

TIMESERIES

COMMODITY PROBLEM



**Under the Guidance of:**

Mr Srikar Muppidi

Great Learning

Bangalore

**Report Submitted by:**

* Kevin Rajkumar - DSEFTBNOV18008
* Sukalp Shivam - DSEFTBNOV18015
* Manas Ghosh - DSEFTBNOV18011
* Gaurav Garhewal - DSEFTBNOV18006

**Batch:**

PGP-DSE- NOV-18 Bangalore Batch

“submitted towards partial fulfilment of the criteria for award of PGPDSE by GLIM”

Acknowledgement

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to our Mentor **Mr Srikar Muppidi** for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express my gratitude to members of Great learning for their kind co-operation and encouragement, which help me in completion of this project.

We also thank all the course faculty of the DSE program for providing us a strong foundation in various concepts of analytics and machine learning.

Our thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

Certification of Completion

I hereby certify that the project titled “Time Series Commodity problem” was undertaken and completed under my guidance and supervision by Kevin Rajkumar, Sukalp Shivam, Manas Ghosh and Gaurav Garhewal, students of the Nov 2018 batch of the Post Graduate Program in Data Science Engineering, Bangalore.

Mr. **Srikar Muppidi**

Date: 3rd April 2019

Contents

**Chapter 1: Project Overview5**

Project Objective5

Description5

Problem Statement5

Domain & Data Source5

Data Constraint5

**Chapter 2: About Modelling Process6**

Missing Values6

Handling Missing Data6

Model Techniques8

Decomposition Models8

Regression Models10

Evaluation Parameters11

**Chapter 3: Exploratory Data Analysis12**

Dataset12

Visualization12

Feature Engineering15

**Chapter 4: Modelling16**

Visualization16

Train-Validation-Test split18

Simple Exponential Model19

Holt’s Linear Model20

Holt-Winter Model21

ARIMA Model22

SARIMA Model23

**Chapter 5: Models Comparison24**

Model Comparison Matrix24

**Chapter 6: Commodities Forecast25**

Forecast25

**Chapter 7: Key Learning & Actionable Insight26**

**Chapter 8: Future Work27**

**Chapter 9: Bibliography28**

**ANNEXURES29**

Executive Summary

# **Background & Need to study:**

The retail price of key commodities is basically taken from Open Government Data Platform of India where prices of every commodity is varying with the different regions of India. The dataset contains retail prices for key commodities recorded around 75 key market centres in India. The granularity of the dataset is at day level. The country is divided into 5 regions i.e. (North, South, East, West, North-East) and each region has different a centre. The dataset contains the Prices/rates of commodities of different regions with respect to time which is from 1997 to 2015.The dataset has 9 commodities i.e. (Onion, Rice, Tea Loose, Tur/Arhar Dal, Sugar, Salt Pack, Milk, Tomato, Sunflower oil) and for each commodity we have prices per kg. The dataset is talking about the trend of each commodity, which is increasing and decreasing according to the data. So, we are basically analysing the trend of each commodity with respect to time and will forecast the trend for next 12 months.

# **Scope & Objectives:**

The scope is to identify trend of each commodity, which is increasing and decreasing according to the data. The objective is to study the monthly trend of most important commodities and then forecast for the next 12 months in each region of India.

# **Approach & Methodology:**

The data is cleaned and prepared to make it suitable for model building. Various time series-based algorithms are used to build a predictive model to predict the price/rates of top commodity for next 12 months. The machine learning models help to identify the trend and seasonality if exists. The models are evaluated using relevant model performance measures and thereby the most relevant model is chosen.

# **Key Learnings:**

From the historical data of per kilogram price of different commodities in various region, we are trying to predict the future price. From this project, we will be experimenting different time series modelling techniques using different libraries e,g: statsmodels.api ARIMA, SARIMA, Holts-winter, Holts linear trend etc. Learning Univariate Forecasting techniques and understand the seasonality, cyclicity, trend and residual factors of the time series.

Chapter 1: Project Overview

The study is about the price of the key commodities across the country. This data has been collected from an open government source. The data explains the overall price variance of across different regions of the country. To understand the different variance of the commodities is to find the key factors affecting these commodities and then forecast for next 12 months for each region.

# **Project Objective:**

Forecasting the price of the important commodities in different regions for next 12 months.

# **Description:**

* The dataset contains retail prices for key commodities recorded around 75 key market centres in India.
* The granularity of the dataset is at day level. The country is divided into 5 regions i.e. (North, South, East, West, North-East) and each region has different a centre.
* The dataset contains the Prices/rates of commodities of different regions with respect to time which is from 1997 to 2015.The dataset has 9 commodities i.e. (Onion, Rice, Tea Loose, Tur/Arhar Dal, Sugar, Salt Pack, Milk, Tomato, Sunflower oil) and for each commodity we have prices per kg.
* The dataset is talking about the trend of each commodity which is increasing and decreasing according to the data. So, we are basically analysing the trend of each commodity with respect to time and will forecast the trend for next 12 months.

# **Problem Statement:**

* The objective is to study the monthly trend of most important commodities and then forecast for the next 12 months in each region of India.

# **Domain & Data Source:**

* Retail
* The data has been taken from an open government source about the retail industry.

# **Data Constraint:**

* There were null values in the original dataset but in a different format, which became difficult to identify.
* There was no seasonality in the given data hence, we could not find a standardize increase or fall in the price.
* There is too much fluctuation in different major commodities hence it was difficult to conclude the forecast for nest 12 month.
* The onion data is very much unstable hence for this particular commodity the prices were very high in some region and very low in some region.

Chapter 2: About Modelling Process

# **Missing Values:**

In time series data, if there are missing values, there are two ways to deal with the incomplete data:

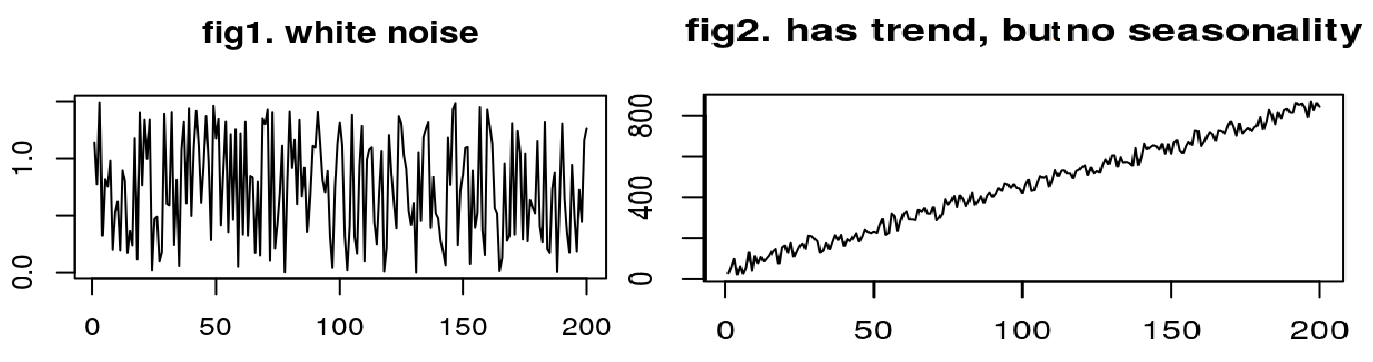
* omit the entire record that contains information.
* Impute the missing information.

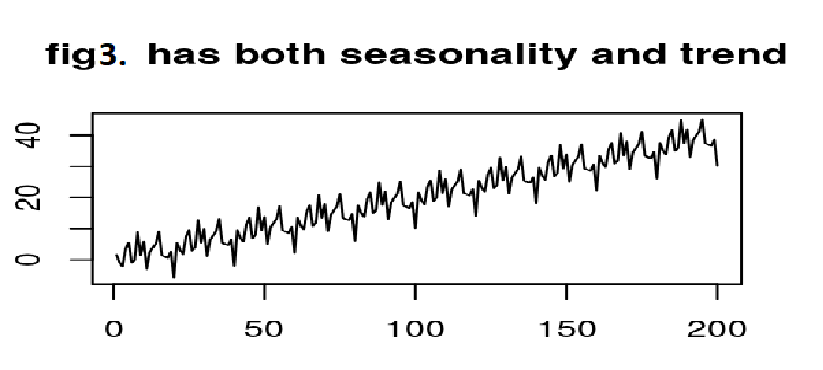
In Statistics, **missing data**, or **missing values**, occur when no data value is stored for the variable in an observation. Missing data are a common occurrence and can have a significant effect on the conclusions that can be drawn from the data.

