TIME SERIES PROJECT BUSINESS REPORT

PG-DSBA

Written by

Priyamvada Singh

Dated: **28-05-2023**

(Format: dd-mm-yyyy)

Table of Contents

Table of Figures

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.

1. Read the data as an appropriate Time Series data and plot the data. - 2 points

- Basic analysis on the given data set:

  i. The first five rows of the dataset:

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

  ii. The last five rows of the data set:

	YearMonth	Sparkling
182	1995-03	1897
183	1995-04	1862
184	1995-05	1670
185	1995-06	1688
186	1995-07	2031

  iii. Total number of records is 187; from 1980-01 to 1995-07.

  iv. Converted time series data from 'object' datatype to 'datetime64':

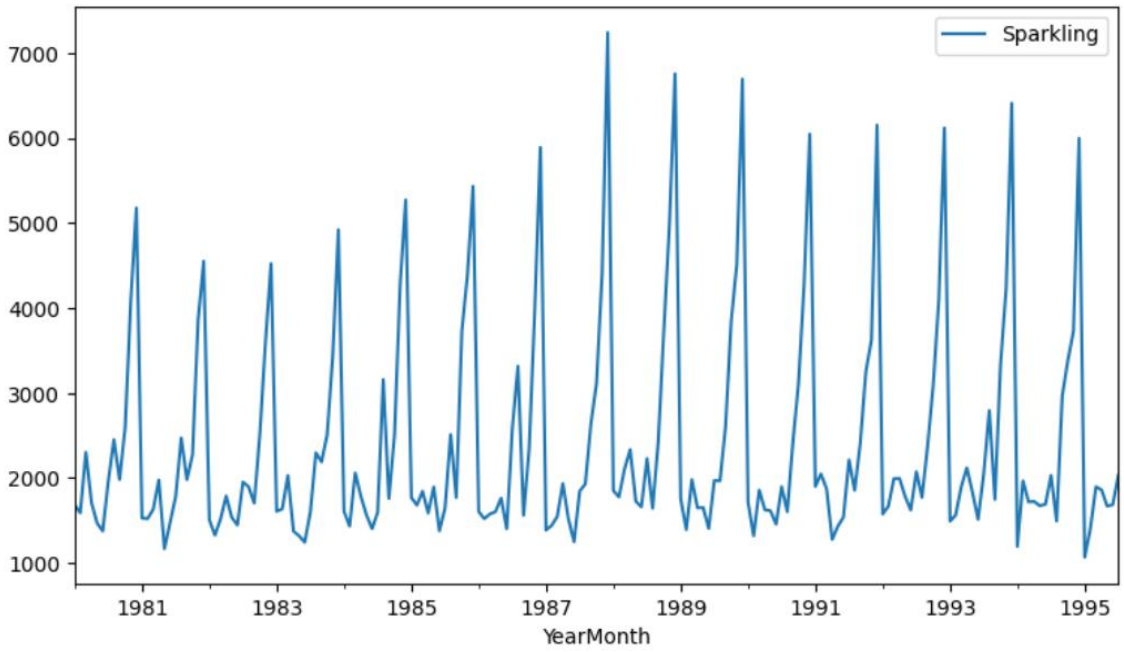
```
YearMonth    object
Sparkling     int64
dtype: object
```

```
YearMonth    datetime64[ns]
Sparkling             int64
dtype: object
```

  v. There are 0 null values in the dataset.

  vi. Below is the plot for the Sparkling timeseries:

Figure 1 - Sparkling.csv Timeseries Plot



2. Perform appropriate Exploratory Data Analysis to understand the data and also perform decomposition. - 9 points

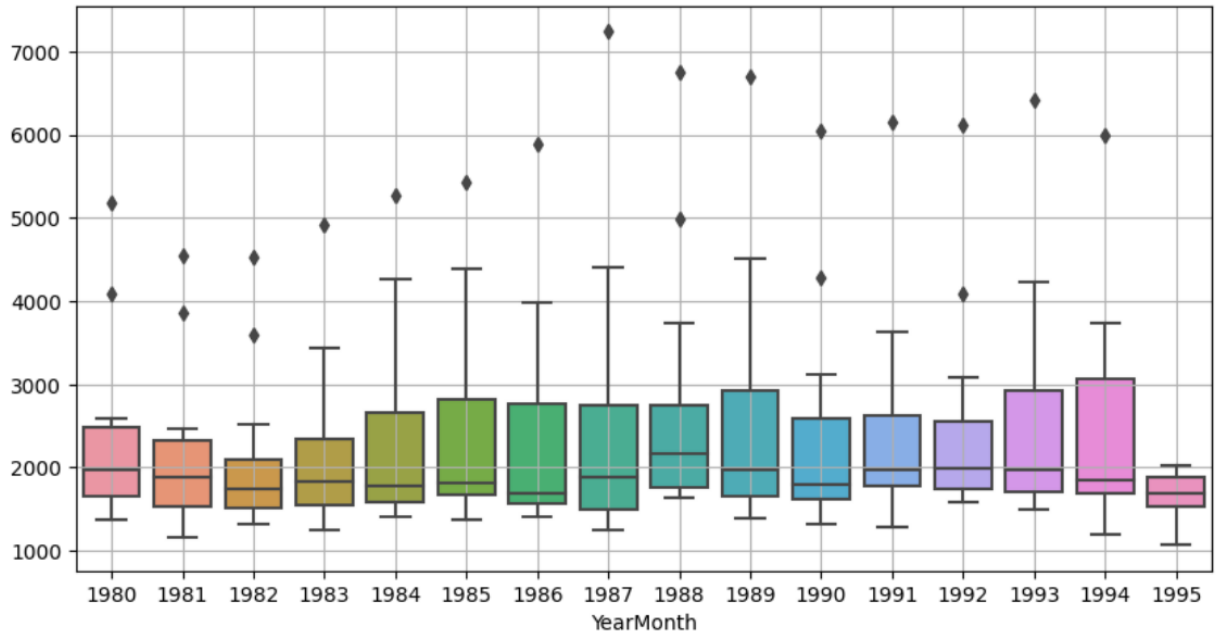
i. The following is the description of each numeric variable:

	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

- *The above information provides the mean, standard deviation, minimum, 25%, 50% (median), 75% and maximum data point values for the timeseries.*

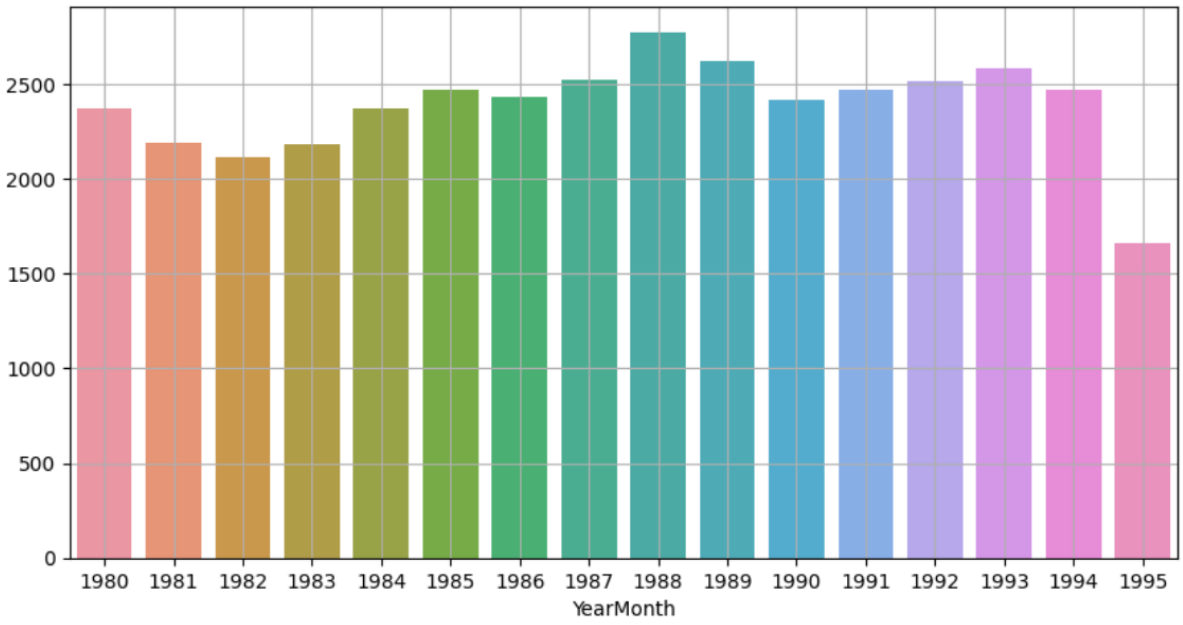
ii. Below is the boxplot for each year for Sparkling wine sales on yearly basis. It shows outliers in almost every years.

Figure 2 - Boxplot I



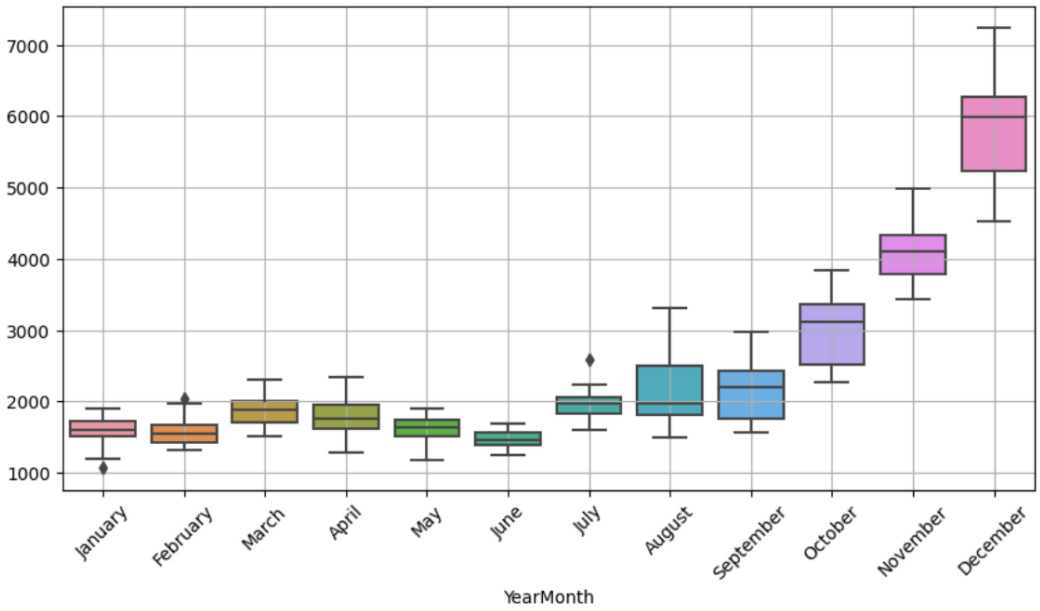
iii. Year 1988 saw the maximum sales for Sparkling wine, as shown below by the bar plot. Although, unlike Sparkling wine sales, Sparkling wine has a fairly uniform volume of sales across years.

