponentialSmoothing	17.547521
Alpha=0.9,SimpleExponentialSmoothing	22.496993
Alpha=0.127,SimpleExponentialSmoothing	29.223811
SimpleAverage	52.412167

	RMSE
ARIMA(1,0,0)	52.315237
SARIMA(0,1,2)(2,0,2,6)	25.766391
ARIMA(0,1,2)	30.903931
SARIMA(0,1,2)(2,0,2,12)	25.344039
ARIMA(0,1,2)	30.903931

When comapring all the models done the Triple Exponential Smoothing of auto with Alpha = 0.1015 ,beta= 0.0 and gamma=0.0 has the lowest score of RMSE with 9.324844.Thus this model is selected as the best model.

12. REBUILDING THE OPTIMUM MODEL ON THE ENTIRE DATA.

After having the Triple Exponential Smoothing as the best model ,lets us see how it perform on the entire data.

The RMSE score of 16.0962 for the entire data.

13. FORECASTING FOR NEXT 12 MONTHS

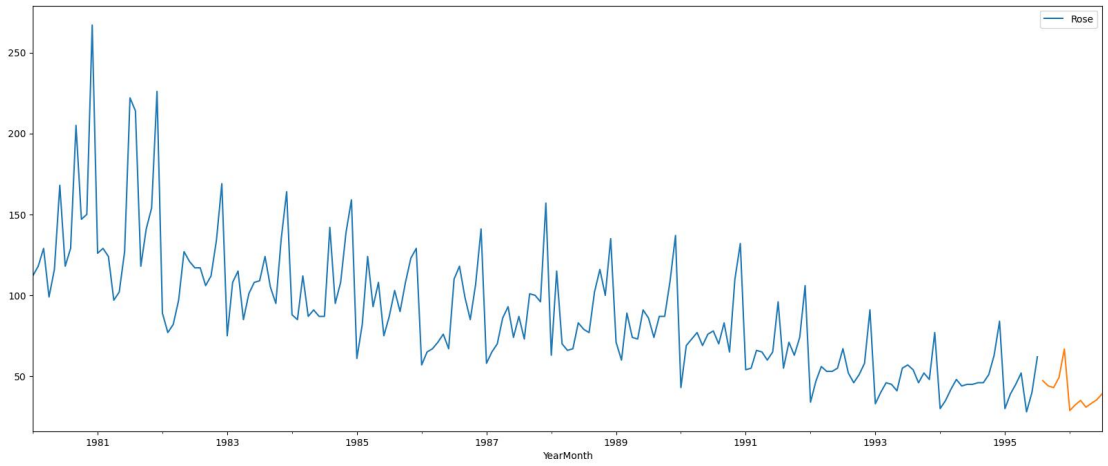
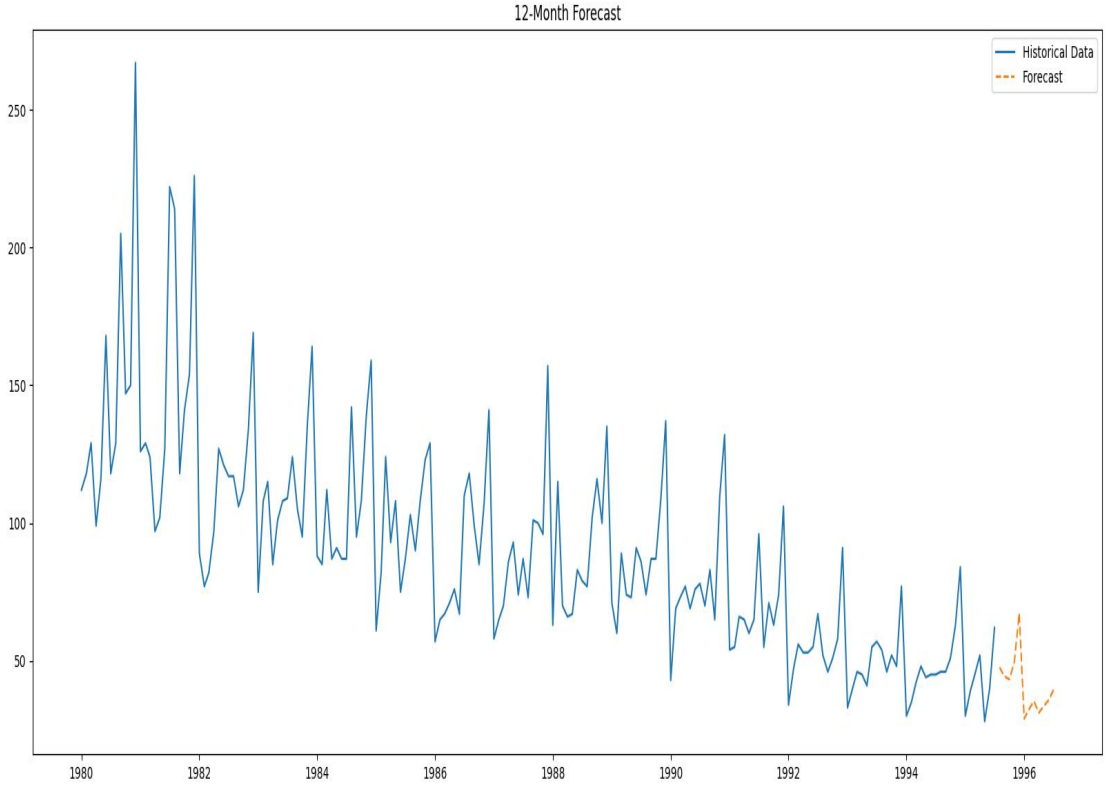


Fig 21:12 months forecast

With trend as additive and seasonal as multiplicative and seasonal peroid to 12 months,the forecasted value as below.

1995-08-01	48.0
1995-09-01	44.0
1995-10-01	43.0
1995-11-01	50.0
1995-12-01	67.0
1996-01-01	29.0
1996-02-01	32.0
1996-03-01	35.0
1996-04-01	31.0
1996-05-01	34.0
1996-06-01	36.0
1996-07-01	40.0



14. ACTIONABLE INSIGHTS AND RECOMMENDATIONS

- The sales for the label *Rose* have been declining year by year.
- Sales tend to peak during the months of November and December. A proper strategy should be devised to evaluate whether to continue producing this label or to discontinue its production.
- Comparisons should be made with competitors' pricing and performance. Additionally, the company should explore potential market opportunities, identify areas for growth, and seek ways to improve sales.
- The BCG matrix can be utilized to assess the position of this label in the market and aid in decision-making.

CONTENTS - FOR LABEL 'SPARKLING' A PRODUCT OF ABC ESTATE WINES

LIST OF FIGURES

LIST OF TABLES

1.PROBLEM DEFINITION

1.1Context:

The 20th century laid the foundation for modern wine production, distribution, and appreciation.It established wine as a global commodity, with distinct reputations for Old World and New World producers.The rise of wine tourism, education, and critique systems created a thriving culture around wine.This century marked the transition of wine from a regional agricultural product to a global industry and cultural symbol.

The wine industry beyond the 20th century is characterized by adaptability and innovation. As the world faces environmental and economic challenges, the industry continues to thrive by embracing sustainability, leveraging technology, and appealing to the diverse tastes of a global consumer base. This era is marked by a balance between preserving age-old traditions and pioneering modern practices.

The sales of wine have undergone significant transformations over the years, influenced by changing consumer preferences, global market dynamics, and innovations in production and distribution.

1.2.Objective:

The primary objective is to analyze 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, capitalize on emerging market opportunities, and maintain a competitive edge in the wine industry.

2.DATA BACKGROUND AND CONTENTS

2.1.Purpose:

The purpose of collecting the data was to forecast the sales of the product **SPARKLING** from ABC Estate Wines. This forecasting aims to help the company enhance their strategies and develop new ideas to maintain a competitive position in the market.

2.2.Data Description and Dictionary:

The data provided is the sales of each month from 01-1980 to 07-1995 for the product **Sparkling.**

Data Dictionary

- **YearMonth:** Contains year and month
- **Sparkling:** Contains sales with respect to yearmonth column