# **Handling Missing Data:**

There are three types of time series data:

* no trend or seasonality (fig1)
* has trend, but no seasonality (fig2)
* have both trend and seasonality (fig3)





Data imputation for missing values varies based on the type of timeseries data. The imputation technique can be explained as:

**Data without trend and seasonality:**

One of the easiest methods to impute or estimating missing values to get a complete sample is replacing each of the missing value with mean or median or mode of the observed data for the variables which is also known as unconditional mean/median/mode imputation.

**Data with trend and without seasonality:**

Interpolation is the process of finding a value between two points on a line or curve. The last valid value before the missing value and the first value after the missing value is used for interpolation.

Line drawn with the equation: (Y - Y0 / X - X0) = (Y1 - Y0)/ (X1 – X0)

Point1: (X0, Y0)

Point2: (X1, Y1)

**Data with trend and seasonality:**

Seasonal adjustment is a statistical method to removes the seasonal component of the time series that exhibits a seasonal pattern. It is usually done when wanting to analyse the trend, and cyclical deviations from trend, of a time series independently of the seasonal components. And then interpolation is done.

In this project, we are dealing with commodities that are having year round production, which infers that there cannot be any seasonality expected from the these commodities. But a trend of price increase is expected in these commodities due to various reasons like inflation, increase in transportation, economic factor etc. So, in this project we will use the linear interpolation method to filling out the missing values.

# **Model Techniques:**

There are two main goals of Time series Analysis:

1. Identifying the nature of the phenomenon represented by the sequence of observations.
2. Forecasting (predicting future values of the time series variables)

Both of these goals require that the pattern of observed time series data is identified and more or less formally described. Once the pattern is established, we can interpret and integrate it with other data (i.e., use it in our theory of the investigated phenomenon, e.g., seasonal commodity prices).

Time series forecasting uses information regarding historical values and associated patterns to predict future activity. Most often, this relates to trend analysis, cyclical fluctuation analysis and issues of seasonality. As with all forecasting methods, success is not guaranteed. The Dataset is having different Commodities where prices are varying with the regions and centres. The time series forecasting methods can be classified as:

# **Decomposition Models:**

In Decomposition technique, we decompose the timeseries into trend, seasonality and irregular components and in Regression technique we make use of auto-regression and moving average. In this technique, we are dividing the series into three components:

* Trend Component: a long-term monotonic change of the average level of the time series.
* Seasonal Component: fluctuations in time series that recur during specific time periods.
* Residual Component that represents all the influences on the time series that are not explained by the other two components.

1. **Simple Exponential Smoothing (SES):**

The simplest of the exponentially smoothing methods is naturally called **simple exponential smoothing** (SES). This method is suitable for forecasting data with no clear trend or seasonal pattern. It requires a single parameter, called *alpha* (*a*), also called the smoothing factor or smoothing coefficient.

This parameter controls the rate at which the influence of the observations at prior time steps decay exponentially. Alpha is often set to a value between 0 and 1. Large values mean that the model pays attention mainly to the most recent past observations, whereas smaller values mean more of the history is taken into account when making a prediction.

A value close to 1 indicates fast learning (that is, only the most recent values influence the forecasts), whereas a value close to 0 indicates slow learning (past observations have a large influence on forecasts).

1. **Double Exponential Smoothing (SES)/ Holt’s Linear Model:**

Double Exponential Smoothing is an extension to Exponential Smoothing that explicitly adds support for trends in the univariate time series.

In addition to the *alpha* parameter for controlling smoothing factor for the level, an additional smoothing factor is added to control the decay of the influence of the change in trend called *beta* (*b*).

The method supports trends that change in different ways: an additive and a multiplicative, depending on whether the trend is linear or exponential respectively.

Double Exponential Smoothing with an additive trend is classically referred to as Holt’s linear trend model, named for the developer of the method Charles Holt.

* Additive Trend**:** Double Exponential Smoothing with a linear trend.
* Multiplicative Trend: Double Exponential Smoothing with an exponential trend.

For longer range (multi-step) forecasts, the trend may continue on unrealistically. As such, it can be useful to dampen the trend over time. Dampening means reducing the size of the trend over future time steps down to a straight line (no trend).

1. **Triple Exponential Smoothing/ Holt-Winter Model:**

Triple Exponential Smoothing is an extension of Exponential Smoothing that explicitly adds support for seasonality to the univariate time series. This method is sometimes called Holt-Winters Exponential Smoothing, named for two contributors to the method: Charles Holt and Peter Winters. In addition to the alpha and beta smoothing factors, a new parameter is added called *gamma*(g) that controls the influence on the seasonal component. As with the trend, the seasonality may be modelled as either an additive or multiplicative process for a linear or exponential change in the seasonality.

* Additive Seasonality: Triple Exponential Smoothing with a linear seasonality.
* Multiplicative Seasonality**:** Triple Exponential Smoothing with an exponential seasonality.

# **Regression Models:**

Time series regression is a statistical method for predicting a future response based on the response history (known as autoregressive dynamics) and the transfer of dynamics from relevant predictors. Time series regression can help us understand and predict the behaviour of dynamic systems from experimental or observational data. Time series regression is commonly used for modelling and forecasting of economic, financial, and biological systems. There are two models of Timeseries regression we will be exploring in this project:

1. **Autoregressive Integrated Moving Average (ARIMA):**

ARIMA is an acronym that stands for Autoregressive Integrated Moving Average. It is a generalization of the simpler Autoregressive Moving Average and adds the notion of integration. This acronym is descriptive, capturing the key aspects of the model itself. Briefly, they are:

* **AR**: *Autoregression*. A model that uses the dependent relationship between an observation and some number of lagged observations.
* **I**: *Integrated*. The use of differencing of raw observations (e.g. subtracting an observation from an observation at the previous time step) in order to make the time series stationary.
* **MA**: *Moving Average*. A model that uses the dependency between an observation and a residual error from a moving average model applied to lagged observations.

This method has three variables to account:

p - Periods to lag for e.g.: (if P= 3 then we will use the three previous periods of our time series in the autoregressive portion of the calculation) P helps adjust the line that is being fitted to forecast the series. Purely autoregressive models resemble a linear regression where the predictive variables are P number of previous periods

d - In an ARIMA model we transform a time series into stationary one (series without trend or seasonality) using differencing. D refers to the number of differencing transformations required by the time series to get stationary.

Stationary time series is when the mean and variance are constant over time. It is easier to predict when the series is stationary. The best way to determine whether or not the series is sufficiently differenced is to plot the differenced series and check to see if there is a constant mean and variance.

q - This variable denotes the lag of the error component, where error component is a part of the time series not explained by trend or seasonality.

1. **Seasonal Autoregressive Integrated Moving Average (SARIMA):**

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component. It adds three new hyperparameters to specify the autoregression (AR), differencing (I) and moving average (MA) for the seasonal component of the series, as well as an additional parameter for the period of the seasonality. Configuring a SARIMA requires selecting hyperparameters for both the trend and seasonal elements of the series.

## Trend Elements: There are three trend elements that require configuration. They are the same as the ARIMA model; specifically:

* p: Trend autoregression order.
* d: Trend difference order.
* q: Trend moving average order.

## Seasonal Elements: There are four seasonal elements that are not part of ARIMA:

* P**:** Seasonal autoregressive order.
* D**:** Seasonal difference order.
* Q**:** Seasonal moving average order.
* m: The number of time steps for a single seasonal period**.**

# **Evaluation Parameters:**

1. **Mean Absolute Percentage Error (MAPE)**

In statistics, the MAPE is the Mean Absolute Percentage Error which is the useful measure to forecast accuracy of the model, it will give us the percentage value, which represents what percentage of the actual value is the error. It is represented as:



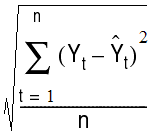
Where:

* $ \{x_i\} $ is the actual observations time series
* $ \{\hat x_i\} $ is the estimated or forecasted time series
* $ N $ is the number of non-missing data points

1. **Root Mean Squared Error (RMSE)**

Root mean squared error is an absolute error measure that squares the deviations to keep the positive and negative deviations from cancelling one another out. This measure also tends to exaggerate large errors, which can help when comparing methods.