Figure 3 – Bar plot I



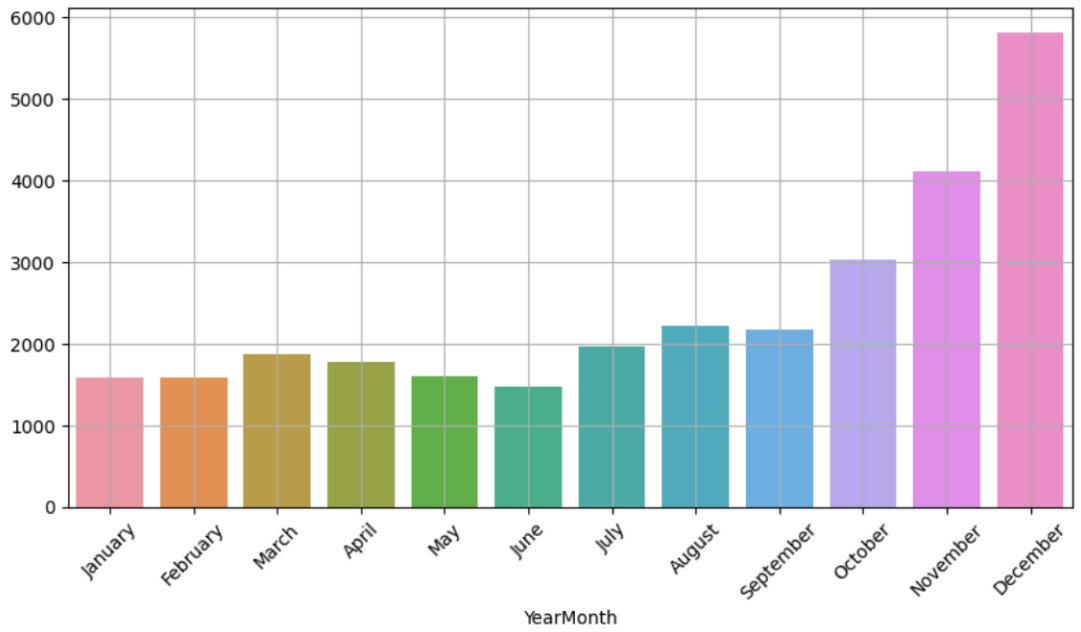
iv. The below boxplot shows the minimum and maximum Sparkling wine sales volume for every month. The widest range is shown in the December boxplot.

Figure 4 - Boxplot II



v. While analysing monthly trend, December showed the highest volume of sales for Sparkling wine, as shown by the bar graph below. This could be due to the Holiday season, i.e., Christmas and New Years' Eve.

Figure 5 – Bar plot II



vi. This trend remains pronounced even when we look at each year differently, as shown in the table below. Every year, wine sales increase in the month of December by a noticeable amount.

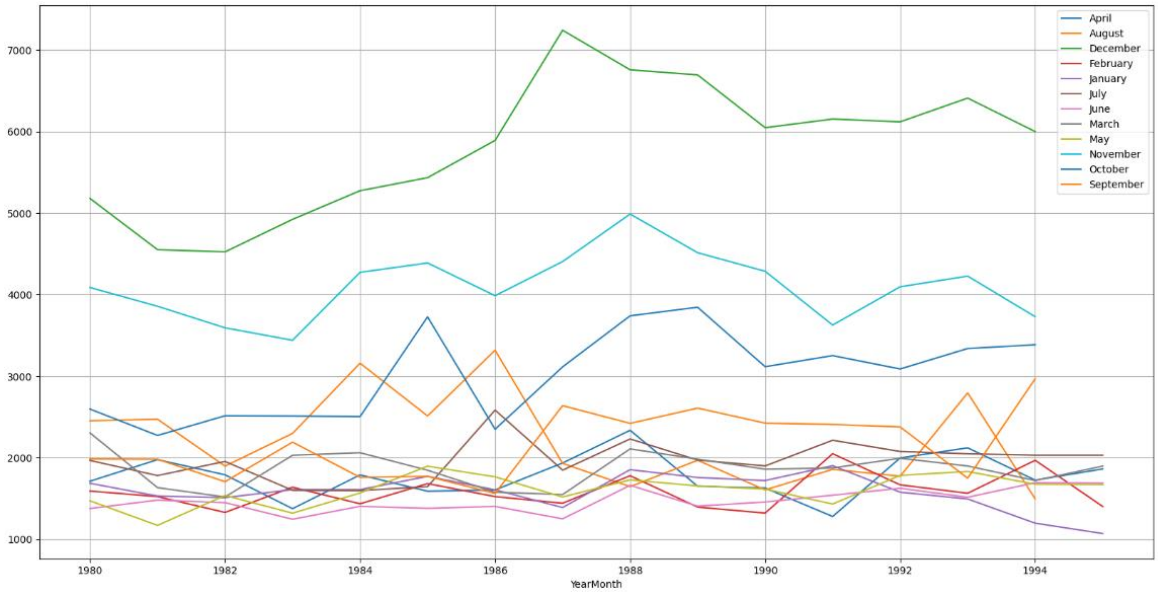
Figure 6 - Pivot Table I

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

Note: The NaN values showing in 1995 after September are because no data is available after that month of the last year, i.e., 1995. The data for 1995 is only available till July.

vii. The below graph shows the monthly wine sale trends with each year on the x-axis. Even here, November and December seem to stand out as the highest number of sales for wine.

Figure 7 - Yearly Trend Plot with Month as Hue



viii. Now, as we did with monthly data for each year, we will also check yearly data for each month with the help of the pivot table.

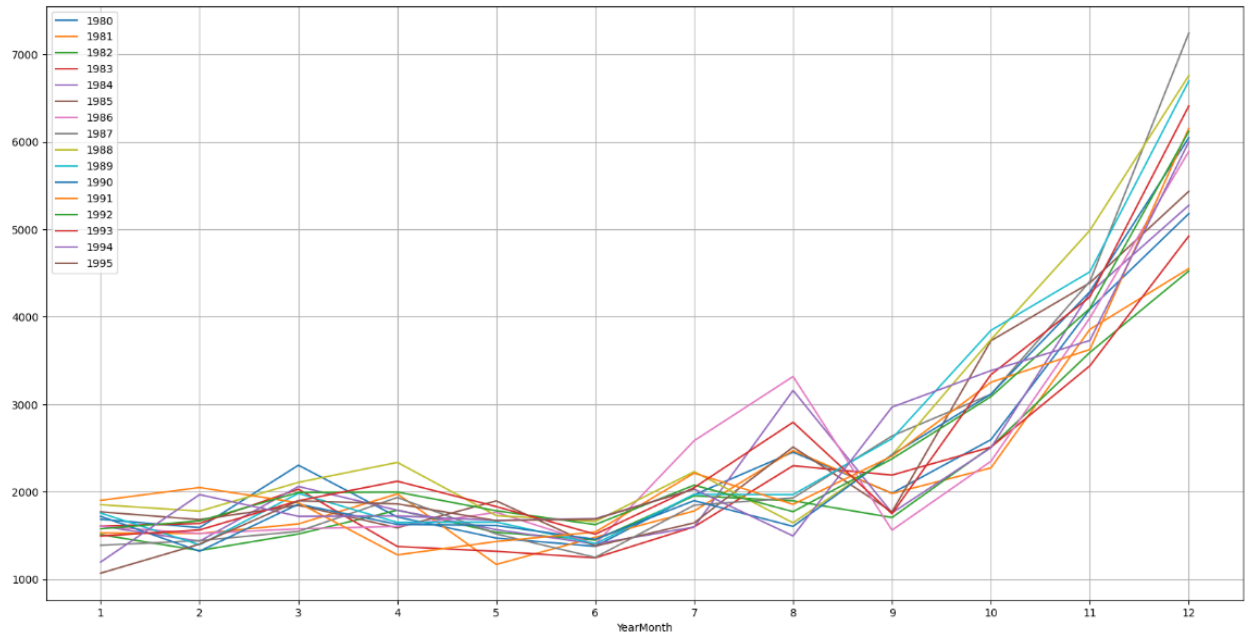
Figure 8 - Pivot Table II

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

The wine sales in the year 1988 show the highest accumulative sales. However, since the total sales for each year is not drastically different (higher or lower), there is no particular yearly trend showing in all months consistently.

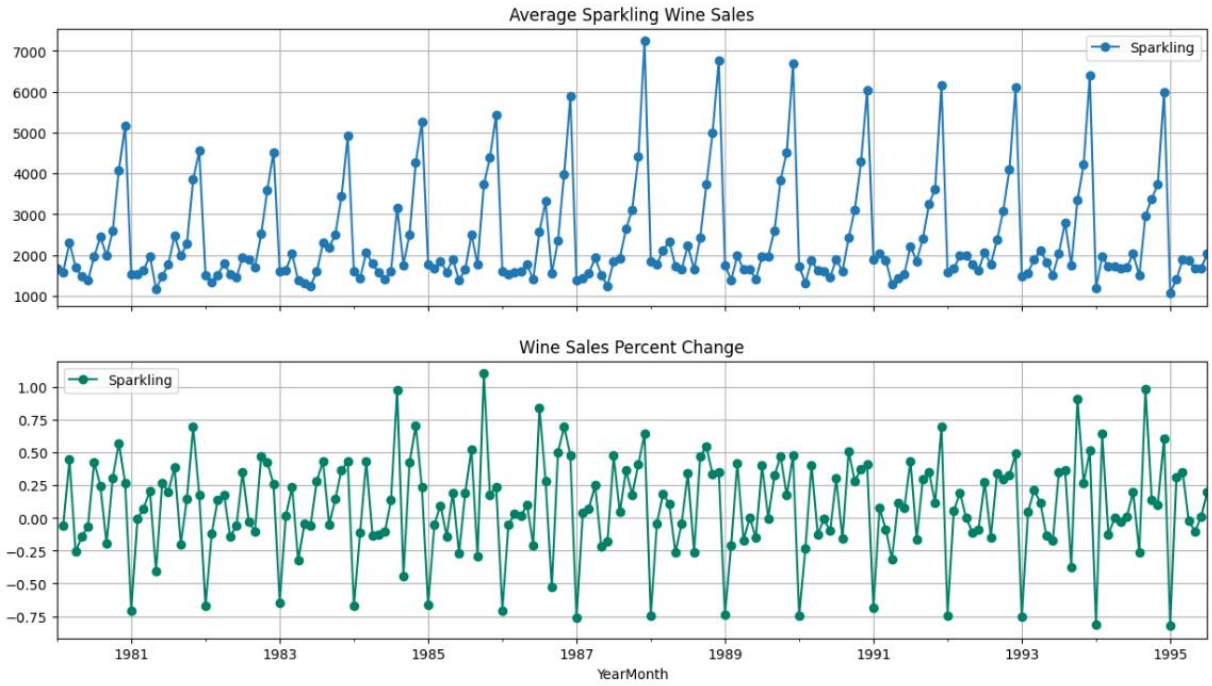
ix. We can see that the yearly data with month on the x-axis, no year stands out as consistently highest when compared month wise.

