# Average Monthly Usage (AMU) prediction & Inventory Modelling using Machine Learning


## Submitted By

Prasanna Chandran

## Research Supervisor

Mr. Suryanarayana Reddy Yarrabothula

## Great Lakes Institute of Management

**Contents**

## *Capstone Project - Process Flow*

# 1. Introduction

ABC Electricals is a multinational corporation and a Fortune 500 company. It is specialized in electrical equipment. The company has expanded its logistics operations over the past 10 years. It has consolidated two of its warehouse sites, while at the same time expanding capacity to 50 tons of throughput per day, covering 14,000 product lines. It has over $30B in annual revenue and 140,000 employees across over 100 countries. The company was motivated to make a fundamental disruptive change in the way it manages its inventory. Their executive team believed that there were inventory opportunities based on the local inventory optimization techniques currently in use. For example, recent fluctuations in demand had led to extensive spend in terms of expedites and overtime to meet customer requests which further motivated finding a more radical solution.

With large product portfolio and geographical presence, it is essential for ABC Electricals to accurately forecast the average monthly material usage to maintain optimum inventory level, it also helps to improve the supply chain efficiency and make effective production planning. Other key focus areas for ABC Electricals are to maintain an optimal lead time and safety stock. Safety stock optimization enables companies to achieve savings and increase inventory turns.

# 2. Problem Statement

For ABC electricals, the system - ITB is the top manufacturing system that contributes 57.50% of production. ITB, in turn consists of multiple manufacturing plants, that are involved in manufacturing vast electrical product base. In this project, we focus on the top 4 manufacturing plants and top 8 products that are commonly sourced & utilized.

We will analyze the optimum monthly materials usage for the top four plants to derive insights and recommendations for ABC electricals. Then, incorporate time series analysis in forecasting the dependent variable ‘Quantity’ for the next five years. This would help the top 4 plants to plan and smoothly run the manufacturing by avoiding any stock out situation or over sourcing, which in turn results in optimizing the sourcing cost.

## 3. Objective

Forecasting average material consumption pattern by using Time Series engine (TS) based on Machine Learning (ML) and combining the best of TS / ML outputs for each of the SKUs based on past deviations to deliver more accurate forecasts for the production planning team.

- To forecast average monthly consumption using hierarchical time series, based on current average monthly usage details available for the actively sold materials.
- To build a predictive model to understand the variation of average material consumed for actively sold products by top 4 manufacturing plants.
- To understand the average monthly usage pattern of the manufacturing plants and suggest a safety stock level.
- To predict the optimum stock coverage days based on the average monthly usage and Materials Requirement Planning Stage.

## 4. Scope

With the globalization of manufacturing operations, having a global procurement network that can support and react to the supply chain needs is important. With shorter product life cycles and changing market demands, selecting a strategic supplier that provides manufacturing locations with consistent global quality and a reliable local service is a challenge. Maintaining an adequate stock and eliminating excess or obsolete inventory is a priority for manufacturing plants. It's impossible to build equipment when the necessary parts are missing. If the parts are not commonly available, the lead time can be weeks or months and result in a Stock-out situation. It could also result in loss of time and money.

Keeping the time constraint and other limiting factors, it's highly unlikely to forecast average monthly usage of all the materials utilized for manufacturing. So, our aim is to focus on the top 4 manufacturing plants and top 8 products commonly used across. Considering the limitations of independent variables available in the data, the dependent variable 'Quantity' sourced, would be the best variable to consider in forecasting the future material requirements. So, time series analysis will be used for forecasting the future quantity to be sourced. Considering the nature of the data, we will be using advanced time series analysis methods incorporating other independent variables like cost and discount to get an accurate forecast rather than other machine learning models

## 5. Data Report

ABC electricals has multiple manufacturing plants falling under certain sub-units. ITB is the top manufacturing unit for ABC electricals. The data has been obtained for the ITB unit for a period of 2013 to 2019.

The data set has transactional details of 7 manufacturing plants belonging to the ITB system for the time period of 2013 to 2019.
The data includes 13 columns and 9,58,934 rows, with details of materials sourced & utilized for manufacturing purpose, the relative cost of purchase and discount obtained from the suppliers.

ITB unit consists the below 7 manufacturing units.

| Plant Code | Plant Name |
|---|---|
| P1F | PHIL 1 MFG |
| CN | Shanghai Plant |
| BEF | BANGALORE EoU FACTORY MFG |
| FR3N | FR-Agriers Manuf |
| ID-PT | ID-PT Abc Electric Manufact |
| FRD5 | FR-Horizon Manuf |
| US62 | MX-PACIFICO BREAKERS |

ABC electricals tend to have a numerical naming convention for their material codes. The top materials analyzed are 11373, 1374, 2985, 13997, 11451, 14001, 13999 & 14000. For better understanding the term ‘material’ is used before these material codes. ABC electricals also tend to have a numerical naming convention for their supplier codes. For better understanding the term ‘Supplier is used before these material codes.

AMU indicates Average Monthly Usage by the manufacturing plants.

## 6. Data Preprocessing

Data includes materials sourced at different days of the month by individual plants. To have a sequential time series, the quantity sourced on different days of the same month are aggregated at month level for top products by individual plant level. Subset of top 8 materials are been created and converted to a time series data to understand the time series components like seasonality, trend and irregularity for individual materials.

## ***Missing values***

The data did not have any missing values. At the time of aggregating the quantity by months, there were products in certain plants that were not sourced on specific months, which ideally says there was zero transaction for the product in that month. For those months, the quantity column was imputed with ‘0’ quantity indicating no material usage in that month.

## ***Outliers***

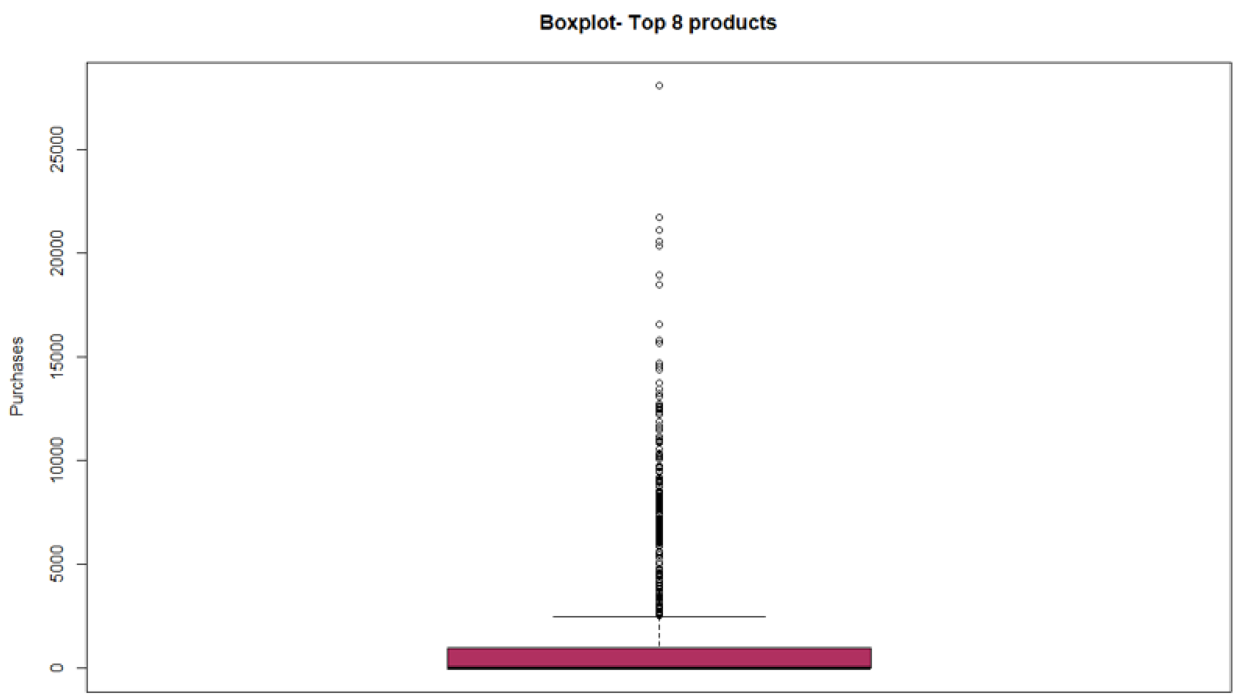


Figure 1.1: Boxplot – Top 8 products.

The boxplot shows there are outlier values. In actual, these are genuine quantities sourced for the manufacturing purpose and it will not be treated as outliers.

It was understood from the ABC electricals, that the sourcing of quantities varies from material to material and based on their significance towards manufacturing, the quantities will be sourced. So certain materials can record a purchase of only one quantity and certain products can be sourced for a quantity of 25000 or higher, based on the requirement.

# 7. Exploratory Data Analysis

## Top manufacturing plants within ITB system

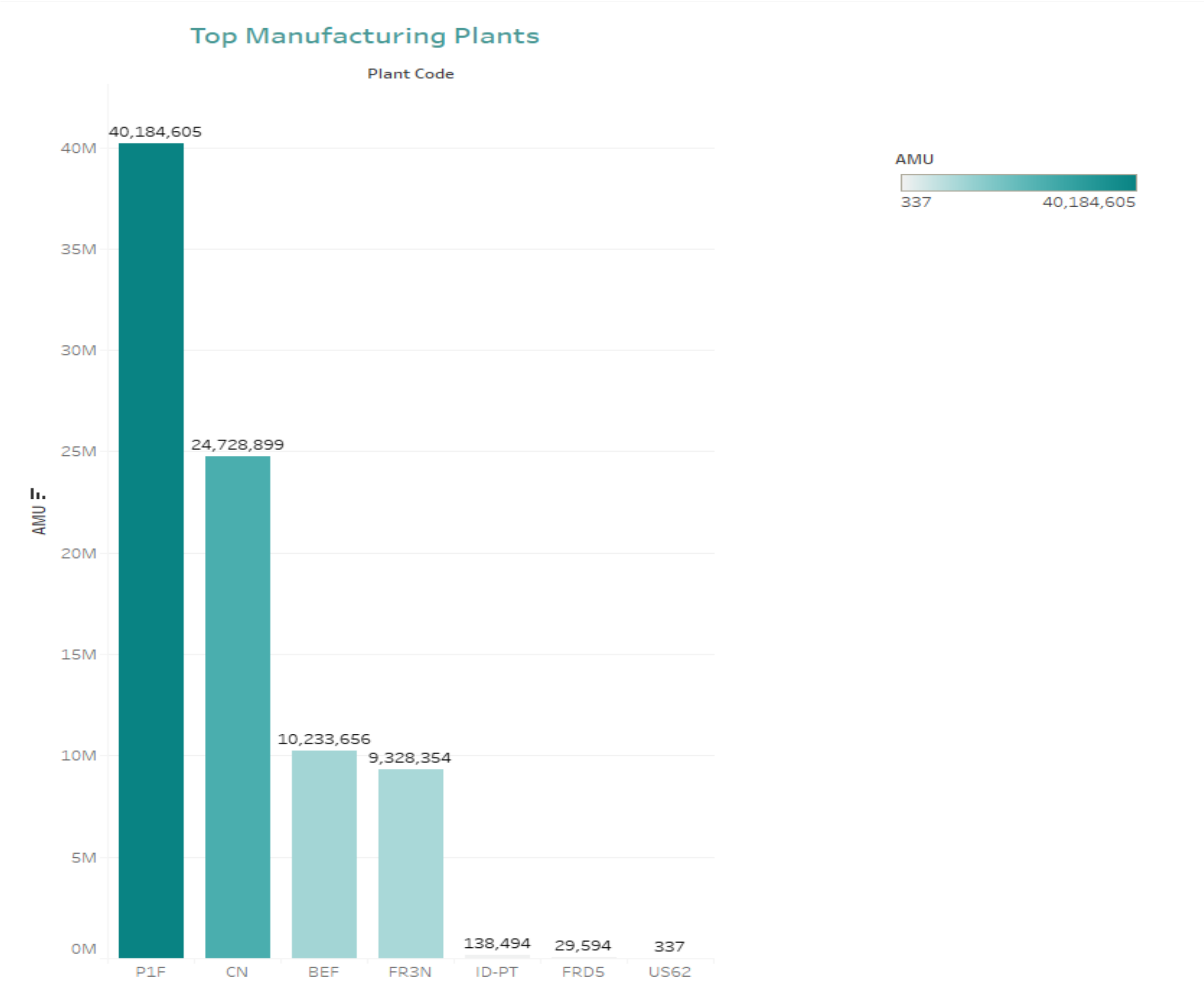


Figure 1.2: Boxplot – Top manufacturing plants

- P1F is the topmost manufacturing plant for ITB system with a massive material sourcing of 40,184,605. Followed by the manufacturing plants CN, BEF, FR3N.

- The plants ID-PT, 29,594 & US62 tend to manufacture in a small scale.

