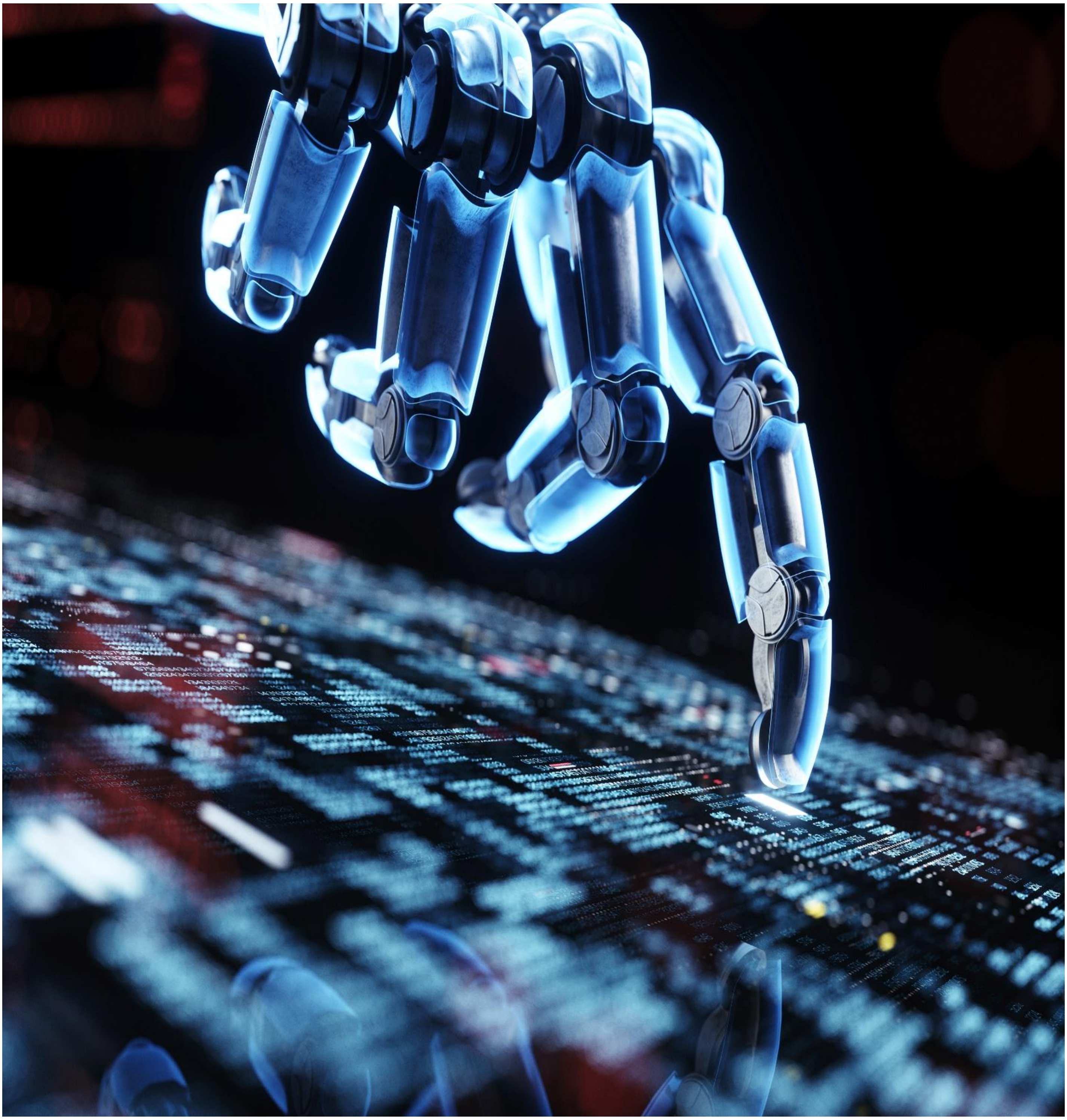
Time Series Forecasting

Student Name: Suneel Kumar Pentapalli

Date: 27/Oct/2023



Contents

List of Figures

List of Tables

Problem 1

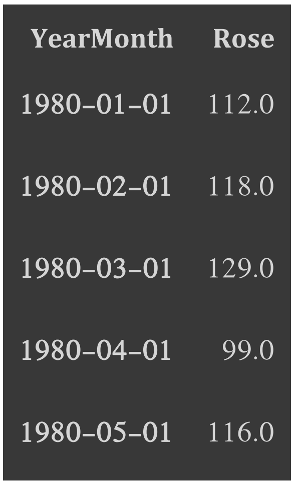
Executive Summary

In the current business report, the data of different types of wine sales in the 20th century has been analyzed. 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 analyze and forecast Wine Sales in the 20th century.

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

Rose:

Top 5 rows from Rose Sales data.



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

Table 1: Rose dataset sample

Visual representation of given time series data

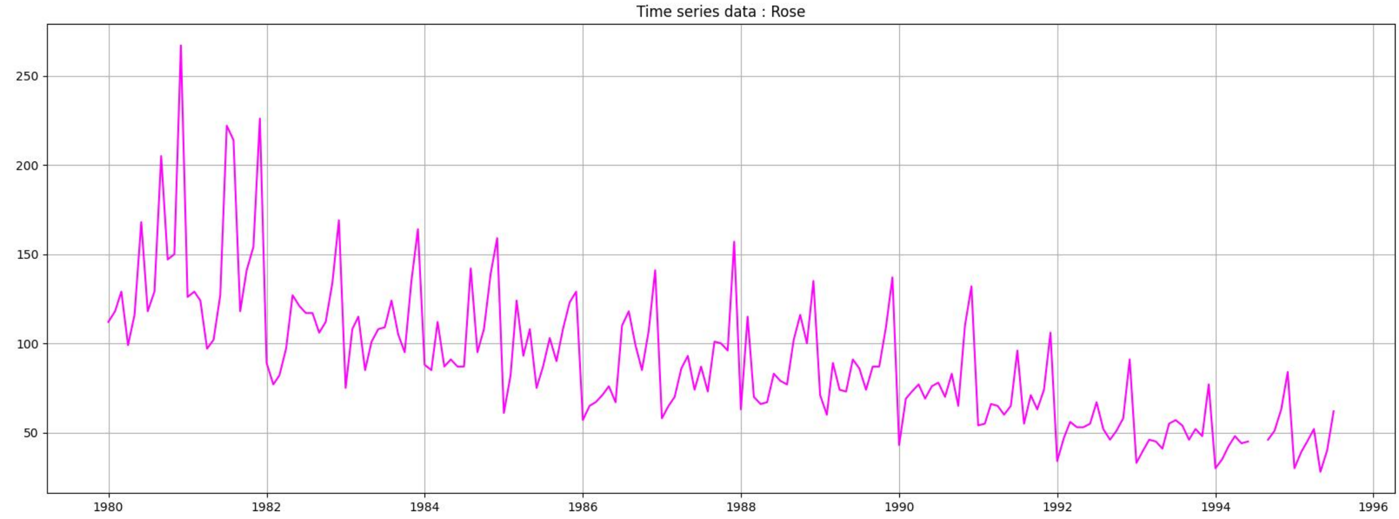
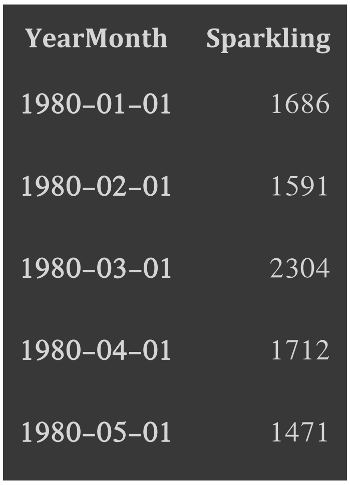


Fig.1- Time Series data: Rose

The gap in time series chart around 1994 represents missing data, which will be handled in further sections.

Sparkling:

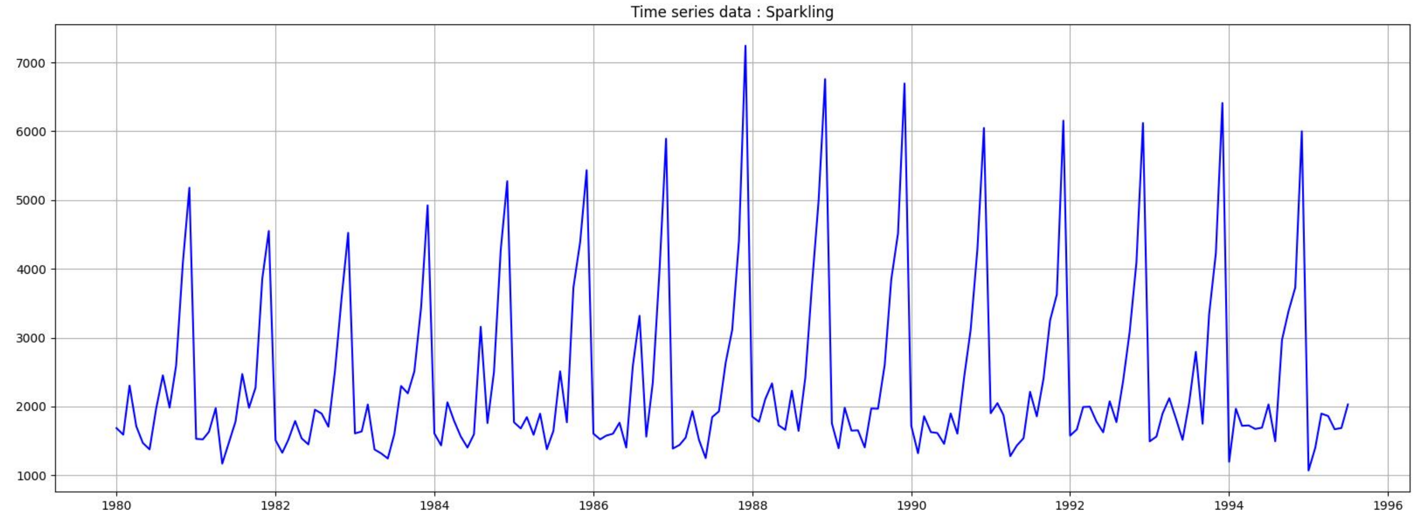
Top 5 rows from Sparkling Sales data.



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

Table 2: Sparkling dataset sample

Visual representation of given time series data



Both rose and sparkling datasets have sales data starting from January 1980 to July 1995 with monthly frequency.

Fig.2- Time Series data: Sparkling

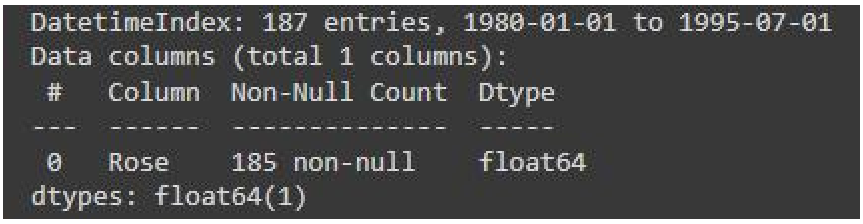
2. Perform appropriate Exploratory Data Analysis to understand the data and also perform decomposition.

Exploratory Data Analysis of Rose data:

Data Info:

Rose dataset has 187 entries as shown below, with 2 nulls as mentioned already in 1994 July and August.

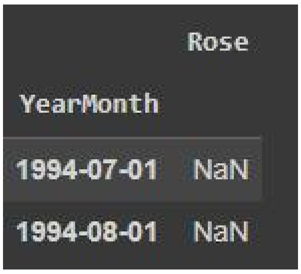
All the values in Rose column are of float datatype.



```
DatetimeIndex: 187 entries, 1980-01-01 to 1995-07-01
Data columns (total 1 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Rose    185 non-null    float64
dtypes: float64(1)
```

Check for nulls:

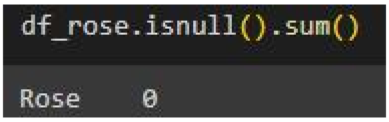
As shown below, series data is missing for time period July 1994 and August 1994. This must be handled in order to decompose our data. There are various methods to treat the null values in time series data, here we will go with quadratic interpolation to impute the null values.



	Rose
YearMonth	
1994-07-01	NaN
1994-08-01	NaN

Table 3: Rose data for null values

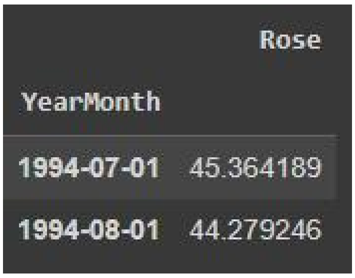
Null value check after quadratic interpolation method using python.



```
df_rose.isnull().sum()

Rose    0
```

Values after null imputation



	Rose
YearMonth	
1994-07-01	45.364189
1994-08-01	44.279246

Table 4: Rose Data Values after null imputation

Time series plot after null imputation,

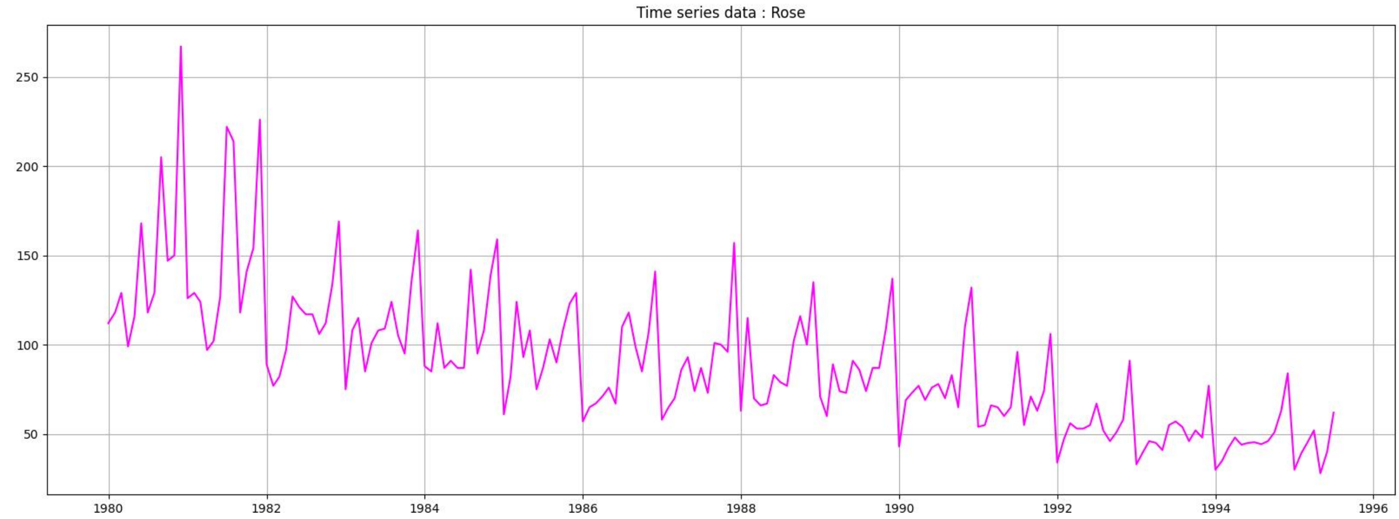
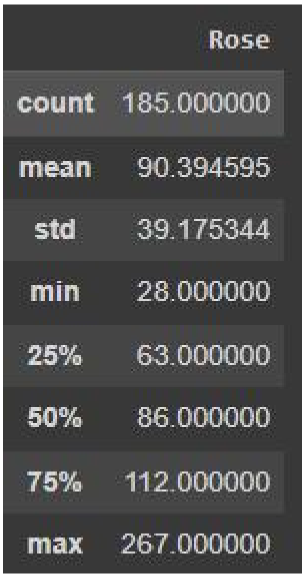


Fig.3- Time Series plot after null imputation

Thus, we have successfully removed nulls from the dataset and now we can proceed with data decomposition without any issues.