Figure 9 - Month Trend Plot with Year as Hue



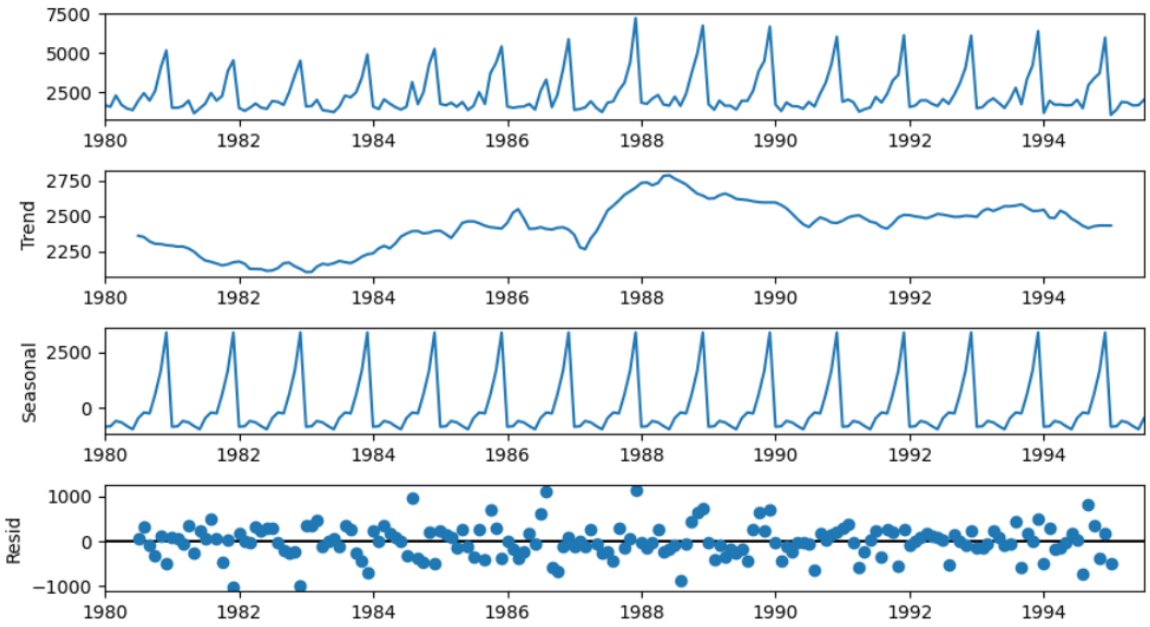
x. The following graphs show the average sales per month over the period from 1980-January to 1996-July and the percentage change in wine sales across the same timeline.

Figure 10 - Month on Month Average Wine Sales with Percent Change



xi. The original timeseries plot shows short-term seasonality and a long-term trend over the years. However, to confirm these characteristics, we need to decompose the timeseries. Below is the decomposed timeseries plot.

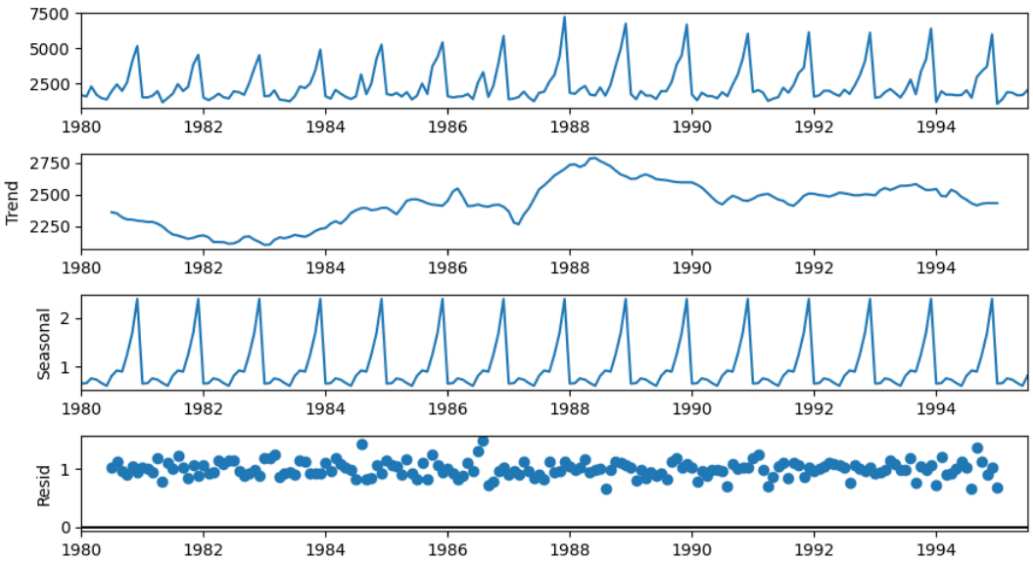
Figure 11 - Timeseries Additive Decomposition



The decomposed timeseries shows pronounced (additive) seasonality along with upward-to-flat trend from 1980 to 1995. The residual pattern also looks random. However, we will also decompose the timeseries to check if multiplicative seasonality suits the data better.

xii. Below is the timeseries decomposition plot with multiplicative seasonality.

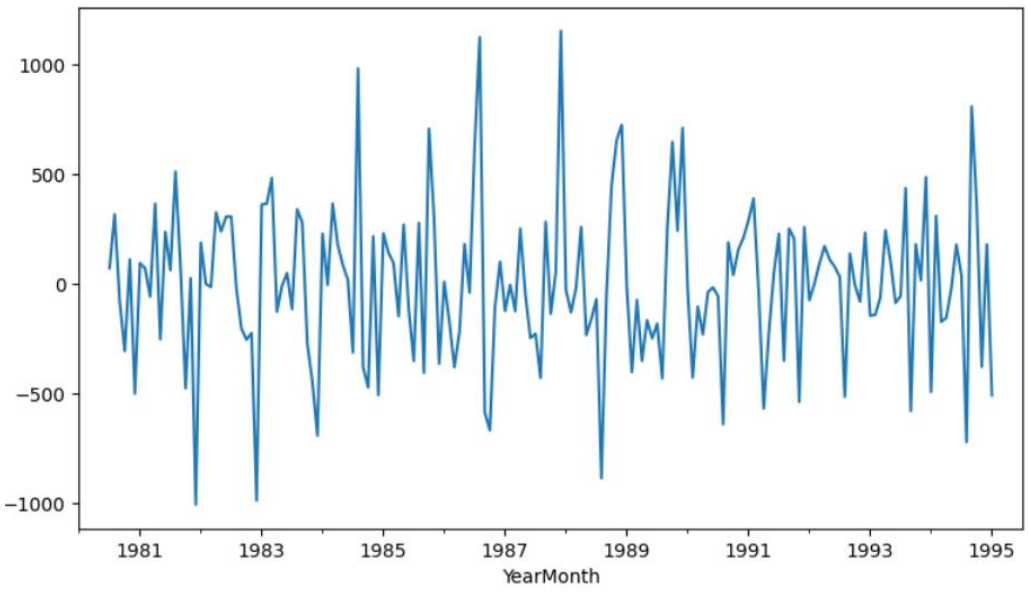
Figure 12 - Timeseries Multiplicative Decomposition



xiii. Checking the residual plots of both:

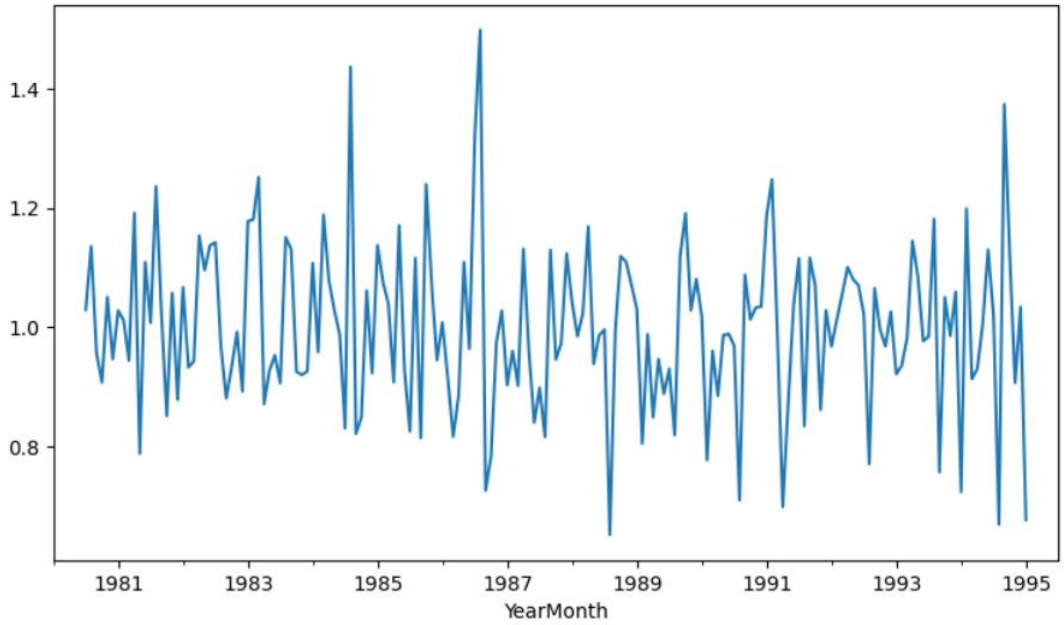
- Additive

Figure 13 - Residual Plot I



- Multiplicative

Figure 14 - Residual Plot 2



None of the residual errors have captured a pattern, which means the time series was decomposed perfectly into trend, seasonality, and residual error. We can default to additive seasonality in this case.

xiv. Example of decomposed elements of the timeseries in data frame format:

- Trend:

```
YearMonth
1980-06-01           NaN
1980-07-01    2360.666667
1980-08-01    2351.333333
1980-09-01    2320.541667
1980-10-01    2303.583333
                 ...
1994-10-01    2428.041667
1994-11-01    2433.583333
1994-12-01    2433.208333
1995-01-01    2433.000000
1995-02-01           NaN
Name: trend, Length: 177, dtype: float64
```

- Seasonality:

```
YearMonth
1980-01-01    0.649843
1980-02-01    0.659214
1980-03-01    0.757440
1980-04-01    0.730351
1980-05-01    0.660609
                 ...
1995-03-01    0.757440
1995-04-01    0.730351
1995-05-01    0.660609
1995-06-01    0.603468
1995-07-01    0.809164
Name: seasonal, Length: 187, dtype: float64
```

- Residuals:

```
YearMonth
1980-06-01         NaN
1980-07-01    1.029230
1980-08-01    1.135407
1980-09-01    0.955954
1980-10-01    0.907513
                ...
1994-10-01    1.122677
1994-11-01    0.906607
1994-12-01    1.033837
1995-01-01    0.676758
1995-02-01         NaN
Name: resid, Length: 177, dtype: float64
```

Formula for additive seasonality: **Observed = Trend + Seasonal + Irregular**

Formula for multiplicative seasonality: **Observed = Trend * Seasonality * Irregular**

3. Split the data into training and test. The test data should start in 1991. - 2 points

i. The data has been split into training and test set. The train set has 132 records while test set has 55 records.

