

**CAPSTONE PROJECT**

**SALES FORECASTING REPORT**

|  |  |
| --- | --- |
| **Author:** | Ghayathri Vasudevan |
| **Date of Submission:** | 12/08/2021 |
| **Version:** | 1.0 |
| **Batch:** | PGPDSBA.O.AUG20\_B |

Contents

[1. Introduction 3](#_Toc82380430)

[2. EDA and Business Implication. 3](#_Toc82380431)

[Univariate Analysis 3](#_Toc82380432)

[Multivariate and Bivariate Analysis 5](#_Toc82380433)

[3. Data Cleaning and Pre-processing 11](#_Toc82380434)

[Identifying Missing values 11](#_Toc82380435)

[Identifying outliers: 11](#_Toc82380436)

[Variable transformation 12](#_Toc82380437)

[Variables Added or Removed 12](#_Toc82380438)

[4. Model building 12](#_Toc82380439)

[Choosing the Right Model 13](#_Toc82380440)

[Product Based Model 13](#_Toc82380441)

[City Based Model 21](#_Toc82380442)

[Efforts to improve model performance 30](#_Toc82380443)

[5. Model validation 31](#_Toc82380444)

[Model validation on Products 31](#_Toc82380445)

[Model validation on City models 35](#_Toc82380446)

[6. Final interpretation / recommendation 40](#_Toc82380447)

[APPENDIX 42](#_Toc82380448)

[Building various models 42](#_Toc82380449)

[Product Based models 42](#_Toc82380450)

[City Based Model 62](#_Toc82380451)

[Model Evaluation 95](#_Toc82380452)

[Product Based Models 96](#_Toc82380453)

[City Based Models 100](#_Toc82380454)

# ****Introduction****

The company Consumer goods appliances is Interested In forecasting the sales of different product sold in different cities across India. Data analyst role is to understand the data by performing exploratory data analysis and building models to draw business insights/strategies that needs to be implemented across different cities of India to tell the business where the sales needs to be improved or where the sales is high and recommend them with suggestions on their business. The fundamental goal of as an data analyst is to help companies make quicker and better decisions, which can take them to the top of their market, or at least – especially in the toughest red oceans – be a matter of long-term survival.

# ****EDA and Business Implication.****

## Univariate Analysis

Univariate analysis refer to the analysis of a single variable. The main purpose of univariate analysis is to summarize and find patterns in the data. The key point is that there is only one variable involved in the analysis.

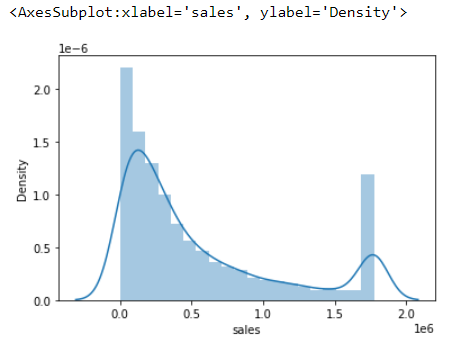
**Histogram plot:**

The bins argument creates class intervals. In this case we are creating 30 such intervals.

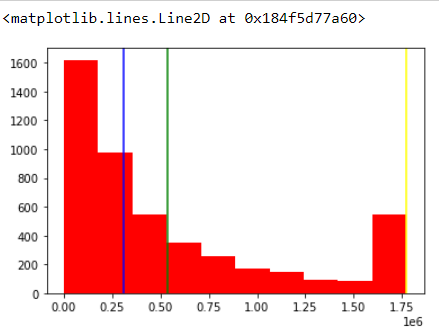
****

In the above histogram, the first array is the frequency in each class and the second array contains the edges of the class intervals. These arrays can be assigned to a variable and used for further analysis

**Distplot:**

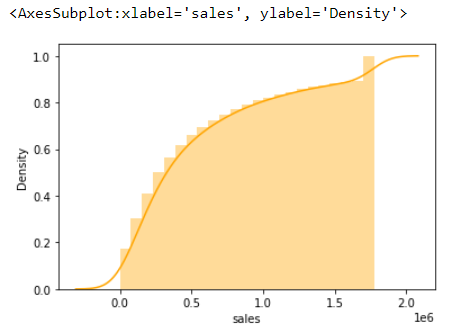
Plots a frequency polygon superimposed on a histogram using the seaborn package, seaborn automatically creates class intervals.

Now let us have a closer look at the distribution by plotting a simple histogram with 30 bins.



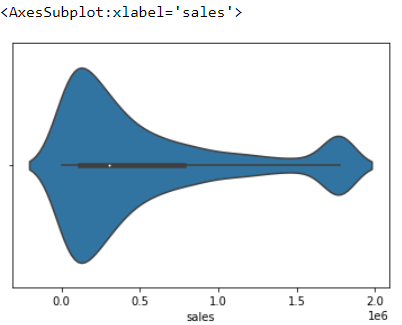
The mode is represented as yellow line, mean is represented in green line and median is represented in blue line. The mode is left extreme and the mean and median are close to each other.

**Cumulative distribution plot:**

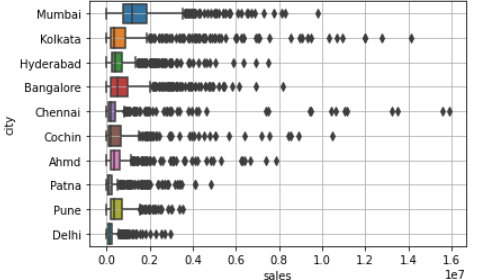


**Violin plot:**

The violin plot shows a vertical mirror image of the distribution along with the original distribution.



**Box plot for sales across different years:**



## Multivariate and Bivariate Analysis

Through bivariate analysis we try to analyse two variables simultaneously. As opposed to univariate analysis where we check the characteristics of a single variable, in bivariate analysis we try to determine if there is any relationship between two variables.

There are essentially 3 major scenarios that we will come across when we perform bivariate analysis

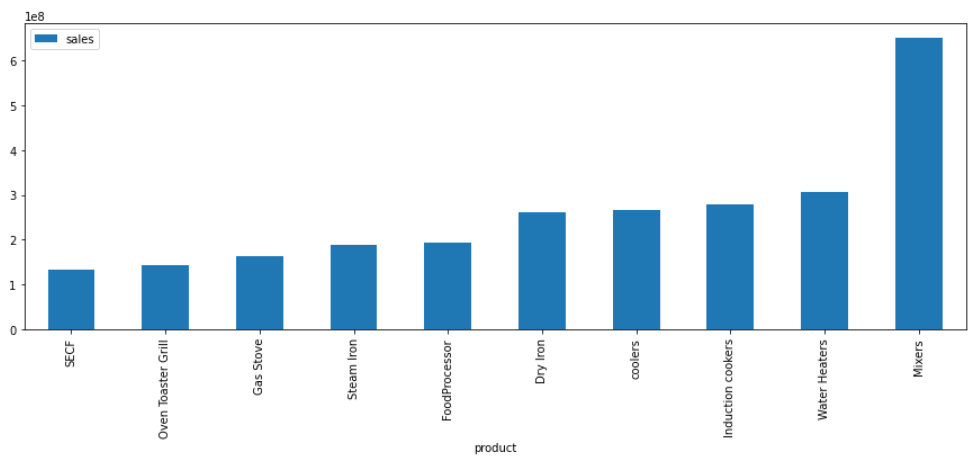
1. Both variables of interest are qualitative
2. One variable is qualitative and the other is quantitative
3. Both variables are quantitative

As there is just 1 continuous variable plotting correlation map and the heat map is not possible. For any of the multivariate analysis it requires 2 or more continuous variables

**Numerical vs. Categorical data**:

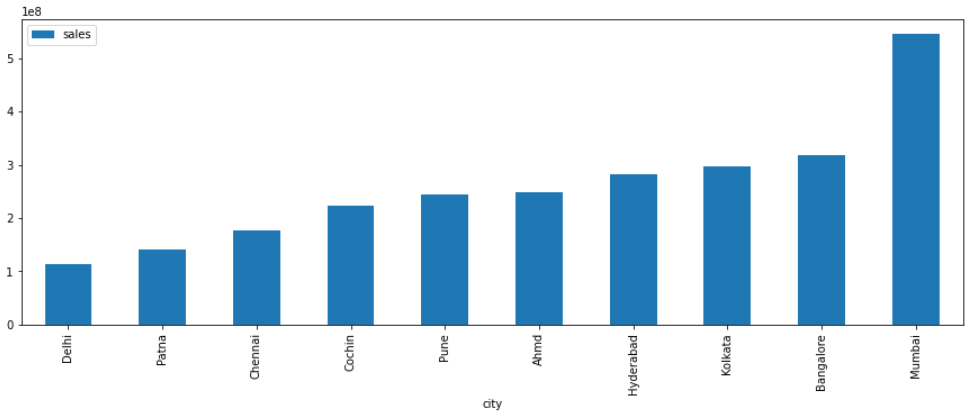
**Bar plot**:

**Product vs. sales**:

****

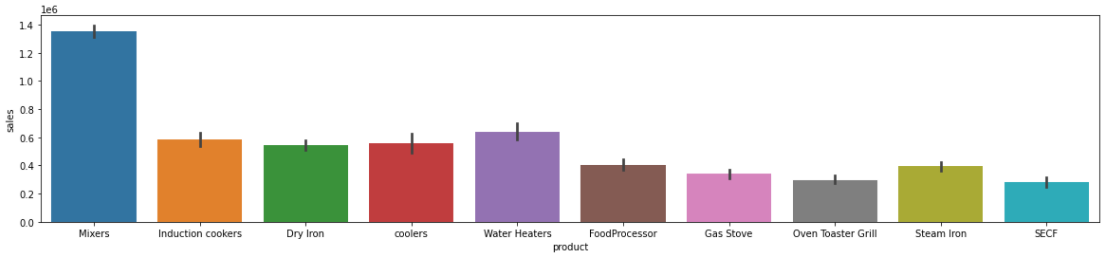
The mixers have higher sales and the SECF are having less sales compared to other products.

**City vs. sales:**

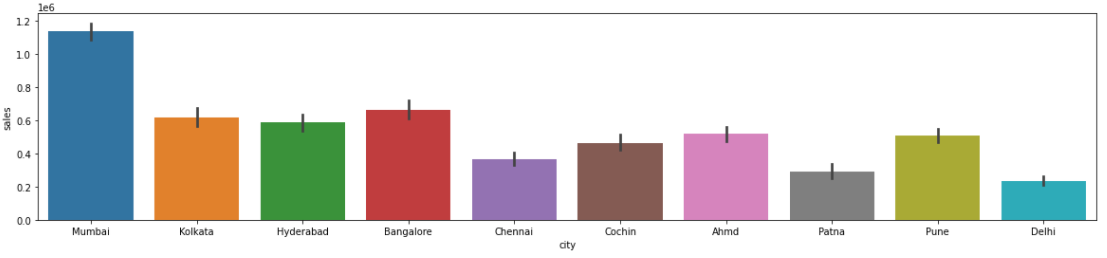
****

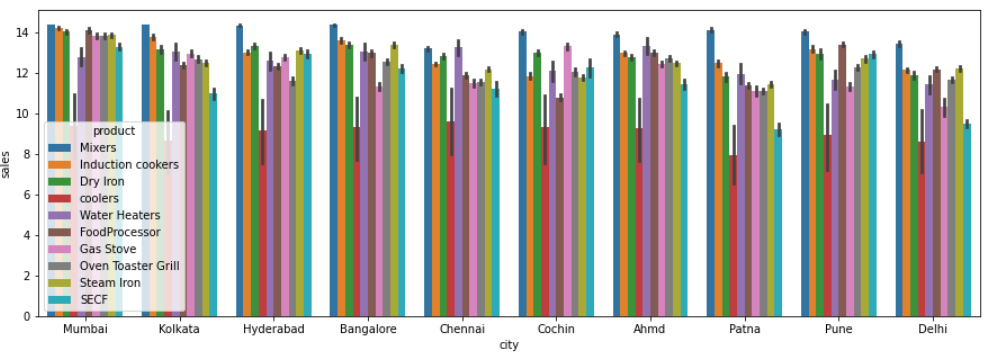
There are 10 cities in that Mumbai has the higher sales compared to the other cities. Delhi has the lowest sales in the products.

**Bar plot for product and sales:**

This is done without the sorting of index.

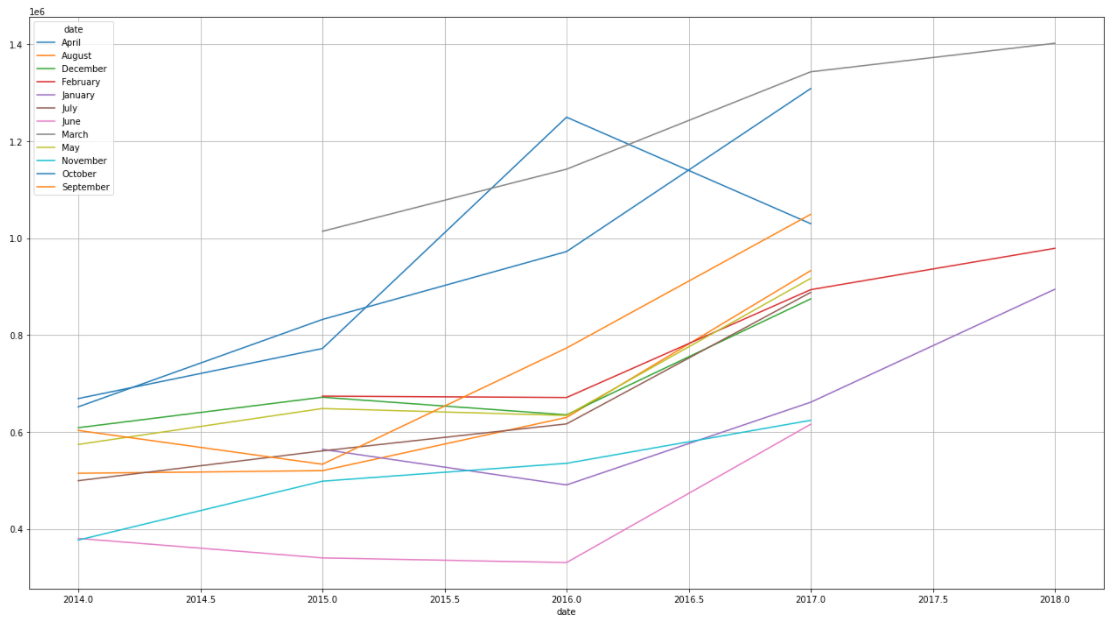
**Bar plot for city and sales:**



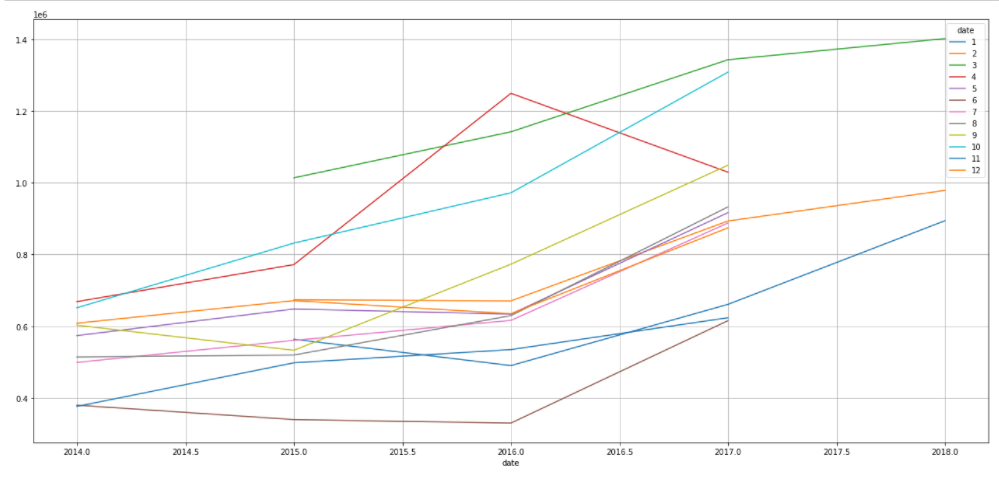


Looking at the product, city and the sales the Kolkata has the highest mixers sales but when it comes to sum of the sales Mumbai has the higher sales among the other cities. Chennai has the higher sales for the coolers. Water heater sales are higher in Chennai and Ahmedabad. The food processor, Oven Toaster Grill, Steam iron, SECF, gas stove and the dry iron sales are higher in Mumbai.

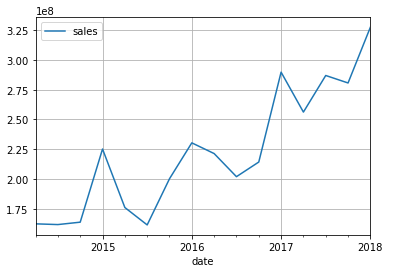
**Monthly Plot:**

****

**Yearly Plot:**

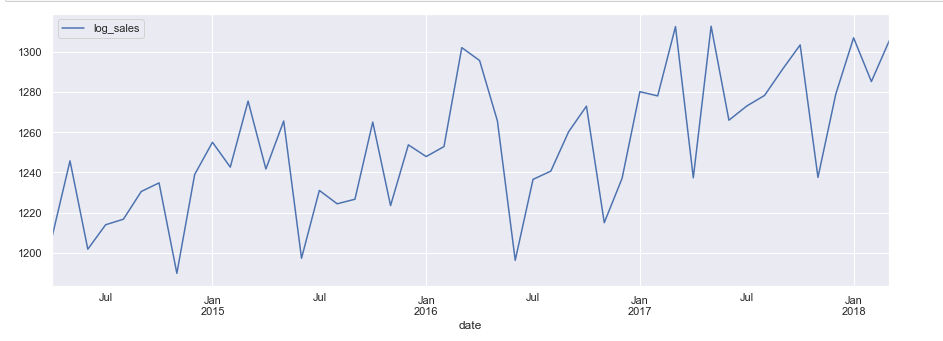
****

**Quarterly Plot:**

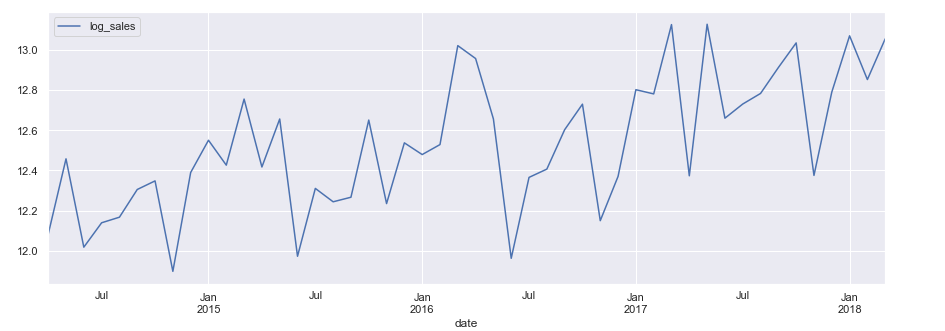
****

**Checking pattern using Trend plot (2014-2018) sales Rate Vs Years**

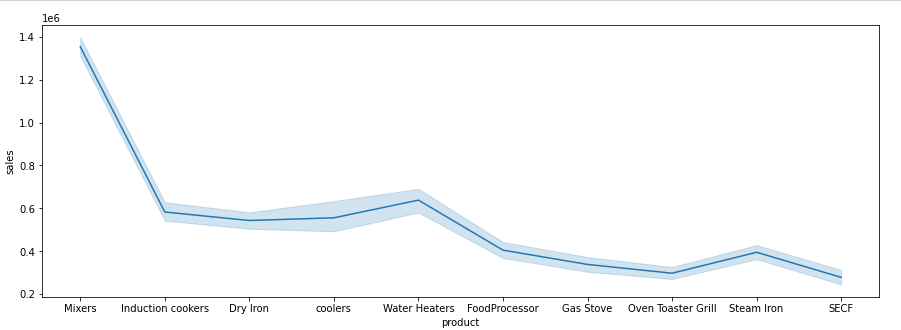
The groupby function is used to get the yearly trend plot for the distribution of sum of sales.



**Distribution of mean of sales:**

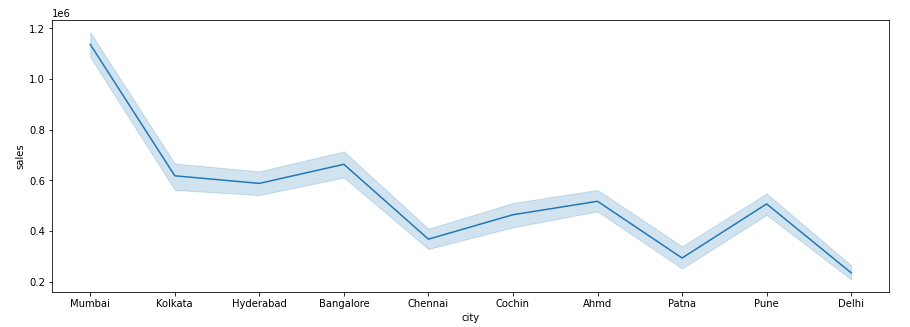


**Line Plot for sales vs product:**

****

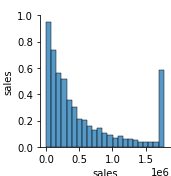
From the line plot it is understood that the highest sales is for the product Mixers and the lowest sales is for SECF.