2.3.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
```

- The variables from the dataset are of object and int.
- With no missing values.

3.READING THE DATA AS APPROPRIATE TIME SERIES DATA:

- The column YearMonth and Sparkling are of object which needs to be converted to datatime.
- The Yearmonth is converted into datetime object and yearmonth column is set as index of the dataframe.The time index allows pandas to recognize the data as a time series.
- Since time series data inherently relies on time as the organizing principle, many time series models depend on an indexed time column for accurate forecasting and analysis.

4.EDA

4.1.PLOTING TIME SERIES FOR UNDERSTANDING THE BEHAVIOUR OF THE DATA

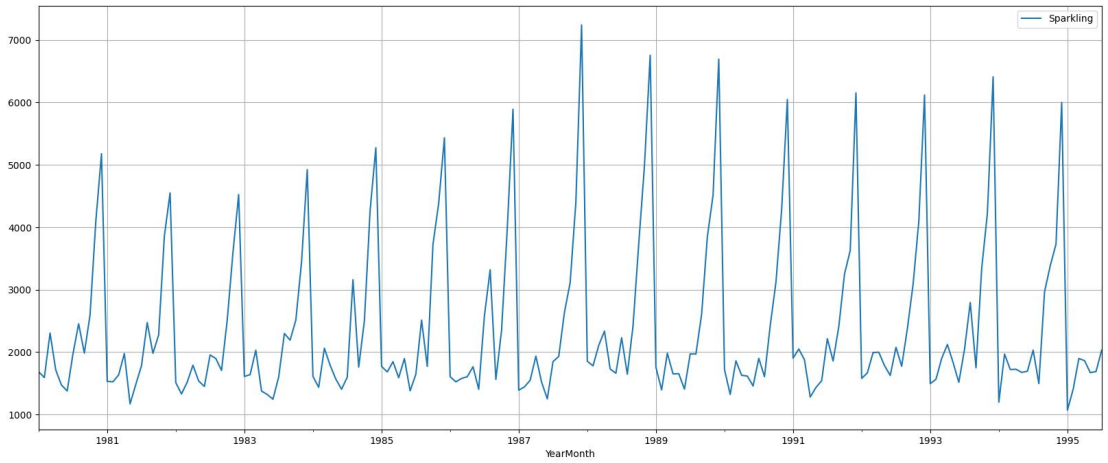


Fig 1:Time series

We can see that there is stabilized trend with proper traces of seasonal pattern associated.

4.2.YEARLY BOXPLOT

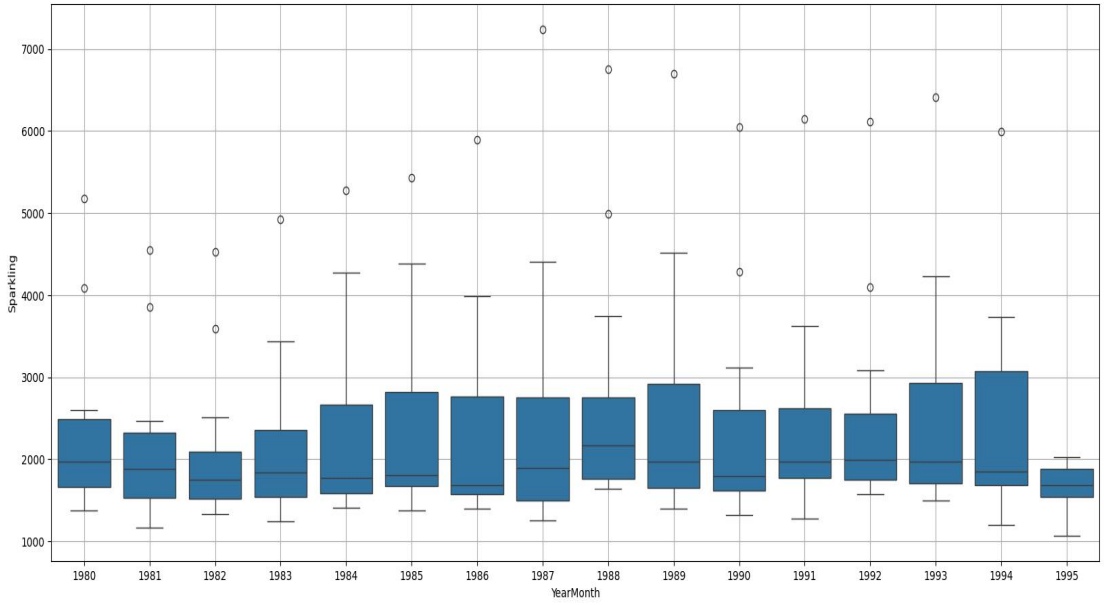


Fig 2:Yearly Boxplot

The sales for the year are stable,except for the year 1995.

4.3.MONTHLY BOXPLOT

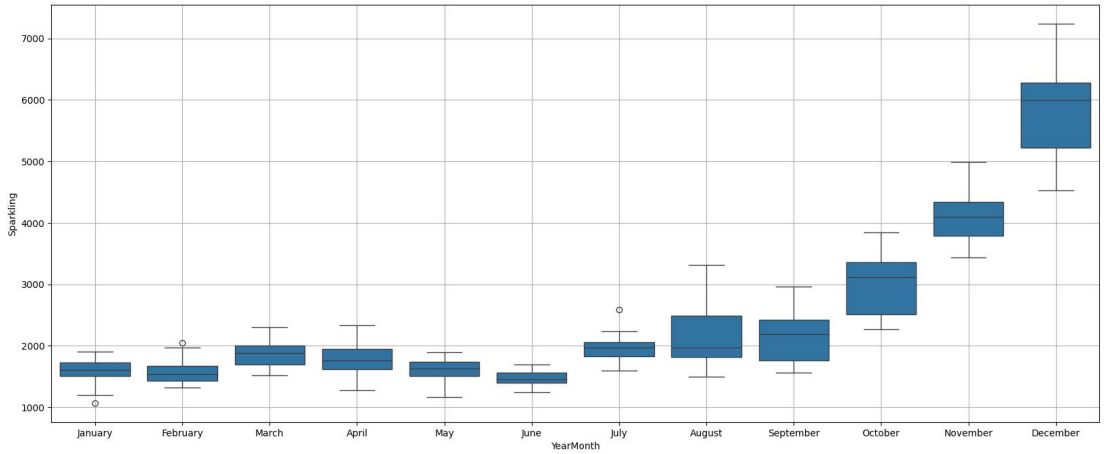


Fig 3:Monthly Boxplot

The sales for month November and December for every year were high,this might be of christmas and new year time.

4.4.SPREAD OF ACCIDENTS ACROSS DIFFERENT YEARS AND WITHIN DIFFERENT MONTHS ACROSS YEARS:

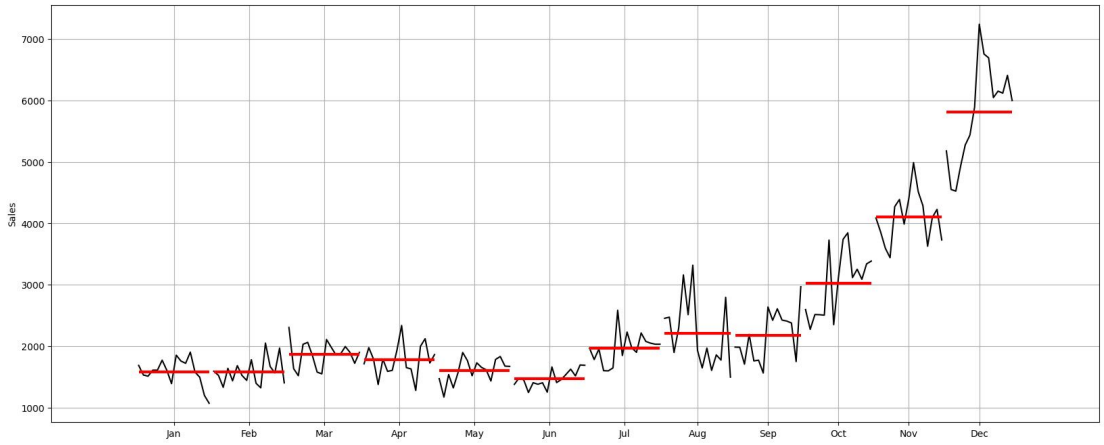


Fig 4:Spread of sales for each month over years

The red line indicates the median value ,the median value for october ,november and december are high.

4.5.PIVOT TABLE:

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 1:Pivot Table

This pivot table gives clear indication of sales for each month of each year,this clearly shows high sales year by year.

4.6.MONTHLY SALES ACROSS YEARS:

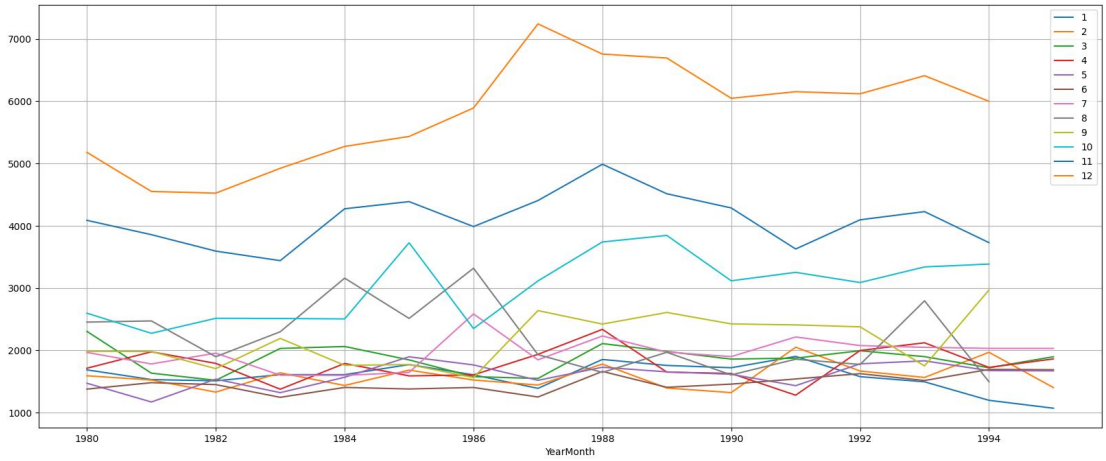


Fig 5:Monthly sales across years

The sales for december alone stands out.

4.7.EMPIRICAL CUMULATIVE DISTRIBUTION

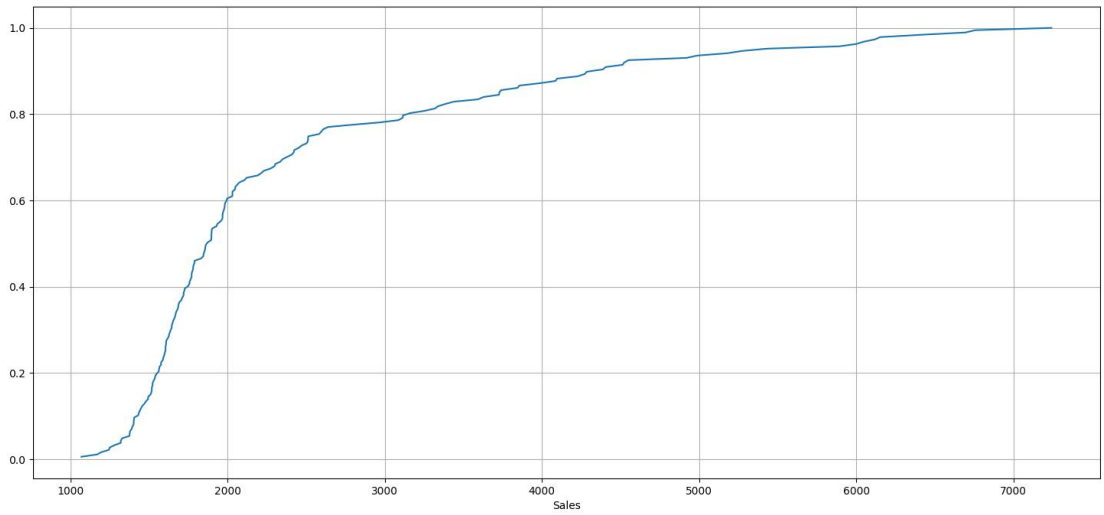


Fig 6:Empirical cumulative distribution

80% of the sales are below 3000 and sales are not beyond 7500.

4.8.AVERAGE SALES PER MONTH AND THE MONTH ON MONTH PERCENTAGE CHANGE OF SALES.

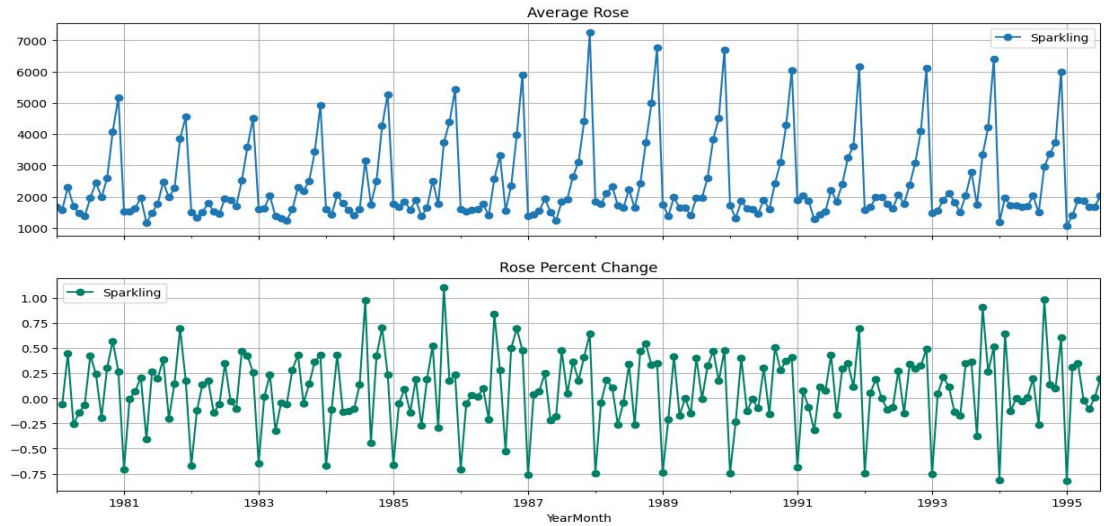


Fig 7:Average Sales and percentage change of sales per month

Every year the sales increases garadually month by month.The sales at the start of the year is higher compared to end of the year.

5.DECOMPOSITION OF TIME SERIES

5.1.ADDITIVE

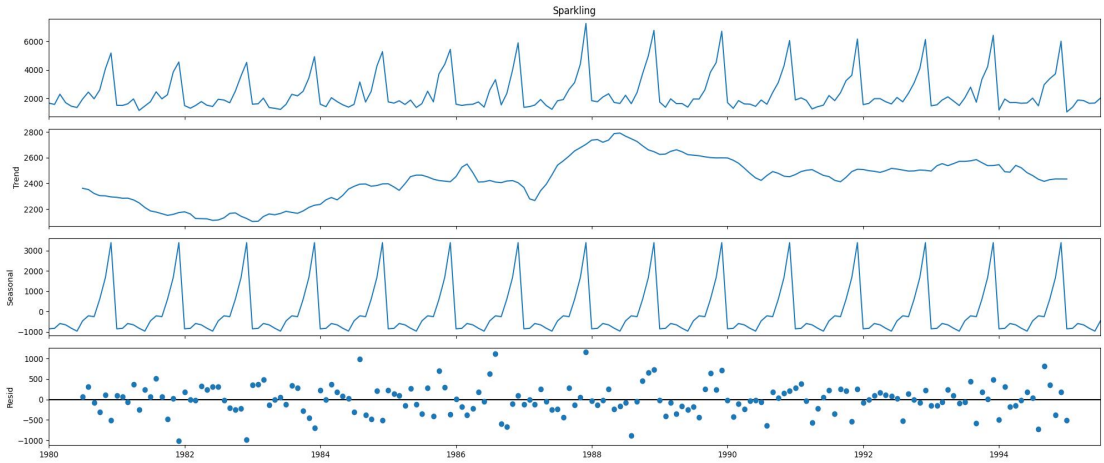


Fig 8:Decomposition-Additive

Which model better suits the data,the residual part shows which can be further used in the model building,looking at the residuals of additive shows the error are not normally distributed,not constant along zero.

5.2.MULTIPLICATIVE

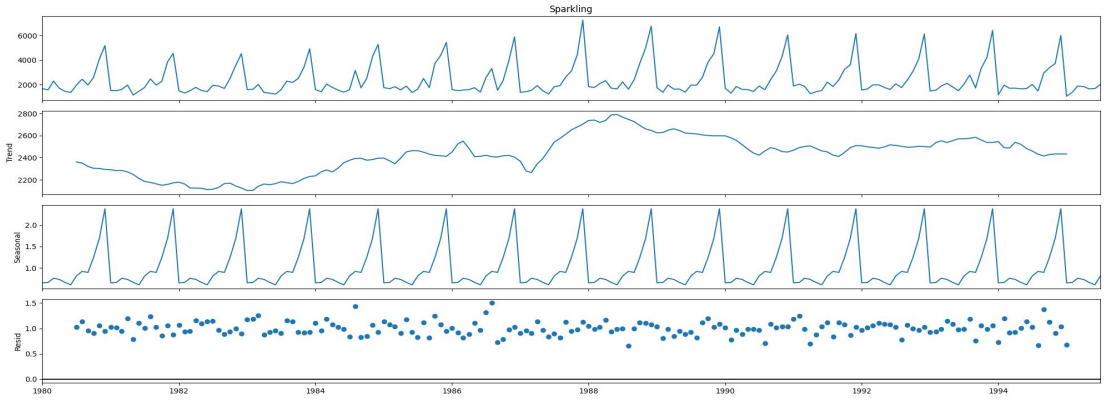


Fig 9:Decomposition-Multiplicative

Looking at the residuals of multiplicative shows the error are not normally distributed and not constant along one.

6.TRAIN AND TEST SPLIT:

- First 70% of the rows are selected as the train set.