Example of train set (the last five rows):

YearMonth	Sparkling
1990-08-01	1605
1990-09-01	2424
1990-10-01	3116
1990-11-01	4286
1990-12-01	6047

Example of test set (the first give rows):

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

4. Build all the exponential smoothing models on the training data and evaluate the model using RMSE on the test data.

Other models such as regression, naïve forecast models and simple average models. should also be built on the training data and check the performance on the test data using RMSE. - 16 points

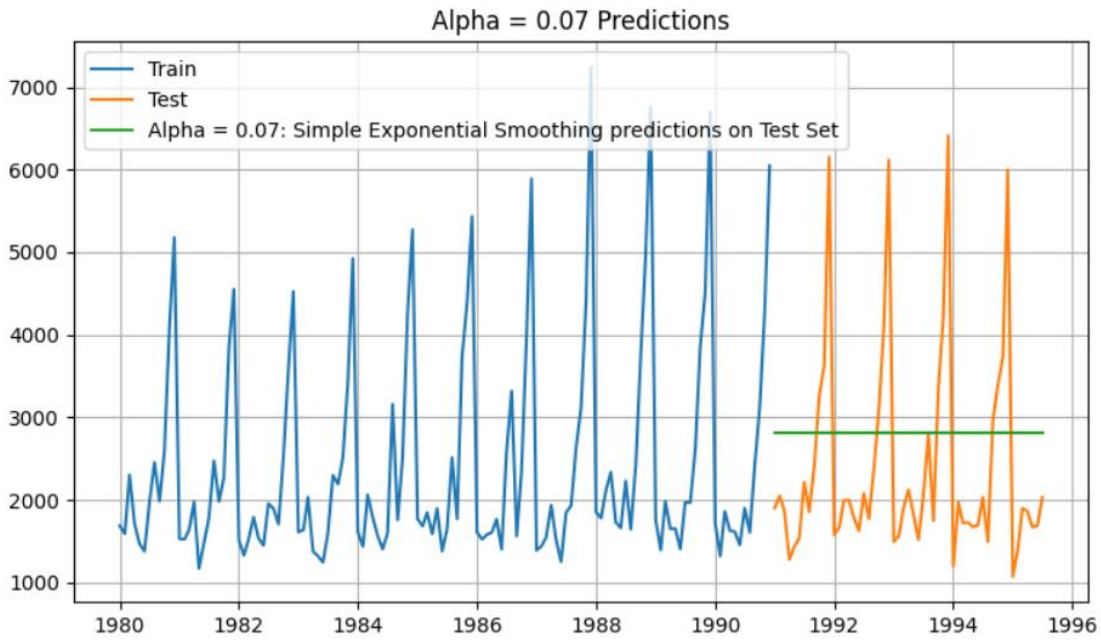
i. Simple Exponential Smoothing model:

- Prediction on test set (first five rows):

```
1991-01-01    2804.675124
1991-02-01    2804.675124
1991-03-01    2804.675124
1991-04-01    2804.675124
1991-05-01    2804.675124
Freq: MS, dtype: float64
```

- Plot for SES:

Figure 15 - SES plot



- This model only captures the level (Alpha), without including trend (Beta) and seasonality (Gamma) of the time series.
- The test **RMSE for this model is 1338**.

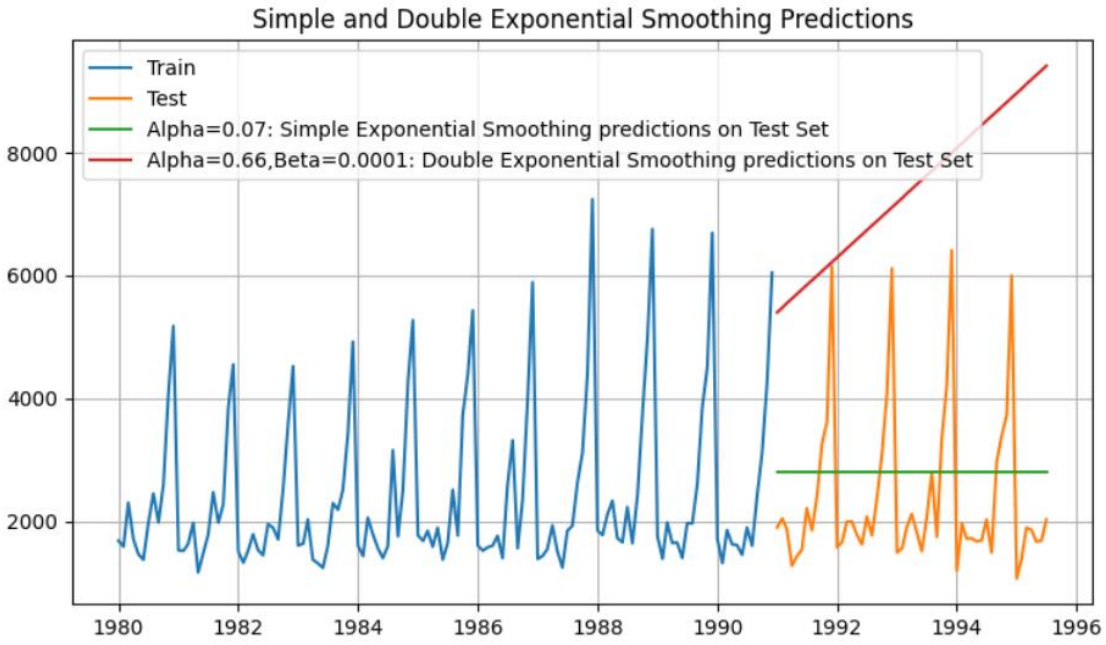
ii. Double Exponential Smoothing model:

- Prediction on test set (first five rows):

```
1991-01-01    5401.733026
1991-02-01    5476.005230
1991-03-01    5550.277433
1991-04-01    5624.549637
1991-05-01    5698.821840
Freq: MS, dtype: float64
```

- Plot for DES:

Figure 16 - DES plot



- This model captures only the level (Alpha) and the trend (Beta), but not seasonality (Gamma).
- The test **RMSE for this model is 5292**.

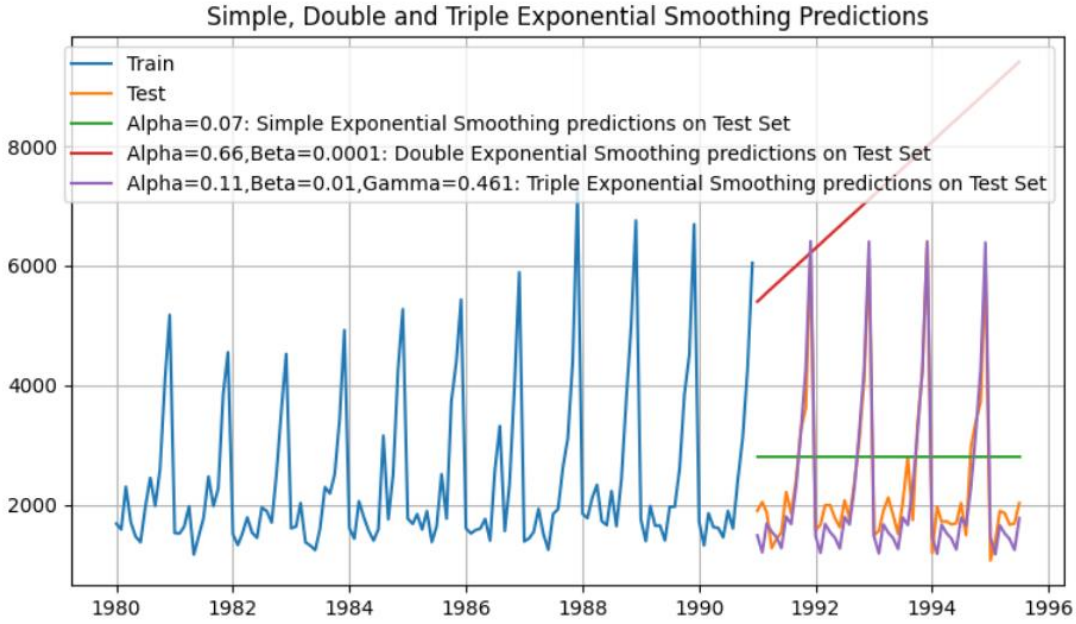
iii. Triple Exponential Smoothing model (seasonality parameter set to 'additive'):

- Prediction on test set (first five rows):

```
1991-01-01    1490.402890
1991-02-01    1204.525152
1991-03-01    1688.734182
1991-04-01    1551.226125
1991-05-01    1461.197883
Freq: MS, dtype: float64
```

- Plot for TES (Additive):

Figure 17 - TES(A) plot



- This model takes in account all three factors; level (Alpha), trend (Beta) and seasonality (Gamma). Since our time series has all these characteristics, the Triple Exponential Smoothing model will be able to best capture the movement of the data.
- The **RMSE of this model is 378.95**.

