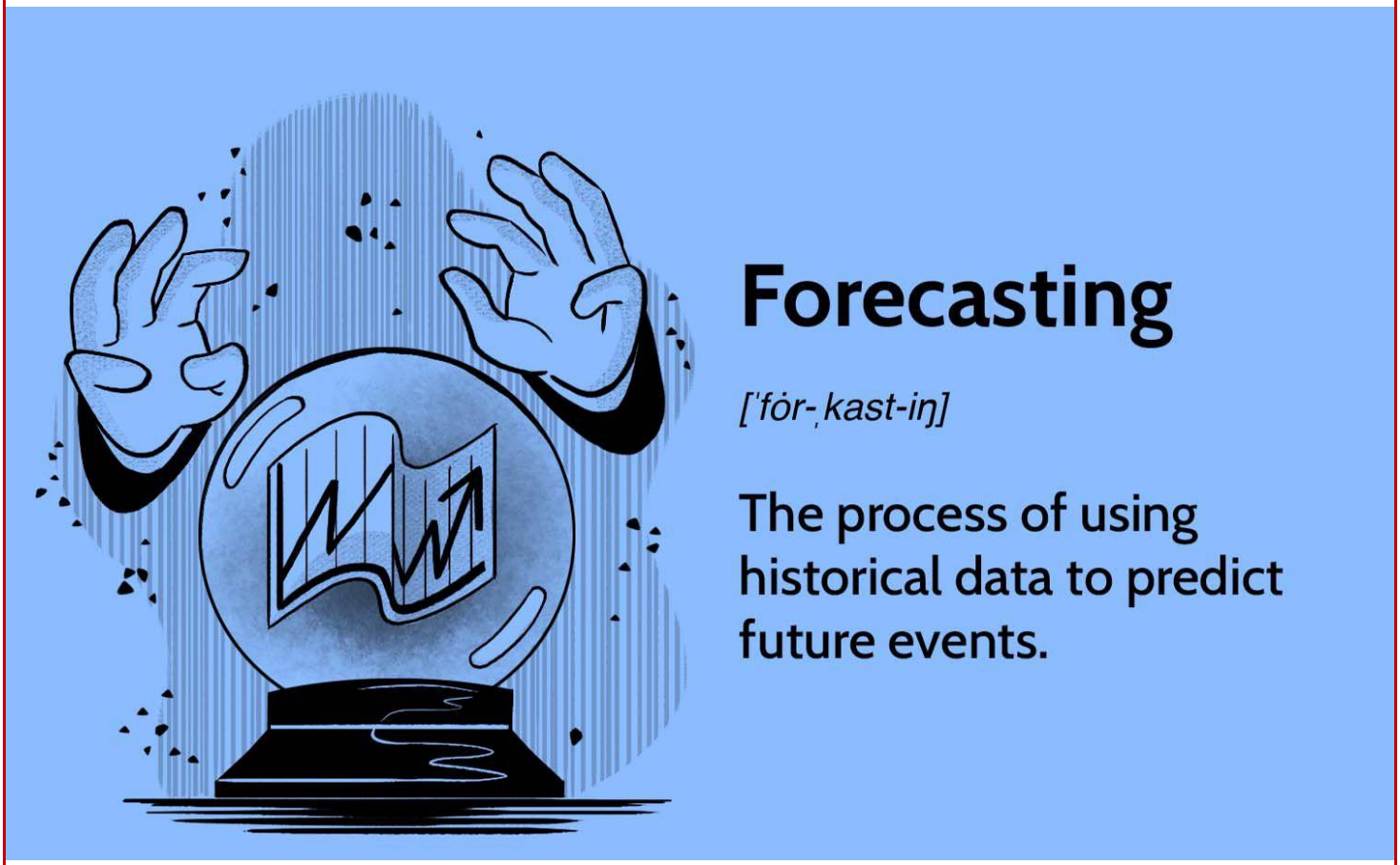
TIME SERIES FORECASTING – CODED PROJECT (Sparkling.csv)



By

R. SUKANYA

CONTENT

LIST OF FIGURES

LIST OF TABLES

Scoring guide (Rubric) - PM Project Rubric

Criteria	Points
Define the problem and perform Exploratory Data Analysis Read the data as an appropriate time series data - Plot the data - Perform EDA - Perform Decomposition	**9**
Data Pre-processing Missing value treatment - Visualize the processed data - Train-test split	**4**
Model Building - Original Data Build forecasting models - Linear regression - Simple Average - Moving Average - Exponential Models (Single, Double, Triple) - Check the performance of the models built	**15**
Check for Stationarity Check for stationarity - Make the data stationary (if needed)	**4**
Model Building - Stationary Data Generate ACF & PACF Plot and find the AR, MA values. - Build different ARIMA models - Auto ARIMA - Manual ARIMA - Build different SARIMA models - Auto SARIMA - Manual SARIMA - Check the performance of the models built	**12**
Compare the performance of the models Compare the performance of all the models built - Choose the best model with proper rationale - Rebuild the best model using the entire data - Make a forecast for the next 12 months	**6**
Actionable Insights & Recommendations Conclude with the key takeaways (actionable insights and recommendations) for the business	**4**
Business Report Quality Adhere to the business report checklist	**6**
TOTAL	**60**

1. Problem Statement – SPARKLING.CSV

Context

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

Objective

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

1.1 Define the problem and perform EDA (Exploratory Data Analysis)

1.1.1 Importing the libraries

Import the necessary libraries for data processing, data visualization, modelling, perform decomposition, build logistic regression model and exponential smoothing models, build ACF and PACF plots, build ARIMA model and to check the model performance.

1.1.2 Reading and loading the dataset Sparkling.csv:

Load and read the dataset using the read_csv() function of pandas. The first five lines of the dataset is as follows:

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

Table 1.1: Reading the first 5 rows of the dataset

1.1.3 Read the data as an appropriate time series data:

Create a dataframe of date column ranging in the period from 31/01/1980 to 31/07/1995. Add this column to the original dataframe as a Time_Stamp column.

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

Table 1.2: Adding the Time_Stamp for making it a timeseries data

1.1.4 Plot the time series data:

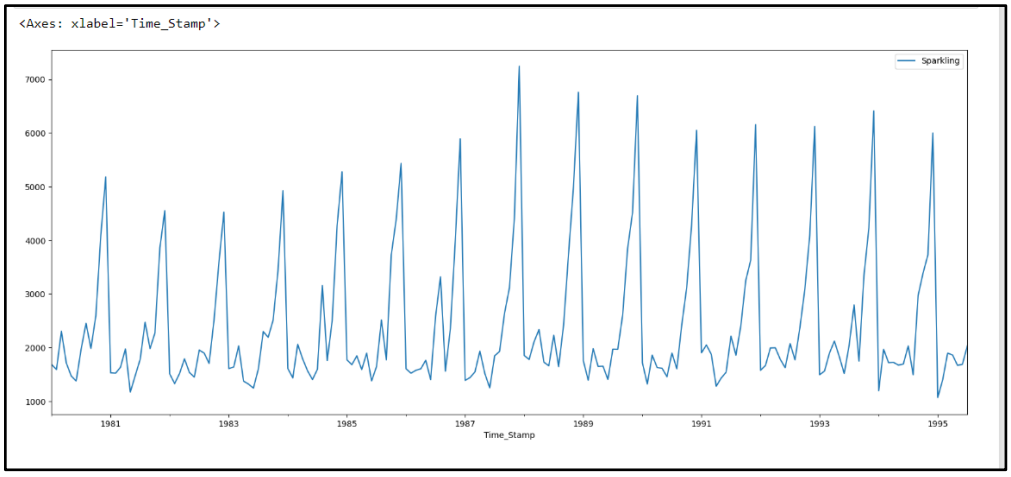


Table 1.3: Time Series data

Insights:

* From the time series plot, it is clearly seen that the data shows seasonality but no trend.

* The maximum sale of Sparkling wine goes more than 7000 in 1988.

1.1.5 Perform EDA

1.1.5.1 Check the datatype of the columns.

Use the info() command to check for the datatypes.

The database has 2 columns namely, YearMonth which is an object datatype and Sparkling column which is int64 datatype. Both the columns have non-null values.

1.1.5.2 Check the shape of the dataset.

Use the shape() command to check the number of rows and columns in the dataset.

```
The dataset has  187 rows and  2 columns.
```

1.1.5.3 Check the descriptive statistics.

Use the describe() function to get the statistical summary of the columns.

	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 1.4: Statistical summary of numerical type column

Insights:

- The basic measures of descriptive statistics tell us how the sale of wine has varied across years.
- This measure of descriptive statistics has been averaged over the whole data without taking the time component into account.
- The maximum sales of wine over the years is 7242 while the minimum is 1070.
- The mean sale of wine over all the years from 1980 to 1995 is 2402 while the median sale is 1874.

1.1.5.4 Plot a boxplot to understand the sales of wine across different years and within different months across years.

1.1.5.4.1 Yearly boxplot

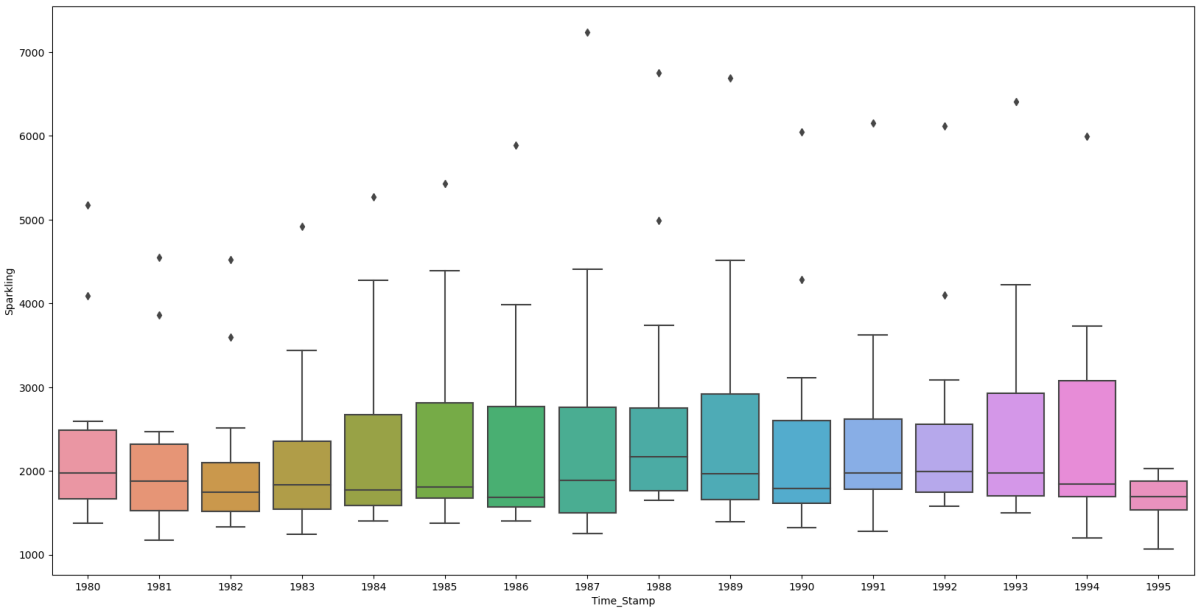


Fig 1.1 Yearly Boxplot of the timeseries data

Insights:

- The yearly boxplot shows that the sale of Sparkling wine has decreased from 1980 to 1995.
- The maximum sale of wine has occured in 1988 while the minimum is in the year 1995.

1.1.5.4.2 Monthly boxplot

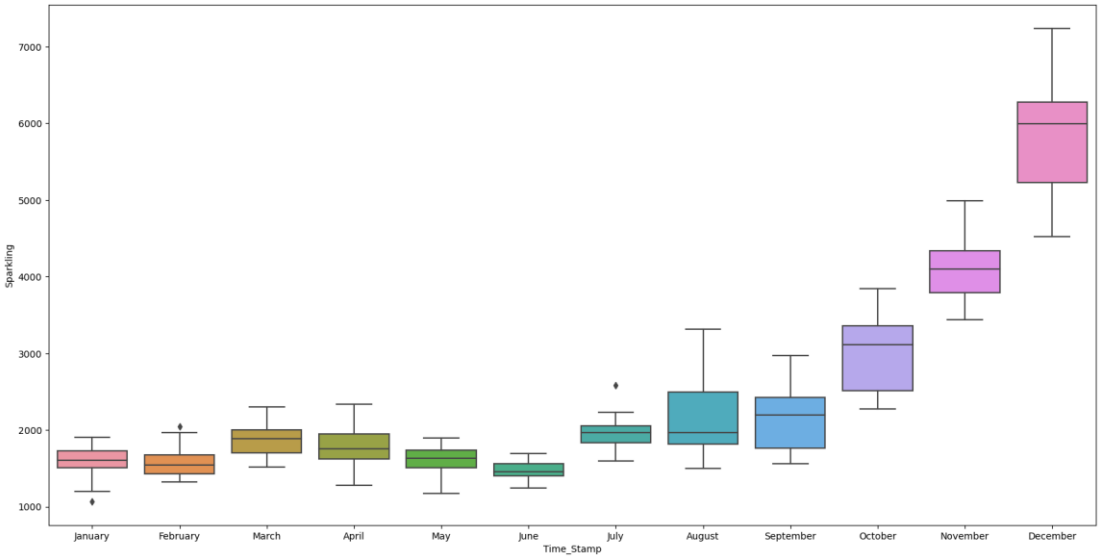


Fig 1.2 Monthly Boxplot of the timeseries data

Insights:

- The monthly boxplot shows an upward trend of sale of wine from January to December.
- We see the sale of wine rising in the months of September, October, November and December, probably due to winters and the festive season.