- Remaining 30% of the rows are selected as test set.

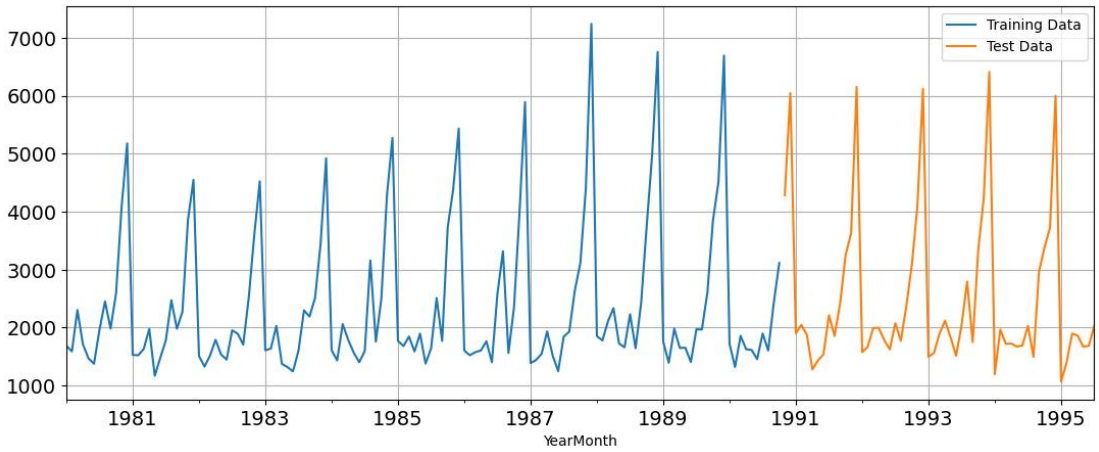


Fig 10:Train and Test split

7.MODEL BUILDING ON ORIGINAL DATA

7.1.LINEAR REGRESSION

- Time indices is created for the train and test dataset to align with the time series, with time instance for train data from 1 to 130 and test data from 131 to 187.
- Linear regression model requires independent and dependent variable,hence the time column created acts as the independent variable and the column Rose acts as the dependent variable.
- Now,let us check how the model has performed on the test data.

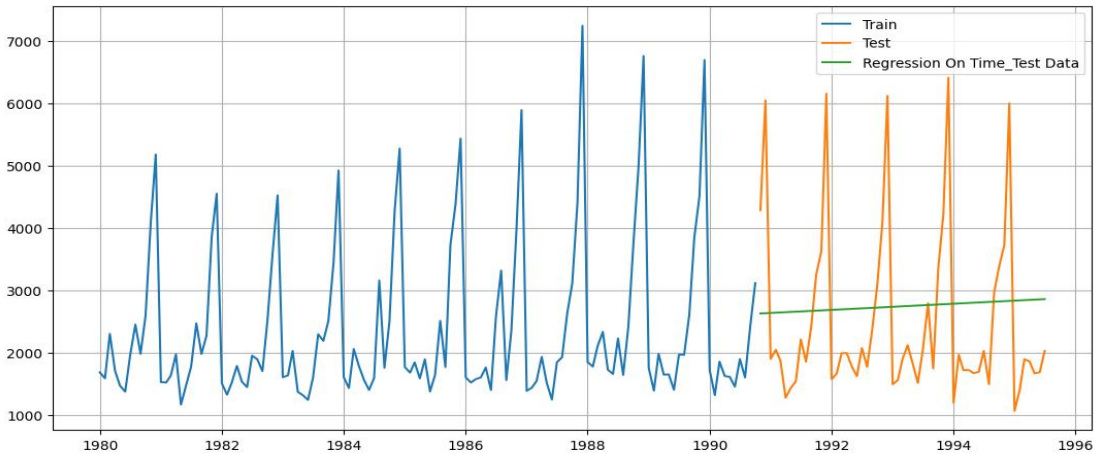


Fig 11:Linear Regression

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data by measuring the average magnitude of prediction errors.

	Test RMSE
RegressionOnTime	1392.438305

- Let us continue with other forecasting model and check the score of RMSE.

7.2.SIMPLE AVERAGE:

- The simple average method takes out the average of the column Rose.
- The mean value will be repeated for each row in the newly created meanforecast column.
- Now,let us check how the model has performed on the test data.

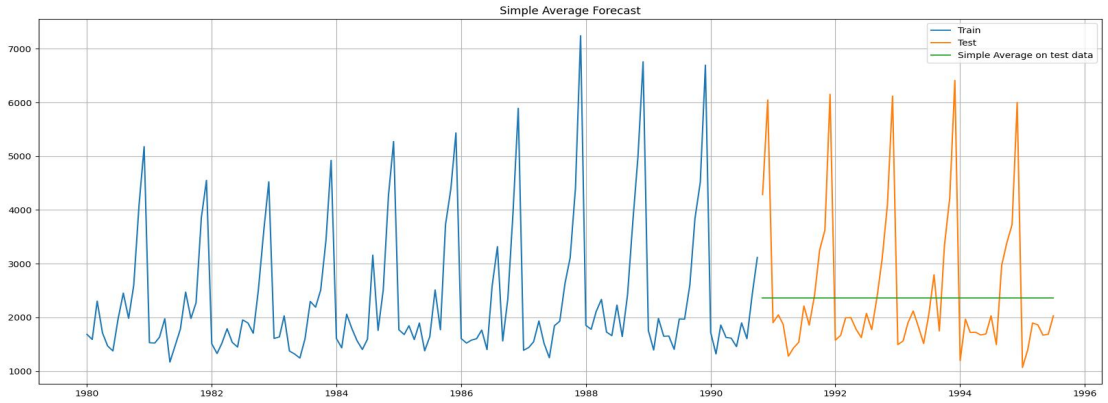


Fig 12:Simple Average

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

	Test RMSE
SimpleAverage	1368.746717

- Let us continue with other forecasting model and check the score of RMSE.

7.3.MOVING AVERAGE:

- Let us calculate the moving average by rolling() which is used to create a rolling window of a specified size over the data. The mean() function then computes the average of the values within that window.
- Moving average is done for different window sizes (2, 4, 6, and 9) and stores the results in new columns.
- Now,let us check how the model has performed on the test data.

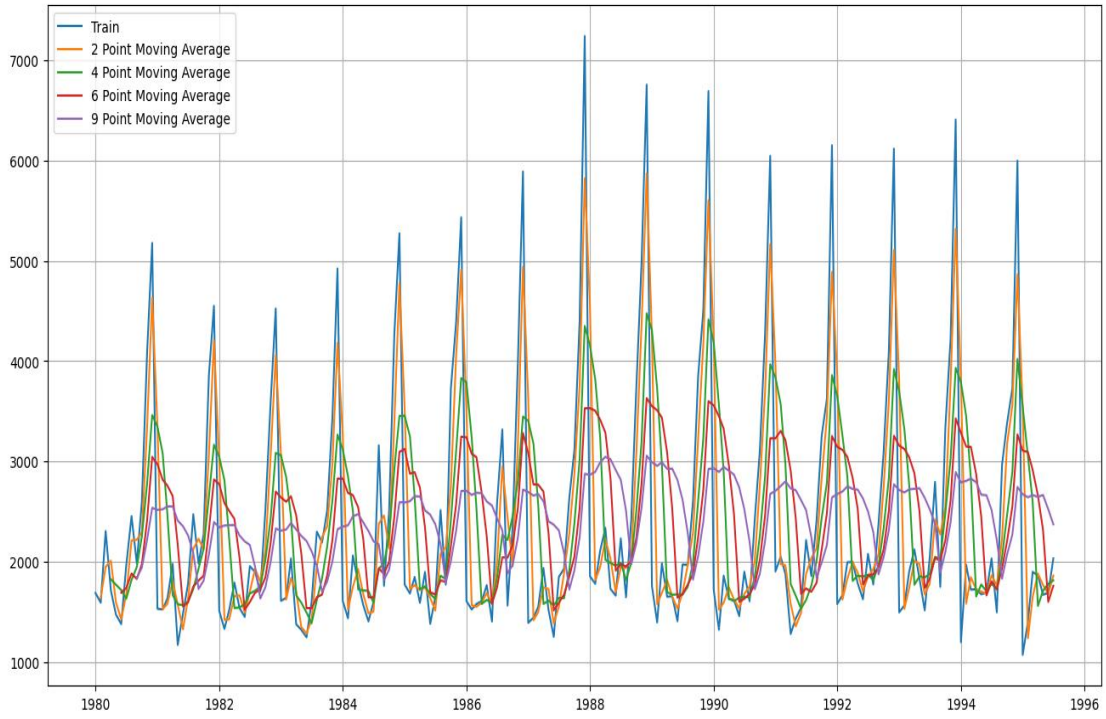


Fig 13:Moving Average

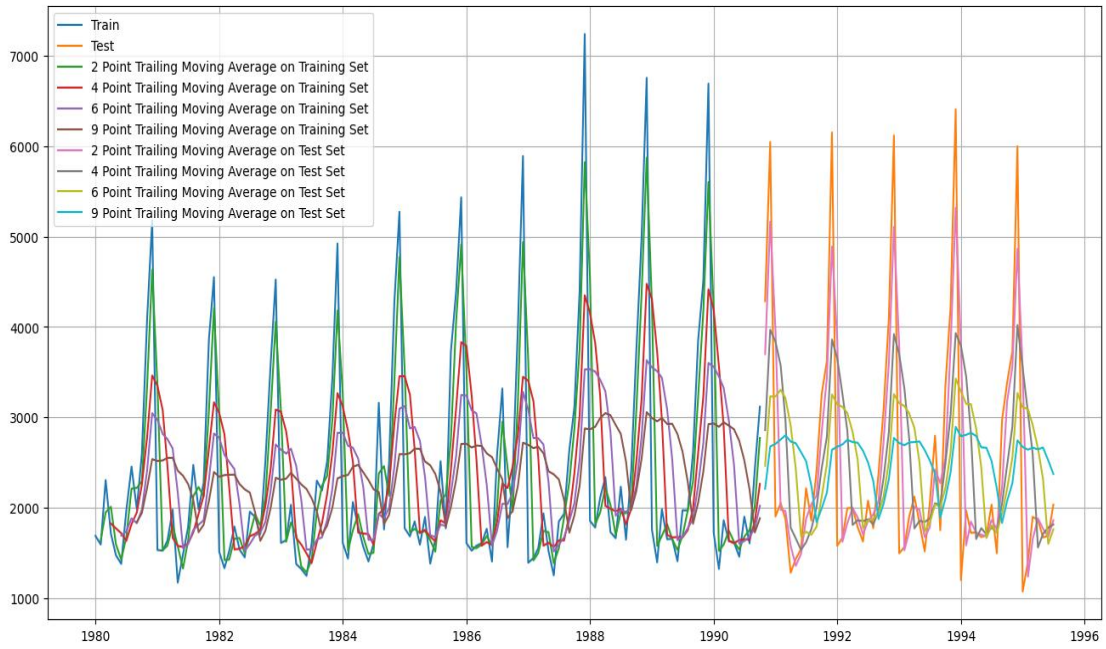


Fig 14:Moving Average on train and test set

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

For 2 point Moving Average Model forecast on the Training Data, RMSE is 811.179
For 4 point Moving Average Model forecast on the Training Data, RMSE is 1184.21
For 6 point Moving Average Model forecast on the Training Data, RMSE is 1337.20
For 9 point Moving Average Model forecast on the Training Data, RMSE is 1422.65

- Let us continue with other forecasting model and check the score of RMSE.

7.4.SINGLE EXPONENTIONAL SMOOTHING

- Simple Exponential Smoothing (SES) model is initialized for time series forecasting on the Rose column of the SES_train DataFrame.

- SES is a forecasting method that applies weights to past observations, with the most recent observations receiving the highest weights.

- SES uses a smoothing parameter (often denoted as α, alpha) that ranges from 0 to 1. This parameter determines the weight given to the most recent observation compared to the previous forecast.

- If α is close to 1, the model will put more weight on recent observations, making it more responsive to changes.