Data description:

The mean of the sales data is around 90.39 where 75% of the sales are below 112 with maximum sales being 267.



	Rose
count	185.000000
mean	90.394595
std	39.175344
min	28.000000
25%	63.000000
50%	86.000000
75%	112.000000
max	267.000000

Table 5: Rose Data description Plot

Data distribution:

As mentioned in above description, 75% of the sales are below 112 and sales between 112 and 267 constitutes to around 25%

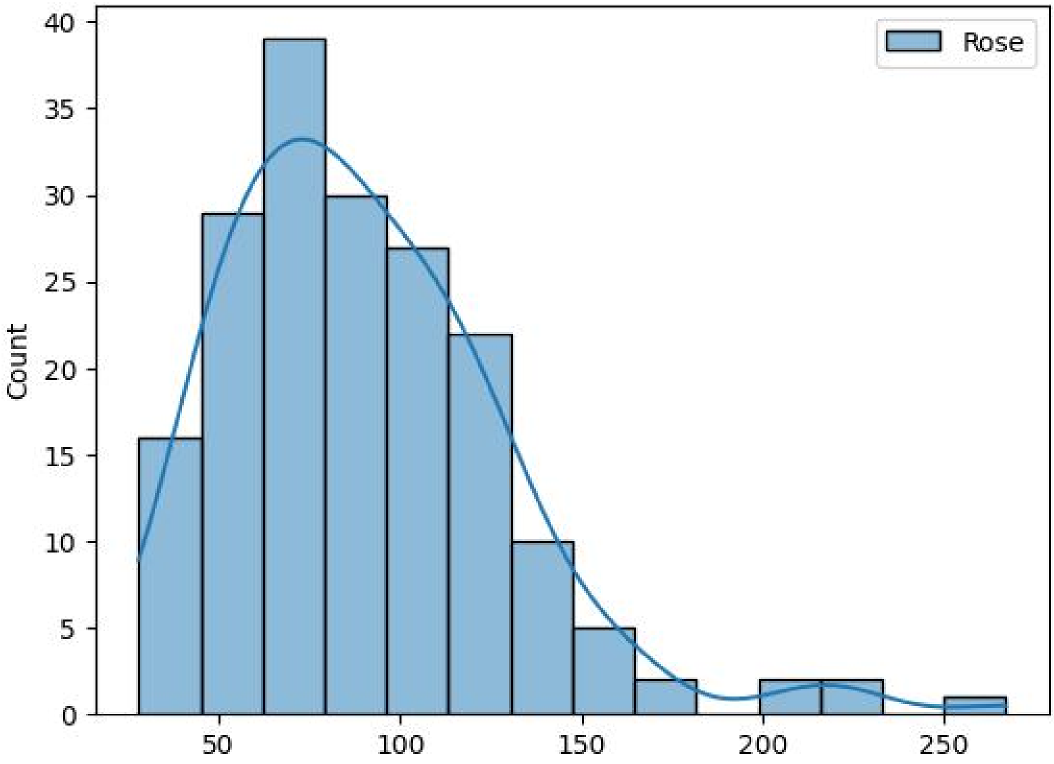


Fig.4- Data Distribution Plot

Data distribution across months:

Below chart shows the monthly sales distribution across all years.

From the data, it has been observed that the mean sales are highest in the month of December when aggregated among all the years, also the lowest sales are observed in the month of January.

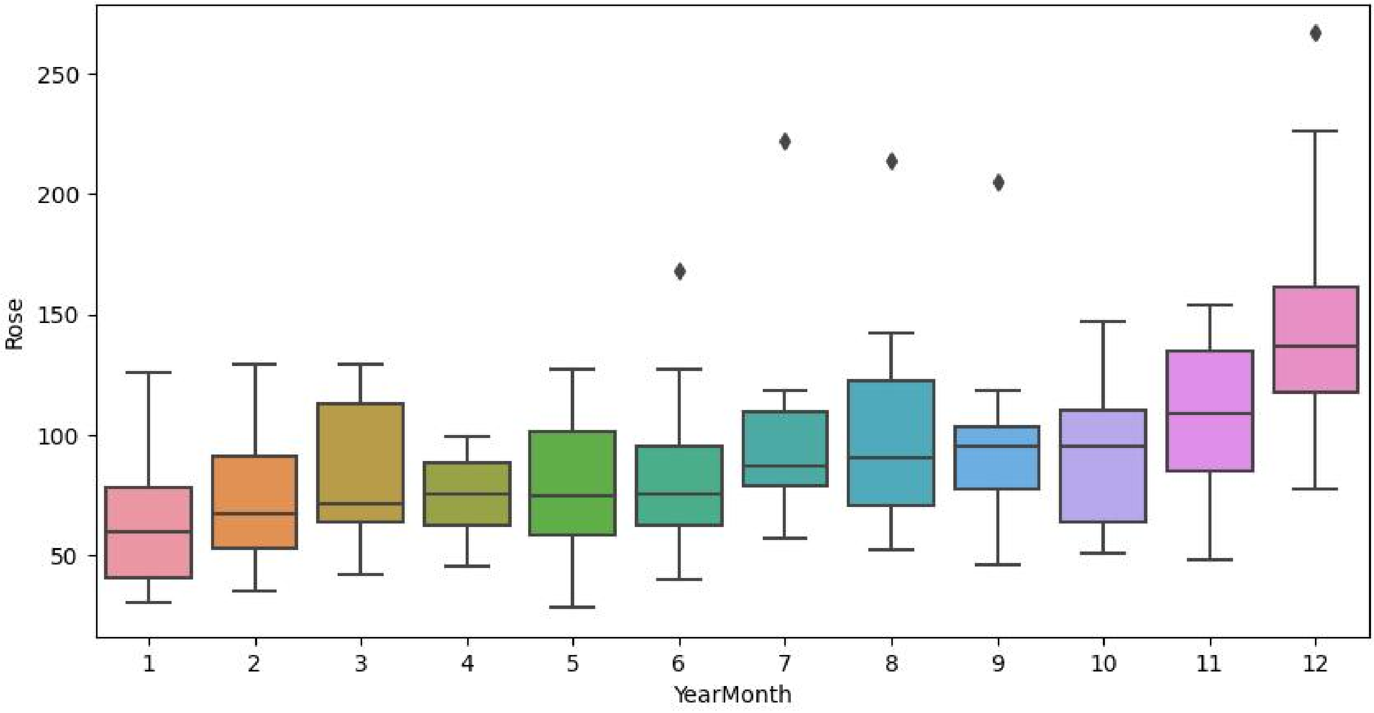


Fig.5- Data Distribution Plot across months

<u>Decomposing Rose data:</u>

Lets decompose the data to understand the trend, seasonality and error/residual components in our data.

Usually each data point is expressed as

Y(t) = Trend(t) + Seasonal(t) + Residual(t) for additive decomposition and

Y(t) = Trend(t) * Seasonal(t) * Residual(t) for multiplicative decomposition.

As the magnitude of peaks in our time series plot reducing at a non constant rate, we will go with multiplicative seasonality. Visual representation of the same is as follows.

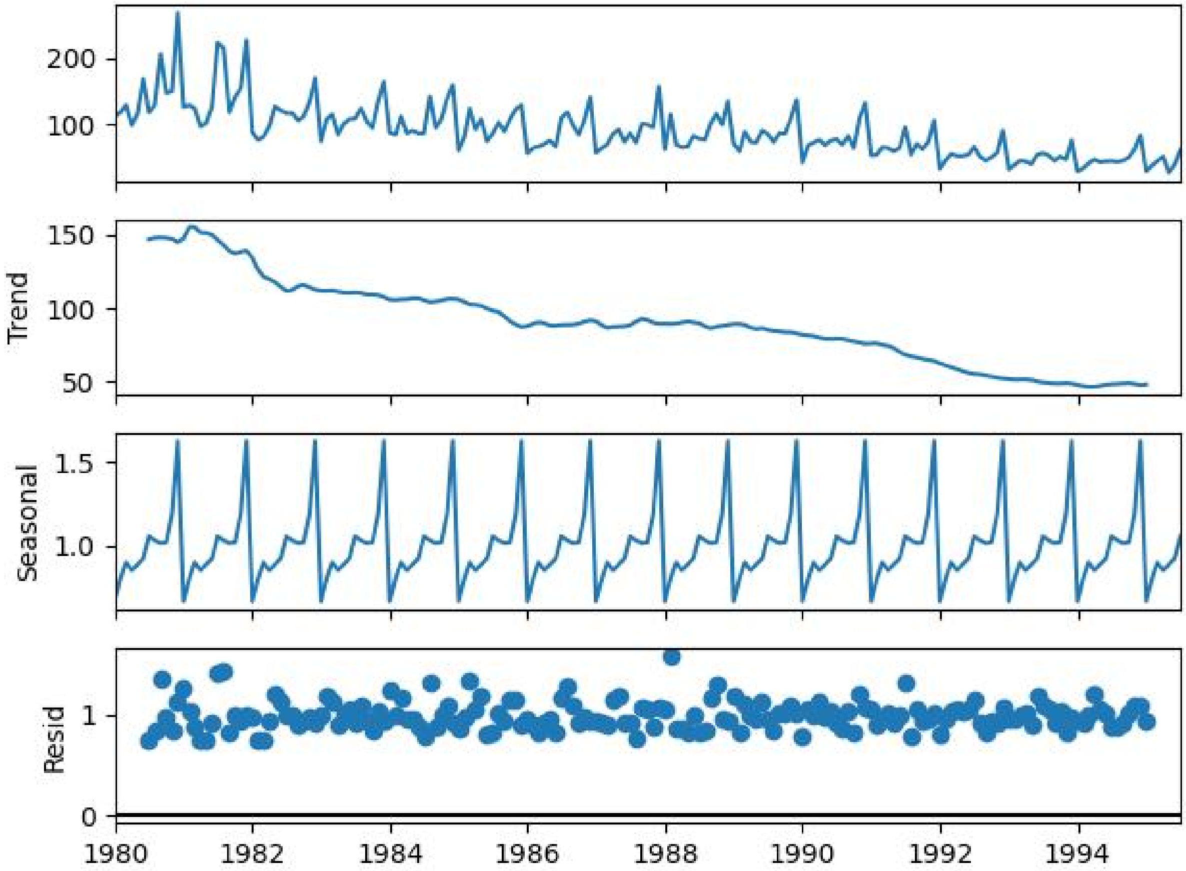


Fig.6- Decomposing Rose data Plot

From the above chart, it is evident that

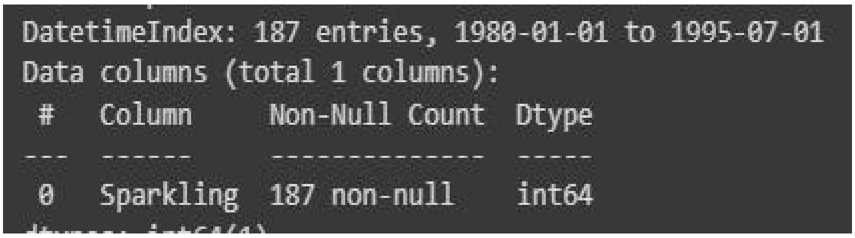
- there is the downward trend over the years,
- almost a constant seasonality is observed across all the years, which indicates there are months where sales are high and month with low sales with a constant repletion of this pattern over years.

<u>Exploratory Data Analysis of Sparkling dataset:</u>

Data Info:

Sparkling sales dataset has 187 entries as shown below, with 0 nulls

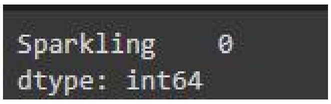
All the values in Sparkling column are of int datatype.



```
DatetimeIndex: 187 entries, 1980-01-01 to 1995-07-01
Data columns (total 1 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Sparkling  187 non-null    int64
```

Check for nulls:

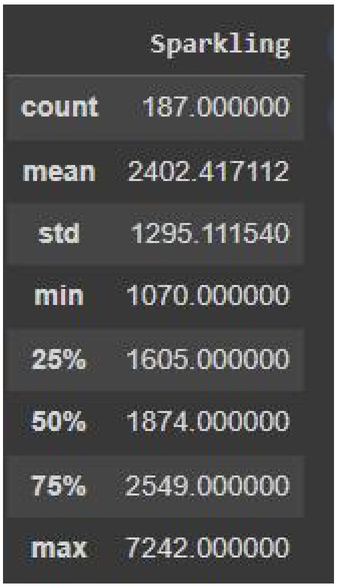
Given data has no nulls for the sparkling data which is evident from the below,



```
Sparkling    0
dtype: int64
```

Data description:

The mean of the sparkling sales data is around 2402.41 where 75% of the sales are below 2549 with maximum sales being 7242.