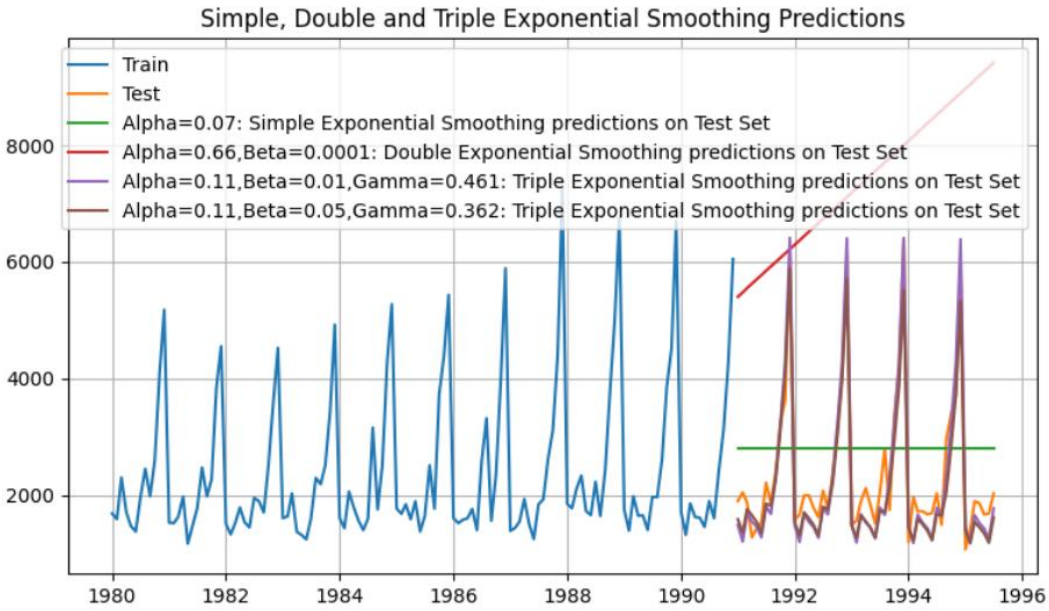
iv. Triple Exponential Smoothing model (seasonality parameter set to 'multiplicative'):

- Prediction on test set (first five rows):

```
1991-01-01    1587.497468
1991-02-01    1356.394925
1991-03-01    1762.929755
1991-04-01    1656.165933
1991-05-01    1542.002730
Freq: MS, dtype: float64
```

- Plot for TES (multiplicative):

Figure 18 - TES(M) plot



- This model is also Triple Exponential Smoothing where the seasonality parameter is set to 'multiplicative'.
- The test **RMSE of this model is 404.29**.

v. Regression on Time model:

- Creating time stamps on train and test data for the Linear Regression model

```
Train Time instance
 [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,
65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95,
96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 12
1, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132]


Test Time instance
 [133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 15
7, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 1
82, 183, 184, 185, 186, 187]
```

Example rows:

Train

YearMonth	Sparkling	Time
1980-01-01	1686	1
1980-02-01	1591	2
1980-03-01	2304	3
1980-04-01	1712	4
1980-05-01	1471	5

Test

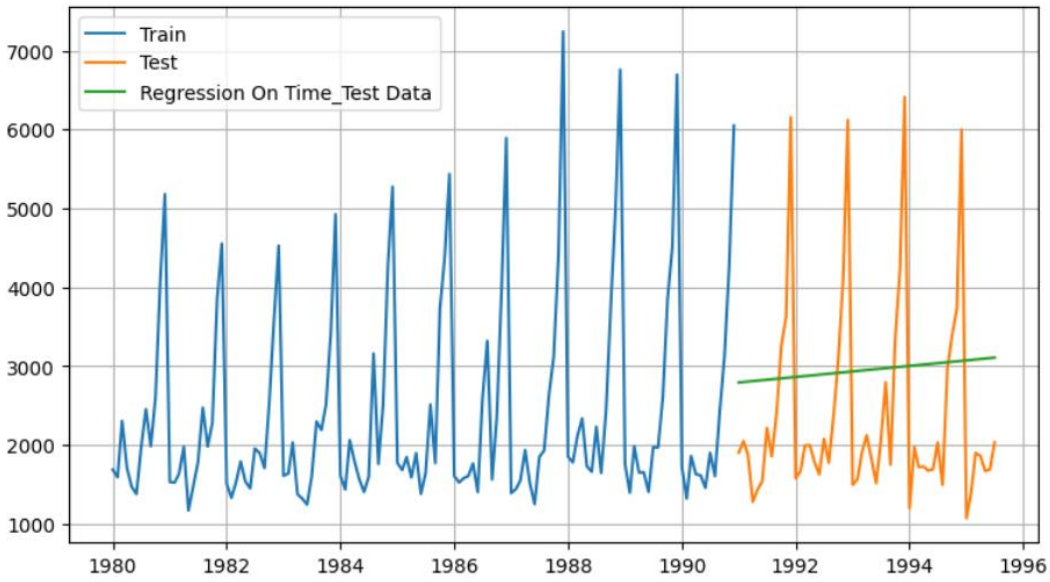
YearMonth	Sparkling	Time
1991-01-01	1902	133
1991-02-01	2049	134
1991-03-01	1874	135
1991-04-01	1279	136
1991-05-01	1432	137

- Predictions on test set

```
array([2791.65209345, 2797.48475196, 2803.31741046, 2809.15006896,
       2814.98272746, 2820.81538597, 2826.64804447, 2832.48070297,
       2838.31336147, 2844.14601998, 2849.97867848, 2855.81133698,
       2861.64399548, 2867.47665399, 2873.30931249, 2879.14197099,
       2884.9746295 , 2890.807288  , 2896.6399465 , 2902.472605  ,
       2908.30526351, 2914.13792201, 2919.97058051, 2925.80323901,
       2931.63589752, 2937.46855602, 2943.30121452, 2949.13387302,
       2954.96653153, 2960.79919003, 2966.63184853, 2972.46450703,
       2978.29716554, 2984.12982404, 2989.96248254, 2995.79514104,
       3001.62779955, 3007.46045805, 3013.29311655, 3019.12577506,
       3024.95843356, 3030.79109206, 3036.62375056, 3042.45640907,
       3048.28906757, 3054.12172607, 3059.95438457, 3065.78704308,
       3071.61970158, 3077.45236008, 3083.28501858, 3089.11767709,
       3094.95033559, 3100.78299409, 3106.61565259])
```

- Plot for predictions made by Regression on Time model:

Figure 19 - RegOnTime plot



- The **RMSE for this model is 1389**.

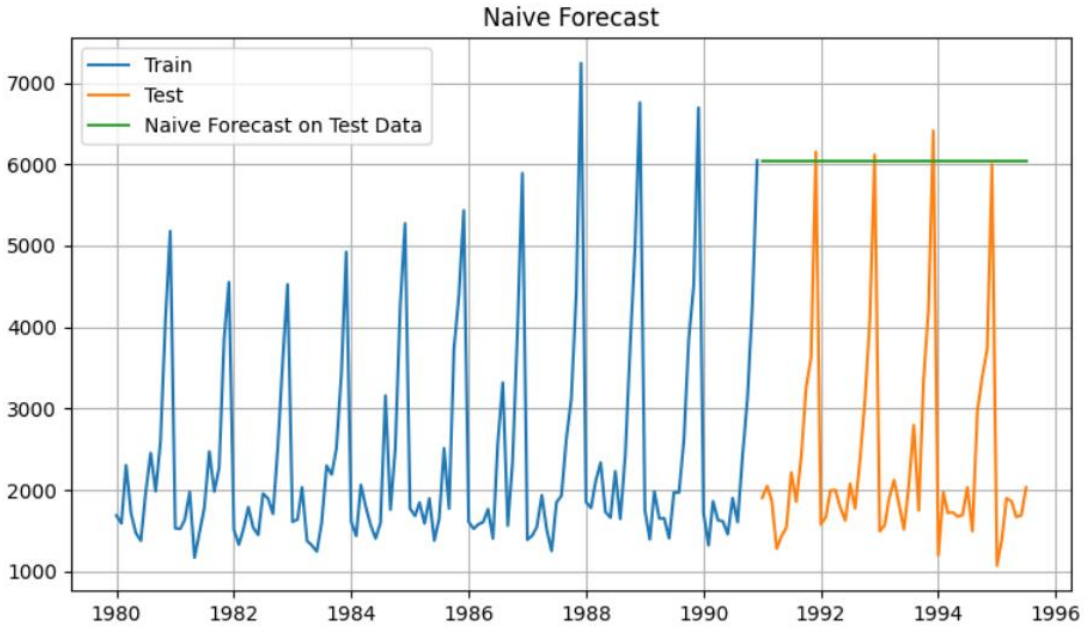
vi. Naïve Forecast model:

- Test predictions

```
YearMonth
1991-01-01    6047
1991-02-01    6047
1991-03-01    6047
1991-04-01    6047
1991-05-01    6047
Name: Naive, dtype: int64
```

- Plot for the Naïve Forecast model:

Figure 20 - Naive Forecast plot



- The **RMSE of the model is 3864**.

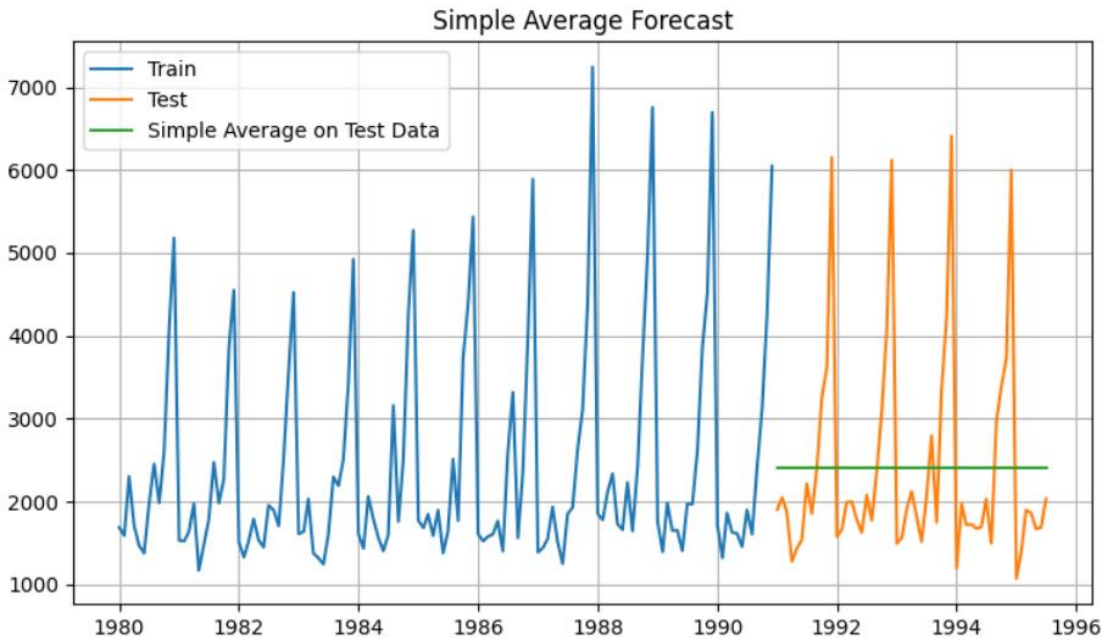
vii. Simple Moving Average model:

Sample predictions on the Test data

YearMonth	Sparkling	Mean_Forecast
1991-01-01	1902	2403.780303
1991-02-01	2049	2403.780303
1991-03-01	1874	2403.780303
1991-04-01	1279	2403.780303
1991-05-01	1432	2403.780303