- If α is close to 0, the model will rely more on historical averages, making it less sensitive to recent fluctuations.
- The variable model_SES_autofit contain a fitted SES model object obtained from calling the fit() method on a SimpleExpSmoothing model.
- The expression model_SES_autofit.params is used to access the estimated parameters of a fitted Simple Exponential Smoothing (SES) model from the statsmodels library.
- The params attribute retrieves the parameters of the fitted SES model. In SES, the key parameter is the smoothing level (often denoted as alpha, α).

{'smoothing_level': 0.03753429913783034,
'smoothing_trend': nan,
'smoothing_seasonal': nan,
'damping_trend': nan,
'initial_level': 1686.0,
'initial_trend': nan,
'initial_seasons': array([], dtype=float64),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the SES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

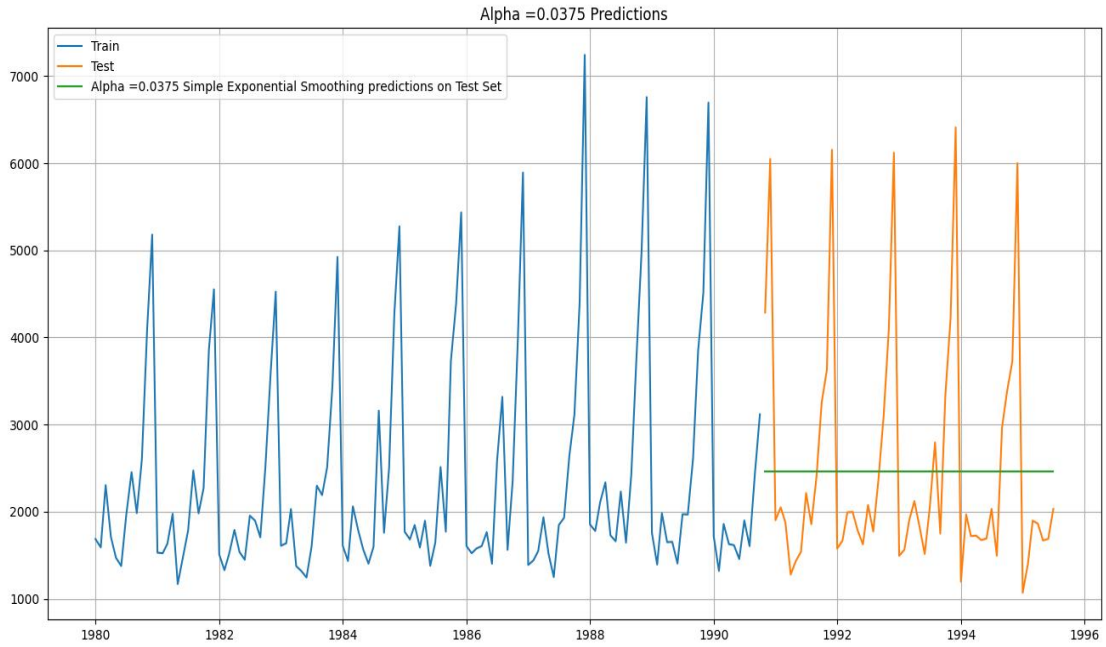


Fig 15:Single Exponential Smoothing

- The score of the evaluation metric **RMSE (Root Mean Squared Error)** indicates how well the model has performed on the test data,let us check the score.

For Alpha =0.0375 Simple Exponential Smoothing Model forecast on the Test Da ta, RMSE is 1362.429

- Let us also perform manual alpha values to check the performance at different alpha values.

	Alpha Values	Train RMSE	Test RMSE
3	0.4	1329.814823	1363.037803
4	0.5	1326.403864	1364.863549
0	0.1	1298.211536	1367.395642
2	0.3	1331.102204	1372.323705
1	0.2	1322.658289	1378.320562
5	0.6	1325.588422	1379.988733
6	0.7	1329.257530	1404.659104
7	0.8	1337.879425	1434.578214
8	0.9	1351.645478	1466.179706

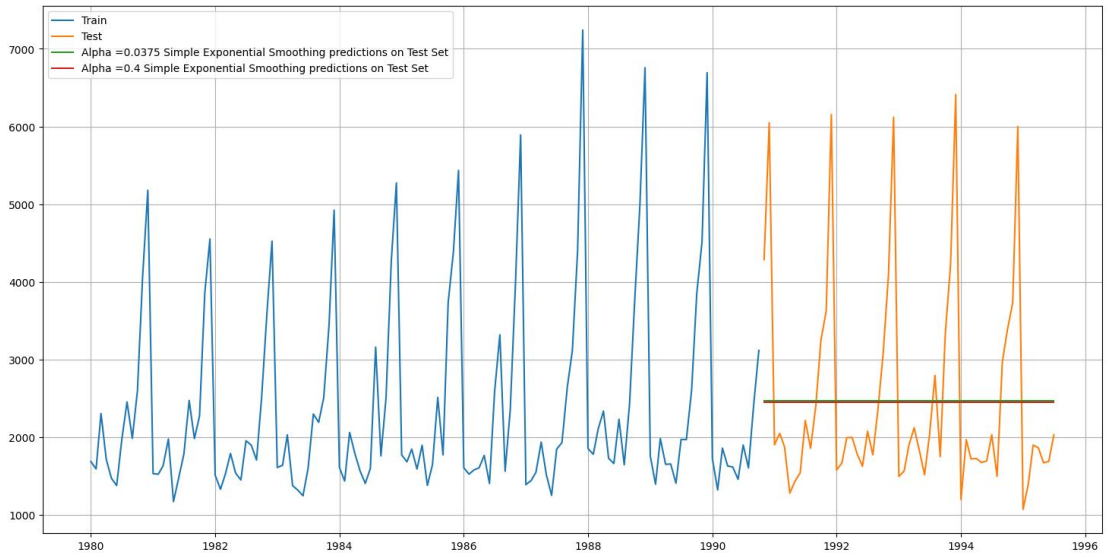


Fig 16:Single Exponential Smoothing

- The manual processed alpha at 0.9 has better RMSE scores compared to auto alpha at 0.127

7.5.DOUBLE EXPONENTIONAL SMOOTHING

- We have fitted a Double Exponential Smoothing (DES) model using

the Exponential Smoothing class from the statsmodels library,with an additive trend model with no seasonal pattern in the model.

- We have fitted a Double Exponential Smoothing model to the time series data while optimizing the smoothing parameters automatically.
- The params attribute retrieves the parameters of the fitted DES model. In DES, the key parameter is the smoothing level and smoothing trend (often denoted as alpha α and Beta).

{'smoothing_level': 0.07568138427743348,
'smoothing_trend': 0.07564849713133869,
'smoothing_seasonal': nan,
'damping_trend': nan,
'initial_level': 1505.5981603919731,
'initial_trend': 6.848682802199605,
'initial_seasons': array([], dtype=float64),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the DES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

For Alpha =0.0756 and Beta =0.0756 Simple Exponential Smoothing Model forec ast on the Test Data, RMSE is 1362.429

- Let us also perform manual alpha and beta values to check the performance at different alpha and beta values.

	Alpha Values	Beta Values	Train RMSE	Test RMSE
0	0.3	0.3	1565.420584	1391.168123
1	0.3	0.4	1660.933342	2027.619573