The formula for calculating RMSE:



where Yt is the actual value of a point for a given time period t, n is the total number of fitted points, and Yt\_cap is the fitted forecast value for the time period t.

Chapter 3: Exploratory Data Analysis

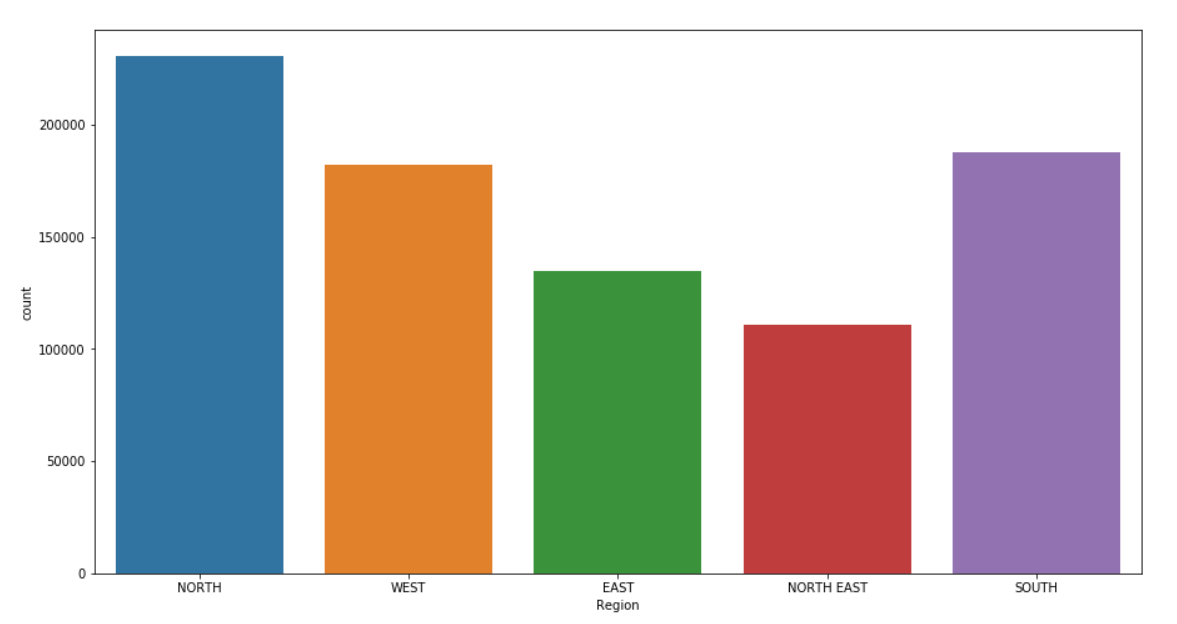
# **Dataset:**

In the dataset provided:

|  |  |
| --- | --- |
| Columns | Description |
| Date | The date on which particular observation has been taken |
| Center | Name of the city where observation is based |
| Region | Center Converges into Five regions |
| Commodity | Name of the product |
| Price per Kg | In INR, the per kilogram price of the commodity |
| Country | Name of the country |

# **Visualization:**

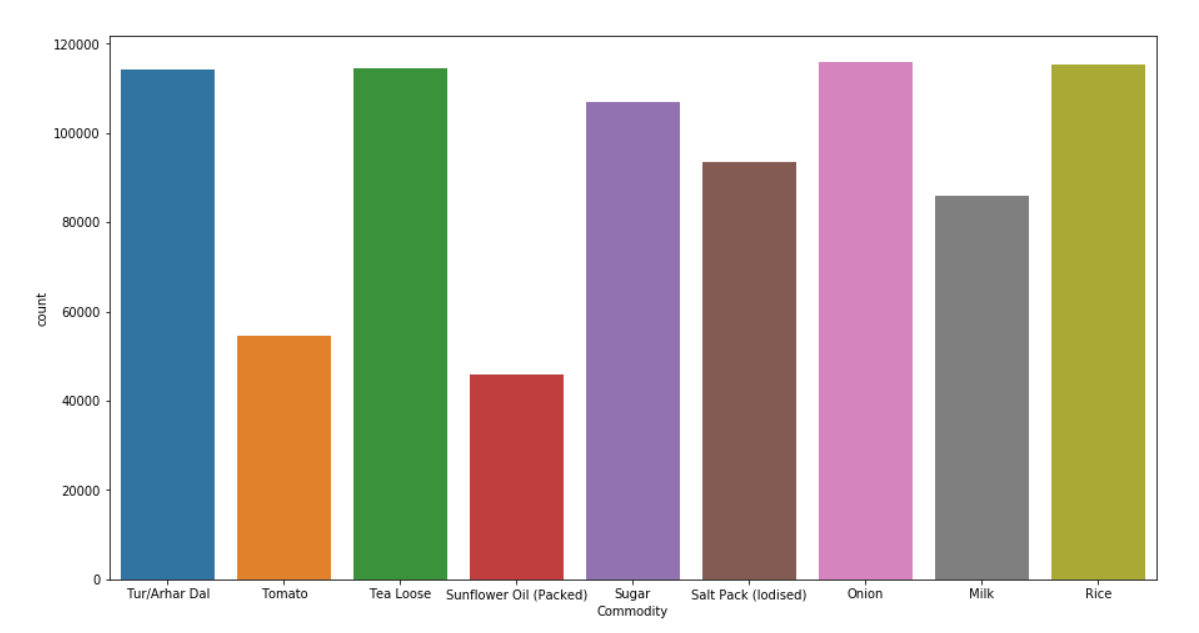
The count of commodities in each region is:



Observations:

* The number of observations taken for north region is higher in comparison to other regions and North-east region has lowest number of count

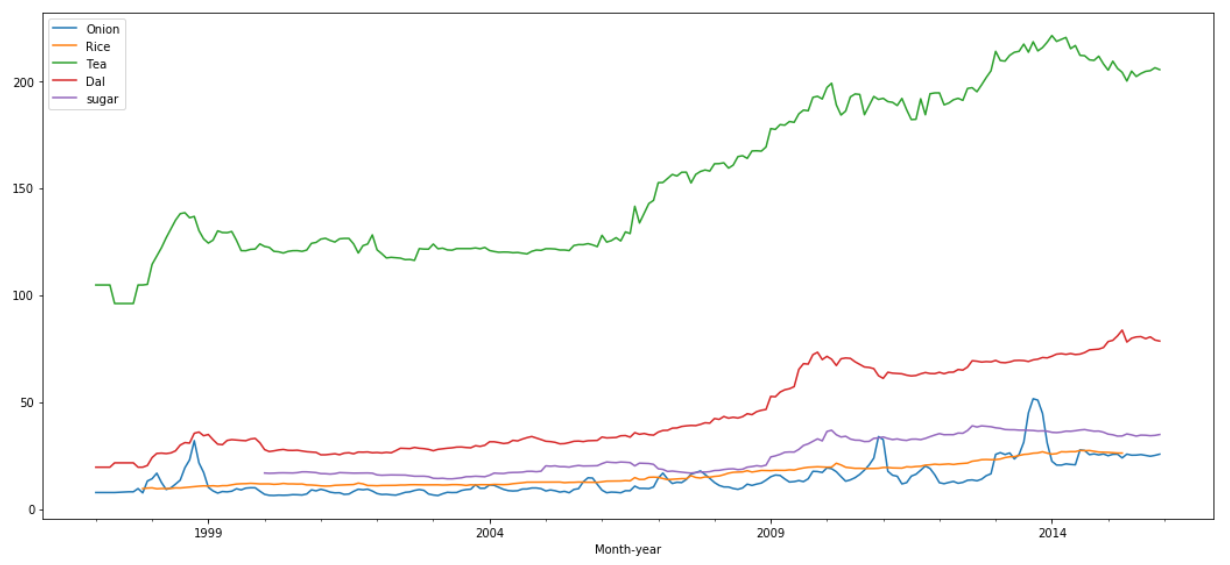
Number of observations taken for each commodity:



Observations:

* Out of 9 commodities, the five most important commodities based on number of observations are chosen and they are listed in order as below:
  + Tur/Arhar Dal
  + Tea Loose
  + Sugar
  + Onion
  + Rice
* As the number of counts are high for these commodities, we will get a better representation of the trend with minimum number of missing values

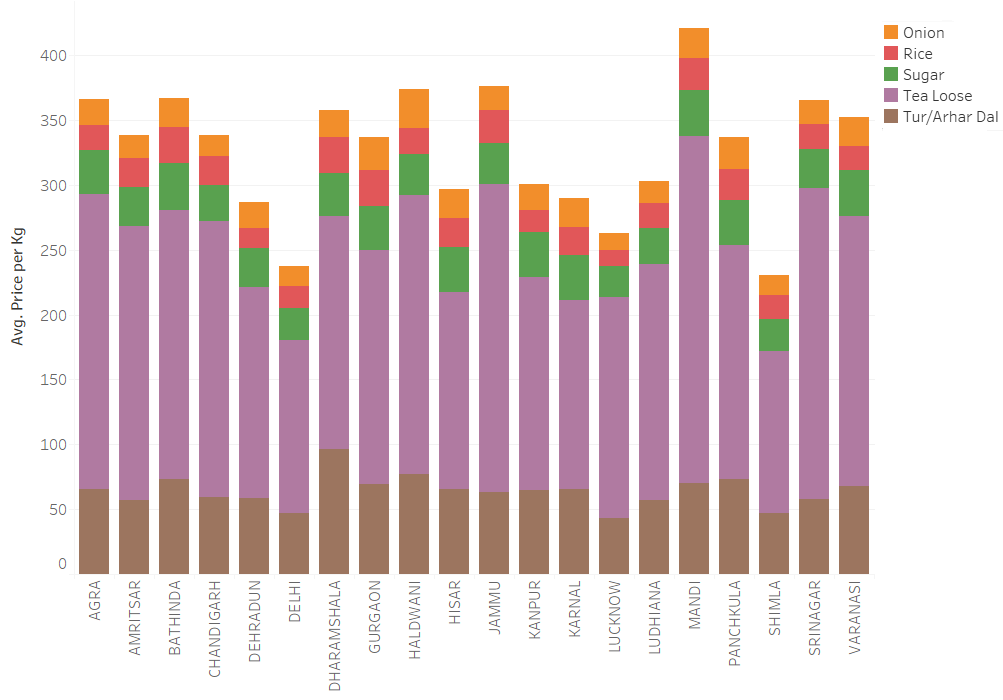
Month-wise behaviour of these important commodities in North region:



Observations:

* There are fluctuations in the price of all the important commodities
* No visible seasonality is present
* The price is increasing over the years for Dal, Rice and Sugar, there are few fluctuations in the timeseries but no big fluctuations that could affect the prediction values.
* In Tea, there is a positive sudden increase in price between 2005 and 2007 but overall the series is showing the trend.
* There are few big fluctuations present in Onion data due to the unprecedented nature of the commodity, there could be multiple reason for the fluctuations like poor weather condition, political factor, supply chain, etc from which the price of the Onion get affected and shows such fluctuation in the timeseries

The centre-wise analysis of important commodities in North region:



Observations:

* The overall summation of averages for all the commodities in each center represent the overall price fluctuation in each center
* Mandi in north region is the most expensive center for important commodities
* Shimla in north region is the cheapest center for important commodities
* There is a fluctuation of around 200 rs in terms of overall price of 5 important commodities
* Onion and Sugar seems to have a low fluctuation among centers representing that the increase or decrease in price for all the centers is mostly common
* Tea is the highly fluctuating commodity in term of centers, means you can get better value for money by moving to different center.

# **Feature Engineering:**

Instead of building forecasting models for all regions and all important commodities, we have built five models for five most important commodities for the chosen Region i.e. Northern region. The granularity of the data given is day-wise, so we have converted it into a month-wise, using the average of all the dates present in any particular month. Our objective is to analyse the month-wise trend for all such commodities and predict for next twelve months. We used train-test split as a data split(2yrs data were used for testing and rest of 14yrs data were used for training) to determine the best model for particular commodity as a part of model evaluation.

The flow-chart for the modelling will be as follows:

For the purpose of learning from the timeseries, we have built five models against five important commodities in North region. We developed five timeseries models under decomposition and regression techniques which we will use to predict and find out the best model technique for each of the top five commodities.

Chapter 4 : Modelling

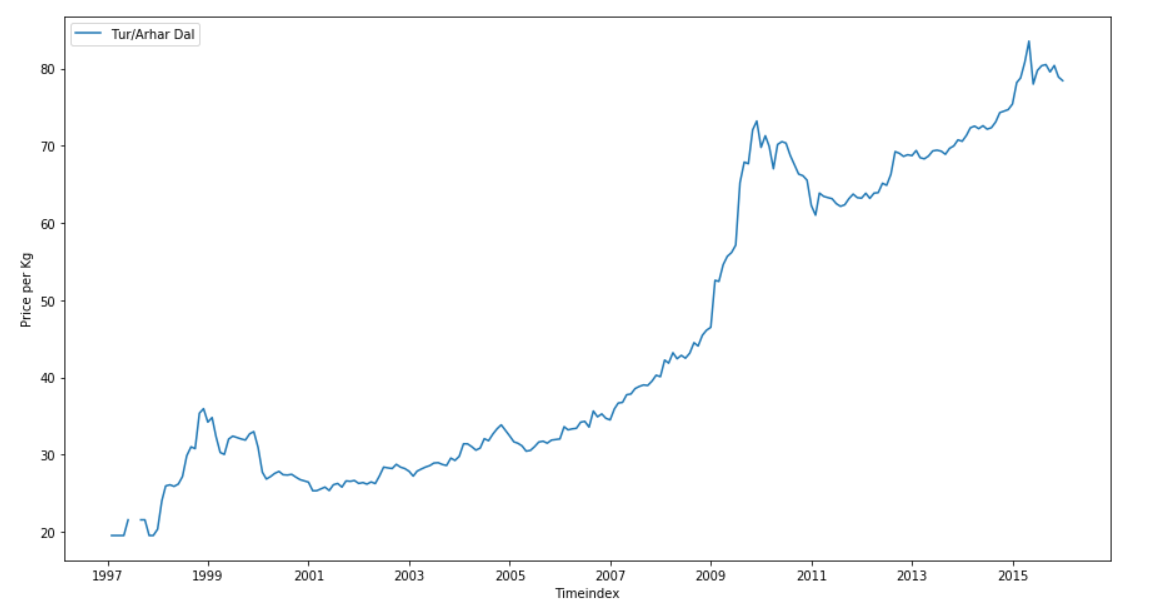
In order to understand the working of the Decomposition and Regression Modelling techniques in time series, we are going to build models using all five commodities for Northern region. For the purpose of this report, we will show the models building using the instance of

Region - North

Commodity - Tur/Arhar Dal

# **Visualization:**

The timeseries of the Tur/Arhar Dal for the north region will look like:



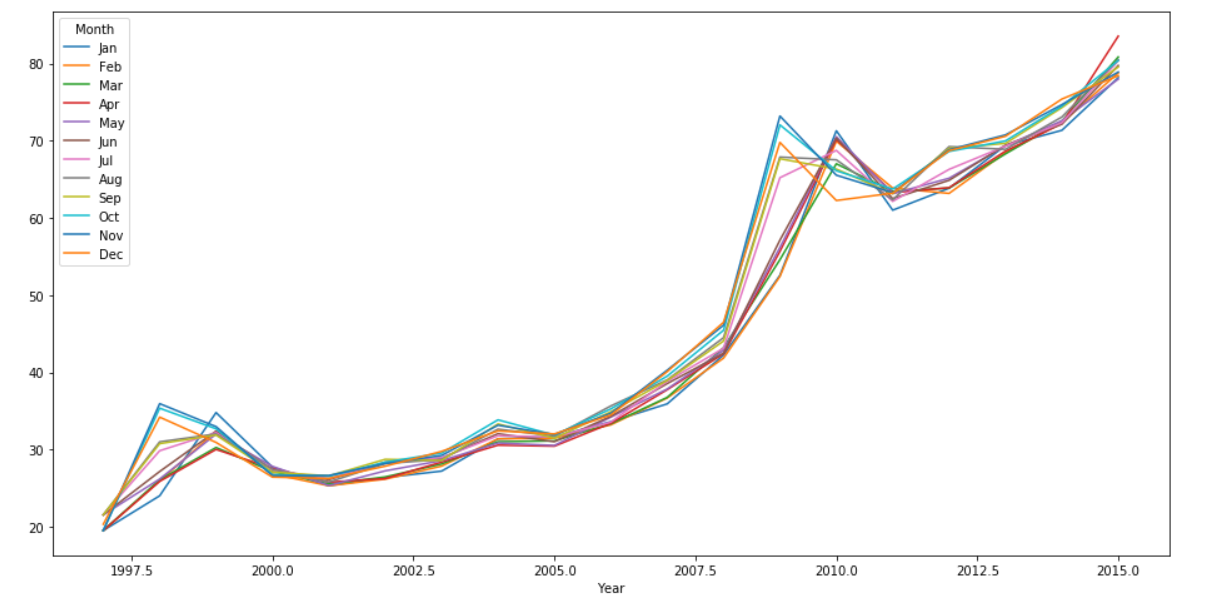
Observation:

* The trend is linear with two big fluctuations
* Price of the commodity is growing with time
* It is clear that the null values are present in the timeseries of the commodity especially 1997 -1999
* There is no visible seasonality present in the timeseries

To fill the missing value, we need to check the trend and seasonality and then we will able to choose the appropriate data imputation method to treat missing values. By creating the columns of month and year and with the use of pivot table, we checked seasonality plot.

**Seasonality Plots:**

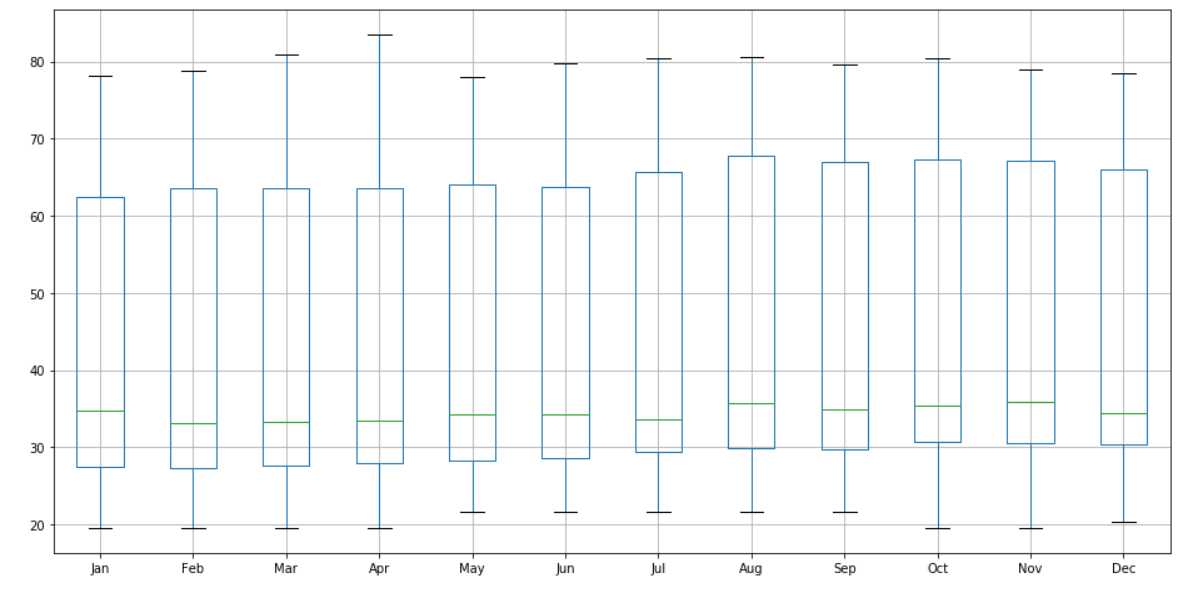
1. Line Plot



Observations:

* There is no particular month which is consistently showing highest or lowest price in the commodity over the years.
* Between 2000 and 2007, the rates are most consistent among months
* For the two big fluctuations in the timeseries, the most affected months are the last three months of an year

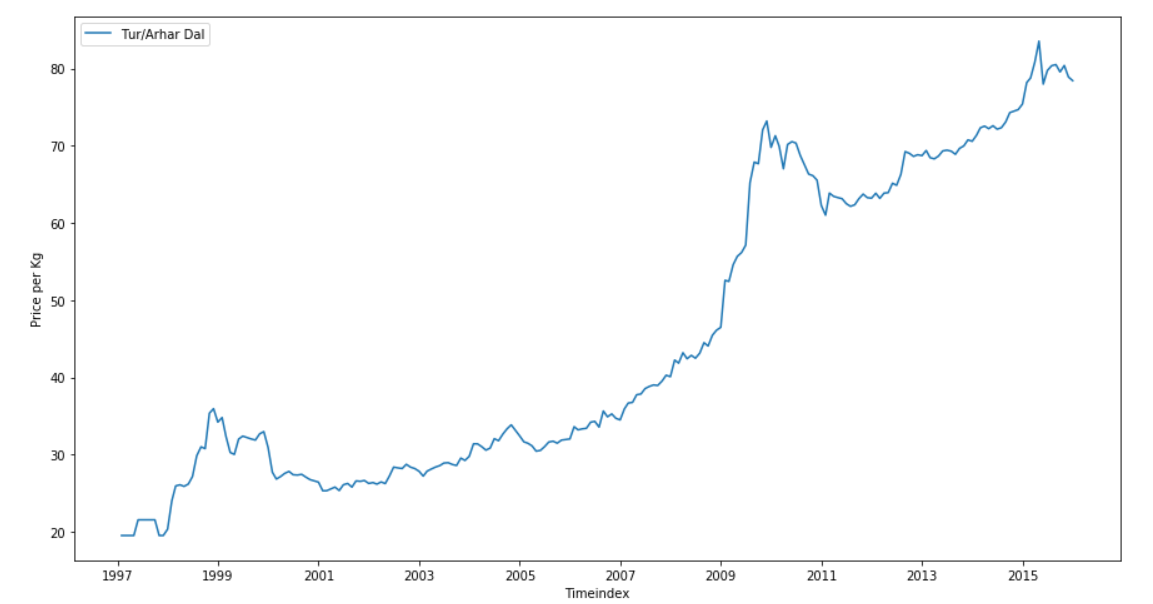
1. Box-Plot



Observations:

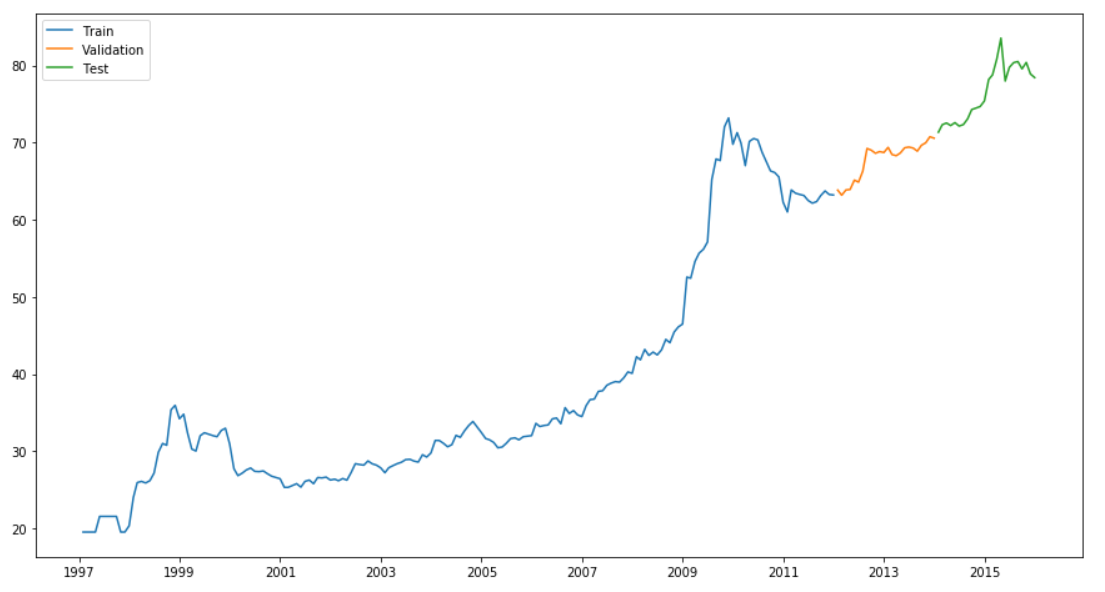
* The fluctuations among all the months look similar
* The median rate of dal price is towards the low side of the box plot represents that the price was consistently low for certain time and then sudden increase happened
* There is no outlier presence in the data