## *Highly sourced materials by ITB for manufacturing purpose*

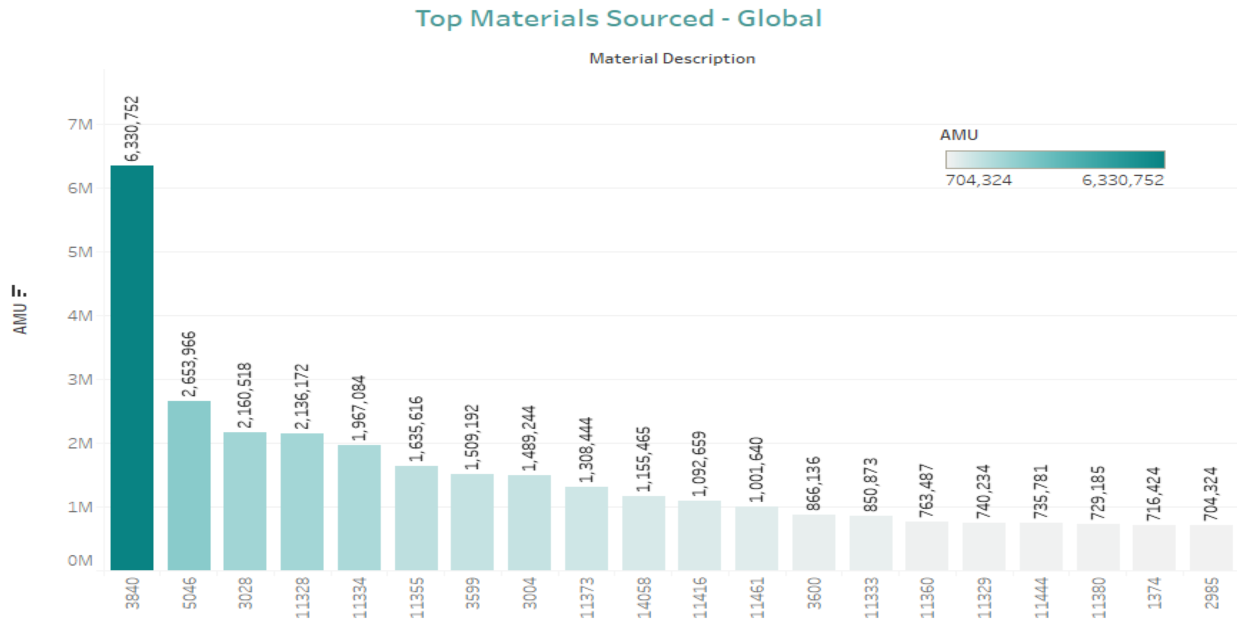


Figure 1.3: Top Materials Sourced - Global

* The material 3840 stands out from other materials sourced, it is enormously sourced and indicates the material is a significant item for the manufacturing process.
* Followed by the products 5046, 3028, 11328, 11334.

## *Top Suppliers*


Figure 1.4: Top Suppliers

* Most of the sourcing is done from the supplier 68, followed by the supplier code 569 & 253.
* This brings up the interesting fact to understand the top discount providers.

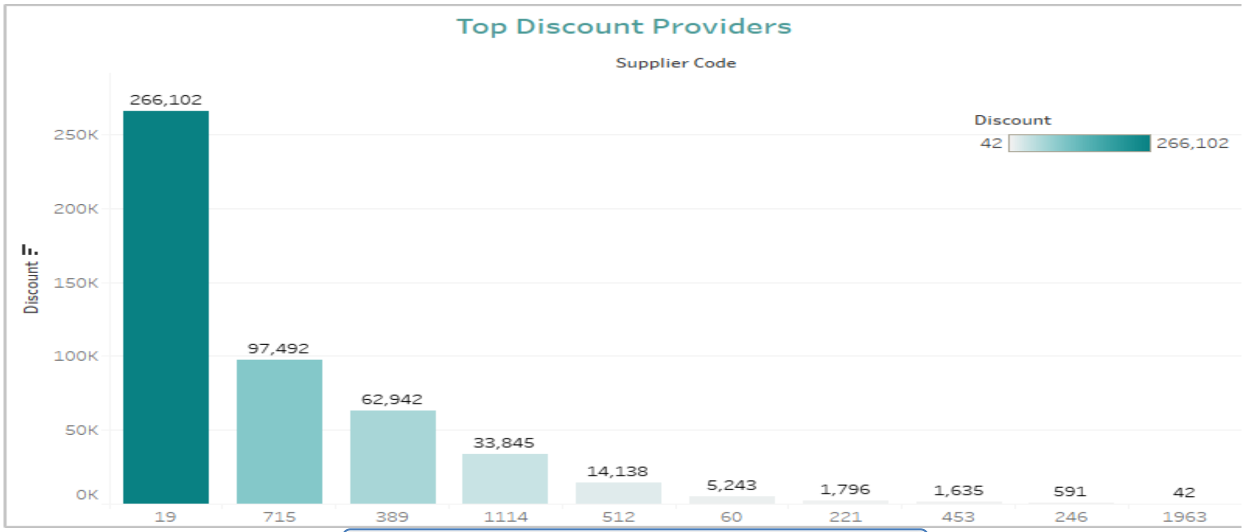


Figure 1.5: Top Discount Providers

Supplier '68' seems to provide very less discount, still ITB system prefers to procure more materials on a consistent basis from the supplier 68. This should be examined further by ABC electricals board to see if considering alternative suppliers, would result in obtaining more discount with the same quality of service.

## ***Materials with higher cost***


Figure 1.6: Higher Costing Materials

* The materials 3004, 3028, 1374 & 11444 are the costliest materials procured by the plants. ABC electricals should focus to understand the reason for high cost and investigate if there any alternative products that can be sourced at a lesser cost to substitute these high cost materials without compromising the manufacturing quality.

## ***Global Purchasing Pattern***

The trend for all the materials sourced by the manufacturing plants within ITB system.

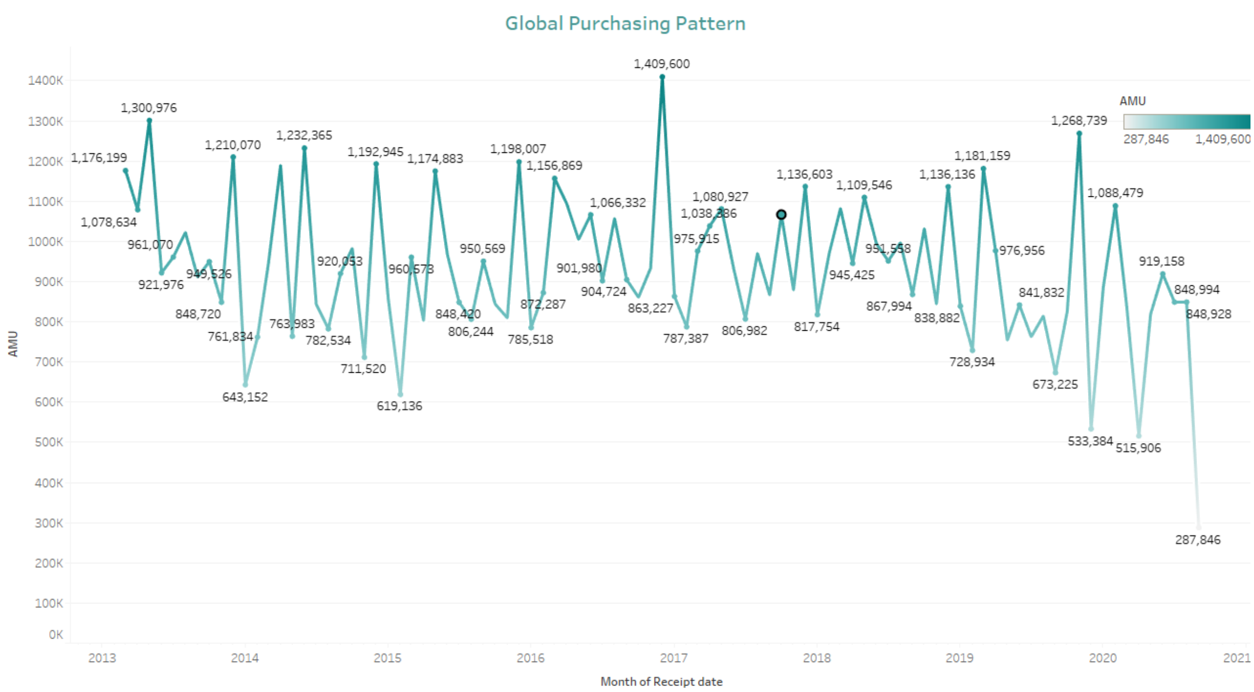


Figure 1.7: Global Purchasing Pattern

* Over the years, the purchasing pattern of the ITB system shows a downward trend. This could possibly indicate, there are certain materials that are no more sourced for the manufacturing purpose. Otherwise, the materials are sourced at a lesser quantity to not incur more cost of sourcing. But this could result in a stock out situation for certain source materials and ABC electricals should review this area to run a smooth manufacturing process.

***Top four manufacturing plants***

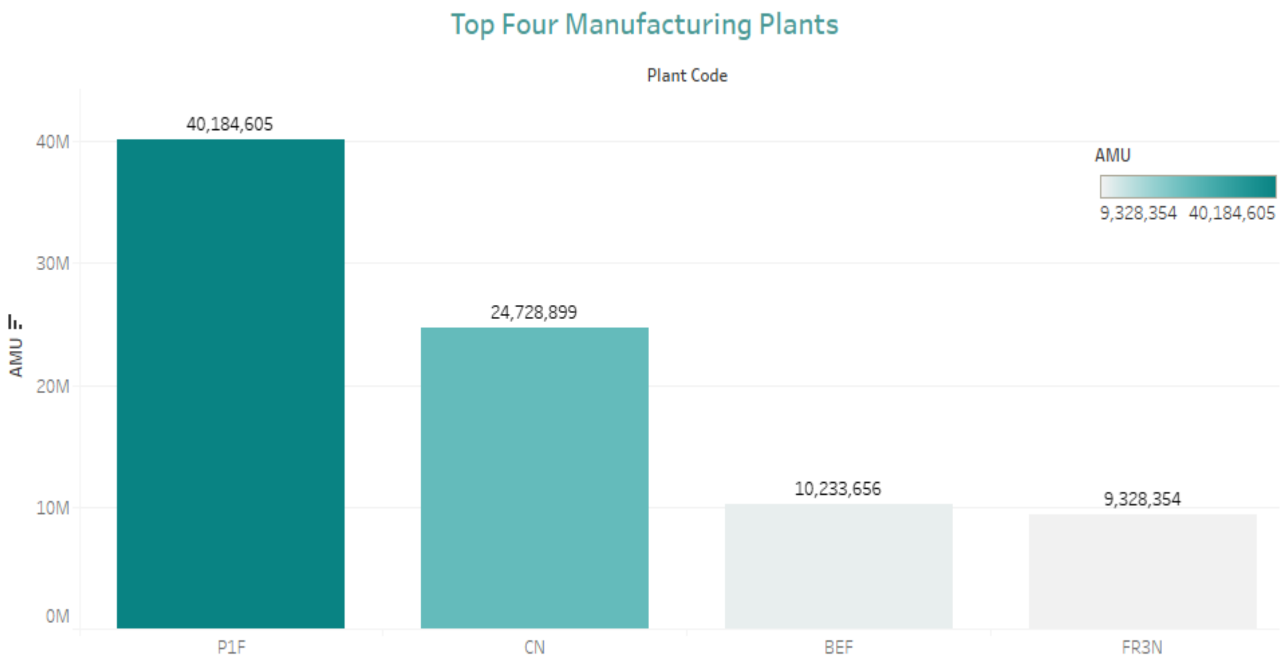


Figure 1.8: Top Four Manufacturing Plants

* Manufacturing plant P1F is the large-scale manufacturer with a significant total quantity usage of 1,433,145.
* Manufacturing plant CN is second highest manufacturer, followed by plants FR & BEF

## ***Trend for the top 8 materials***

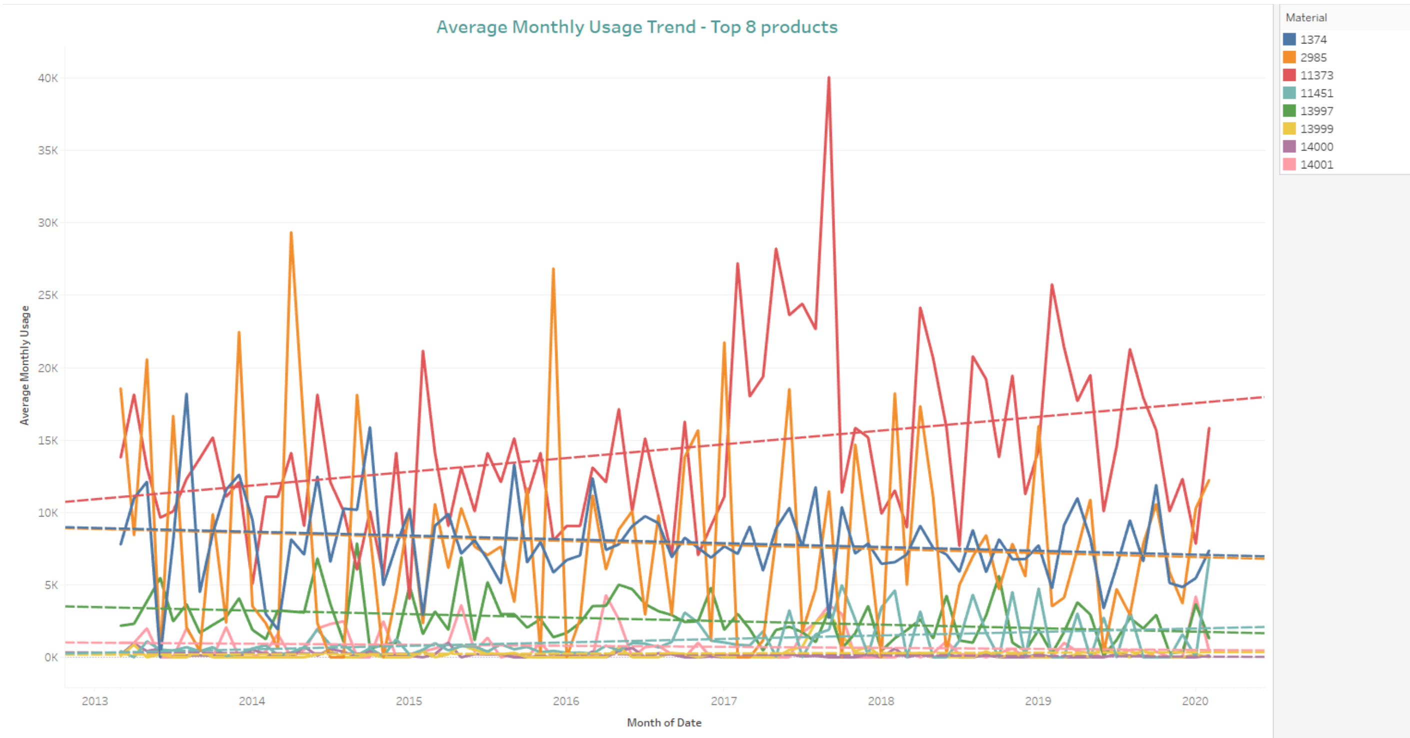


Figure 1.9: AMU Trend – Top 8 Materials

* The graph shows the trend for top 8 materials sourced for manufacturing purpose.