8	0.4	0.3	1555.966630	2604.106418
16	0.5	0.3	1525.288403	3979.170916
2	0.3	0.5	1757.552973	4102.282829

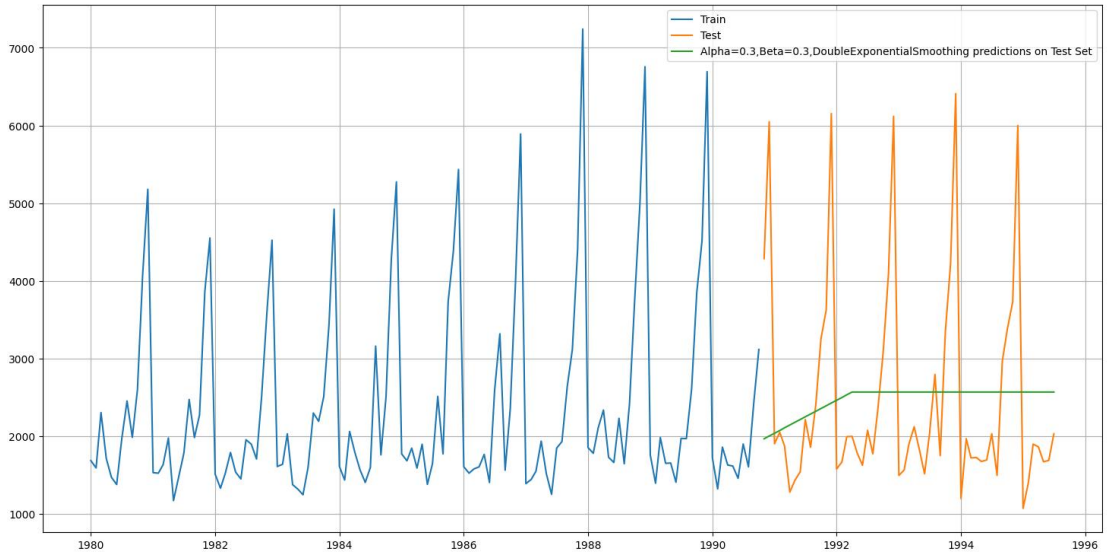


Fig 17:Double Exponential Smoothing

The auto processed RMSE at alpha and beta =0.0756 has better score compared to auto alpha and beta =0.3

7.6.TRIPLE EXPONENTIONAL SMOOTHING

- We have fitted a Triple Exponential Smoothing (DES) model using

the Exponential Smoothing class from the statsmodels library,with an additive trend model with seasonal pattern as additive.

- We have fitted a Triple Exponential Smoothing model to the time series data while optimizing the smoothing parameters automatically.

- The params attribute retrieves the parameters of the fitted TES model. In TES, the key parameter is the smoothing level , smoothing trend and smoothing seasonal (often denoted as alpha α ,Beta and gamma).

{'smoothing_level': 0.07598944334560086,
'smoothing_trend': 0.03257983107648835,
'smoothing_seasonal': 0.47924862401964097,
'damping_trend': nan,
'initial_level': 2356.5262910513907,
'initial_trend': -0.7868042592276397,
'initial_seasons': array([-636.24270225, -722.99141342, -398.63320667, -473.44414522,
-808.44189536, -815.36463524, -384.23734682, 72.99624204,
-237.45744559, 272.31932254, 1541.3947672 , 2590.08848375]),
'use_boxcox': False,
'lamda': None,
'remove_bias': False}

- A new column named 'predict' to the TES_test DataFrame.
- Forecasted values equal to the length of the test dataset is produced which is typically the portion of the dataset used to evaluate the model's performance.
- Now,let us check how the model has performed on the test data.

For Alpha=0.0759,Beta=0.0325,Gamma=0.4792, Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 366.947

- Let us also perform manual alpha and beta values to check the performance at different alpha ,beta values and gamma values.

	Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE
6	0.3	0.3	0.9	506.909702	1142.778891
384	0.9	0.3	0.3	546.764241	1155.154361
264	0.7	0.4	0.3	522.090986	1168.325276
337	0.8	0.5	0.4	585.886668	1181.964081
320	0.8	0.3	0.3	521.173565	1187.359541

The auto processed at alpha = 0.0759 and beta =0.0325 and gamma= 0.4792 has better score compared to manual.

Lets us compare the RMSE scores for the models so far.

	Test RMSE
Alpha=0.0759,Beta=0.0325,Gamma=0.4792,TripleExponentialSmoothing	366.947443
2pointTrailingMovingAverage	811.178937
Alpha=0.3,Beta=0.3,Gamma=0.9,TripleExponentialSmoothing	1142.778891
4pointTrailingMovingAverage	1184.213295
6pointTrailingMovingAverage	1337.200524
Alpha=0.0375,SimpleExponentialSmoothing	1362.428949
Alpha=0.4,SimpleExponentialSmoothing	1363.037803
SimpleAverage	1368.746717
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	1391.168123
RegressionOnTime	1392.438305
9pointTrailingMovingAverage	1422.653281

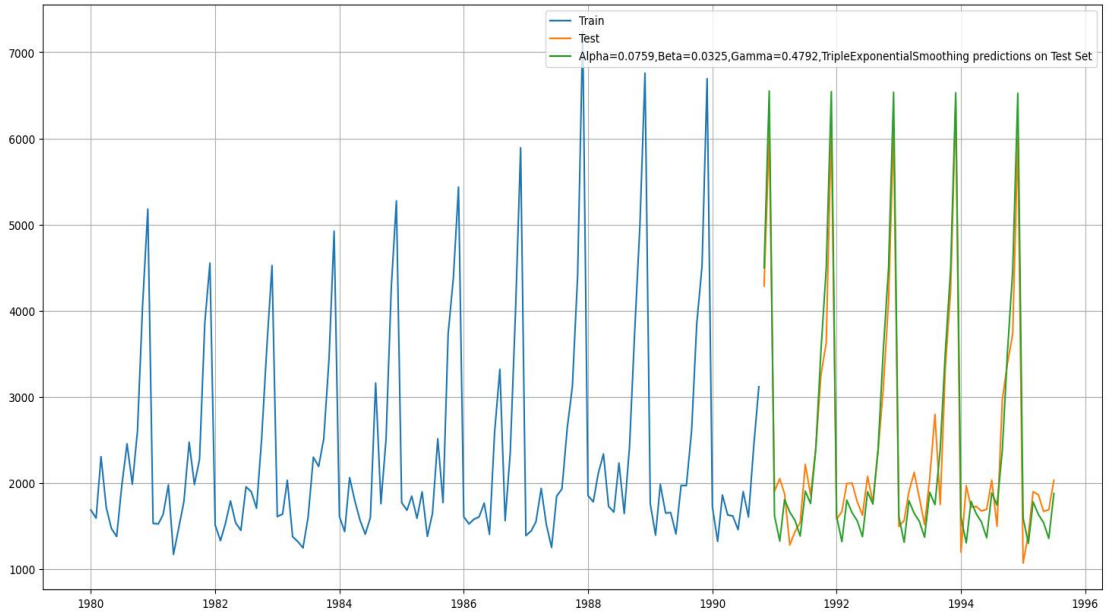
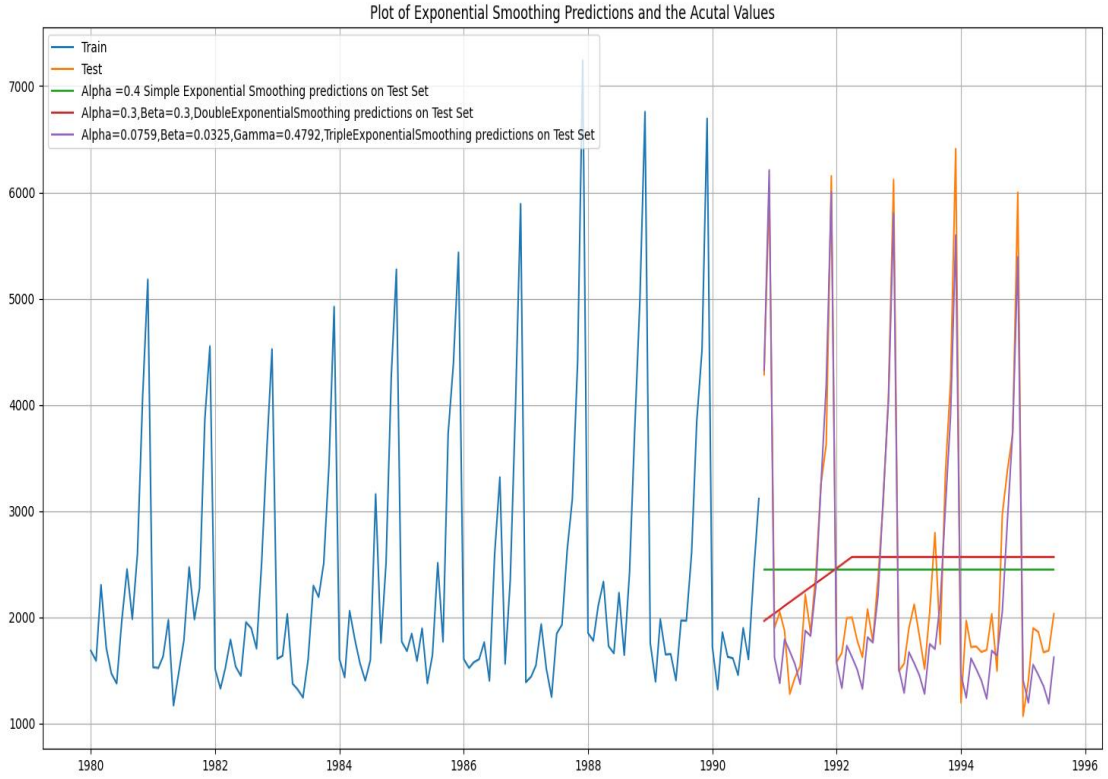


Fig 18:Triple Exponential Smoothing