**Line Plot for sales vs city:**

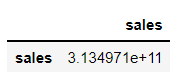
****

From the line plot it is understood that Mumbai has the highest sales and Delhi has the lowest sales.

**Pair plot:**

Since there is only one continuous variable the pair plot looks like the below shown image: 

**Covariance:**

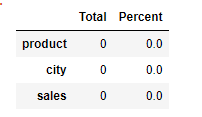


**Correlation:**



# ****Data Cleaning and Pre-processing****

## Identifying Missing values



From the above table, it is evident that there are no missing values in the dataset.

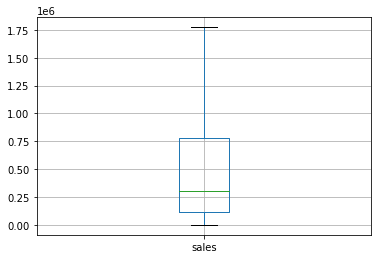
## Identifying outliers:

The box plot is created for the continuous variables to check if there are any outliers present in the dataset. The outliers are the extreme values.

There are more number of outliers in the sales column of the dataset. ****

Therefore the treatment of outliers is required. The removal of outliers is done by computing the lower range and upper range, IQR.

**After the removal of outliers:**

****

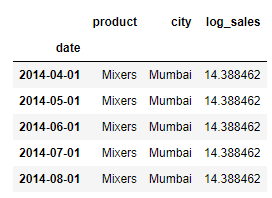
## Variable transformation

The sales column which has huge numbers and it is hard to perform operation on huge values. So, we have converted the numerical values into logs.Formula applied is log(x+1) since we had ‘0’ in the sales column log(0) would be negative infinity. To avoid negative infinity we have done log(x+1).

## Variables Added or Removed

The log\_sales is added to the dataset instead of the sales column and this is used for the further analysis.

So now the head of dataset is shown as follows:

****

# ****Model building****

The model building process involves setting up ways of collecting data, understanding and paying attention to what is important in the data to answer the questions you are asking, finding a statistical, mathematical or a simulation model to gain understanding and make predictions. Model building needs to be done after EDA to understand the behaviour of sales in the given dataset.

## Choosing the Right Model

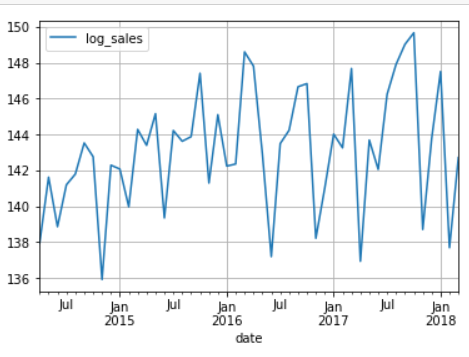
* Performed Time Series Forecasting on the given dataset to understand the trend and seasonality of the sales over years.
* Built models on top 3 products and cities since it contributes 80% of the revenue for the company.
* Grouped dataset by categorical variables such as Product and city
* Performed splitting of dataset into train and test (70:30)
* Built various model(Linear Regression, ARIMA, SARIMA, Triple Exponential Smoothing).
* ARIMA,SARIMA and Triple Exponential models are built to understand the trend and seasonality.

Below are the best models built on Product and City.

### Product Based Model

MIXERS:

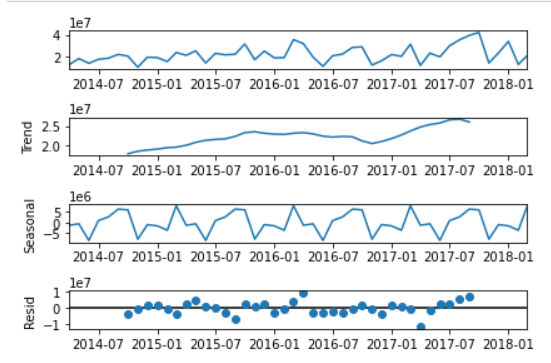
The time series are plotted for the Mixers data to understand the sales behaviour.



There is a seasonal pattern of sales.

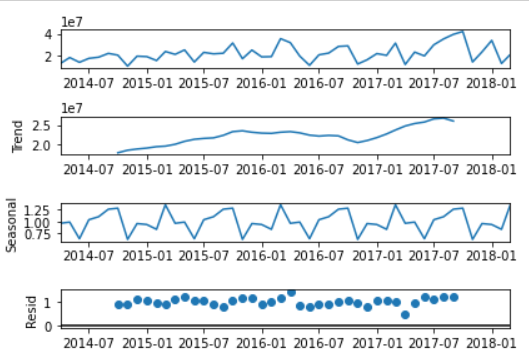
Decomposition of the time series is done when we decompose a time series into components, we usually combine the trend and cycle into a single trend-cycle component (sometimes called the trend for simplicity).Often this is done to help improve understanding of the time series, but it can also be used to improve forecast accuracy.

**Additive decomposition:**

****

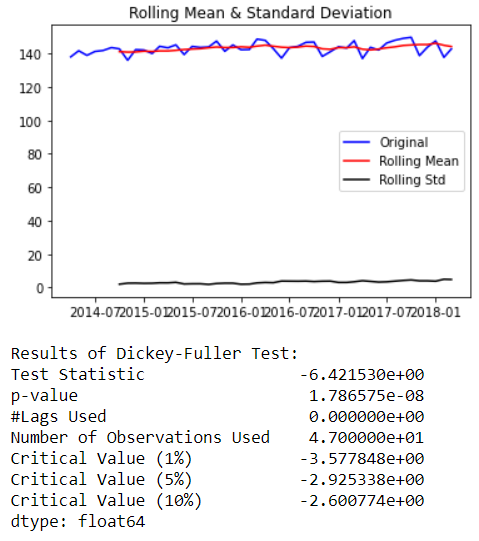
The residuals are located around 0 from the plot of residuals in the decomposition.

**Multiplicative decomposition:**

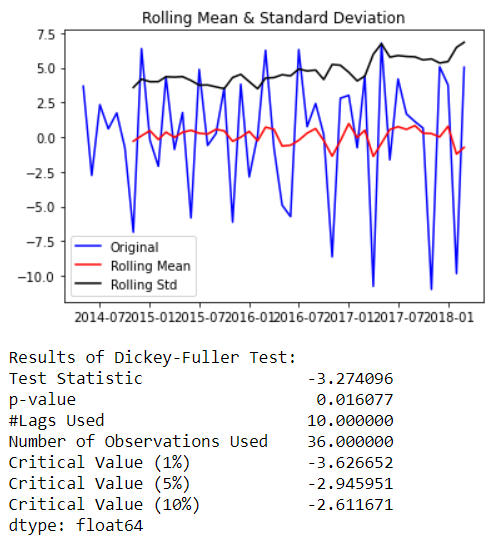
****

For the multiplicative model lot of residuals are located around 1.

Checking the stationarity for the mixers dataset:

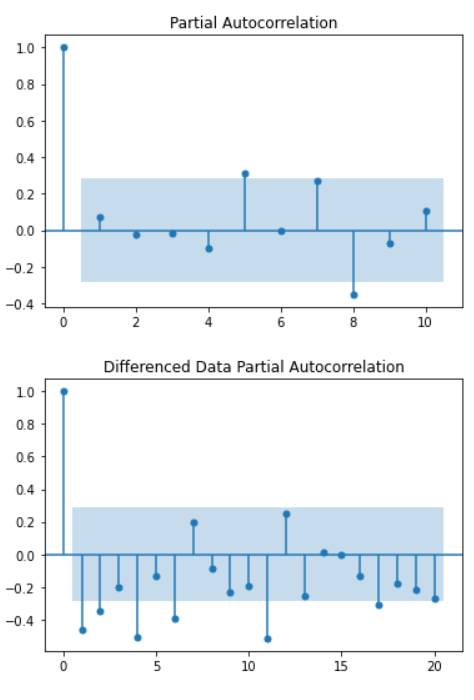


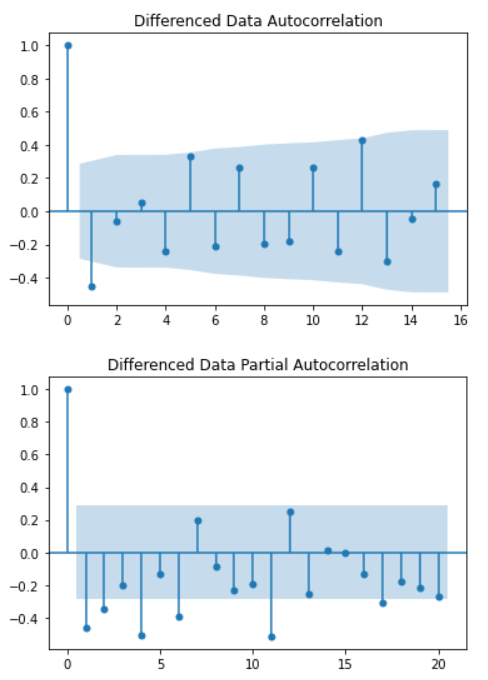
Let us take a difference of order 1 and check whether the Time Series is stationary or not.



Now the series are stationary the rolling standard goes above the mean and the p value is less than 0.5.

Plot the Auto correlation and Partial Auto correlation function plots:

**

**

From the above plots we can say that there is some seasonality.

**Splitting of Training and testing data:**

April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set.

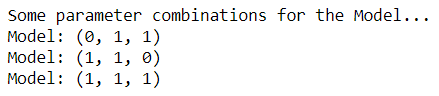
Now the model buildings are done based on the train and test set.

**Auto ARIMA**

An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):

The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 2

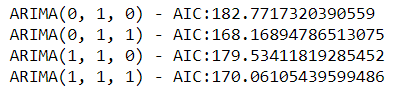
We have kept the value of d as 1 as we need to take a difference of the series to make it stationary.



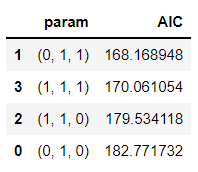
An empty Data frame is created with column names only



The AIC values for the different params are generated

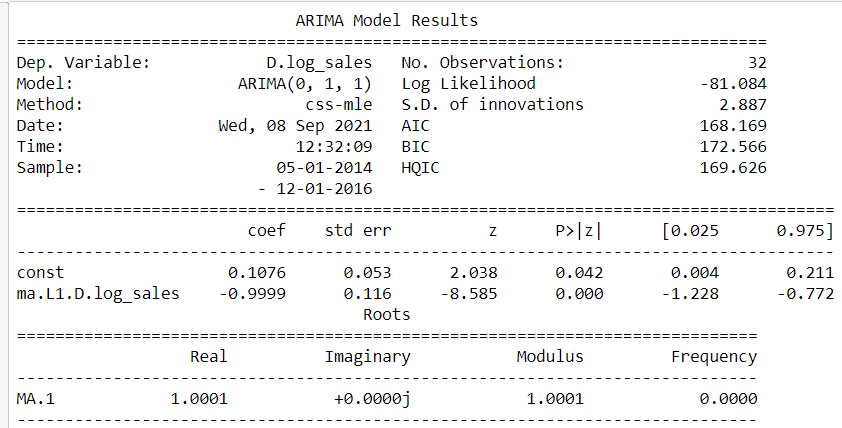


The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



The lowest AIC value is for 0,1,1 param which is further used to get the result summary.

The results for the auto ARIMA for the mixers:



Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.

**Manual ARIMA**



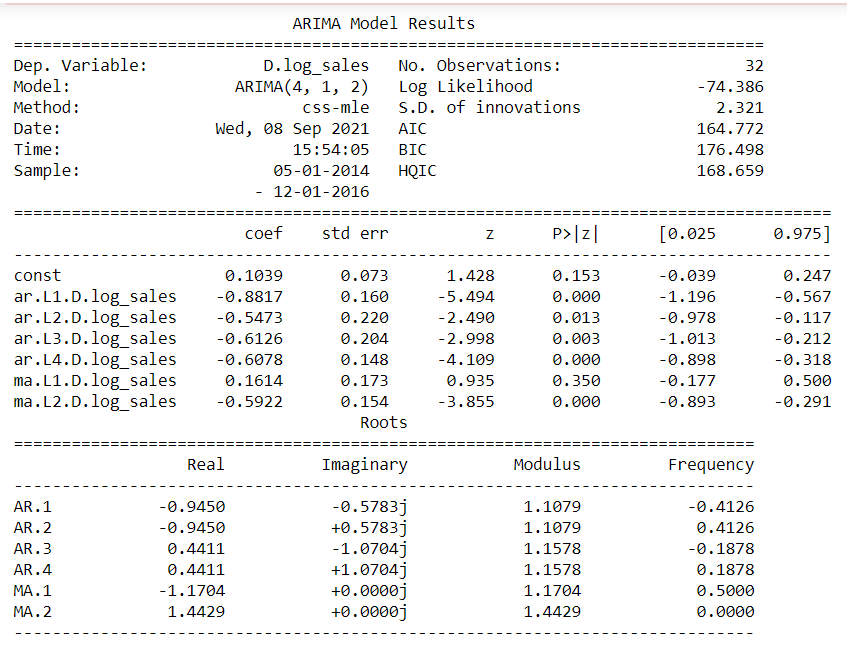
Here, we have taken alpha=0.05.

P= 2 and q= 1

The Auto-Regressive parameter in an ARIMA model is 'p' which comes from the significant lag before which the PACF plot cuts-off to 2. The Moving-Average parameter in an ARIMA model is 'q' which comes from the significant lag before the ACF plot cuts-off to 1.

The result of manual ARIMA:

The result is generated considering the p as 2 and q as 1 and differencing to 1.



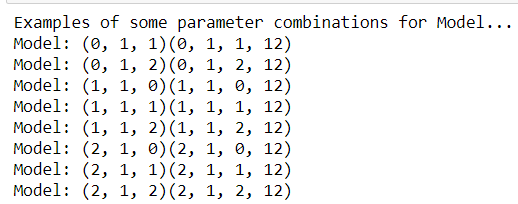
**SARIMA**

Build an Automated version of a SARIMA model for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC).

From the above differenced auto correlation plot we can get the seasonality value with which auto SARIMA model is built.

We see that there can be a seasonality of 1 as well as 12. We will run our auto SARIMA models by setting seasonality both as 1 and 12.

These are the param combinations which we get when the seasonality is set to 12.

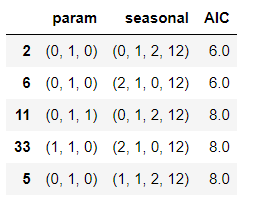
****

An empty dataset is generated to run through the loop.

****

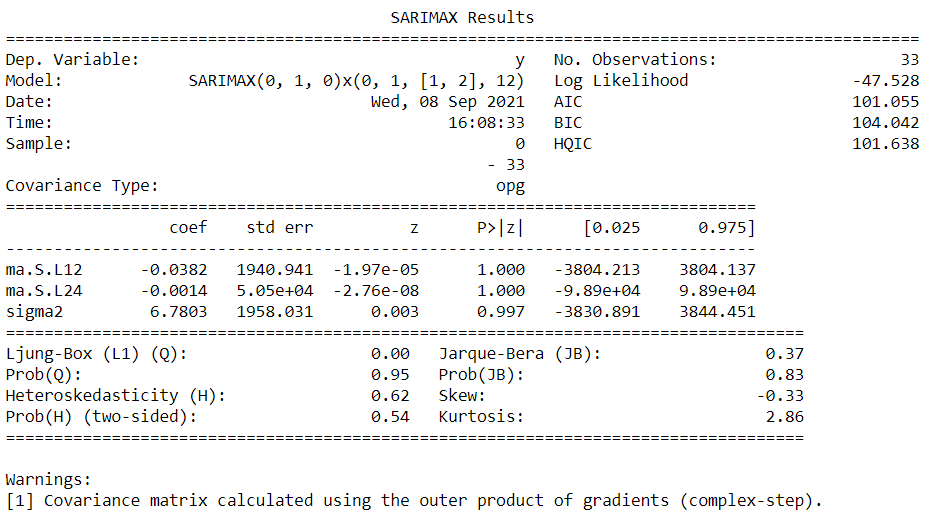
****

The AIC values for the param combination are listed above. Now these are sorted ascending order.

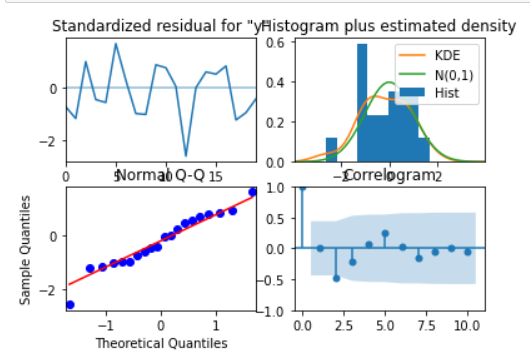
****

The best param will be (0,1,0)(0,1,2,12) with the lowest AIC value.

With this the auto SARIMA summary is generated.

****

Now the results of auto SARIMA is plotted:

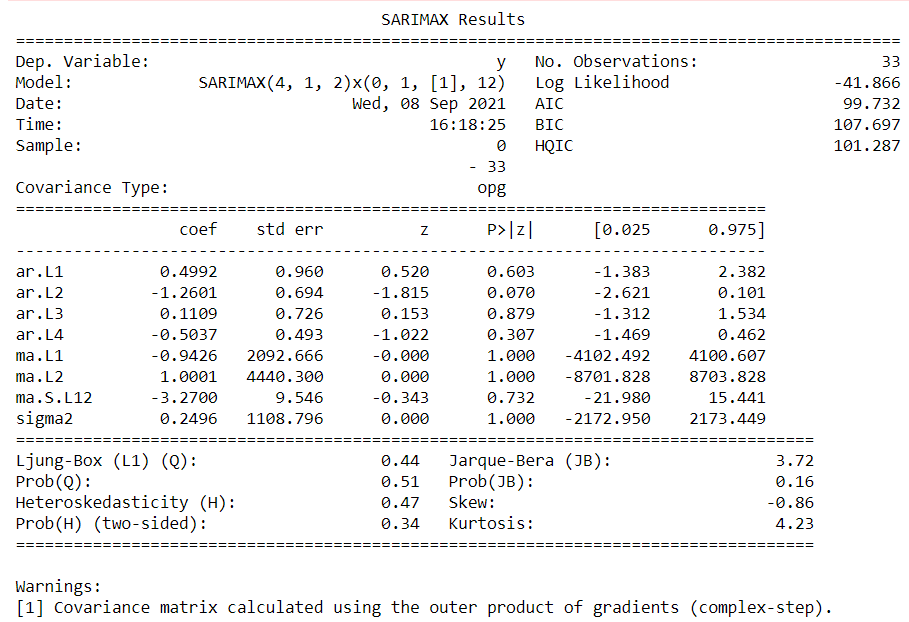
****

Setting the seasonality as 12 for the second iteration of the auto SARIMA model.

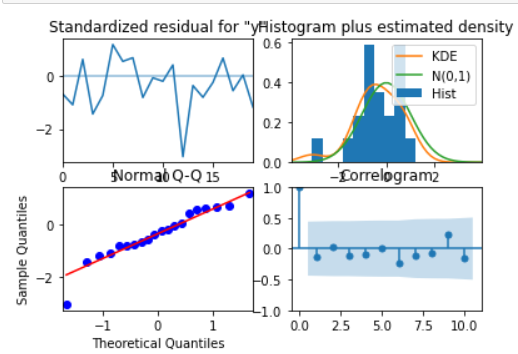
The param combination when the seasonality is set to 12.

The best param combination is (4,1,2)(0,1,1,12)

The result is generated for the best combination.

****

The result of auto SARIMA model is plotted:



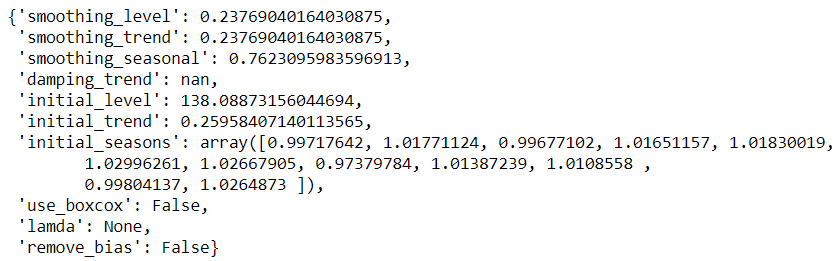
**Triple exponential smoothing**

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

The train and the test set is copied for the TES train and TES test set.

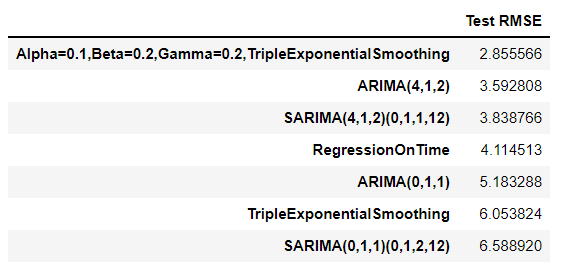
The TES model is created with trend as and seasonal as multiplicative.