- Plot for the model

Figure 21 - Simple Average plot



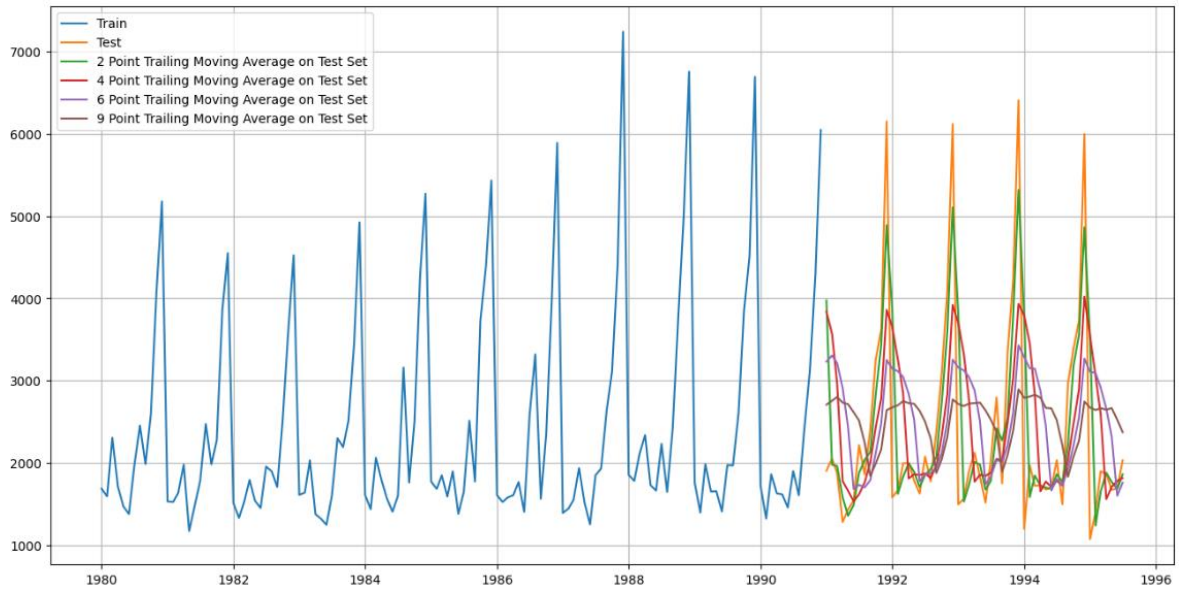
- The **RMSE for this model is 1275.**

viii. Moving Average model with rolling mean of 2, 4, 6, 9 on Test data:

YearMonth	Sparkling	Trailing_2	Trailing_4	Trailing_6	Trailing_9
1980-01-01	1686	NaN	NaN	NaN	NaN
1980-02-01	1591	1638.5	NaN	NaN	NaN
1980-03-01	2304	1947.5	NaN	NaN	NaN
1980-04-01	1712	2008.0	1823.25	NaN	NaN
1980-05-01	1471	1591.5	1769.50	NaN	NaN
1980-06-01	1377	1424.0	1716.00	1690.166667	NaN
1980-07-01	1966	1671.5	1631.50	1736.833333	NaN
1980-08-01	2453	2209.5	1816.75	1880.500000	NaN
1980-09-01	1984	2218.5	1945.00	1827.166667	1838.222222
1980-10-01	2596	2290.0	2249.75	1974.500000	1939.333333

- Plot for Moving Average model:

Figure 22 - Moving Averages plot



- The RMSE for each rolling mean is as given below:

  RMSE for 2pointTrailingMovingAverage is 813

  RMSE for 4pointTrailingMovingAverage is 1157

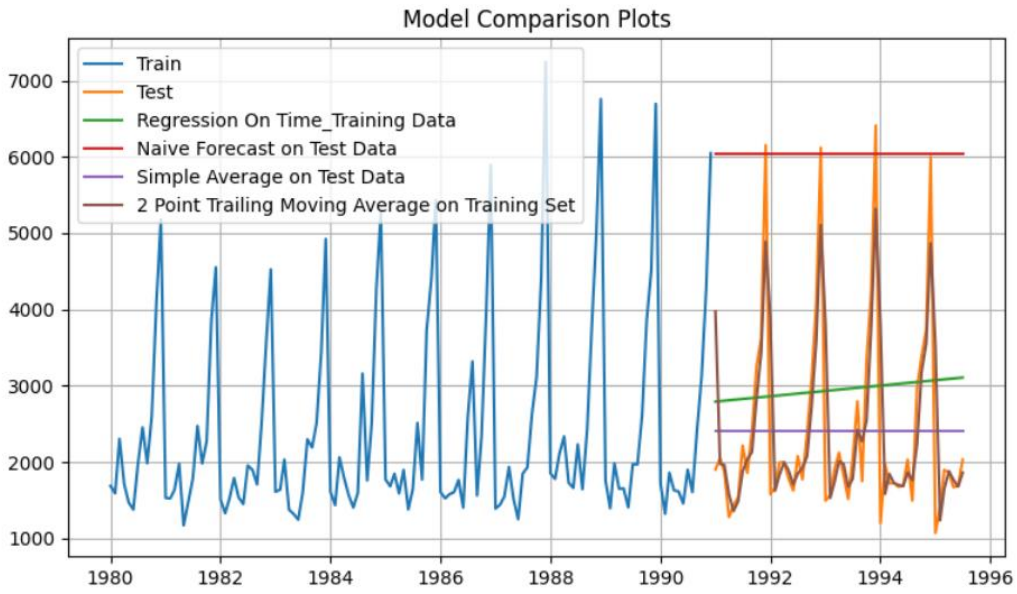
  RMSE for 6pointTrailingMovingAverage is 1284

  RMSE for 9pointTrailingMovingAverage is 1346

- The best performing is the 2-point trailing Moving Average.

- Below is the plot that compares the 2-point trailing average with all other models built previously.

Figure 23 - Model Comparison



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. - 4 points

i. Dicky-Fuller Test is used to check if a time series is stationary or non-stationary.

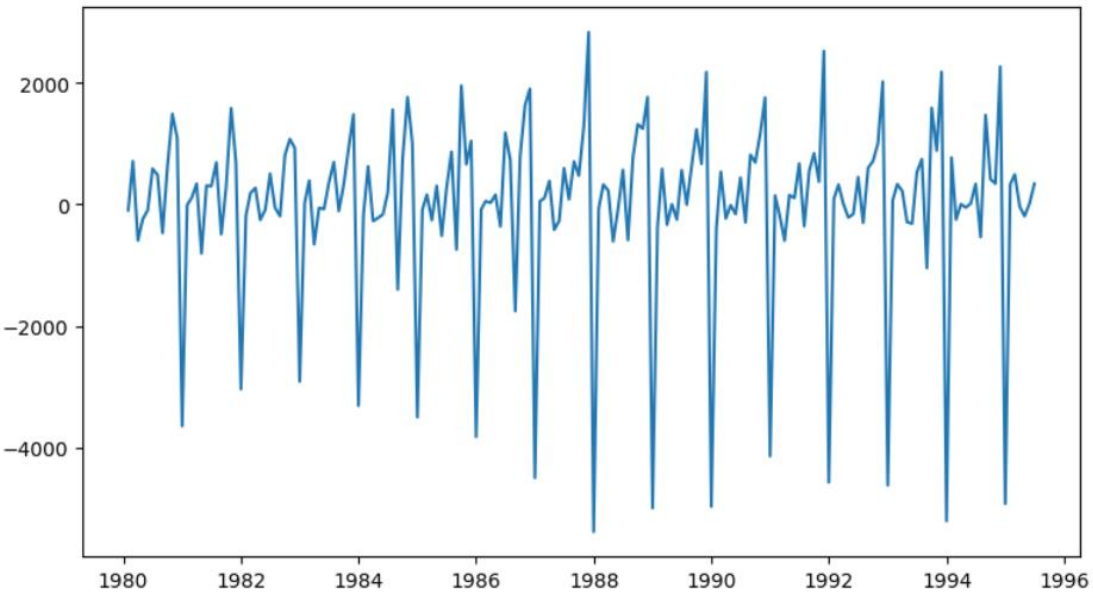
- Null Hypothesis => Time Series is non-stationary.
- Alternate Hypothesis => Time Series is stationary.

ii. The p-value given by the Dicky-Fuller test for this timeseries is 0.601, which is higher than 0.05 significance level, hence failing to reject the null hypothesis that the time series is non-stationary.

It is concluded by the Dicky-Fuller Test that the time series is non-stationary.

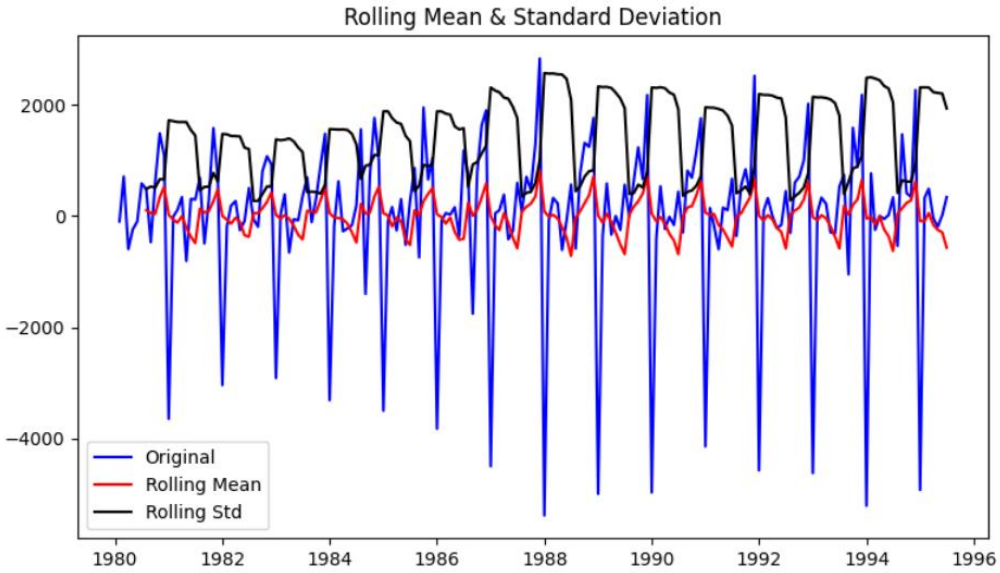
iii. Converting the timeseries into stationary by differencing once. After this step, the timeseries plot now looks like the following:

Figure 24 - Timeseries Plot after Differencing with 'd' = 1



iv. After differencing the time series, the p-value is 0.0000 which is much lower than 0.05. Therefore, we can successfully reject the null hypothesis that the time series is non-stationary.