Further,let us perform the ARIMA and SARIMA model,for that the these model assumes that the entire data is stationary.Let us perform stationarity check to continue with ARIMA and SARIMA.

8.CHECK FOR STATIONARITY

To check the stationarity of the data lets us perform Augmented Dickey-Fuller test.

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

- We check the p-value to know whether the data is stationary or not.
- We formulate hypothesis for the result.

Null Hypothesis:The data is non-stationary.

Alternate Hypothesis: The data is stationary.

- Since the p-value id greater than 0.05 we fail to reject the null hypothesis thus indicating data is non-stationary.Further we perform one order differencing and lest check the result.

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

- Since the p-value id less than 0.05 reject the null hypothesis thus indicating data is stationary.

9.MODEL BUILDING ON STATIONARY DATA

9.1.ACF AND PACF PLOT

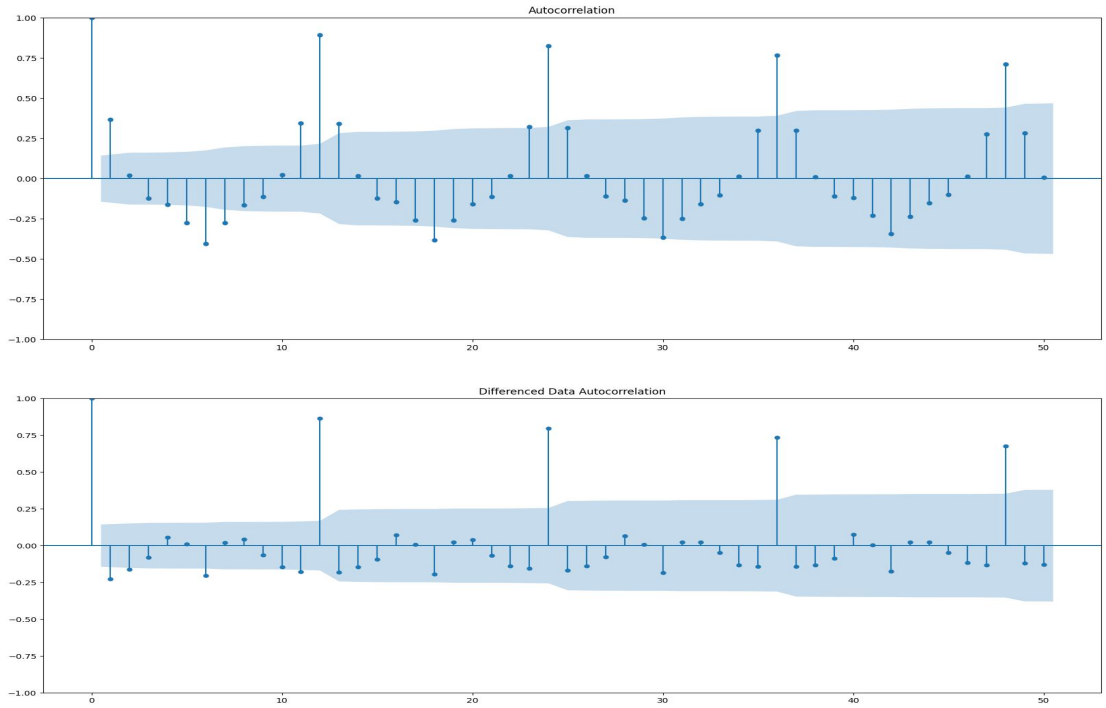
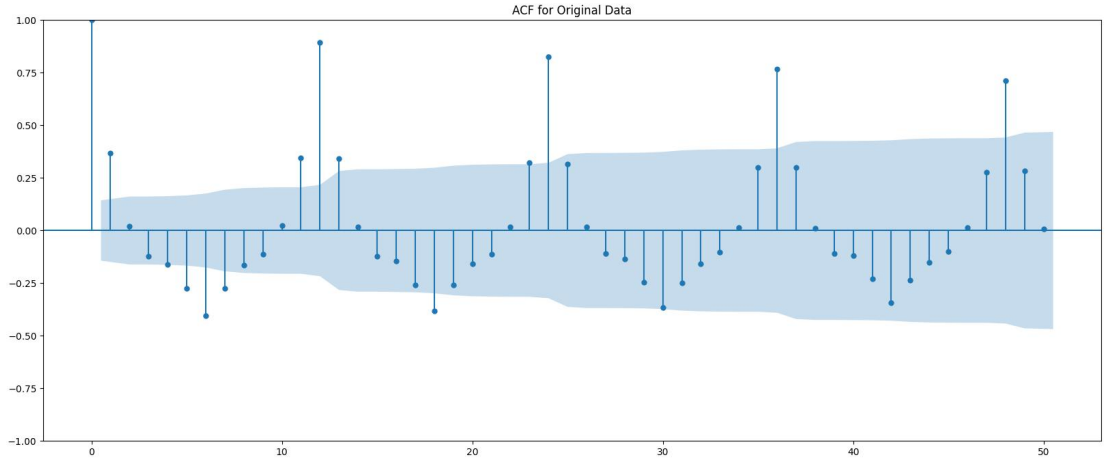


Fig 19:ACF and PACF plot

9.2.FINDING AR AND MA VALUES USING PACF AND ACF PLOT



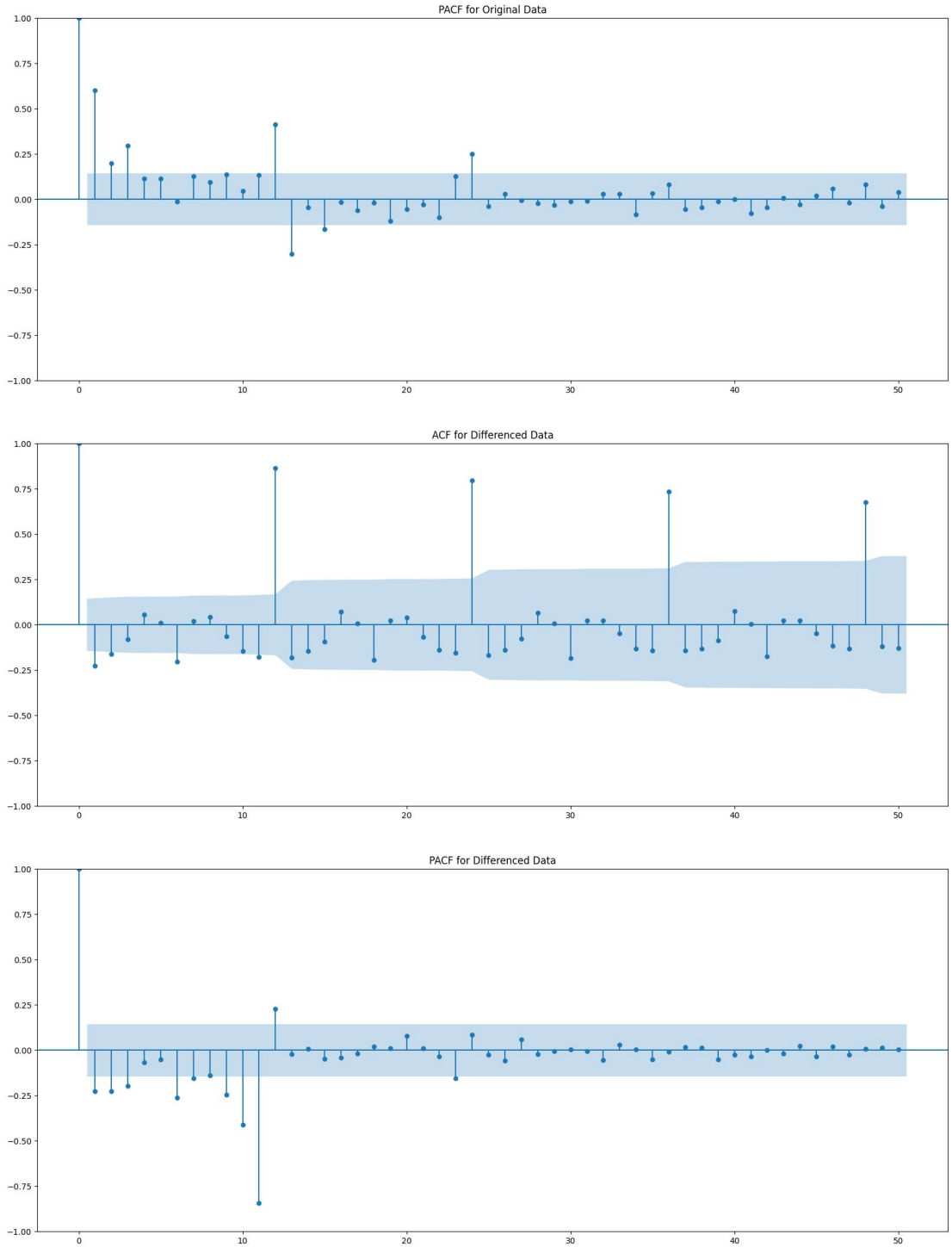


Fig 20:ACF and PACF plot on original and differenced data

- By anlayzing the ACF and PACF plot we have extracted the AR and MA as 4 and 3 with which we can perform the manual ARIMA and manual SARIMA.

9.3.ARIMA MODEL-AUTO

- Itertools is imported , which provides functions that create iterators for efficient looping.
- p and q are defined as the range from 0 to 2 (inclusive), which means it can take the values 0, 1, or 2.
- d is defined as a range from 1 to 2 (inclusive), which means it can only take the value 1.
- Each combinations of p,d,q are shown below.For the ARIMA model the p,d,q represents the parameters for the non-seasonal components,where:
    - p: The number of lag observations included in the model, also known as the autoregressive (AR) term.
    - d: The number of times that the raw observations are differenced to make the time series stationary.
    - q: The size of the moving average (MA) window, representing the lagged forecast errors in the prediction model.
- These parameters are crucial in defining the structure of the ARIMA model and tailoring it to capture the underlying patterns in the data effectively.

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)

An empty DataFrame named ARIMA_AIC is created with two columns: param and A IC.

"param': Will store the (p,d,q)parameter combinations for the ARIMA model using brute force approach.

'AIC': Will store the corresponding Akaike Information Criterion (AIC) values for each model. AIC is used to evaluate the goodness of fit of statistical models, with lower values indicating better fit.

	param	AIC
8	(2, 1, 2)	2178.109724
7	(2, 1, 1)	2193.974962
2	(0, 1, 2)	2194.034361
5	(1, 1, 2)	2194.959653
4	(1, 1, 1)	2196.050086
1	(0, 1, 1)	2217.939206
6	(2, 1, 0)	2223.899470
3	(1, 1, 0)	2231.137663
0	(0, 1, 0)	2232.719438

- With this the ARIMA model is fitted to a time series where the data is indexed by date.