	Sparkling
count	187.000000
mean	2402.417112
std	1295.111540
min	1070.000000
25%	1605.000000
50%	1874.000000
75%	2549.000000
max	7242.000000

Table 6: Sparkling Data Description

Data distribution:

As mentioned in above description, 75% of the sales are below 2549 and sales between 2549 and 7242 constitutes to around 25%

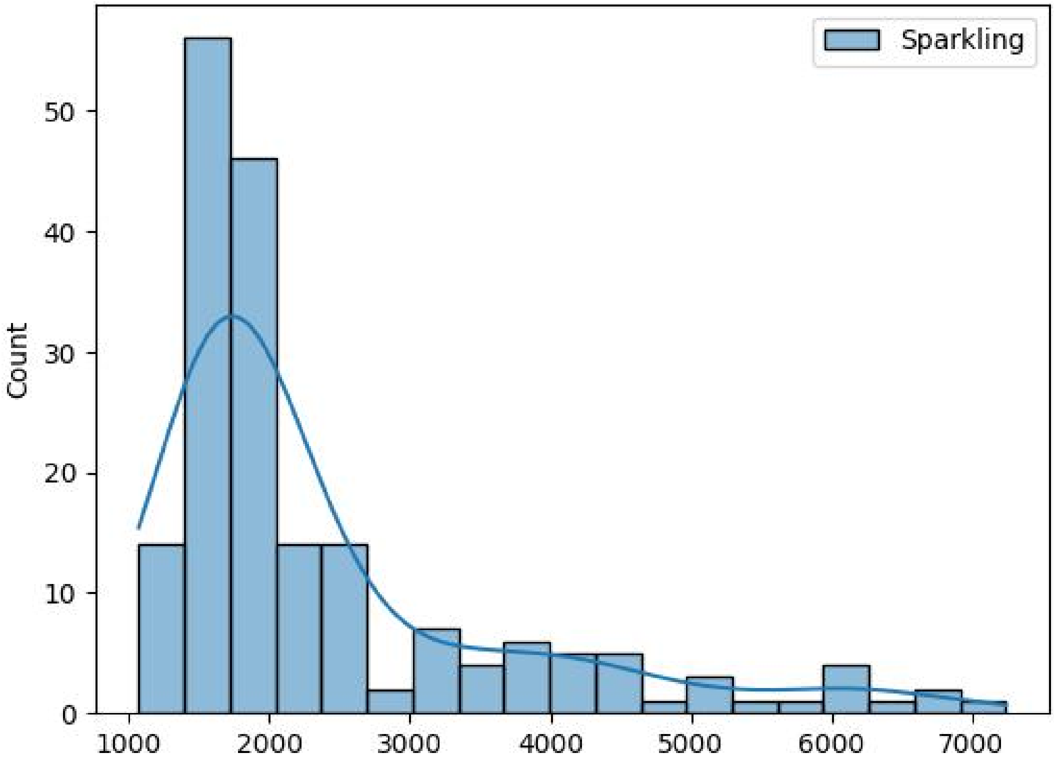


Fig.7- Sparkling Data distribution Plot

Data distribution across months:

Below chart shows the monthly sales distribution across all years.

From the data, it has been observed that the mean sales are highest in the month of December when aggregated among all the years, also the lowest sales are observed in the month of June.

Hence we can conclude that wine sales are hitting peak for both Rose and Sparkling types in the month of December.

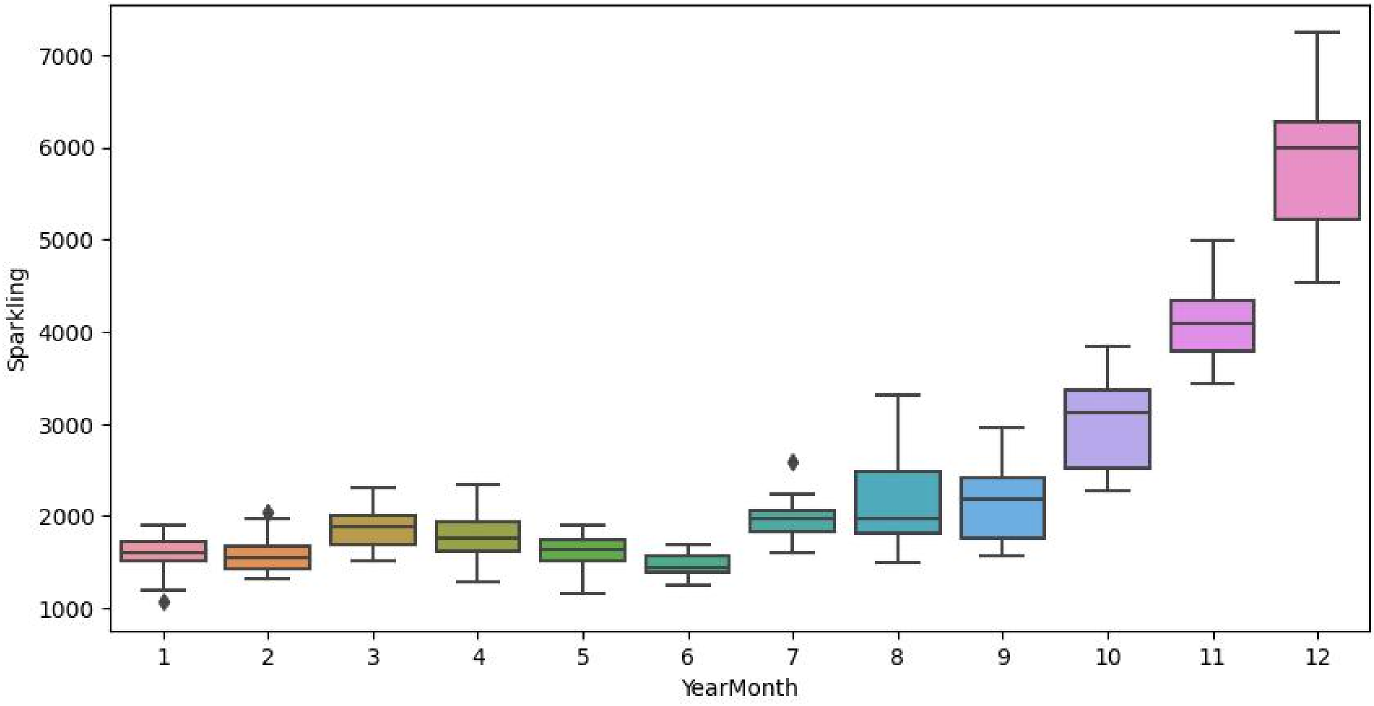


Fig.8- Sparkling data distribution plot across the year

Decomposing Rose data:

Lets decompose the data to understand the trend, seasonality and error/residual components in our data.

Usually each data point is expressed as

Y(t) = Trend(t) + Seasonal(t) + Residual(t) for additive decomposition and

Y(t) = Trend(t) * Seasonal(t) * Residual(t) for multiplicative decomposition.

As the magnitude of peaks in our time series plot are not varying at a constant rate, we will go with multiplicative seasonality. Visual representation of the same is as follows.

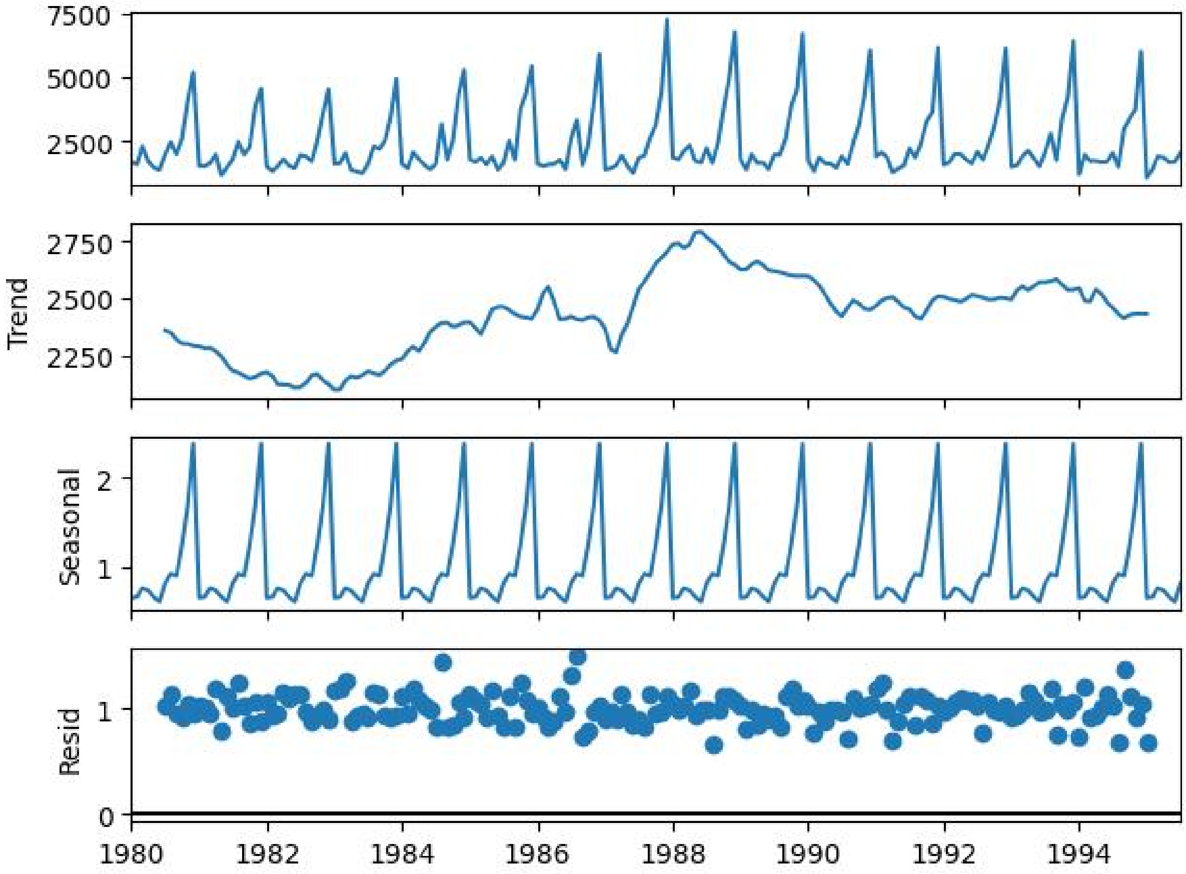


Fig.9- Plot for decomposing Rose data

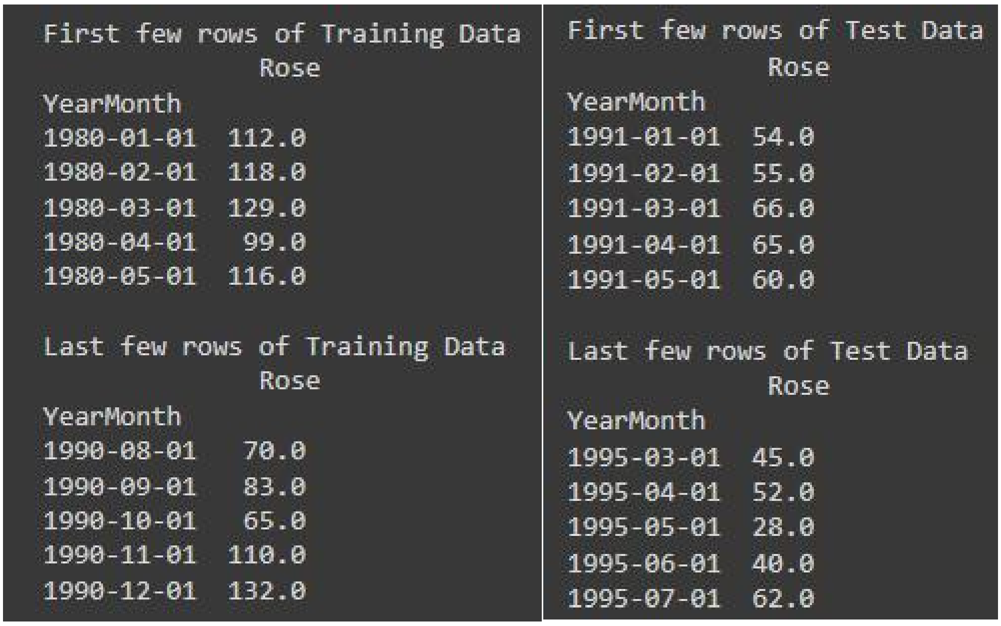
From the above chart, it is evident that

- sales trend is not linear, infact sales have seen a rise around the year 1989 and from there on the trend has been shifted downwards.
- almost a constant seasonality is observed across all the years, which indicates there are months where sales are high and month with low sales with a constant repletion of this pattern over years.

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

Rose:

Data has been split into training and test data set with test data starting from Jan 1991.



```
First few rows of Training Data
                    Rose
YearMonth
1980-01-01  112.0
1980-02-01  118.0
1980-03-01  129.0
1980-04-01   99.0
1980-05-01  116.0

Last few rows of Training Data
                    Rose
YearMonth
1990-08-01   70.0
1990-09-01   83.0
1990-10-01   65.0
1990-11-01  110.0
1990-12-01  132.0
```

```
First few rows of Test Data
                    Rose
YearMonth
1991-01-01  54.0
1991-02-01  55.0
1991-03-01  66.0
1991-04-01  65.0
1991-05-01  60.0

Last few rows of Test Data
                    Rose
YearMonth
1995-03-01  45.0
1995-04-01  52.0
1995-05-01  28.0
1995-06-01  40.0
1995-07-01  62.0
```

Table 7: Training and test data set Plot for Rose

Plot for Rose sales data by differentiating Test and train data:

In the below chart, data in blue indicates training data and the orange line indicates test data.