1.1.5.5 Monthly boxplot to show the sale of wine in different months with mean line

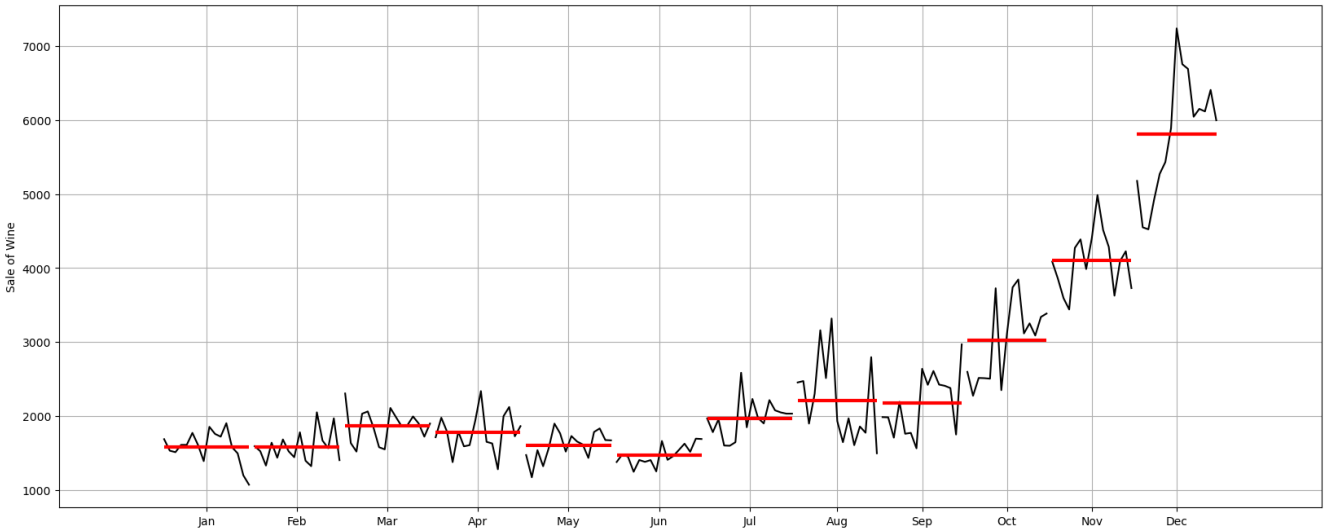


Fig 1.3 Monthly Boxplot with the Mean line

Insights:

This plot shows us the behaviour of the Time Series ('Sparkling Sales' in this case) across various months. The red line is the mean value. We see that the median sale of Sparkling wine is maximum in the month of December.

1.1.5.6 Plot a graph of monthly Sparkling Sales across years

Make a pivot table of the months vs different years.

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

Table 1.5 Pivot table of months vs years of sale of Sparkling wine

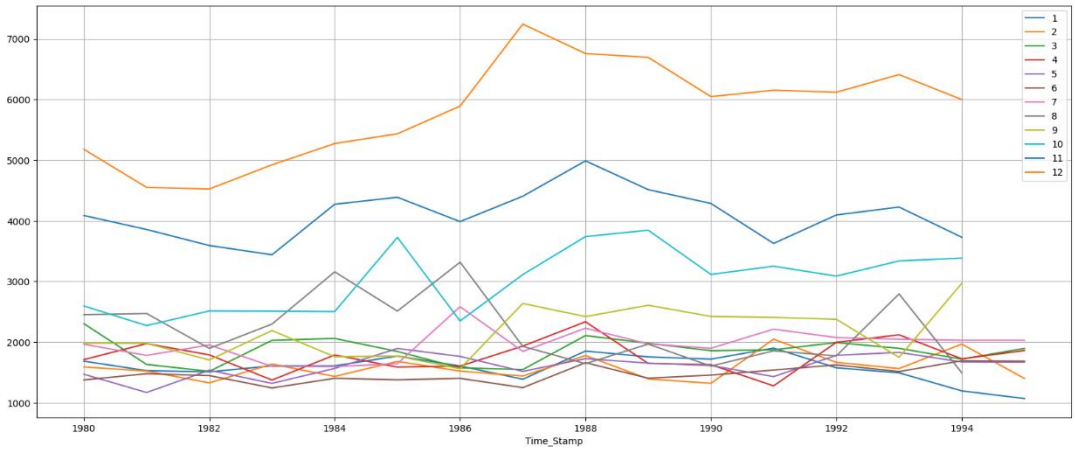


Fig 1.4 Monthly sale of Sparkling wine across years

Insights:

- The plot shows the trend of month across the years.
- The monthly sales across years shows a clear distinction between the winter months, while the summer months are mostly overlapping each other.
- We see that December month fares the best in sale of wines.
- We see that from the 8th month itself, that is the month of September, the sale of wines starts picking up.

1.1.5.7 Plot the Empirical Cumulative Distribution

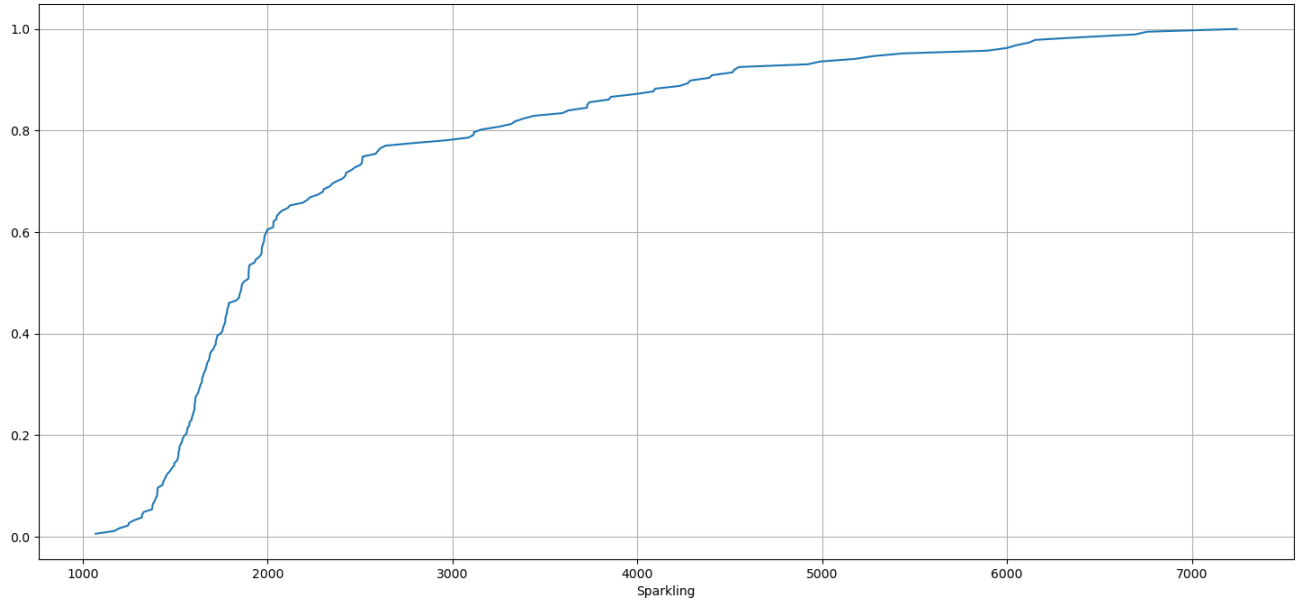


Fig 1.5 Empirical Cumulative Distribution of Sparkling Wine

1.1.6 Perform Decomposition:

1.1.6.1 Additive Decomposition

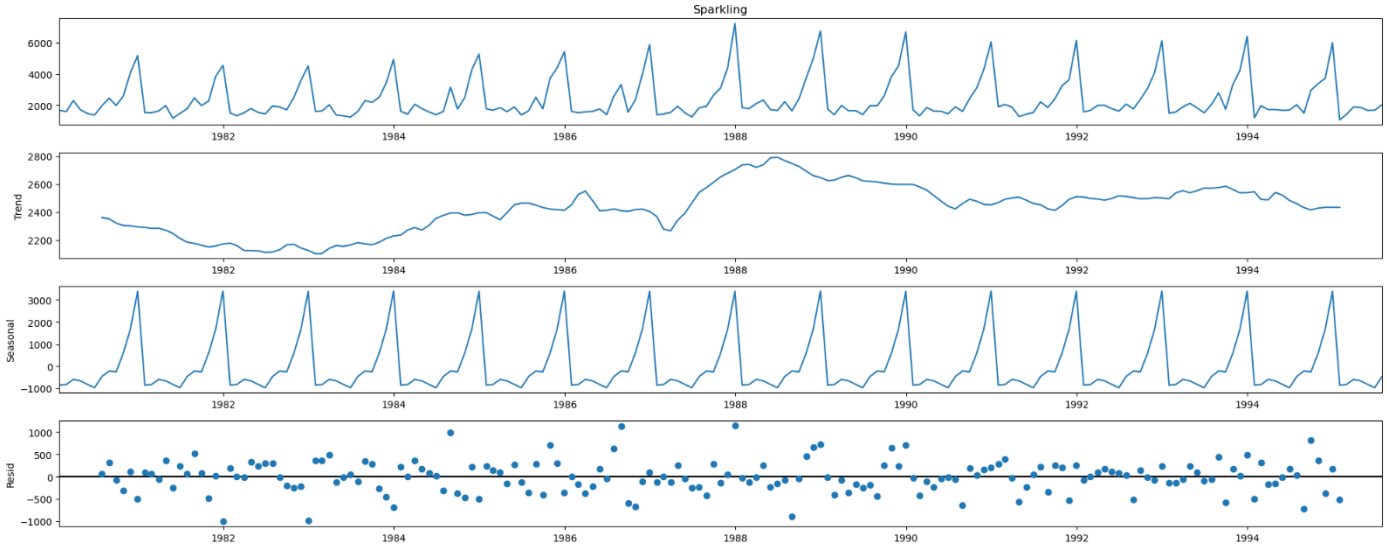


Fig 1.6 Additive Decomposition

1.6.2 Multiplicative Decomposition

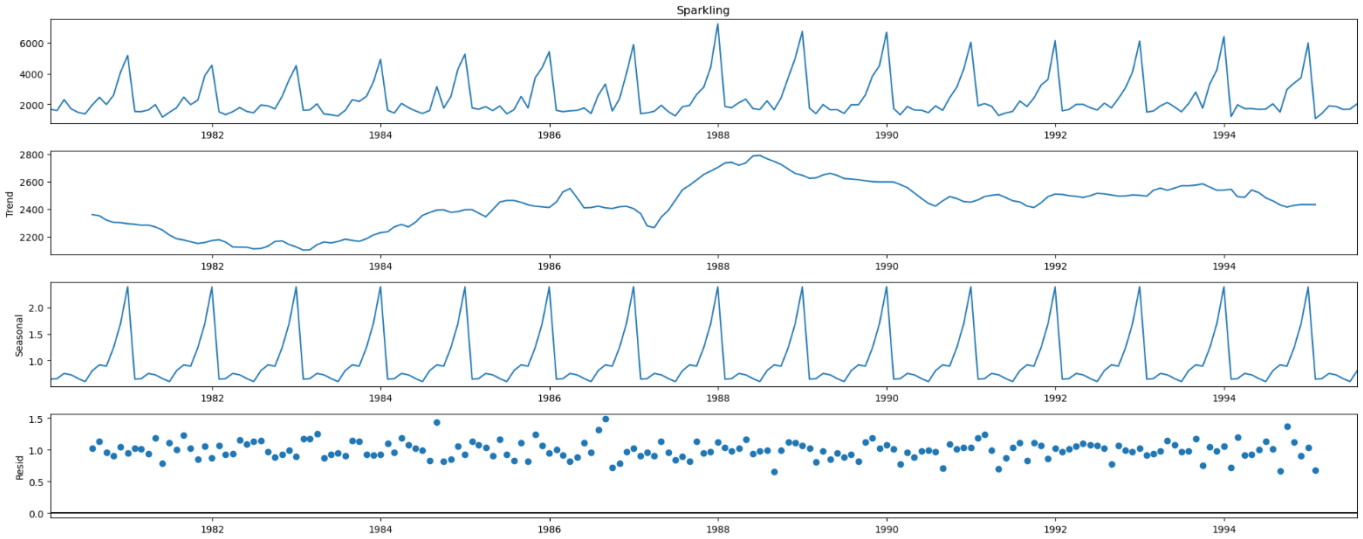


Fig 1.7 Multiplicative Decomposition

Insights:

After decomposition additively and multiplicatively and observing the trend, seasonality and residuals, we come to a conclusion that the residuals is almost uniform and flat in the multiplicative decomposition.

Hence, we go for multiplicative decomposition.