- The summary provides insights into the model's fit and helps assess the chosen parameters' adequacy.

- Now,let us check how the model has performed on the test data for p=2,d=1,q=2.

	RMSE
ARIMA(2,1,2)	1325.166404

Let us perform SARIMA model further and check the RMSE score.

9.4.SARIMA MODEL-AUTO

List of comprehension SARIMA parameter combinations are generated by including a fixed seasonal period of 6.The seasonal period of 6 was determined from looking at the seasonal part of the decomposition.

Varible pdq created which represents combinations of parameters for the non-seasonal ARIMA part of the model.

Variable model_pdq extends these combinations to include seasonal parameters for SARIMA.

Each combinations of p,d,q and P,D,Q are shown below.For the SARIMA model the p,d,q represents the parameters for the non-seasonal components ans P,D,Q represents the seasonal components where:

- p: The number of lag observations included in the model, also known as the autoregressive (AR) term.
- d: The number of times that the raw observations are differenced to make the time series stationary.
- q: The size of the moving average (MA) window, representing the lagged forecast errors in the prediction model.

P - Seasonal AutoRegressive (SAR) Order:

- The number of lagged seasonal terms to include in the model.
- It accounts for relationships between observations at the same position in different seasonal cycles (e.g., sales in December this year vs. December last year).

D- Seasonal Differencing:

- The number of times the data needs to be differenced to remove seasonal trends and achieve stationarity.
- If the data exhibits a recurring seasonal pattern, applying a seasonal differencing step (e.g., subtracting values from 12 months ago) can help remove it.

Q - Seasonal Moving Average (SMA) Order:

- The number of lagged forecast errors to include in the seasonal model.
- This captures the influence of past seasonal prediction errors on the current seasonal period.

- These parameters are crucial in defining the structure of the SARIMA model and tailoring it to capture the underlying patterns in the data effectively.

Some parameter combinations for the Model
Model: (0, 1, 1)(0, 0, 1, 6)
Model: (0, 1, 2)(0, 0, 2, 6)
Model: (1, 1, 0)(1, 0, 0, 6)
Model: (1, 1, 1)(1, 0, 1, 6)
Model: (1, 1, 2)(1, 0, 2, 6)
Model: (2, 1, 0)(2, 0, 0, 6)
Model: (2, 1, 1)(2, 0, 1, 6)
Model: (2, 1, 2)(2, 0, 2, 6)

An empty DataFrame named ARIMA_AIC is created with two columns: param , AIC and seasonal.

'param': Will store the non-seasonal parameters (p,d,q) parameter combinations for the SARIMA model using brute force approach.

'seasonal': Will store the seasonal parameters (P,D,Q,S)parameter combinations for the SARIMA model.

'AIC': Will store the corresponding Akaike Information Criterion (AIC) values for each model. AIC is used to evaluate the goodness of fit of statistical models, with lower values indicating better fit.

	param	seasonal	AIC
53	(1, 1, 2)	(2, 0, 2, 6)	1694.342838
26	(0, 1, 2)	(2, 0, 2, 6)	1694.840012