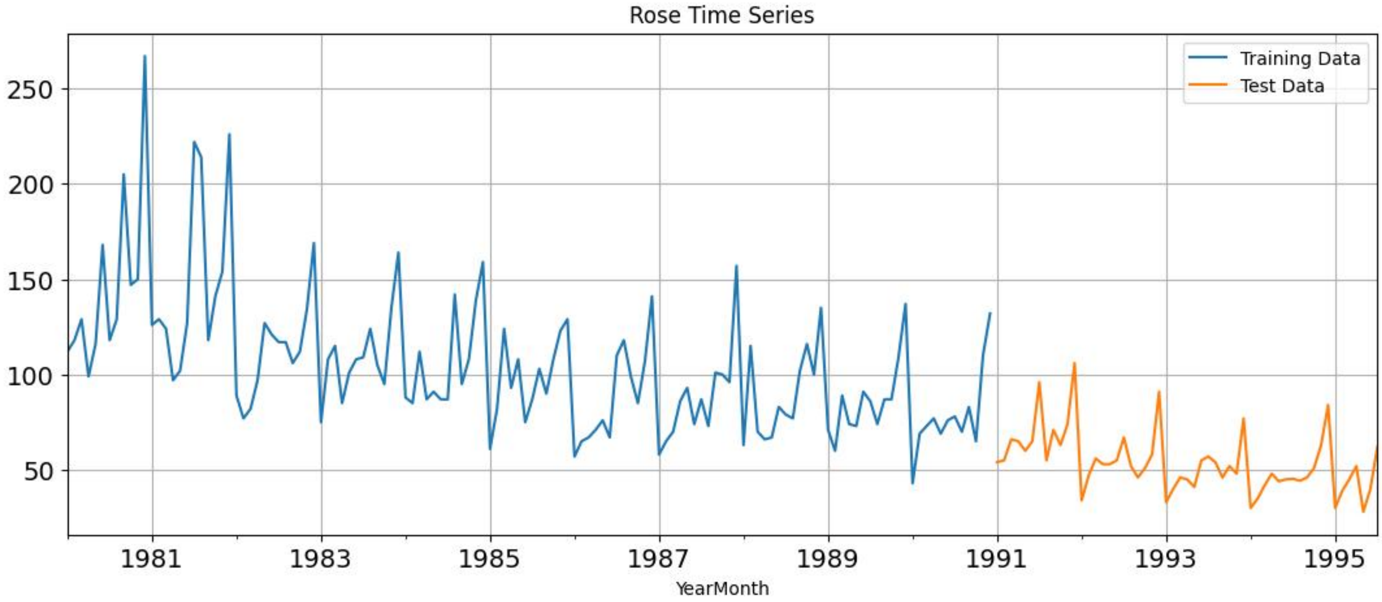
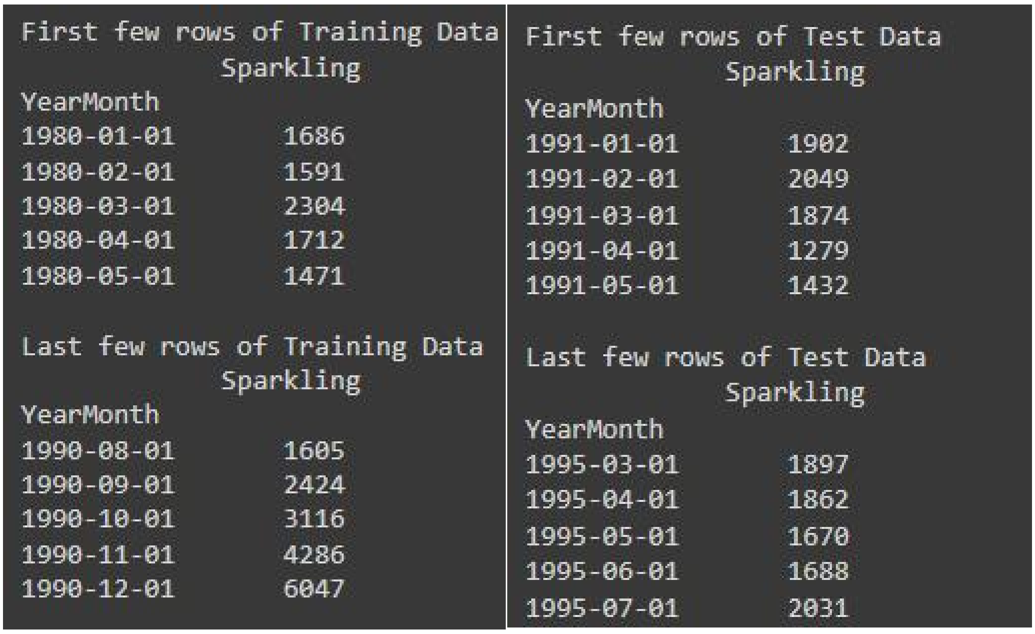


Fig.10- Plot for Rose sales data by differentiating Test and train data

<u>Sparkling:</u>

Data has been split into training and test data set with test data starting from Jan 1991.



```
First few rows of Training Data
                 Sparkling
YearMonth
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
                 Sparkling
YearMonth
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
                 Sparkling
YearMonth
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
                 Sparkling
YearMonth
1995-03-01           1897
1995-04-01           1862
1995-05-01           1670
1995-06-01           1688
1995-07-01           2031
```

Table 8: Training and test data set plot for Sparkling

Plot for Sparkling sales data by differentiating Test and train data:

In the below chart, data in blue indicates training data and the orange line indicates test data.

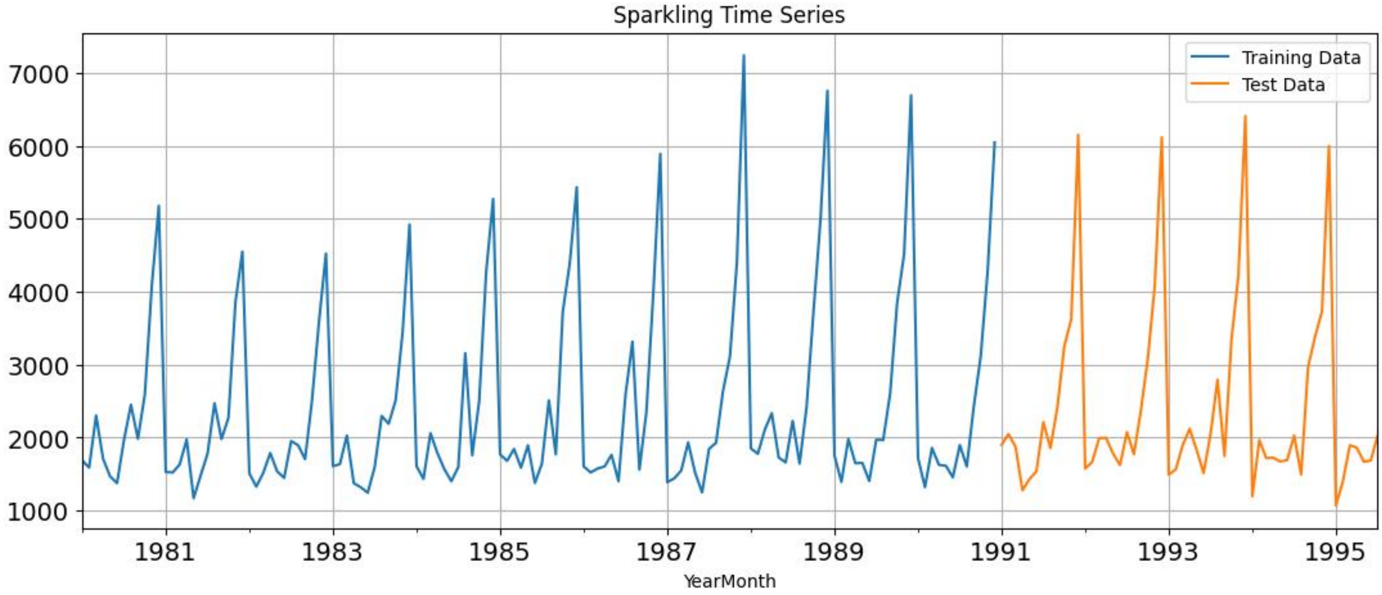


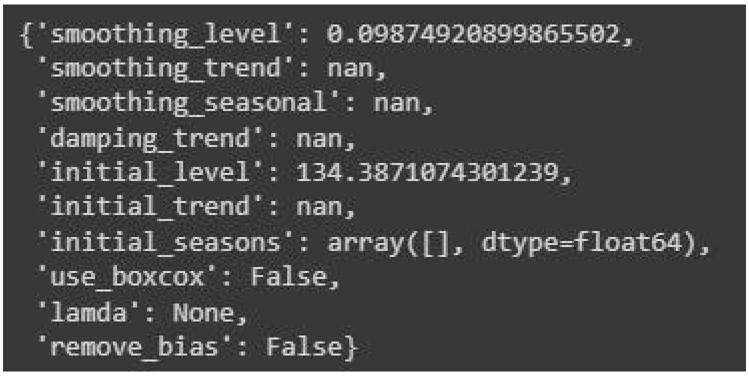
Fig.11- Plot for Sparkling Sales Data by differentiating Test and train data

4. Build all the 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 and simple average models. should also be built on the training data and check the performance on the test data using RMSE.

Models on Rose Data:

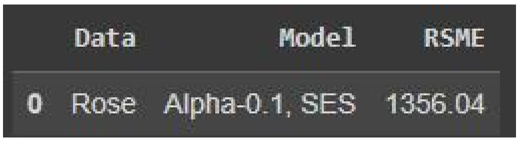
Simple Exponential Smoothing:A simple exponential smoothing is one of the simplest ways to forecast a time series. The basic idea of this model is to assume that the future will be more or less the same as the (recent) past. Thus, the only pattern that this model will learn from demand history is its level.

After fitting the Simple Exponential model from python's statsmodel library, below parameters are picked as the best. As this is simple exponential smoothing, learning will be made based on smoothed level(alpha) only as shown below,



```
{'smoothing_level': 0.09874920899865502,
 'smoothing_trend': nan,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 134.3871074301239,
 'initial_trend': nan,
 'initial_seasons': array([], dtype=float64),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Now, we have passed the training data to fit the model and performed prediction on test length. RSME(Root mean squared error) calculated between actual test values(i.e., from 1991,Jan) and the predicted values for the same time range is as follows,



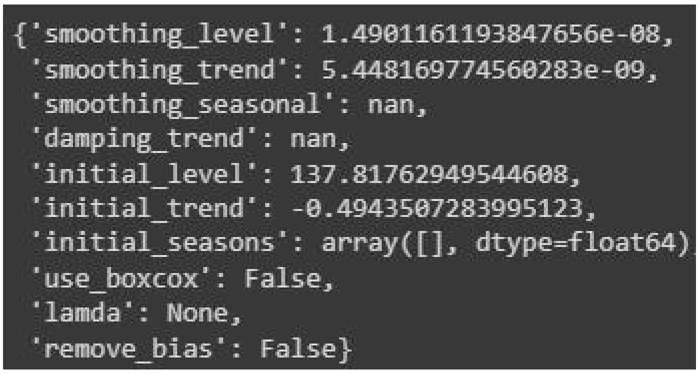
	Data	Model	RSME
0	Rose	Alpha-0.1, SES	1356.04

Table 9: RSME Score on Rose Data using SES Model

Double Exponential Smoothing:

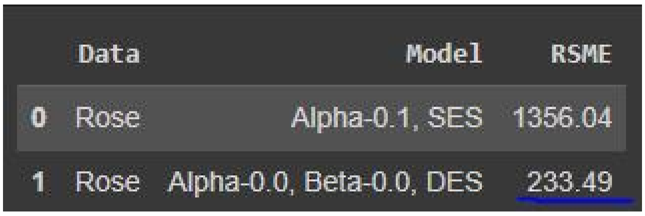
Double exponential similar to simple exponential smoothing but in addition to level(alpha), this model also considers trend(beta) for learning and uses it for forecasting.

The best params picked by Holt(Double Exponential Smoothing) model are as follows,



```
{'smoothing_level': 1.4901161193847656e-08,
 'smoothing_trend': 5.448169774560283e-09,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 137.81762949544608,
 'initial_trend': -0.4943507283995123,
 'initial_seasons': array([], dtype=float64)
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Now, we have passed the training data to fit the model and performed prediction on test length. RSME(Root mean squared error) calculated between actual test values(i.e., from 1991,Jan) and the predicted values for the same time range is as follows,

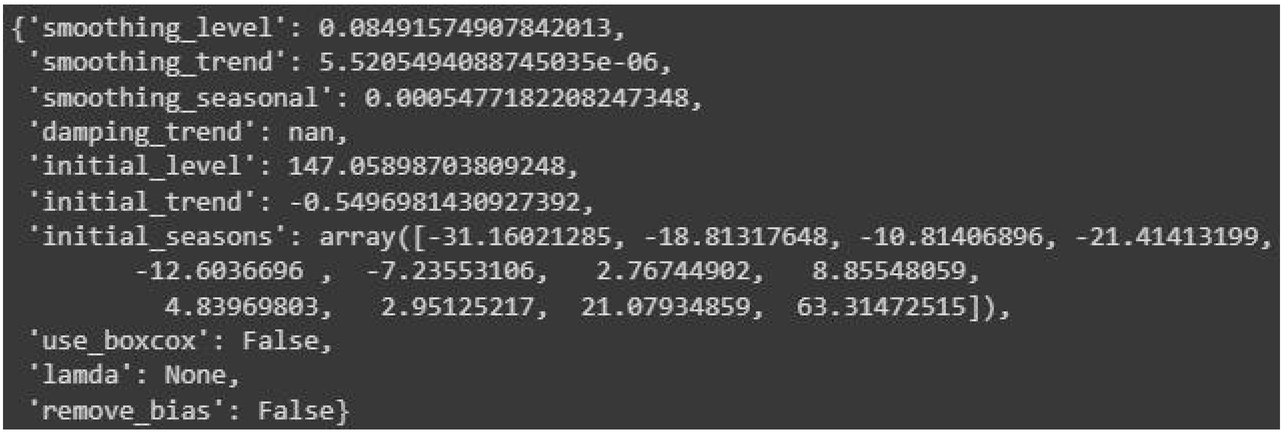


	Data	Model	RSME
0	Rose	Alpha-0.1, SES	1356.04
1	Rose	Alpha-0.0, Beta-0.0, DES	233.49