The TES model is fit to params :



The alpha = 0.23

Beta= 0.23, gamma= 0.76

Using the above values the prediction is done.

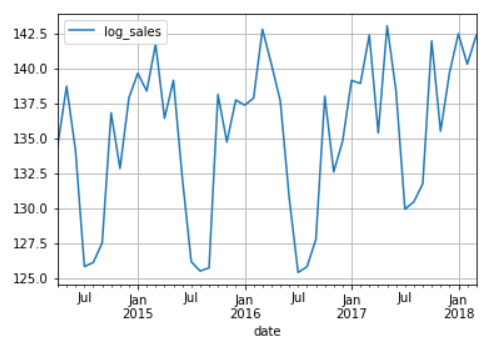
**NOTE:**

Please refer to the appendix for more models built on products..

### City Based Model

The grouping of the cities is done and its done again with regard to the date.

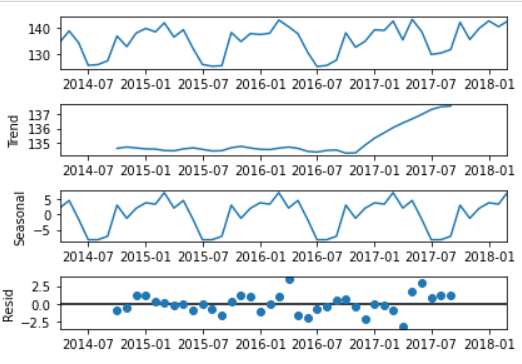
**MUMBAI:**

The time series are plotted for the Mumbai data to understand the sales behaviour. 

There is a seasonal pattern of sales and an upward trend so there is no stationarity.

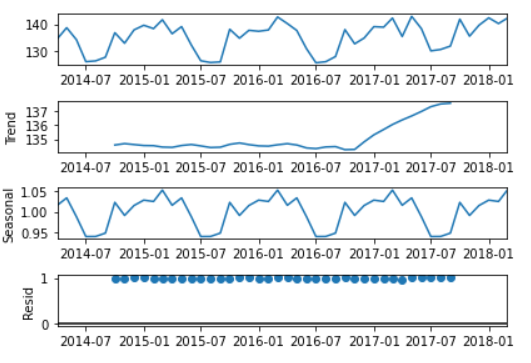
**Decomposition of the time series:**

**Additive decomposition:**

****

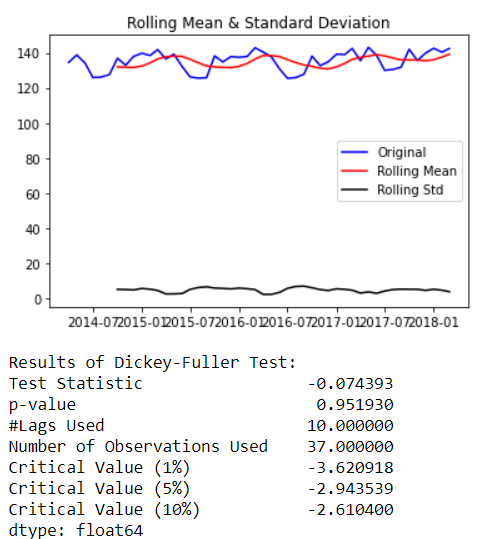
From the above it is clear that there is an upward trend and there is seasonal pattern and the residuals are located mostly at 0.

**Multiplicative decomposition:**

****

From the above it is clear that there is an upward trend and there is seasonal pattern and the residuals are located mostly at 1.

**Checking the stationarity for the Mumbai dataset:**

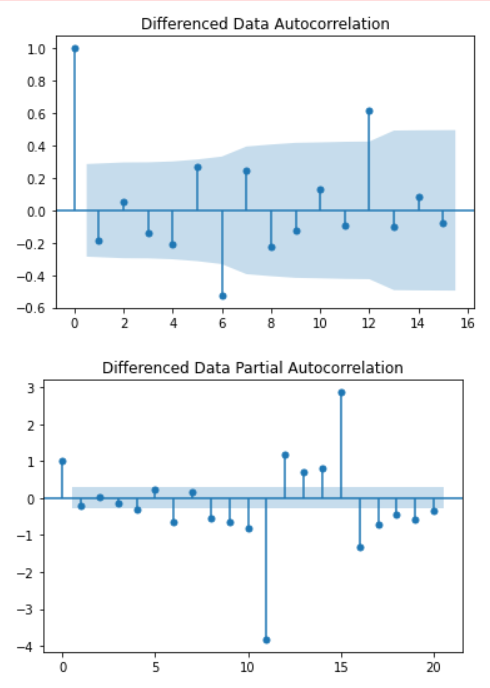
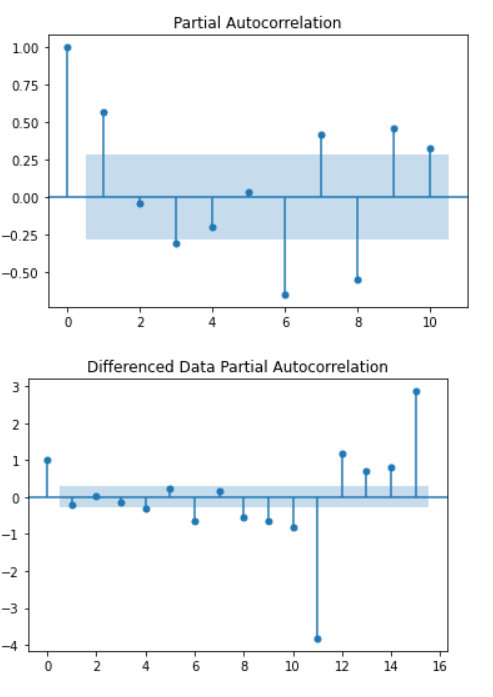
****

The series are not stationary since the pvalue is greater than 0.05.

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



Now the series are stationary the rolling standard goes above the mean and the p value is less than 0.5.

Plot the Auto correlation and Partial Auto correlation function plots *for the Mumbai dataset*:**

From the above plots we can say that there is some seasonality.

Splitting of Training and testing data:

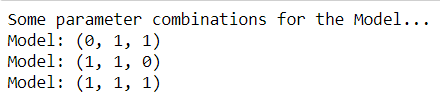
April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set and they are named as train1a and test1a.

Now the model building are done based on the train and test set.

An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):

**Auto ARIMA**

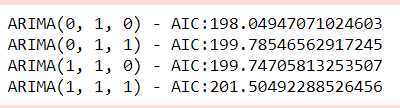
The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 2

**

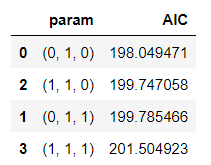
An empty Data frame is created with column names only

**

The AIC values for the different params are generated

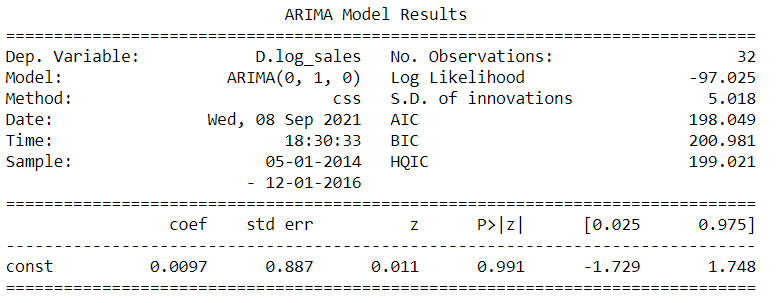
**

The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



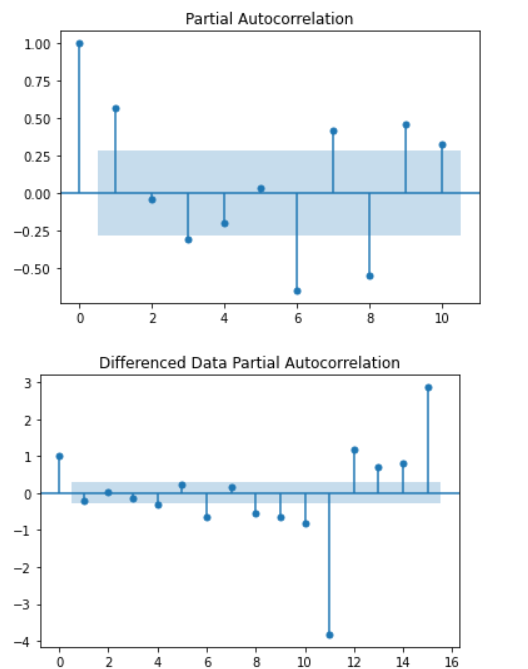
The lowest AIC value is for 0,1,0 param which is further used to get the result summary.

The results for the auto ARIMA for Mumbai:



Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.

**Manual ARIMA**

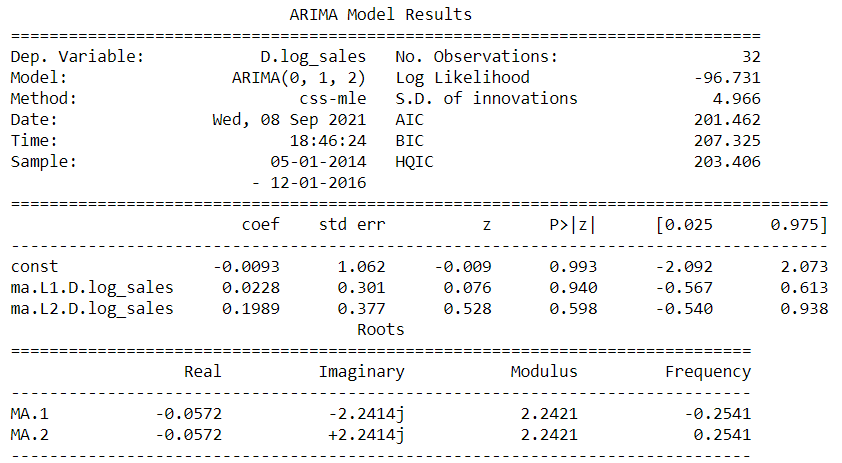


Here, we have taken alpha=0.05.P= 0 and q= 0

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.

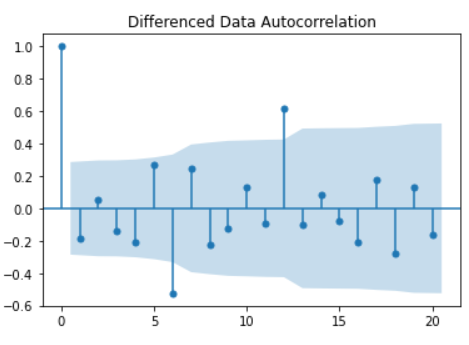
The result of manual ARIMA :

The result is generated considering the p as 0 and q as 1 and differencing to 2.



Build an Automated version of a SARIMA model for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC).

**SARIMA**



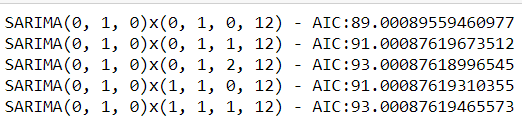
We see that there can be a seasonality of 6 as well as 12. We will run our auto SARIMA models by setting seasonality both as 6 and 12.

These are the param combinations which we get when the seasonality is set to 12..

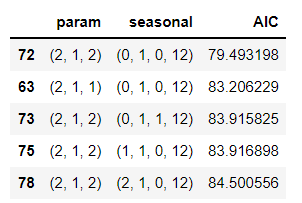


An empty dataset is generated to run through the loop.



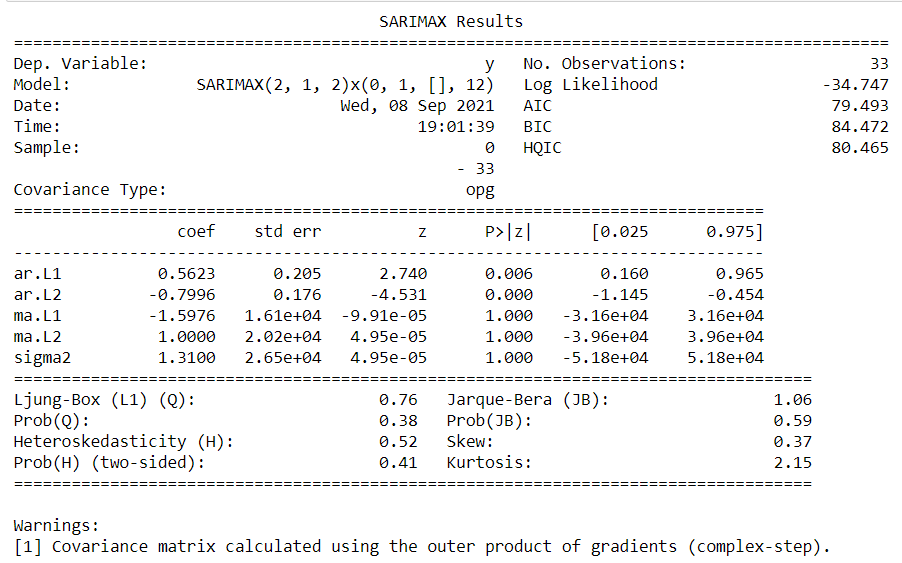


The AIC values for the param combination are listed above. Now these are sorted ascending order.



The best param will be (**2**,1,2)(**2**,0,2,12) with the lowest AIC value.

With this the auto SARIMA summary is generated.

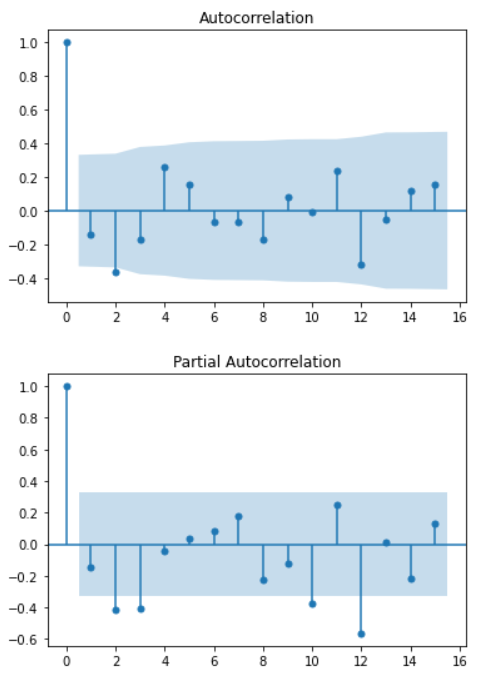
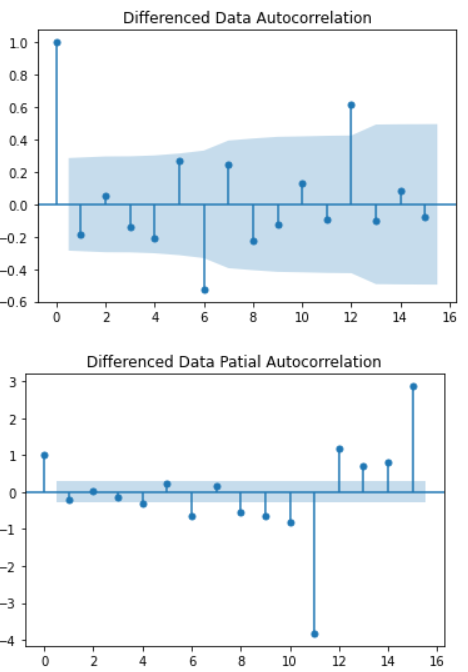


Now the results of auto SARIMA is plotted:



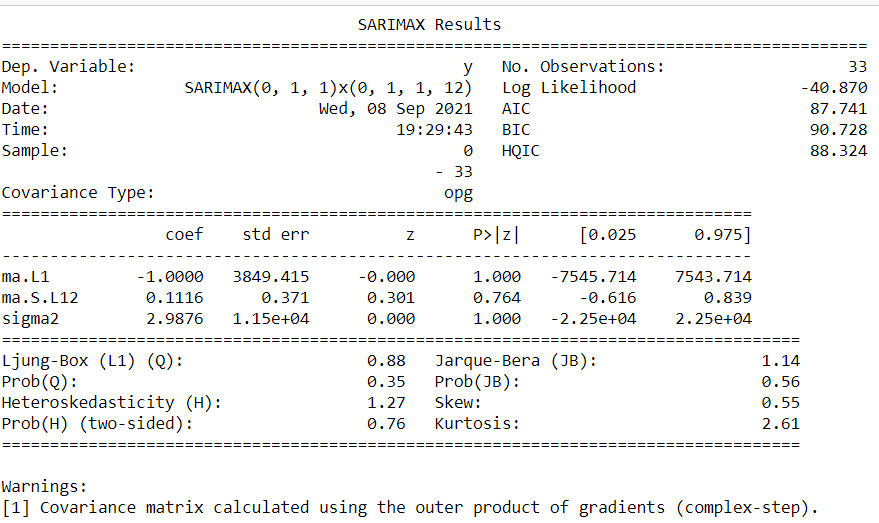
Setting the seasonality as 12 for the second iteration of the auto SARIMA model.

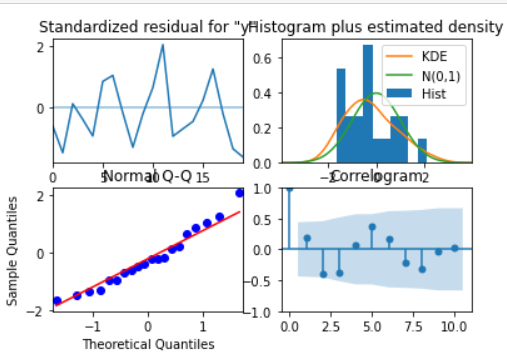
The param combination when the seasonality is set to 12.



The best param combination is (0,1,1)(0,1,1,12)

The result is generated for the best combination.





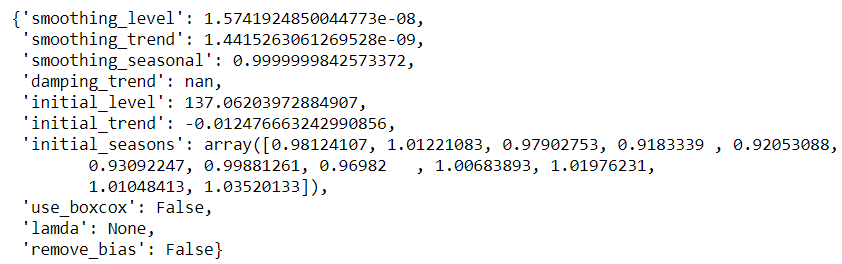
**Triple exponential smoothing**

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

The train and the test set is copied for the TES train and TES test set which is named as TES\_train\_M and TES\_test\_M.

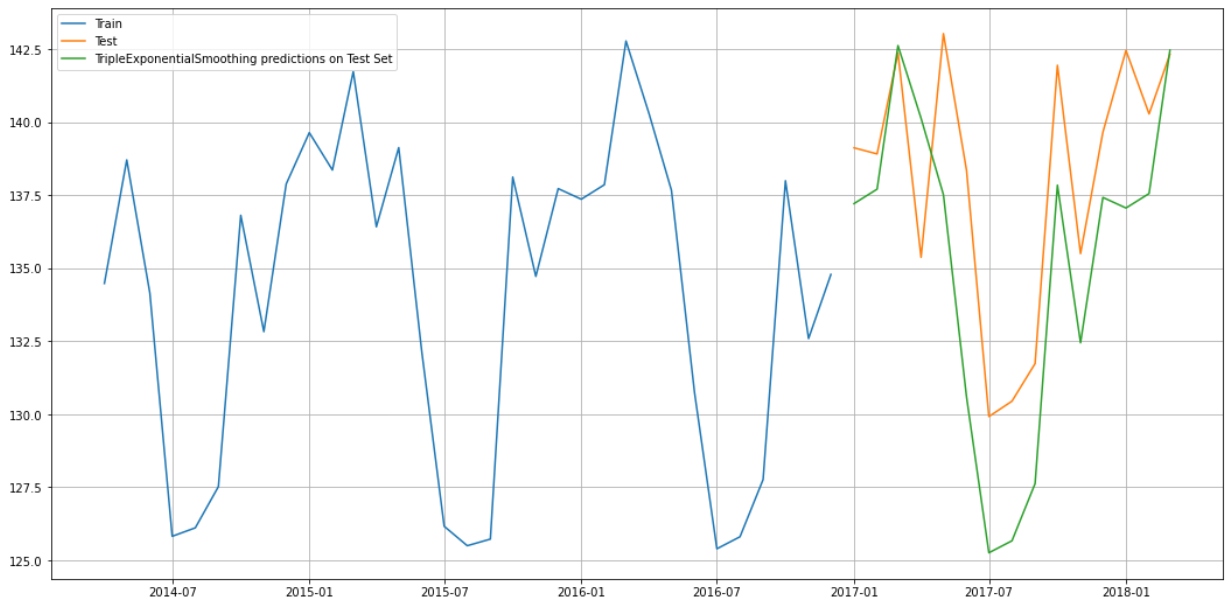
The TES model is created with trend as and seasonal as multiplicative.