80	(2, 1, 2)	(2, 0, 2, 6)	1695.573448
17	(0, 1, 1)	(2, 0, 2, 6)	1708.125765
44	(1, 1, 1)	(2, 0, 2, 6)	1710.045544

- With this the SARIMA model is fitted to a time series where the data is indexed by date.
- The summary provides insights into the model's fit and helps assess the chosen parameters' adequacy.
- Now,let us check how the model has performed on the test data for p=1,d=1,q=2 and P=2,D=0,Q=2,S=6.

RMSE Score of 642.8318385268345 for SARIMA

9.5.ARIMA MODEL-MANUAL

With the AR=4 and MA=3 taken from PACF and ACF plot and d=1 of differencing order lets us perform ARIMA AND SARIMA

The RMSE score was 1254.697332623067

9.6.SARIMA MODEL-MANUAL

With the AR=4 and MA=3 taken from PACF and ACF plot and P=1 ,Q=1 and D=1 by observing the PACF and ACF model.With recurring behaviour in seasonality the D=1 is taken.

The RMSE score was 379.3691256939014

10.COMPARING PERFROMANCE OF ALL MODELS:

	Test RMSE
Alpha=0.0759,Beta=0.0325,Gamma=0.4792,TripleExponentialSmoothing	366.947443
2pointTrailingMovingAverage	811.178937
Alpha=0.3,Beta=0.3,Gamma=0.9,TripleExponentialSmoothing	1142.778891
4pointTrailingMovingAverage	1184.213295
6pointTrailingMovingAverage	1337.200524
Alpha=0.0375,SimpleExponentialSmoothing	1362.428949
Alpha=0.4,SimpleExponentialSmoothing	1363.037803
SimpleAverage	1368.746717
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	1391.168123
RegressionOnTime	1392.438305
9pointTrailingMovingAverage	1422.653281

	RMSE
ARIMA(2,0,0)	1359.327261
SARIMA(1,1,2)(2,0,2,6)	642.831839
ARIMA(2,1,2)	1325.166404

When comapring all the models done the Triple Exponential Smoothing of auto with Alpha = 0.0759 ,beta= 0.0325 and gamma=0.4792 has the lowest score of RMSE wit h 366.947443 .Thus this model is selected as the best model.

11.REBUILDING THE OPTIMUM MODEL ON THE ENTIRE DATA.

After having the Triple Exponential Smoothing as the best model ,lets us see how it perform on the entire data.

The RMSE score of 366.87082512979816 for the entire data.

12.FORECASTING FOR NEXT 12 MONTHS

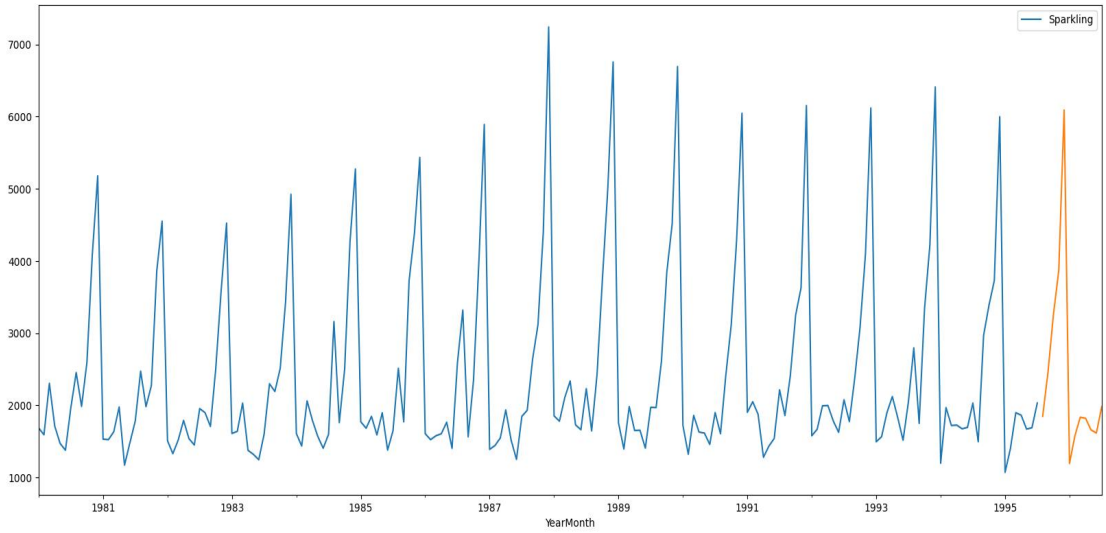
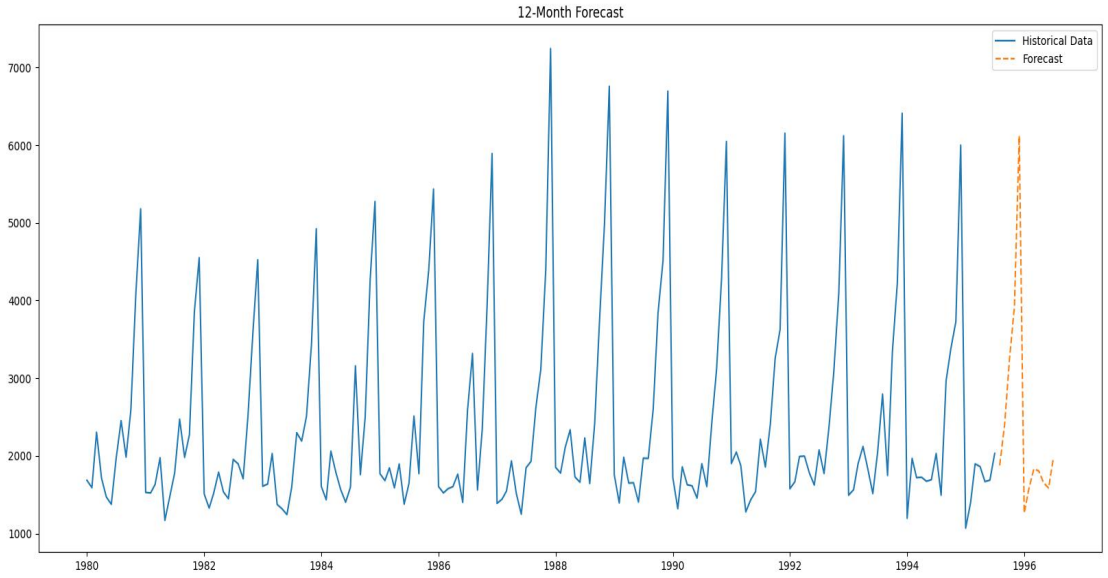


Fig 21:12 months forecast

With trend as additive and seasonal as additive and seasonal peroid to 12 months,the forecasted value as below.

1995-08-01	1878.0
1995-09-01	2405.0
1995-10-01	3242.0
1995-11-01	3923.0
1995-12-01	6119.0
1996-01-01	1263.0
1996-02-01	1592.0
1996-03-01	1832.0
1996-04-01	1807.0

1996-05-01 1652.0
1996-06-01 1587.0
1996-07-01 1977.0



13. ACTIONABLE INSIGHTS AND RECOMMENDATIONS

- Sales remained constant from 1980 to 1995, indicating that this label is well-received by consumers and may have potential for growth.
- Marketing strategies should be implemented to enhance sales, ensuring proper availability in stores and gaining a deeper understanding of the target audience and their preferences to capitalize on opportunities for increased sales.
- Introducing new bottle designs and innovative labeling could attract new consumers and refresh the brand's image for existing customers.