From seasonality plot, it can be seen that the timeseries of Tur/Arhar dal for north region has the trend in it but no seasonality. As we have discussed in Chapter 2, we will use the linear interpolation method of timeseries as there is no seasonality but trend. After applying the linear interpolation, the timeseries will be:



# **Train-Validation-Test Split:**

Now we will split the timeseries data into train, validation and test by keeping last 24 months for test and validation and others in the train set. We will hyper tune our model using validation dataset and evaluate using test dataset.

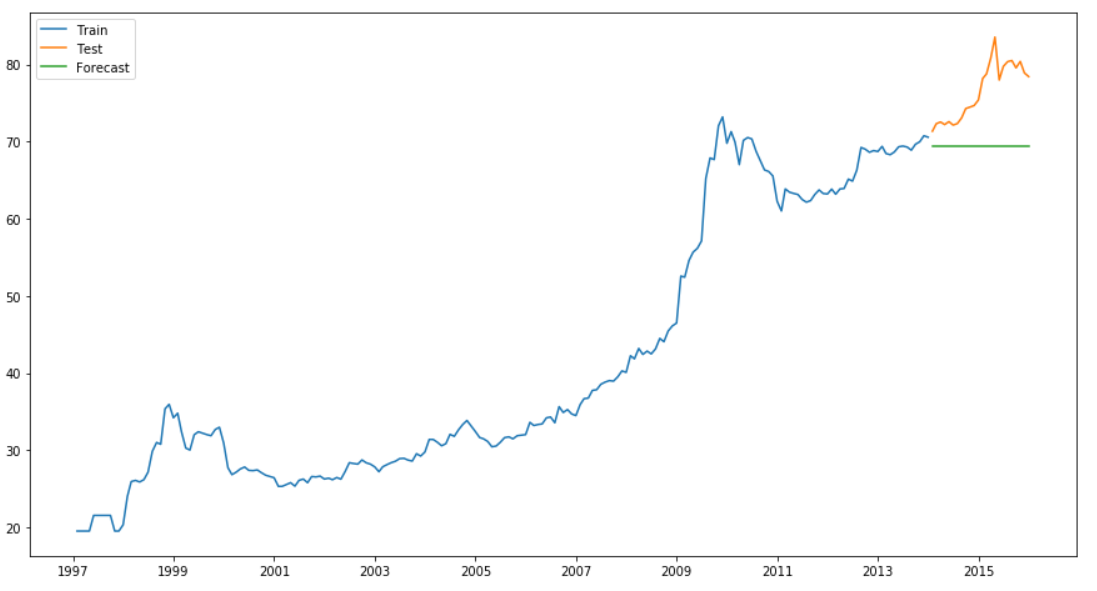


# **Simple Exponential Model:**

Summary:

|  |  |
| --- | --- |
| Parameters: | endog (*array-like*) – Time series |
| Returns: | results |
| Return type: | SimpleExpSmoothing class |

In this model, we will predict the test which will look like:



Learnings:

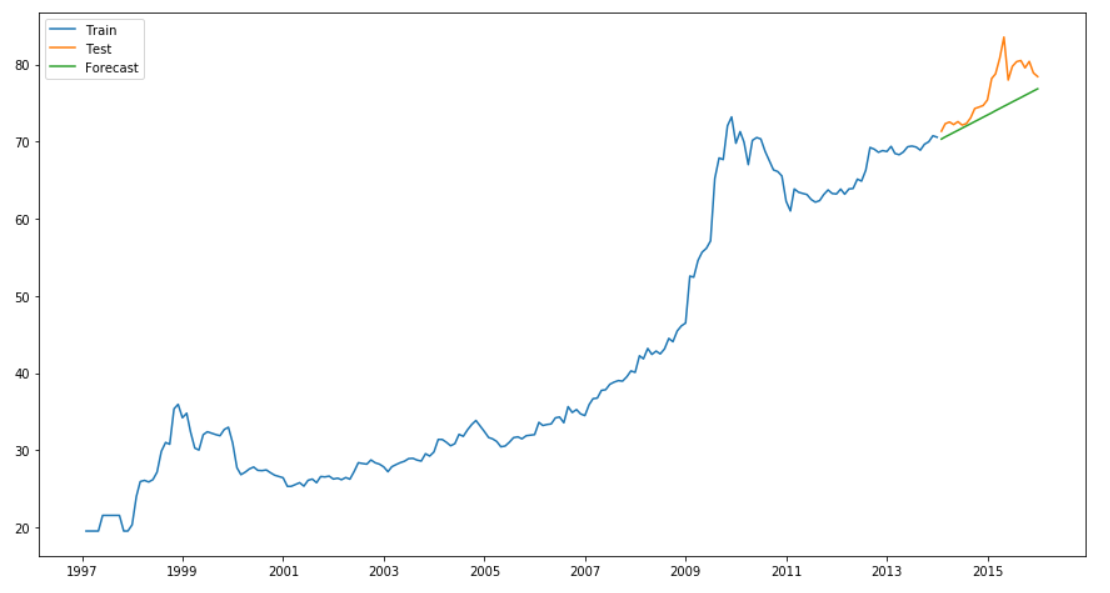
* The MAPE value for the test is 9.11 %
* The best smoothing level is 0.15 on validation dataset
* Model works with the last observed data and give us a line
* It does not take seasonality and trend into consideration as a result we see a straight line in our forecasting prediction.

# **Holt’s Linear Model:**

Summary:

|  |  |
| --- | --- |
| Parameters: | * endog (*array-like*) – Time series * exponential ([*bool*](https://docs.python.org/3/library/functions.html#bool)*, optional*) – Type of trend component. * damped ([*bool*](https://docs.python.org/3/library/functions.html#bool)*, optional*) – Should the trend component be damped. |
| Returns: | results |
| Return type: | Holt class |

Under this model, the prediction for the commodity will look like:



Learnings:

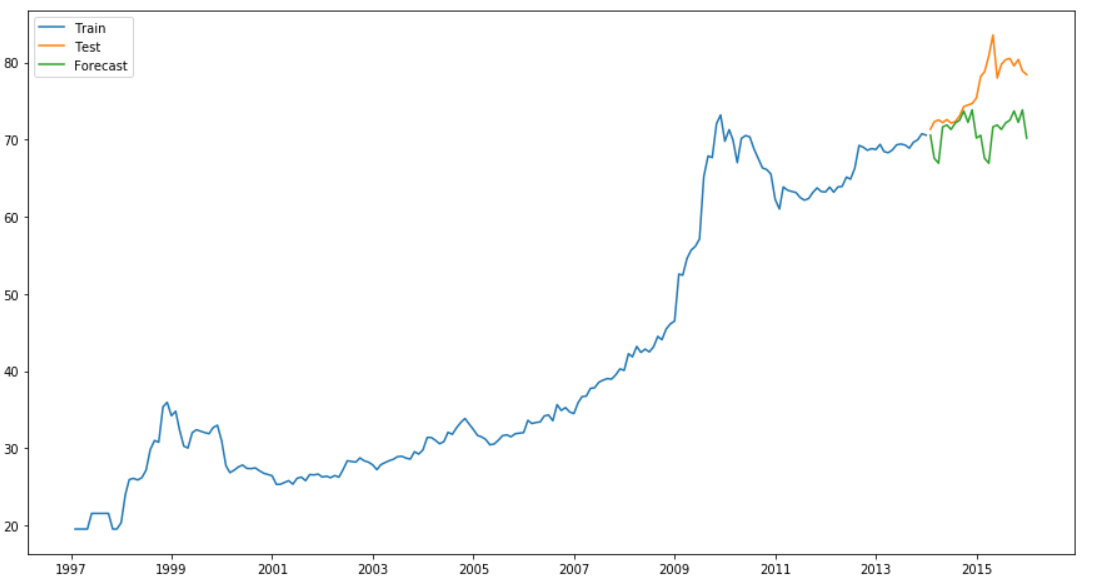
* The MAPE value for the test is 3.63 %
* The best smoothing level is 0.25 and smoothing slope is 0.45 as checked on validation dataset
* The forecasts generated by Holt’s linear method display a constant trend (increasing or decreasing) indefinitely into the future
* For the timeseries without seasonality, this model will give the best forecast

# **Holt-Winter Model:**

Summary:

|  |  |  |
| --- | --- | --- |
| Parameters: | | * endog (*array-like*) – Time series * trend (*{"add", "mul", "additive", "multiplicative", None}, optional*) – Type of trend component. * damped ([*bool*](https://docs.python.org/3/library/functions.html#bool)*, optional*) – Should the trend component be damped. * seasonal (*{"add", "mul", "additive", "multiplicative", None}, optional*) – Type of seasonal component. * seasonal\_periods ([*int*](https://docs.python.org/3/library/functions.html#int)*, optional*) – The number of seasons to consider for the holt winters. |
| Returns: | results | |
| Return type: | ExponentialSmoothing class | |

The test forecast for this model will be:



Learnings:

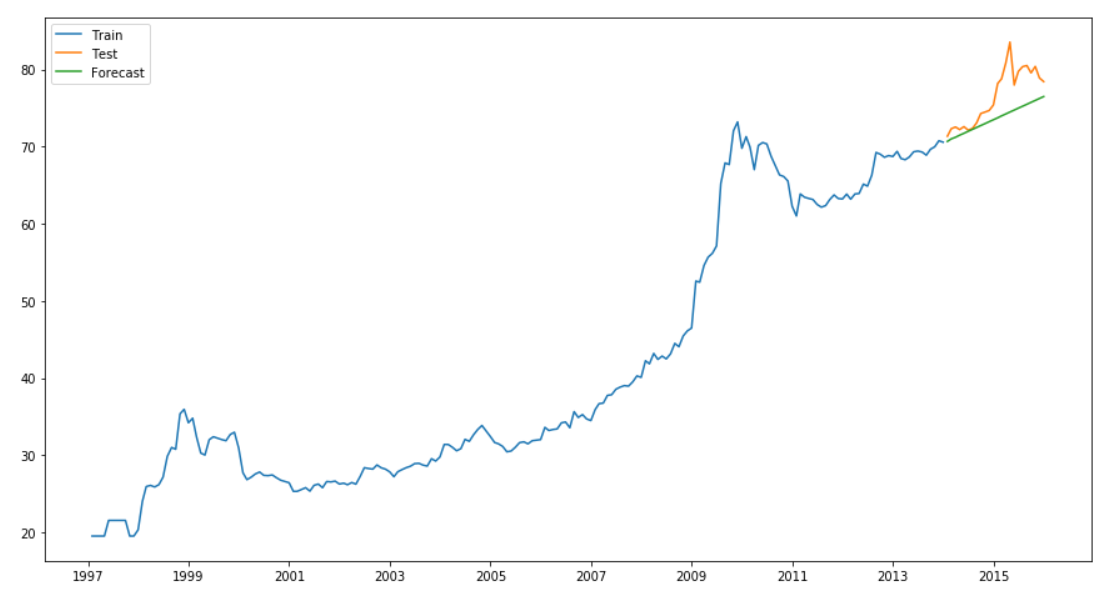
* The MAPE value for test is 6.64%
* The best smoothing level is 0.85, smoothing slope is 0.35 and smoothing seasonal is 0.90, as checked on validation dataset
* This model does consider trend and seasonality in the train data
* Even though, we had no seasonality in the Dal commodity, the model was able to find the pattern and repeat it for the test data

# **ARIMA Model:**

Summary

|  |  |
| --- | --- |
| Parameters: | * endog (*array-like*) – The endogenous variable. * order (*iterable*) – The (p,d,q) order of the model for the number of AR parameters, differences, and MA parameters to use. * exog (*array-like, optional*) – An optional array of exogenous variables. This should *not* include a constant or trend. You can specify this in the *fit* method. * dates (*array-like of datetime, optional*) – An array-like object of datetime objects. If a pandas object is given for endog or exog, it is assumed to have a DateIndex. * freq ([*str*](https://docs.python.org/3/library/stdtypes.html#str)*, optional*) – The frequency of the time-series. A Pandas offset or ‘B’, ‘D’, ‘W’, ‘M’, ‘A’, or ‘Q’. This is optional if dates are given. |

This model will predict test as:



Learnings:

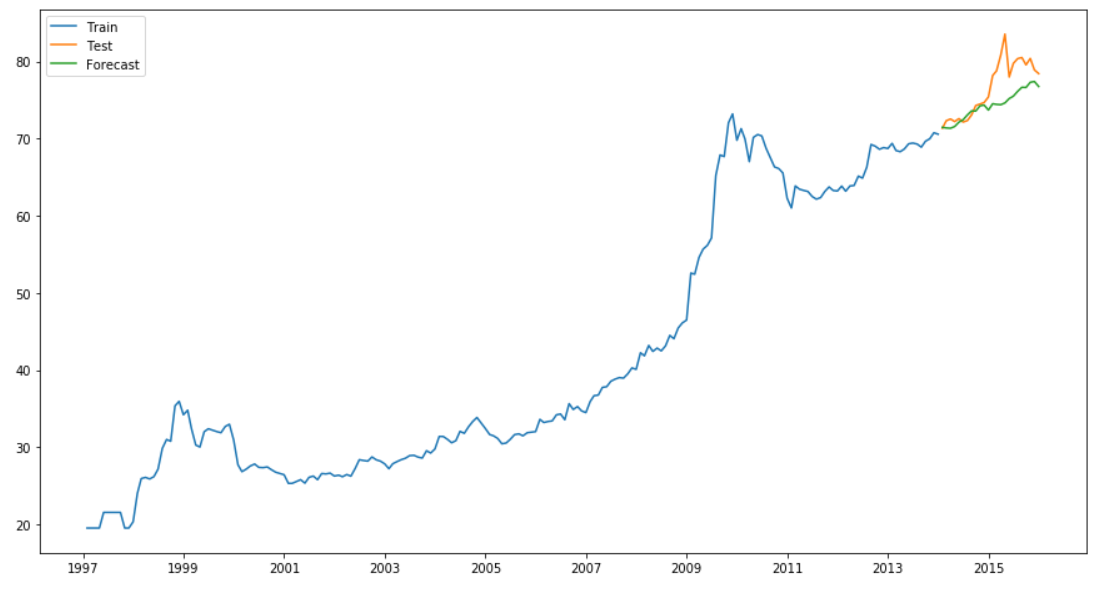
* The MAPE value for the test is 3.59%
* ARIMA has the auto-regressive term for prediction in slope and moving average to smoothen the series
* The value of p,d,q can be calculated by checking the combination with best AIC figure
* The best values are p=1, d=1 and q=1 for this commodity

# **SARIMA Model:**

Summary

|  |  |
| --- | --- |
| Important  Parameters: | * endog (*array\_like*) – The observed time-series process yy * exog (*array\_like, optional*) – Array of exogenous regressors, shaped nobs x k. * order (*iterable or iterable of iterables, optional*) – The (p,d,q) order AR, deference, MA terms * seasonal\_order (*iterable, optional*) – The (P,D,Q,s) order of the seasonal component of the model for the AR parameters, differences, MA parameters, and periodicity * enforce\_stationarity (*boolean, optional*) – Whether or not to transform the AR parameters to enforce stationarity in the autoregressive component of the model. Default is True. * enforce\_invertibility (*boolean, optional*) – Whether or not to transform the MA parameters to enforce invertibility in the moving average component of the model. Default is True |

Under this model, the prediction will look like:



Learnings:

* The MAPE value for the test is 2.92%
* This model has all the qualities of ARIMA model, in addition to that it has AR, MA and difference term with respect to seasonality also
* It works with AR and MA terms for both seasonality and trend
* The best value for p,d,q and P,D,Q has been calculated by checking the best AIC figure
* The best values are – (1,1,1) and (0,1,1,12) for order and seasonal order respectively.