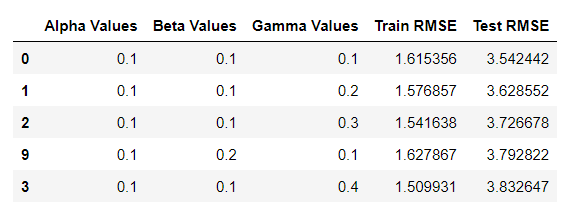
The TES model is fit to params :

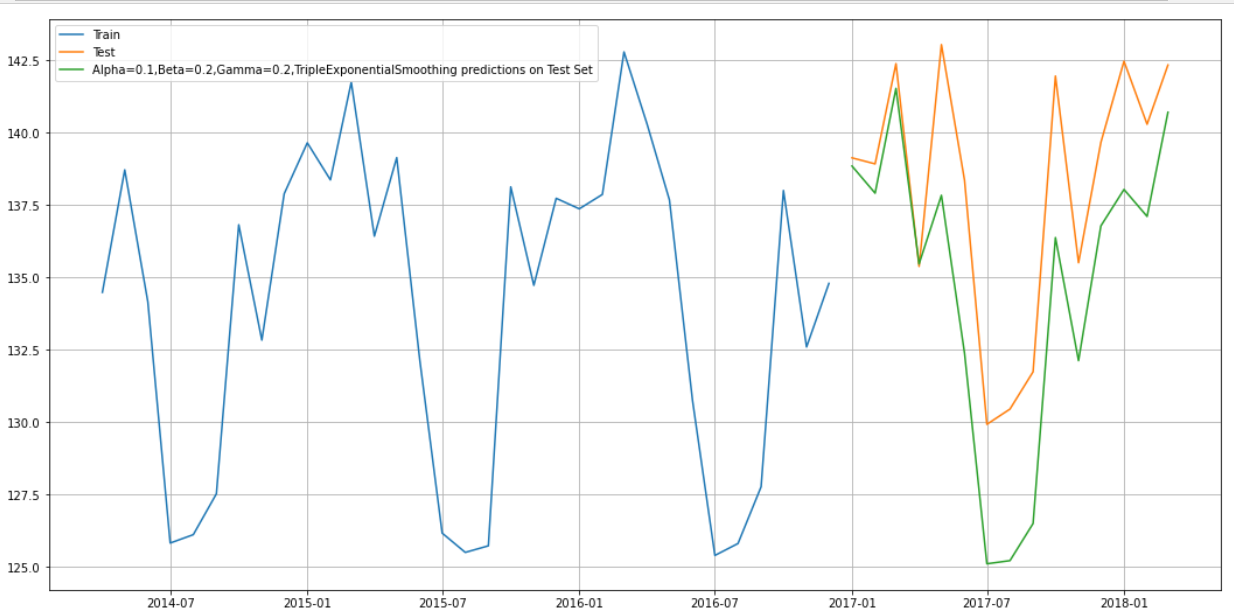


The alpha = 1.57,Beta= 1.44,gamma= 0.99

Using the above values the prediction is done







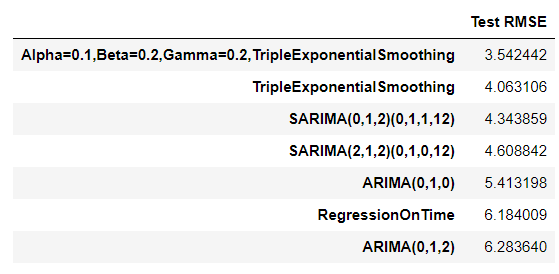
**NOTE:**

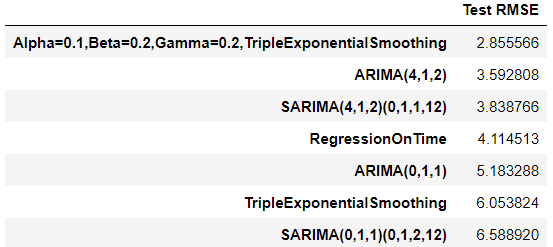
For further models on products and cities Please refer to appendix.

## Efforts to improve model performance

* Model tuning has been performed to increase the RMSE score.
* Lesser the RMSE score greater the accuracy.
* For every model: Model tuning has been performed on SARIMA,ARIMA and Triple Exponential Smoothing.
* Future predictions on the models have also been done to predict sales.

Here, the below chart shows that the model tuning done for the city Mumbai and product Mixers and the best RMSE score is given by Triple Exponential Smoothing. So the optimum model among other models is performed well by TES.





# ****Model validation****

1. Model validated based on train test split and the RMSE scores has been calculated for finding the accuracy.
2. Triple exponential smoothing ,ARIMAX and SARIMAX models have been built and model tuning has been done for getting best RMSE score.
3. Decomposition is also done to improve the accuracy.

## Model validation on Products

**Mixers**

**Auto ARIMA**

The results of auto ARIMA summary is used to forecast with the length of the test set to get predicted auto ARIMA.

The mean squared error are imported from sklearn metrics library to get the root mean squared error score.

The RMSE for the predicted auto ARIMA is 5.18

Now this score is inserted to dataset form with respect to best param chosen combination.



**Manual ARIMA**

The results of manual ARIMA summary is used to forecast with the length of the test set to get predicted manual ARIMA.

The RMSE for the predicted manual ARIMA is 3.59

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous auto ARIMA RMSE score.

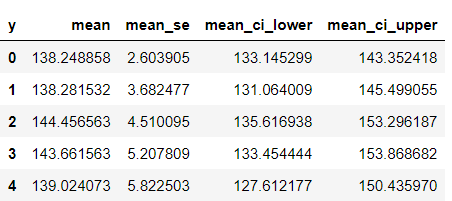


**Auto SARIMA**

Prediction on the Test Set using auto SARIMA model and evaluate the model.

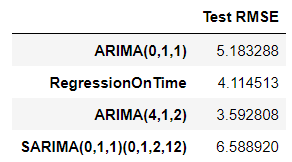
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA :



The RMSE for the predicted auto SARIMA is 6.58

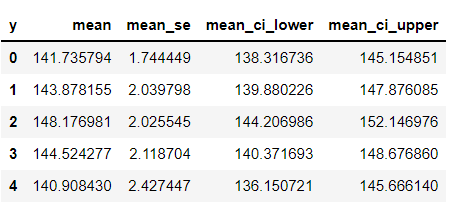
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



Prediction on the Test Set using auto SARIMA model for seasonality of 12 and evaluate the model.

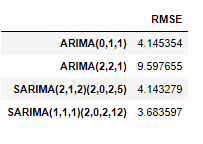
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA for seasonality as 12 :

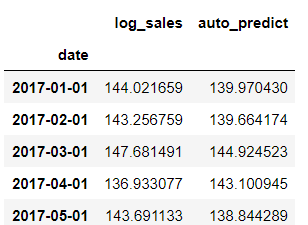


The RMSE for the predicted manual ARIMA is 3.83

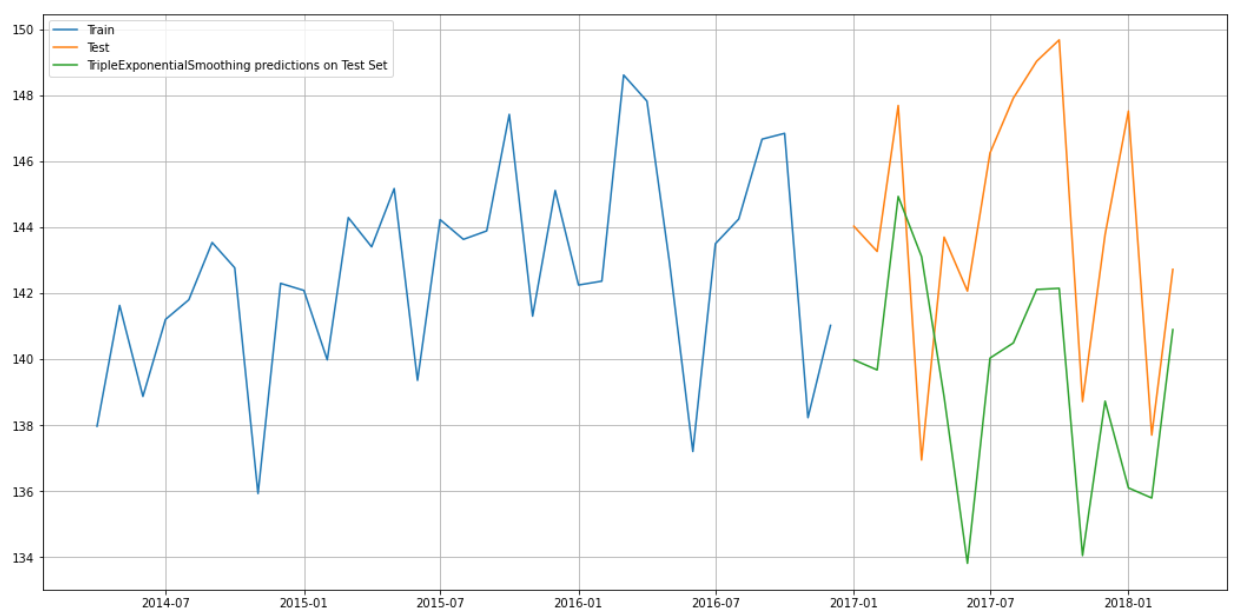
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



**Triple Exponential Smoothing**

Prediction on the test data is done to forecast to length of the test set:

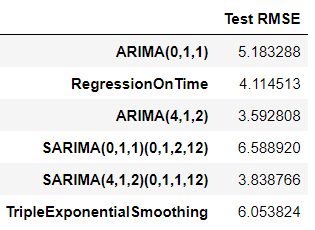
Plotting on both the Training and Test using autofit



The RMSE for the TES with the best alpha, beta, gamma values using the test set are:



Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score

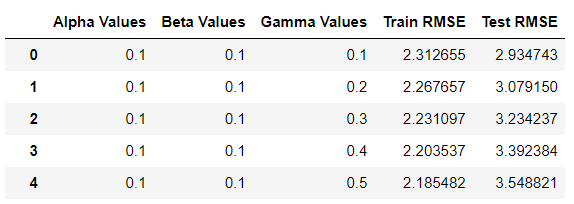


First we will define an empty data frame to store our values from the loop

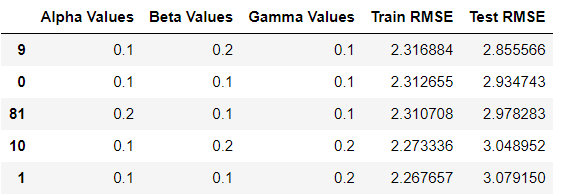


Now a for loop is generated with different alpha, beta, gamma values.

The result of different alpha , beta, gamma

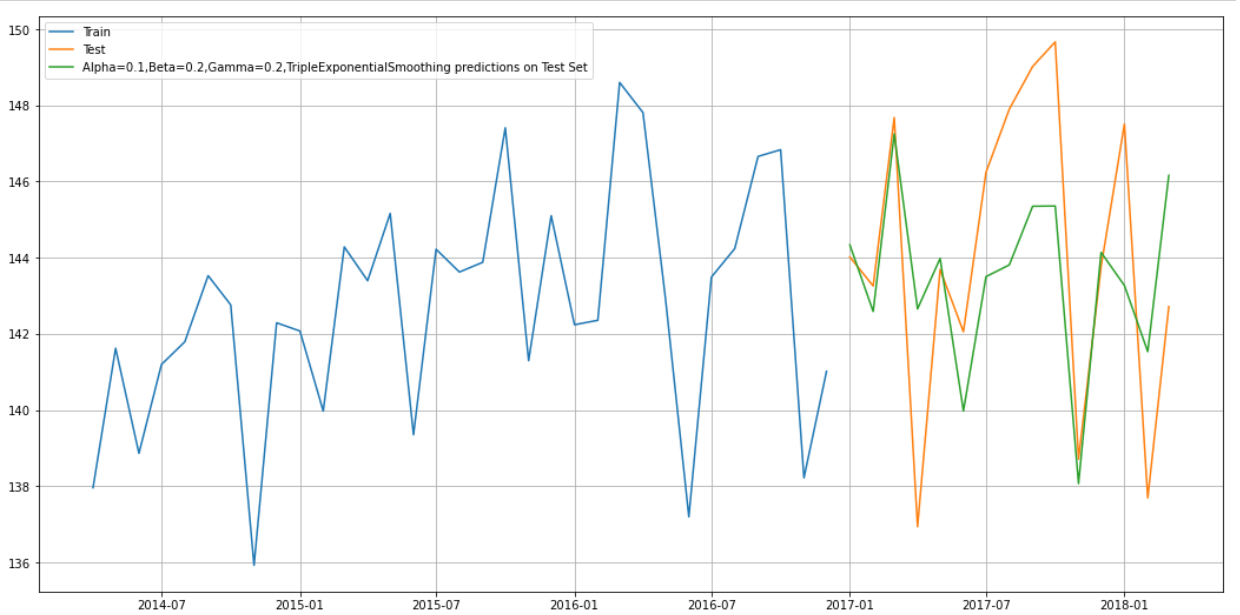


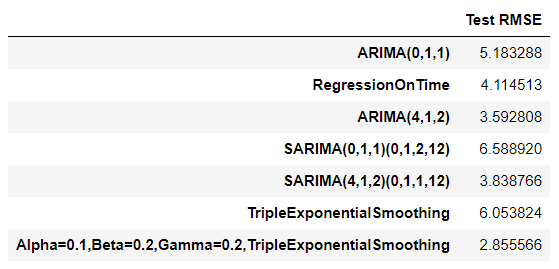
The same is sorted based on test RMSE,



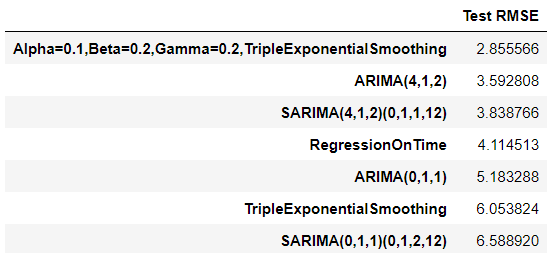
The best values are 0.1,0.2,0.1

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





Sorting of values based on the test RMSE.



## Model validation on City models

Mumbai

**Auto ARIMA**

The results of auto ARIMA summary is used to forecast with the length of the test set to get predicted auto ARIMA.

The mean squared error are imported from sklearn metrics library to get the root mean squared error score.

The RMSE for the predicted auto ARIMA is 5.41

Now this score is inserted to dataset form with respect to best param chosen combination.

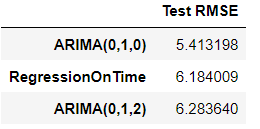


**Manual ARIMA**

The results of manual ARIMA summary is used to forecast with the length of the test set to get predicted manual ARIMA.

The RMSE for the predicted manual ARIMA is 6.28

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous auto ARIMA RMSE score.

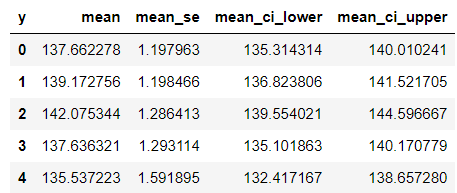


**Auto SARIMA**

Prediction on the Test Set using auto SARIMA model and evaluate the model.

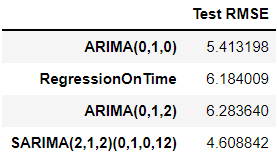
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA :



The RMSE for the predicted auto SARIMA is 4.60

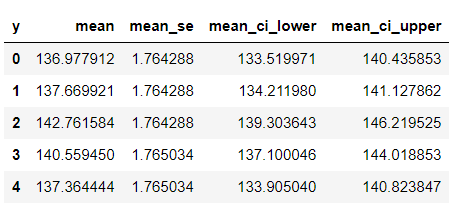
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



Prediction on the Test Set using auto SARIMA model for seasonality of 12 and evaluate the model.

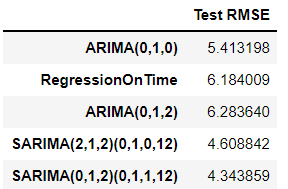
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA for seasonality as 12 :



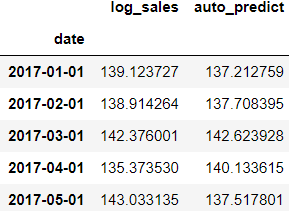
The RMSE for the predicted manual ARIMA is 4.34

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.

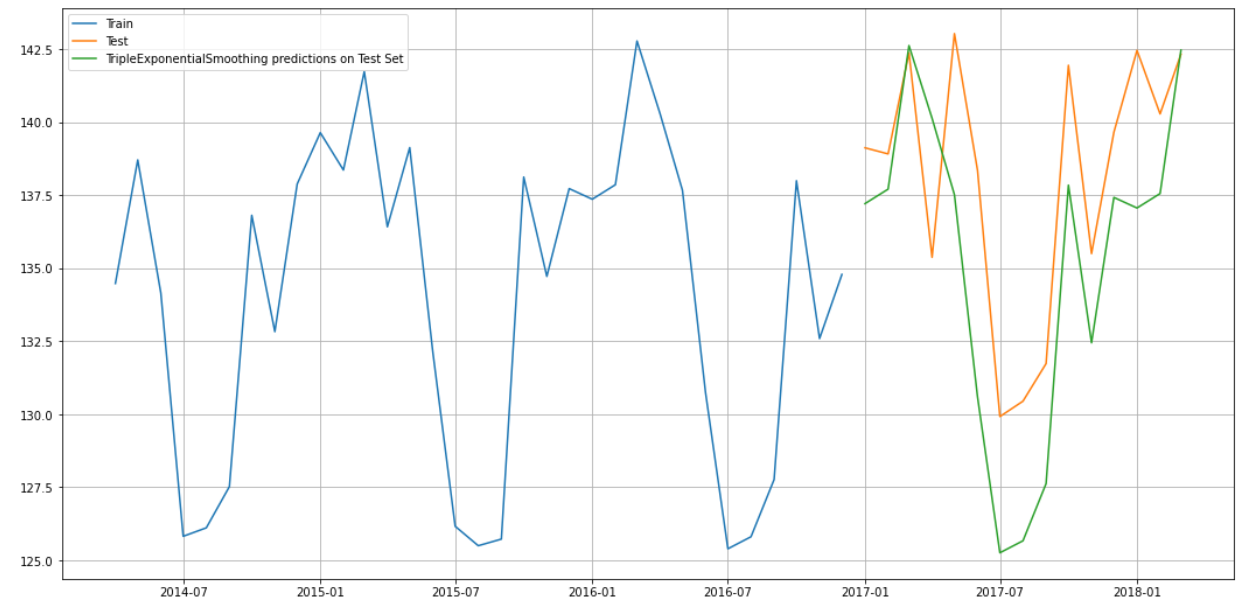


**Triple Exponential Smoothing**

Prediction on the test data is done to forecast to length of the test set:



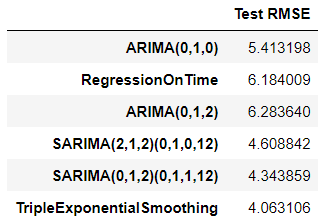
Plotting on both the Training and Test using autofit



The RMSE for the TES with the best alpha, beta, gamma values using the test set are:

Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 4.063

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score

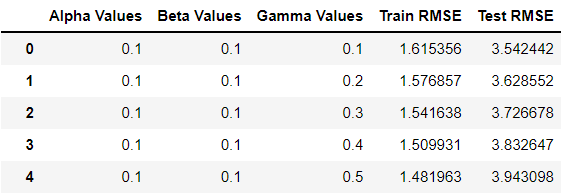


First we will define an empty data frame to store our values from the loop

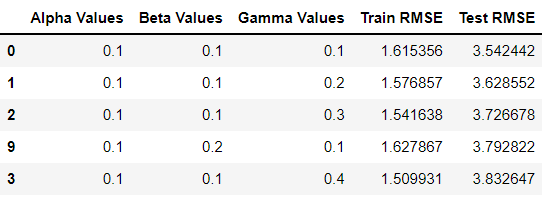


Now a for loop is generated with different alpha, beta, gamma values.