Table 10: RSME Score on Rose Data using DES Model

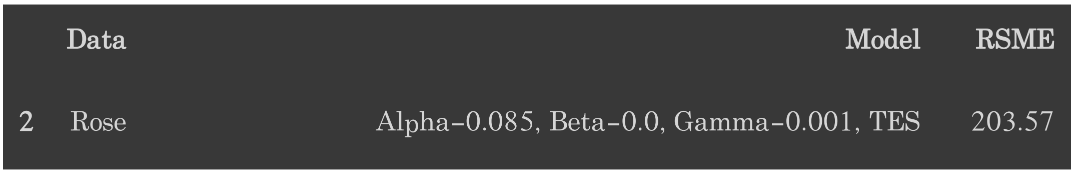
Triple Exponential Smoothing:

Triple Exponential Smoothing model learns from the data by going one step further as this model includes seasonality(gamma) also for learning and predictions as shown below,



```
{'smoothing_level': 0.08491574907842013,
 'smoothing_trend': 5.5205494088745035e-06,
 'smoothing_seasonal': 0.0005477182208247348,
 'damping_trend': nan,
 'initial_level': 147.05898703809248,
 'initial_trend': -0.5496981430927392,
 'initial_seasons': array([-31.16021285, -18.81317648, -10.81406896, -21.41413199,
        -12.6036696 ,  -7.23553106,   2.76744902,   8.85548059,
          4.83969803,   2.95125217,  21.07934859,  63.31472515]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

RSME Score on test data using TES model is as follows,



	Data	Model	RSME
2	Rose	Alpha-0.085, Beta-0.0, Gamma-0.001, TES	203.57

Table 11: RSME Score on Rose Data using TES Model

Plot for predicted and actual values for each model:

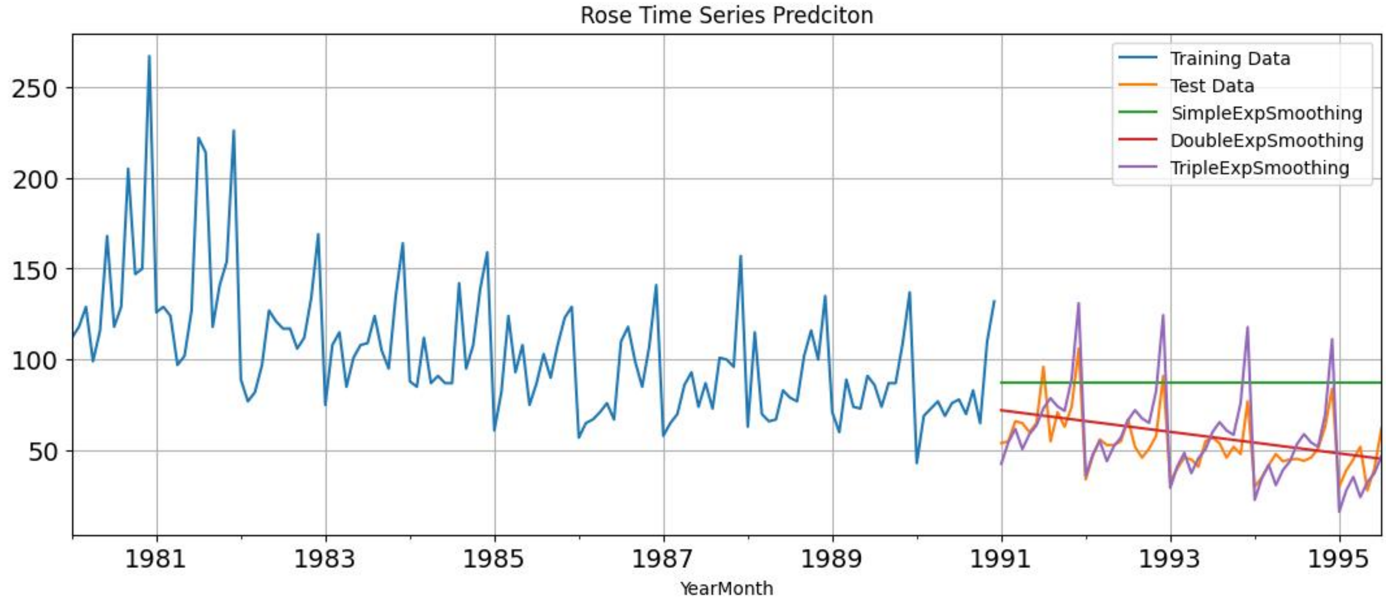


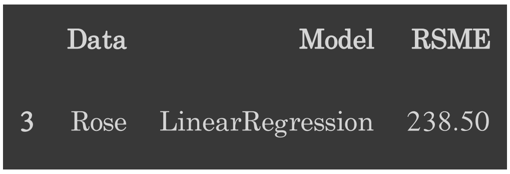
Fig.12: Plot for Rose Time Series Prediction using SES,DES and TES Model

Of all the above smoothing models, Triple exponential smoothing seems to have performed well with least RSME score of 203.57 which is obvious because our data has seasonality in it.

Other models:

Linear Regression:

A Linear Regression model forecasts a single values for entire test series by fitting a line equation model on training data, as it doesn't consider any seasonality and trend, the predicted values will be less accurate.

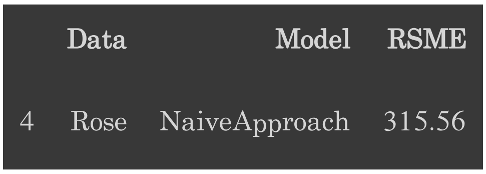


	Data	Model	RSME
3	Rose	LinearRegression	238.50

Table 12: RSME Score on Rose Data using Linear Regression Model

Naïve Approach:

For this particular naive model, we say that 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.

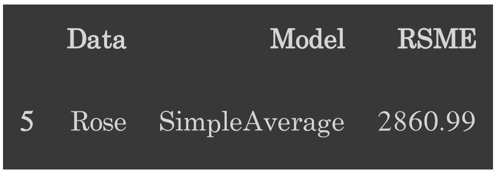


	Data	Model	RSME
4	Rose	NaiveApproach	315.56

Table 13: RSME Score on Rose Data using NaiveApproach Model

Simple Average:

For this particular simple average method, we will forecast by using the average of the training values.



	Data	Model	RSME
5	Rose	SimpleAverage	2860.99

Table 14: RSME Score on Rose Data using SimpleAverage Model

Time series plots for all above three models on test data is as follows,

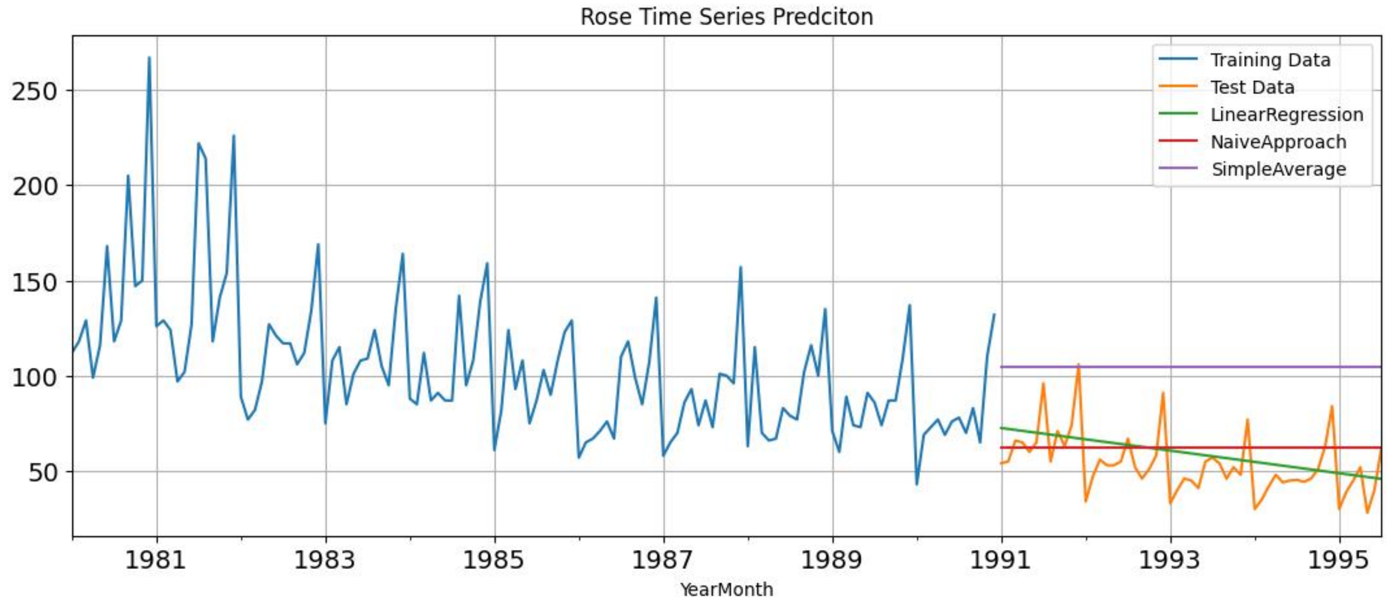


Fig.13: Plot for Rose Time Series Prediction using LR,NA and SA Model

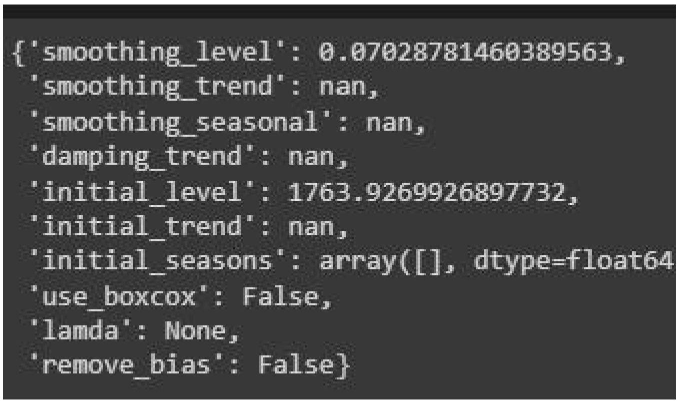
Of all the models mentioned above, Triple exponential smoothing model is performing the best on rose sale data.

Models on Sparkling Data:

Simple Exponential Smoothing:

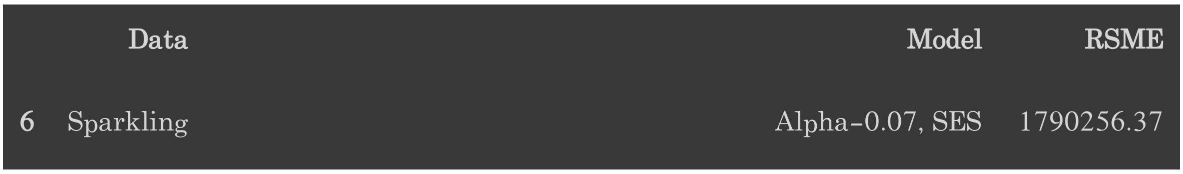
A simple exponential smoothing is one of the simplest ways to forecast a time series. The basic idea of this model is to assume that the future will be more or less the same as the (recent) past. Thus, the only pattern that this model will learn from demand history is its level.

After fitting the Simple Exponential model from python's statsmodel library, below parameters are picked as the best. As this is simple exponential smoothing, learning will be made based on smoothed level(alpha) only as shown below,



```
{'smoothing_level': 0.07028781460389563,
 'smoothing_trend': nan,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 1763.9269926897732,
 'initial_trend': nan,
 'initial_seasons': array([], dtype=float64
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Now, we have passed the training data to fit the model and performed prediction on test length. RSME(Root mean squared error) calculated between actual test values(i.e., from 1991,Jan) and the predicted values for the same time range is as follows,



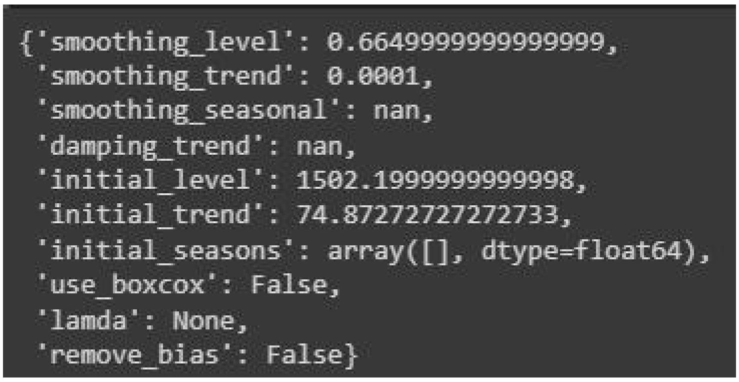
	Data	Model	RSME
6	Sparkling	Alpha-0.07, SES	1790256.37

Table 15: RSME Score on Sparkling Data using SES Model

Double Exponential Smoothing:

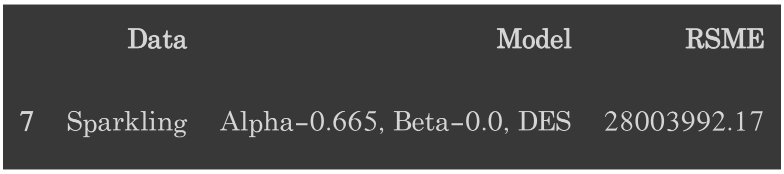
Double exponential similar to simple exponential smoothing but in addition to level(alpha), this model also considers trend(beta) for learning and uses it for forecasting.

