Rose Wine

ANALYSIS AND PREDICTION

SUHAS PAWAR | TIME SERIES FORECASTING

Problem Statement:

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

Data set for the Problem: Rose.csv

Please do perform the following questions on each of these two data sets separately.

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

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

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

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

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.

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.

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

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

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

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

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

First of all we will import all the necessary libraries like Pandas, Numpy, seaborn, os to set the path and Statsmodels as base library for all the models.

After setting the path we will import Rose dataset

We will read the dataset First by using pandas library.

Then We will Apply head and tail to see how data looks first and last 5 rows.

First 5 Rows:

YearMonth	Rose
1980-01	112.00
1980-02	118.00
1980-03	129.00
1980-04	99.00
1980-05	116.00

Last 5 Rows:

YearMonth	Rose
1995-03	45.00
1995-04	52.00
1995-05	28.00
1995-06	40.00
1995-07	62.00

Null Value check:

We will check for null condition, to see whether Dataset has null value or not.

	YearMonth	Rose
174	1994-07	nan
175	1994-08	nan

Row no. 174 and 175 has Null values, we will impute this By using *forward filling method.*

Time_Stamp	Rose
1980-01-31	112.00
1980-02-29	118.00
1980-03-31	129.00
1980-04-30	99.00
1980-05-31	116.00
...	...
1994-05-31	44.00
1994-06-30	45.00
1994-07-31	nan
1994-08-31	nan
1994-09-30	46.00

Shape of the Data:

```
(187, 2)
```

So Dataset has 187 rows and 2 Columns.

Data Info:

```
<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: 3.0+ KB
```

We can see that YearMonth column is of Object Data type while Rose is having float data type.

Rose column has 2 missing values.

Here we make an assumption that the date starts and ends as mentioned below

We are assuming the Date Starts from 1980-01-31 to 1995-07-31.

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

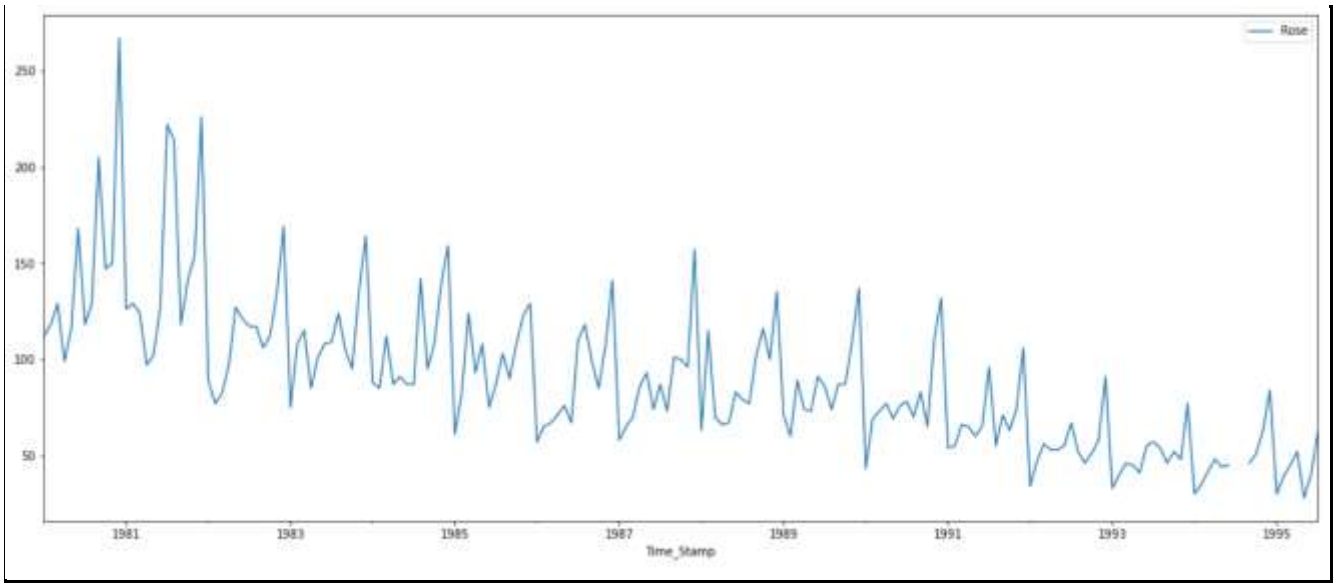
Total 187 entries needs to be added to Original Dataframe while we need to remove YearMonth Column.

And need to make This new Column as Index.

	YearMonth	Rose	Time_Stamp
0	1980-01	112.00	1980-01-31
1	1980-02	118.00	1980-02-29
2	1980-03	129.00	1980-03-31
3	1980-04	99.00	1980-04-30
4	1980-05	116.00	1980-05-31

Time_Stamp	Rose
1980-01-31	112.00
1980-02-29	118.00
1980-03-31	129.00
1980-04-30	99.00
1980-05-31	116.00

We will plot the original dataset of Rose.



We can see discontinuity in Year 1994-95 as 2 values are missing of Sales-Rose Wines. From Above plot we can see that Data has Negative Trend, Seasonality.

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

Description:

	Rose
count	187.00
mean	89.91
std	39.24
min	28.00
25%	62.50
50%	85.00
75%	111.00
max	267.00

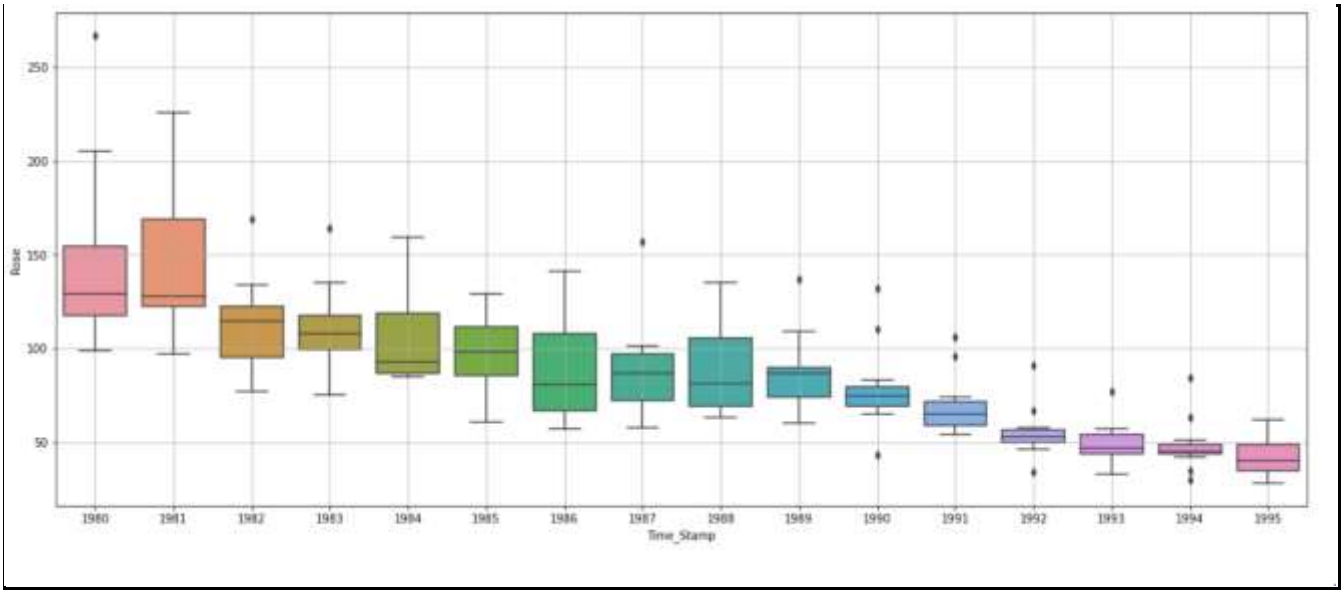
from above describe Function we can see that Highest Sales of Rose wine is 267.Mean and Median are almost same here.

Info check after Imputation of Null values:

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 187 entries, 1980-01-31 to 1995-07-31
Data columns (total 1 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Rose    187 non-null    int32
dtypes: int32(1)
memory usage: 2.2 KB
```

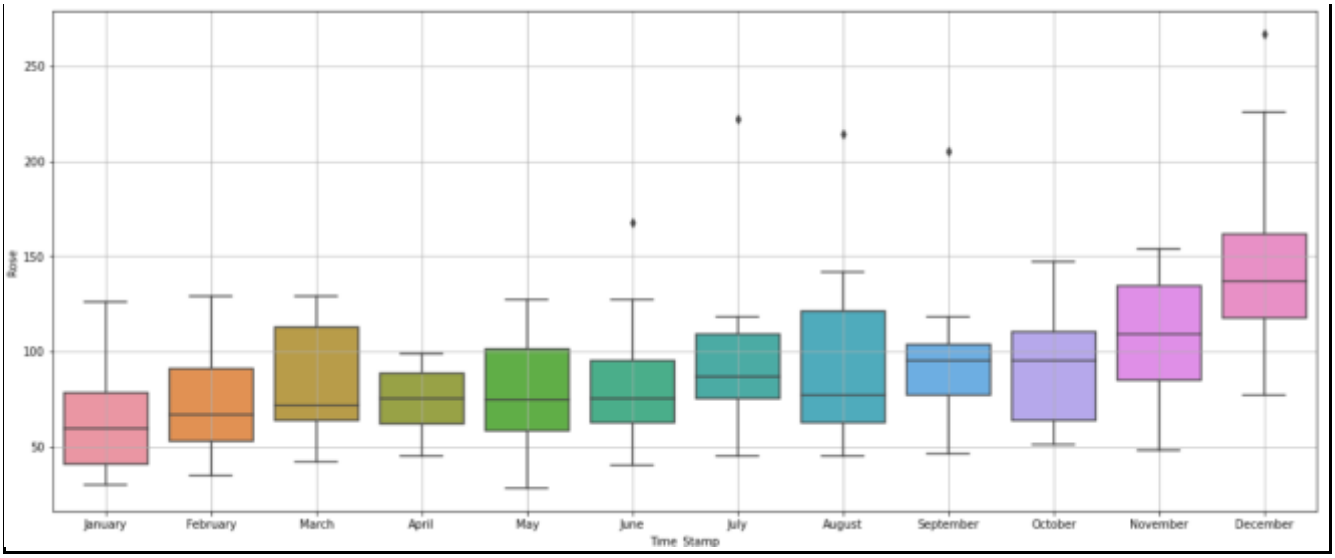
Total 187 rows we have enties from Jan 1980 to July 1995.

Yearly plot:

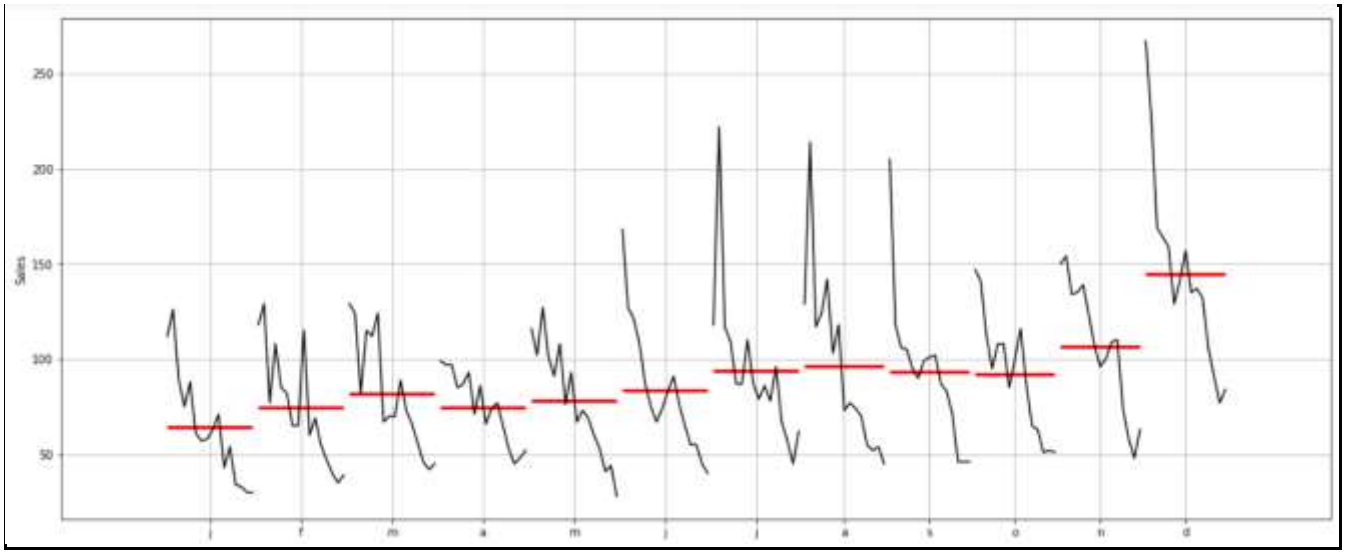


From Yearly plot we can see that Sales of Rose wine are getting down by every year, This continuous decrease in the Sales needs a deep study.

Monthly plot:



Here we come with interesting insights from Monthly plot, From January to October Median of Sales is almost Same. Only Sales are increasing in November and December. In This December Only Sales has Crossed figure of 250 wines.

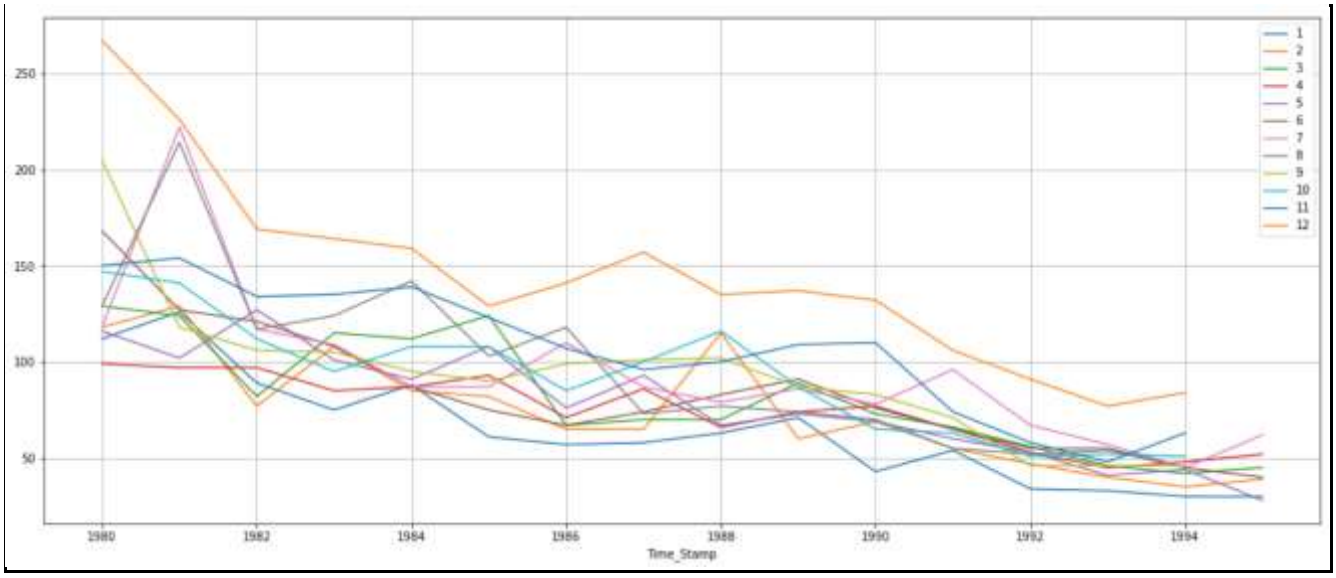


Highlight The maximum sales:

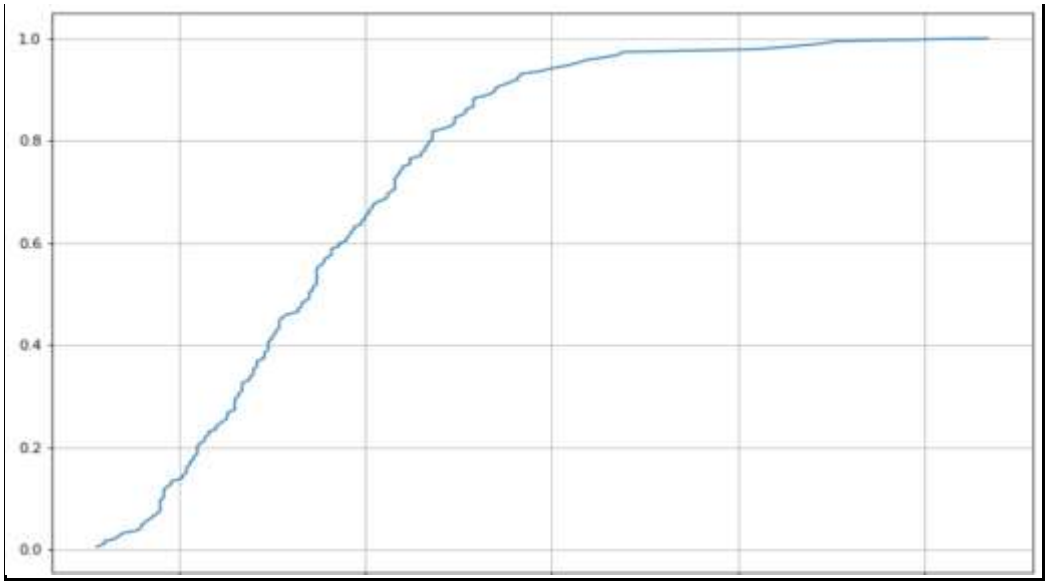
Time_Stamp	1	2	3	4	5	6	7	8	9	10	11	12
Time_Stamp												
1980	112.000000	118.000000	129.000000	99.000000	116.000000	168.000000	118.000000	129.000000	205.000000	147.000000	150.000000	267.000000
1981	126.000000	129.000000	124.000000	97.000000	102.000000	127.000000	222.000000	214.000000	118.000000	141.000000	154.000000	226.000000
1982	89.000000	77.000000	82.000000	97.000000	127.000000	121.000000	117.000000	117.000000	106.000000	112.000000	134.000000	169.000000
1983	75.000000	108.000000	115.000000	85.000000	101.000000	108.000000	109.000000	124.000000	105.000000	95.000000	135.000000	164.000000
1984	88.000000	85.000000	112.000000	87.000000	91.000000	87.000000	87.000000	142.000000	95.000000	108.000000	139.000000	159.000000
1985	61.000000	82.000000	124.000000	93.000000	108.000000	75.000000	87.000000	103.000000	90.000000	108.000000	123.000000	129.000000
1986	57.000000	65.000000	67.000000	71.000000	76.000000	67.000000	110.000000	118.000000	99.000000	85.000000	107.000000	141.000000
1987	58.000000	65.000000	70.000000	86.000000	93.000000	74.000000	87.000000	73.000000	101.000000	100.000000	96.000000	157.000000
1988	63.000000	115.000000	70.000000	66.000000	67.000000	83.000000	79.000000	77.000000	102.000000	116.000000	100.000000	135.000000
1989	71.000000	60.000000	89.000000	74.000000	73.000000	91.000000	86.000000	74.000000	87.000000	87.000000	109.000000	137.000000
1990	43.000000	69.000000	73.000000	77.000000	69.000000	76.000000	78.000000	70.000000	83.000000	65.000000	110.000000	132.000000
1991	54.000000	55.000000	66.000000	65.000000	60.000000	65.000000	96.000000	55.000000	71.000000	63.000000	74.000000	106.000000
1992	34.000000	47.000000	56.000000	53.000000	53.000000	55.000000	67.000000	52.000000	46.000000	51.000000	58.000000	91.000000
1993	33.000000	40.000000	46.000000	45.000000	41.000000	55.000000	57.000000	54.000000	46.000000	52.000000	48.000000	77.000000
1994	30.000000	35.000000	42.000000	48.000000	44.000000	45.000000	45.000000	45.000000	46.000000	51.000000	63.000000	84.000000

1980 and 1981 are only Two years where Sales are Highest, But from this point the sales has decreased.

The orange line is for December month, It is Alienated from Other lines.

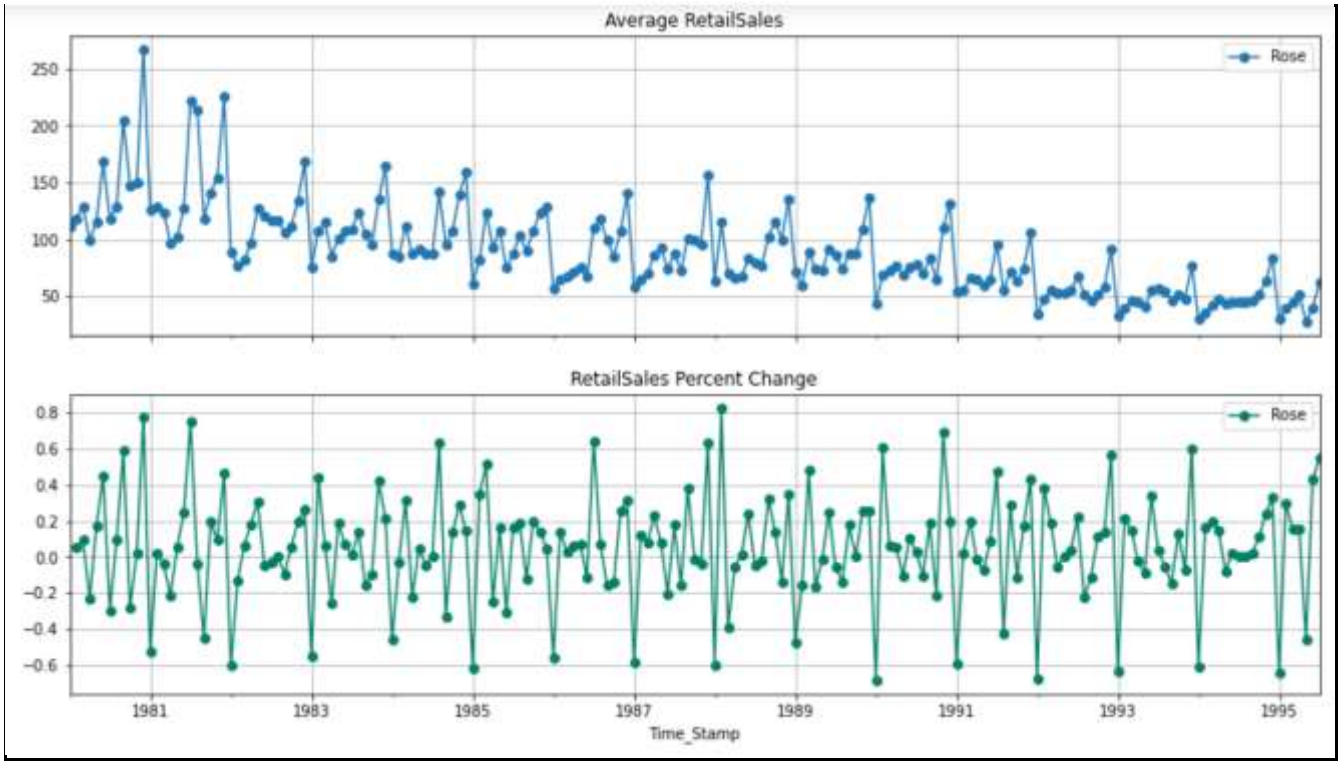


Empirical Cumulative Distribution.



This particular graph tells us what percentage of data points refer to what number of Sales. 50% o f the sales are below 100. Maximum sales is close 260.

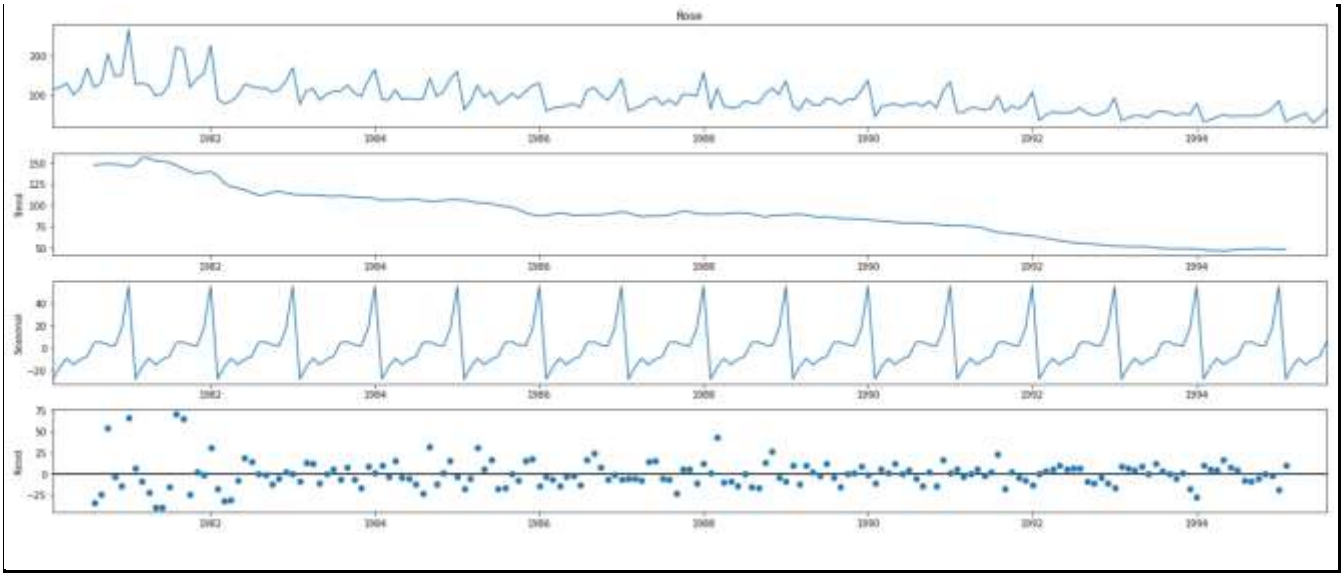
Plot of average Sales per month and the month on month percentage change of Sales.



In This plots we can see there is drastic change at the end of every year.

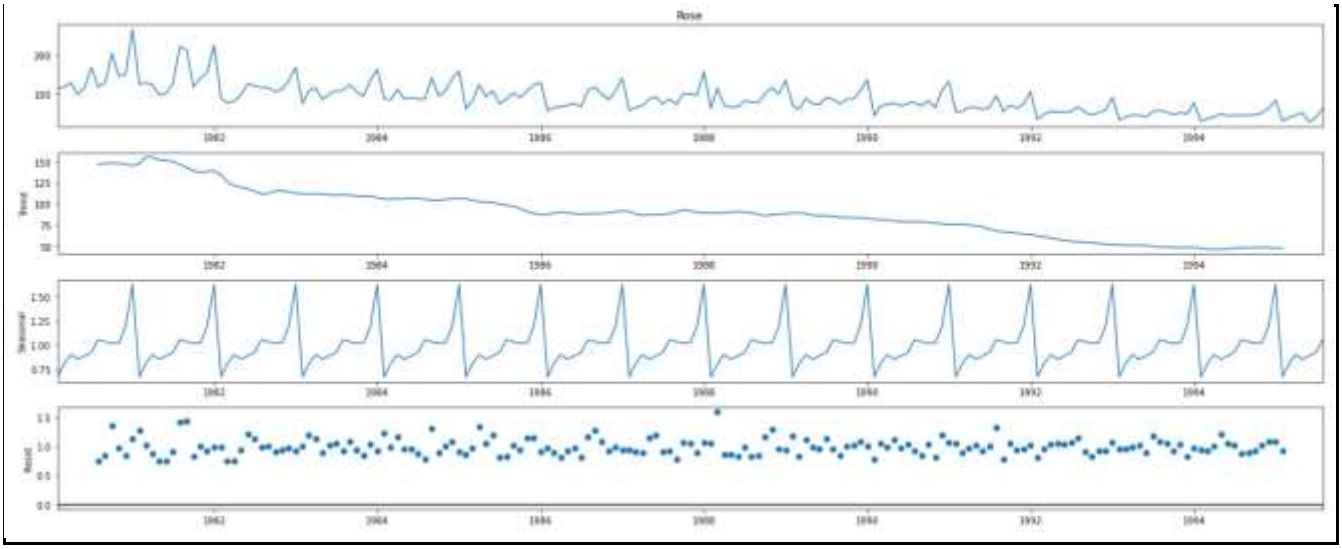
Decompose the Time Series and plot the different components.

Additive Decomposition:



From above Additive Decomposition we can see that There is Negative trend and Seasonality is there.residuals are showing patterns, Might me missing some quality Information.

Multiplicative Decomposition:



For the multiplicative decomposition series, we see that a lot of residuals are located around 1.

We can see the values of trend and Seasonality for Year 1980. In Case of Seasonality it will mostly repeat and In case of Trend it will keep decreasing overall.

```
Trend
 Time_Stamp
1980-01-31      nan
1980-02-29      nan
1980-03-31      nan
1980-04-30      nan
1980-05-31      nan
1980-06-30      nan
1980-07-31   147.08
1980-08-31   148.12
1980-09-30   148.37
1980-10-31   148.08
1980-11-30   147.42
1980-12-31   145.12
Name: trend, dtype: float64
```

```
Seasonality
 Time_Stamp
1980-01-31   0.67
1980-02-29   0.81
1980-03-31   0.90
1980-04-30   0.85
1980-05-31   0.89
1980-06-30   0.92
1980-07-31   1.06
1980-08-31   1.04
1980-09-30   1.02
1980-10-31   1.02
1980-11-30   1.19
1980-12-31   1.63
Name: seasonal, dtype: float64
```

```
Residual
 Time_Stamp
1980-01-31    nan
1980-02-29    nan
1980-03-31    nan
1980-04-30    nan
1980-05-31    nan
1980-06-30    nan
1980-07-31   0.76
1980-08-31   0.84
1980-09-30   1.36
1980-10-31   0.97
1980-11-30   0.85
1980-12-31   1.13
Name: resid, dtype: float64
```

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

Training Data is till the end of 1990. Test Data is from the beginning of 1991 to the last time stamp provided.

First few rows of Training Data

Time_Stamp	Rose
1980-01-31	112
1980-02-29	118
1980-03-31	129
1980-04-30	99
1980-05-31	116

Last few rows of Training Data

Time_Stamp	Rose
1990-08-31	70
1990-09-30	83
1990-10-31	65
1990-11-30	110
1990-12-31	132

First few rows of Test Data

Time_Stamp	Rose
1991-01-31	54
1991-02-28	55
1991-03-31	66
1991-04-30	65
1991-05-31	60

Last few rows of Test Data

Time_Stamp	Rose
1995-03-31	45
1995-04-30	52
1995-05-31	28
1995-06-30	40
1995-07-31	62

Shape of Train Data: 132 Rows and 1 Column

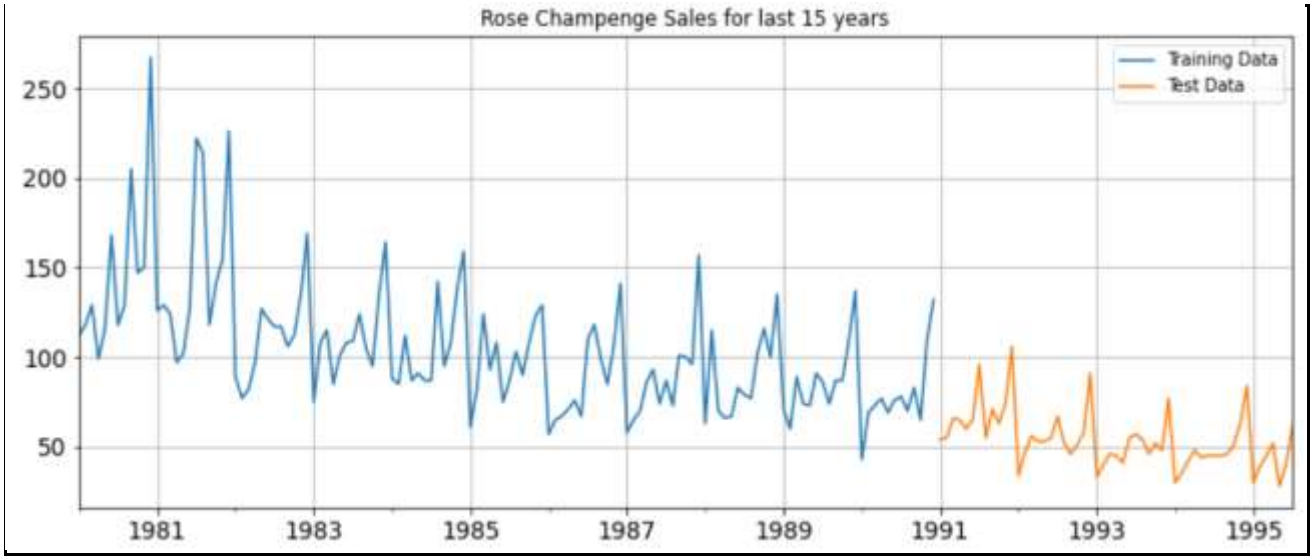
Shape of Test Data : 55 Rows and 1 Column