The result of different alpha , beta, gamma

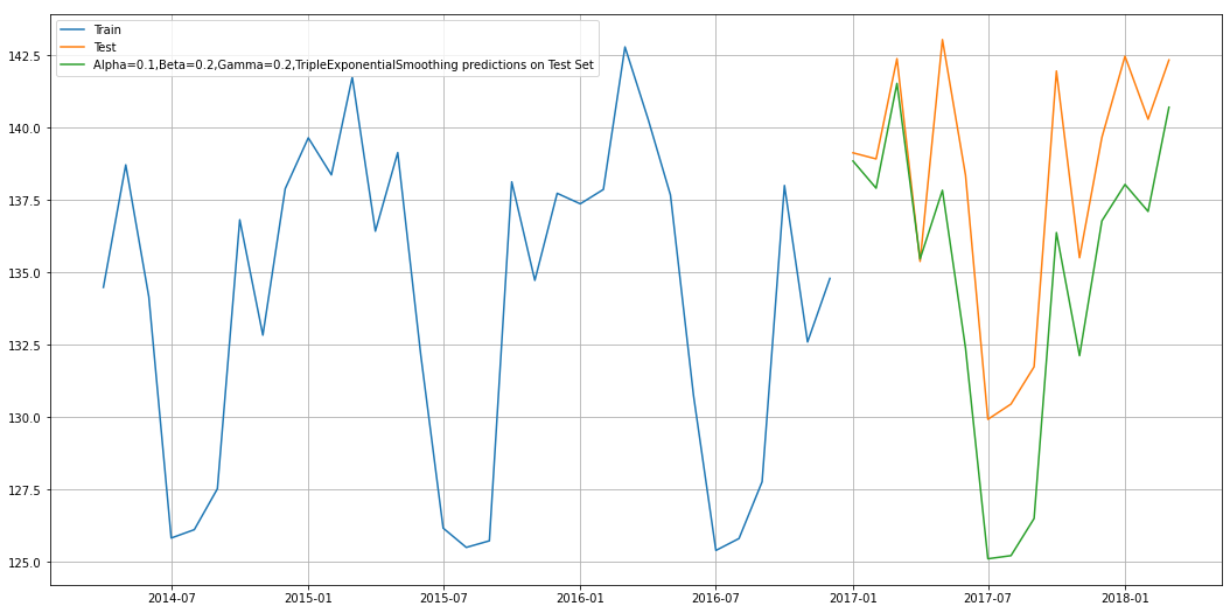


The same is sorted based on test RMSE,

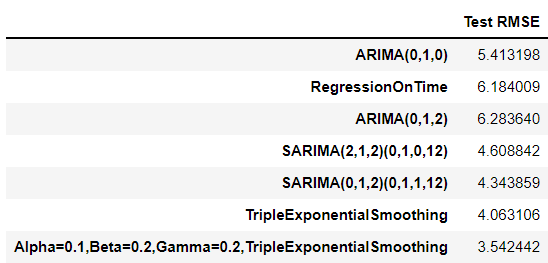


The best values are 0.1,0.1,0.1

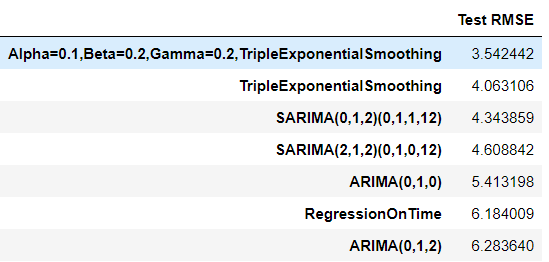
Plotting on both the Training and Test data using brute force alpha, beta and gamma determination



RMSE SCORES:



Sorting of values based on the test RMSE.



**NOTE:**

For further validation of models on products and cities Please refer to appendix.

# ****Final interpretation / recommendation****

With the past history of records it is understood that:

**Among the top 3 products:**

* The mixers has the highest sales when compared to the other products
* Mixers has highest sales in Mumbai during 2017
* Coolers does not have sales during the month of July, August, September for all years and all cities.
* Patna has least sale for coolers during 2014

**Among the top 3 cities:**

* Mumbai has the highest sales when compared to the other products
* Bangalore city performs the best with respect to RMSE score .It has the lowest RMSE score.
* The RMSE score is at the lowest when triple exponential model is used for all top 3 products and cities. Hence the triple exponential model is the optimum model compared to the other model

**Overall Model Performance:**

Different models on the Training Data is built and then the accuracy (or error) on the training and test data is checked. The model which performs the best on the test data is an optimum model for us. Here, the triple exponential model performs best on the test data. Various marketing strategies need to be adopted for the product which brings lower sales. These models can be used in Supply Chain Management.

Below table gives suggestion for the business:



# APPENDIX

* 1. MODEL BUILDING AND INTERPRETATION:

## Building various models

The grouping of sales are done for the products and cities separately.

Now various time series models are built for the top 3 products and top 3 cities separately.

The top 3 products are **Mixers**, **Coolers** and **Induction Cookers**.

### Product Based models

Induction Cookers:

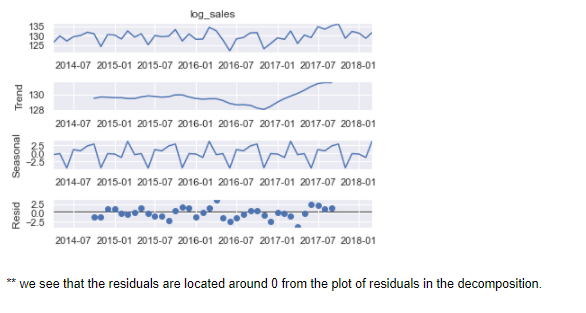
The time series are plotted for the Induction Cookers data to understand the sales behaviour.



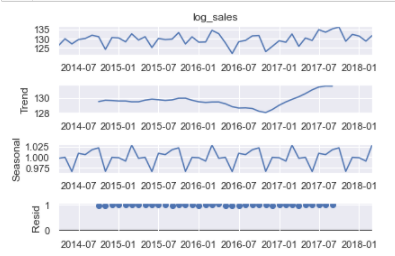
There is a seasonal pattern of sales.

Decomposition of the time series and plotting of the same.

Additive decomposition:

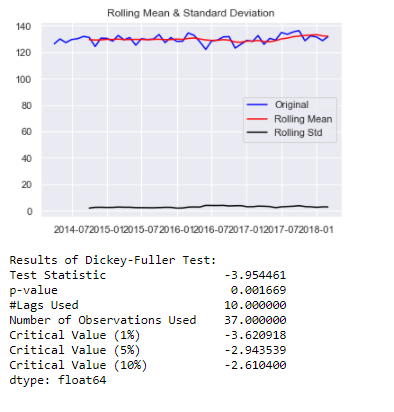


Multiplicative decomposition:

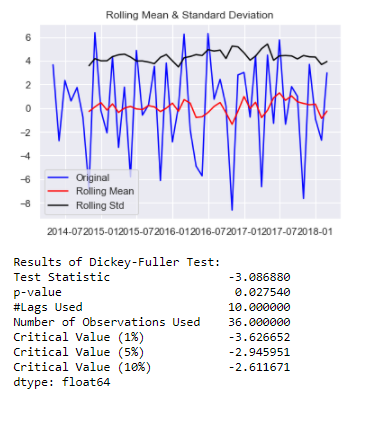


For the multiplicative model lot of residuals are located around 1.

Checking the stationarity for the Induction Cookers dataset:

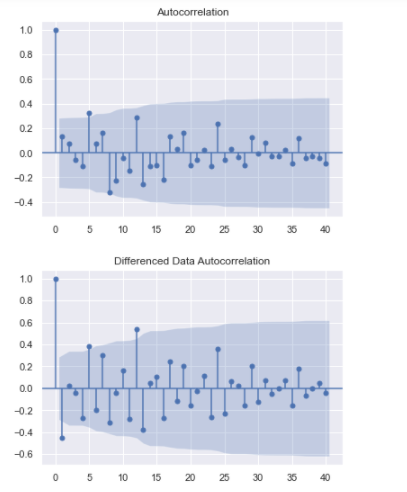


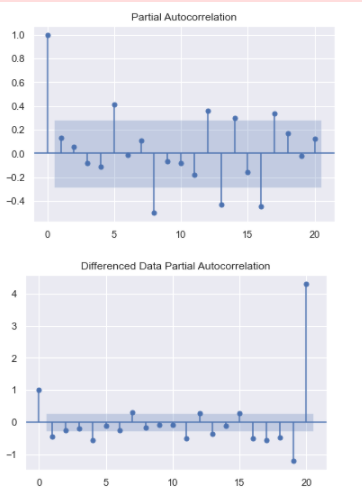
Let us take a difference of order 1 and check whether the Time Series is stationary or not.



Now the series are stationary the rolling std goes above the mean and the p value is less than 0.5.

**Plot the Auto correlation and Partial Auto correlation function plots:**





From the above plots we can say that there is some seasonality.

Splitting of Training and testing data:

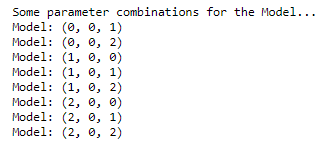
April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set and they are named as train1 and test1.

Now the model building are done based on the train and test set.

##### Auto ARIMA

**An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):**

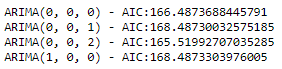
The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 3



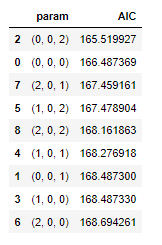
An empty Data frame is created with column names only



The AIC values for the different params are generated

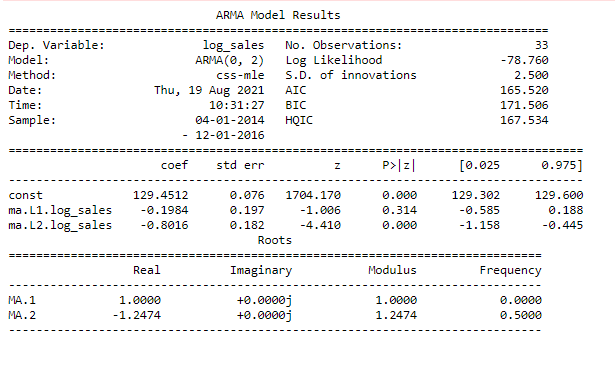


The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



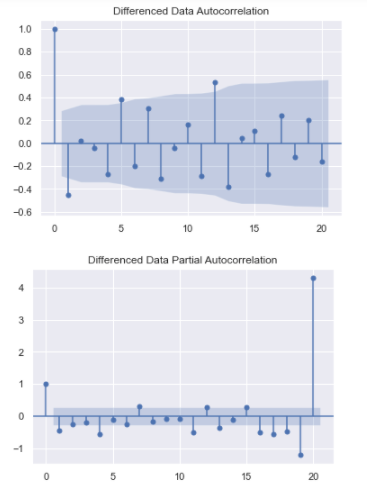
The lowest AIC value is for 0,0,2 param which is further used to get the result summary.

The results for the auto ARIMA for the Induction Cookers :



**Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.**

##### Manual ARIMA

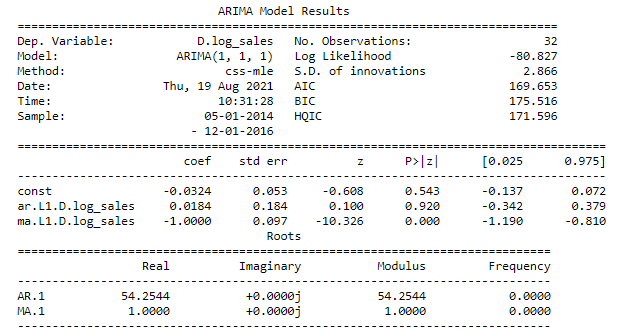
****

Here, we have taken alpha=0.05. P= 1 and q= 1

The Auto-Regressive parameter in an ARIMA model is 'p' which comes from the significant lag before which the PACF plot cuts-off to 1. The Moving-Average parameter in an ARIMA model is 'q' which comes from the significant lag before the ACF plot cuts-off to 1.

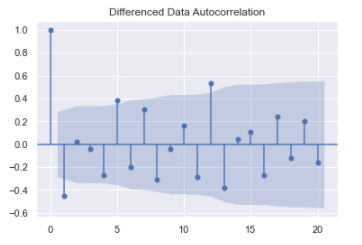
The result of manual ARIMA :

The result is generated considering the p as 1 and q as 1 and differencing to 1.



**Build an Automated version of a SARIMA model for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC).**

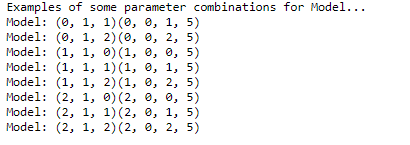
##### SARIMA



From the above differenced auto correlation plot we can get the seasonality value with which auto SARIMA model is built.

We see that there can be a seasonality of 5 as well as 12. We will run our auto SARIMA models by setting seasonality both as 5 and 12.

These are the param combinations which we get when the seasonality is set to 5.



An empty dataset is generated to run through the loop.



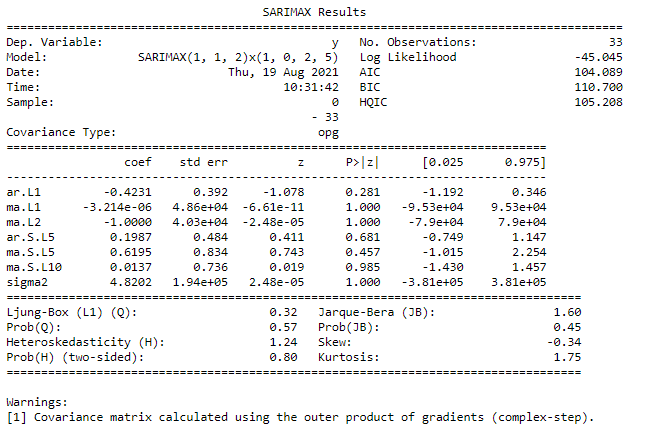


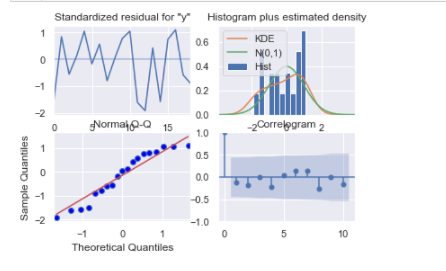
The AIC values for the param combination are listed above. Now these are sorted ascending order.



The best param will be (1,1,2)(1,0,2,5) with the lowest AIC value.

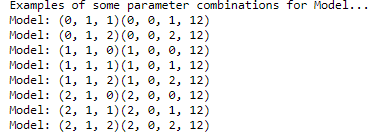
With this the auto SARIMA summary is generated.



Now the results of auto SARIMA is plotted:

Setting the seasonality as 12 for the second iteration of the auto SARIMA model.

The param combination when the seasonality is set to 12.



The empty data frame is defined:



The AIC values for the different param combination are listed as follows:

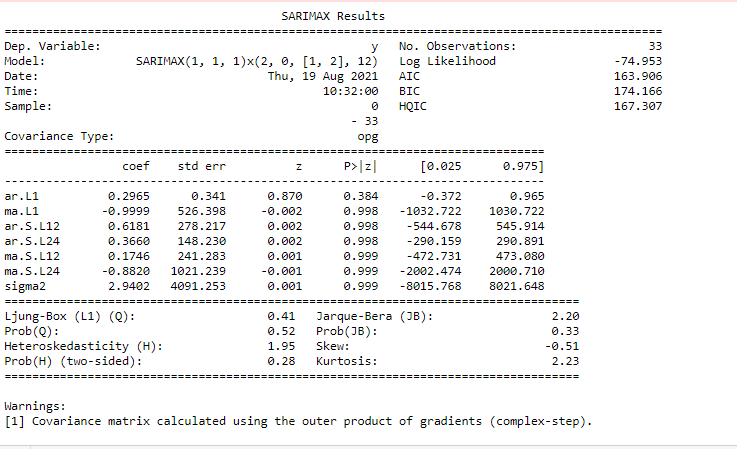


Now the AIC values are sorted :

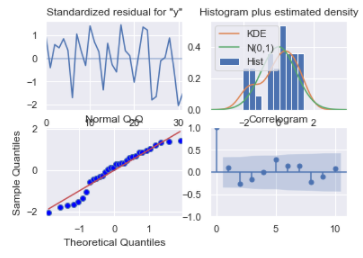


The best param combination is (1,1,1)(2,0,2,12)

The result is generated for the best combination.



The results of auto SARIMA model is plotted:

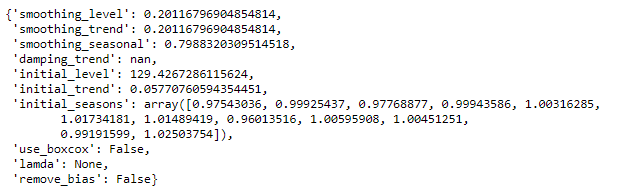


##### Triple exponential smoothing

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

The train and the test set is copied for the TES train and TES test set which is named as TES\_train\_I and TES\_test\_I.

The TES model is created with trend as and seasonal as multiplicative.

The TES model is fit to params :

The alpha = 0.2011,Beta= 0.2011, gamma= 0.7988

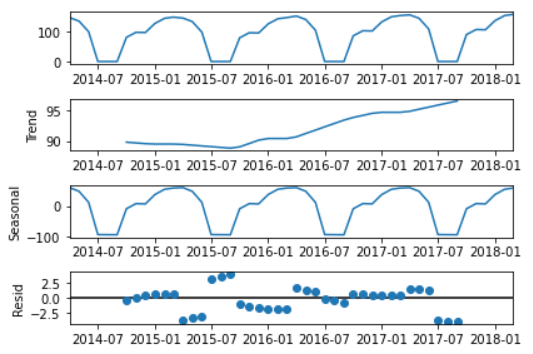
Using the above values the prediction is done.

COOLERS:

The time series are plotted for the Coolers data to understand the sales behaviour.

Decomposition of the time series and plotting of the same.

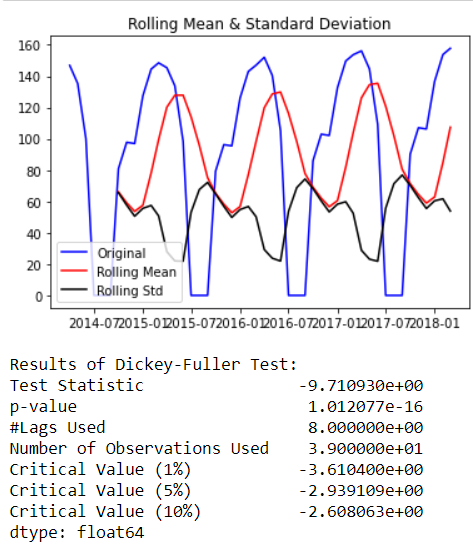
Additive decomposition:



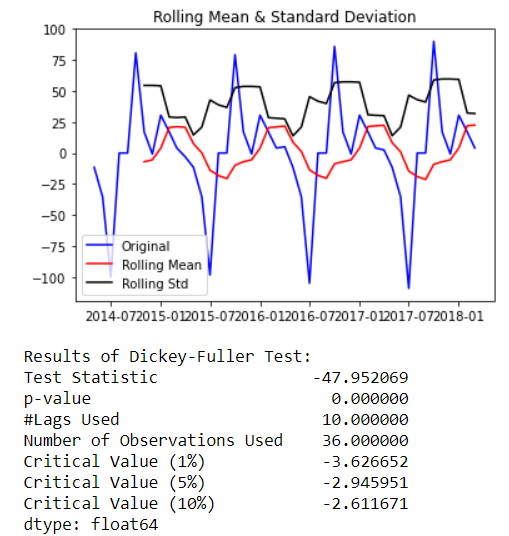
There is a seasonal pattern of sales also there is an increasing trend.

Not able to plot multiplicative decomposition since we have 0 values in sales of coolers.

Checking the stationarity for the Dry Iron dataset:

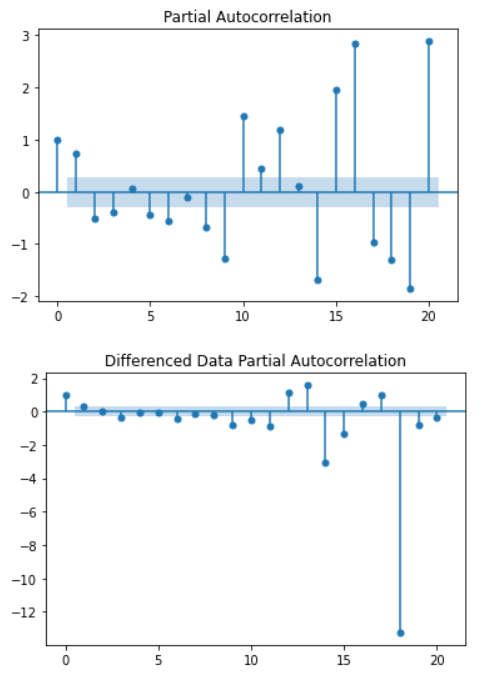


*Let us take a difference of order 1 and check whether the Time Series is stationary or not.*



Now the series are stationary the rolling std goes above the mean and the p value is less than 0.5.

Plot the Auto correlation and Partial Auto correlation function plots:





From the above plots we can say that there is some seasonality.

Splitting of Training and testing data:

April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set which are named as train2 and test2.

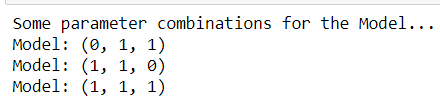
Now the model building are done based on the train and test set.

##### Auto ARIMA

**An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):**

The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 2

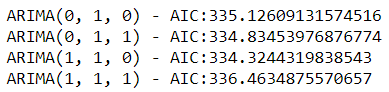
We have kept the value of d as 1 as we need to take a difference of the series to make it stationary.