The best params picked by Holt(Double Exponential Smoothing) model are as follows,



```
{'smoothing_level': 0.6649999999999999,
 'smoothing_trend': 0.0001,
 'smoothing_seasonal': nan,
 'damping_trend': nan,
 'initial_level': 1502.1999999999998,
 'initial_trend': 74.87272727272733,
 'initial_seasons': array([], dtype=float64),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

Now, we have passed the training data to fit the model and performed prediction on test length. RSME(Root mean squared error) calculated between actual test values(i.e., from 1991,Jan) and the predicted values for the same time range is as follows,

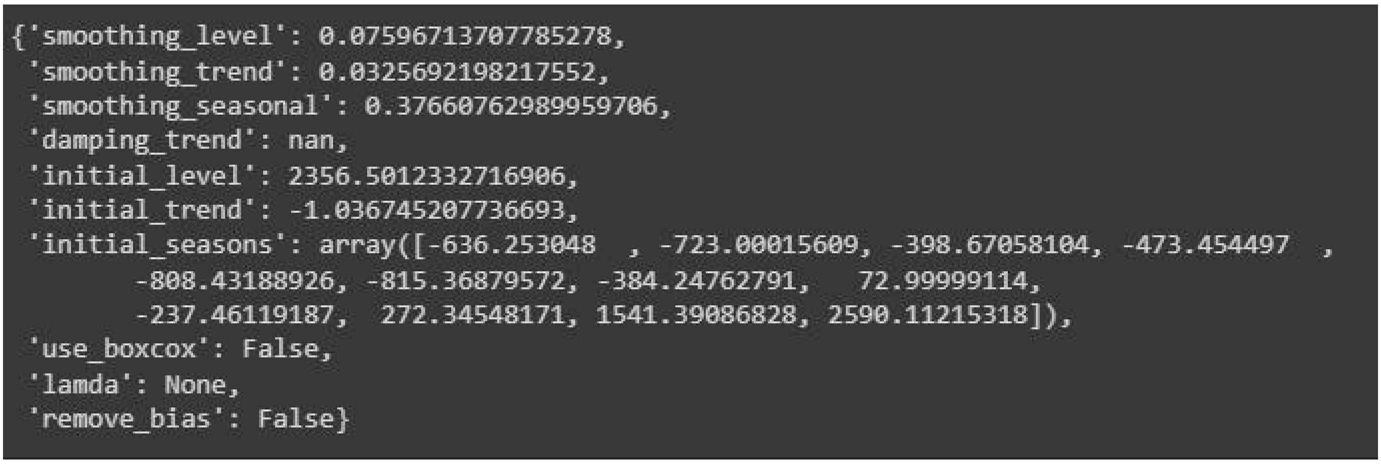


	Data	Model	RSME
7	Sparkling	Alpha-0.665, Beta-0.0, DES	28003992.17

Table 16: RSME Score on Sparkling Data using DES Model

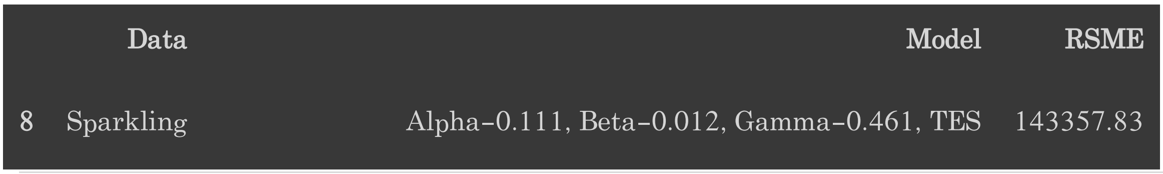
Triple Exponential Smoothing:

Triple Exponential Smoothing model learns from the data by going one step further as this model includes seasonality(gamma) also for learning and predictions as shown below,



```
{'smoothing_level': 0.07596713707785278,
 'smoothing_trend': 0.0325692198217552,
 'smoothing_seasonal': 0.37660762989959706,
 'damping_trend': nan,
 'initial_level': 2356.5012332716906,
 'initial_trend': -1.036745207736693,
 'initial_seasons': array([-636.253048  , -723.00015609, -398.67058104, -473.454497  ,
        -808.43188926, -815.36879572, -384.24762791,   72.99999114,
        -237.46119187,  272.34548171, 1541.39086828, 2590.11215318]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

RSME Score on test data using TES model is as follows,



	Data	Model	RSME
8	Sparkling	Alpha-0.111, Beta-0.012, Gamma-0.461, TES	143357.83

Table 17: RSME Score on Sparkling Data using TES Model

Plot for predicted and actual values for each model:

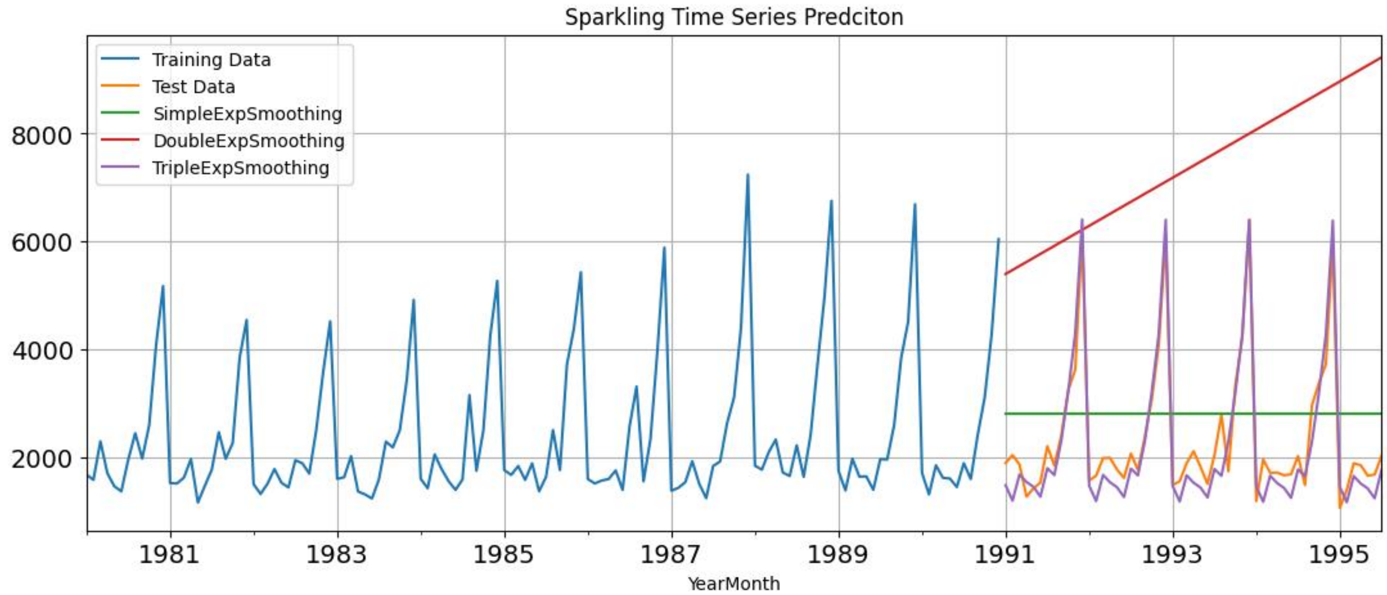


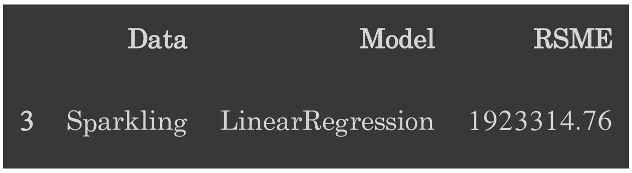
Fig.14: Plot for Sparkling Time Series Prediction using SES,DES and TES Model

Of all the above smoothing models, Triple exponential smoothing seems to have performed well with least RSME score of 143357.83 which is obvious because our data has seasonality in it.

Other models:

Linear Regression:

A Linear Regression model forecasts a single values for entire test series by fitting a line equation model on training data, as it doesn't consider any seasonality and trend, the predicted values will be less accurate.

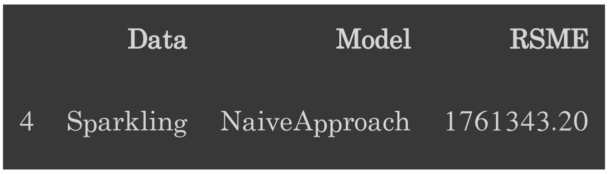


	Data	Model	RSME
3	Sparkling	LinearRegression	1923314.76

Table 18: RSME Score on Sparkling Data using LinearRegression Model

Naïve Approach:

For this particular naive model, we say that 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.

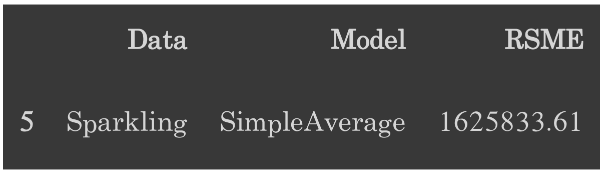


	Data	Model	RSME
4	Sparkling	NaiveApproach	1761343.20

Table 19: RSME Score on Sparkling Data using NaiveApproach Model

Simple Average:

For this particular simple average method, we will forecast by using the average of the training values.



	Data	Model	RSME
5	Sparkling	SimpleAverage	1625833.61

Table 20: RSME Score on Sparkling Data using SimpleAverage Model

Time series plots for all above three models on test data is as follows,

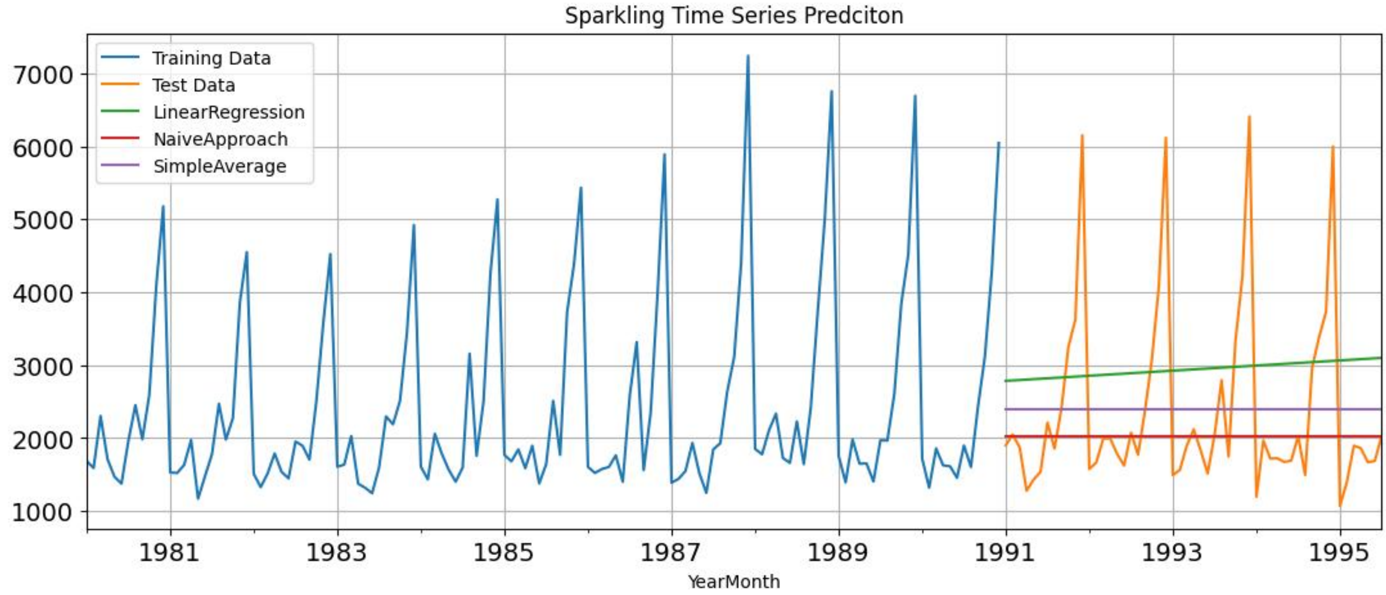


Fig.15: Plot for Sparkling Time Series Prediction using LR,NA and SA Model

By looking at prediction capabilities of all the models on sparkling dataset, TES model seems to be the best.

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. Note: Stationarity should be checked at alpha = 0.05.

A time series has stationarity when the observations are not dependent on the time. Statistical properties of these time series will not change with time thus they will have constant mean, variance, and covariance.

The time series which have trends or with seasonality, are not stationary. Because trends will have a change in the movement of data concerning time which will cause the change in mean over time. Whereas seasonality occurs when the pattern in time series shows a variation for a regular time interval which will cause the variance to change over time.

Stationarity can be checked using Augmented Dickey-Fuller (ADF) test.

Dickey-Fuller Test - Dicky Fuller Test on the timeseries is run to check for stationarity of data.

Null Hypothesis H0 : Time Series is non-stationary.

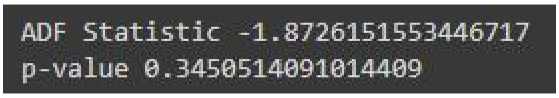
Alternate Hypothesis Ha : Time Series is stationary.

So Ideally if p-value < 0.05(given significance level) then null hypothesis: TS is non-stationary is rejected else the TS is non-stationary is failed to be rejected.

As forecasting models like ARIMA assumes the data to be stationary, it is always advisable to make the data stationary before fitting the model with data.

Rose wine sales Data:

Upon performing ADF test on entire Rose sales data, we got p-value as follows,



```
ADF Statistic -1.8726151553446717
p-value 0.3450514091014409
```

the p-value 0.34 is very large, and not smaller than 0.05 and therefore we donot have enough evidence to reject null hypothesis, thus our data is not a stationary series. But fortunately we can convert non-stationary data to stationary data by detrending the timeseries data using any of the below transformation techniques,

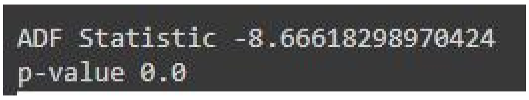
Log transforming of the data

Taking the square root of the data

Taking the cube root

Proportional change

Lets use log transform on entire rose data now and lets check the p-value again using ADF test,



```
ADF Statistic -8.66618298970424
p-value 0.0
```

After applying log transform, p-value has become less than 0.05 hence making our data stationary.