* Over the years, the material code '11373' is commonly procured and used for manufacturing purpose by the top manufacturing plants. An upward trend is seen and clearly indicates, it is a key ingredient used for manufacturing end products of ABC Electricals by the top plants.

* Other materials are sourced at a constant level and few observe a slight downward trend, this indicates there could be alternate source materials used for manufacturing or the demand for the manufactured product is slightly fading.

## *Top materials sourced by the top Manufacturing Plants*

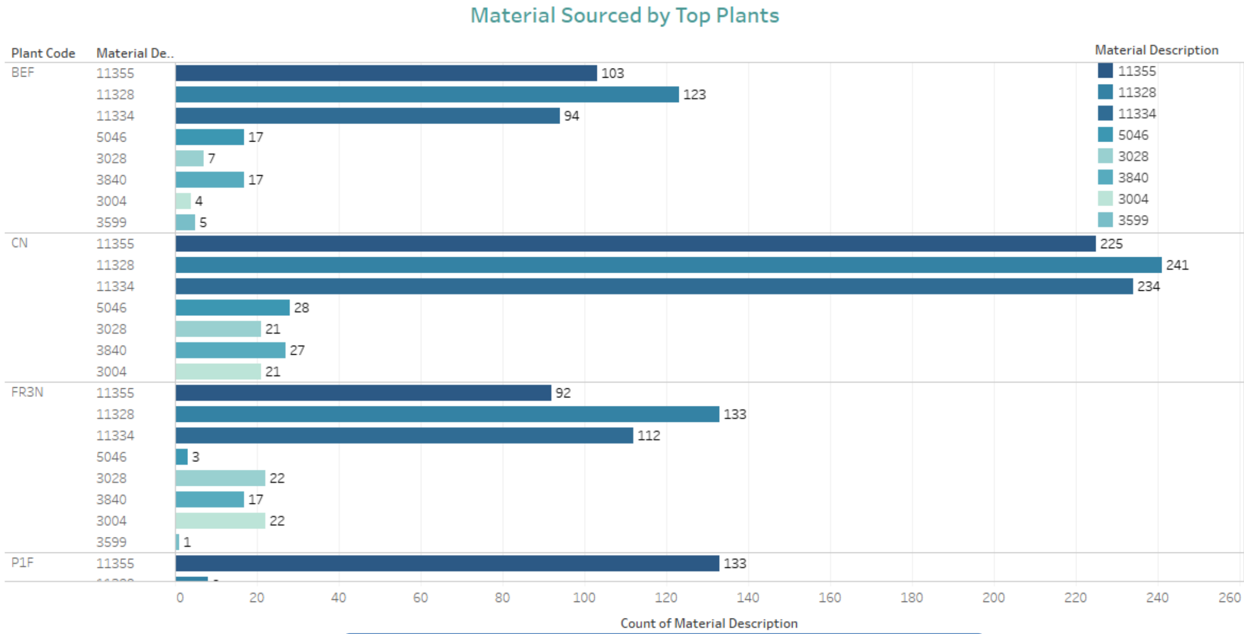


Figure 2.0: Materials Sourced by Top Plants

- Material code ‘11373’ is the highly sourced material for production by the three top plants BEF, CN & FR.
- Material code ‘13997’ is the second highest sourced material for production by the three top plants BEF, CN & FR.
- Plant code P1F, sources a high amount of the material ‘1374’, followed by material ‘2985’

# Detailed analysis of material – 11373

## Purchasing Pattern of material – 11373

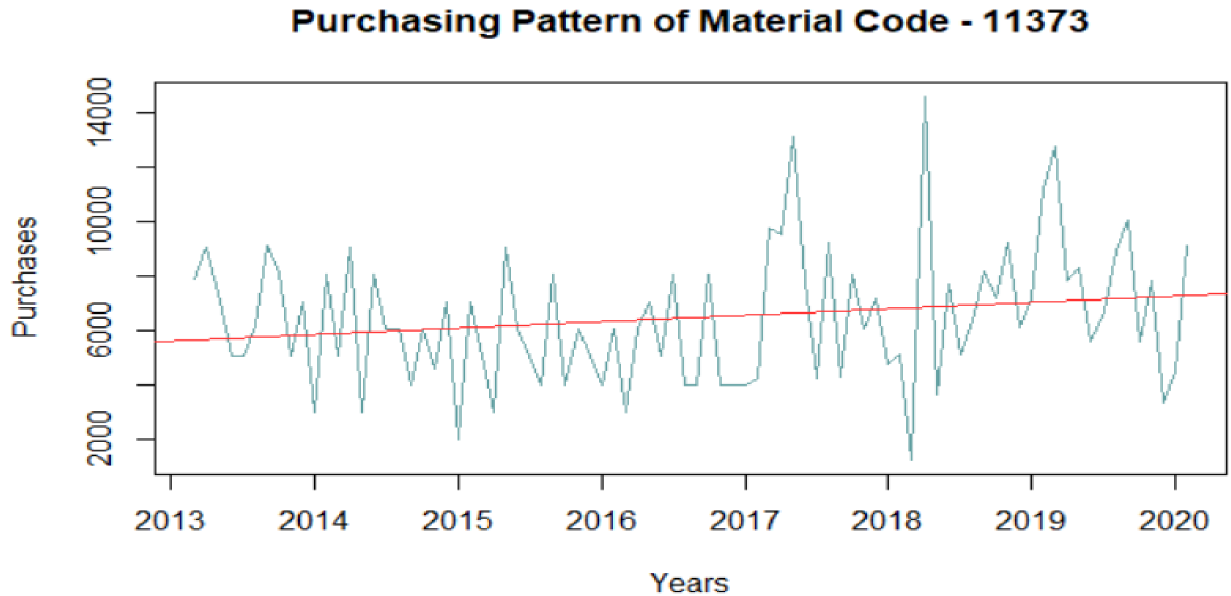


Figure 2.1: Purchasing Pattern – "11373"

It's understood, material '11373' is highly sourced by top three products. It is the key source material and shows an upward trend.

## Seasonality of material – 11373

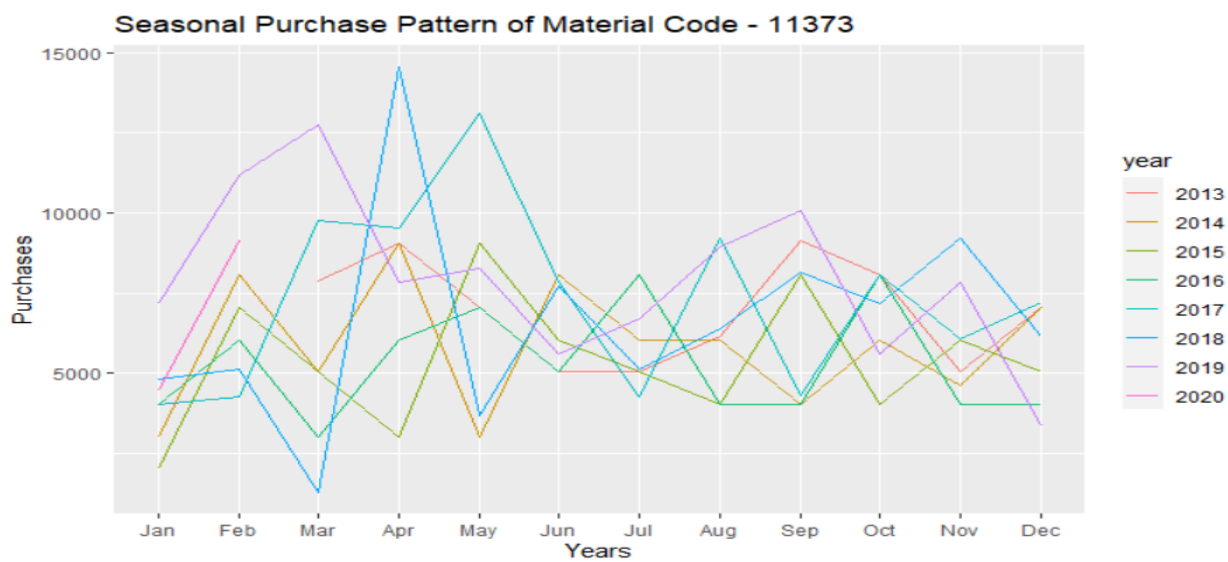


Figure 2.2: Seasonality – "11373"

Throughout the years, a seasonality is observed. This can be better understood by decomposing the time series data for the material.

***Decomposition of material – 11373***

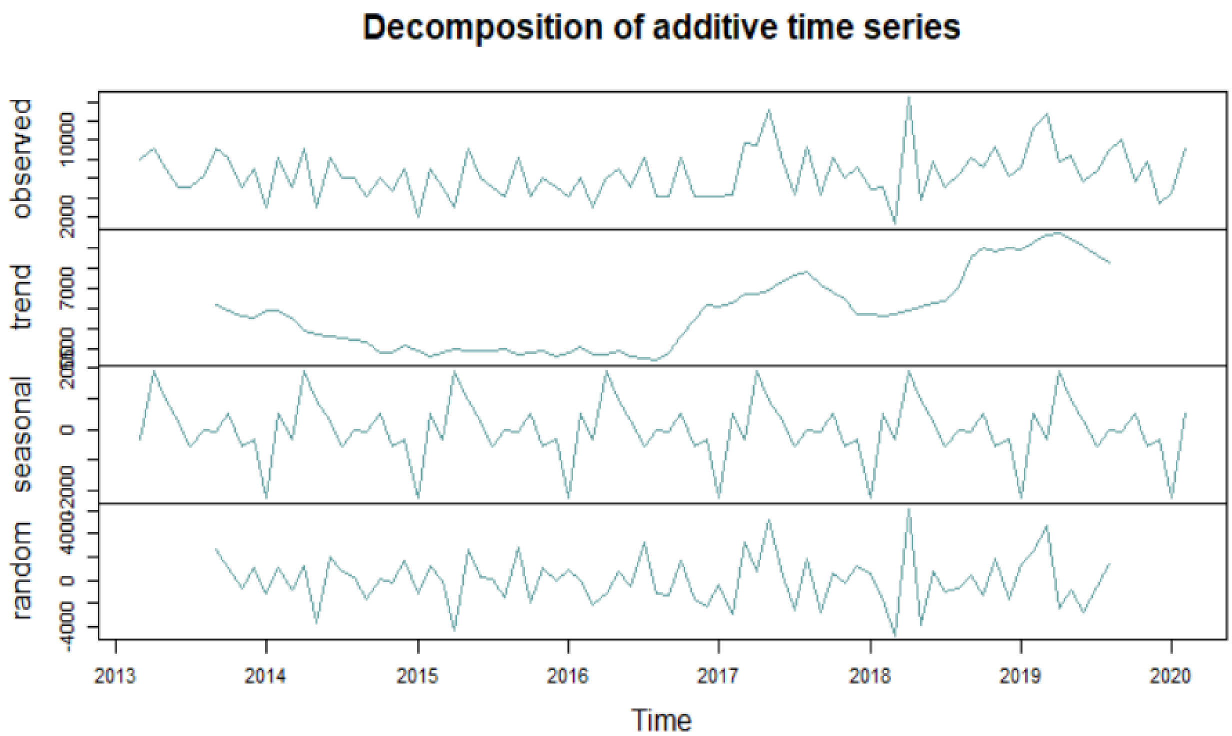


Figure 2.3: Decomposition – "11373"

The material shows an interesting purchase pattern. Whereas, the material sourcing shows a slight downward trend until mid of 2016 and then shows a significant upward trend.

The material shows a seasonality pattern. The monthly box plot explains the monthly purchase pattern.