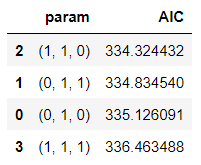
An empty Data frame is created with column names only



The AIC values for the different params are generated

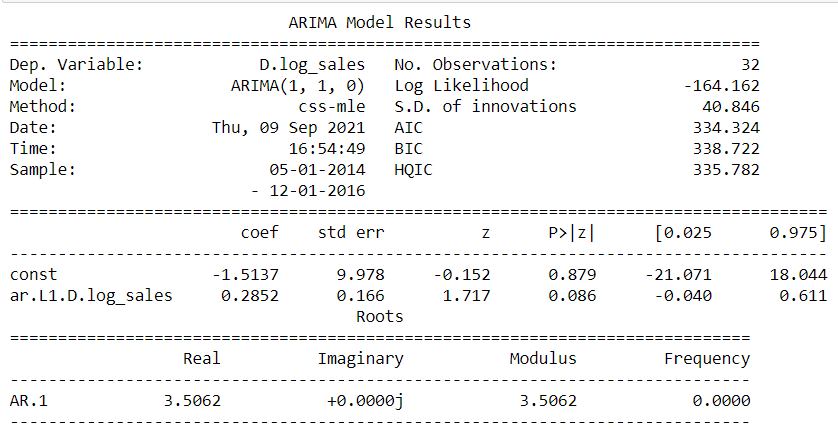


The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



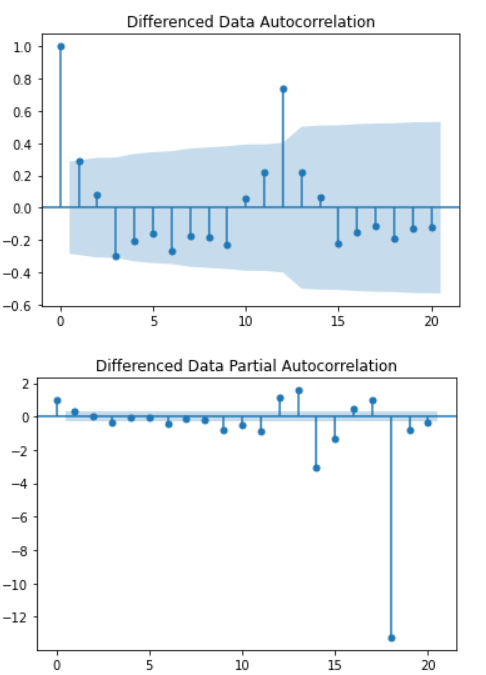
The lowest AIC value is for 1,1,0 param which is further used to get the result summary.

The results for the auto ARIMA for the coolers:



Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.

##### Manual ARIMA

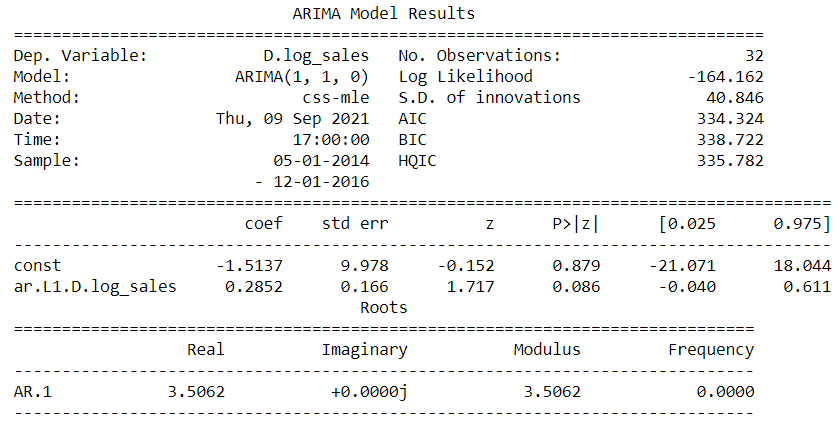


Here, we have taken alpha=0.05.

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 1.

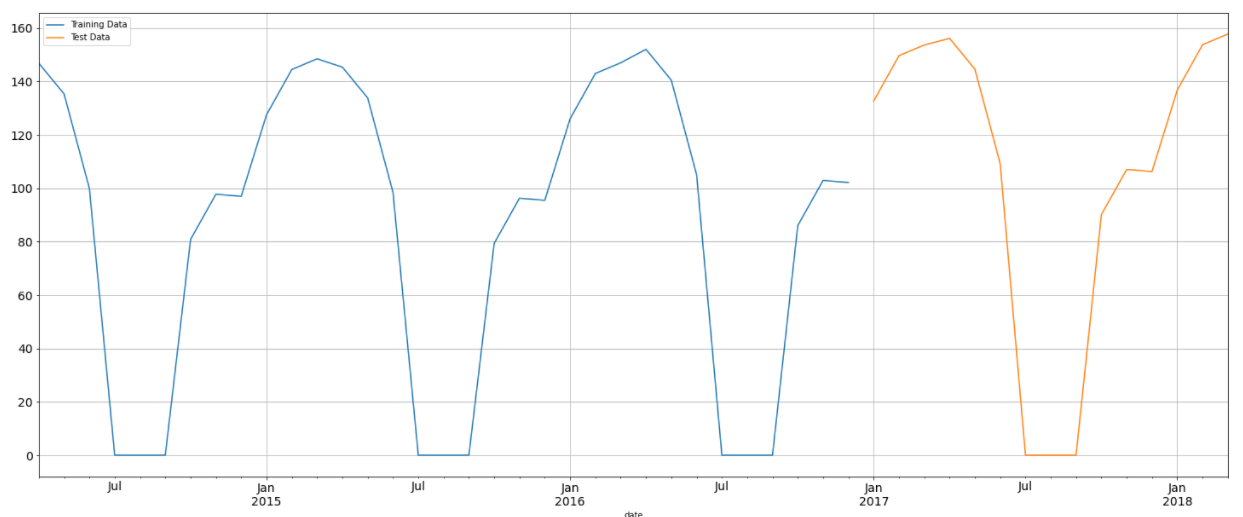
The result of manual ARIMA :

The result is generated considering the p as 1 and q as 0 and differencing to 1.

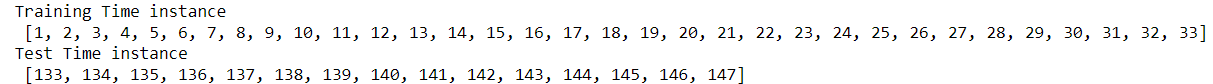


##### Linear Regression

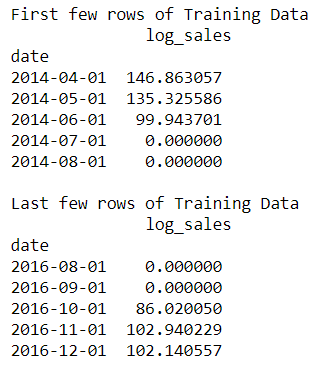
The train and test set for the Coolers data is plotted graphically.



The train and the test time instance:

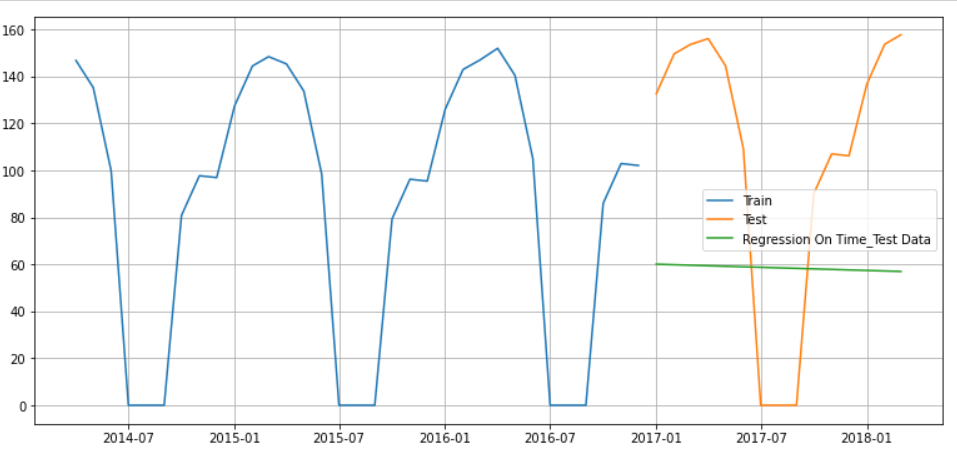


First few rows of the training and test data:

****

the linear regression is imported from sklearn linear model library.

The linear regression model is fitted to the time and the log\_sales.



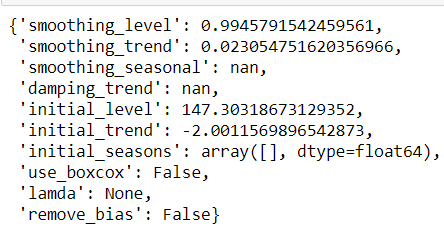
##### Triple exponential smoothing

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

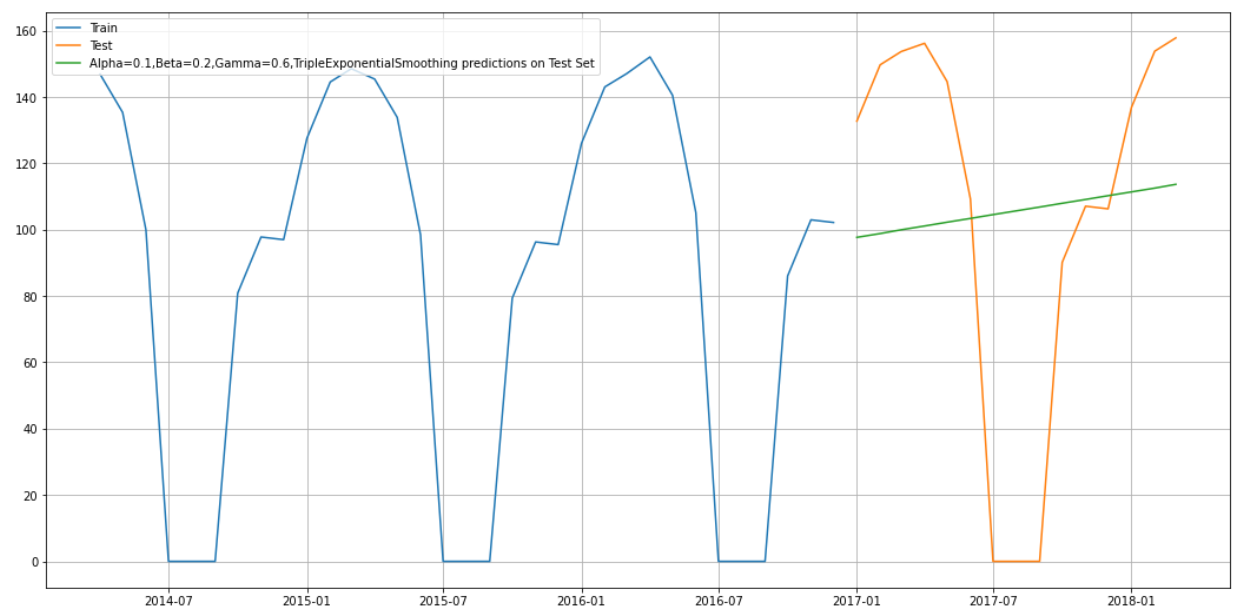
The train and the test set is copied for the TES train and TES test set which is named as TES\_train3 and TES\_test3.

The TES model is created with trend as and seasonal as multiplicative.

The TES model is fit to params :



The alpha =0.99,Beta= 0.02, gamma= NAN

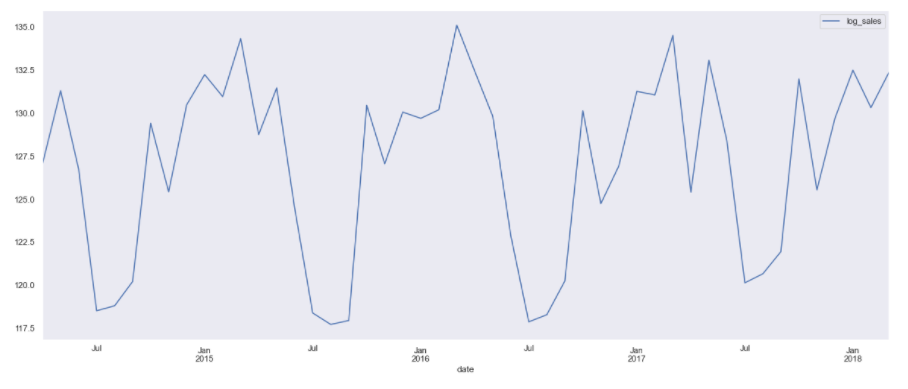


Using the above values the prediction is done.

### City Based Model

#### Bangalore

The time series are plotted for the Bangalore data to understand the sales behaviour.

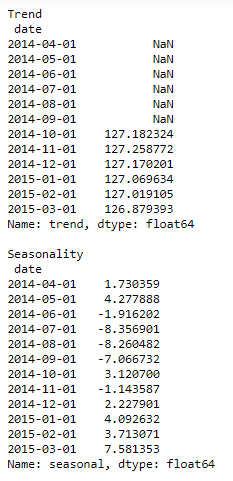


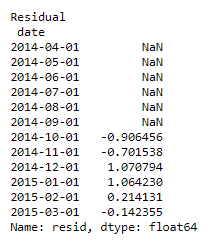
There is a seasonal pattern of sales.

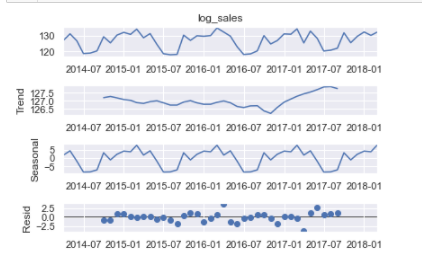
Decomposition of the time series and plotting of the same.

Additive decomposition:

The head of the trend, seasonality, residual are shown as follows:



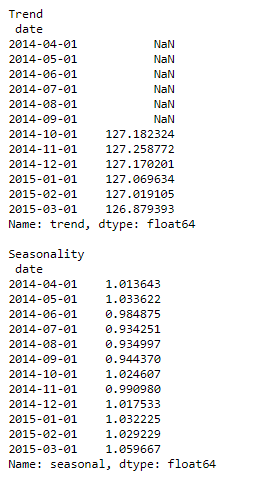


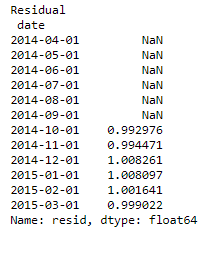


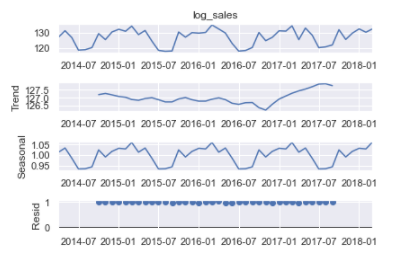
we see that the residuals are located around 0 from the plot of residuals in the decomposition.

Multiplicative decomposition:

The head of the trend, seasonality and residual are shown as follows:

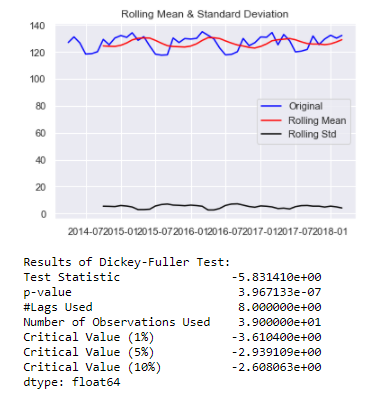




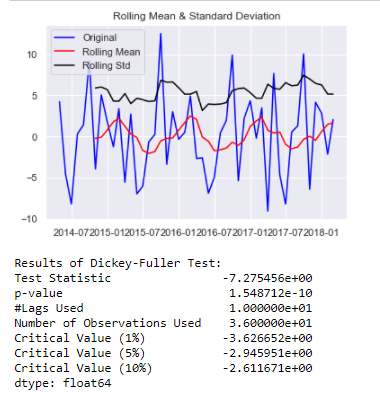


for the multiplicative model lot of residuals are located around 1.

Checking the stationarity for the Bangalore dataset:

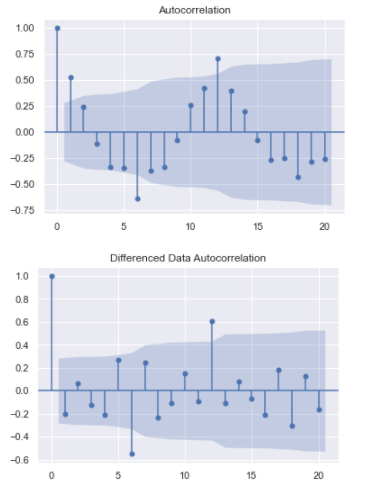


Let us take a difference of order 1 and check whether the Time Series is stationary or not.



Now the series are stationary the rolling std goes above the mean and the p value is less than 0.5.

Plot the Auto correlation and Partial Auto correlation function plots:





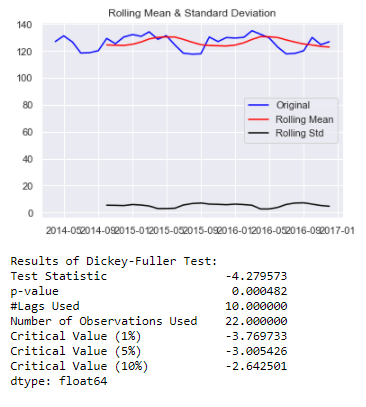
From the above plots we can say that there is some seasonality.

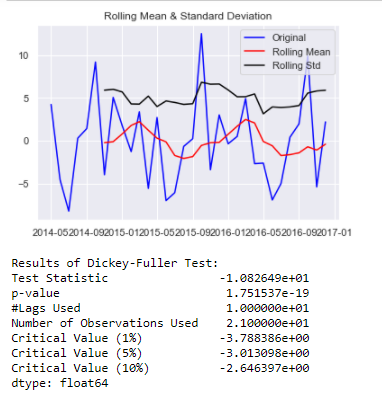
Splitting of Training and testing data:

April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set which are named as train1b and test1b.

Now the model building are done based on the train and test set.

Check for stationarity of the Training Data Time Series.



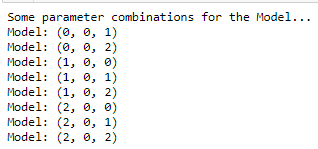


We see that after taking a difference of order 1 the series have become stationary at α = 0.05.

**An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):**

##### Auto ARIMA

The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 3 and d as 0.



An empty Data frame is created with column names only



The AIC values for the different params are generated

ARIMA(0, 0, 0) - AIC:208.83430762422

ARIMA(0, 0, 1) - AIC:200.5446109650528

ARIMA(0, 0, 2) - AIC:189.8142734011679

ARIMA(1, 0, 0) - AIC:197.56478961358795

ARIMA(1, 0, 1) - AIC:199.30012583337555

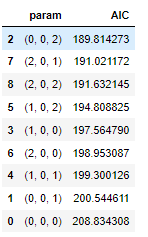
ARIMA(1, 0, 2) - AIC:194.80882460128595

ARIMA(2, 0, 0) - AIC:198.95308721751795

ARIMA(2, 0, 1) - AIC:191.02117190447888

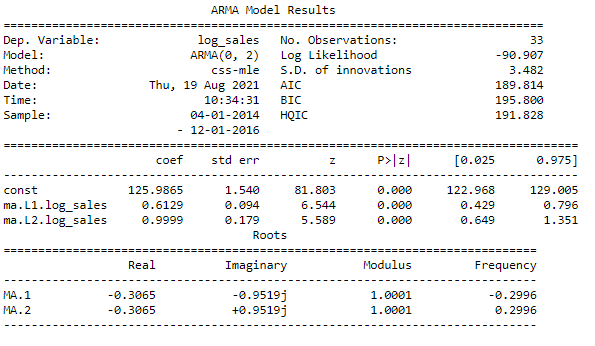
ARIMA(2, 0, 2) - AIC:191.63214543115296

The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



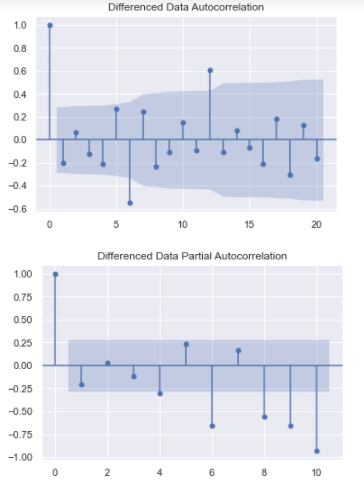
The lowest AIC value is for 0,0,2 param which is further used to get the result summary.

The results for the auto ARIMA for the Bangalore:



**Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.**

##### Manual ARIMA

****

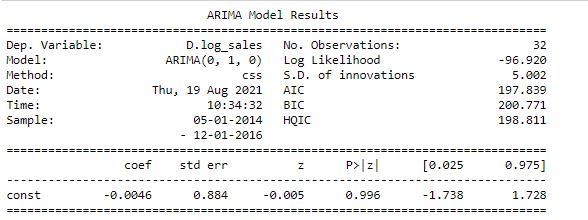
Here, we have taken alpha=0.05.

P= 0 and q= 0

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.

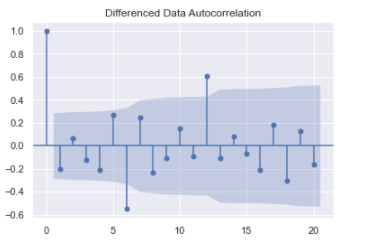
The result of manual ARIMA :

The result is generated considering the p as 0 and q as 0 and differencing to 1.