Visualization of transformed data is as follows,

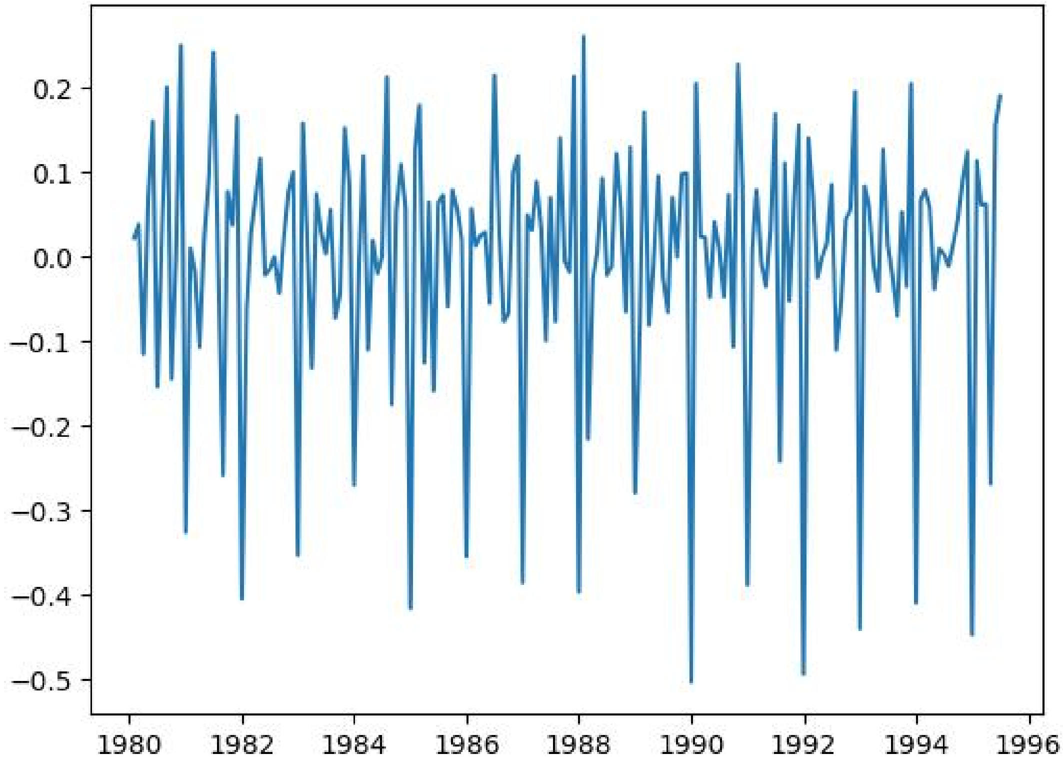
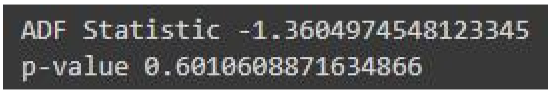


Fig.16- Rose Wine Sales Data

Sparkling Wine Sales Data:

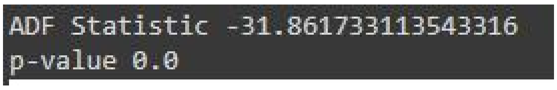
Upon performing ADF test on entire Rose sales data, we got p-value as follows,



```
ADF Statistic -1.3604974548123345
p-value 0.6010608871634866
```

the p-value 0.6 is very large, and not smaller than 0.05 and therefore we donot have enough evidence to reject null hypothesis, thus our data is not a stationary series.

Again, lets use log transform on entire rose data now and lets check the p-value again using ADF test,



```
ADF Statistic -31.861733113543316
p-value 0.0
```

After applying log transform, p-value has become less than 0.05 hence making our data stationary.

Visualization of transformed data is as follows,

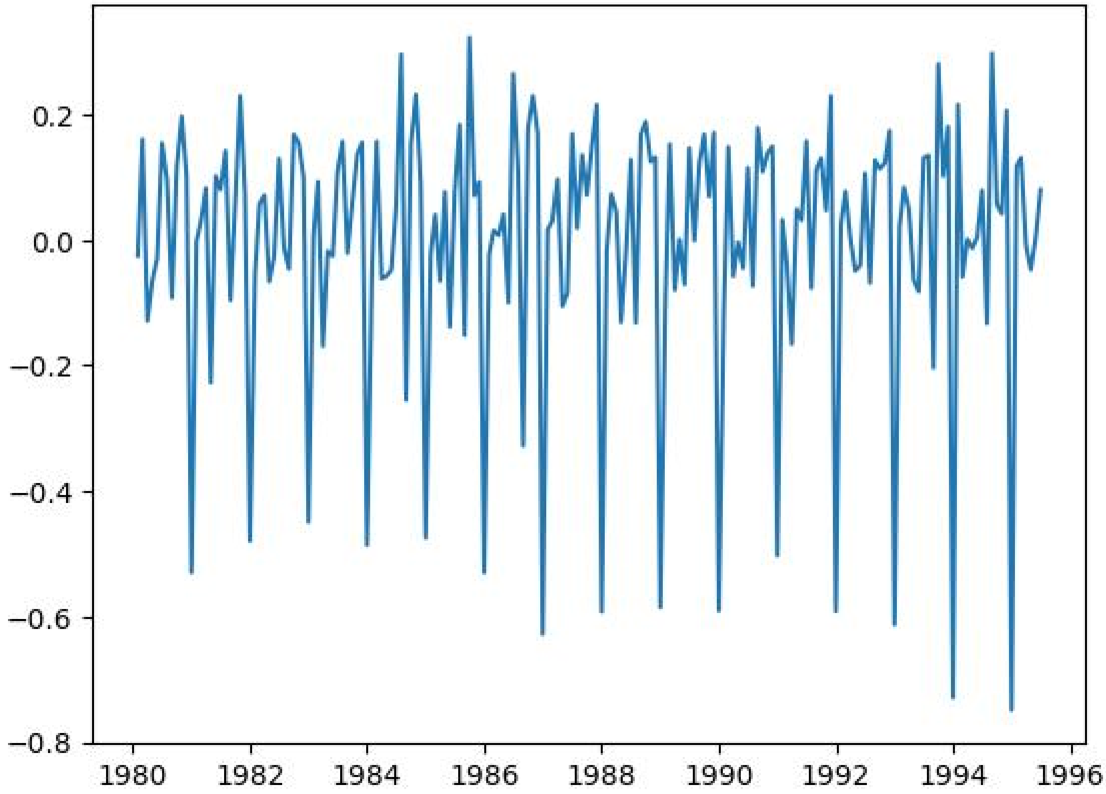


Fig.17- Sparkling Wine Sales data

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.

ARIMA, short for 'AutoRegressive Integrated Moving Average', is a forecasting algorithm based on the idea that the information in the past values of the time series can alone be used to predict the future values.

ARIMA model takes two parameters as input before fitting the data, one is data and the other is order parameter.

Order parameter contains below,

p is the order of the AR term(how many past values to be considered)

q is the order of the MA term(how many past errors to be considered)

d is the number of differencing required to make the time series stationary

for the current problem statement, we tool values of p, d and q ranging from 0 to 3

Tools to check model performance:

The Akaike information criterion (AIC) is a mathematical method for evaluating how well a model fits the data it was generated from. In statistics, AIC is used to compare different possible models and determine which one is the best fit for the data. AIC is calculated from:

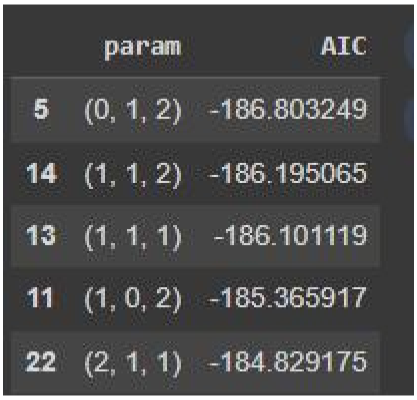
- the number of independent variables used to build the model.
- the maximum likelihood estimate of the model (how well the model reproduces the data).

For a model to be called a good fit, the AIC score should be as low as possible. We will use the AIC score in the below models in order to compare performance against other model.

Rose Sales Data:

ARIMA:

After fitting model with all combination of order parameters on train data, AIC score in ascending order on train data is as follows,



	param	AIC
5	(0, 1, 2)	-186.803249
14	(1, 1, 2)	-186.195065
13	(1, 1, 1)	-186.101119
11	(1, 0, 2)	-185.365917
22	(2, 1, 1)	-184.829175

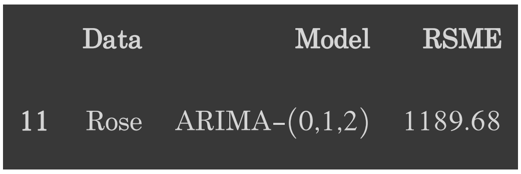
Table 21: AIC Score of Rose Sales Data

Order parameters(p,d,q) used:

[(0, 0, 0), (0, 0, 1), (0, 0, 2), (0, 1, 0), (0, 1, 1), (0, 1, 2), (0, 2, 0), (0, 2, 1), (0, 2, 2), (1, 0, 0), (1, 0, 1), (1, 0, 2), (1, 1, 0), (1, 1, 1), (1, 1, 2), (1, 2, 0), (1, 2, 1), (1, 2, 2), (2, 0, 0), (2, 0, 1), (2, 0, 2), (2, 1, 0), (2, 1, 1), (2, 1, 2), (2, 2, 0), (2, 2, 1), (2, 2, 2)]

Of all the orders, (p,d,q)=(0,1,2) performed best with AIC score of -186.8.

Hence lets build ARIMA model on rose data with above order parameter and check the RSME score on test data,



	Data	Model	RSME
11	Rose	ARIMA-(0,1,2)	1189.68

Table 22: RSME Score on Rose Data using ARIMA Model

Now the RSME score is not better than previously built exponential smoothing models above. Lets proceed further with other models.

SARIMA:

An extension to ARIMA that supports the direct modeling of the seasonal component of the series is called SARIMA.

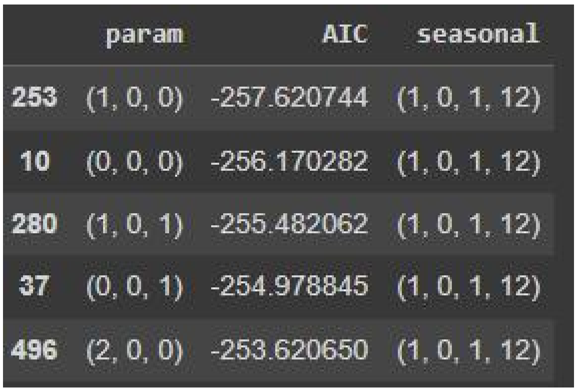
SARIMA model takes additional component called seasonality in order to learn from the data,

In our case, we have picked 12 months as seasonality parameter as it is evident from our data,

In addition to above order parameters, for SARIMA, below seasonal params have been used,

[(0, 0, 0, 12), (0, 0, 1, 12), (0, 0, 2, 12), (0, 1, 0, 12), (0, 1, 1, 12), (0, 1, 2, 12), (0, 2, 0, 12), (0, 2, 1, 12), (0, 2, 2, 12), (1, 0, 0, 12), (1, 0, 1, 12), (1, 0, 2, 12), (1, 1, 0, 12), (1, 1, 1, 12), (1, 1, 2, 12), (1, 2, 0, 12), (1, 2, 1, 12), (1, 2, 2, 12), (2, 0, 0, 12), (2, 0, 1, 12), (2, 0, 2, 12), (2, 1, 0, 12), (2, 1, 1, 12), (2, 1, 2, 12), (2, 2, 0, 12), (2, 2, 1, 12), (2, 2, 2, 12)]

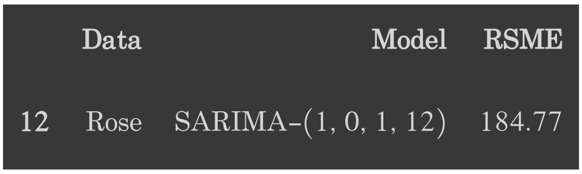
The least AIC scores of all above parameters on rose training data is as follows,



	param	AIC	seasonal
253	(1, 0, 0)	-257.620744	(1, 0, 1, 12)
10	(0, 0, 0)	-256.170282	(1, 0, 1, 12)
280	(1, 0, 1)	-255.482062	(1, 0, 1, 12)
37	(0, 0, 1)	-254.978845	(1, 0, 1, 12)
496	(2, 0, 0)	-253.620650	(1, 0, 1, 12)

Table 23: AIC Score of all Parameters on Rose Data

Parameter combination, order=(1, 0, 1) and seasonal_order=(1, 0, 1, 12) has given the least AIC score, hence lets build our model using the same and lets find the RSME score on test data.



	Data	Model	RSME
12	Rose	SARIMA-(1, 0, 1, 12)	184.77

Table 24: RSME Score on Rose Data using SARIMA Model

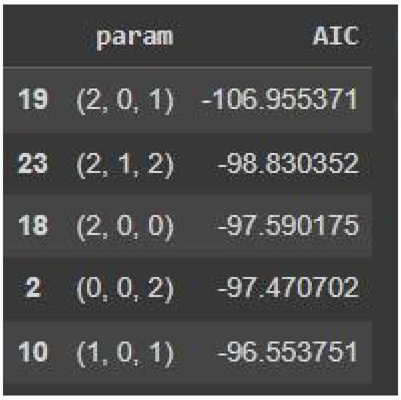
The RSME score is 184.77 which is the least of all our previously models built. Hence we will use this model only to predict future sales going further.

Sparkling Sales Data:

Similarly lets follow the same process to find the best parameters and models to forecast our Sparkling data.

ARIMA:

After fitting model with all combination of order parameters on train data, AIC score in ascending order on train data is as follows,



	param	AIC
19	(2, 0, 1)	-106.955371
23	(2, 1, 2)	-98.830352
18	(2, 0, 0)	-97.590175
2	(0, 0, 2)	-97.470702
10	(1, 0, 1)	-96.553751

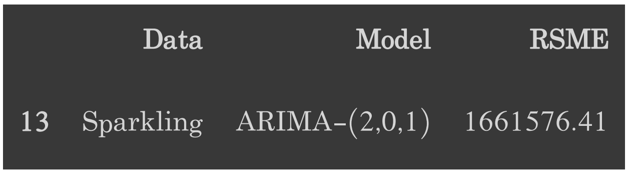
Table 25: AIC Score of Sparkling Sales Data

Order parameters(p,d,q) used:

[(0, 0, 0), (0, 0, 1), (0, 0, 2), (0, 1, 0), (0, 1, 1), (0, 1, 2), (0, 2, 0), (0, 2, 1), (0, 2, 2), (1, 0, 0), (1, 0, 1), (1, 0, 2), (1, 1, 0), (1, 1, 1), (1, 1, 2), (1, 2, 0), (1, 2, 1), (1, 2, 2), (2, 0, 0), (2, 0, 1), (2, 0, 2), (2, 1, 0), (2, 1, 1), (2, 1, 2), (2, 2, 0), (2, 2, 1), (2, 2, 2)]

Of all the orders, (p,d,q)=(2,0,1) performed best with AIC score of -106.9.