```
(132, 1)
(55, 1)
```

Train data Information:

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 132 entries, 1980-01-31 to 1990-12-31
Data columns (total 1 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Rose    132 non-null    int32
dtypes: int32(1)
memory usage: 1.5 KB
```

Train and Test data plot:



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

1.Simple Exponential Smoothing

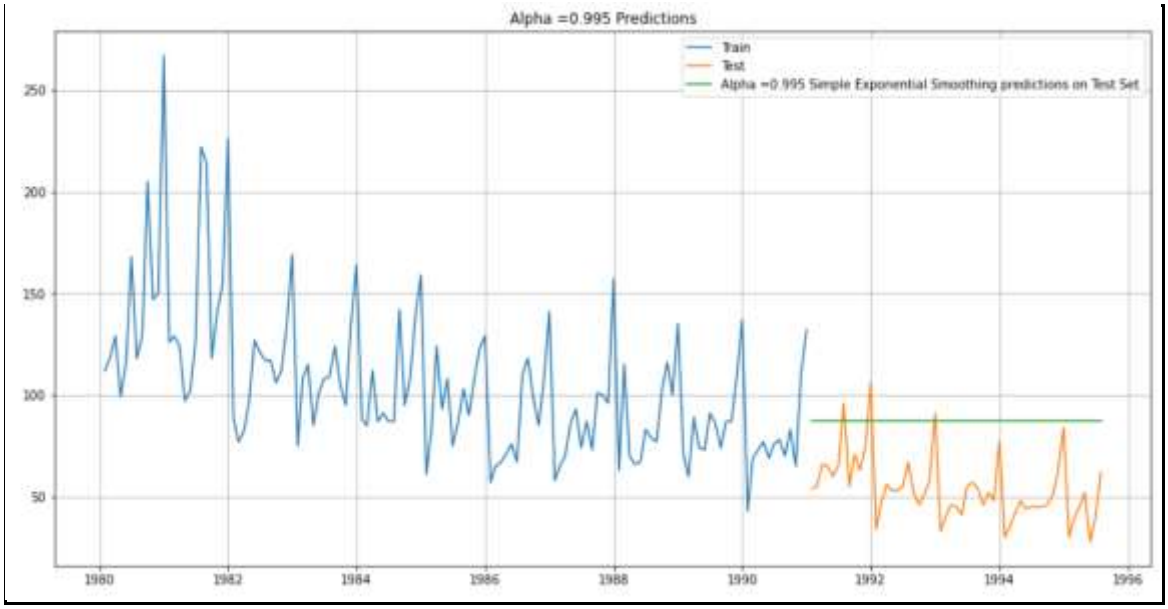
We have created the test and train data for the upcoming Models.

First we will create model on train Data, will test model performance on Test Data.

The Test data predictions for first 5 rows:

Time_Stamp	Rose	predict
1991-01-31	54	87.10
1991-02-28	55	87.10
1991-03-31	66	87.10
1991-04-30	65	87.10
1991-05-31	60	87.10

Plotting on both the Training and Test data:



Model Evaluation for α = 0.995 : Simple Exponential Smoothing

For Alpha =0.995 Simple Exponential Smoothing Model forecast on the Test Data, RMSE is 36.817

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82

Setting different alpha values. the higher the alpha value more weightage is given to the more recent observation. That means, what happened recently will happen again. We will run a loop with different alpha values to understand which particular value works best for alpha on the test set.

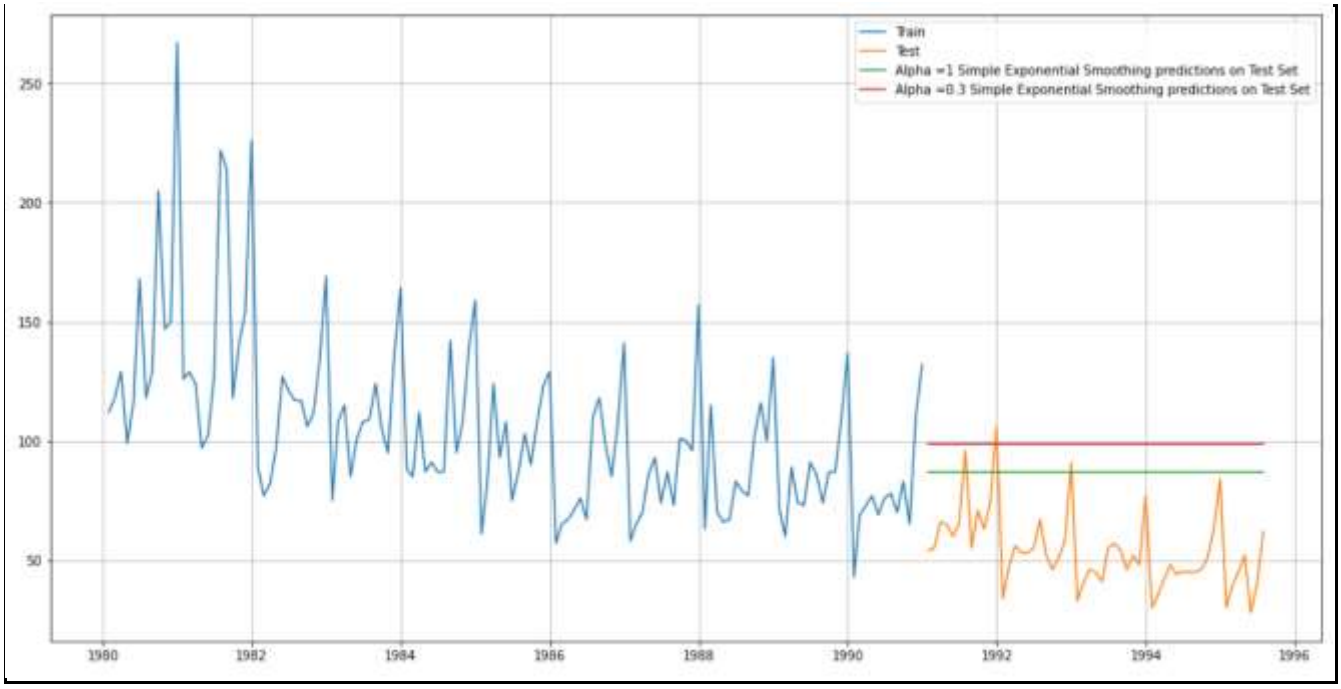
First we will define an empty dataframe to store our values from the loop

Alpha Values	Train RMSE	Test RMSE

Model Evaluation:

	Alpha Values	Train RMSE	Test RMSE
0	0.30	32.47	47.53
1	0.40	33.04	53.79
2	0.50	33.68	59.66
3	0.60	34.44	64.99
4	0.70	35.32	69.72
5	0.80	36.33	73.79
6	0.90	37.48	77.16

Plotting on both the Training and Test data



The Final Result of Simple Exponential Smoothing in form of RMSE:

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53

Model 2 : Double Exponential Smoothing (Holt's Model)

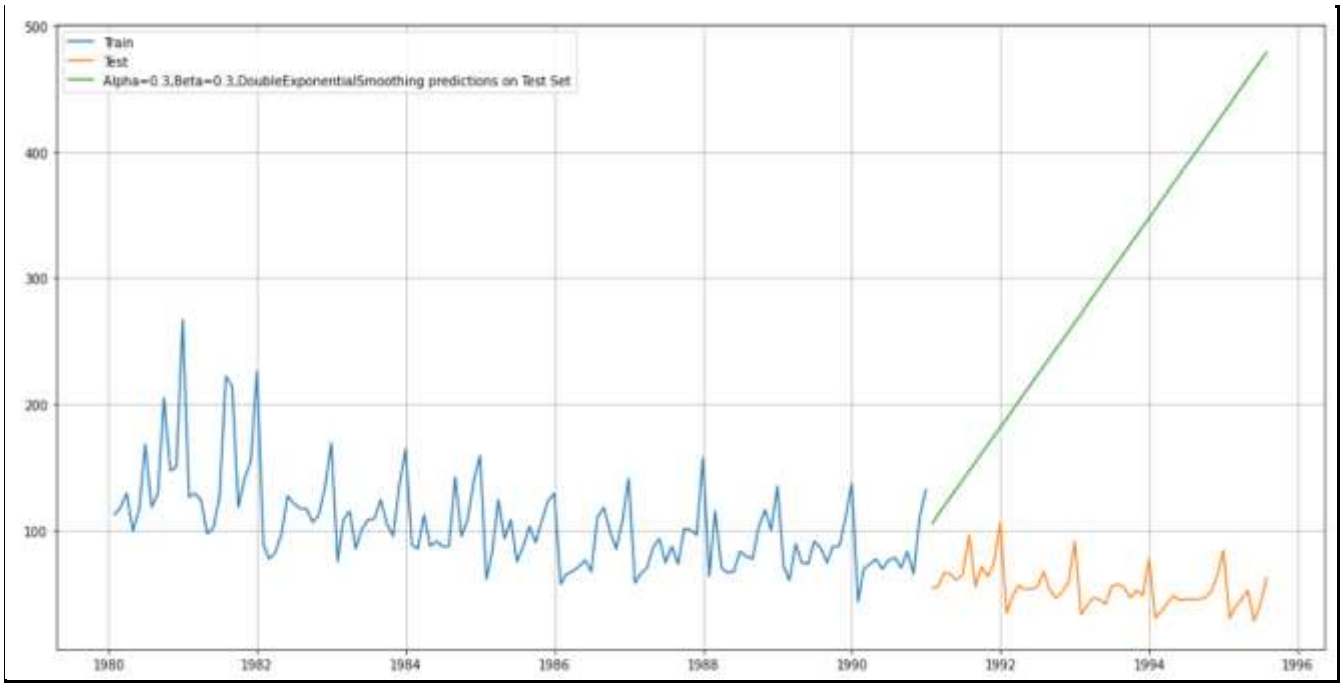
Two parameters α and β are estimated in this model. Level and Trend are accounted for in this model. First We will create a model and then we will predict on test data.

We will check the results for Different values of Alpha and Beta. Let us sort the data frame in the ascending ordering of the 'Test RMSE' and the 'Test MAPE' values.

	Alpha Values	Beta Values	Train RMSE	Test RMSE
0	0.30	0.30	35.94	265.59
8	0.40	0.30	36.75	339.33
1	0.30	0.40	37.39	358.78
16	0.50	0.30	37.43	394.30
24	0.60	0.30	38.35	439.32

Test RMSE is lowest for Alpha=0.3 and Beta =0.3. hence we will select these values for better performance.

Plotting on both the Training and Test data



Test RMSE :

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59

For this the Test RMSE is too high. So we cannot go further with this model. Lets deep dive more, to check about other models.

Method 3: Triple Exponential Smoothing (Holt - Winter's Model)

Three parameters α , β and γ are estimated in this model. Level, Trend and Seasonality are accounted for in this model.

Autofit parameters for the Triple Exponential smoothing:

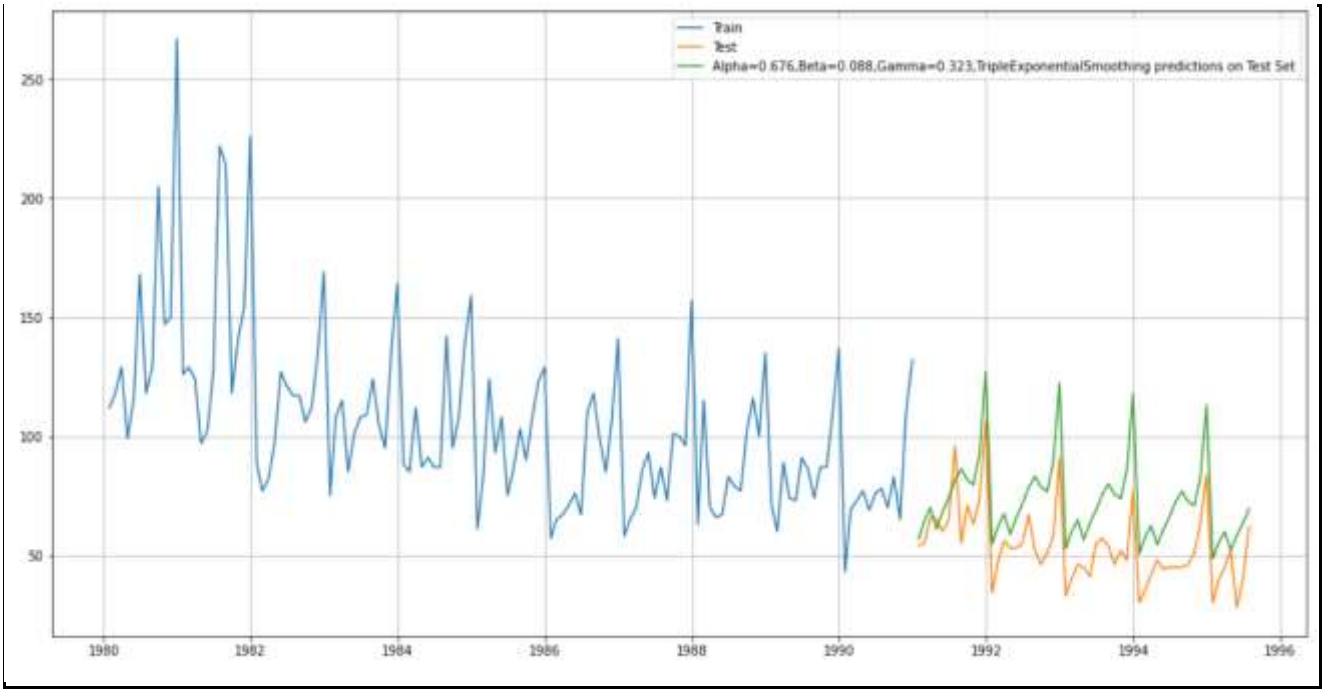
```
{'smoothing_level': 0.06475408609774214,
 'smoothing_trend': 0.05307002085376367,
 'smoothing_seasonal': 3.0517776640274286e-08,
 'damping_trend': nan,
 'initial_level': 61.109156896449434,
 'initial_trend': -0.3800333306252026,
 'initial_seasons': array([1.84498345, 2.09357803, 2.2872201 , 1.99940217, 2.24762785,
       2.45077751, 2.6939182 , 2.86477256, 2.71790667, 2.65917328,
       3.0997045 , 4.27511796]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

The above fit of the model is by the best parameters that Python thinks for the model. It uses a brute force method to choose the parameters.

Lets predict on the test data:

Time_Stamp	Rose	auto_predict
1991-01-31	54	56.75
1991-02-28	55	64.20
1991-03-31	66	69.93
1991-04-30	65	60.95
1991-05-31	60	68.31

Plotting on both the Training and Test using autofit



For Alpha=0.676, Beta=0.088, Gamma=0.323, Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 21.170

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17

Lets define an empty dataframe to store our values from the loop

Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE

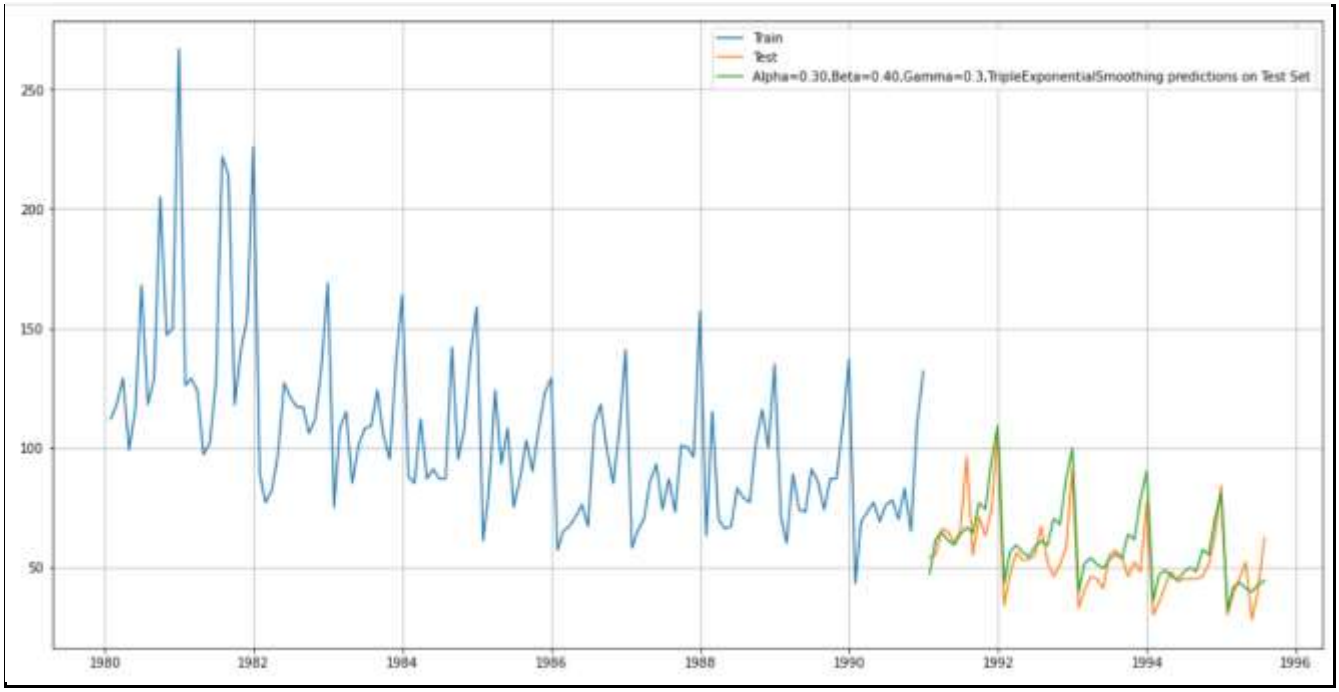
We will append all the results to it, by changing values of alpha, beta , gamma.

After Sorting the first five rows are

	Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE
8	0.30	0.40	0.30	28.11	10.95
1	0.30	0.30	0.40	27.40	11.19
69	0.40	0.30	0.80	32.60	12.61
16	0.30	0.50	0.30	29.09	14.40
131	0.50	0.30	0.60	32.14	16.70

So this model is giving RMSE 10.95 lowest so far, and Ideal values are 0.3,0.4,0.3 for Alpha, Beta , Gamma.

Plotting on both the Training and Test data using brute force alpha, beta and gamma determination



The sorted results of all the models so far:

	Test RMSE
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59

For this data, we had both trend and seasonality so by definition Triple Exponential Smoothing is supposed to work better than the Simple Exponential Smoothing as well as the Double Exponential Smoothing. However, since this was a model building exercise we had gone on to build different models on the data and have compared these model with the best RMSE value on the test data.

Model 4: Linear Regression

For this particular linear regression, we are going to regress the 'Rose-Sales' variable against the order of the occurrence. For this we need to modify our training data before fitting it into a linear regression.

```
Training Time instance
 [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 3
4, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65,
66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97,
98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123,
124, 125, 126, 127, 128, 129, 130, 131, 132]
Test Time instance
 [133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157,
158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 18
3, 184, 185, 186, 187]
```

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

Training Data:

First few rows of Training Data

Time_Stamp	Rose	time
1980-01-31	112	1
1980-02-29	118	2
1980-03-31	129	3
1980-04-30	99	4
1980-05-31	116	5

Last few rows of Training Data

Time_Stamp	Rose	time
1990-08-31	70	128
1990-09-30	83	129
1990-10-31	65	130
1990-11-30	110	131
1990-12-31	132	132

Test Data:

First few rows of Test Data

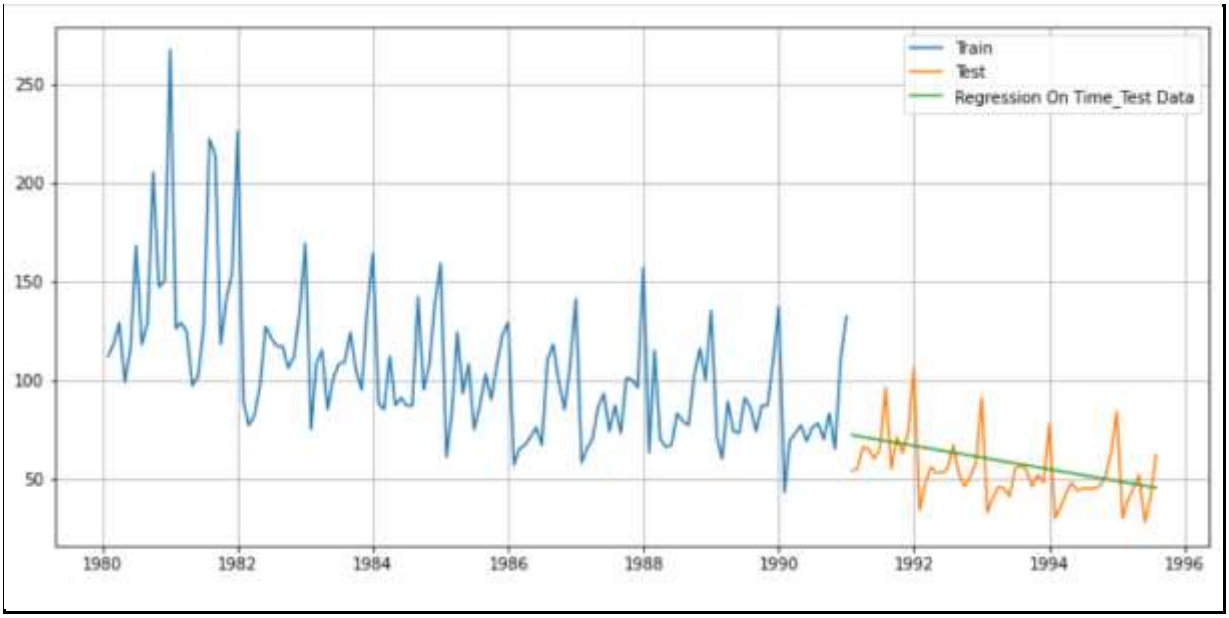
Time_Stamp	Rose	time
1991-01-31	54	133
1991-02-28	55	134
1991-03-31	66	135
1991-04-30	65	136
1991-05-31	60	137

Last few rows of Test Data

Time_Stamp	Rose	time
1995-03-31	45	183
1995-04-30	52	184
1995-05-31	28	185
1995-06-30	40	186
1995-07-31	62	187

Now our training and test data has been modified, let us go ahead use Linear regression to build the model on the training data and test the model on the test data.

Prediction plot:



Model Evaluation

For RegressionOnTime forecast on the Test Data, RMSE is 15.276

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28

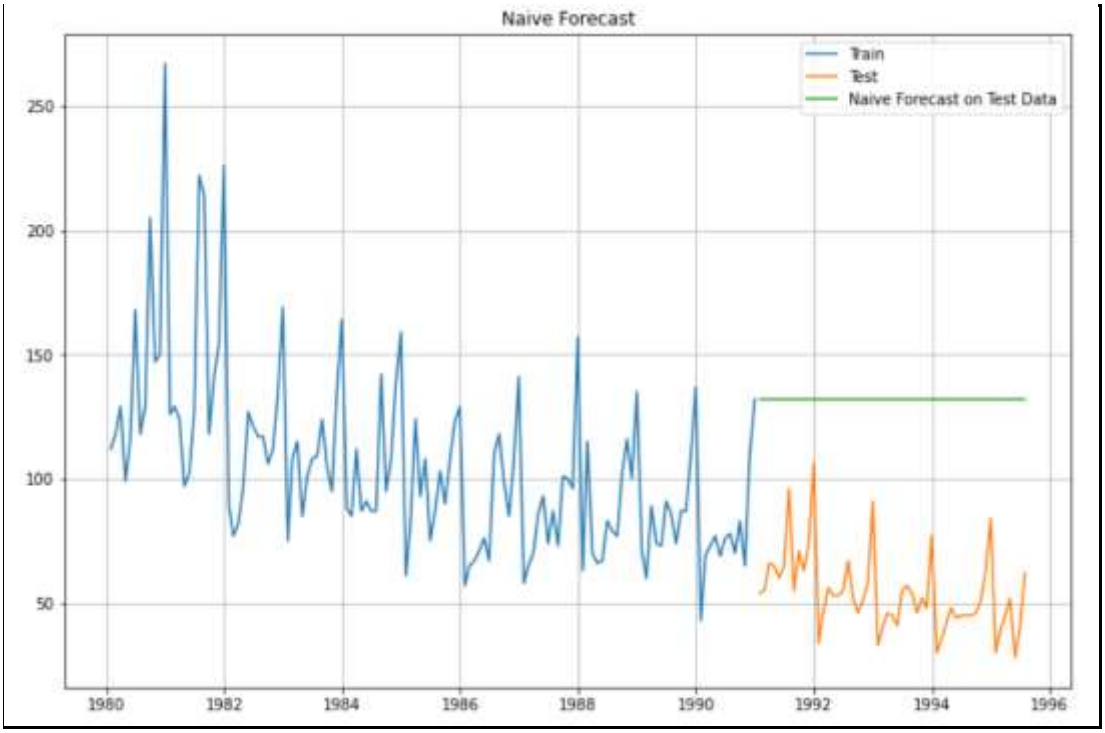
Model 5: Naive Approach:

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.

Prediction : First 5 rows:

```
Time_Stamp
1991-01-31    132
1991-02-28    132
1991-03-31    132
1991-04-30    132
1991-05-31    132
Name: naive, dtype: int32
```

Prediction Plot:



Model Evaluation

For RegressionOnTime forecast on the Test Data, RMSE is 79.739

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74

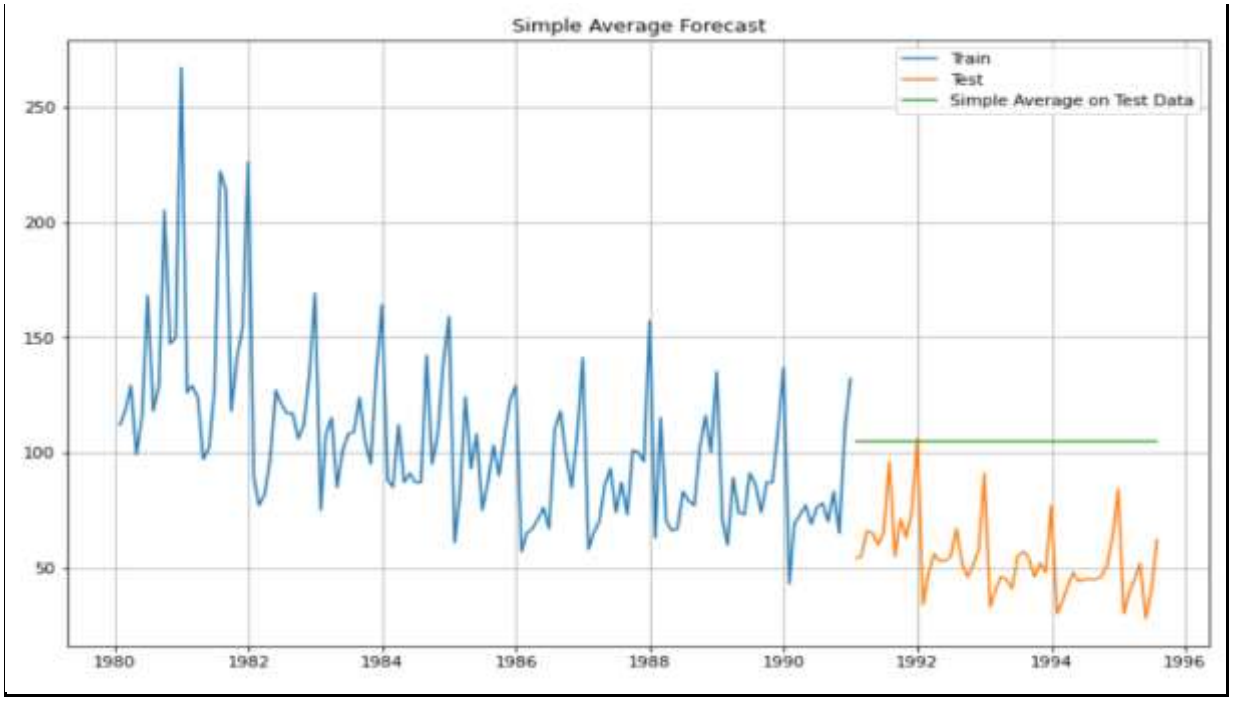
Model 6: Simple Average

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

Forecast:

Time_Stamp	Rose	mean_forecast
1991-01-31	54	104.94
1991-02-28	55	104.94
1991-03-31	66	104.94
1991-04-30	65	104.94
1991-05-31	60	104.94

Prediction Plot vs Test Data:



Model Evaluation

For Simple Average forecast on the Test Data, RMSE is 53.481

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48

Model 7: Moving Average(MA)

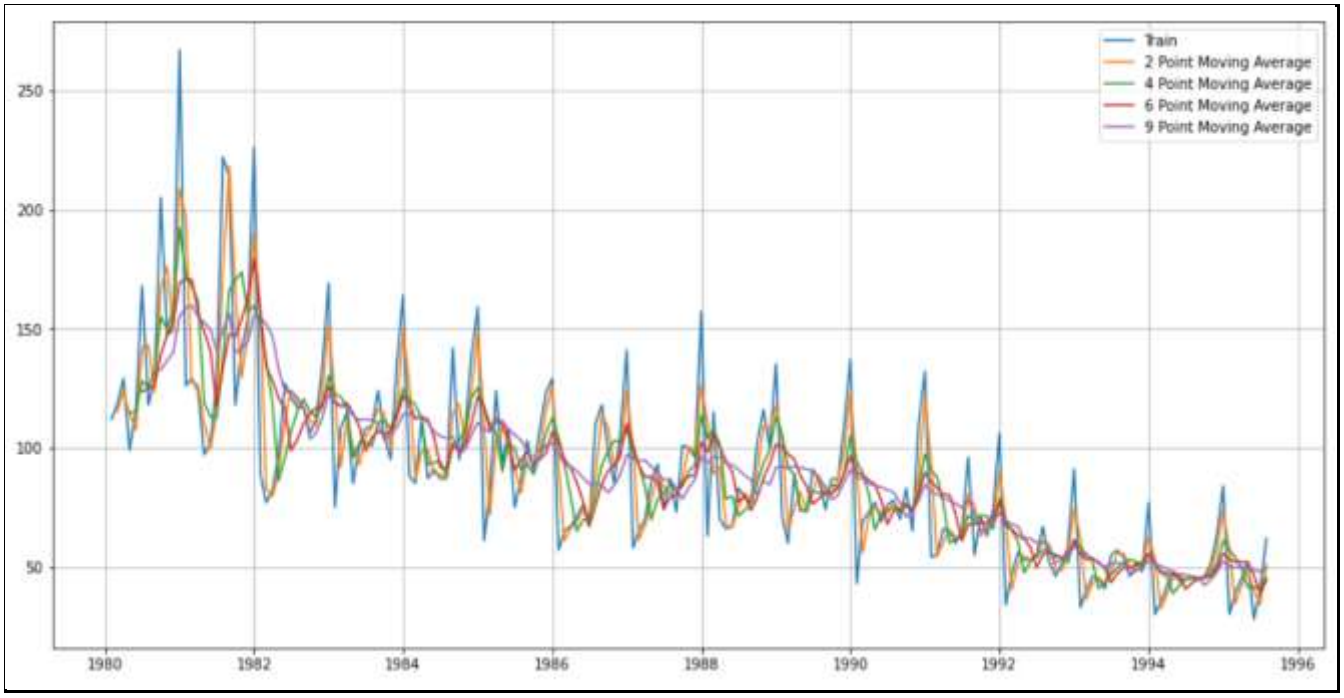
For the moving average model, we are going to calculate rolling means (or moving averages) for different intervals. The best interval can be determined by the maximum accuracy (or the minimum error) over here.
For Moving Average, we are going to average over the entire data.

Time_Stamp	Rose
1980-01-31	112
1980-02-29	118
1980-03-31	129
1980-04-30	99
1980-05-31	116

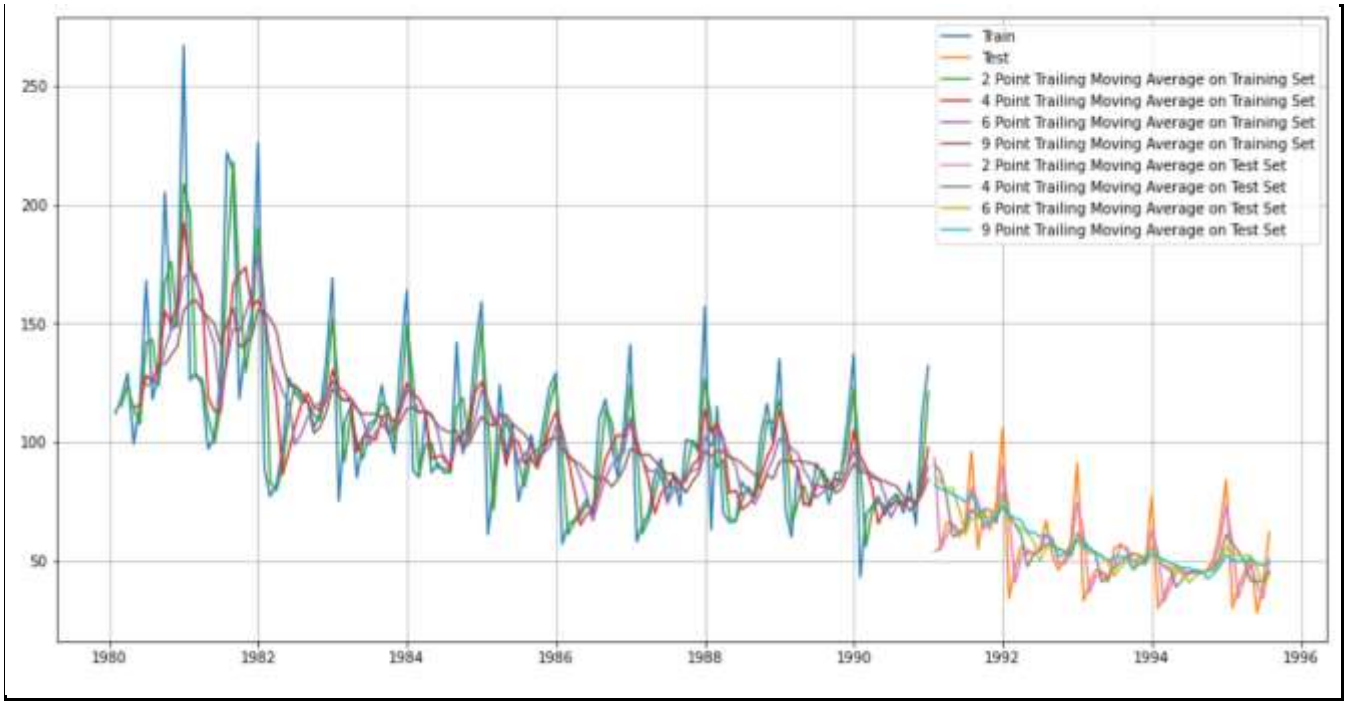
For Trailing Average,

Time_Stamp	Rose	Trailing_2	Trailing_4	Trailing_6	Trailing_9
1980-01-31	112	nan	nan	nan	nan
1980-02-29	118	115.00	nan	nan	nan
1980-03-31	129	123.50	nan	nan	nan
1980-04-30	99	114.00	114.50	nan	nan
1980-05-31	116	107.50	115.50	nan	nan

Plot for 2,4,6,9 Point Moving Average



Let us split the data into train and test and plot this Time Series. The window of the moving average is need to be carefully selected as too big a window will result in not having any test set as the whole series might get averaged over.



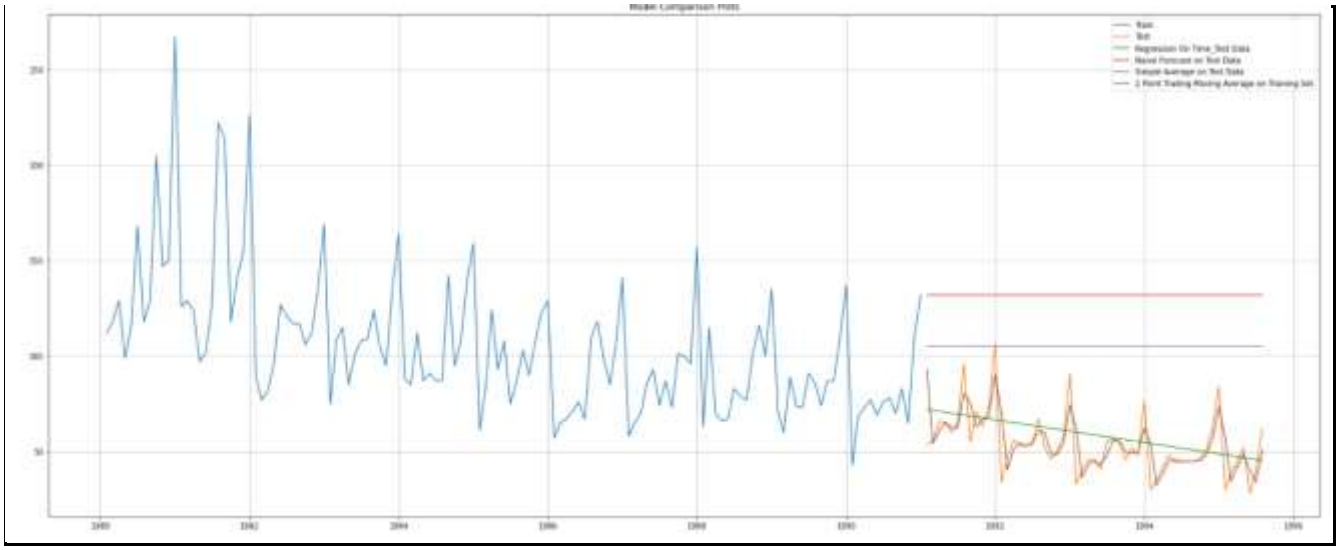
Model Evaluation

Done only on Test data

For 2 point Moving Average Model forecast on the Training Data, RMSE is 11.529
For 4 point Moving Average Model forecast on the Training Data, RMSE is 14.455
For 6 point Moving Average Model forecast on the Training Data, RMSE is 14.572
For 9 point Moving Average Model forecast on the Training Data, RMSE is 14.731

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73

Plotting on both Training and Test data



Sorted by RMSE values on the Test Data:

	Test RMSE
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
RegressionOnTime	15.28
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
SimpleAverageModel	53.48
NaiveModel	79.74
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59

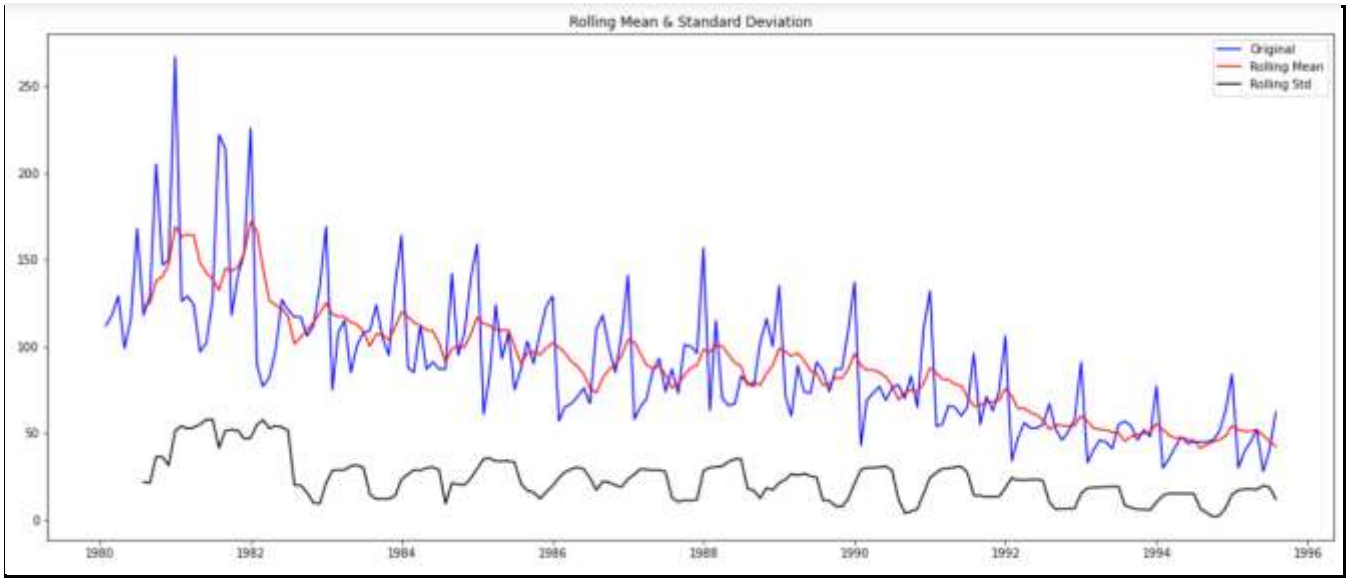
5.Check for the stationarity of the data on which the model is being built on using appropriate statistical tests and also mention the hypothesis for the statistical test. If the data is found to be non-stationary, take appropriate steps to make it stationary. Check the new data for stationarity and comment. Note: Stationarity should be checked at alpha = 0.05.

Check for Stationarity:

Dicky Fuller Test

Null Hypothesis Ho- Series is not Stationary

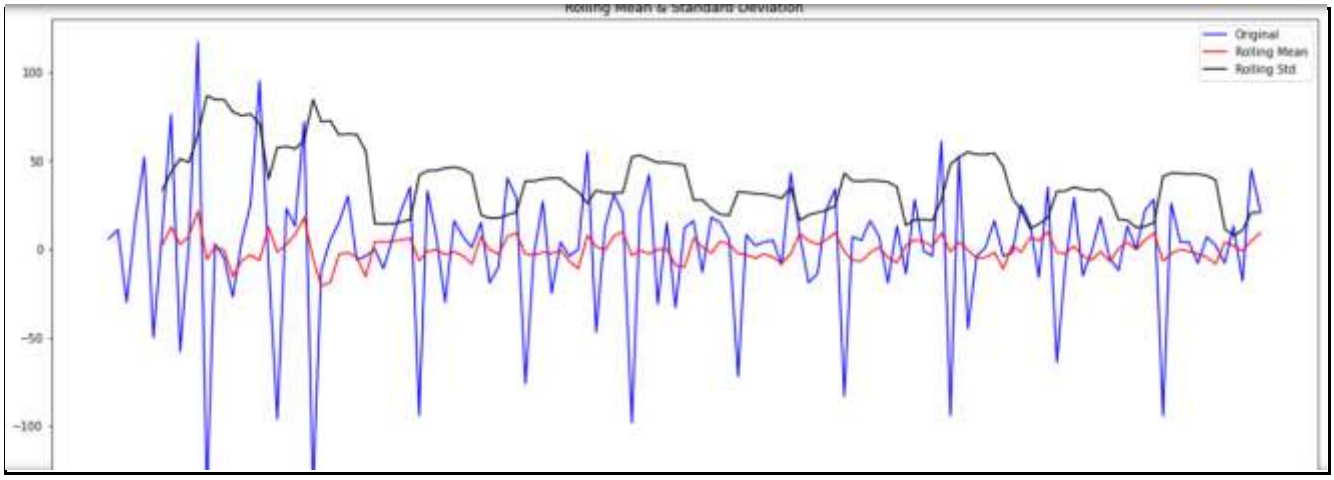
Alternative Hypothesis H1- Series is Stationary



P value is higher that 0.05 Hence Null Hypothesis is True, we will take Order 1 differencing to make series stationary

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.87
p-value                          0.34
#Lags Used                      13.00
Number of Observations Used    173.00
Critical Value (1%)             -3.47
Critical Value (5%)             -2.88
Critical Value (10%)            -2.58
dtype: float64
```

We check stationarity at initial level, but series is not stationary as P value is higher than 0.05 difference of order 1.




```
Results of Dickey-Fuller Test:
Test Statistic                  -6.59
p-value                          0.00
#Lags Used                      12.00
Number of Observations Used    118.00
Critical Value (1%)             -3.49
Critical Value (5%)             -2.89
Critical Value (10%)            -2.58
dtype: float64
```

We see that at P value is less than 0.05 the Time Series is indeed stationary.

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. 8

Automated Verion of ARIMA

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

```
Some parameter combinations for the Model...
Model: (0, 1, 0)
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)
```

We will Apply all values of p and q. check which combination is giving us the low AIC score.

```
ARIMA(0, 1, 0) - AIC:1335.1526583086775
ARIMA(0, 1, 1) - AIC:1280.7261830464336
ARIMA(0, 1, 2) - AIC:1276.8353748558739
ARIMA(1, 1, 0) - AIC:1319.3483105801956
ARIMA(1, 1, 1) - AIC:1277.7757578176656
ARIMA(1, 1, 2) - AIC:1277.359222277722
ARIMA(2, 1, 0) - AIC:1300.6092611744687
ARIMA(2, 1, 1) - AIC:1279.0456894093138
ARIMA(2, 1, 2) - AIC:1279.2986939364855
```

Sort the above AIC values in the ascending order to get the parameters for the minimum AIC value

	param	AIC
2	(0, 1, 2)	1276.84
5	(1, 1, 2)	1277.36
4	(1, 1, 1)	1277.78
7	(2, 1, 1)	1279.05
8	(2, 1, 2)	1279.30
1	(0, 1, 1)	1280.73
6	(2, 1, 0)	1300.61
3	(1, 1, 0)	1319.35
0	(0, 1, 0)	1335.15

ARIMA Model Results

Dep. Variable:	D.Rose	No. Observations:	131
Model:	ARIMA(0, 1, 2)	Log Likelihood	-634.418
Method:	css-mle	S.D. of innovations	30.167
Date:	Tue, 22 Jun 2021	AIC	1276.835
Time:	20:45:21	BIC	1288.336
Sample:	02-29-1980	HQIC	1281.509
	- 12-31-1990		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.4885	0.085	-5.742	0.000	-0.655	-0.322
ma.L1.D.Rose	-0.7601	0.101	-7.499	0.000	-0.959	-0.561
ma.L2.D.Rose	-0.2398	0.095	-2.518	0.012	-0.427	-0.053

Roots

	Real	Imaginary	Modulus	Frequency
MA.1	1.0001	+0.0000j	1.0001	0.0000
MA.2	-4.1696	+0.0000j	4.1696	0.5000

Predict on the Test Set using this model and evaluate the model.