***Monthly Purchasing Pattern – 11373***

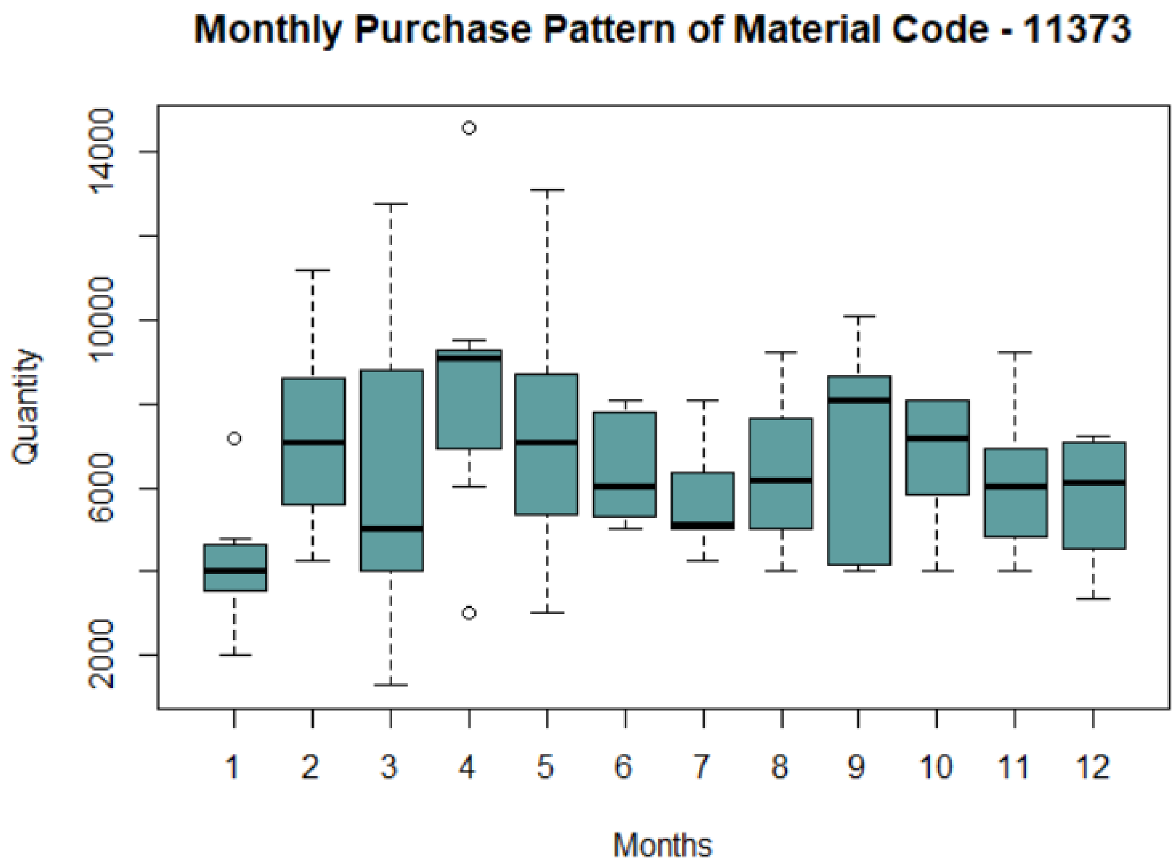


Figure 2.4: Boxplot – "11373"

The median sourcing and the production are usually higher in the starting of the year apart from January.

It was understood, the main supplier is located at the high snowfall area and at winters sourcing this material is difficult, hence the month of September and October records high sourcing to tackle the following months.

However, January records the least material availability due to the climatic conditions, which in turn affects the manufacturing process for ABC electricals.

# *Detailed analysis of material – 1374*

## *Purchasing Pattern of material – 1374*

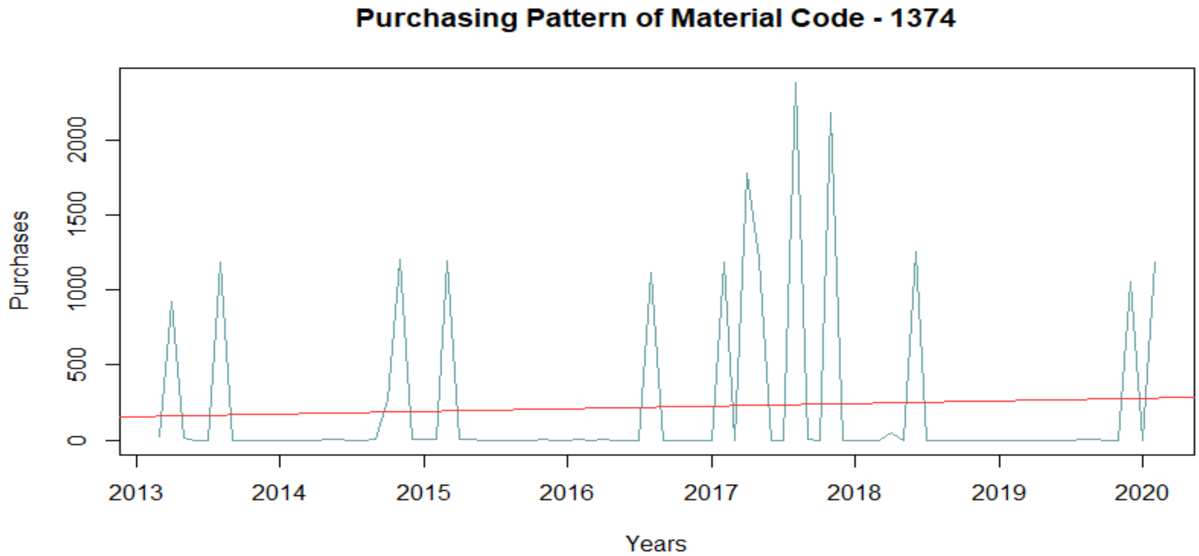


Figure 2.5: Purchasing Pattern– "1374"

This material is sourced at certain months, stored and utilized in manufacturing. Then as per the demand, it is sourced again.

## *Seasonality of material – 1374*

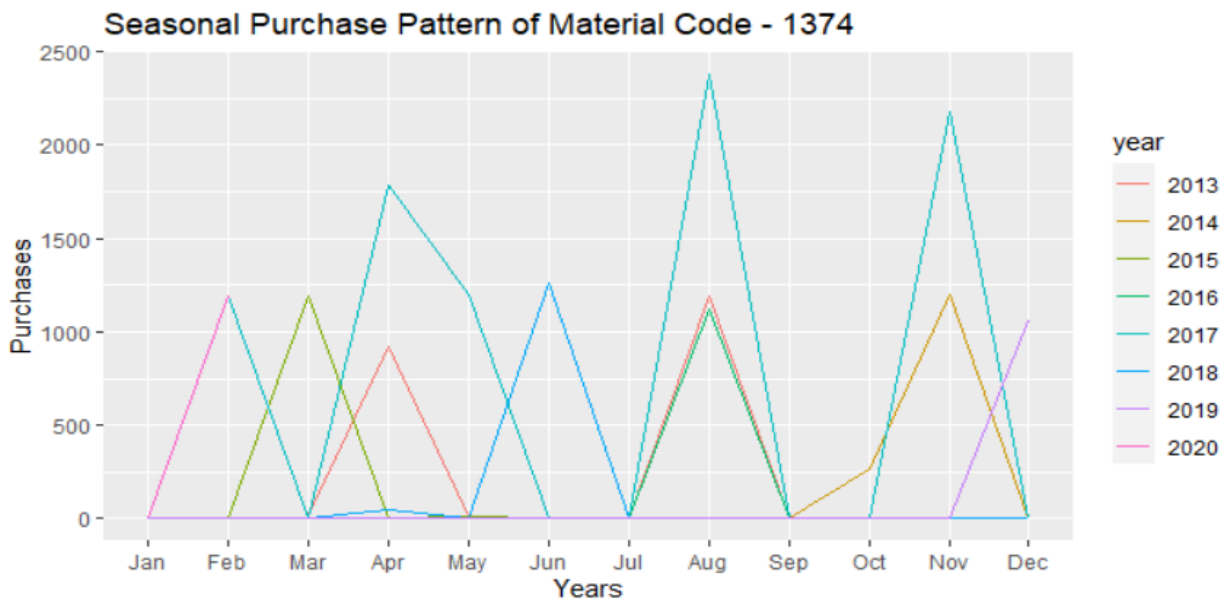


Figure 2.6: Seasonality – "1374"

***Decomposition of material – 1374***

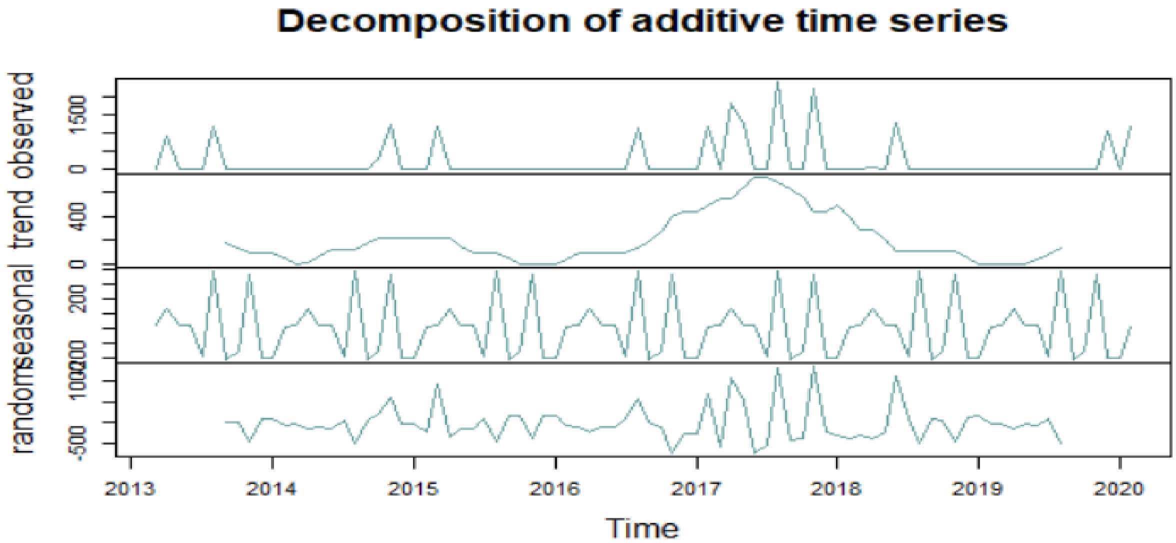


Figure 2.7: Seasonality – "11373"

The material sourcing shows a mixed trend of both upward and downward.

***Monthly Purchasing Pattern of material – 1374***

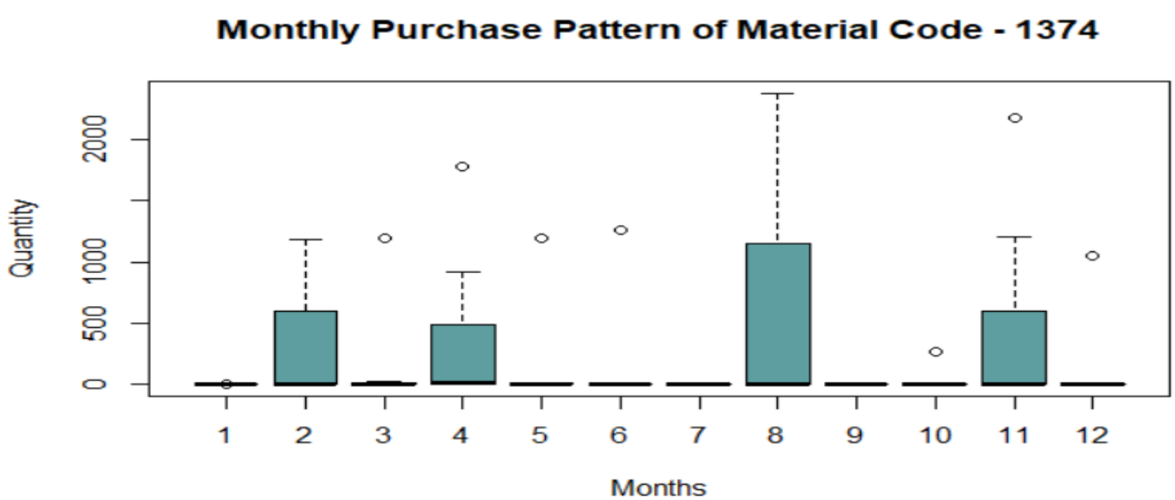


Figure 2.8: Boxplot – "1374"

The material '1374' is usually sourced in bulk on specific months and stored for manufacturing purpose. Other months record a minimal sourcing based on the requirements or stock outs. September month records the highest sourcing

## Detailed analysis of material – 1374

### Purchasing Pattern of material – 1374

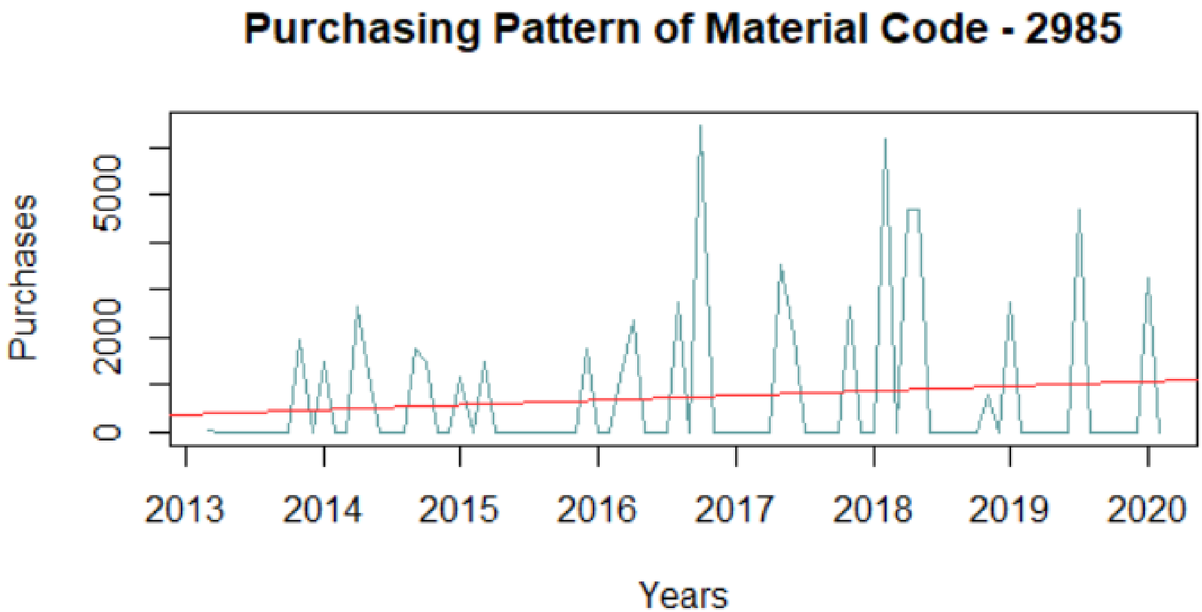


Figure 2.9: Purchasing Pattern – "2985"

Overall the material '1374' shows a slight upward trend.