1.2 Data Preprocessing

1.2.1 Missing value Treatment:

Use the isnull() function to find if there are any missing values in the dataset. We can see that there are no missing values in the dataset.

1.2.2 Visualize the processed data

1.2.2.1 Plot the time series data

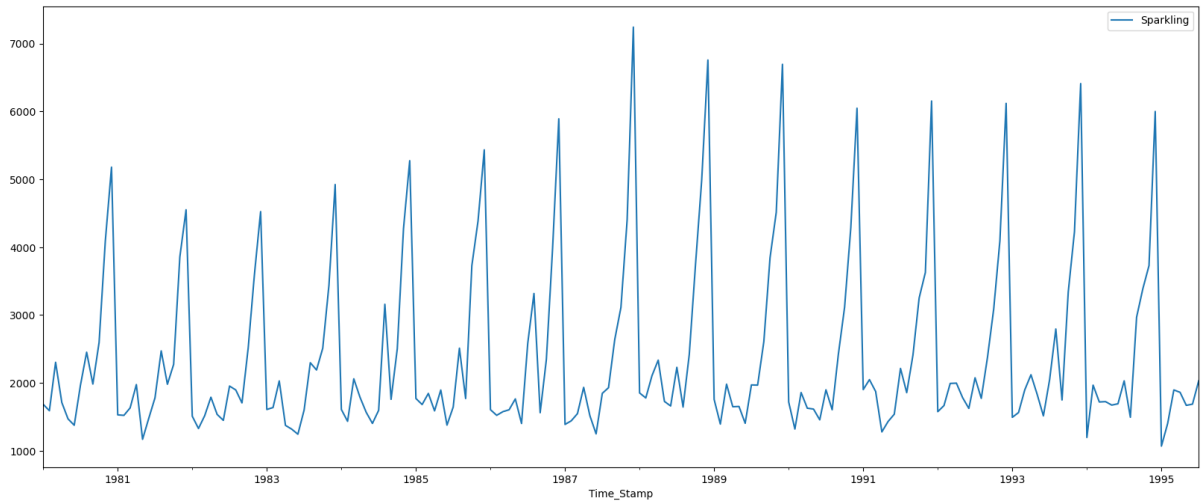


Fig 1.8 Plot of the time series data

1.2.2.2 Plot the average sale of wine per month and the month on month percentage change of sale of wine.

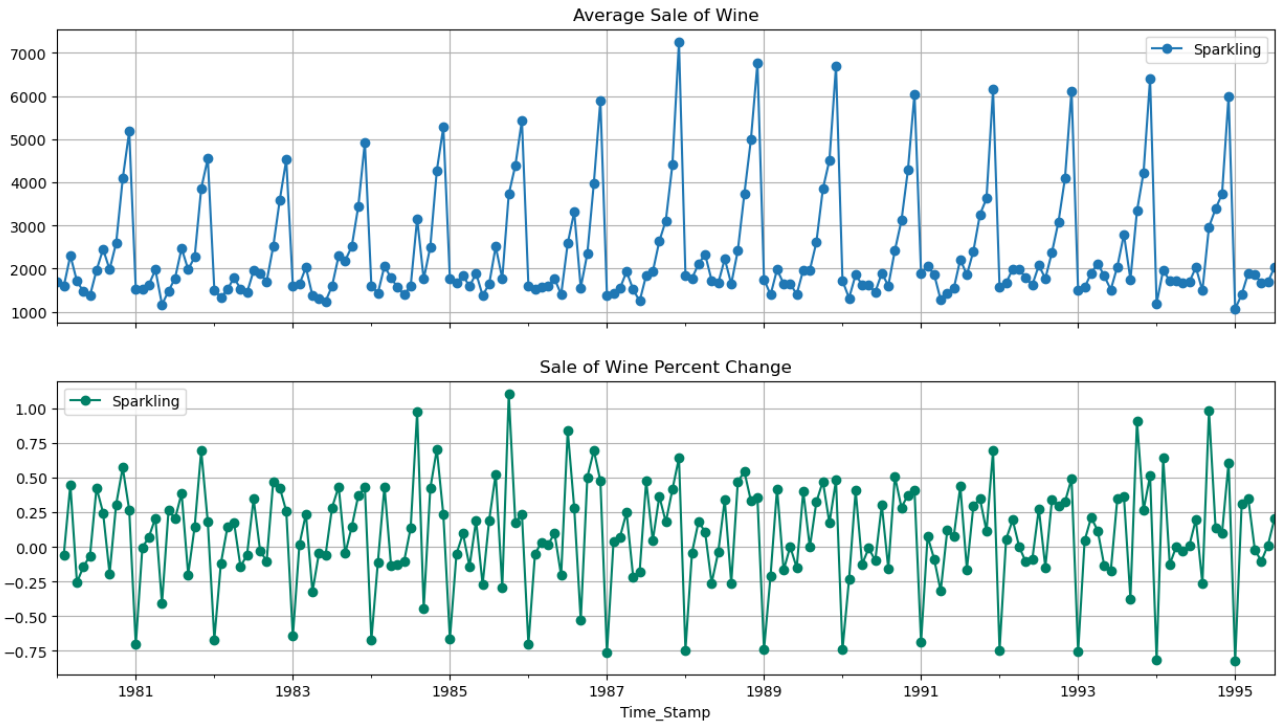


Fig 1.9: Average sale of wine per month and Percent change of sale of wine per month

1.2.2.3 Resampling the monthly data into a decade, quarterly and yearly format and comparing the Time Series plots

Decade plot

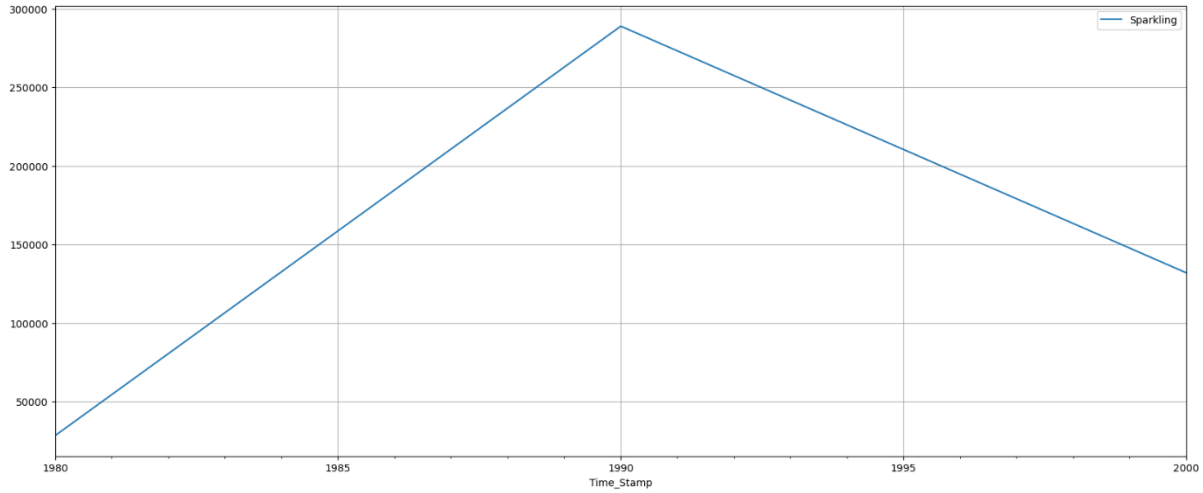


Fig 1.10: Decade Plot by resampling monthly data into a decade

Yearly plot

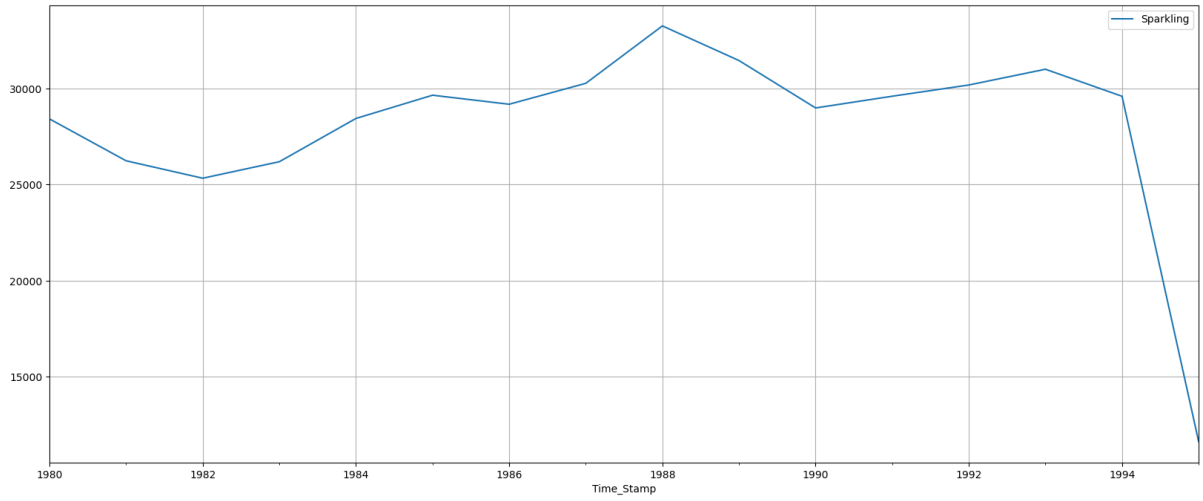


Fig 1.11: Yearly plot by resampling monthly data into years

Quarterly plot

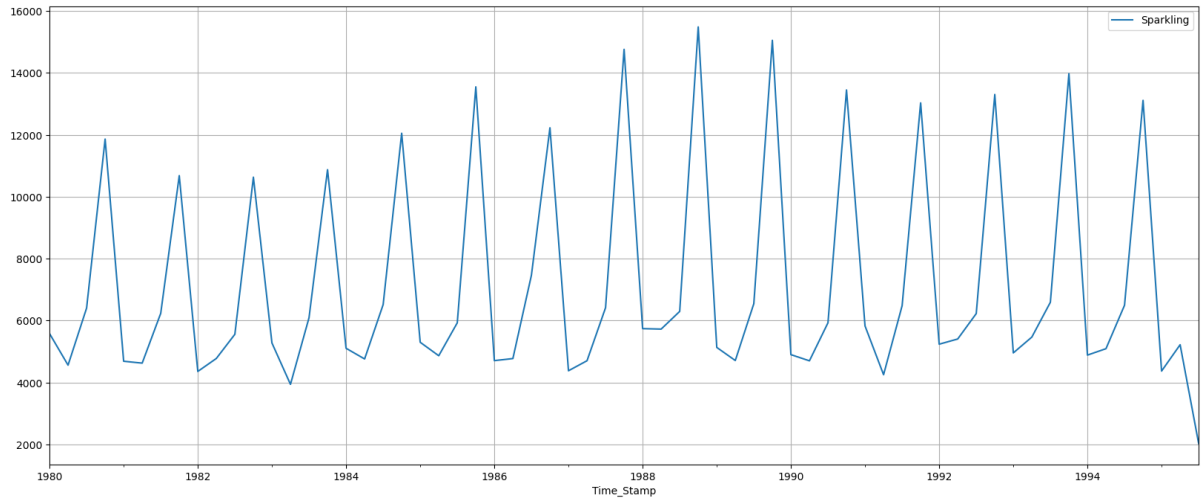


Fig 1.12: Quarterly plot by resampling monthly data into quarters

Insights:

On resampling, we observe that the decade plot and yearly plot are able to capture the trend of the sale of wine over the years, while the quarterly plot could capture only the seasonality.