Figure 25 - Dicky-Fuller Test

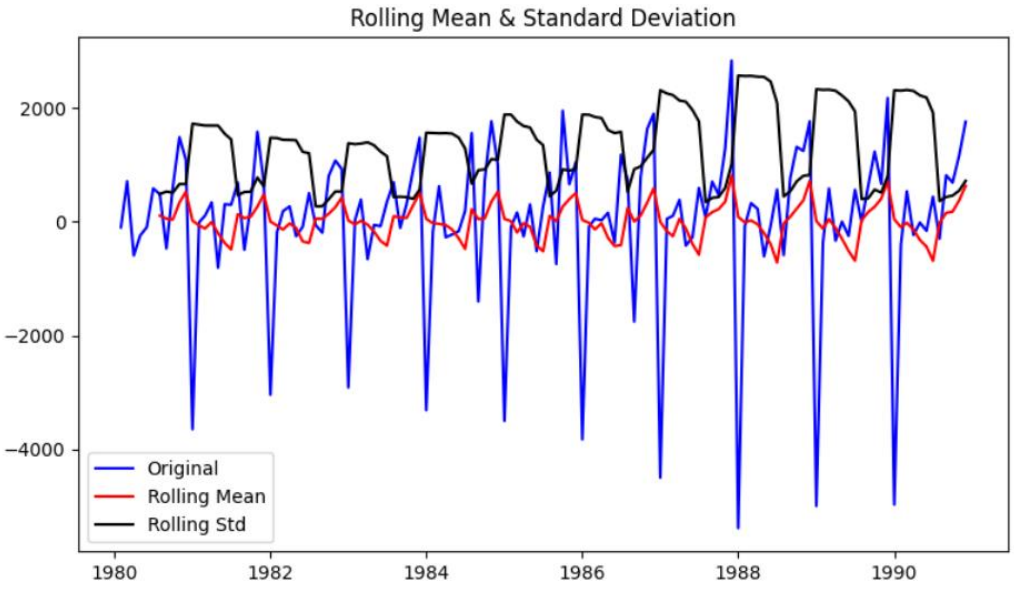



```
Results of Dickey-Fuller Test:
Test Statistic                -45.050301
p-value                         0.000000
#Lags Used                     10.000000
Number of Observations Used    175.000000
Critical Value (1%)             -3.468280
Critical Value (5%)             -2.878202
Critical Value (10%)            -2.575653
dtype: float64
```

The time series is now 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. - 11 points

i. To start building the ARIMA/SARIMA models, we first need to convert the timeseries train data into stationary. The plot for train data after converting it into stationary with differencing (d=1) is given below:




```
Results of Dickey-Fuller Test:
Test Statistic                -8.005007e+00
p-value                        2.280104e-12
#Lags Used                     1.100000e+01
Number of Observations Used    1.190000e+02
Critical Value (1%)           -3.486535e+00
Critical Value (5%)           -2.886151e+00
Critical Value (10%)          -2.579896e+00
dtype: float64
```

- The p-value is 2.280104e-12, i.e., lower than 0.05, thus making the timeseries train data stationary.

ii. ARIMA model:

- Creating different value combinations for p, d, q as shown below.

```
[(1, 0, 1),
 (1, 0, 2),
 (1, 0, 3),
 (2, 0, 1),
 (2, 0, 2),
 (2, 0, 3),
 (3, 0, 1),
 (3, 0, 2),
 (3, 0, 3)]
```

NOTE: I initially tried a higher range of values, but my computer was not equipped well enough to process it and generate the AIC values with different combinations. Due to this limitation, I limited my parameter values.

- Generated and sorted AIC (Akaike Information Criteria) in ascending order for each combination as shown below.

	param	AIC
5	(2, 0, 3)	2205.771123
8	(3, 0, 3)	2209.378707
7	(3, 0, 2)	2235.191616
3	(2, 0, 1)	2236.590860
2	(1, 0, 3)	2242.106123
0	(1, 0, 1)	2246.005400
1	(1, 0, 2)	2246.935700
4	(2, 0, 2)	2248.277281
6	(3, 0, 1)	2248.562804

- Since (p,d,q) = (2,0,3) is giving the lowest AIC, this set of parameters will be used to build the auto ARIMA model.

- Auto ARIMA model with (p,d,q) = (2,0,3) is shown below.

Figure 26 - ARIMA model summary

```
                               SARIMAX Results
==========================================================================================
Dep. Variable:                      Sparkling   No. Observations:                  132
Model:                 ARIMA(2, 0, 3)   Log Likelihood               -1095.886
Date:                Sun, 04 Jun 2023   AIC                             2205.771
Time:                        18:58:22   BIC                             2225.951
Sample:                    01-01-1980   HQIC                            2213.971
                         - 12-01-1990
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const       2403.7508    131.266     18.312      0.000    2146.475    2661.026
ar.L1          1.7076      0.047     36.679      0.000       1.616       1.799
ar.L2         -0.9713      0.047    -20.771      0.000      -1.063      -0.880
ma.L1         -1.5874      0.181     -8.757      0.000      -1.943      -1.232
ma.L2          0.5615      0.321      1.750      0.080      -0.067       1.190
ma.L3          0.2291      0.205      1.120      0.263      -0.172       0.630
sigma2       1.18e+06      0.001   1.14e+09      0.000    1.18e+06    1.18e+06
===================================================================================
Ljung-Box (L1) (Q):                   0.21   Jarque-Bera (JB):                33.98
Prob(Q):                              0.65   Prob(JB):                         0.00
Heteroskedasticity (H):               2.25   Skew:                             0.80
Prob(H) (two-sided):                  0.01   Kurtosis:                         4.90
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 9.03e+25. Standard errors may be unstable.
```

- The **RMSE for this model is 1029**.

iii. SARIMA model:

- Creating different value combinations for P, D, Q as shown below with the seasonal factor of 12, as the data is monthly.

```
[(1, 0, 1, 12),
 (1, 0, 2, 12),
 (1, 0, 3, 12),
 (2, 0, 1, 12),
 (2, 0, 2, 12),
 (2, 0, 3, 12),
 (3, 0, 1, 12),
 (3, 0, 2, 12),
 (3, 0, 3, 12)]
```

- Generated and sorted AIC (Akaike Information Criteria) in ascending order for each combination as shown below.

	param	seasonal	AIC
81	(3, 0, 2)	(2, 0, 3, 12)	691.883567
82	(3, 0, 2)	(3, 0, 1, 12)	1389.689536
73	(3, 0, 1)	(3, 0, 1, 12)	1390.195272
83	(3, 0, 2)	(3, 0, 2, 12)	1390.696386
74	(3, 0, 1)	(3, 0, 2, 12)	1391.067066
...	...	...	...
24	(1, 0, 2)	(1, 0, 3, 12)	6762.435432
78	(3, 0, 2)	(1, 0, 3, 12)	6764.469133
90	(3, 0, 3)	(2, 0, 3, 12)	6766.893568
63	(2, 0, 3)	(2, 0, 3, 12)	6797.517564
69	(3, 0, 1)	(1, 0, 3, 12)	6833.539250

- Since (p,d,q) = (3,0,2) and (P,D,Q,F) = (2,0,3,12) is giving the lowest AIC, this set of parameters will be used to build the auto SARIMA model. However, on trying this parameter combination, the RMSE generated was huge, i.e., 1119563837705146.1.

- For this reason, multiple parameter combinations were tried and the optimal performance was given by the model with parameters (p,d,q) = (3,0,2) and (P,D,Q,F) = (3,0,1,12) with RMSE of 2913.

- The final auto SARIMA model is shown below.