### Seasonality of material – 1374

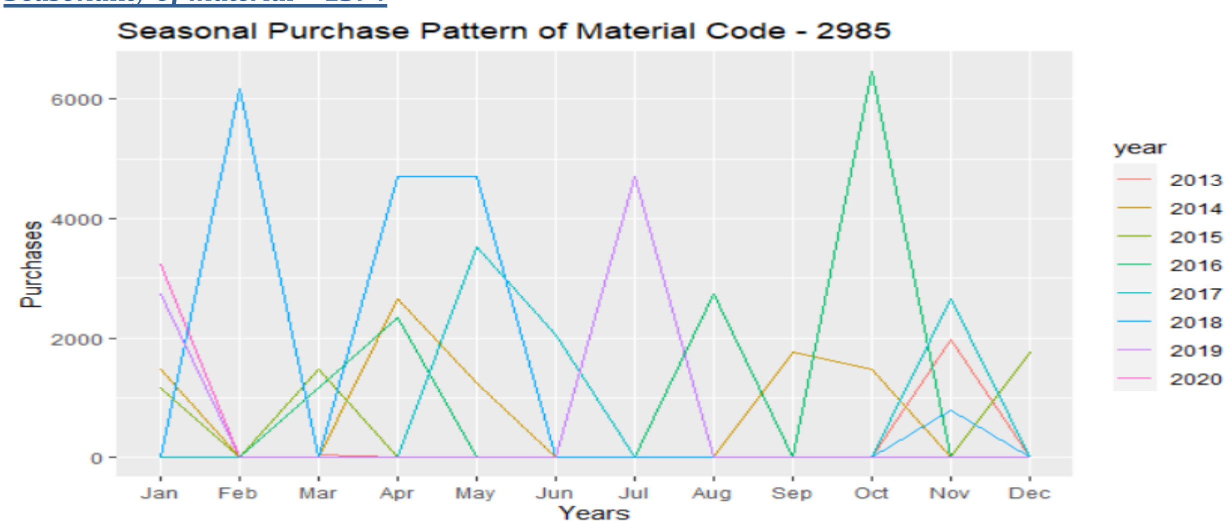


Figure 3.0: Seasonality – "2985"

## *Decomposition of material – 1374*

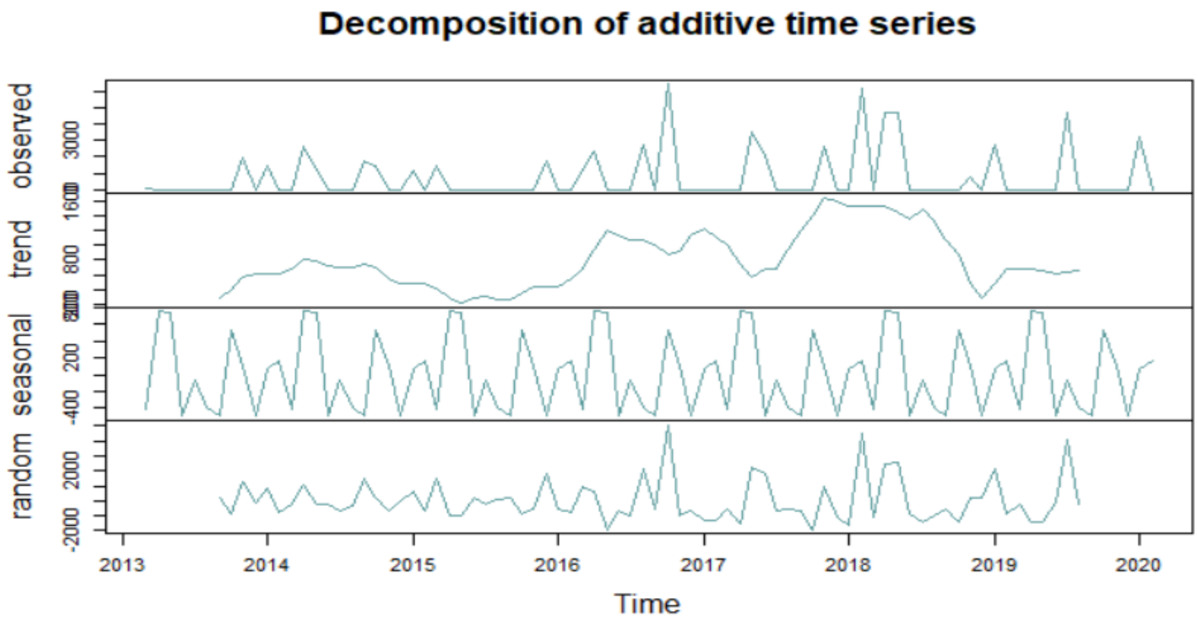


Figure 3.1: Decomposition – "2985"

Material has a mixed trend throughout and a seasonality pattern is observed.

## *Monthly Purchasing Pattern of material – 1374*

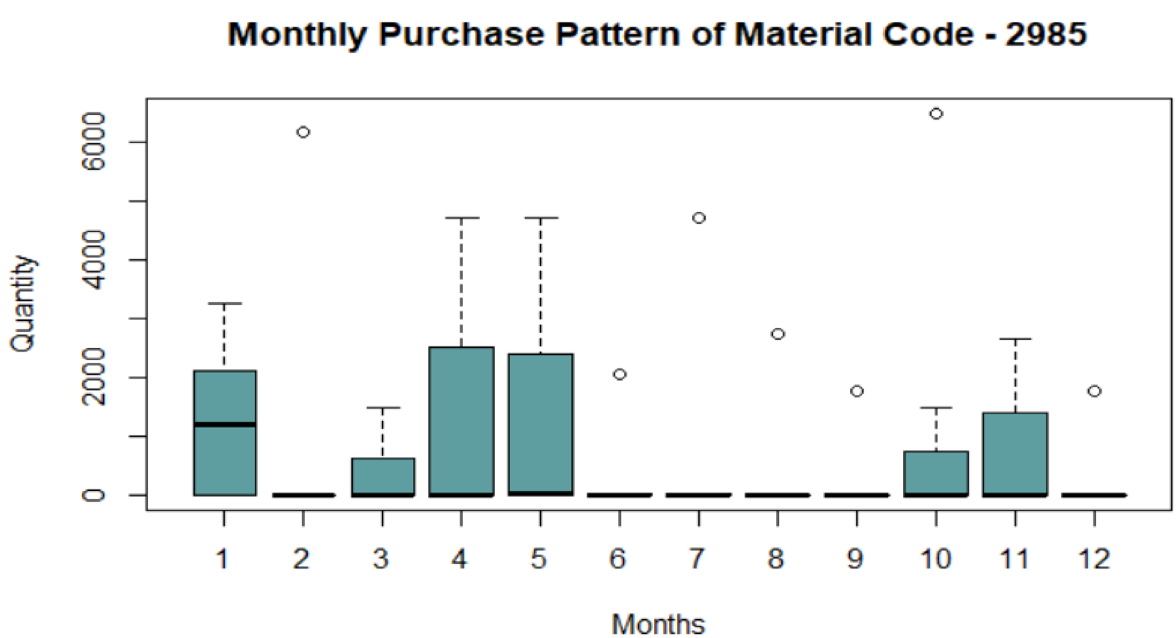


Figure 3.2: Boxplot - "2985"

Material '2985' is highly sourced in the month of January. In the first part of the year the sourcing is high and minimal sourcing observed in the mid-year. October and November record a high sourcing and usage in the second part of the year.

## 8. Data Split

The data set is divided into train and validation tests. We have total of 84 months of data. Hence, we have chosen 60 months of data as train set and 24 months of data as test set. We have will build one time series and one neural network model on this data set.

## 9. Modelling Approach

Building an ARIMA model for the top commonly used material code "11373".

The Auto ARIMA model gives us ARIMA (0,0,0) with MAPE (Mean Absolute Percentage Error) of 33.96 and RMSE (Root Mean Square Error) of 2142.99 on the train data. The AIC (Akaike Information Criterion) value is 1097.67. We need to check the stability of the model before testing the accuracy of model. For our model to be stable, below three criteria should be met.

1. Residuals Should be uncorrelated.
2. Mean of the residuals should be zero (Or close to zero)
3. Residuals should be normal

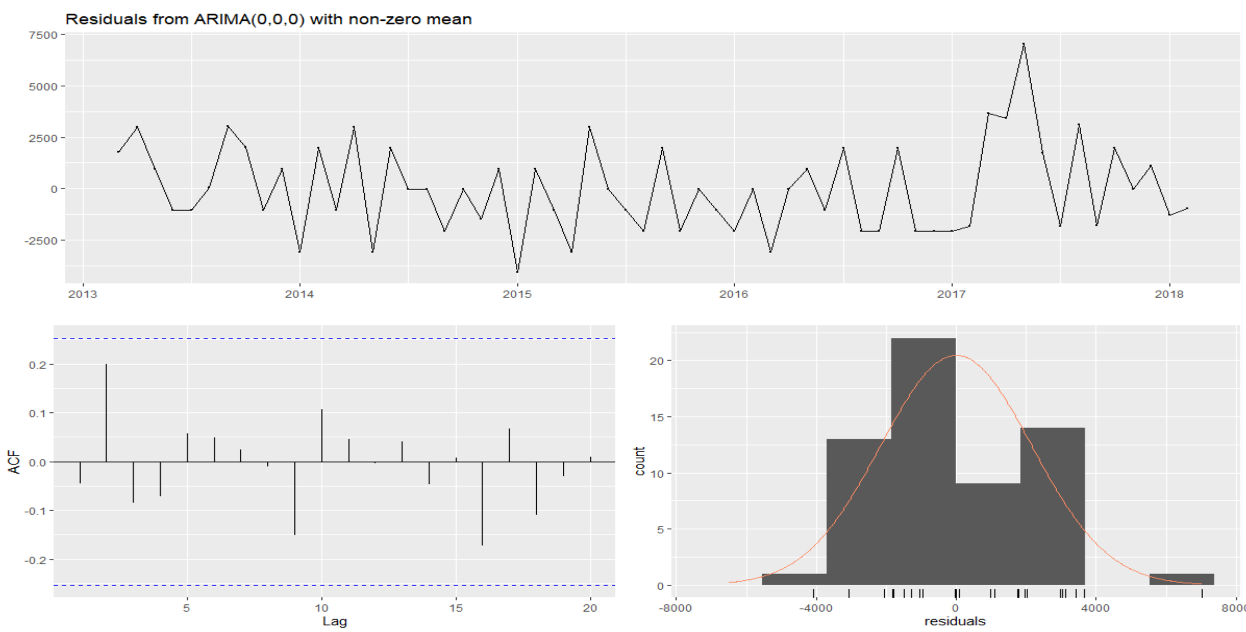


Figure 3.3: Residual Plot

* The residuals almost look normal, they have nonzero mean but closer to zero and the ACF plot shows us that the residuals are uncorrelated.
* The Box-Ljung test for the Auto ARIMA model gives a p value of 0.72, indicating there is no correlation.

***Predicting Average Monthly Usage for the next 24 months***

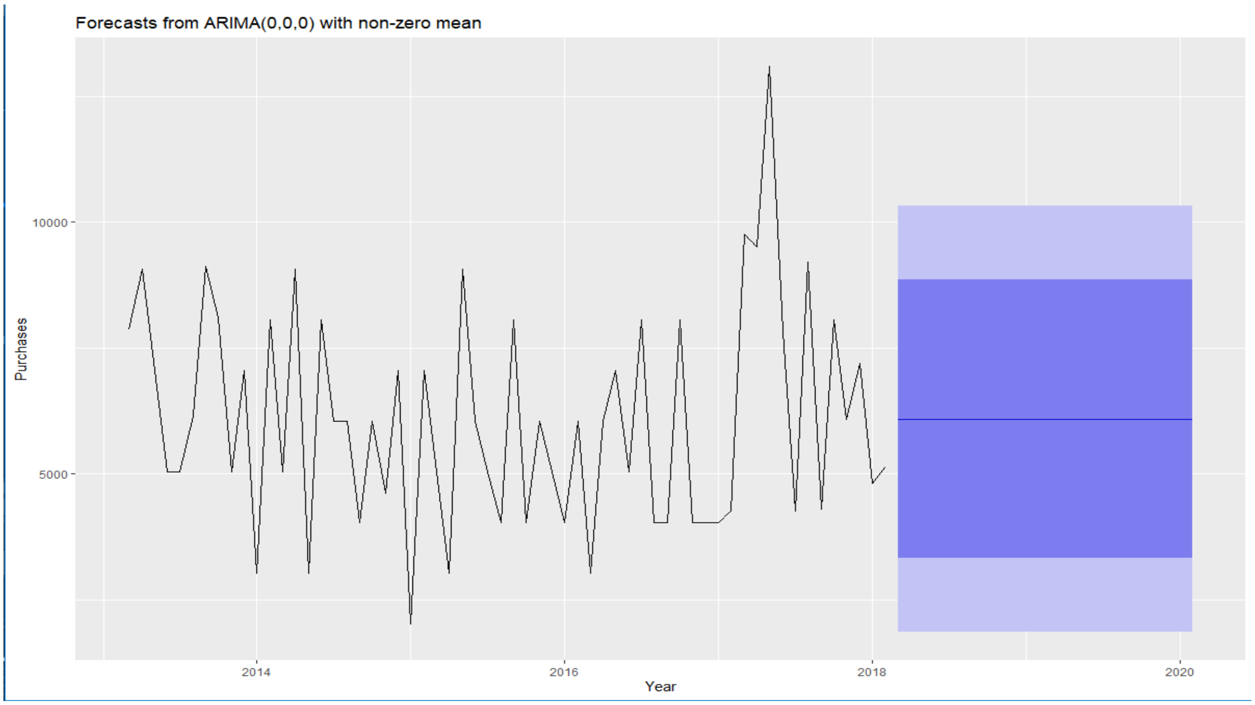


Figure 3.4: Auto Arima - Forecast

The forecasted values are shown above. We can clearly see that the trend and seasonality are not captured by Auto ARIMA properly. The accuracy evaluation of test data gives us the MAPE & RMSE of infinity, as there are months with zero purchases.

```
> accuracy(model.predict, valid.data)
            ME RMSE MAE  MPE MAPE ACF1 Theil's U
Test set -Inf  Inf Inf -Inf  Inf   NA       Inf
```

In this scenario, Mean Arctangent Percentage Error – MAAPE (*see appendix*) would be the best prediction evaluator as it ignores observations with missing values in Y (e.g. #N/A or blank). We get an MAAPE value of 1.57.
In order to incorporate trend and seasonality in the prediction, a model tuning is required with appropriate (p, d, q) & (P, D, Q).