	Test RMSE
ARIMA(0,1,2)	15.63

Performance of Models so far:

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
ARIMA(0,1,2)	15.63

Still this point Triple exponential has performed the best.

Automated Version of SARIMA

In Model SARIMA, we are considering Seasonal P,D,Q,S.

It is extension of ARIMA model by considering Seasonality factor.

```
Examples of some parameter combinations for Model...
Model: (0, 1, 1)(0, 0, 1, 12)
Model: (0, 1, 2)(0, 0, 2, 12)
Model: (1, 1, 0)(1, 0, 0, 12)
Model: (1, 1, 1)(1, 0, 1, 12)
Model: (1, 1, 2)(1, 0, 2, 12)
Model: (2, 1, 0)(2, 0, 0, 12)
Model: (2, 1, 1)(2, 0, 1, 12)
Model: (2, 1, 2)(2, 0, 2, 12)
```

Top 5 Parameters AIC score:

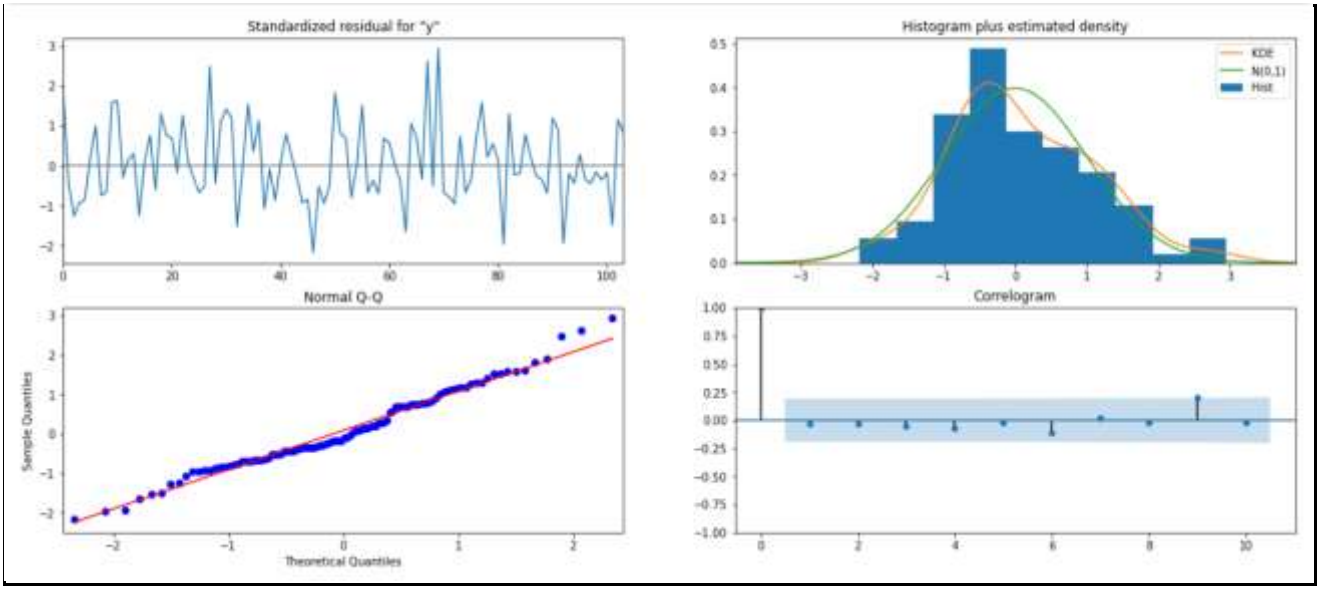
	param	seasonal	AIC
26	(0, 1, 2)	(2, 0, 2, 12)	887.94
80	(2, 1, 2)	(2, 0, 2, 12)	890.67
69	(2, 1, 1)	(2, 0, 0, 12)	896.52
78	(2, 1, 2)	(2, 0, 0, 12)	897.35
70	(2, 1, 1)	(2, 0, 1, 12)	897.64