1.2.3 Train-Test Split

We go for a train test split in the ratio of 75:25

First few rows of Training Data

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

Last few rows of Training Data

Time_Stamp	YearMonth	Sparkling
1991-04-30	1991-04	1279
1991-05-31	1991-05	1432
1991-06-30	1991-06	1540
1991-07-31	1991-07	2214
1991-08-31	1991-08	1857

Table 1.6: Train dataset

First few rows of Test Data

Time_Stamp	YearMonth	Sparkling
1991-09-30	1991-09	2408
1991-10-31	1991-10	3252
1991-11-30	1991-11	3627
1991-12-31	1991-12	6153
1992-01-31	1992-01	1577

Last few rows of Test Data

Time_Stamp	YearMonth	Sparkling
1995-03-31	1995-03	1897
1995-04-30	1995-04	1862
1995-05-31	1995-05	1670
1995-06-30	1995-06	1688
1995-07-31	1995-07	2031

Table 1.7: Test dataset

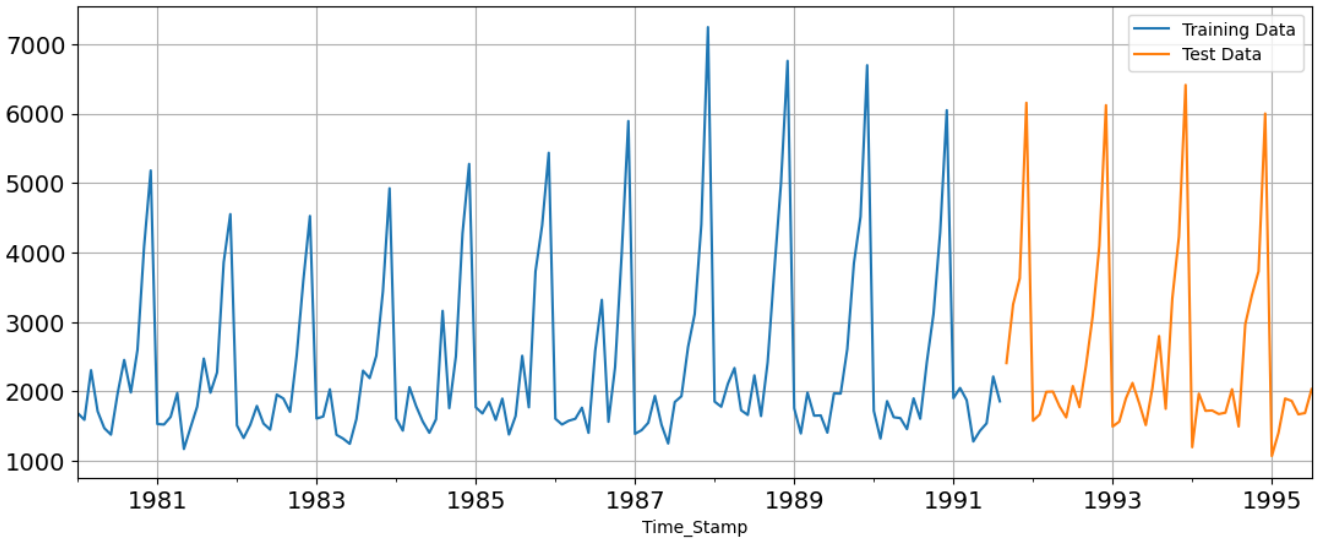


Fig 1.13: Visual plot of training and test data

1.2 Model Building

Build forecasting models - Linear regression - Simple Average - Moving Average - Exponential Models (Single, Double, Triple)

1.2.1 Linear Regression Model

Generate numerical time instance order for both training and test time set.

Add the time instance to the train and test set.

First few rows of Training Data

Time_Stamp	YearMonth	Sparkling	time
1980-01-31	1980-01	1686	1
1980-02-29	1980-02	1591	2
1980-03-31	1980-03	2304	3
1980-04-30	1980-04	1712	4
1980-05-31	1980-05	1471	5

Last few rows of Training Data

Time_Stamp	YearMonth	Sparkling	time
1991-04-30	1991-04	1279	136
1991-05-31	1991-05	1432	137
1991-06-30	1991-06	1540	138
1991-07-31	1991-07	2214	139
1991-08-31	1991-08	1857	140

First few rows of Test Data

Time_Stamp	YearMonth	Sparkling	time
1991-09-30	1991-09	2408	141
1991-10-31	1991-10	3252	142
1991-11-30	1991-11	3627	143
1991-12-31	1991-12	6153	144
1992-01-31	1992-01	1577	145

Last few rows of Test Data

Time_Stamp	YearMonth	Sparkling	time
1995-03-31	1995-03	1897	183
1995-04-30	1995-04	1862	184
1995-05-31	1995-05	1670	185
1995-06-30	1995-06	1688	186
1995-07-31	1995-07	2031	187

Table 1.8: Time instance for Linear Regression

Use the LinearRegression() function on the training dataset and test the model on the test data.

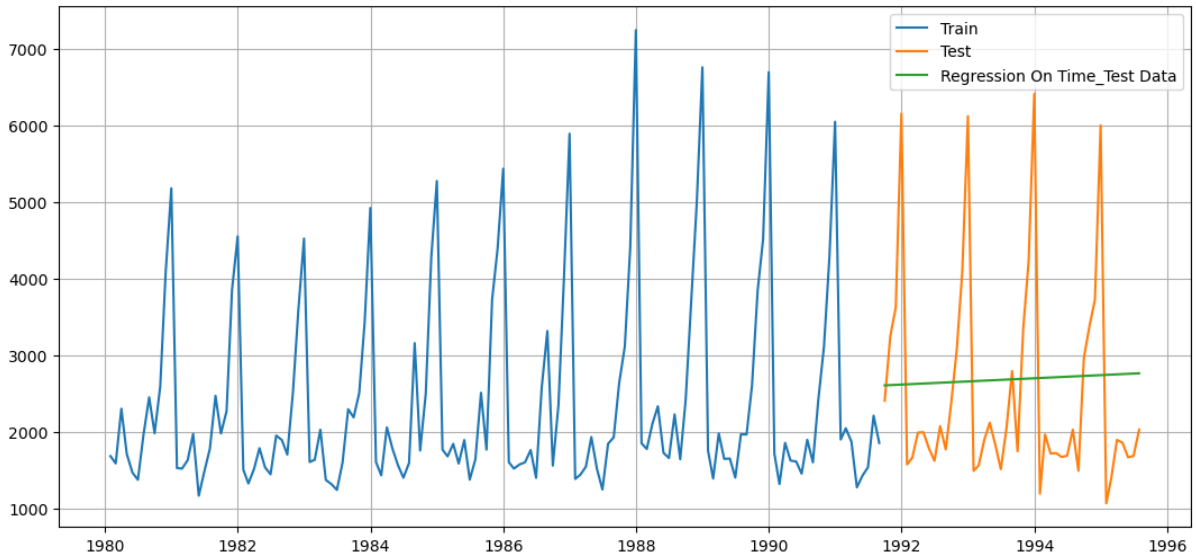


Fig 1.14: Linear Regression on Test data

Model Evaluation:

RMSE for Linear Regression for Test data is 1363.119

1.2.2 Simple Average Model

We forecast using the average of the training values.

Time_Stamp	YearMonth	Sparkling	mean_forecast
1991-09-30	1991-09	2408	2367.471429
1991-10-31	1991-10	3252	2367.471429
1991-11-30	1991-11	3627	2367.471429
1991-12-31	1991-12	6153	2367.471429
1992-01-31	1992-01	1577	2367.471429

Table 1.9 Simple Average using mean of training values

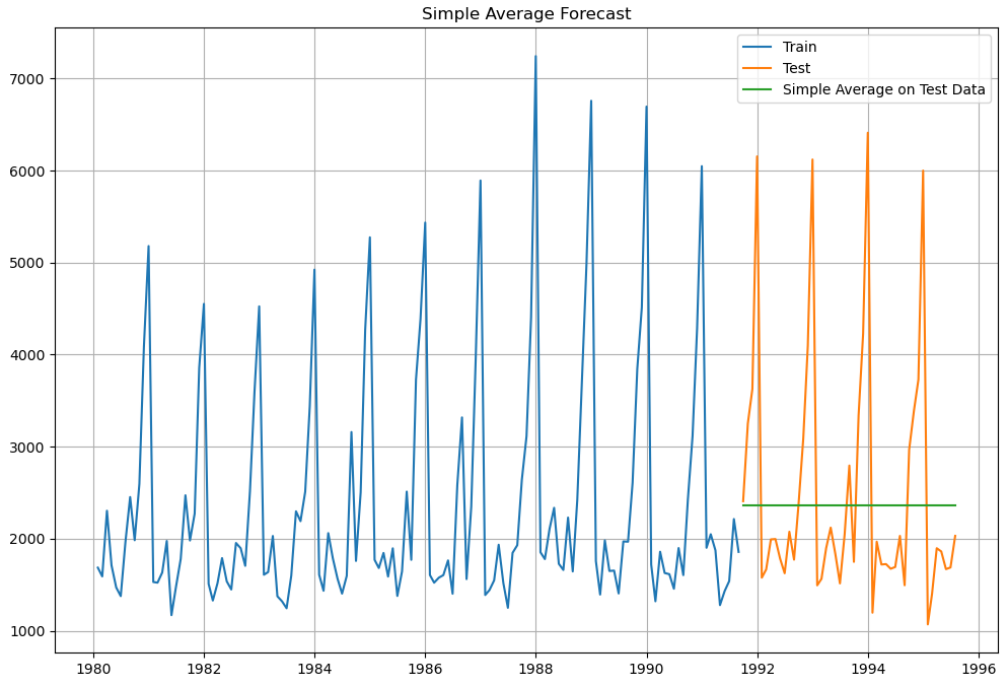


Fig 1.15: Simple Average Model

Model Evaluation:

For Simple Average forecast on the Test Data,RMSE is 1351.788

1.3.3 Moving Average Model

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.

We calculate the moving averages of different intervals of 2, 4, 6 and 9.

Time_Stamp	YearMonth	Sparkling	Trailing_2	Trailing_4	Trailing_6	Trailing_9
1980-01-31	1980-01	1686	NaN	NaN	NaN	NaN
1980-02-29	1980-02	1591	1638.5	NaN	NaN	NaN
1980-03-31	1980-03	2304	1947.5	NaN	NaN	NaN
1980-04-30	1980-04	1712	2008.0	1823.25	NaN	NaN
1980-05-31	1980-05	1471	1591.5	1769.50	NaN	NaN

Table 1.10: Moving average model for intervals of 2,4,6,9

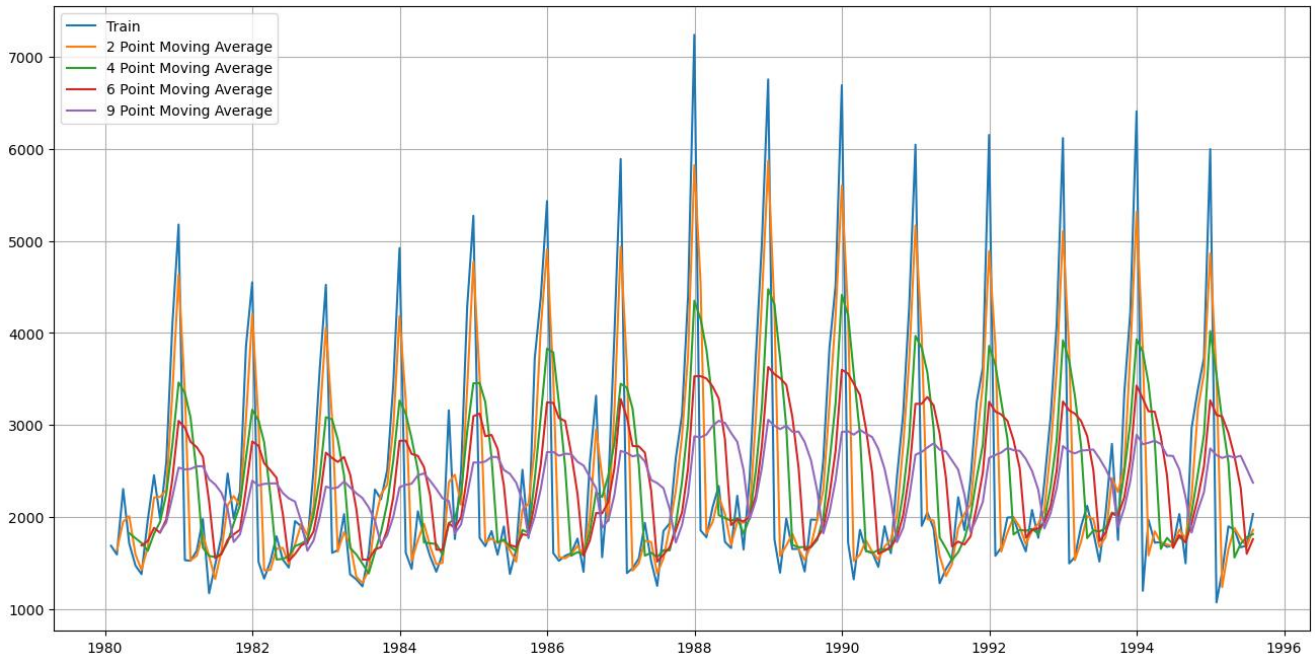


Fig 1.16: Moving average plot for rolling means of 2,4,6,9

Split the dataset into training and test and then apply the rolling means on the training dataset and then on the test dataset.