ARIMA models expects the data to be stationary to give the accurate predictions.

### *(i) Difference*

With the r code ndiffs(data), the number of differences required for time series data to be stationary can be found out. We got an output of d = 1, hence we will be taking a difference of 1 to make or time series stationary.

### *(ii) Auto Correlation Function*

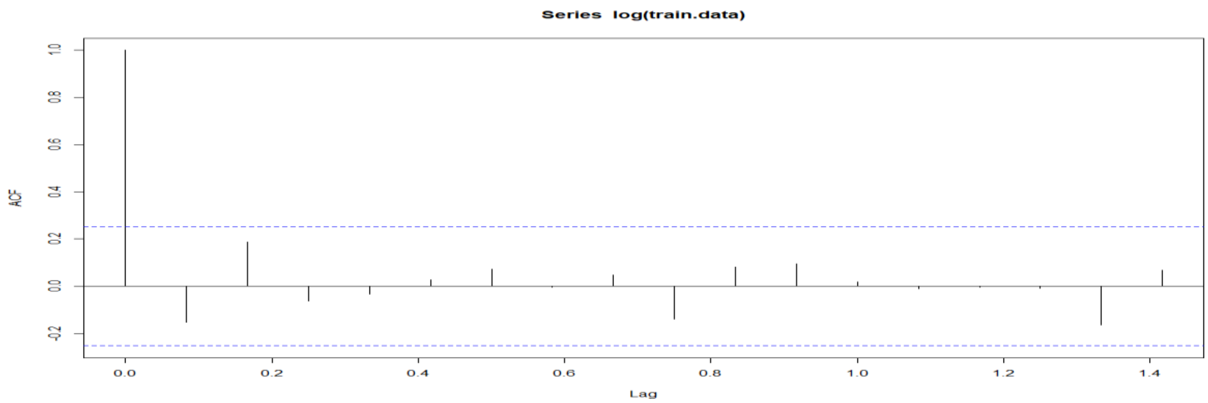


Figure 3.5: ACF Plot

Q value is 1

### *(iii) Partial Auto Correlation Function*

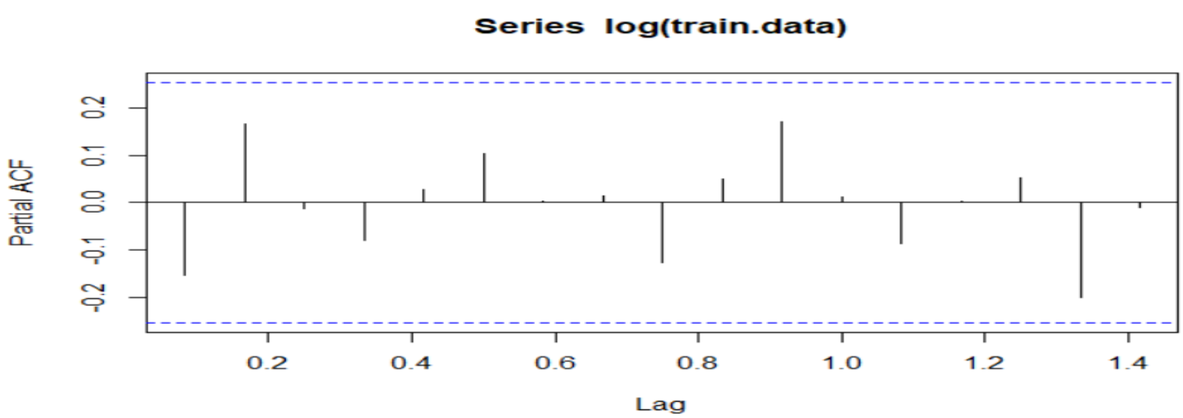


Figure 3.6: PACF Plot

P value is 0

After trying different combinations of (p, d, q) we will get best accuracy on (0,1,1). Hence, we will build ARIMA (0,1,1). It gives us a better MAAPE of 1.55 on the test data. As shown below. seasonality is also captured for this model.

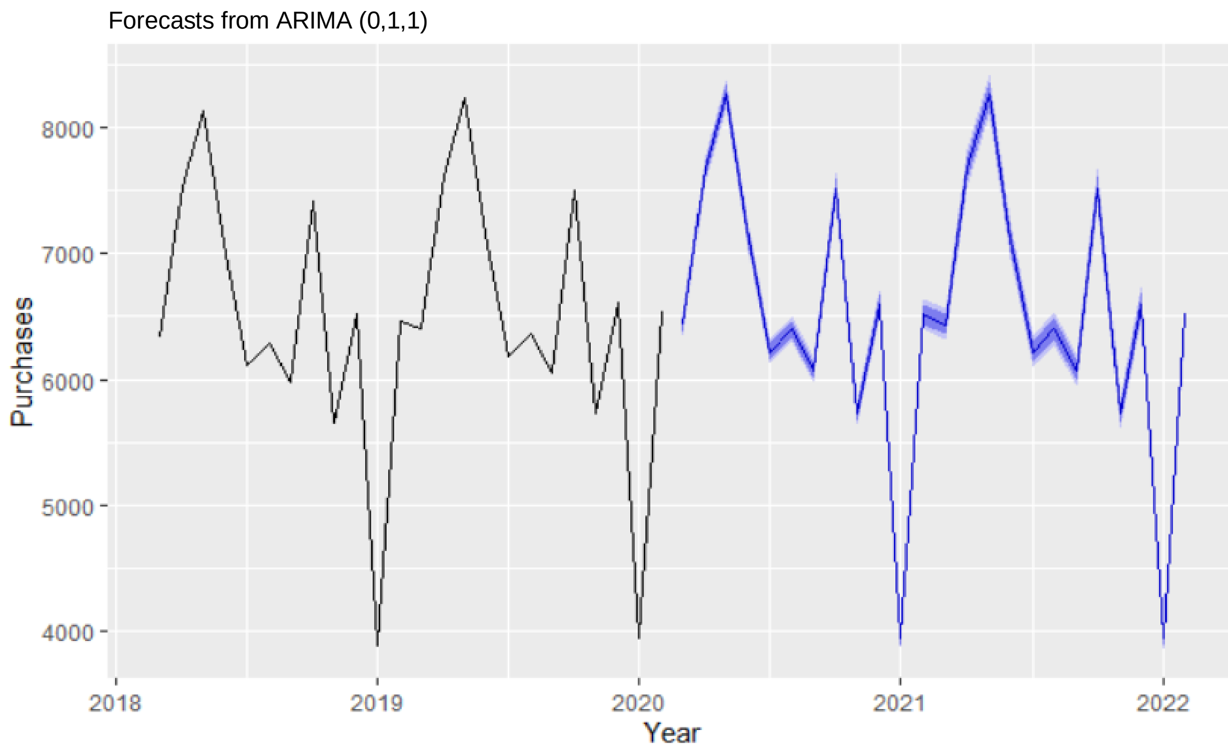


Figure 3.7: ARIMA (0,1,1) - Forecast

## *Linear Regression*

A simple linear regression model could be run to predict the Average Monthly Usage with the help of the dependent variables cost and discount. We get an RMSE of 367.9 for the product code “11373”. However, Linear model will not be able to capture much of the variation because, there are not many numeric variables present in the data. Hence, it is not preferable to use simple linear regression, rather go for Time Series modelling.

### *Progress in project as on 10/11/2020:*

- Data was collected and data analysis along with EDA was done.
- Identified the primary products and top plants to work on based on their purchasing pattern.
- Linear Regression and Time Series models were built on training data sets and the models were tested on validation sets.

***Next Steps:***

* Further exploration of EDA to find out any interesting insights
* Building a hierarchical time series analysis using ARIMA models on selected materials in selected manufactured plants.
* Building Neural Network models
* Ensemble of all models by averaging the prediction values to get the better accuracy
* Comparing the model performance on different models built.

***Challenges:***

* As building individual models for each time series is time consuming, we are trying to implement a research paper on hierarchical or grouped time series to forecast the Average Monthly Usage figures for a set of materials or hierarchy of manufacturing plants.
* Implementation of ensemble of all models to achieve better performance

## 10. Actionable insights & recommendations

* ABC electricals should review the high costing materials 3004, 3028, 1374 & 11444, to decide if the same can be obtained at a cheaper cost.

* The plants ID-PT, 29,594 & US62 tend to manufacture in small scale and ABC electricals should review the cost of operation in these plants, as these plants could incur more cost towards freight and logistics to source a very minimal material.

* With the global sourcing and material usage pattern, a downward trend is observed. This is a critical area where ABC electricals should deep dive to investigate the reason and come up with necessary action plans.

* Major sourcing is done from the supplier "68", who does not provide a good discount compared to other suppliers. ABC electricals could bargain for a discount or look for an alternate supplier who could provide a good discount for their bulk material purchases.

* ABC electricals should look for an alternate supplier, who could supply the top commonly sourced product "11373" during the winters.

## *11. References & Bibliography*


1. *https://www.academia.edu/2767638/Time_series_and_forecasting_in_R*
2. *https://www.academia.edu/2996103/Optimal_combination_forecasts_for_hierarchical_time_series*
3. *https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/*
4. *https://cran.r-project.org/web/packages/rtweet/vignettes/auth.html*
5. *https://www.apics-houston.org/blogpost/1656776/304529/Why-is-Demand-Forecasting-Important-for-Effective-Supply-Chain-Management*
6. *https://www.bcg.com/publications/2011/supply-chain-management-go-to-market-strategy-sales-operations-planning-hidden-supply-chain-engine.aspx*
7. *https://hbr.org/2019/07/setting-better-sales-goals-with-analytics*
8. *https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/*
9. *https://www.oreilly.com/ideas/evaluating-machine-learning-models/page/4/offline-evaluation-mechanisms-hold-out-validation-cross-validation-and-bootstrapping*
10. *www.data.gov.in (GDP Data)*
11. *https://twitter.com/xelec (Twitter API URL)*
12. *Bates, J. M., & Granger, C. W. J. (1969). The combination of forecasts. Operational Research Quarterly, 20(4), 451–468. https://doi.org/10.1057/jors.1969.103*
13. *Hyndman, R. J., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008). Forecasting with exponential smoothing: The state space approach. Berlin: Springer-Verlag.*
14. *Athanasopoulos, G., Ahmed, R. A., & Hyndman, R. J. (2009). Hierarchical forecasts for Australian domestic tourism. International Journal of Forecasting, 25, 146–166. https://robjhyndman.com/publications/hierarchical-tourism/*
15. *Crone, S. F., Hibon, M., & Nikolopoulos, K. (2011). Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction. International Journal of Forecasting, 27(3), 635–660. https://doi.org/10.1016/j.ijforecast.2011.04.001*
16. *https://www.academia.edu/37898847/Inventory_management*
17. *https://www.sciencedirect.com/science/article/pii/S0169207016000121*

# 12. Appendix

## Data Dictionary:

*System*: ERP system/unit that includes multiple manufacturing plants located globally.

*Plant Code*: Unique code used to indicate the manufacturing plants of ABC Electricals.

*Plant Name:* Unique name of each manufacturing plants.

*Logistic reference*: Material/Part reference number used for manufacturing purpose.

*Material description*: Unique description for each material sourced by ABC Electricals.

*Supplier Code*: Third party supplier codes of ABC Electricals.

*Quantity*: Average monthly material used for manufacturing by plants.

*Goods Receipt Date*: Indicates the date when the materials are received in the plant.

*Unit Cost*: Per unit cost of the products sourced.

*UOM*: Different Unit of Measures used for the products.

*Total Cost*: Total cost incurred at the time of sourcing the products.

*Discount:* Discount received at the time of sourcing products.

## Reason for using MAAPE:

*A new metric of absolute percentage error for intermittent demand forecasts*

The mean absolute percentage error (MAPE) is one of the most widely used measures of forecast accuracy, due to its advantages of scale-independency and interpretability. However, MAPE has the significant disadvantage that it produces infinite or undefined values for zero or close-to-zero actual values. In order to address this issue in MAPE, we propose a new measure of forecast accuracy called the Mean Arctangent Absolute Percentage Error (MAAPE). MAAPE has been developed through looking at MAPE from a different angle. In essence, MAAPE is a slope as an angle, while MAPE is a slope as a ratio, considering a triangle with adjacent and opposite sides that are equal to an actual value and the difference between the actual and forecast values, respectively. MAAPE inherently preserves the philosophy of MAPE, overcoming the problem of division by zero by using bounded influences for outliers in a fundamental manner through considering the ratio as an angle instead of a slope.