From the above result we will choose (0,1,2)(2,0,2,12).

```
                                SARIMAX Results
==========================================================================================
Dep. Variable:                                   y   No. Observations:                  132
Model:             SARIMAX(0, 1, 2)x(2, 0, 2, 12)   Log Likelihood                -436.969
Date:                            Tue, 22 Jun 2021   AIC                             887.938
Time:                                    20:46:08   BIC                             906.448
Sample:                                         0   HQIC                            895.437
                                              - 132
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ma.L1         -0.8428    174.455     -0.005      0.996    -342.767     341.082
ma.L2         -0.1572     27.458     -0.006      0.995     -53.973      53.659
ar.S.L12       0.3467      0.079      4.374      0.000       0.191       0.502
ar.S.L24       0.3023      0.076      3.996      0.000       0.154       0.451
ma.S.L12       0.0767      0.133      0.577      0.564      -0.184       0.337
ma.S.L24      -0.0726      0.146     -0.498      0.618      -0.358       0.213
sigma2       251.3159   4.38e+04      0.006      0.995   -8.57e+04    8.62e+04
===================================================================================
Ljung-Box (L1) (Q):                   0.10   Jarque-Bera (JB):                 2.33
Prob(Q):                              0.75   Prob(JB):                         0.31
Heteroskedasticity (H):               0.88   Skew:                             0.37
Prob(H) (two-sided):                  0.70   Kurtosis:                         3.03
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

Results:



Predict on the Test Set using this model and evaluate the model.

y	mean	mean_se	mean_ci_lower	mean_ci_upper
0	62.87	15.93	31.65	94.09
1	70.54	16.15	38.89	102.19
2	77.36	16.15	45.71	109.01
3	76.21	16.15	44.56	107.86
4	72.75	16.15	41.10	104.40

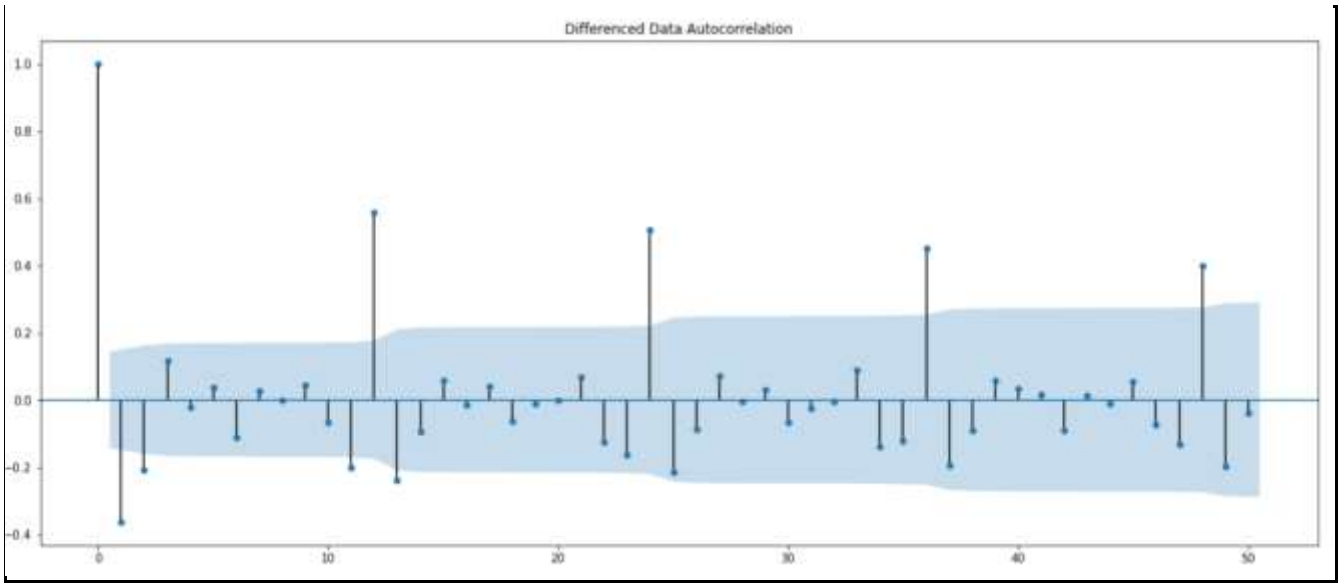
RMSE for SARIMA model is:

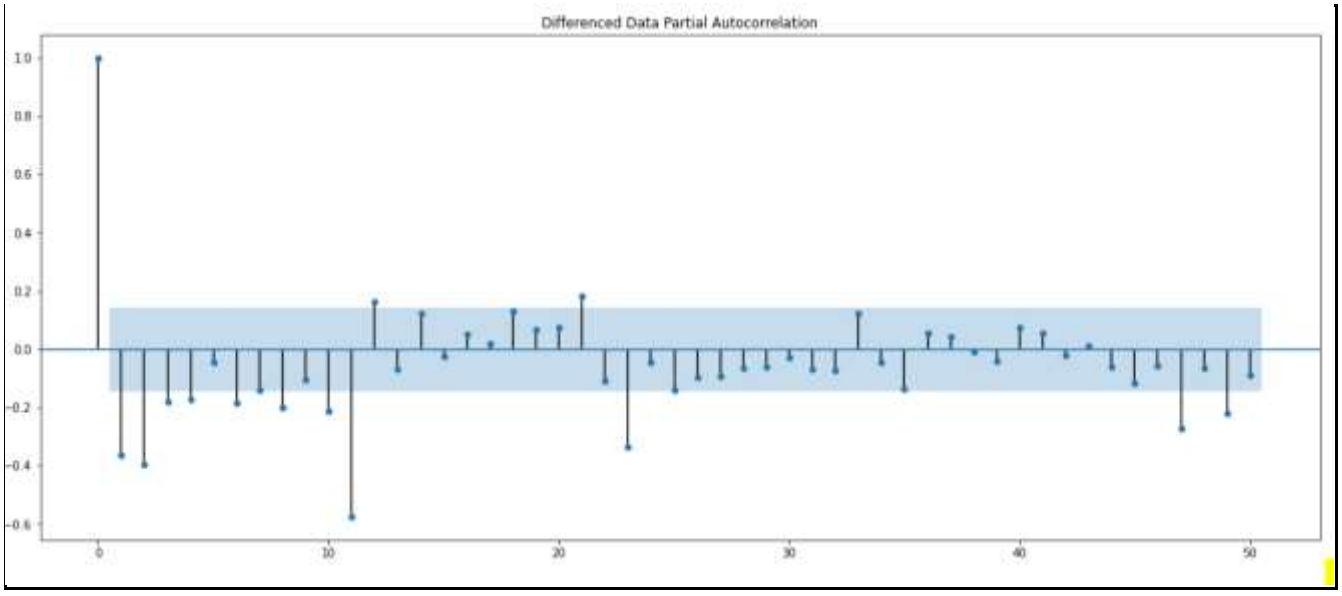
	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
ARIMA(0,1,2)	15.63
SARIMA(0,1,2)(2,0,2,12)	26.95

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

Manual ARIMA

Let us look at the ACF and the PACF plots





Here, we have taken alpha=0.05.

The Auto-Regressive parameter in an ARIMA model is 'p' which comes from the significant lag before which the PACF plot cuts-off to 0. The Moving-Average parameter in an ARIMA model is 'q' which comes from the significant lag before the ACF plot cuts-off to 0. By looking at the above plots, we can say that both the PACF and ACF plot cuts-off at lag 4 and 2.

ARIMA Model Results

Dep. Variable:	D.Rose	No. Observations:	131
Model:	ARIMA(4, 1, 2)	Log Likelihood	-633.876
Method:	css-mle	S.D. of innovations	29.793
Date:	Tue, 22 Jun 2021	AIC	1283.753
Time:	20:46:10	BIC	1306.754
Sample:	02-29-1980 - 12-31-1990	HQIC	1293.099

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.1905	0.576	-0.331	0.741	-1.319	0.938
ar.L1.D.Rose	1.1685	0.087	13.391	0.000	0.997	1.340
ar.L2.D.Rose	-0.3562	0.132	-2.693	0.007	-0.616	-0.097
ar.L3.D.Rose	0.1855	0.132	1.402	0.161	-0.074	0.445
ar.L4.D.Rose	-0.2227	0.091	-2.443	0.015	-0.401	-0.044
ma.L1.D.Rose	-1.9506	nan	nan	nan	nan	nan
ma.L2.D.Rose	1.0000	nan	nan	nan	nan	nan

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	1.1027	-0.4116j	1.1770	-0.0569
AR.2	1.1027	+0.4116j	1.1770	0.0569
AR.3	-0.6863	-1.6643j	1.8003	-0.3122
AR.4	-0.6863	+1.6643j	1.8003	0.3122
MA.1	0.9753	-0.2209j	1.0000	-0.0355
MA.2	0.9753	+0.2209j	1.0000	0.0355

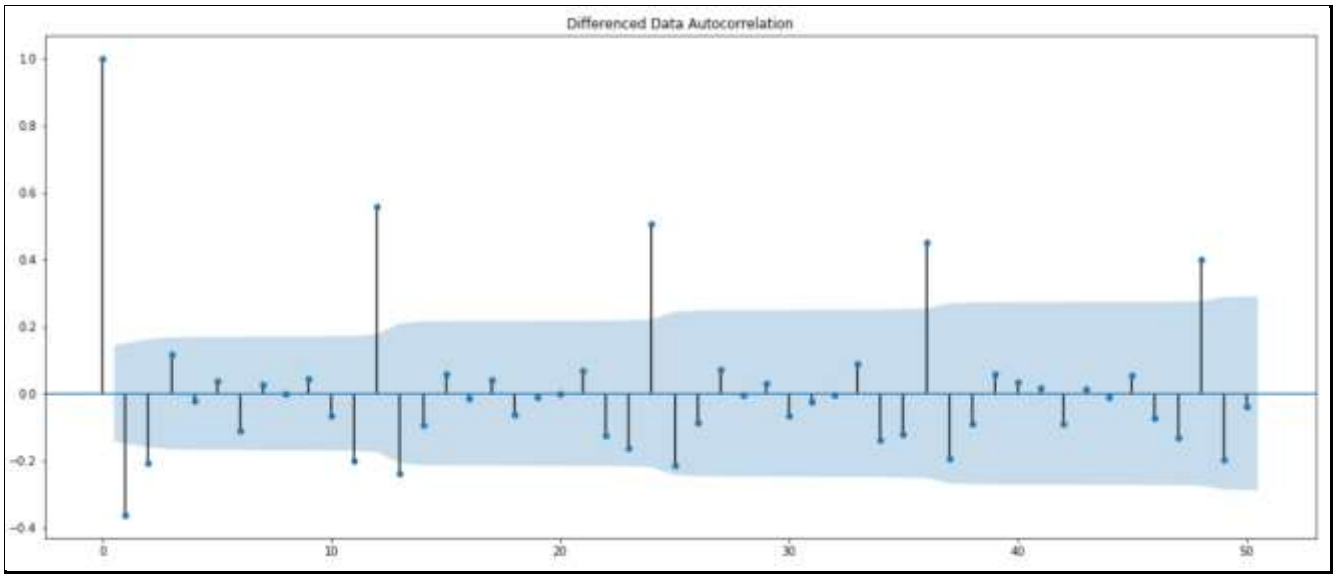
Predict on the Test Set using this model and evaluate the model.

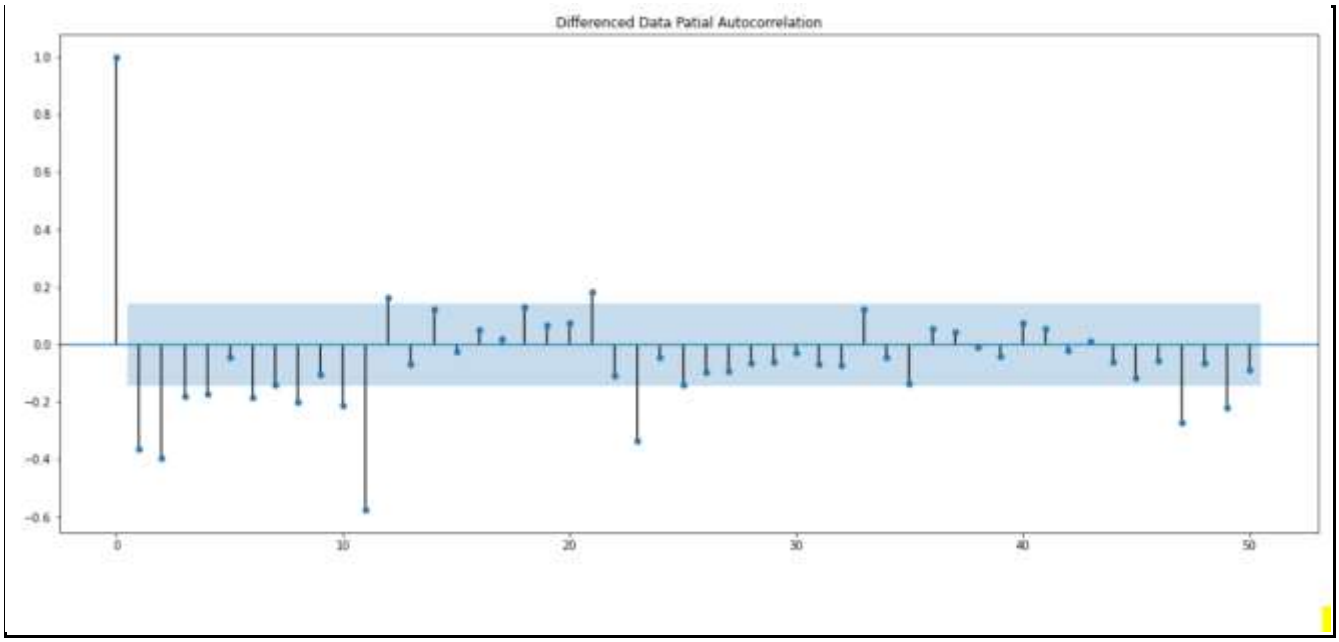
RMSE for Manual ARIMA model is 33.969200767905 53.

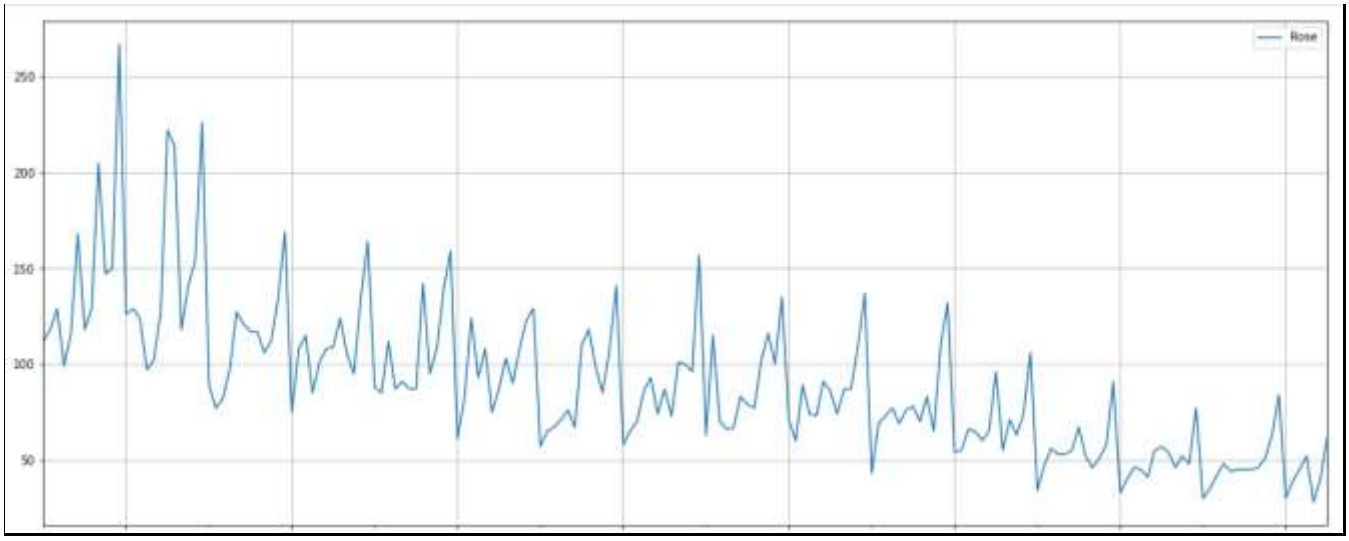
The Overall Results till now,

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
ARIMA(0,1,2)	15.63
SARIMA(0,1,2)(2,0,2,12)	26.95
ARIMA(4,1,2)	33.97

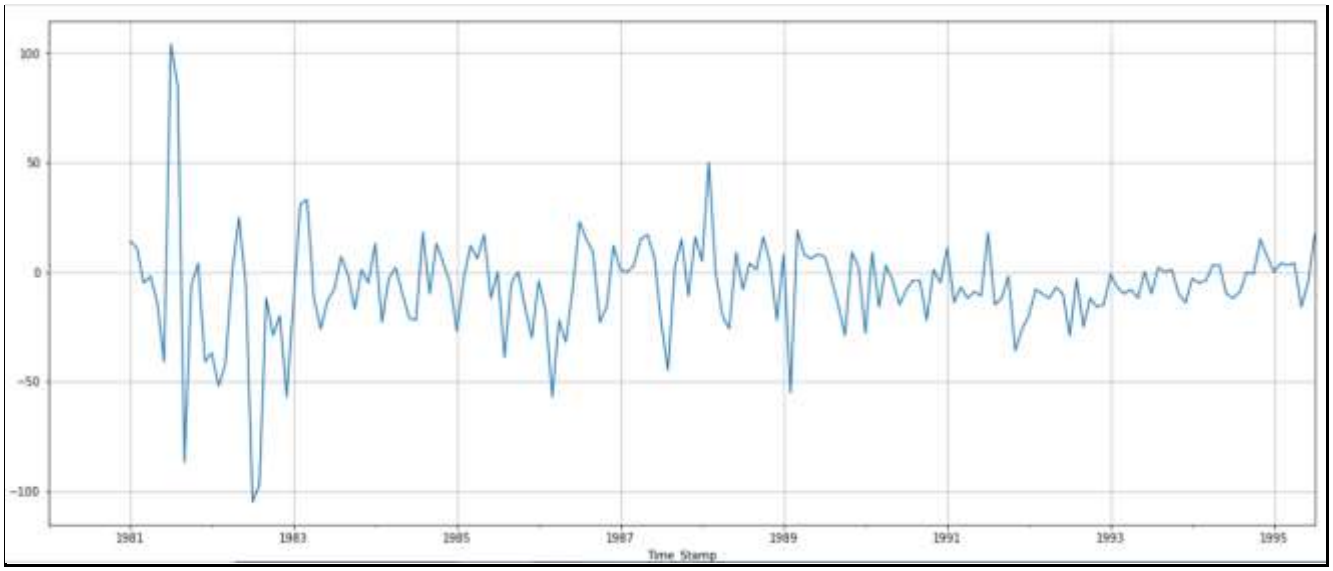
SARIMA Model : Manually looking at ACF and PACF



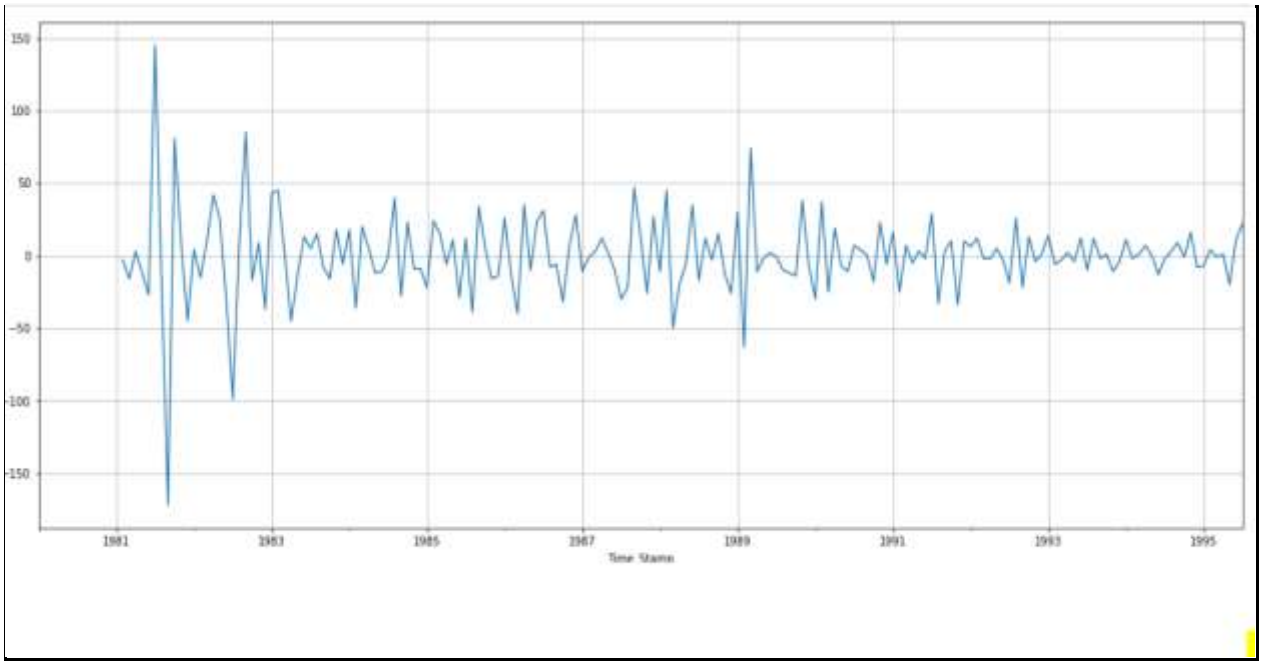




We see that there is a trend and a seasonality. So, now we take a seasonal differencing and check the series.

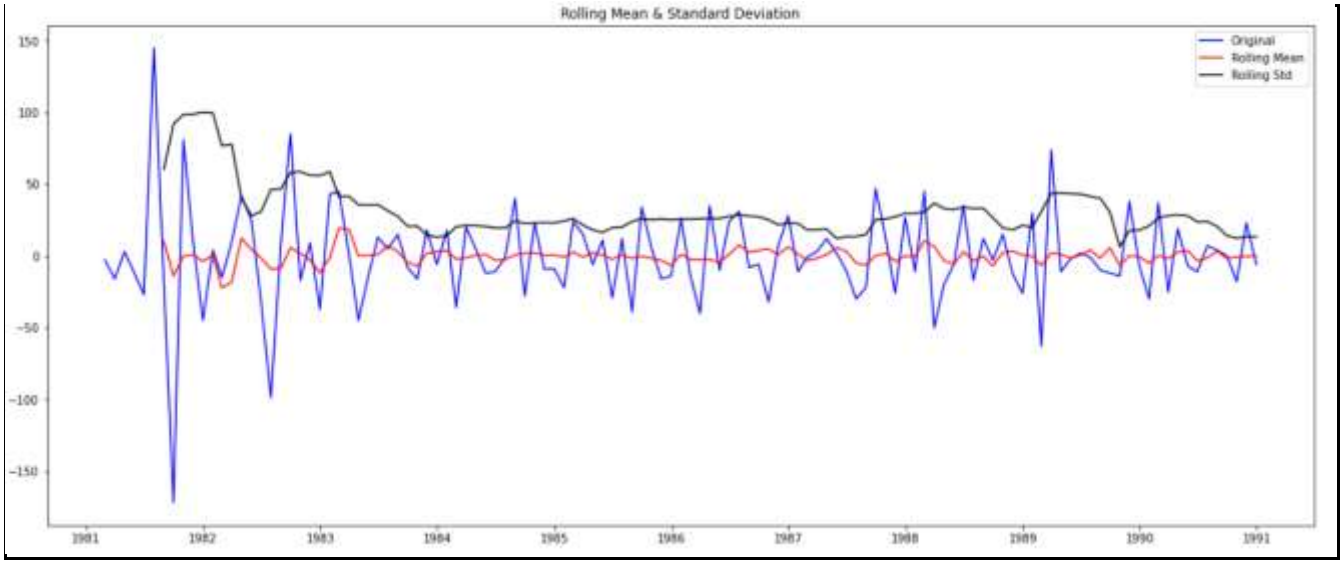


We see that there might be a slight trend which can be noticed in the data. So we take a differencing of first order on the seasonally differenced series.

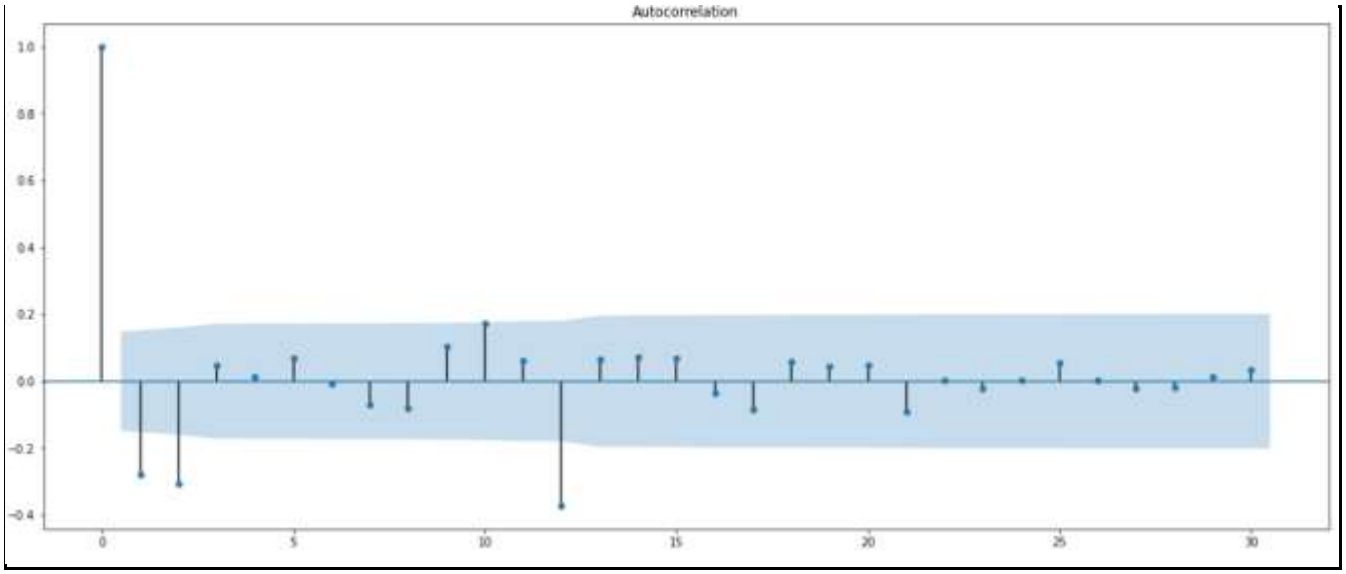


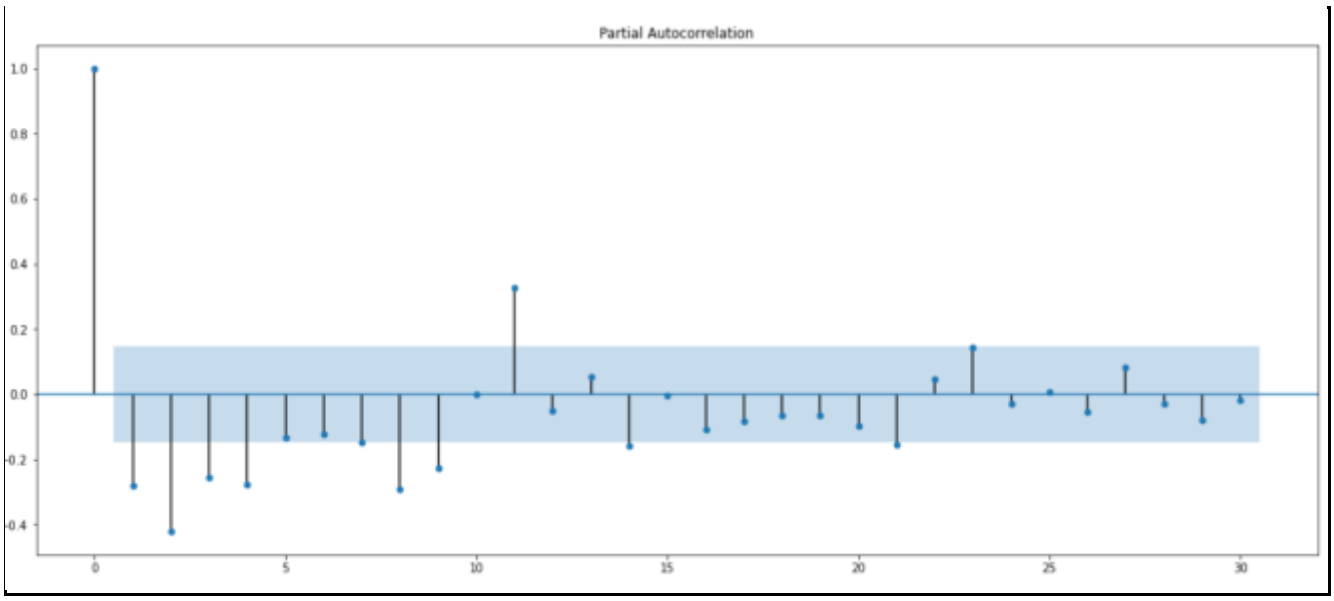
Now we see that there is almost no trend present in the data. Seasonality is only present in the data.

Let us go ahead and check the stationarity of the above series before fitting the SARIMA model.