**Build an Automated version of a SARIMA model for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC).**

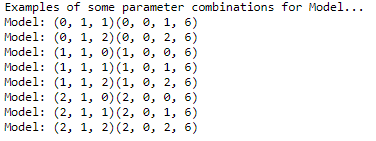
##### SARIMA



From the above differenced auto correlation plot we can get the seasonality value with which auto SARIMA model is built.

We see that there can be a seasonality of **6** as well as 12. We will run our auto SARIMA models by setting seasonality both as **6** and 12.

These are the param combinations which we get when the seasonality is set to **6**.



An empty dataset is generated to run through the loop.



SARIMA(0, 1, 0)x(0, 0, 0, 6) - AIC:190.06541657446513

SARIMA(0, 1, 0)x(0, 0, 1, 6) - AIC:151.9695891793937

SARIMA(0, 1, 0)x(0, 0, 2, 6) - AIC:114.35648290352103

SARIMA(0, 1, 0)x(1, 0, 0, 6) - AIC:152.03815121076553

SARIMA(0, 1, 0)x(1, 0, 1, 6) - AIC:135.65695079593405

SARIMA(0, 1, 0)x(1, 0, 2, 6) - AIC:100.55521802439979

SARIMA(0, 1, 0)x(2, 0, 0, 6) - AIC:89.97348923674738

SARIMA(0, 1, 0)x(2, 0, 1, 6) - AIC:91.40215303922274

SARIMA(0, 1, 0)x(2, 0, 2, 6) - AIC:90.00612125316852

SARIMA(0, 1, 1)x(0, 0, 0, 6) - AIC:186.02111579627314

SARIMA(0, 1, 1)x(0, 0, 1, 6) - AIC:148.01308925261537

SARIMA(0, 1, 1)x(0, 0, 2, 6) - AIC:108.34898843013968

SARIMA(0, 1, 1)x(1, 0, 0, 6) - AIC:153.3245685631622

SARIMA(0, 1, 1)x(1, 0, 1, 6) - AIC:131.01175803310633

SARIMA(0, 1, 1)x(1, 0, 2, 6) - AIC:95.75185624689851

SARIMA(0, 1, 1)x(2, 0, 0, 6) - AIC:89.35517429649133

SARIMA(0, 1, 1)x(2, 0, 1, 6) - AIC:90.16289264412285

SARIMA(0, 1, 1)x(2, 0, 2, 6) - AIC:81.00616098745303

SARIMA(0, 1, 2)x(0, 0, 0, 6) - AIC:178.99231627320495

SARIMA(0, 1, 2)x(0, 0, 1, 6) - AIC:144.0906107359565

SARIMA(0, 1, 2)x(0, 0, 2, 6) - AIC:104.48239047045199

SARIMA(0, 1, 2)x(1, 0, 0, 6) - AIC:146.95615995342706

SARIMA(0, 1, 2)x(1, 0, 1, 6) - AIC:117.3876890804542

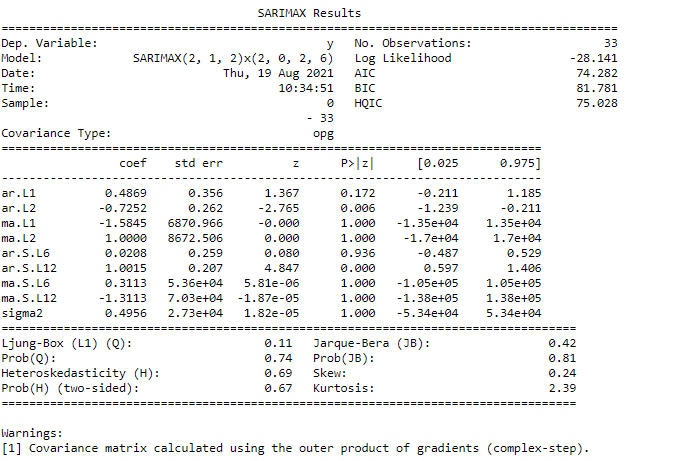
SARIMA(0, 1, 2)x(1, 0, 2, 6) - AIC:86.54487100991456

The AIC values for the param combination are listed above. Now these are sorted ascending order.

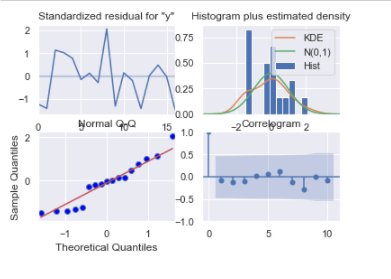


The best param will be (2,1,2)(2,0,2,**6**) with the lowest AIC value.

With this the auto SARIMA summary is generated.



Now the results of auto SARIMA is plotted:



Setting the seasonality as 12 for the second iteration of the auto SARIMA model.

The param combination when the seasonality is set 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)

The empty data frame is defined:



The AIC values for the different param combination are listed as follows:

SARIMA(0, 1, 0)x(0, 0, 0, 12) - AIC:190.06541657446513

SARIMA(0, 1, 0)x(0, 0, 1, 12) - AIC:112.49933965200827

SARIMA(0, 1, 0)x(0, 0, 2, 12) - AIC:49.6448828338513

SARIMA(0, 1, 0)x(1, 0, 0, 12) - AIC:90.15485429307493

SARIMA(0, 1, 0)x(1, 0, 1, 12) - AIC:88.58725895068142

SARIMA(0, 1, 0)x(1, 0, 2, 12) - AIC:31.627851768924824

SARIMA(0, 1, 0)x(2, 0, 0, 12) - AIC:40.50033681980945

SARIMA(0, 1, 0)x(2, 0, 1, 12) - AIC:42.500336828025

SARIMA(0, 1, 0)x(2, 0, 2, 12) - AIC:33.20520717756418

SARIMA(0, 1, 1)x(0, 0, 0, 12) - AIC:186.02111579627314

SARIMA(0, 1, 1)x(0, 0, 1, 12) - AIC:106.64086652802675

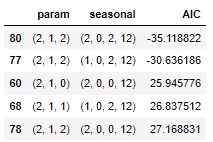
SARIMA(0, 1, 1)x(0, 0, 2, 12) - AIC:44.74030840064366

SARIMA(0, 1, 1)x(1, 0, 0, 12) - AIC:88.67358559838242

SARIMA(0, 1, 1)x(1, 0, 1, 12) - AIC:79.92546504396486

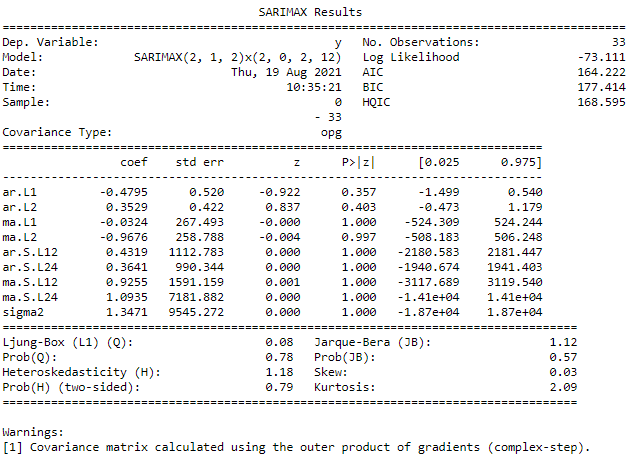
SARIMA(0, 1, 1)x(1, 0, 2, 12) - AIC:28.19780713477483

Now the AIC values are sorted :

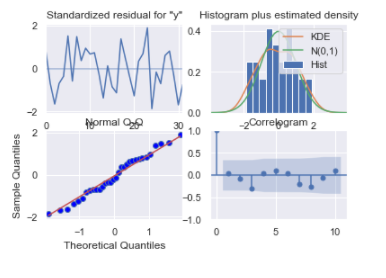


The best param combination is (2,1,1)(2,0,2,12)

The result is generated for the best combination.



The results of auto SARIMA model is plotted:



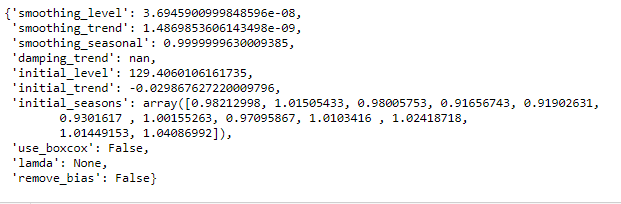
##### Triple exponential smoothing

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

The train and the test set is copied for the TES train and TES test set which are named as TES\_trainB and TES\_testB.

The TES model is created with trend as and seasonal as multiplicative.

The TES model is fit to params :



The alpha = 3.6945900999848596e-08,

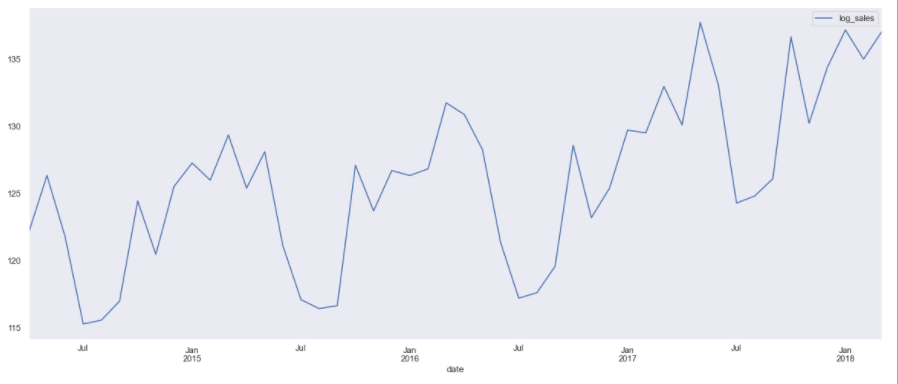
Beta= 1.4869853606143498e-09,

Gamma= 0.9999999630009385

Using the above values the prediction is done.

#### Kolkata

The time series are plotted for the Kolkata data to understand the sales behaviour.

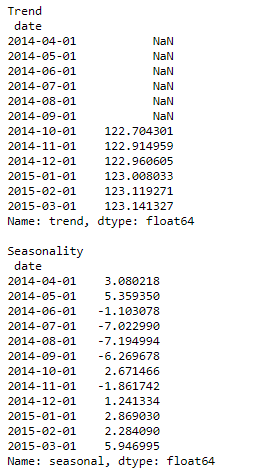


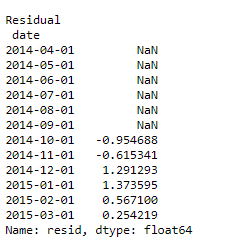
There is a seasonal pattern of sales and there is an upward trend.

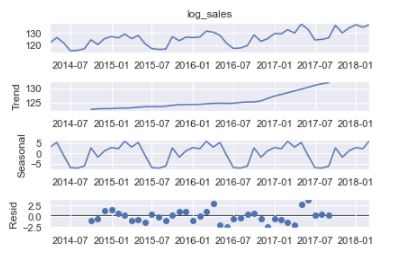
Decomposition of the time series and plotting of the same.

Additive decomposition:

The head of the trend, seasonality, residual are shown as follows:



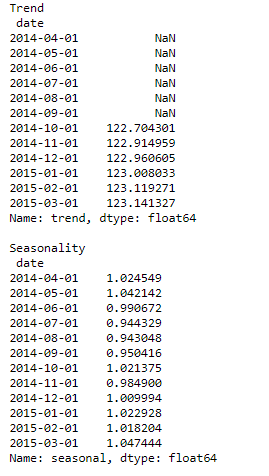


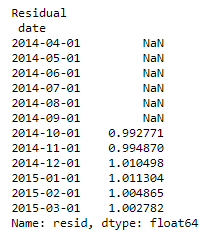


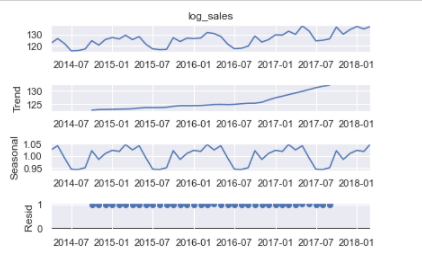
we see that the residuals are located around 0 from the plot of residuals in the decomposition.

Multiplicative decomposition:

The head of the trend, seasonality and residual are shown as follows:

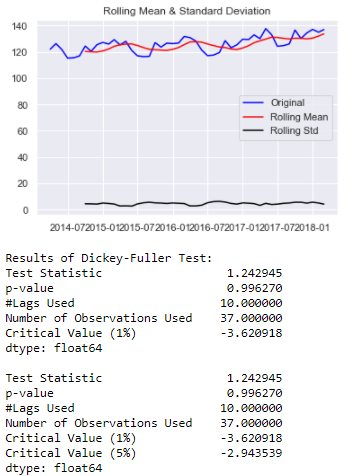




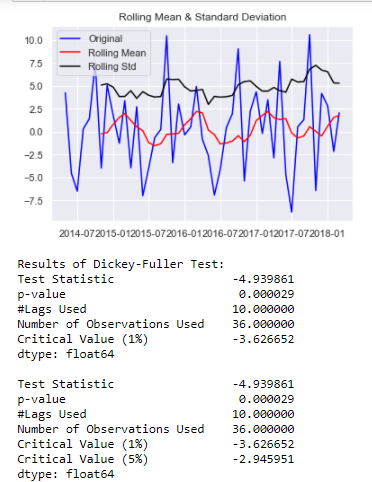


for the multiplicative model lot of residuals are located around 1.

Checking the stationarity for the Kolkata dataset:



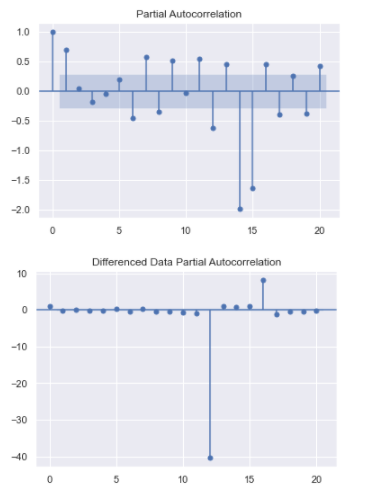
Let us take a difference of order 1 and check whether the Time Series is stationary or not.



Now the series are stationary the rolling std goes above the mean and the p value is less than 0.5.

Plot the Auto correlation and Partial Auto correlation function plots:





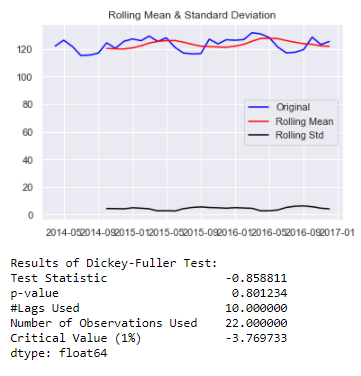
From the above plots we can say that there is some seasonality.

Splitting of Training and testing data:

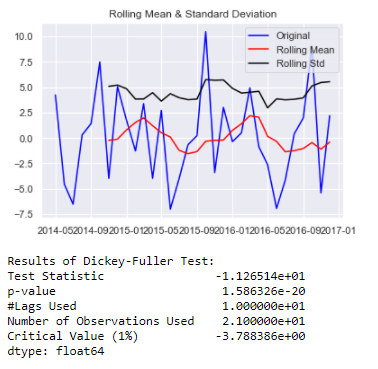
April 2014 is the starting year and the ending year is march of 2018 of the data set. Now the splitting of dataset is done that is from the year April 2014 to December 2016 is the training data and January of 2017 to march 2018 is the testing set which are named as trainK and testK.

Now the model building are done based on the train and test set.

Check for stationarity of the Training Data Time Series.



The training data time series is not stationary.



We see that after taking a difference of order 1 the series have become stationary at α = 0.05.

**An Automated version of an ARIMA model is built for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC):**

##### Auto ARIMA

The loop is created to helps us in getting a combination of different parameters of p and q in the range of 0 and 3 and d as 0.

Some parameter combinations for the Model...

Model: (0, 0, 1)

Model: (0, 0, 2)

Model: (1, 0, 0)

Model: (1, 0, 1)

Model: (1, 0, 2)

Model: (2, 0, 0)

Model: (2, 0, 1)

Model: (2, 0, 2)

An empty Data frame is created with column names only



The AIC values for the different params are generated

ARIMA(0, 0, 0) - AIC:200.1089229716399

ARIMA(0, 0, 1) - AIC:192.540049155766

ARIMA(0, 0, 2) - AIC:184.2297056633583

ARIMA(1, 0, 0) - AIC:189.59800326808693

ARIMA(1, 0, 1) - AIC:191.4475904854787

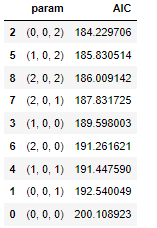
ARIMA(1, 0, 2) - AIC:185.83051437612357

ARIMA(2, 0, 0) - AIC:191.26162073167623

ARIMA(2, 0, 1) - AIC:187.83172457149362

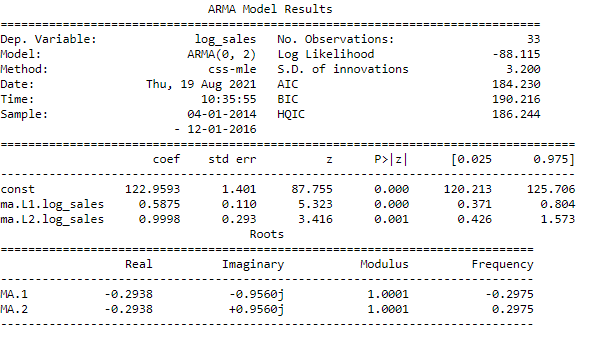
ARIMA(2, 0, 2) - AIC:186.00914244370557

The above AIC values are sorted in the ascending order to get the parameters for the minimum AIC value



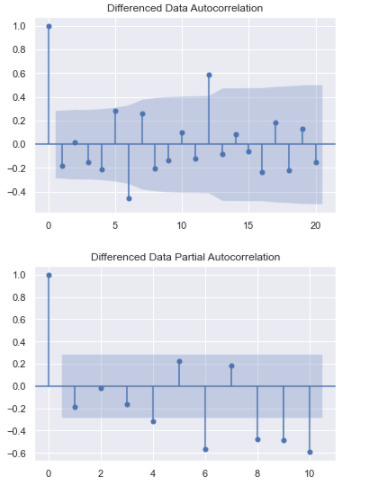
The lowest AIC value is for 0,0,2 param which is further used to get the result summary.

The results for the auto ARIMA for the Kolkata:



**Build a version of the ARIMA model for which the best parameters are selected by looking at the ACF and the PACF plots.**

##### **Manual ARIMA**

****

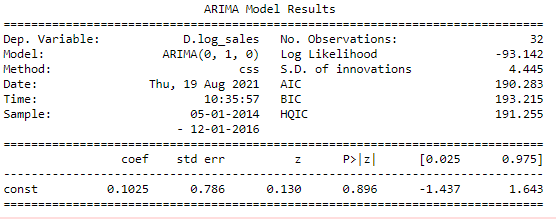
Here, we have taken alpha=0.05.

P= 0 and q= 0

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.

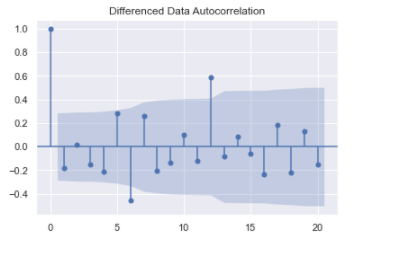
The result of manual ARIMA :

The result is generated considering the p as 0 and q as 0 and differencing to 1.



**Build an Automated version of a SARIMA model for which the best parameters are selected in accordance with the lowest Akaike Information Criteria (AIC).**

##### SARIMA



From the above differenced auto correlation plot we can get the seasonality value with which auto SARIMA model is built.

We see that there can be a seasonality of **6** as well as 12. We will run our auto SARIMA models by setting seasonality both as **6** and 12.

These are the param combinations which we get when the seasonality is set to **6**.

Examples of some parameter combinations for Model...

Model: (0, 1, 1)(0, 0, 1, 6)

Model: (0, 1, 2)(0, 0, 2, 6)

Model: (1, 1, 0)(1, 0, 0, 6)

Model: (1, 1, 1)(1, 0, 1, 6)

Model: (1, 1, 2)(1, 0, 2, 6)

Model: (2, 1, 0)(2, 0, 0, 6)

Model: (2, 1, 1)(2, 0, 1, 6)

Model: (2, 1, 2)(2, 0, 2, 6)

An empty dataset is generated to run through the loop.