## ***Code Output:***

```
> # Hierarchical Time Series Analysis.
> library(readr)
> library(tidyverse)
> library(fabletools)
> library(fpp3)
> library(forecast)
> library(tseries)
> library(hts)
> library(ggplot2)
> library(dlookr)
> library(outliers)
> library(dplyr)
> library(psych)
> library(zoo)
> top8 <- read.csv(choose.files())
> str(top8)
'data.frame':	2688 obs. of  4 variables:
 $ Material: int  1374 1374 1374 1374 1374 1374 1374 1374 1374 1374 ...
 $ Date    : chr  "3/1/2013" "4/1/2013" "5/1/2013" "6/1/2013" ...
 $ Plant   : chr  "CN" "CN" "CN" "CN" ...
 $ AMU     : int  24 924 15 0 0 1188 1 1 0 0 ...
> top8$Material <- as.character(top8$Material)
> top8$Date <- as.Date(as.yearmon(top8$Date))
> top8$Plant <- as.character(top8$Plant)
> top8$AMU <- as.numeric(top8$AMU)
> boxplot(top8$AMU, main="Purchases", boxwex=0.1)
> outlier_top8 <- boxplot.stats(top8$AMU)
> head(outlier_top8)
$stats
[1]    0.0    0.0    0.0  999.5 2485.0

$n
[1] 2688

$conf
[1] -30.45967  30.45967

$out
  [1]  2646  2744  6468  3528  2646  6174  4704  4704  2744  4704  3234  7872  9072  7056  5040
 [16]  5040  6160  9132  8096  5040  7056  3024  8064  5040  9072  3024  8064  6048  6048  4032
 [31]  6048  4608  7056  7056  5040  3024  9072  6048  5040  4032  8064  4032  6048  5040  4032
 [46]  6048  3024  6048  7056  5040  8064  4032  4032  8064  4032  4032  4032  4256  9760  9520
 [61] 13104  7840  4256  9216  4304  8064  6080  7200  4800  5120 14560  3680  7712  5120  6384
 [76]  8160  7168  9216  6144  7168 11200 12768  7840  8288  5600  6720  8960 10080  5600  7840
 [91]  3360  4480  9120  3168  3243  2750  2662  2962  4508  2622  2702  3000  3600  3528  2646
[106]  4704  3234  4998  4410  4032  3024  6048  7056  3024  4032  4032  3024  4032  3024  3168
[121]  3024 14560  2560  3248  9072 13776 15680 13440 28080  3360  3840  5120  4480  7040  4800
[136]  7680  3072  3072  5600  3840  3360  4480  3360  2672  3000  4000  2940  2592  3136  3024
[151]  3024  3024  3024  3024  3024  3024  3024  2560  3360  2560  2560  5120  2560  2560  3360
[166]  3360  3456  4608  3168  4320  4512  4752  3024  2736  6792  3967  3224  2532  2694  2726
[181]  7760  7690 11105  7620 15830  4505  8566 11570 12608  9489  2987  8136  4720 12479  6600
[196]  7888 10176 14422  3185  7690 10208  2905  7887  9740  7146  8089  6749  5106 10935  6551
[211]  7955  5859  6715  7020 12340  7411  7798  9056  9739  5905  6915  8254  7468  6882  7664
[226]  4769  9025  2955  5297 10313  7625  6990  2706 10364  7856  6453  6568  7105  9025  7558
[241]  4580  5908  8777  5888  8140  6766  6786  7725  4770  9118 10982  8200  3386  6267  9455
[256]  6627 11894  5133  3797  5463  5001 18520  8423 20580 16604  9676 18941 21128 11466 14700
[271]  6173  4410  9094  9114  4410  8526  7057  5586  7645  3823 11663 20387  2650  9212  3727
[286]  8820 10094  2941  4704  2941  7350 15681 21756  4704 13233  4704 11466  8820  8134  7057
[301]  4998 12643  4998  6958  8428  4704  7056  5587 13230  3528  4117  7448 10878  2940  7841
[316] 10584  5880  3724  7056 12250  2672  3024  3024  3024  3056  4032  4032  3024  5040  3024
[331]  4032  5040  3024  4032  4032  4032  3024  3024  3024  5040  3024  4032  3024  4032  4032
[346]  7392  3136  3248  4032  6400  3840  5584  6400  3840  4800  5856  2560  7040  3360  3072
[361]  6144  6720  2560  6496  5600  4480  4480  4480  5600  5600  6720

> sum(top8$quantity<0)
[1] 0
> summary(top8$AMU ~ top8$Material)
top8$AMU     N= 2688

+-------------+-----+----+----------+
|             |     |   N|  top8$AMU|
+-------------+-----+----+----------+
|top8$Material|11373| 336|3579.70238|
|             |11451| 336| 287.01190|
|             | 1374| 336|1987.92560|
|             |13997| 336| 639.33036|
|             |13999| 336|  59.11310|
|             |14000| 336|  38.71429|
```

```
|              |14001| 336| 179.75298|
|              | 2985| 336|1957.02381|
+-------------+-----+----+----------+
|      Overall|     |2688|1091.07180|
+-------------+-----+----+----------+
> prod_11373 <- subset(top8, Material == '11373')
> prod_1374 <- subset(top8, Material == '1374')
> prod_2985 <- subset(top8, Material == '2985')
> p.11373 <- ts(prod_11373[,4], start=c(2013, 3), end=c(2020, 2), frequency=12)
> p.1374 <- ts(prod_1374[,4], start=c(2013, 3), end=c(2020, 2), frequency=12)
> p.2985 <- ts(prod_2985[,4], start=c(2013, 3), end=c(2020, 2), frequency=12)
> plot(p.11373, xlab = 'Years', ylab = 'Purchases',
+      main = 'Purchasing Pattern of Material Code - 11373',col = 'cadetblue')
> abline(reg=lm(p.11373 ~ time(p.11373)), col = "firebrick1")
> ggseasonplot(p.11373, xlab = 'Years', ylab = 'Purchases',
+              main = 'Seasonal Purchase Pattern of Material Code - 11373')
> decompose(p.11373)
$x
       Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
2013               7872  9072  7056  5040  5040  6160  9132  8096  5040  7056
2014  3024  8064  5040  9072  3024  8064  6048  6048  4032  6048  4608  7056
2015  2016  7056  5040  3024  9072  6048  5040  4032  8064  4032  6048  5040
2016  4032  6048  3024  6048  7056  5040  8064  4032  4032  8064  4032  4032
2017  4032  4256  9760  9520 13104  7840  4256  9216  4304  8064  6080  7200
2018  4800  5120  1280 14560  3680  7712  5120  6384  8160  7168  9216  6144
2019  7168 11200 12768  7840  8288  5600  6720  8960 10080  5600  7840  3360
2020  4480  9120

$seasonal
             Jan         Feb         Mar         Apr         May         Jun         Jul
2013                          -321.25231  1881.49769   906.05324   258.94213  -568.16898
2014 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2015 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2016 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2017 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2018 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2019 -2237.44676   510.10880  -321.25231  1881.49769   906.05324   258.94213  -568.16898
2020 -2237.44676   510.10880
             Aug         Sep         Oct         Nov         Dec
2013   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2014   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2015   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2016   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2017   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2018   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2019   -14.94676   -75.33565   523.88657  -550.78009  -312.55787
2020

$trend
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct
2013                         NA       NA       NA       NA       NA       NA 6603.000 6485.000
2014 6443.000 6480.333 6263.167 5965.333 5862.000 5844.000 5802.000 5718.000 5676.000 5424.000
2015 5466.000 5340.000 5424.000 5508.000 5484.000 5460.000 5460.000 5502.000 5376.000 5418.000
2016 5418.000 5544.000 5376.000 5376.000 5460.000 5334.000 5292.000 5217.333 5423.333 5848.667
2017 6572.000 6629.333 6856.667 6868.000 6953.333 7170.667 7334.667 7402.667 7085.333 6942.000
2018 6392.000 6310.000 6352.667 6476.000 6569.333 6656.000 6710.667 7062.667 7794.667 7993.333
2019 7988.000 8162.000 8349.333 8364.000 8241.333 8068.000 7840.000 7641.333       NA       NA
2020       NA       NA
          Nov      Dec
2013 6317.000 6275.000
2014 5424.000 5592.000
2015 5460.000 5334.000
2016 6245.333 6614.000
2017 6759.333 6361.333
2018 7905.333 8009.333
2019       NA       NA
2020

$random
              Jan          Feb          Mar          Apr          May          Jun          Jul
2013                                     NA           NA           NA           NA           NA
2014 -1181.553241  1073.557870  -901.914352  1225.168981 -3744.053241  1961.057870   814.168981
2015 -1212.553241  1205.891204   -62.747685 -4365.497685  2681.946759   329.057870   148.168981
2016   851.446759    -6.108796 -2030.747685 -1209.497685   689.946759  -552.942130  3340.168981
2017  -302.553241 -2883.442130  3224.585648   770.502315  5244.613426   410.391204 -2510.497685
2018   645.446759 -1700.108796 -4751.414352  6202.502315 -3795.386574   797.057870 -1022.497685
2019  1417.446759  2527.891204  4739.918981 -2405.497685  -859.386574 -2726.942130  -551.831019
2020           NA           NA
              Aug          Sep          Oct          Nov          Dec
2013           NA  2604.335648  1087.113426  -726.219907  1093.557870
2014   344.946759 -1568.664352   100.113426  -265.219907  1776.557870
```

```
2015 -1455.053241  2763.335648 -1909.886574  1138.780093    18.557870
2016 -1170.386574 -1315.997685  1691.446759 -1662.553241 -2269.442130
2017  1828.280093 -2705.997685   598.113426  -128.553241  1151.224537
2018  -663.719907   440.668981 -1349.219907  1861.446759 -1552.775463
2019  1333.613426           NA           NA           NA           NA
2020

$figure
 [1]  -321.25231  1881.49769   906.05324   258.94213  -568.16898   -14.94676   -75.33565
 [8]   523.88657  -550.78009  -312.55787 -2237.44676   510.10880

$type
[1] "additive"

attr(,"class")
[1] "decomposed.ts"
> plot(decompose(p.11373), col = 'cadetblue')
> boxplot(p.11373 ~ cycle(p.11373), col = 'cadetblue',
+         main = 'Monthly Purchase Pattern of Material Code - 11373',
+         ylab = 'Quantity', xlab = 'Months')
> ggtsdisplay(p.1374)
> autoplot(decompose(p.1374))
> plot(p.1374, xlab = 'Years', ylab = 'Purchases',
+      main = 'Purchasing Pattern of Material Code - 1374',col = 'cadetblue')
> abline(reg=lm(p.1374 ~ time(p.1374)), col = "firebrick1")
> ggseasonplot(p.1374, xlab = 'Years', ylab = 'Purchases',
+              main = 'Seasonal Purchase Pattern of Material Code - 1374')
> decompose(p.1374)
$x
      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
2013             24  924   15    0    0 1188    1    1    0    0
2014    0    0    1    1    2    3    0    0    2  268 1201    2
2015    3    3 1194    6   10    0    0    1    1    1    2    0
2016    0    2    1    4    1    0    0 1120    0    0    0    0
2017    0 1188    0 1782 1200    0    1 2382    2    1 2182    0
2018    0    0    0   50    0 1260    0    0    0    1    0    0
2019    0    0    0    0    0    0    1    2    2    0    0 1056
2020    1 1188

$seasonal
            Jan        Feb        Mar        Apr        May        Jun        Jul        Aug
2013                         19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2014 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2015 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2016 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2017 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2018 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2019 -195.83275   10.72975   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863
2020 -195.83275   10.72975
            Sep        Oct        Nov        Dec
2013 -208.53414 -157.61748  367.73669 -195.99248
2014 -208.53414 -157.61748  367.73669 -195.99248
2015 -208.53414 -157.61748  367.73669 -195.99248
2016 -208.53414 -157.61748  367.73669 -195.99248
2017 -208.53414 -157.61748  367.73669 -195.99248
2018 -208.53414 -157.61748  367.73669 -195.99248
2019 -208.53414 -157.61748  367.73669 -195.99248
2020

$trend
             Jan          Feb          Mar          Apr          May          Jun          Jul
2013                                    NA           NA           NA           NA           NA
2014  99.7500000  50.2500000    0.7916667  11.9583333  73.1250000 123.2500000 123.4583333
2015 224.0833333 224.1250000 224.1250000 212.9583333 151.8750000 101.8333333 101.6250000
2016   1.0833333  47.7083333  94.2916667  94.2083333  94.0833333  94.0000000  94.0000000
2017 440.8750000 493.5000000 546.1666667 546.2916667 637.2500000 728.1666667 728.1666667
2018 489.7916667 390.5000000 291.1666667 291.0833333 200.1666667 109.2500000 109.2500000
2019   0.1250000   0.2500000   0.4166667   0.4583333   0.4166667  44.4166667  88.4583333
2020          NA          NA
             Aug          Sep          Oct          Nov          Dec
2013          NA 178.4583333 139.0416667 100.0416667  99.6250000
2014 123.7083333 173.5416667 223.4583333 224.0000000 224.2083333
2015 101.4583333  51.7083333   1.9166667   1.4583333   1.0833333
2016 143.4166667 192.7916667 266.8333333 390.8750000 440.8333333
2017 678.6666667 629.1666667 557.0000000 434.8333333 437.3333333
2018 109.2500000 109.2500000 107.1666667 105.0833333  52.5833333
2019 138.0000000          NA          NA          NA          NA
2020

$random
            Jan        Feb        Mar        Apr        May        Jun        Jul        Aug
```

```
2013                                         NA          NA          NA          NA          NA          NA
2014   96.08275  -60.97975  -19.25058 -138.25058  -93.42419 -143.54919   70.74942 -505.07697
2015  -25.25058 -231.85475  950.41609 -334.25058 -164.17419 -125.13252   92.58275 -481.82697
2016  194.74942  -56.43808 -112.75058 -217.50058 -115.38252 -117.29919  100.20775  595.21470