Figure 27 - SARIMA model summary

```
                                     SARIMAX Results
==========================================================================================
Dep. Variable:                           Sparkling   No. Observations:                  131
Model:             SARIMAX(3, 0, 2)x(3, 0, [1], 12)   Log Likelihood                -684.845
Date:                              Sun, 04 Jun 2023   AIC                           1389.690
Time:                                      18:58:40   BIC                           1414.907
Sample:                                  02-01-1980   HQIC                          1399.868
                                       - 12-01-1990
Covariance Type:                               opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1         -0.5225      0.449     -1.165      0.244      -1.402       0.357
ar.L2         -0.0315      0.198     -0.159      0.874      -0.421       0.358
ar.L3          0.0630      0.142      0.443      0.658      -0.216       0.341
ma.L1         -0.2228      0.440     -0.507      0.612      -1.084       0.639
ma.L2         -0.6148      0.360     -1.708      0.088      -1.320       0.091
ar.S.L12       0.7275      0.436      1.667      0.095      -0.128       1.583
ar.S.L24       0.1392      0.325      0.428      0.669      -0.498       0.777
ar.S.L36       0.1866      0.178      1.046      0.295      -0.163       0.536
ma.S.L12      -0.2055      0.434     -0.474      0.636      -1.055       0.644
sigma2      1.686e+05   2.61e+04      6.464      0.000    1.17e+05     2.2e+05
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):                 8.66
Prob(Q):                              0.98   Prob(JB):                         0.01
Heteroskedasticity (H):               1.22   Skew:                             0.30
Prob(H) (two-sided):                  0.59   Kurtosis:                         4.38
===================================================================================

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

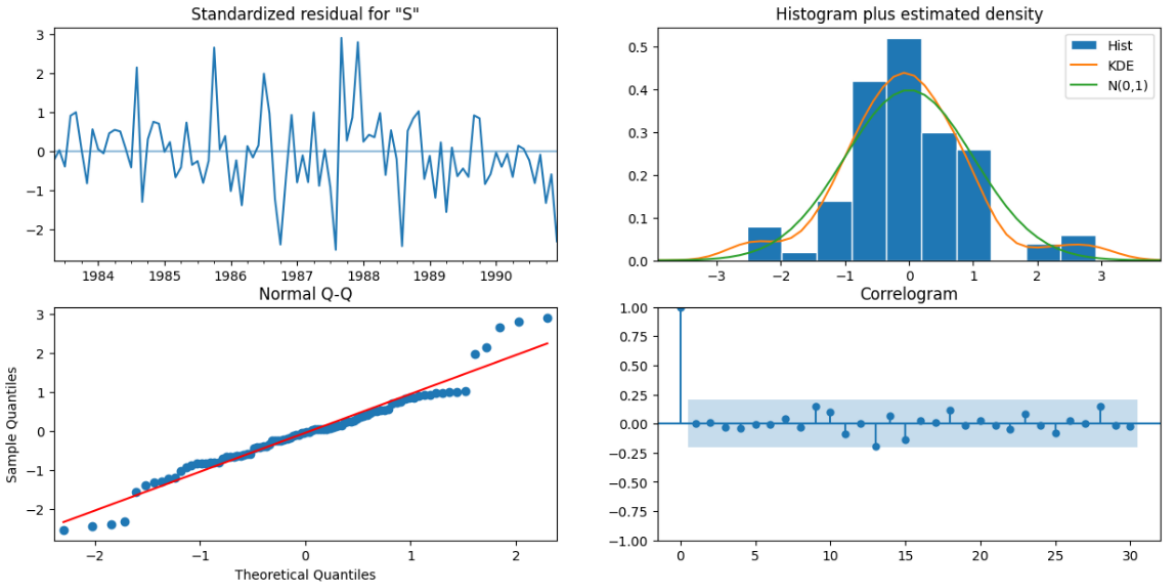
- The following is the summary frame for the model:

Sparkling	mean	mean_se	mean_ci_lower	mean_ci_upper
1991-01-01	-4736.610769	410.623179	-5541.417410	-3931.804128
1991-02-01	-37.915558	512.123774	-1041.659711	965.828596
1991-03-01	308.665232	522.868386	-716.137973	1333.468437
1991-04-01	6.906194	530.657086	-1033.162583	1046.974971
1991-05-01	-216.024663	534.417131	-1263.462993	831.413668

- The **RMSE for this model is 2913**.

- Model diagnostics are shown below in the form of plots.

Figure 28 - Diagnostics plot for SARIMA model



- Observations:
    - The top left plot, The Standardized Residual plot shows 1-step-ahead standardized residuals. To verify if the model has been built correctly, there should be NO obvious pattern for the residuals in this plot, which is the case in the above graph as well.
    - Normal Q-Q plot shows compares the distribution of residuals to normal distribution. If the distribution of the residuals is normal, then all the points should lie along the red line, except for some values at the end. In the above graph, the residual is fairly matching the ideal Q-Q plot for a good model.
    - The correlogram plot is the ACF plot of the residuals instead of that of the data. 95% of the correlations for lag >0 (our lags parameter is set to lags=30) should not be significant, shown within the blue shade. If there is a significant correlation in the residuals, it means that there is information in the data that was not captured by the model. But in our Correlogram, that is not the case. This signifies that our model is built well.
    - Lastly, the Histogram plot with KDE (orange) is to check if the residuals have a normal distribution with N(0,1) (green). The distribution of the residual is fairly consistent with a normal distribution.

7. Build a table (create a data frame) with all the models built along with their corresponding parameters and the respective RMSE values on the test data. - 2 points

Figure 29 - Table of All the Models Built Along with Their RMSE and Parameters

	Test RMSE	p	d	q	P	D	Q	F
Alpha=0.07; SES	1338.008384	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Alpha=0.66, Beta=0.0001; DES	5291.879833	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Alpha=0.11, Beta=0.01, Gamma=0.461; TES	378.951023	NaN	NaN	NaN	NaN	NaN	NaN	NaN
RegressionOnTime	1389.135175	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaiveModel	3864.279352	NaN	NaN	NaN	NaN	NaN	NaN	NaN
SimpleAverageModel	1275.081804	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2pointTrailingMovingAverage	813.400684	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4pointTrailingMovingAverage	1156.589694	NaN	NaN	NaN	NaN	NaN	NaN	NaN
6pointTrailingMovingAverage	1283.927428	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9pointTrailingMovingAverage	1346.278315	NaN	NaN	NaN	NaN	NaN	NaN	NaN
AutoARIMAModel(2,1,3)	1028.539360	2.0	1.0	3.0	NaN	NaN	NaN	NaN
AutoSARIMAModel(3,0,1,12)	2913.319642	3.0	0.0	2.0	3.0	0.0	1.0	12.0

- The above table shows all the relevant parameters as well as the corresponding RMSE for each model built. Clearly, the Triple Exponential Smoothing model has outperformed all others, giving the lowest Root Squared Mean Error.

8. Based on the model-building exercise, build the most optimum model(s) on the complete data and predict 12 months into the future with appropriate confidence intervals/bands. - 3 points

i. I have chosen to go with Triple Exponential Smoothing model as it has performed the best during the model building exercise.

ii. First, I have trained the model on the full data available in the timeseries. The model summary is given below.

Figure 30 - FINAL MODEL - Triple Exponential Smoothing (Additive) Model

```
{'smoothing_level': 0.07596713146311772,
 'smoothing_trend': 0.03256921715086211,
 'smoothing_seasonal': 0.37660762886165167,
 'damping_trend': nan,
 'initial_level': 2356.500087258337,
 'initial_trend': -0.8449338106956193,
 'initial_seasons': array([-636.25474139, -723.00153617, -398.66964817, -473.45571811,
        -808.43306854, -815.37001157, -384.24814771,   73.00119661,
        -237.46281546,  272.34574748, 1541.39349329, 2590.11477306]),
 'use_boxcox': False,
 'lamda': None,
 'remove_bias': False}
```

iii. Then, I have created an empty data frame with monthly time stamps for the next 12 months.

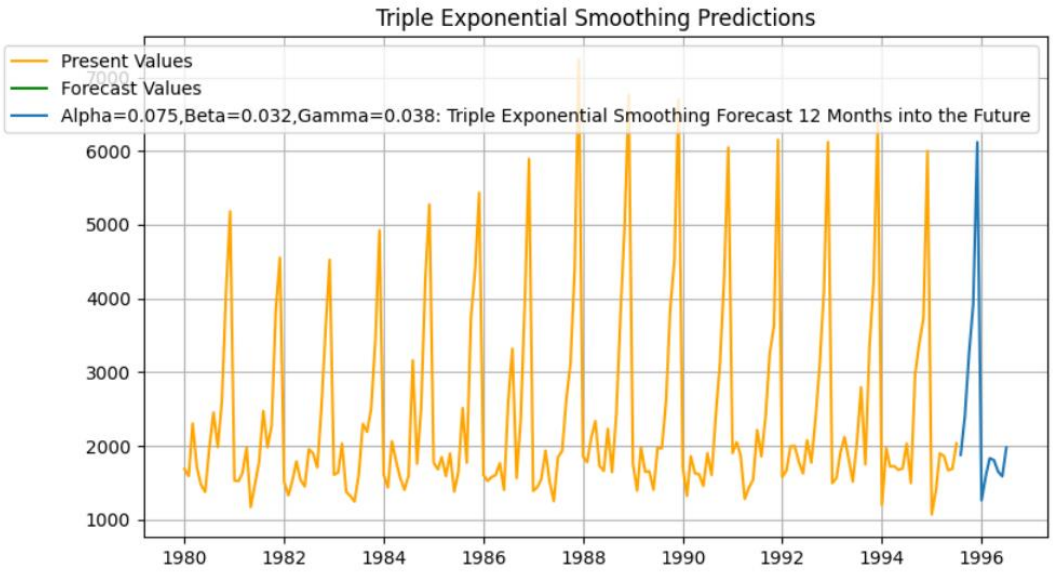
	Sparkling
1995-08-31	None
1995-09-30	None
1995-10-31	None
1995-11-30	None
1995-12-31	None
1996-01-31	None
1996-02-29	None
1996-03-31	None
1996-04-30	None
1996-05-31	None
1996-06-30	None
1996-07-31	None

iv. I have used my model trained on the full data to predict values for the above MonthYear as shown below.

```
1995-08-01    1877.431801
1995-09-01    2405.285747
1995-10-01    3242.105698
1995-11-01    3922.189524
1995-12-01    6118.502404
1996-01-01    1262.618990
1996-02-01    1592.137914
1996-03-01    1831.652945
1996-04-01    1806.470072
1996-05-01    1651.723185
1996-06-01    1586.507708
1996-07-01    1977.014975
Freq: MS, dtype: float64
```

v. The following represents this forecast in the visual format of a plot:

Figure 31 - FINAL MODEL Forecast Plot



9. 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. - 9 points

Finds and Suggestions for the Company:

i. The Triple Exponential Smoothing model (additive) performed the best.

ii. Following are the predictions for the next 12 months based on monthly interval for Sparkling wine sales:

```
1995-08-01    1877.431801
1995-09-01    2405.285747
1995-10-01    3242.105698
1995-11-01    3922.189524
1995-12-01    6118.502404
1996-01-01    1262.618990
1996-02-01    1592.137914
1996-03-01    1831.652945
1996-04-01    1806.470072
1996-05-01    1651.723185
1996-06-01    1586.507708
1996-07-01    1977.014975
Freq: MS, dtype: float64
```

iii. December is the best performing month in the original timeseries and according to the forecast values. As the winters approach, Sparkling wine begins to see higher and higher sales volume.

iv. The business should be aware of this trend since it could point out to the fact that people buy more wine when festivities are involved.

v. To leverage this trend, the business can create festive campaigns to promote wine sales in the summer season, to replicate winter season sales trends. This could lead to more profits as the summer sales go up just like the winter sales.

Steps Taken to Complete this Project:

i. Step 1: Plotted the time series to get a sense of its trend and seasonality, and whether they are present or not.

ii. Step 2: Did basic checks like convert dates into their appropriate datetime format, set it as index, check null values and describe the data.

iii. Step 3: Performed EDA on the data to see monthly as well as annual sales trend for the wine.

iv. Step 3 Decomposed the series to clearly segregate its trend, seasonality and residual component.

v. Step 4: Split the timeseries into train and test set as a preparation for building various models on it.

vi. Step 5: Built all Exponential Smoothing models along with Regression on Time, Naïve Forecast, Simple Average and Moving Averages models with 2, 4, 6, 9 rolling mean values.

vii. Step 6: Built a table comparing the RMSE of all the models built till the last step to get a feel of which direction the project is moving.

viii. Step 7: Checked the stationarity of the timeseries with the help of the Dicky-Fuller Test.

ix. Step 8: In this case, since the timeseries was not stationary, converted it into stationary timeseries by differencing once. This step is a prerequisite for building ARIMA and SARIMA models.

x. Step 9: Built the auto ARIMA and auto SARIMA models by generating various combinations of required parameters like p, d, q, P, D, Q, F that produced different AIC scores. The parameter combinations with the lowest AIC scores were taken to build the final models.

xi. Step 10: It is worth noting that the lowest AIC was not always giving the best parameter combinations. Therefore, another step was taken where different parameter values were tried manually to see which model gave the best RMSE. Other supporting elements like summary frame and diagnostics plots were created to make the understanding of the model more robust.

xii. Step 11: Finally, a final table was built to compare the performance of ALL models from start to finish.

xiii. Step 12: Triple Exponential Smoothing model was chosen based on the best RMSE and built.

xiv. Step 13: An empty data frame with time stamps for the next 12 months was created for the purpose of forecasting with the final model.

xv. Step 14: The TES model was trained on the full dataset and predictions were generated on the empty data frame created with future time stamps. Other supporting visuals were also created to understand the forecast values better.

-----THE DOCUMENT ENDS HERE-----