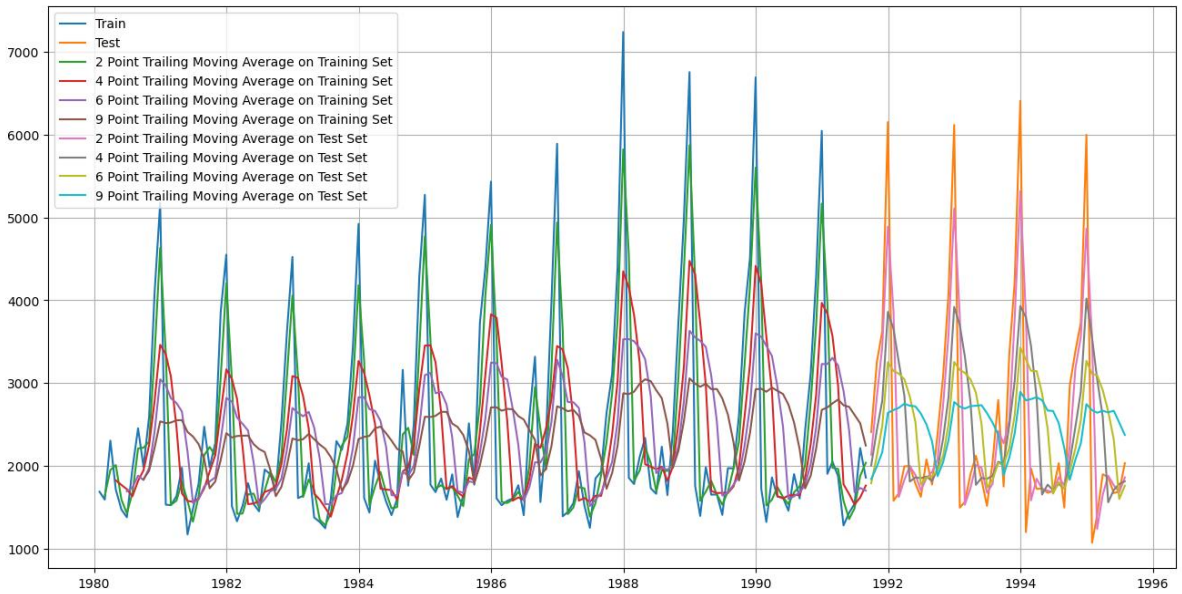


Fig 1.17: Moving Average plot for rolling means of 2,4,6,9 for training and test dataset

Model Evaluation:

For 2 point Moving Average Model forecast on the Test Data, Test RMSE is 823.047
For 4 point Moving Average Model forecast on the Test Data, Test RMSE is 1181.849
For 6 point Moving Average Model forecast on the Test Data, Test RMSE is 1317.753

For 9 point Moving Average Model forecast on the Test Data, Test RMSE is 1403.221

1.3.4 Simple Exponential Smoothing

Apply the simple exponential smoothing to the training dataset and fit it. Then, test the same on the test data.

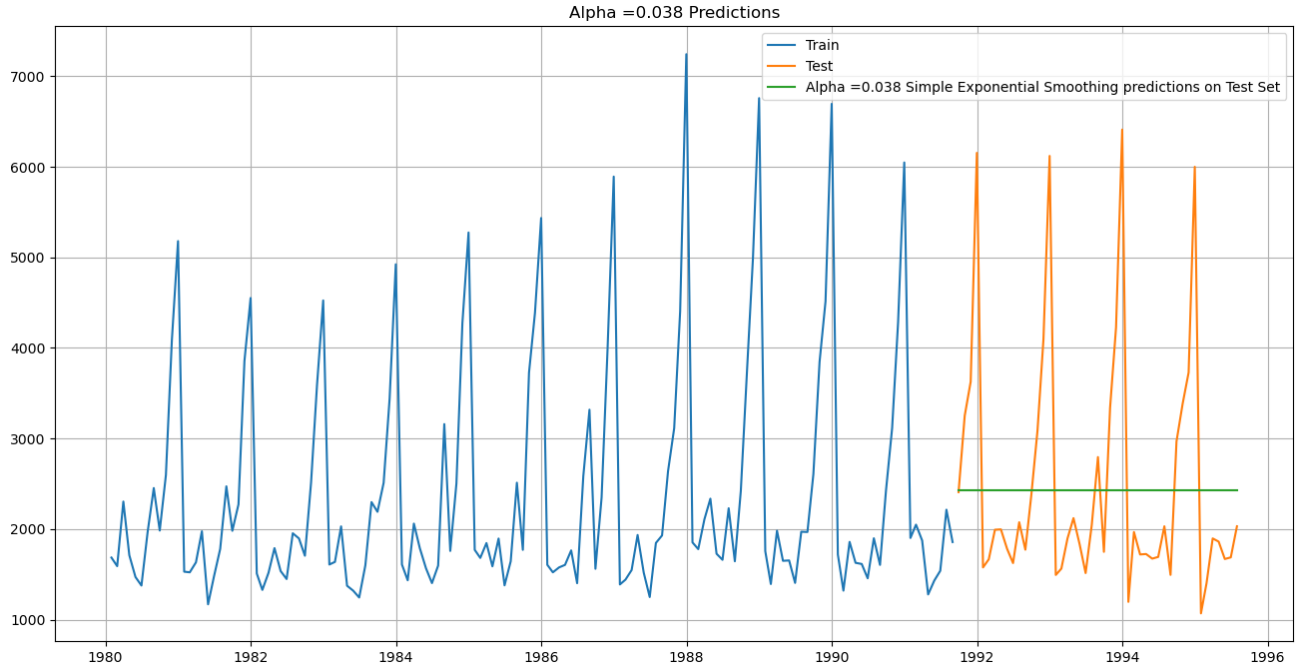


Fig 1.18: Simple Exponential Smoothing for alpha = 0.038

Model Evaluation:

For Alpha =0.038 Simple Exponential Smoothing Model forecast on the Test Data, RMSE is 1346.767

1.3.5 Double Exponential Smoothing/Holt's Method

Apply the Double exponential smoothing to the training dataset, fit it and then test the same using the test data.

	Alpha Values	Beta Values	Train RMSE	Test RMSE
39	0.7	1.0	1836.397580	1409.796784
49	0.9	0.4	1601.418561	1424.105538
51	0.9	0.6	1716.675398	1425.161134
32	0.7	0.3	1513.986303	1439.385942
40	0.8	0.3	1523.787387	1442.396655

Table 1.11: Increasing Test RMSE values for different combinations of alpha and beta

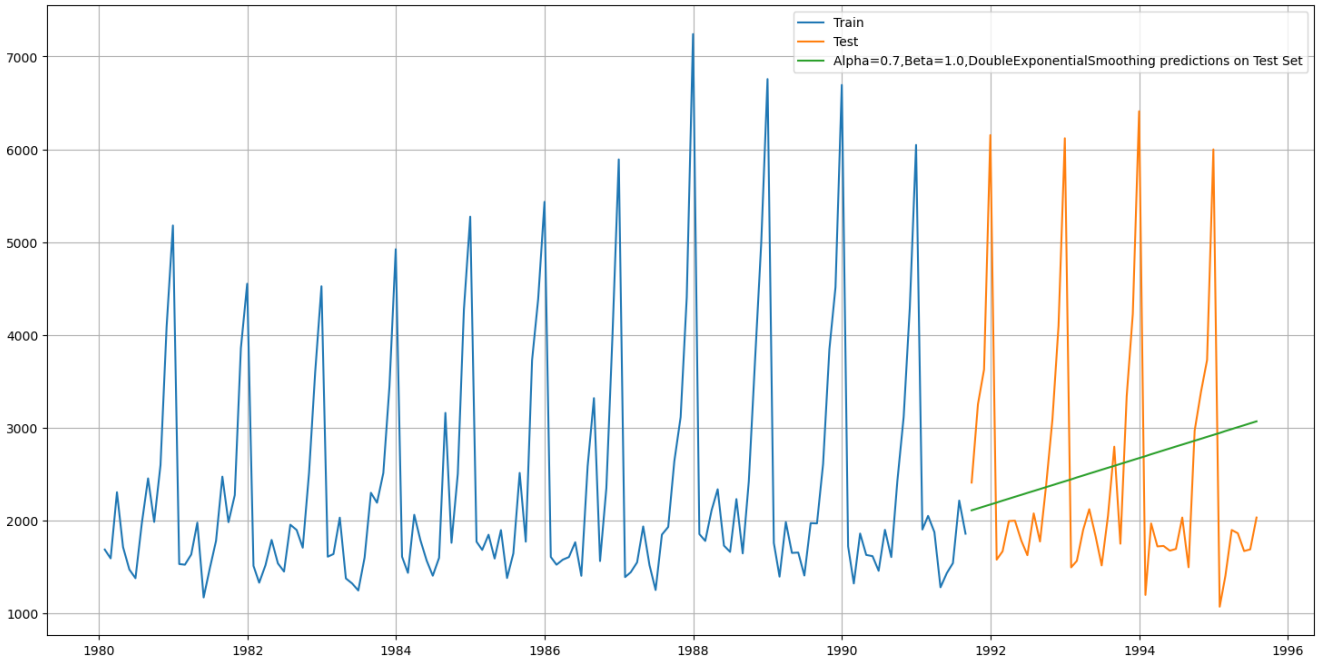


Fig 1.19: Double Exponential Smoothing with Alpha=0.7, Beta=1

Model Evaluation:

For Alpha =0.7 and Beta=1, Double Exponential Smoothing Model forecast on the Test Data, RMSE is 1409.796784

1.3.6 Triple Exponential Smoothing with auto generated values of alpha, beta and gamma

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

Apply the Triple exponential smoothing to the training data and then test it on the test data.

First, build the model using autofit and the auto parameters generated is used to test the testing data.

Time_Stamp	YearMonth	Sparkling	auto_predict
1991-09-30	1991-09	2408	2377.726305
1991-10-31	1991-10	3252	3286.168165
1991-11-30	1991-11	3627	4372.177145
1991-12-31	1991-12	6153	6235.827176
1992-01-31	1992-01	1577	1753.043132

Table 1.12: Triple Exponential Model using auto parameters, alpha=0.075, beta=0.0636, gamma=0.348

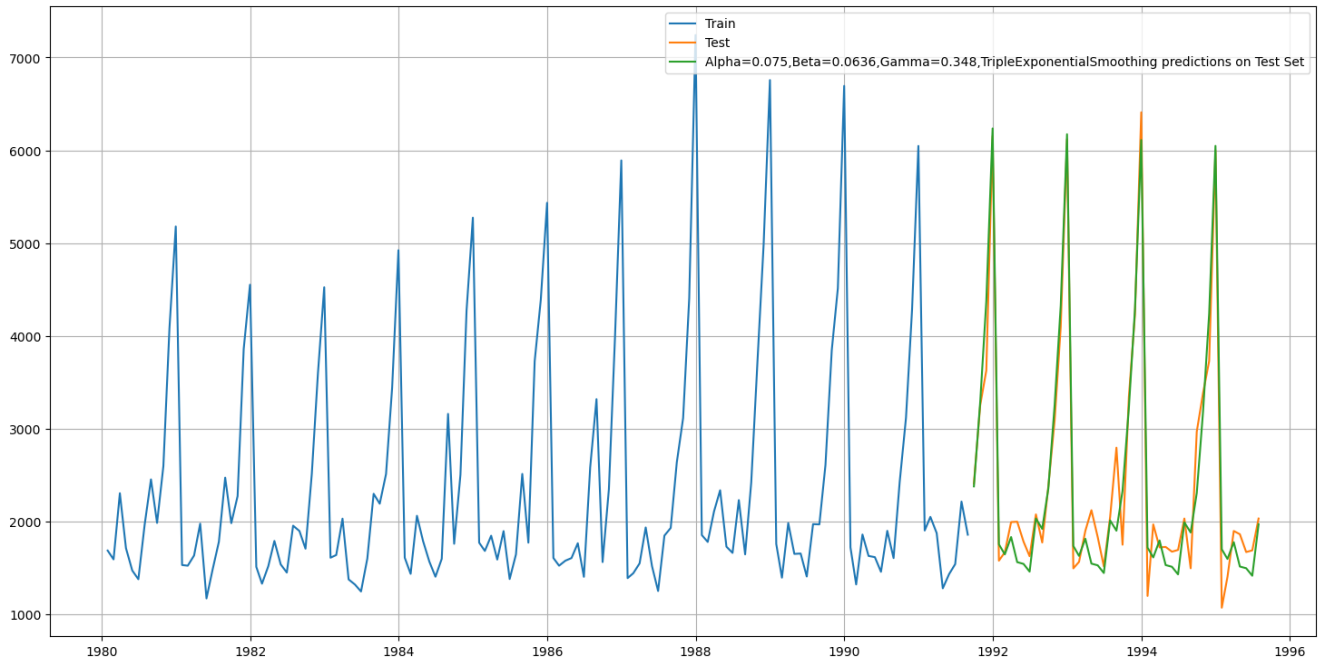


Fig 1.20: Triple Exponential Smoothing with alpha=0.075, beta0.0636, gamma=0.348

Model Evaluation:

For Alpha=0.075,Beta=0.0636,Gamma=0.348, Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 321.534

1.3.7: Triple Exponential Smoothing Model with best values of alpha,beta, gamma

Find different combinations of alpha, beta and gamma and find the Test RMSE for all. Arrange in ascending order to find the best combination of alpha, beta and gamma which gives the least RMSE.