2017 -245.04225  683.77025 -565.62558 1108.41609  540.45081 -751.46586 -532.95891 1321.96470
2018 -293.95891 -401.22975 -310.62558 -368.37558 -222.46586 1127.45081   84.95775 -490.61863
2019  195.70775  -10.97975  -19.87558 -127.75058  -22.71586  -67.71586  106.74942 -517.36863
2020         NA         NA
            Sep        Oct        Nov        Dec
2013   31.07581   19.57581 -467.77836   96.36748
2014   36.99248  202.15914  609.26331  -26.21586
2015  157.82581  156.70081 -367.19502  194.90914
2016   15.74248 -109.21586 -758.61169 -244.84086
2017 -418.63252 -398.38252 1379.42998 -241.34086
2018   99.28414   51.45081 -472.82002  143.40914
2019         NA         NA         NA         NA
2020

$figure
 [1]   19.45891  127.29225   22.29919   23.29919 -194.20775  381.36863 -208.53414 -157.61748
 [9]  367.73669 -195.99248 -195.83275   10.72975

$type
[1] "additive"

attr(,"class")
[1] "decomposed.ts"
> plot(decompose(p.1374), col = 'cadetblue')
> boxplot(p.1374 ~ cycle(p.1374), col = 'cadetblue',
+         main = 'Monthly Purchase Pattern of Material Code - 1374',
+         ylab = 'Quantity', xlab = 'Months')
> ggtsdisplay(p.2985)
> autoplot(decompose(p.2985))
> plot(p.2985, xlab = 'Years', ylab = 'Purchases',
+      main = 'Purchasing Pattern of Material Code - 2985',col = 'cadetblue')
> abline(reg=lm(p.2985 ~ time(p.2985)), col = "firebrick1")
> ggseasonplot(p.2985, xlab = 'Years', ylab = 'Purchases',
+              main = 'Seasonal Purchase Pattern of Material Code - 2985')
> decompose(p.2985)
$x
      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
2013            50    0    0    2    3    0    0    0 1972    0
2014 1471    0    0 2646 1248    0    0    0 1764 1470    0    0
2015 1176    0 1470    0    0    0    0    0    0    1    0 1764
2016    0    0 1176 2352    0    0    0 2744    0 6468    0    0
2017    0    0    0    0 3528 2058    0    0    0    0 2646    0
2018    0 6174    0 4704 4704    0    0    0    0    0  784    0
2019 2744    0    0    0    1    0 4704    0    0    0    0    0
2020 3234    0

$seasonal
            Jan        Feb        Mar        Apr        May        Jun        Jul        Aug
2013                       -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2014   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2015   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2016   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2017   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2018   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2019   71.64294  169.49711 -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678
2020   71.64294  169.49711
            Sep        Oct        Nov        Dec
2013 -500.57234  528.94155  106.10127 -500.22512
2014 -500.57234  528.94155  106.10127 -500.22512
2015 -500.57234  528.94155  106.10127 -500.22512
2016 -500.57234  528.94155  106.10127 -500.22512
2017 -500.57234  528.94155  106.10127 -500.22512
2018 -500.57234  528.94155  106.10127 -500.22512
2019 -500.57234  528.94155  106.10127 -500.22512
2020

$trend
           Jan       Feb       Mar       Apr       May       Jun       Jul       Aug       Sep
2013                             NA        NA        NA        NA        NA        NA  289.4167
2014  611.5417  611.4167  684.9167  819.6667  798.7500  716.5833  704.2917  692.0000  753.2500
2015  490.0000  490.0000  416.5000  281.7917  220.5833  294.0833  318.5833  269.5833  257.3333
2016  441.0833  555.4167  669.7500  939.2083 1208.6667 1135.1667 1061.6667 1061.6667 1012.6667
2017 1233.1667 1118.8333 1004.5000  735.0000  575.7500  686.0000  686.0000  943.2500 1200.5000
2018 1519.0000 1519.0000 1519.0000 1519.0000 1441.4167 1363.8333 1478.1667 1335.2500 1078.0000
2019  490.0833  686.0833  686.0833  686.0833  653.4167  620.7500  641.1667  661.5833        NA
2020        NA        NA
           Oct       Nov       Dec
```

```
2013  397.5833  559.8333  611.7500
2014  704.2500  542.0000  490.0000
2015  343.0833  441.0833  441.0833
2016  865.6667  914.6667 1147.4167
2017 1396.5000 1641.5000 1604.7500
2018  882.0000  490.0417  294.0833
2019        NA        NA        NA
2020

$random
             Jan         Feb         Mar         Apr         May         Jun         Jul
2013                                  NA          NA          NA          NA          NA
2014   787.81539  -780.91377  -266.41377  1068.83623  -285.10822  -227.46933  -643.93461
2015   614.35706  -659.49711  1472.00289 -1039.28877  -954.94155   195.03067  -258.22627
2016  -512.72627  -724.91377   924.75289   655.29456 -1943.02488  -646.05266 -1001.30961
2017 -1304.80961 -1288.33044  -585.99711 -1492.49711  2217.89178  1861.11400  -625.64294
2018 -1590.64294  4485.50289 -1100.49711  2427.50289  2528.22512  -874.71933 -1417.80961
2019  2182.27373  -855.58044  -267.58044 -1443.58044 -1386.77488  -131.63600  4123.19039
2020          NA          NA
             Aug         Sep         Oct         Nov         Dec
2013          NA   211.15567  -926.52488  1306.06539  -111.52488
2014  -292.73322  1511.32234   236.80845  -648.10127    10.22512
2015   129.68345   243.23900  -871.02488  -547.18461  1823.14178
2016  2081.60012  -512.09433  5073.39178 -1020.76794  -647.19155
2017  -543.98322  -699.92766 -1925.44155   898.39873 -1104.52488
2018  -935.98322  -577.42766 -1410.94155   187.85706   206.14178
2019  -262.31655          NA          NA          NA          NA
2020

$figure
 [1] -418.50289  757.49711  734.35822 -489.11400  -60.35706 -399.26678 -500.57234  528.94155
 [9]  106.10127 -500.22512   71.64294  169.49711

$type
[1] "additive"

attr(,"class")
[1] "decomposed.ts"
> plot(decompose(p.2985), col = 'cadetblue')
> boxplot(p.2985 ~ cycle(p.1374), col = 'cadetblue',
+         main = 'Monthly Purchase Pattern of Material Code - 2985',
+         ylab = 'Quantity', xlab = 'Months')
> adf.test(p.11373)

	Augmented Dickey-Fuller Test

data:  p.11373
Dickey-Fuller = -3.5868, Lag order = 4, p-value = 0.03956
alternative hypothesis: stationary

> kpss.test(p.11373)

	KPSS Test for Level Stationarity

data:  p.11373
KPSS Level = 0.46942, Truncation lag parameter = 3, p-value = 0.04855

> ndiffs(p.11373)
[1] 1
> adf.test(p.1374)

	Augmented Dickey-Fuller Test

data:  p.1374
Dickey-Fuller = -2.8955, Lag order = 4, p-value = 0.2086
alternative hypothesis: stationary

> adf.test(p.2985)

	Augmented Dickey-Fuller Test

data:  p.2985
Dickey-Fuller = -4.9455, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(p.2985) : p-value smaller than printed p-value
> kpss.test(p.11373)

	KPSS Test for Level Stationarity
```

```
data:  p.11373
KPSS Level = 0.46942, Truncation lag parameter = 3, p-value = 0.04855

> ndiffs(p.11373)
[1] 1
> kpss.test(p.1374)

         KPSS Test for Level Stationarity

data:  p.1374
KPSS Level = 0.14747, Truncation lag parameter = 3, p-value = 0.1

Warning message:
In kpss.test(p.1374) : p-value greater than printed p-value
> ndiffs(p.1374)
[1] 0
> kpss.test(p.2985)

         KPSS Test for Level Stationarity

data:  p.2985
KPSS Level = 0.23025, Truncation lag parameter = 3, p-value = 0.1

Warning message:
In kpss.test(p.2985) : p-value greater than printed p-value
> ndiffs(p.2985)
[1] 0
> adf.test((diff(log(p.11373))))

         Augmented Dickey-Fuller Test

data:  (diff(log(p.11373)))
Dickey-Fuller = -7.488, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test((diff(log(p.11373)))) : p-value smaller than printed p-value
> kpss.test((diff(log(p.11373))))

         KPSS Test for Level Stationarity

data:  (diff(log(p.11373)))
KPSS Level = 0.041768, Truncation lag parameter = 3, p-value = 0.1

Warning message:
In kpss.test((diff(log(p.11373)))) : p-value greater than printed p-value
> ggAcf((diff(log(p.11373))))
> autoplot(diff(p.11373))
> autoplot(diff(log(p.11373)))
> train.data <- window(p.11373, start=c(2013,3), end=c(2018, 2))
> valid.data <- window(p.11373, start=c(2018,3), end=c(2020,2))
> plot(train.data, xlab = 'Years', ylab = 'Purchases',
+      main = 'Train Date - 11373',col = 'cadetblue')
> plot(log(train.data), xlab = 'Years', ylab = 'Purchases',
+      main = 'Train Date - 11373',col = 'cadetblue')
> model.arima<-auto.arima(train.data)
> summary(model.arima)
Series: train.data
ARIMA(0,0,0) with non-zero mean

Coefficients:
           mean
      6087.6667
s.e.   276.6594

sigma^2 estimated as 4670255:  log likelihood=-545.33
AIC=1094.67   AICc=1094.88   BIC=1098.86

Training set error measures:
                        ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set -1.212709e-12 2142.993 1755.867 -14.1188 33.96239 0.7703704 -0.04505592
> autoplot(model.arima)
> checkresiduals(model.arima)

         Ljung-Box test

data:  Residuals from ARIMA(0,0,0) with non-zero mean
Q* = 6.6134, df = 11, p-value = 0.8295

Model df: 1.   Total lags used: 12
```

```
> #Checking for Correlation between residuals
> Box.test(model.arima$residuals, type = "Ljung-Box")

	Box-Ljung test

data:  model.arima$residuals
X-squared = 0.128, df = 1, p-value = 0.7205

> mean(model.arima$residuals)
[1] -1.212709e-12
> shapiro.test(model.arima$residuals)

	Shapiro-Wilk normality test

data:  model.arima$residuals
W = 0.95192, p-value = 0.01919

> checkresiduals(model.arima)

	Ljung-Box test

data:  Residuals from ARIMA(0,0,0) with non-zero mean
Q* = 6.6134, df = 11, p-value = 0.8295

Model df: 1.   Total lags used: 12

> model.predict<-predict(model.arima,n.ahead = 24)
> model.predict<-exp(model.predict$pred)
> model.predict
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018         Inf Inf Inf Inf Inf Inf Inf Inf Inf Inf
2019 Inf Inf Inf Inf Inf Inf Inf Inf Inf Inf Inf Inf
2020 Inf Inf
> autoplot(forecast(model.arima, h=24))+ylab("Purchases")+xlab("Year")
> accuracy(model.predict, valid.data)
           ME RMSE MAE  MPE MAPE ACF1 Theil's U
Test set -Inf  Inf Inf -Inf  Inf   NA       Inf
> MAAPE(model.predict, valid.data, na.rm = TRUE)
[1] 1.570796
> acf(log(train.data)) #q =1
> pacf(log(train.data)) #p = 2
> model.arima1<-arima(log(train.data), order = c(0,1,1), seasonal = list(order=c(0,1,1), period=1
2))
> model.predict1<-predict(model.arima1,n.ahead = 24)
> model.predict1<-exp(model.predict1$pred)
> summary(model.arima1)

Call:
arima(x = log(train.data), order = c(0, 1, 1), seasonal = list(order = c(0,
    1, 1), period = 12))

Coefficients:
          ma1     sma1
      -0.8944  -0.8341
s.e.   0.0710   0.5933

sigma^2 estimated as 0.1543:  log likelihood = -29.81,  aic = 65.62

Training set error measures:
                     ME      RMSE       MAE      MPE     MAPE      MASE       ACF1
Training set 0.07121234 0.3477329 0.2576241 0.712802 2.974037 0.5683302 -0.3122975
> MAAPE(model.predict1, valid.data, na.rm = TRUE)
[1] 1.559064
> autoplot(forecast(model.predict1, h=24))+ylab("Purchases")+xlab("Year")
> mydata <- read.csv(choose.files())
> prod_lm <- subset(mydata, Material.Description == '11373')
> library(caTools)
> set.seed(7)
> spl = sample.split(mydata$AMU, SplitRatio = 0.7)
> train_data = subset(mydata, spl==TRUE)
> test_data = subset(mydata, spl==FALSE)
> lm1 <- lm(mydata$AMU ~ mydata$Total.Cost + mydata$Discount, train_data)
> summary(lm1)

Call:
lm(formula = mydata$AMU ~ mydata$Total.Cost + mydata$Discount,
    data = train_data)

Residuals:
   Min     1Q Median     3Q    Max
-18381    -43    -16    -12  97690
```

```
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       12.4783665  0.8181693   15.25   <2e-16 ***
mydata$Total.Cost  0.2167072  0.0005147  421.02   <2e-16 ***
mydata$Discount   -6.9230933  0.0608240 -113.82   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 780.2 on 958930 degrees of freedom
Multiple R-squared:  0.1733, Adjusted R-squared:  0.1733
F-statistic: 1.005e+05 on 2 and 958930 DF,  p-value: < 2.2e-16

> pred_lm <- predict(lm1, data = test_data)
> accuracy(lm1$fitted.values, test_data$AMU)
                ME     RMSE      MAE MPE MAPE
Test set -3.779977 871.3846 156.4777 NaN  Inf
> RMSE(pred_lm)
[1] 367.9102
```

## *13. Checklist for Interim Report Submission*

| **Before the Interim Report is submitted all the following items must be addressed. Cross out the incorrect option.** | | |
|---|---|---|
| **1** | Have you shared the feedback on Synopsis from Program Director with your Mentor? | **YES** |
| **2** | Have you incorporated the changes suggested in the feedback? | **YES** |
| **3** | If the answer to (2) is NO, have you explained why the changes cannot / should not be made? | **NA** |
| **4** | What proportion of total project work have you completed? | **50%** |
| **5** | Have you put all raw codes and output in the Appendix? | **YES** |
| **6** | Have you numbered all charts/figures/tables/graphs etc? | **YES** |
| **7** | Have you sent the Interim Report to your Mentor at least 7 days before the due date? | **YES** |
| **8** | Have you incorporated the feedback from your Mentor in the Interim Report? | **YES** |
| **9** | Have you followed ALL the guidelines provided in Guidelines for the Interim Report? | **YES** |