```
Results of Dickey-Fuller Test:
Test Statistic                  -3.69
p-value                          0.00
#Lags Used                      11.00
Number of Observations Used    107.00
Critical Value (1%)             -3.49
Critical Value (5%)             -2.89
Critical Value (10%)            -2.58
dtype: float64
```





Here, we have taken alpha=0.05.

We are going to take the seasonal period as 12. We will keep the p(1) and q(1) parameters same as the ARIMA model.

The Auto-Regressive parameter in an SARIMA model is 'P' which comes from the significant lag after which the PACF plot cuts-off to 0. The Moving-Average parameter in an SARIMA model is 'q' which comes from the significant lag after which the ACF plot cuts-off to 0. Remember to check the ACF and the PACF plots only at multiples of 12 (since 12 is the seasonal period). By looking at the plots we see that the ACF and the PACF do not directly cut-off to 0.

This is a common problem while building models by looking at the ACF and the PACF plots. But we are able to explain the model.

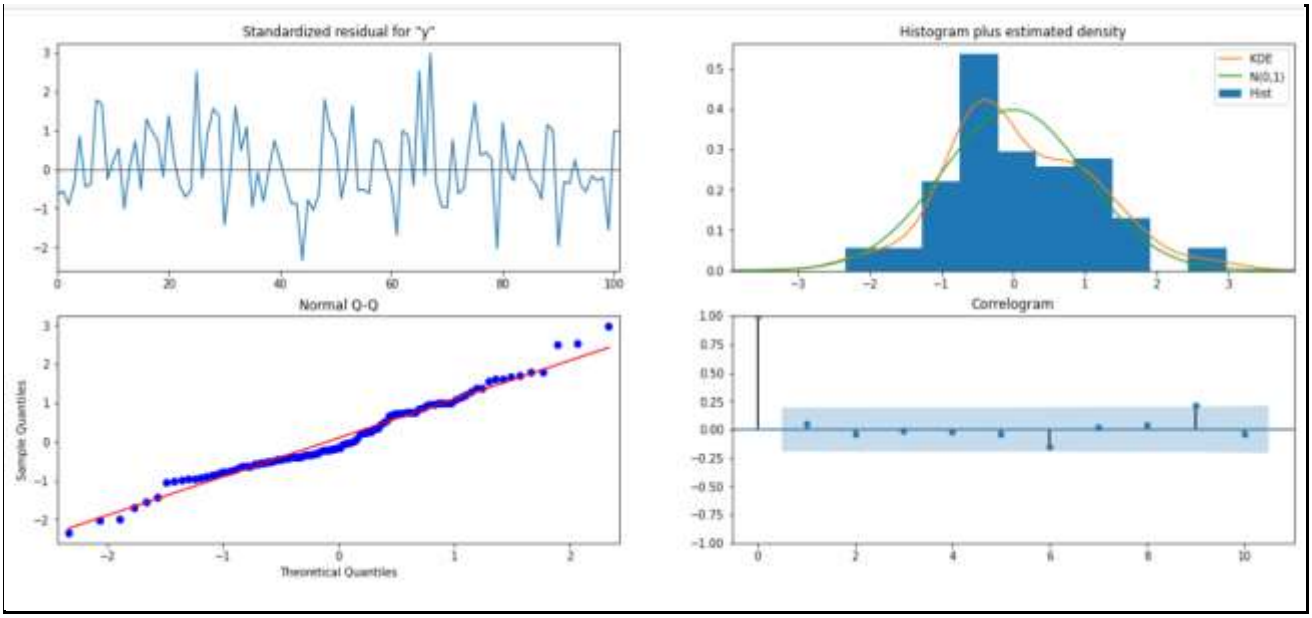
SARIMAX Results

Dep. Variable:	y	No. Observations:	132
Model:	SARIMAX(2, 1, 4)x(2, 0, [1, 2], 12)	Log Likelihood	-426.513
Date:	Tue, 22 Jun 2021	AIC	875.026
Time:	20:46:28	BIC	903.901
Sample:	0	HQIC	886.718
	- 132		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.5879	0.319	1.844	0.065	-0.037	1.213
ar.L2	-0.6237	0.326	-1.911	0.056	-1.263	0.016
ma.L1	-403.5287	96.174	-4.196	0.000	-592.027	-215.031
ma.L2	608.0760	85.220	7.135	0.000	441.048	775.104
ma.L3	-450.2434	112.062	-4.018	0.000	-669.881	-230.606
ma.L4	242.3857	114.697	2.113	0.035	17.583	467.189
ar.S.L12	0.3954	0.144	2.754	0.006	0.114	0.677
ar.S.L24	0.2686	0.102	2.626	0.009	0.068	0.469
ma.S.L12	0.0596	0.185	0.322	0.747	-0.303	0.422
ma.S.L24	-0.0566	0.143	-0.397	0.691	-0.336	0.223
sigma2	0.0015	0.001	1.993	0.046	2.5e-05	0.003

Ljung-Box (L1) (Q):	0.22	Jarque-Bera (JB):	1.88
Prob(Q):	0.64	Prob(JB):	0.39
Heteroskedasticity (H):	0.76	Skew:	0.33
Prob(H) (two-sided):	0.43	Kurtosis:	3.11

Results Dignostics for Manual SARIMA:



Predict on the Test Set using this model and evaluate the model.

	y	mean	mean_se	mean_ci_lower	mean_ci_upper
0		61.66	15.59	31.10	92.22
1		70.96	15.64	40.30	101.62
2		76.12	15.65	45.44	106.79
3		75.26	15.65	44.57	105.94
4		72.43	15.66	41.74	103.12

RMSE for Manual SARIMA is **26.696148760797275**

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
ARIMA(0,1,2)	15.63
SARIMA(0,1,2)(2,0,2,12)	26.95
ARIMA(4,1,2)	33.97
SARIMA(2,1,4)(2,0,2,12)	26.70

8.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.

	Test RMSE
Alpha=0.995,Simple Exponential Model	36.82
Alpha=0.3,SimpleExponentialSmoothing	47.53
Alpha=0.3,Beta=0.3,DoubleExponentialSmoothing	265.59
Alpha=0.676,Beta=0.088,Gamma=0.323,TripleExponentialSmoothing	21.17
Alpha=0.3,Beta=0.4,Gamma=0.3,TripleExponentialSmoothing	10.95
RegressionOnTime	15.28
NaiveModel	79.74
SimpleAverageModel	53.48
2pointTrailingMovingAverage	11.53
4pointTrailingMovingAverage	14.46
6pointTrailingMovingAverage	14.57
9pointTrailingMovingAverage	14.73
ARIMA(0,1,2)	15.63
SARIMA(0,1,2)(2,0,2,12)	26.95
ARIMA(4,1,2)	33.97
SARIMA(2,1,4)(2,0,2,12)	26.70

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

If we see all the RMSE, We can Say that Triple exponential Smoothing can perform Well for this series, This series has level, Seasonality and Trend also.

So we will Select Triple Exponential Smoothing model, we will give train data as Whole Data. That is, Test and Train data earlier, Now all data will be used for Training for Purpose.

We are predicting here for Next 12 Months.

Parameters:

```
{'smoothing_level': 0.10670810408409581,
 'smoothing_trend': 4.386089535696353e-08,
 'smoothing_seasonal': 4.6406369545555414e-05,
 'damping_trend': nan,
 'initial_level': 44.80878821492276,
 'initial_trend': -0.17017928788121756,
 'initial_seasons': array([2.40950781, 2.73642394, 3.01847615, 2.69736542, 2.94860714,
        3.1936259 , 3.58257845, 3.6914497 , 3.48628986, 3.44546151,
        3.99944747, 5.51335091]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

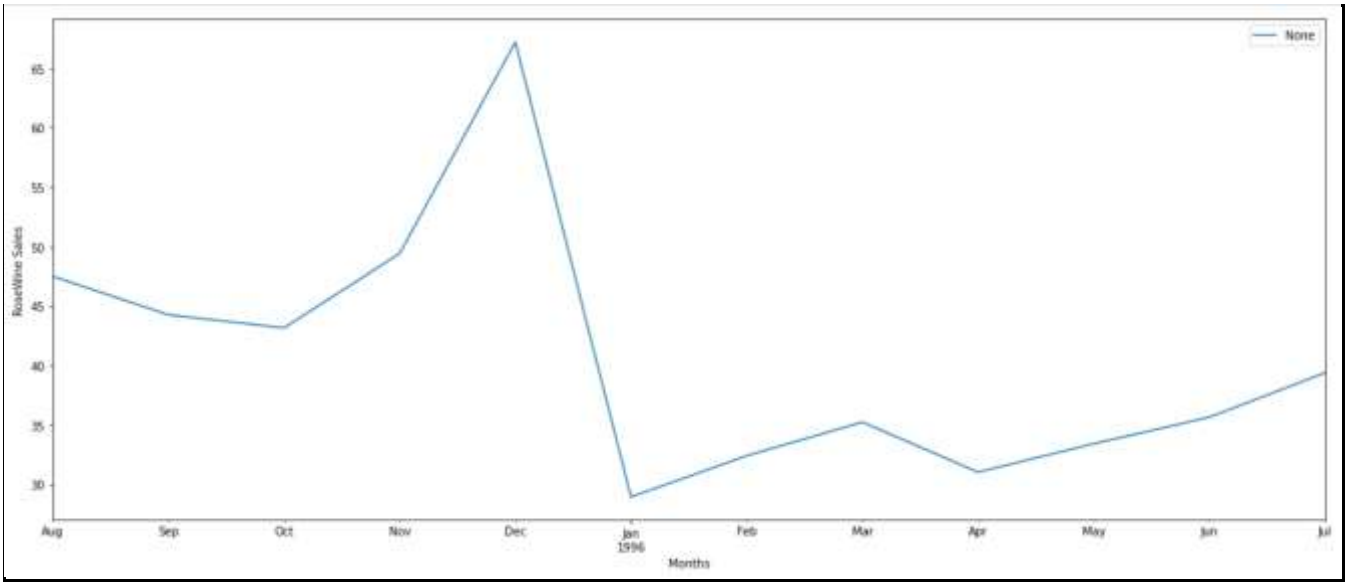
Final Model results:

ExponentialSmoothing Model Results

Dep. Variable:	Rose	**No. Observations:**	187
Model:	ExponentialSmoothing	**SSE**	48456.867
Optimized:	True	**AIC**	1071.219
Trend:	Additive	**BIC**	1122.917
Seasonal:	Multiplicative	**AICC**	1075.290
Seasonal Periods:	12	**Date:**	Tue, 22 Jun 2021
Box-Cox:	False	**Time:**	22:05:00
Box-Cox Coeff.:	None		

	coeff	code	optimized
smoothing_level	0.1067081	alpha	True
smoothing_trend	4.3861e-08	beta	True
smoothing_seasonal	4.6406e-05	gamma	True
initial_level	44.808788	l.0	True
initial_trend	-0.1701793	b.0	True
initial_seasons.0	2.4095078	s.0	True
initial_seasons.1	2.7364239	s.1	True
initial_seasons.2	3.0184761	s.2	True
initial_seasons.3	2.6973654	s.3	True
initial_seasons.4	2.9486071	s.4	True
initial_seasons.5	3.1936259	s.5	True
initial_seasons.6	3.5825785	s.6	True
initial_seasons.7	3.6914497	s.7	True
initial_seasons.8	3.4862899	s.8	True
initial_seasons.9	3.4454615	s.9	True
initial_seasons.10	3.9994475	s.10	True
initial_seasons.11	5.5133509	s.11	True

Sales for the Next 12 Months,



Rose wine Sales and Particular Month :

	Sales
1995-08-31	47.50
1995-09-30	44.27
1995-10-31	43.16
1995-11-30	49.42
1995-12-31	67.19
1996-01-31	28.95
1996-02-29	32.42
1996-03-31	35.25
1996-04-30	31.04
1996-05-31	33.43
1996-06-30	35.66
1996-07-31	39.39

10.Comment on the model thus built and report your findings and suggest the measures that the company should be taking for future sales. Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present. 5

To predict Sales in 12 Months we have studied, Analysed and applied different Models on the existing Data.

While Doing the EDA part we got to know that

1.There is downward trend, and also series has Seasonality, Trend and Level.

2.There are two missing values, we have filled that with the forward filling method.

3.The sales are High in the month of December, it is might due to Festivals like New year, Christmas.

4.The High Stock should be Ready from the month of August to December.

5.As per Predicted, The Maximum sales are happening in the month of December, and it is around 67, so it will be okay if we got 67 wines in stock in The Dec-1995.

6.Here we come with interseting insights from Monthly plot, From January to October Median of Sales is almost Same. Only Sales are increasing in November and December.In This December Only Sales has Crossed figure of 250 wines.

7.It needs to be analysed further, Why the Sales are Decreasing every passing year. Quality issue, branding, Advertisement issue that can be analysed from different data.

To Predict the Results, We have applied Double,Triple ExponentialSmoothing.Also applied Regression, NaiveModel, SimpleAverageModel, TrailingMovingAverage, Then automated ARIMA, SARIMA by comparing AIC values and Manual ARIMA, SARIMA by Partial Autocorrelation Function and Auto-correlation Function plots.

After Applying all the Models, we found that RMSE value for Triple exponential smoothing was a lowest, so we have decided to go with this model. We have applied the whole Data earlier it was Test and Train, we have applied full data, fit the model and then predicted the Next 12 Months Sales and plotted the same.

THE END