	Alpha Values	Beta Values	Gamma Values	Train RMSE	Test RMSE
336	0.8	0.5	0.3	589.395333	455.263457
1	0.3	0.3	0.4	404.438824	463.321309
265	0.7	0.4	0.4	558.615438	479.698239
152	0.5	0.6	0.3	514.136543	482.979769
344	0.8	0.6	0.3	627.702937	498.316351

Table 1.13: Triple Exponential Smoothing for different values of alpha, beta, gamma

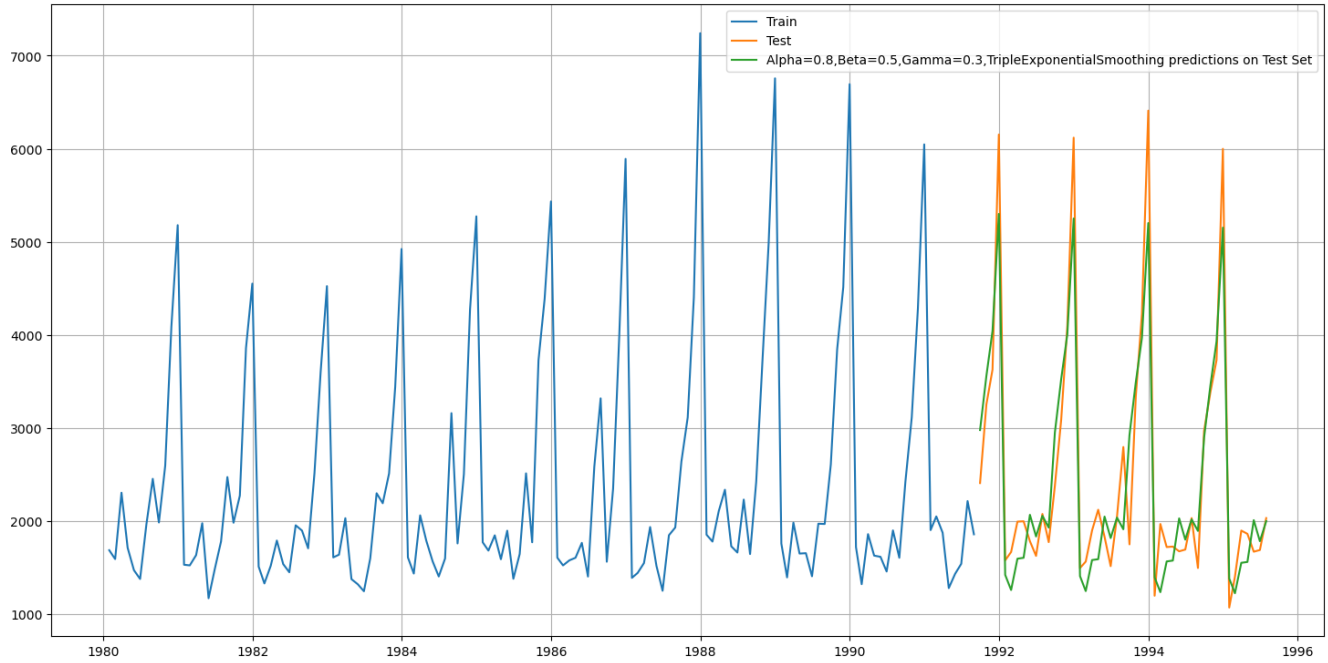


Fig 1.21: Triple Exponential Smoothing with alpha=0.8, beta=0.5, gamma=0.3

Model Evaluation:

For Alpha=0.8,Beta=0.5,Gamma=0.3, Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 455.263457

1.3.8 Check the performance of the models built

	Test RMSE
Alpha=0.075,Beta=0.0636,Gamma=0.348,TripleExponentialSmoothing	321.534274
Alpha=0.8,Beta=0.5,Gamma=0.3,TripleExponentialSmoothing	455.263457
2pointTrailingMovingAverage	823.047225
4pointTrailingMovingAverage	1181.849251
6pointTrailingMovingAverage	1317.752534
Alpha=0.038,SimpleExponentialSmoothing	1346.767434
SimpleAverageModel	1351.787809
LinearRegressionOnTime	1363.118736
9pointTrailingMovingAverage	1403.220949
Alpha=0.7,Beta=1.0,DoubleExponentialSmoothing	1409.796784

Table 1.14: RMSE of the models built

Insights:

We see that out of all the models built using Linear Regression, Simple Average, Moving Average, Simple Exponential Smoothing, Double Exponential Smoothing and Triple Exponential Smoothing, we see that the best model with the least root mean square error (RMSE) is the model with:

Alpha=0.075,Beta=0.0636,Gamma=0.348, Original TripleExponentialSmoothing

Hence, we will build the full model using Triple Exponential Smoothing with the above parameters for level(alpha), trend(beta) and seasonality(gamma).

1.3.8.1 Building the full model with Triple Exponential Smoothing with Alpha=0.075, Beta=0.0636 and Gamma = 0.348

Build the full model using Triple Exponential Smoothing with alpha=0.075, beta=0.0636 and gamma=0.348. Check the RMSE.

RMSE for Triple Exponential Smoothing with Alpha=0.075, Beta=0.0636, Gamma=0.348 on the full model is 347.9618

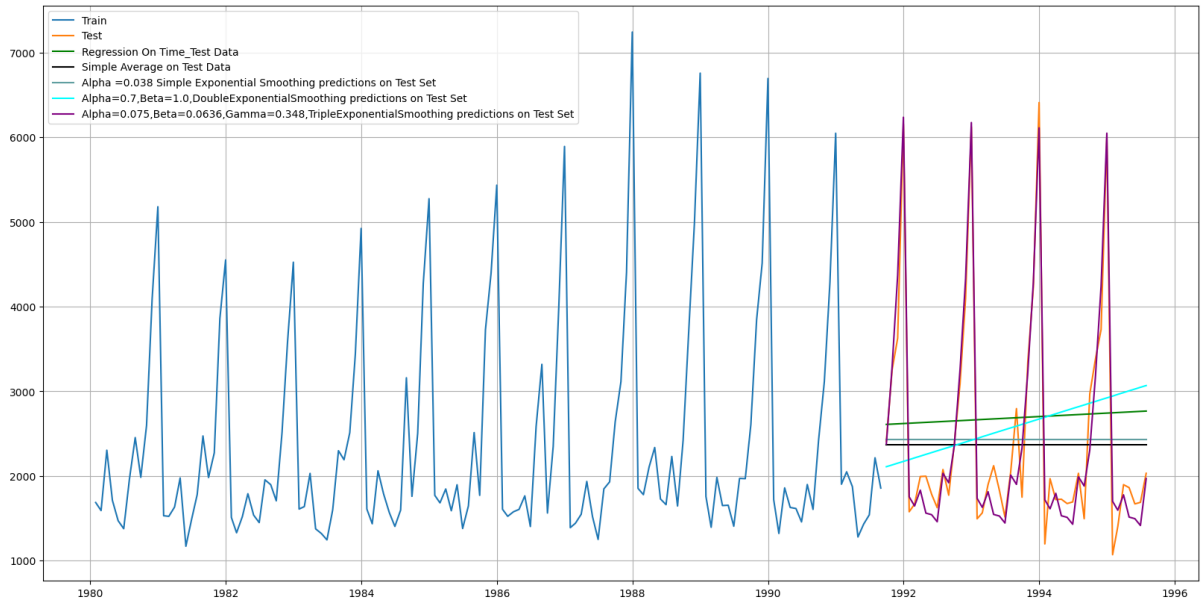


Fig 1.22: Different Model Plots on the Test data

1.4 Check for Stationarity

Test for stationarity of the series using Dickey-Fuller test.

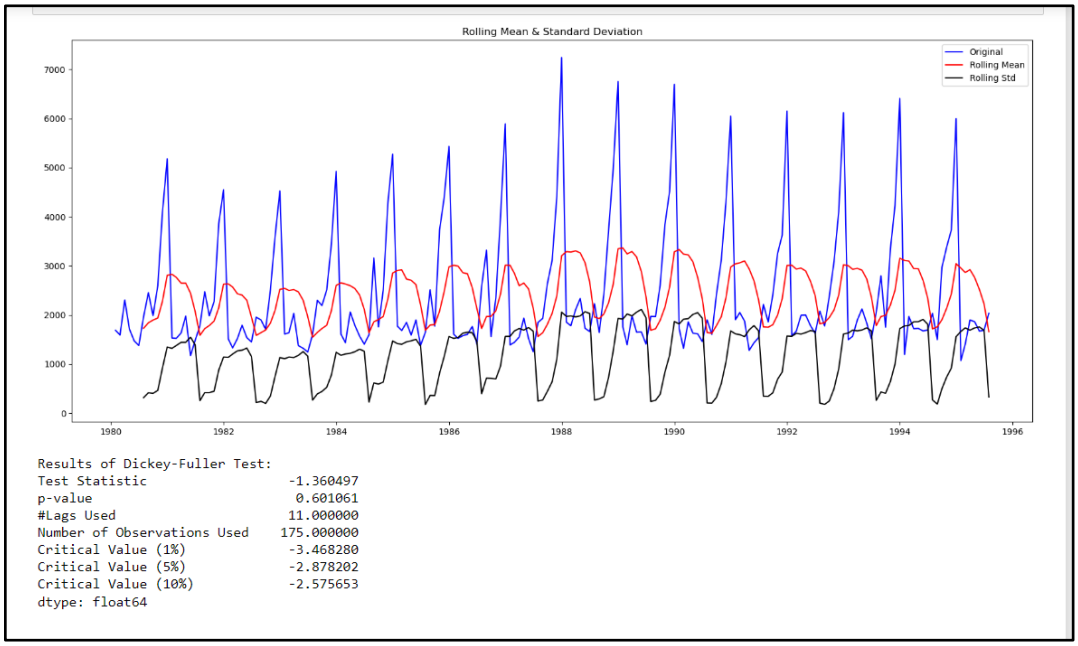


Fig 1.23: Dickey-Fuller Test to check the stationarity

Insights:

As p > 0.05, the time series is not stationary.

We see that at 5% significant level the Time Series is non-stationary.

Let us take a difference of order 1 and check whether the Time Series is stationary or not.

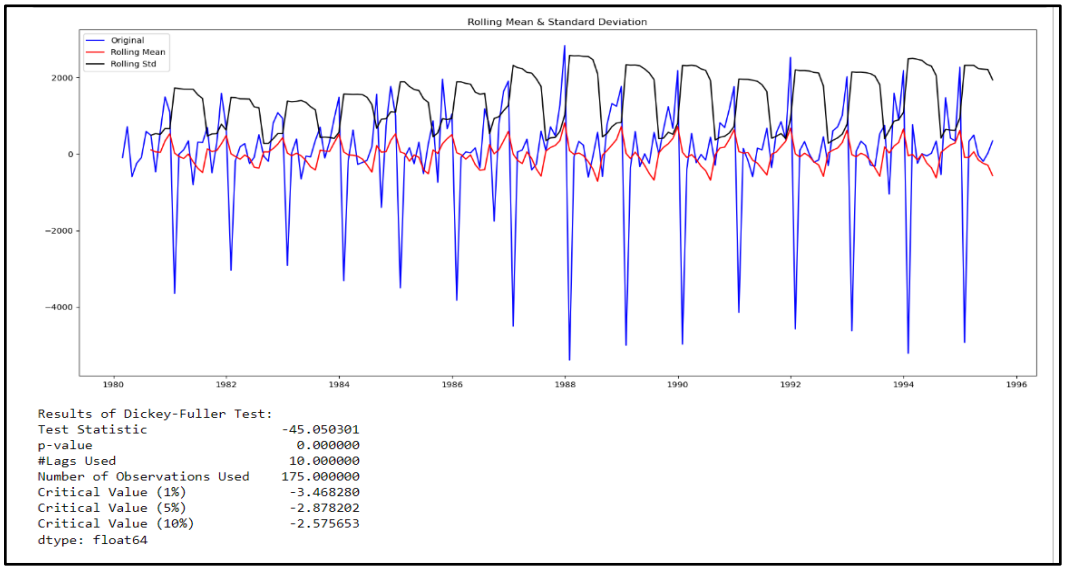


Fig 1.24: Dickey-Fuller Test after first order differencing

Insights:

As p < 0.05, We see that after first differencing, and at α = 0.05, the Time Series is indeed stationary.

1.5 Model Building – Stationary Data

1.5.1 - Generate ACF & PACF Plot and find the AR, MA values.