Chapter 5 : Models Comparison

As we have discussed in Chapter 2, there are various ways of evaluating the model. In this section, we will compare all those models by taking evaluation measures into account- Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE).  These are the measure of accuracy of a method for constructing time series values in statistics. We trained our models using training dataset, hyper parameter tuning using validation dataset and then evaluated model using test dataset.

# **Model Comparison Matrix:**

Using Mean Absolute Percentage Error:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MAPE | Tur/Arhar Dal | Tea Loose | Sugar | Onion | Rice |
| Simple Exponential | 9.11% | 4.40% | 3.15% | 30.42% | 13.18% |
| Holt’s Linear Model | 3.63% | **2.71%** | **2.66 %** | 206.48% | 8.58% |
| Holt-Winter Model | 6.64% | 4.04% | 3.15 % | 33.39% | 5.73% |
| ARIMA model | 3.59% | 7.58% | 5.61% | 43.99% | **3.54%** |
| SARIMA model | **2.92%** | 8.03% | 7.56% | **18.53%** | 4.76% |

Using Root Mean Square Error:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RMSE | Tur/Arhar Dal | Tea Loose | Sugar | Onion | Rice |
| Simple Exponential | 7.97 | 10.39 | 1.35 | 7.45 | 3.54 |
| Holt’s Linear Model | 3.55 | **6.50** | **1.10** | 51.79 | 2.65 |
| Holt-Winter Model | 6.59 | 9.20 | 1.23 | 8.21 | 1.67 |
| ARIMA model | 3.62 | 18.15 | 2.47 | 11.4 | **1.08** |
| SARIMA model | **3.18** | 19.48 | 3.89 | **5.40** | 1.36 |

Observations:

* For two of the commodities, the best model is SARIMA but as we have already discussed there is no seasonality attached to the dataset. The better results we are getting because of the fluctuation attached with the dataset
* Other commodities are giving better result with linear models ,i.e., Holt and ARIMA model which works with Trend

Chapter 6 : Commodities Forecast

With respect to the best model, we will now forecast for next twelve months in each commodity

# **Forecast:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2016 | Tur/Arhar Dal | Tea Loose | Sugar | Onion | 2015-16 | Rice |
| Jan | 79.40 | 205.44 | 34.92 | 24.73 | **May** | 26.18 |
| Feb | 79.41 | 206.59 | 35.18 | 22.78 | **June** | 26.25 |
| March | 79.49 | 207.76 | 35.44 | 21.42 | **July** | 26.33 |
| April | 79.82 | 208.92 | 35.69 | 20.21 | **Aug** | 26.41 |
| May | 80.06 | 210.08 | 35.96 | 18.98 | **Sep** | 26.49 |
| June | 80.40 | 211.24 | 36.21 | 18.72 | **Oct** | 26.57 |
| July | 81.01 | 212.39 | 36.47 | 20.12 | **Nov** | 26.65 |
| Aug | 81.53 | 213.56 | 36.73 | 21.59 | **Dec** | 26.73 |
| Sep | 81.52 | 214.72 | 36.99 | 22.11 | **Jan** | 26.81 |
| Oct | 82.18 | 215.88 | 37.25 | 22.81 | **Feb** | 26.89 |
| Nov | 82.19 | 217.04 | 37.51 | 21.76 | **March** | 26.97 |
| Dec | 81.63 | 218.19 | 37.76 | 20.62 | **April** | 27.05 |

Observations:

* The price given for each commodity is the consumer price, i.e., the price on which retailers will sell it to the consumers. We are making this model for the retail store to give them better return on investment on buying commodities
* The price of Tur/Arhar dal has fluctuations so our prediction is based on the ups and downs in the curve. Overall, price will increase.
* Tea price is going to increase with time, the rate of increase in price is almost a rupee per month
* Sugar price is going to increase slowly with lower rate
* Onion as we have taken SARIMA model and the fluctuations are very high in this commodity. The prediction is showing those fluctuations but overall, price is decreasing.
* The linear trend slope is very small in Rice commodity

Chapter 7 : Key Learnings and Actionable Insights

# **Key Learning:**

* A time series is a sequence of numerical data points in successive order. In investing, a time series tracks the movement of the chosen data points, such as a commodities price, over a specified period of time with data points recorded at regular intervals.
* Depending upon the data distribution and missing values followed by existence of Trend, seasonality, unexplained errors and cyclicity, the choice of models will be different.
* In this project, we choose Holt’s linear model for 4 out of 5 commodities, the main reason that this model is performing better than others is the presence of trend factor and no seasonality in those time series.

# **Actionable Insight:**

**For Retail Stores:**

* The price of Tur/Arhar Dal is showing the increasing trend with short dip at the end of the year. So, it is better to stock this commodity and sell within the year when price is high.
* Tea Loose price is expected to increase in next 12 months. The rate of this increasing trend is almost 1 Rs per month per Kg. Tea Loose has a good shelf life, so you can warehouse this commodity and sell in the higher rate next month.
* Sugar price has not big trend in its historical data, the increase is also pretty much linear with the small slope. The price is expected to increase but not at higher rate. The growth in sugar business will also be small but the good news is that it does not have any downward trend. So if you want a consistent growth, Sugar is the right business for you
* Onion has a huge fluctuation in its timeseries, even though the series is stationary but fluctuations are not seasonal, we can call them cyclic but it is impossible to predict exact happening of those fluctuations in timeseries problem. The fluctuations even have 200% increment in the value of Onion. Also, onion is a highly perishable commodity, so you cannot warehouse it for long time. The prediction we are making is based on SARIMA model, i.e., we are taking seasonality into account which is actually the fluctuations. You can take risk with this commodity as returns will be high if fluctuations are positive.
* Rice has the smallest incremental slope in all the commodities, the rate of rice is expected to grow by two rupees by the end of the year. The business will be slow moving but the demand of rice will always be there in the market. So you can start rice business, if you are already established player as you will not expected to received big returns in term of price inflation.

Chapter 8 : Future Work

* The granularity in the dataset was day-wise but due to major missing values we smoothened it into a month-wise series, it will be interested to check the day-wise price fluctuation in the time series. We will also be able to check the affect on price during political campaigns, holidays, etc.
* We will also make use of Facebooks ‘Prophet’ package to build a better model. Prophet is optimized for the business forecast tasks, which typically have any of the following characteristics:
  + hourly, daily, or weekly observations with at least a few months (preferably a year) of history
  + strong multiple “human-scale” seasonalities: day of week and time of year
  + important holidays that occur at irregular intervals that are known in advance (e.g. the Super Bowl)
  + a reasonable number of missing observations or large outliers
  + historical trend changes, for instance due to product launches or logging changes
  + trends that are non-linear growth curves, where a trend hits a natural limit or saturates

Chapter 9 : Bibliography

* Business Statistics- A First Course By David M. Levine
* Introductory Time Series – By Jason Browniee
* <https://data.gov.in/> - Commodity Data
* <https://machinelearningmastery.com/> - Seasonality and Trend Analysis
* <https://www.datascience.com> – Time Series Introduction
* <https://www.statisticssolutions.com/> - Time Series Models
* <https://machinelearningmastery.com/> - Evaluation parameters
* <https://research.fb.com/prophet-forecasting-at-scale/-> Forecasting Method

ANNEXURE

The codes are uploaded on –

<https://github.com/GauravGarhewal/Capstone-Project>