Hence lets build ARIMA model on rose data with above order parameter and check the RSME score on test data,



	Data	Model	RSME
13	Sparkling	ARIMA-(2,0,1)	1661576.41

Table 26: RSME Score on Sparkling Data using ARIMA Model

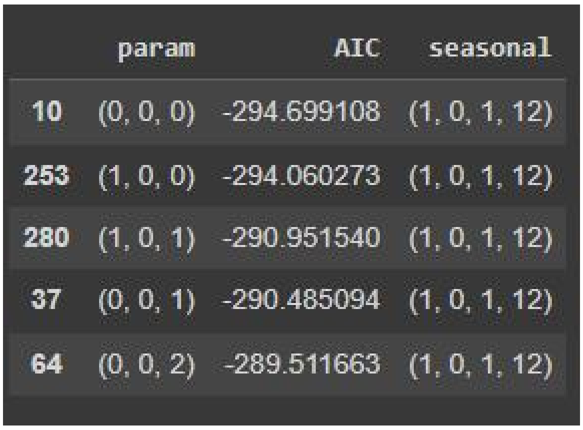
Now the RSME score is not better than previously built exponential smoothing models above. Lets proceed further with other models.

SARIMA:

In addition to above order parameters, for SARIMA, below seasonal params have been used,

[(0, 0, 0, 12), (0, 0, 1, 12), (0, 0, 2, 12), (0, 1, 0, 12), (0, 1, 1, 12), (0, 1, 2, 12), (0, 2, 0, 12), (0, 2, 1, 12), (0, 2, 2, 12), (1, 0, 0, 12), (1, 0, 1, 12), (1, 0, 2, 12), (1, 1, 0, 12), (1, 1, 1, 12), (1, 1, 2, 12), (1, 2, 0, 12), (1, 2, 1, 12), (1, 2, 2, 12), (2, 0, 0, 12), (2, 0, 1, 12), (2, 0, 2, 12), (2, 1, 0, 12), (2, 1, 1, 12), (2, 1, 2, 12), (2, 2, 0, 12), (2, 2, 1, 12), (2, 2, 2, 12)]

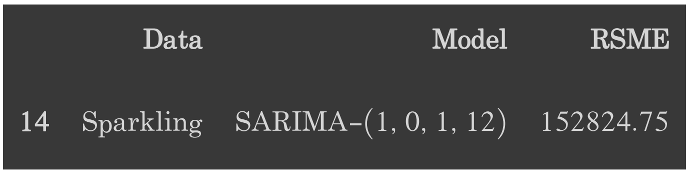
The least AIC scores of all above parameters on sparkling training data is as follows,



	param	AIC	seasonal
10	(0, 0, 0)	-294.699108	(1, 0, 1, 12)
253	(1, 0, 0)	-294.060273	(1, 0, 1, 12)
280	(1, 0, 1)	-290.951540	(1, 0, 1, 12)
37	(0, 0, 1)	-290.485094	(1, 0, 1, 12)
64	(0, 0, 2)	-289.511663	(1, 0, 1, 12)

Table 27: AIC Score of all Parameters on Sparkling Data

Parameter combination, order=(0, 0, 0) and seasonal_order=(1, 0, 1, 12) has given the least AIC score, hence lets build our model using the same and lets find the RSME score on test data.



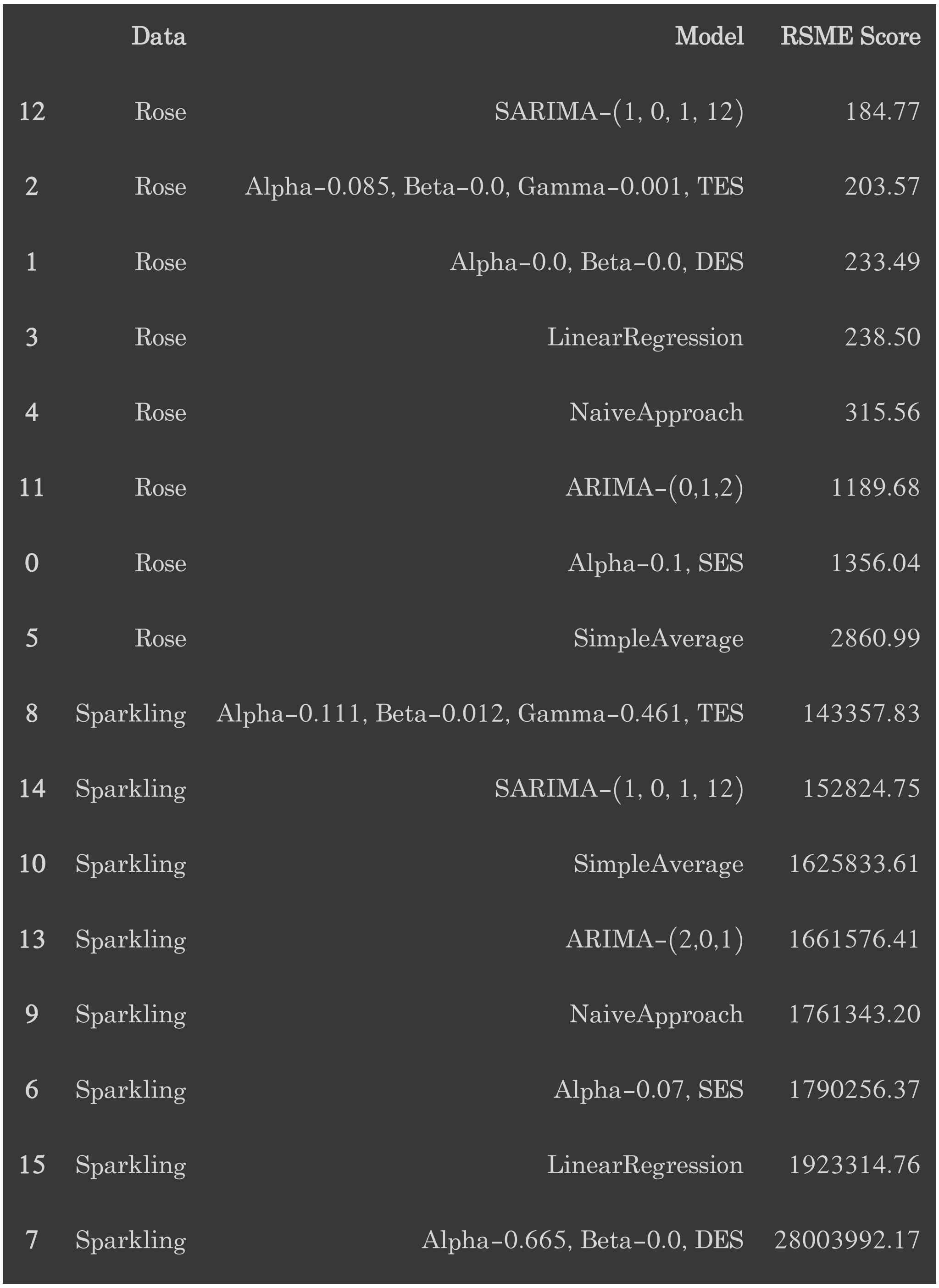
	Data	Model	RSME
14	Sparkling	SARIMA-(1, 0, 1, 12)	152824.75

Table 28: RSME Score on Sparkling Data using SARIMA Model

The RSME score is 152824.75 which is the not the least of all our previously models built but second least RSME score after Triple exponential model.

7. Build a table (create a data frame) with all the models built along with their corresponding parameters and the respective RMSE values on the test data.

The RSME scores of all the models built so far along with the parameters used are as follows,



	Data	Model	RSME Score
12	Rose	SARIMA-(1, 0, 1, 12)	184.77
2	Rose	Alpha-0.085, Beta-0.0, Gamma-0.001, TES	203.57
1	Rose	Alpha-0.0, Beta-0.0, DES	233.49
3	Rose	LinearRegression	238.50
4	Rose	NaiveApproach	315.56
11	Rose	ARIMA-(0,1,2)	1189.68
0	Rose	Alpha-0.1, SES	1356.04
5	Rose	SimpleAverage	2860.99
8	Sparkling	Alpha-0.111, Beta-0.012, Gamma-0.461, TES	143357.83
14	Sparkling	SARIMA-(1, 0, 1, 12)	152824.75
10	Sparkling	SimpleAverage	1625833.61
13	Sparkling	ARIMA-(2,0,1)	1661576.41
9	Sparkling	NaiveApproach	1761343.20
6	Sparkling	Alpha-0.07, SES	1790256.37
15	Sparkling	LinearRegression	1923314.76
7	Sparkling	Alpha-0.665, Beta-0.0, DES	28003992.17

Table 29: RSME Score on Rose and Sparkling Data using all Models

8. 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.

From the above table of RSME scores, we can conclude that for **forecasting Rose sales data, SARIMA model would be optimal with parameters (1,0,1,12).**

Similarly **for forecasting Sparkling sales data, Triple Exponential Smoothing model is the best with least RSME score of 143357.83** followed by SARIMA model with parameter (1,0,1,12) with RSME score of 152824.75

Using SARIMA Model to forecast next 12 months(i.e., from Aug,1995 to July,1996) Rose sales data with confidence interval of 5%:

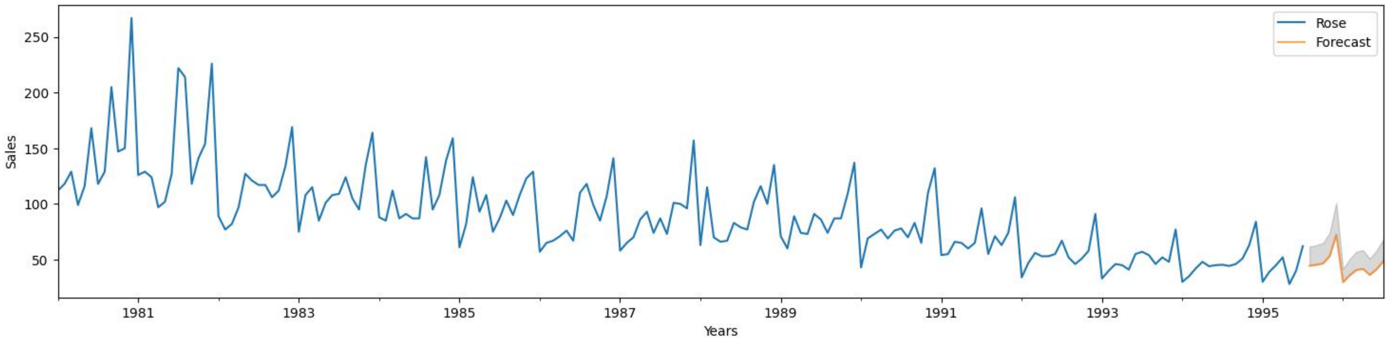


Fig.18 – Rose sales data Plot using SARIMA Model

Using TES Model to forecast next 12 months(i.e., from Aug,1995 to July,1996) Sparkling sales data with confidence interval of 5%:

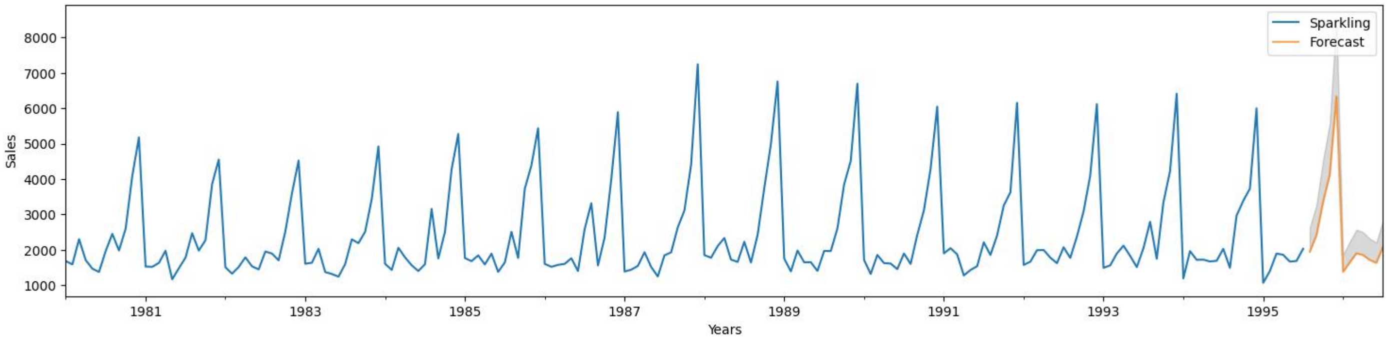


Fig.19 – Sparkling sales data Plot using TES Model

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

Steps Followed:

- Read time series data with appropriate index and date parameters.
- Performed EDA and imputed identified nulls in dataset.
- Visualized Time series data to check trends and seasonality.
- Decomposed data of both the time series to view trend and seasonal components individually.
- Have split the data into training and test sets.

- Fitted data in all the exponential models and calculated RSME scores on test data and predicted values.
- Created automated models to find the best ARIMA/SARIMA models using the least (Akaike Information Criterion)AIC score.
- Found the optimum models based on all the above created models and forecasted next 12 months sales.

<u>Conclusion and Suggestions based above models:</u>

Rose Sales Forecasting:

The sales for rose wine has been continuously declining over the years and its predicted to decline further.

Hence necessary steps to be taken to bring back the rose wine sales upwards.

Even the amount/number of sales for Rose wines is comparatively lower than that of Sparkling wine sales.

Sparkling Sales Forecasting:

Sales for Sparkling wines is pretty much consistent over the years and sales in future are also predicted to be consistent.

For both of the wines, December has been the peak month with highest number of sales.

Hence any necessary steps can be taken to further increase the sales in those months by producing more and catering to more customers.

Also, in contrary, we can plan man power, inventory and production accordingly in less demand seasons like January for Rose wines and June for Sparkling wine.