1.5.1.1 ACF PLOT: We generate the ACF plot for the first order differenced dataset.

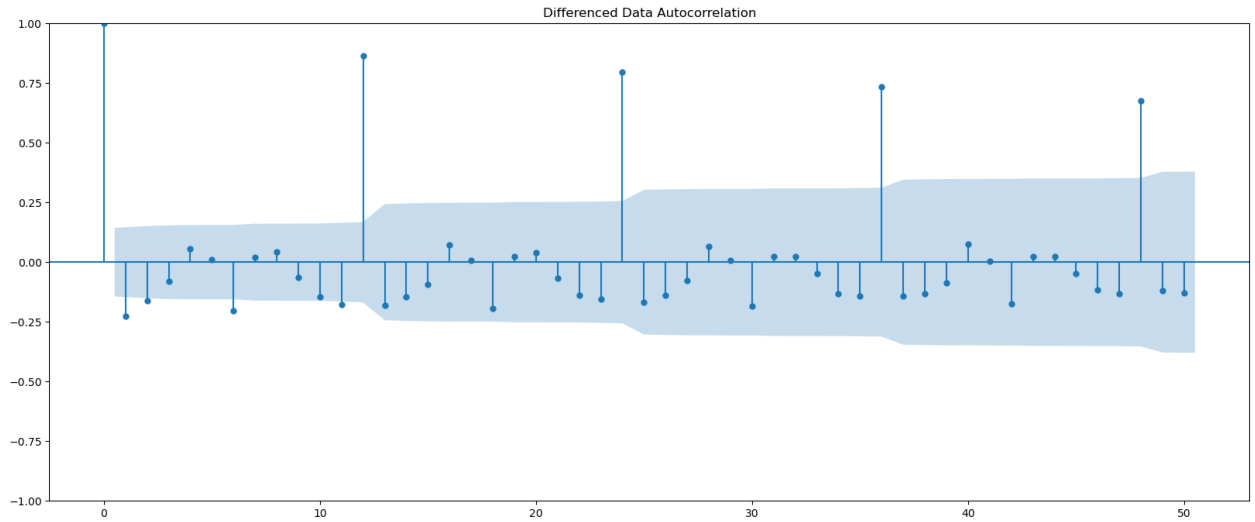


Fig 1.25: ACF Plot of first-order differenced dataset

1.5.1.2 PACF Plot:

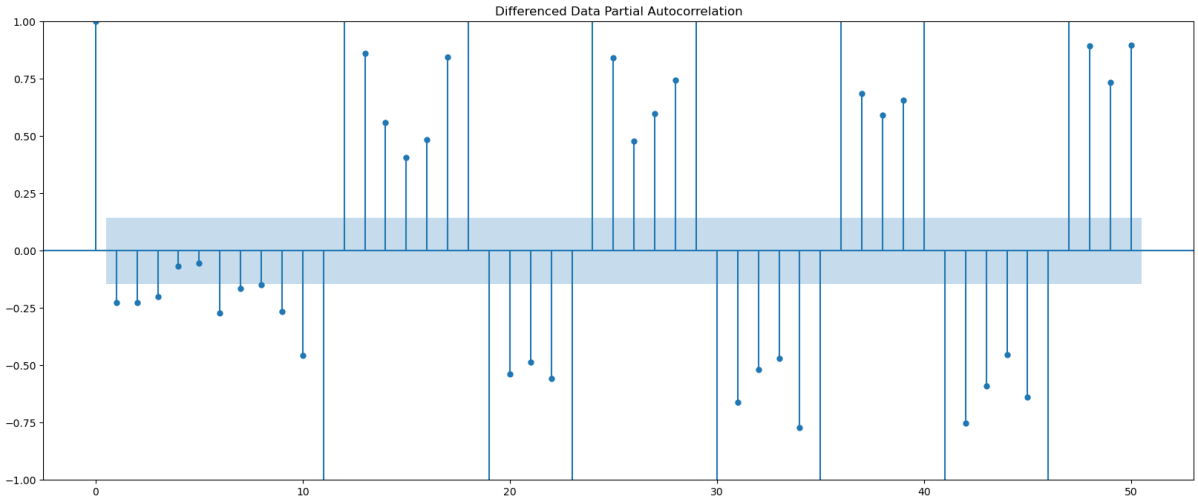


Fig 1.26: PACF Plot of first-order differenced dataset

Insights:

Everything outside the confidence interval that is, the blue band is the significance interval. From the differentiated ACF plot, we can get the q value for the MA model on analyzing all the spikes beyond the blue area. Hence, q = 2.

From the differentiated PACF plot, we can get the value of p for the AR model. Here, p = 3 which gives the spikes outside the blue boundary of the PACF plot.

Thus, p=3, q=2 and d=1

AR value = p = 3

MA value = q = 2

1.5.2 Build ARIMA Models

1.5.2.1 Auto ARIMA

Get a combination of different parameters of p and q in the range of 0 and 2.

Keep the value of d as 1 as we need to take a difference of the series to make it stationary.

Build the Auto ARIMA model using the ARIMA() function on the training data and then test it on the test data.

```
ARIMA(0, 1, 0) - AIC:2407.597117207587
ARIMA(0, 1, 1) - AIC:2400.4928944373605
ARIMA(0, 1, 2) - AIC:2366.1755184903864
ARIMA(1, 1, 0) - AIC:2405.4826254611853
ARIMA(1, 1, 1) - AIC:2367.84569493432
ARIMA(1, 1, 2) - AIC:2366.8908432891567
ARIMA(2, 1, 0) - AIC:2398.416332454428
ARIMA(2, 1, 1) - AIC:2365.962586640441
ARIMA(2, 1, 2) - AIC:2349.177549741146
```

Table 1.15: Auto ARIMA Model with AIC

Arrange the Akaike Information Criteria (AIC) in the ascending order. We see that the best combination is (2,1,2) with the lowest RMSE of 2349.17.

Build the ARIMA() model once again on the training dataset with the best AIC.

The result summary is as follows:

```
                               SARIMAX Results
==========================================================================================
Dep. Variable:                      Sparkling   No. Observations:                  140
Model:                 ARIMA(2, 1, 2)   Log Likelihood               -1169.589
Date:                Sat, 13 Apr 2024   AIC                           2349.178
Time:                        10:06:37   BIC                           2363.850
Sample:                    01-31-1980   HQIC                          2355.140
                         - 08-31-1991
Covariance Type:                  opg
==========================================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------
ar.L1          1.2739      0.050     25.651      0.000       1.177       1.371
ar.L2         -0.5492      0.083     -6.577      0.000      -0.713      -0.386
ma.L1         -1.9056      0.065    -29.226      0.000      -2.033      -1.778
ma.L2          0.9166      0.066     13.950      0.000       0.788       1.045
sigma2      1.148e+06   3.21e-08   3.57e+13      0.000    1.15e+06    1.15e+06
==========================================================================================
Ljung-Box (L1) (Q):                   0.24   Jarque-Bera (JB):                18.19
Prob(Q):                              0.62   Prob(JB):                         0.00
Heteroskedasticity (H):               2.63   Skew:                             0.67
Prob(H) (two-sided):                  0.00   Kurtosis:                         4.17
==========================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 5.76e+29. Standard errors may be unstable.
```

```
: predicted_auto_ARIMA = results_auto_ARIMA.forecast(steps=len(test))
```

Table 1.16: Auto ARIMA result summary

Predict on the test set and evaluate the auto ARIMA

RMSE for Auto ARIMA(2,1,2) is 1296.238042

1.5.2.2 MANUAL ARIMA

By looking at the ACF and PACF plots, it is very clearly seen that p=3 and q=2.

Build the manual ARIMA by passing the parameters p=3, d=1 and q=2, as the order of differencing is 1 on the training data.

(p, d, q) = (3, 1, 2)

Once the model is built on the training data, forecast on the length of the test dataset.

Find the RMSE between the already existing test data and the predicted data using the manual ARIMA built and tested on the tested data.

SARIMAX Results

Dep. Variable:	Sparkling	No. Observations:	140
Model:	ARIMA(3, 1, 2)	Log Likelihood	-1176.023
Date:	Sat, 13 Apr 2024	AIC	2364.047
Time:	10:06:39	BIC	2381.653
Sample:	01-31-1980 - 08-31-1991	HQIC	2371.202
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.4558	0.110	-4.131	0.000	-0.672	-0.240
ar.L2	0.3179	0.105	3.015	0.003	0.111	0.525
ar.L3	-0.2262	0.188	-1.203	0.229	-0.595	0.142
ma.L1	-5.438e-07	2.644	-2.06e-07	1.000	-5.182	5.182
ma.L2	-1.0000	0.133	-7.495	0.000	-1.262	-0.738
sigma2	1.235e+06	7.46e-07	1.66e+12	0.000	1.24e+06	1.24e+06

Ljung-Box (L1) (Q):	0.03	Jarque-Bera (JB):	8.71
Prob(Q):	0.87	Prob(JB):	0.01
Heteroskedasticity (H):	2.78	Skew:	0.54
Prob(H) (two-sided):	0.00	Kurtosis:	3.60

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 3.93e+29. Standard errors may be unstable.

Table 1.17: Manual ARIMA result summary

Predict on the TEST set and evaluate the model.

RMSE for Manual ARIMA(3,1,2) is 1325.7710347694842

1.5.2.3 AUTO SARIMA

The best parameters with the lowest AIC (Akaike Information Criteria) are taken.

We see that there can be a seasonality of 12. We will run our auto SARIMA models by setting seasonality to 12.

```
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)
```

Run the SARIMAX() function on the training data to get the auto parameters for level, trend and seasonality.

The data with the different combinations along with their AIC values is displayed. Arrange them in the ascending order and choose the combination with the lowest AIC value.

SARIMAX Results

Dep. Variable:	y	No. Observations:	140
Model:	SARIMAX(1, 1, 2)x(1, 0, 2, 12)	Log Likelihood	-828.767
Date:	Sat, 13 Apr 2024	AIC	1671.534
Time:	10:08:25	BIC	1690.563
Sample:	0	HQIC	1679.254
	- 140		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.5843	0.235	-2.484	0.013	-1.045	-0.123
ma.L1	-0.1705	0.202	-0.844	0.399	-0.566	0.225
ma.L2	-0.7425	0.157	-4.717	0.000	-1.051	-0.434
ar.S.L12	1.0375	0.015	68.641	0.000	1.008	1.067
ma.S.L12	-0.5303	0.096	-5.513	0.000	-0.719	-0.342
ma.S.L24	-0.1414	0.118	-1.199	0.230	-0.373	0.090
sigma2	1.471e+05	1.87e+04	7.858	0.000	1.1e+05	1.84e+05

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	9.16
Prob(Q):	0.95	Prob(JB):	0.01
Heteroskedasticity (H):	1.56	Skew:	0.30
Prob(H) (two-sided):	0.18	Kurtosis:	4.26

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

Table 1.18: Auto SARIMA result summary

Insights:

- **Jarque-Bera test** states error is normal or non-normal.

  #Null Hypothesis: Normal,

  #Alternate Hypothesis: Non-Normal

  As the p-value 0.01 < 0.05 so Alternate Hypothesis is true, hence Errors are non-normal.

- **Ljung-Box** states Null hypothesis: no autocorrelation, Alternate hypothesis: autocorrelation

  As the p-value of Ljung-Box test is 0.95 > 0.05 , so no autocorrelation.

- **Heteroskedasticity test** Null:Homo, Alternate: Hetero

  As the p-value is 0.18 > 0.05 so Homo, pattern is not observed.

Conclusion: Errors are not normally distributed with no autocorrelation and Heteroskedasticity(patterns) is not observed.

Diagnostics Plot for errors in Auto SARIMA:

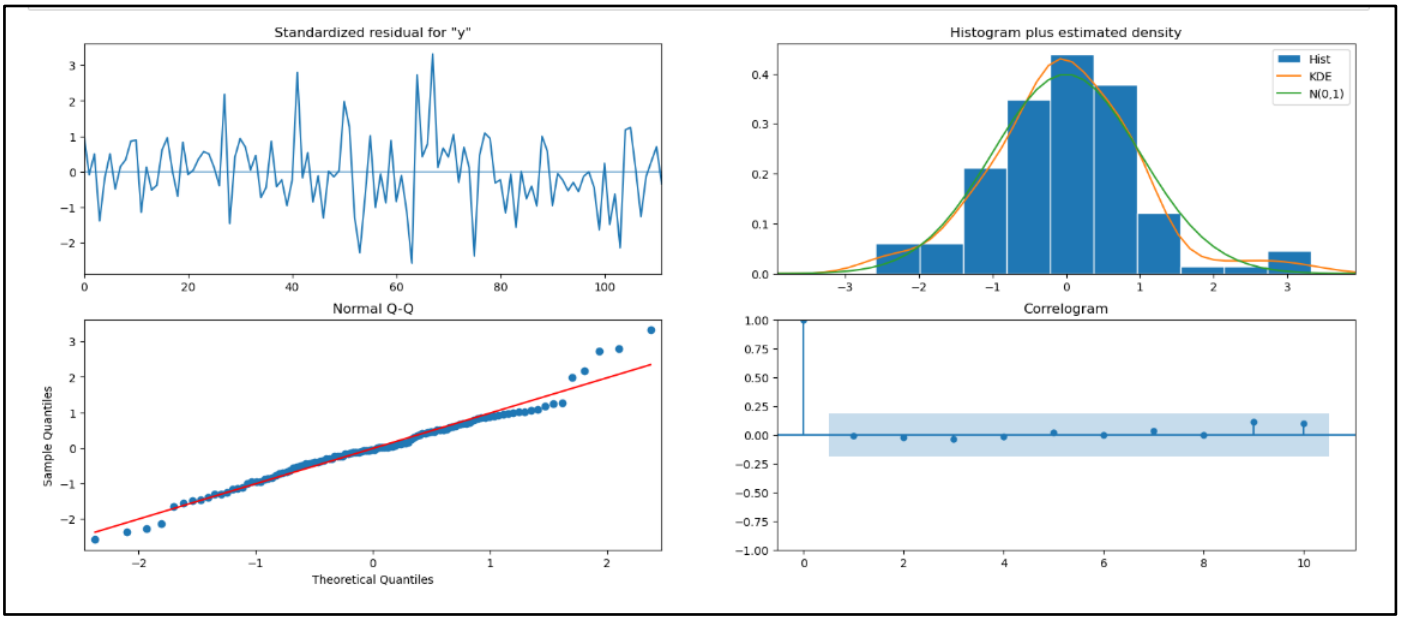


Fig 1.27: Diagnostics Plot in Auto SARIMA

Predict on the Test Model and evaluate the model:

RMSE for Auto SARIMA(1,1,2)(1,0,2,12) is 468.2342

1.5.2.4 MANUAL SARIMA

Build a version of the SARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots. As the Seasonality is 12, do a seasonal differentiation of ACF and PACF plots.

Order of the series (p, d, q) = (3, 1, 1) Seasonal order of the time series (P, D, Q)F = (0, 1, 0) 12

The train data passes the stationarity test as the p-value < 0.05.

Run the SARIMAX() function on the training data with order (3,1,1) and seasonal order (0,1,0,12).

SARIMAX Results

Dep. Variable:	y	No. Observations:	140
Model:	SARIMAX(3, 1, 1)x(0, 1, [], 12)	Log Likelihood	-925.162
Date:	Sun, 14 Apr 2024	AIC	1860.323
Time:	21:03:58	BIC	1874.425
Sample:	0 - 140	HQIC	1866.051
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.2676	0.076	3.523	0.000	0.119	0.416
ar.L2	-0.1351	0.092	-1.472	0.141	-0.315	0.045
ar.L3	0.0688	0.083	0.824	0.410	-0.095	0.232
ma.L1	-1.0000	0.099	-10.053	0.000	-1.195	-0.805
sigma2	1.715e+05	5.8e-07	2.96e+11	0.000	1.72e+05	1.72e+05

Ljung-Box (L1) (Q):	0.02	Jarque-Bera (JB):	13.00
Prob(Q):	0.90	Prob(JB):	0.00
Heteroskedasticity (H):	1.00	Skew:	0.12
Prob(H) (two-sided):	0.99	Kurtosis:	4.57

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 6.14e+26. Standard errors may be unstable.

Table 1.19: Manual SARIMA results summary

Diagnostic Plot:

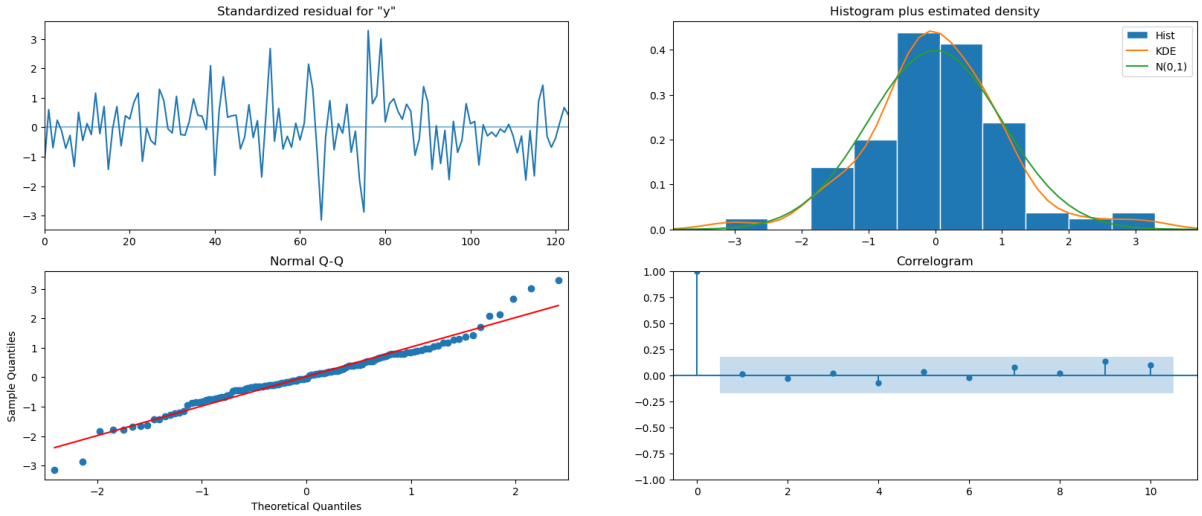


Fig 1.28: Diagnostic Plot for errors in Manual SARIMA

Predict the model on the Test data and evaluate the model:

RMSE for Auto SARIMA(2,1,3)(0,1,0,12) is 407.409

1.5.3 Check the performance of the models built

	RMSE
AUTO ARIMA(2,1,2)	1296.238042
MANUAL ARIMA(3,1,2)	1325.771035
AUTO SARIMA(1,1,2)(1,0,2,12)	468.234211
MANUAL SARIMA(3,1,1)(0,1,0,12)	407.409834

Table 1.20: Performance of ARIMA and SARIMA models

1.6 Compare the performance of the models built

	RMSE
MANUAL SARIMA(3,1,1)(0,1,0,12)	407.409834
AUTO SARIMA(1,1,2)(1,0,2,12)	468.234211
AUTO ARIMA(2,1,2)	1296.238042
MANUAL ARIMA(3,1,2)	1325.771035

Table 1.21: RMSE of ARIMA and SARIMA models in ascending order

1.6.1 Choose the best model with proper rationale

After comparing the Root Mean Squared Errors of all the four models, namely, Manual ARIMA, Auto ARIMA, Manual SARIMA and Auto SARIMA, we see the following:

- The best model so far is the **Manual SARIMA model** with the order being (3,1,1) and seasonal order being (0,1,0) with seasonality of 12.
- The next best model is the Auto SARIMA model.
- The RMSE is greatly reduced once the seasonality component is also taken in the time series.

1.6.2 Rebuild the best model using the entire data

Use SARIMAX() function with order (3,1,1) and seasonality order (0,1,0,12) to build the full model.

SARIMAX Results

Dep. Variable:	Sparkling	No. Observations:	187
Model:	SARIMAX(3, 1, 1)x(0, 1, [], 12)	Log Likelihood	-1277.533
Date:	Sun, 14 Apr 2024	AIC	2565.067
Time:	21:06:45	BIC	2580.775
Sample:	01-31-1980 - 07-31-1995	HQIC	2571.441
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.1200	0.057	2.112	0.035	0.009	0.231
ar.L2	-0.0865	0.077	-1.125	0.260	-0.237	0.064
ar.L3	0.0433	0.075	0.580	0.562	-0.103	0.190
ma.L1	-1.0000	0.081	-12.275	0.000	-1.160	-0.840
sigma2	1.761e+05	4.63e-07	3.81e+11	0.000	1.76e+05	1.76e+05

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	24.32
Prob(Q):	0.95	Prob(JB):	0.00
Heteroskedasticity (H):	1.37	Skew:	0.15
Prob(H) (two-sided):	0.24	Kurtosis:	4.82

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 1.51e+27. Standard errors may be unstable.

Table 1.22: Manual SARIMA model for the full data

Diagnostics test for the full model:

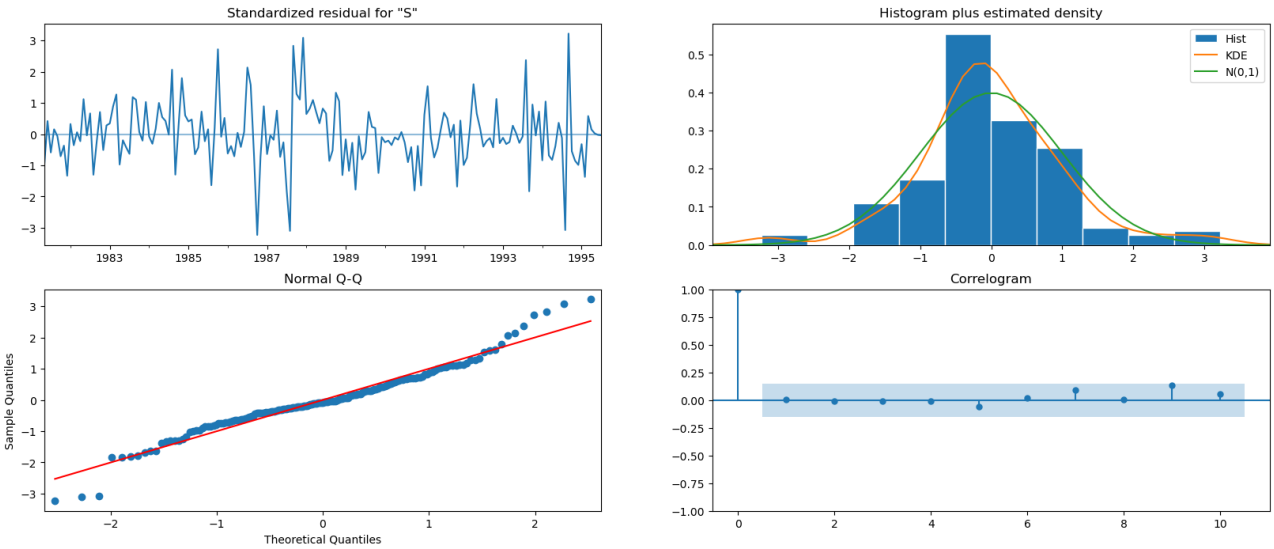


Fig 1.29: Diagnostics test for errors in the full model

1.6.3 Evaluate the model on the whole and predict 12 months into the future (till the end of next year)

- Apply the Manual SARIMA with the order and seasonal parameters to run on the full model.
- Calculate the RMSE for the full model.
- Forecast for the next 12 months that is, from 31/08/1995 to 31/07/1996.

RMSE of the Full Model 576.745

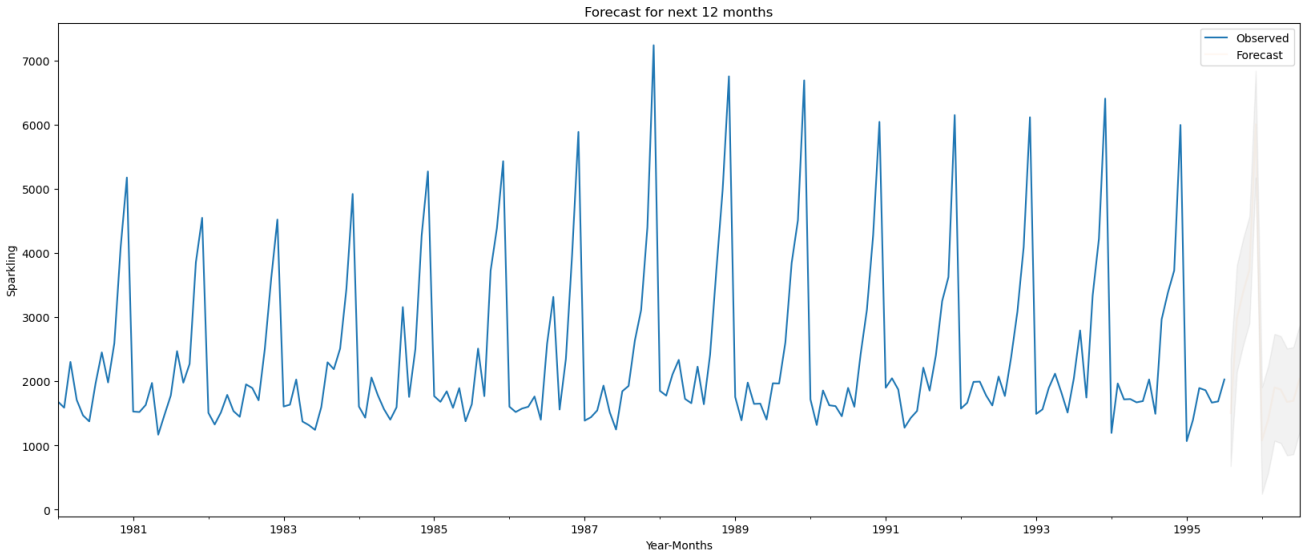


Fig 1.30: Forecast for the next 12 months from 1995 to 1996

```
1995-08-31    1504.665943
1995-09-30    2978.349590
1995-10-31    3394.811883
1995-11-30    3739.106890
1995-12-31    6009.218412
1996-01-31    1080.182978
1996-02-29    1412.181856
1996-03-31    1907.189617
1996-04-30    1872.189111
1996-05-31    1680.188330
1996-06-30    1698.188616
1996-07-31    2041.188696
Freq: M, Name: predicted_mean, dtype: float64
```

Table 1.23: Predicted values from 31/08/1995 to 31/07/1996

1.7 Actionable Insights and Recommendations

Insights:

- The graph of the time series data contains seasonality but no trend.
- There are outliers in the dataset.
- The yearly boxplot shows that the sale of Sparkling wine has decreased from 1980 to 1995. The maximum sale of wine has occured in 1988 while the minimum is in the year 1995.
- December month across the years has the highest sales of Sparkling wine.
- The data is right skewed as the mean is greater than median
- Multiplicative decomposition is the type of decomposition used for the dataset as the residuals do not show any particular trend.
- Among all the models used to model the data, Triple Exponential Smoothing is the one which is chosen with specific values of alpha, beta and gamma as this model has the lowest RMSE value.
- The DataFrame is not stationary, hence Dicky Fuller method is used to make it stationary
- Manual SARIMA model with seasonality 12 has the lowest RMSE value, hence it is the most preferred model.
- Since SARIMA takes into account the seasonality aspect, it is a more powerful model as compared to ARIMA.

Recommendations:

- The ABC Estate Wines company must ensure the stock availability of Sparkling type wines on a higher during the month of OCT, NOV & DEC on all years
- Overfitting issues are possible, hence the model chosen should be tested with different hyperparameters and then moved to production.
- SARIMA model thrives in capturing the periodic surges in demand. Hence, the business can up its supply during the peak demand periods.
- The summer months see an average sale of Sparkling wine, hence ABC Estate Wines company can try to market their wine more during the summer months.
- Winter season and the Christmas season is the influencing factor for the sale of wine. The ABC company can try to bring increase it further by increasing the supply of Sparkling wine.

****