SARIMA(0, 1, 0)x(0, 0, 0, 6) - AIC:182.57313543246144

SARIMA(0, 1, 0)x(0, 0, 1, 6) - AIC:147.38006719345933

SARIMA(0, 1, 0)x(0, 0, 2, 6) - AIC:110.50801717128168

SARIMA(0, 1, 0)x(1, 0, 0, 6) - AIC:148.7912588968967

SARIMA(0, 1, 0)x(1, 0, 1, 6) - AIC:133.5097790428919

SARIMA(0, 1, 0)x(1, 0, 2, 6) - AIC:99.46898285299025

SARIMA(0, 1, 0)x(2, 0, 0, 6) - AIC:88.24398598359974

SARIMA(0, 1, 0)x(2, 0, 1, 6) - AIC:89.96679558701743

SARIMA(0, 1, 0)x(2, 0, 2, 6) - AIC:88.6771336004403

SARIMA(0, 1, 1)x(0, 0, 0, 6) - AIC:178.43065703999247

SARIMA(0, 1, 1)x(0, 0, 1, 6) - AIC:143.31001727102978

SARIMA(0, 1, 1)x(0, 0, 2, 6) - AIC:103.95305686838003

SARIMA(0, 1, 1)x(1, 0, 0, 6) - AIC:150.47459802104723

SARIMA(0, 1, 1)x(1, 0, 1, 6) - AIC:129.75780051321607

SARIMA(0, 1, 1)x(1, 0, 2, 6) - AIC:95.15846977701763

SARIMA(0, 1, 1)x(2, 0, 0, 6) - AIC:87.27938286707348

SARIMA(0, 1, 1)x(2, 0, 1, 6) - AIC:88.42626645267205

SARIMA(0, 1, 1)x(2, 0, 2, 6) - AIC:78.72117910412135

SARIMA(0, 1, 2)x(0, 0, 0, 6) - AIC:172.36359435834237

SARIMA(0, 1, 2)x(0, 0, 1, 6) - AIC:136.35878493182958

SARIMA(0, 1, 2)x(0, 0, 2, 6) - AIC:100.13183634131317

SARIMA(0, 1, 2)x(1, 0, 0, 6) - AIC:144.0016028550539

SARIMA(0, 1, 2)x(1, 0, 1, 6) - AIC:116.77715991134406

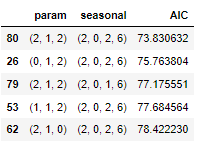
SARIMA(0, 1, 2)x(1, 0, 2, 6) - AIC:88.45603109083784

SARIMA(0, 1, 2)x(2, 0, 0, 6) - AIC:86.76996599424517

SARIMA(0, 1, 2)x(2, 0, 1, 6) - AIC:87.79359189142262

SARIMA(0, 1, 2)x(2, 0, 2, 6) - AIC:75.76380434886289

The AIC values for the param combination are listed above. Now these are sorted ascending order.

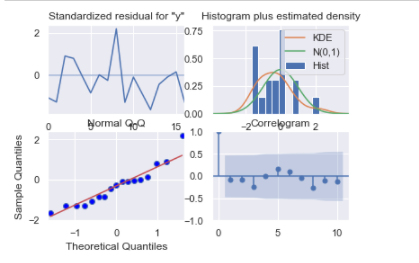


The best param will be (2,1,2)(2,0,2,**6**) with the lowest AIC value.

With this the auto SARIMA summary is generated.



Now the results of auto SARIMA is plotted:



Setting the seasonality as 12 for the second iteration of the auto SARIMA model.

The param combination when the seasonality is set 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)

The empty data frame is defined:



The AIC values for the different param combination are listed as follows:

SARIMA(0, 1, 0)x(0, 0, 0, 12) - AIC:182.57313543246144

SARIMA(0, 1, 0)x(0, 0, 1, 12) - AIC:108.51119696041505

SARIMA(0, 1, 0)x(0, 0, 2, 12) - AIC:48.7833553152237

SARIMA(0, 1, 0)x(1, 0, 0, 12) - AIC:89.67442353246493

SARIMA(0, 1, 0)x(1, 0, 1, 12) - AIC:88.04857662283861

SARIMA(0, 1, 0)x(1, 0, 2, 12) - AIC:30.546265024684434

SARIMA(0, 1, 0)x(2, 0, 0, 12) - AIC:40.629045455738215

SARIMA(0, 1, 0)x(2, 0, 1, 12) - AIC:42.629045457068024

SARIMA(0, 1, 0)x(2, 0, 2, 12) - AIC:31.54773011306026

SARIMA(0, 1, 1)x(0, 0, 0, 12) - AIC:178.43065703999247

SARIMA(0, 1, 1)x(0, 0, 1, 12) - AIC:102.00541762524261

SARIMA(0, 1, 1)x(0, 0, 2, 12) - AIC:43.732770721920836

SARIMA(0, 1, 1)x(1, 0, 0, 12) - AIC:86.83686620970309

SARIMA(0, 1, 1)x(1, 0, 1, 12) - AIC:78.66227584349949

SARIMA(0, 1, 1)x(1, 0, 2, 12) - AIC:26.46028641233406

SARIMA(0, 1, 1)x(2, 0, 0, 12) - AIC:42.034441985502355

SARIMA(0, 1, 1)x(2, 0, 1, 12) - AIC:44.0344419419129

SARIMA(0, 1, 1)x(2, 0, 2, 12) - AIC:27.34573425401461

SARIMA(0, 1, 2)x(0, 0, 0, 12) - AIC:172.36359435834237

SARIMA(0, 1, 2)x(0, 0, 1, 12) - AIC:98.24014896159083

SARIMA(0, 1, 2)x(0, 0, 2, 12) - AIC:39.23851873815824

SARIMA(0, 1, 2)x(1, 0, 0, 12) - AIC:86.40491397671178

SARIMA(0, 1, 2)x(1, 0, 1, 12) - AIC:75.94317792486572

SARIMA(0, 1, 2)x(1, 0, 2, 12) - AIC:26.486605886556738

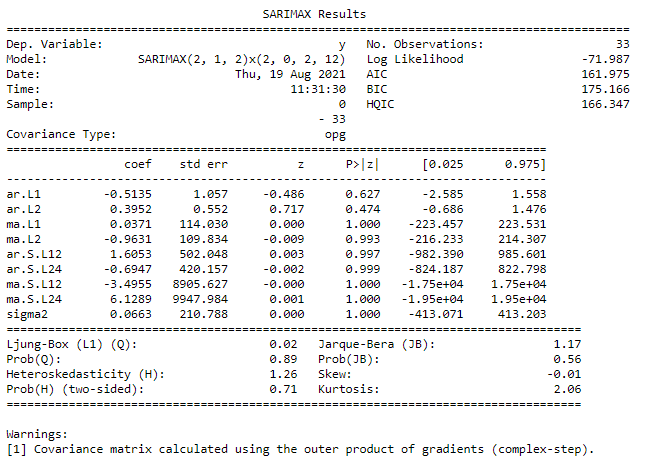
SARIMA(0, 1, 2)x(2, 0, 0, 12) - AIC:40.41010156365998

Now the AIC values are sorted :

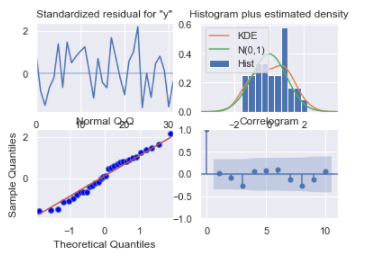


The best param combination is (2,1,2)(2,0,2,12)

The result is generated for the best combination.



The results of auto SARIMA model is plotted:



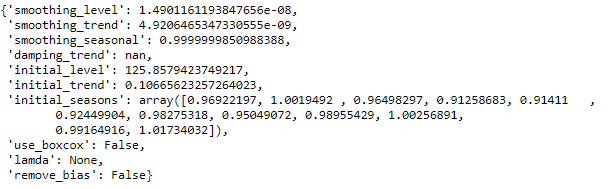
##### Triple exponential smoothing

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

The train and the test set is copied for the TES train and TES test set which are named as TES\_trainK and TES\_testK.

The TES model is created with trend as and seasonal as multiplicative.

The TES model is fit to params :



The alpha = 1.49,

Beta= 4.92,

Gamma= 0.99

## Model Evaluation

Test of predictive model against the test set using various appropriate performance metrics

### Product Based Models

#### Induction Cookers

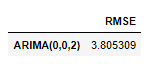
##### Auto ARIMA

The results of auto ARIMA summary is used to forecast with the length of the test set to get predicted auto ARIMA.

The mean squared error are imported from sklearn metrics library to get the root mean squared error score.

The RMSE for the predicted auto ARIMA is 

Now this score is inserted to dataset form with respect to best param chosen combination.

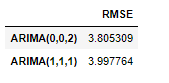


##### Manual ARIMA

The results of manual ARIMA summary is used to forecast with the length of the test set to get predicted manual ARIMA.

The RMSE for the predicted manual ARIMA is 

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous auto ARIMA RMSE score.

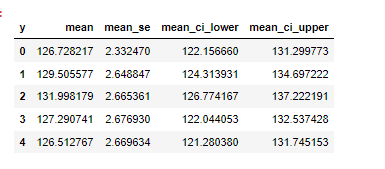


##### Auto SARIMA

Prediction on the Test Set using auto SARIMA model and evaluate the model.

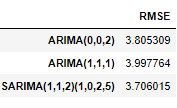
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA :



The RMSE for the predicted auto SARIMA is 

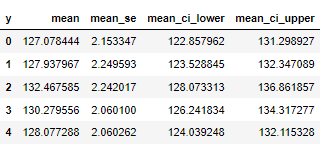
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



Prediction on the Test Set using auto SARIMA model for seasonality of 12 and evaluate the model.

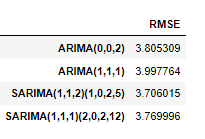
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA for seasonality as 12 :



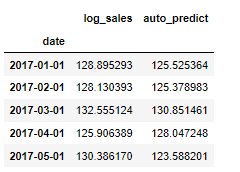
The RMSE for the predicted manual ARIMA is 

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.

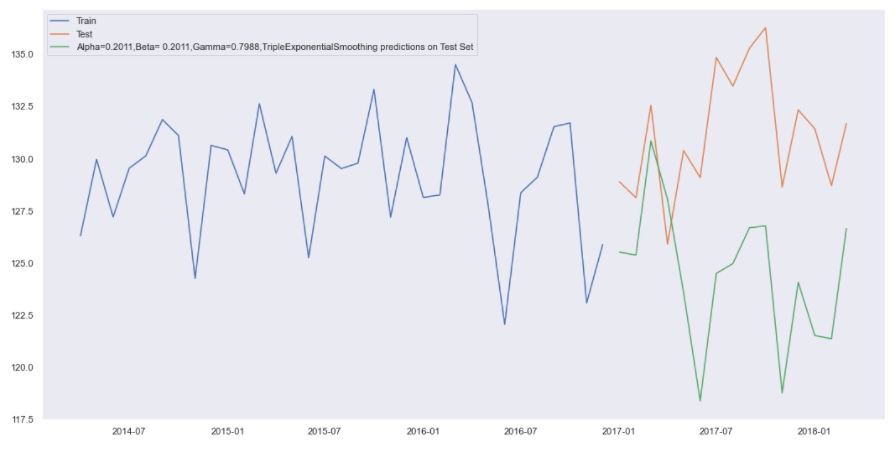


##### Triple Exponential Smoothing

Prediction on the test data is done to forecast to length of the test set:



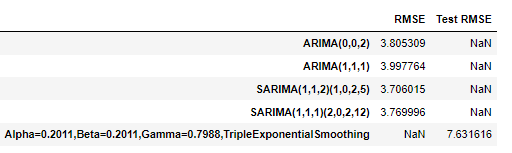
Plotting on both the Training and Test using autofit



The RMSE for the TES with the best alpha, beta, gamma values using the test set are:



Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score

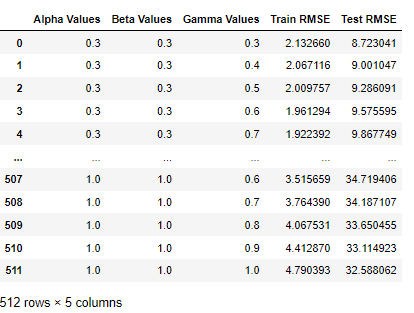


First we will define an empty data frame to store our values from the loop

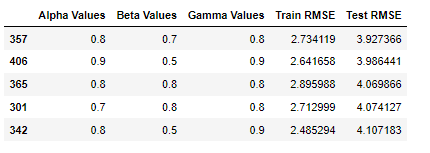


Now a for loop is generated with different alpha, beta, gamma values.

The result of different alpha , beta, gamma



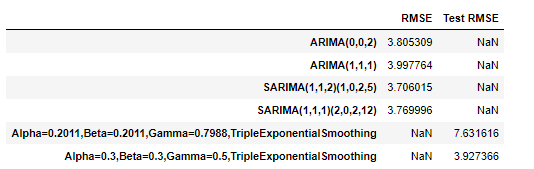
The same is sorted based on test RMSE,



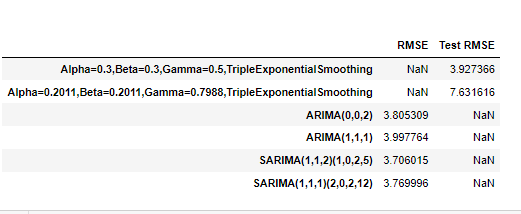
The best values are 0.3,0.3,0.5

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





Sorting of values based on the test RMSE.



### City Based Models

#### Bangalore

##### Auto ARIMA

The results of auto ARIMA summary is used to forecast with the length of the test set to get predicted auto ARIMA.

The mean squared error are imported from sklearn metrics library to get the root mean squared error score.

The RMSE for the predicted auto ARIMA is 5.755798128582992

Now this score is inserted to dataset form with respect to best param chosen combination.

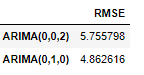


##### Manual ARIMA

The results of manual ARIMA summary is used to forecast with the length of the test set to get predicted manual ARIMA.

The RMSE for the predicted manual ARIMA is 4.862615931531502

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous auto ARIMA RMSE score.

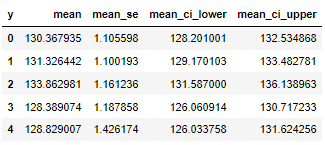


##### Auto SARIMA

Prediction on the Test Set using auto SARIMA model and evaluate the model.

The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA :



The RMSE for the predicted auto SARIMA is 2.727602635056308

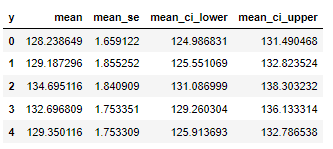
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



Prediction on the Test Set using auto SARIMA model for seasonality of 12 and evaluate the model.

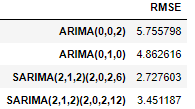
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA for seasonality as 12 :



The RMSE for the predicted manual ARIMA is 3.4511872082035095

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



##### Triple Exponential Smoothing

Prediction on the test data is done to forecast to length of the test set:



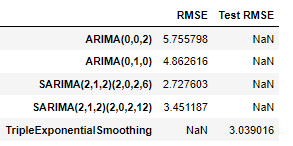
Plotting on both the Training and Test using autofit



The RMSE for the TES with the best alpha, beta, gamma values using the test set are:

Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 3.039

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score

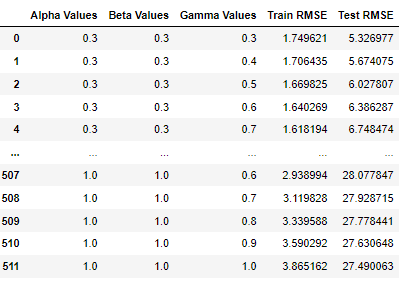


First we will define an empty data frame to store our values from the loop

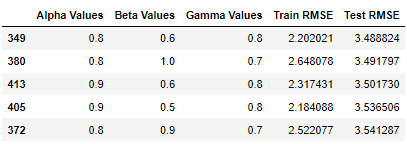


Now a for loop is generated with different alpha, beta, gamma values.

The result of different alpha , beta, gamma

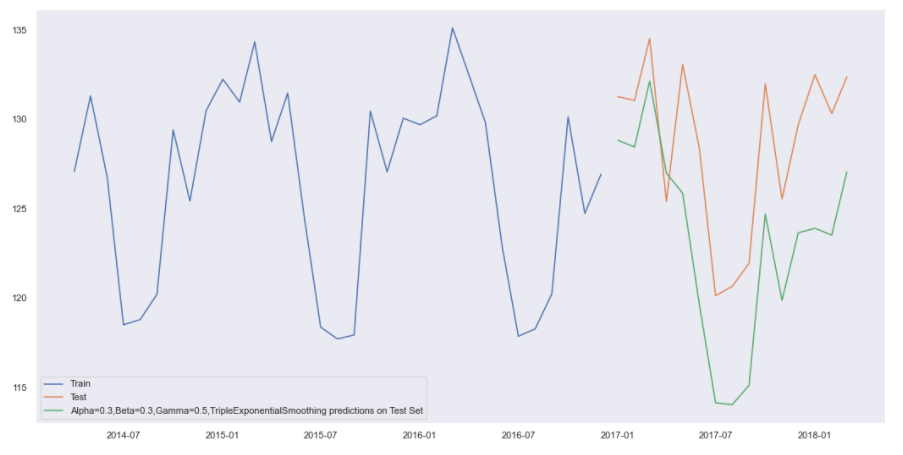


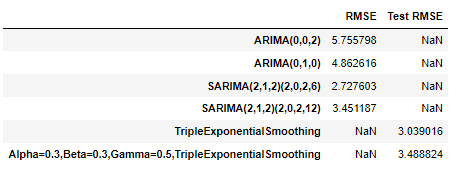
The same is sorted based on test RMSE,



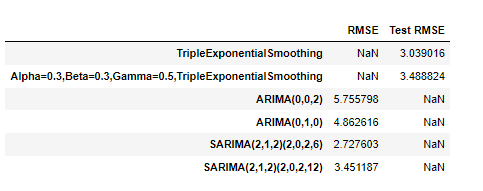
The best values are 0.3,0.3,0.5

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





Sorting of values based on the test RMSE.



#### Kolkata

##### Auto ARIMA

The results of auto ARIMA summary is used to forecast with the length of the test set to get predicted auto ARIMA.

The mean squared error are imported from sklearn metrics library to get the root mean squared error score.

The RMSE for the predicted auto ARIMA is 10.167474898927646

Now this score is inserted to dataset form with respect to best param chosen combination.



##### Manual ARIMA

The results of manual ARIMA summary is used to forecast with the length of the test set to get predicted manual ARIMA.

The RMSE for the predicted manual ARIMA is 7.117171395725669

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous auto ARIMA RMSE score.

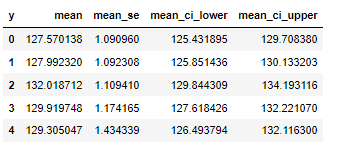


##### Auto SARIMA

Prediction on the Test Set using auto SARIMA model and evaluate the model.

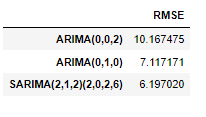
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA :



The RMSE for the predicted auto SARIMA is 6.197019914043381

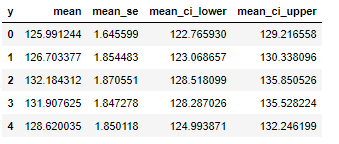
Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



Prediction on the Test Set using auto SARIMA model for seasonality of 12 and evaluate the model.

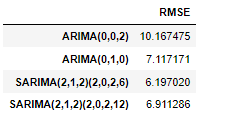
The results of auto SARIMA summary is used to forecast with the length of the test set to get predicted auto SARIMA.

The result of predicted auto SARIMA for seasonality as 12 :



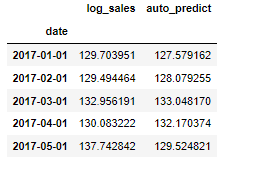
The RMSE for the predicted manual ARIMA is 6.911286325000788

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score.



##### Triple Exponential Smoothing

Prediction on the test data is done to forecast to length of the test set:



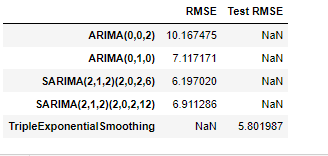
Plotting on both the Training and Test using autofit



The RMSE for the TES with the best alpha, beta, gamma values using the test set are:

Triple Exponential Smoothing Model forecast on the Test Data, RMSE is 5.802

Now this score is inserted to dataset form with respect to best param chosen combination along with the previous RMSE score

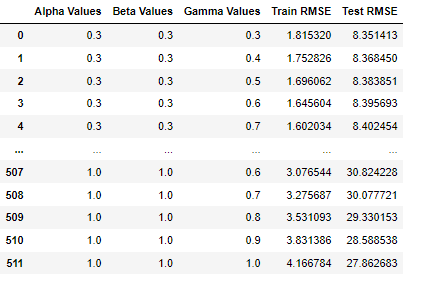


First we will define an empty data frame to store our values from the loop

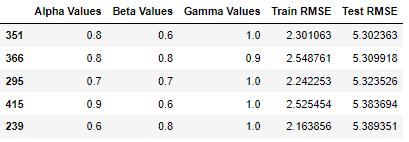


Now a for loop is generated with different alpha, beta, gamma values.

The result of different alpha , beta, gamma



The same is sorted based on test RMSE,

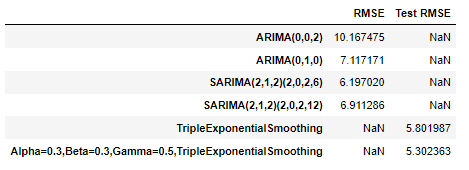


The best values are 0.3,0.3,0.5

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



RMSE scores for the Kolkata data:



Now they are sorted according to test RMSE:

