Project Report

# Detecting Changes in Customer's Purchasing Pattern Through Change Point Analysis

40.011 Data and Business Analytics

Damian Ong 1004670
Gao Xin 1004517
Chia Yu Ying 1004609
Tan Yunyi 1004645
Solai Lakshimi Priya 1004700

## Content Page

## 1.1 Executive Summary

### Background

Our client, Integrated Decision Systems Consultancy Pte Ltd (IDSC) specialises in the provision of advanced analytics through consultancy in various sectors. Given a dataset of bread sales of 6 different locations, our project objective was to detect changes in customer purchasing pattern using changepoint analysis and provide suggestions to help our client maximise sales. By implementing changepoint analysis, we managed to determine changepoint locations, duration of change, demand patterns and predictability of sales.

### Strategic Imperatives

We explored 2 different methods of identifying changepoints that we will refer to as 'change in mean' and 'change in mean and variance' in R. Next, we tested out these methods with the Poisson and exponential distribution test statistics. Since change in mean requires manual setting of penalty value to adjust sensitivity of the algorithm, change in mean and variance seemed to be a better option as it automatically adjusts the sensitivity of the algorithm to identify changepoints more accurately. We ultimately chose 'change in mean and variance' method because it accounted for our largely scattered datapoints unlike the 'change in mean' method. Unlike Poisson distribution, exponential distribution mainly reflected 0 changepoints as exponential distribution can better identify changepoints for datasets over a period greater than 2 years.

### Predictive analysis

To predict future sales, we used the Prophet library, a procedure for forecasting time-series data. Turnover ratio (sales divided by inventory) was calculated as well, which aided in cross-checking with forecast demand patterns observed to ensure coherence and further justify the results for our predictive analysis. Generally, high and low turnover ratios corresponded to the increasing and decreasing forecast trends respectively. Corelating it to the changepoint locations and type of change point observed, it implies that either inventory was mostly sold and cleared due to popular demand or specific unpopular products in specific locations caused inventory levels to increase leading to decrease in sales.

### Conclusion

To conclude, our suggestions are to increase inventory for products at locations with increasing forecasted trend to meet future demand and maximise potential revenue. Likewise, for products at locations with decreasing forecasted trend, inventory should be decreased to minimise losses due to possible oversupply, thereby preventing wastage of financial resources.

# Report

## 1.2 Company introduction

Integrated Design Systems Consultancy (IDSC) specialises in the provision of advanced analytics insights to business decision makers through consultancy and other augmented intelligence services. Their industry expertise are in healthcare, public sector, manufacturing and retail and distribution. Their mission is to harness the power of decision technology such as Augmented Intelligence, Advanced Analytics and AI to help their clients distil their corporate wisdom into explicit models that make intelligent decisions consistently and coherently, promote better communication, learn and accumulate knowledge as a corporate body over time. Ultimately helping their clients gain and maintain a competitive edge.

## 1.3 Problem Definition and Motivation

The primary challenge of any supermarket is to be able to provide enough products of a certain freshness to meet demand. However, as the demand is uncertain, it is difficult for a supermarket to balance between understocking and over-stocking, which may result in lost sales and wastages. Especially when different implemented sales strategies, like promotions, can affect demand substantially. Hence, our project title is to detect changes in customer purchase patterns through Change Point Analysis. Our objective is to analyse customer behaviour patterns and use Change Point Analysis to forecast demand. An accurate source code is to be produced in the end as IDSC requires the right time series to analyse how customer behaviour changes over time and forecast demand. The scope of the

project is to determine the times when purchased patterns have changed, duration of change by products and demand patterns and predictability of the sales data.

### 1.4.1 Methodology: Pre-analysis

For our pre-analysis of the data, we first used parallel coordinates to get an overview of how each column of data that the client has given to us is related. Fig 1.1 is an example of how we used parallel coordinates to visualize the relationships between the different columns of data given by client, for all the products at location 5.

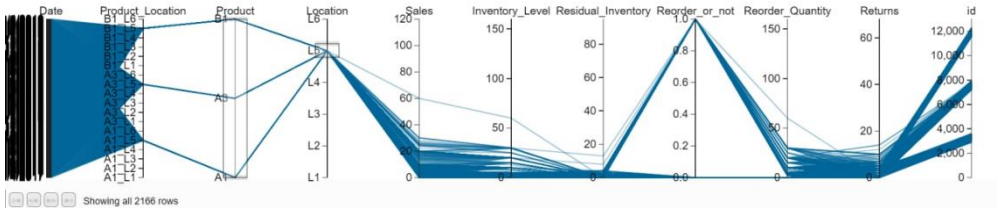



Fig 1.1

Next, Microsoft Excel and SQL were used for data wrangling, to filter out only those columns of data that we deem will be useful for our analysis, as shown in Fig 1.2.

| Product_Location | Product | Location | Date | Sales | Inventory_Level | Residual_Inventory | Reorder_or_not | Reorder_Quantity | Returns |
|---|---|---|---|---|---|---|---|---|---|
| A1_L1 | A1 | L1 | 10/4/2017 | 5 | 6 | 0 | 1 | 6 | 1 |
| A1_L1 | A1 | L1 | 11/4/2017 | 4 | 4 | 0 | 1 | 4 | 0 |
| A1_L1 | A1 | L1 | 12/4/2017 | 4 | 4 | 0 | 1 | 4 | 0 |
| A1_L1 | A1 | L1 | 13/4/2017 | 2 | 4 | 0 | 1 | 4 | 2 |

| Date | Sales |
|---|---|
| 10/4/2017 | 5 |
| 11/4/2017 | 4 |
| 12/4/2017 | 4 |
| 13/4/2017 | 2 |

Fig 1.2

### 1.4.2 Methodology: Changepoint Analysis

We used the 'changepoint' package that was available in R to conduct changepoint analysis, the complete R code that we have produced for analysis can be found in the appendix.

A changepoint (cpt) is an instance in time where the statistical properties before and after this time point differ. There are a few goals that we wish to achieve through cpt analysis:

Firstly, given the Sales vs Date data, we check if a change has occurred. If change(s) has occurred, we identify the number of cpt(s) detected and the date(s) in which these change(s) has occurred. Next, we identify the differences between the pre- and post-change data, mainly if sales have decreased or increased and provide the parameter estimates. Finally, we identify patterns of cpts detected across different product/ locations.

We performed cpt analysis on each product at each location, using R's 'Changepoint' package. A popular approach for cpt analysis on R is to use penalised optimization which requires a choice of a penalty value.

The choice of penalty can be done through multiple ways, one way is to manually set penalty value and perform tuning to find the most optimal number of cpts. Another way is to use penalty functions on R that can calculate the optimal penalty values for us.

We explored 3 different ways to identify cpts using R:

1. Identifying cpts through **changes in mean**, using Pruned Exact Linear Time (PELT) method.
2. Identifying cpts through **changes in mean and variance**, we used the Binary Segmentation (Bin Seg) method, penalty function: SIC. This assumes a Poisson Distribution.

3. Identifying cpts through **changes in mean and variance**, we used the Binary Segmentation (Bin Seg) method, penalty function: SIC. This assumes an Exponential Distribution.

***Method 1: Identifying changepoints through changes in mean, using PELT method.***

PELT: The PELT method uses a common approach of identifying cpts through minimising a cost function over possible numbers and locations of cpts. (Gachomo Dorcas Wambui, 2015)

**Our general approach at using method 1:**

Step 1: Method 1 is used to find a suitable penalty value. We plotted a graph of number of cpts VS penalty value and picked a initial penalty value where the graph becomes relatively constant. From Fig 1.3, we set our initial penalty value = 60. Fig 1.4 shows the 6 cpts identified when penalty value = 60 for product A1 at Location 2.

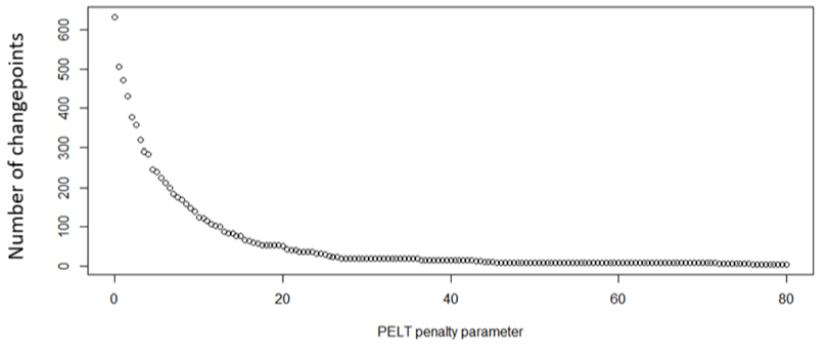



Fig 1.3 (no. of cpts VS penalty value graph for product A1 at location 1)

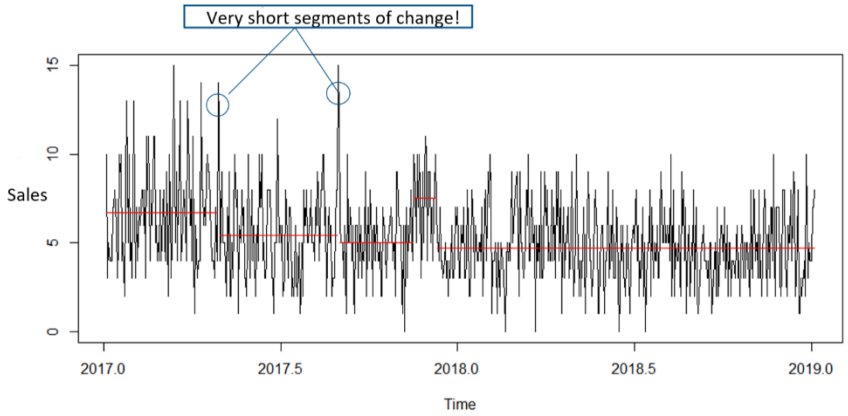



Fig 1.4

Step 2: Tune the sensitivity of the algorithm by manually adjusting the penalty value. By adjusting the penalty value and visual verification, we are able to identify the optimal cpts. In fig 1.5, notice how the 2 short segments from Fig1.3 are eliminated because we decreased the sensitivity of our algorithm by increasing penalty value to 80, leaving us with 3 optimal cpts.

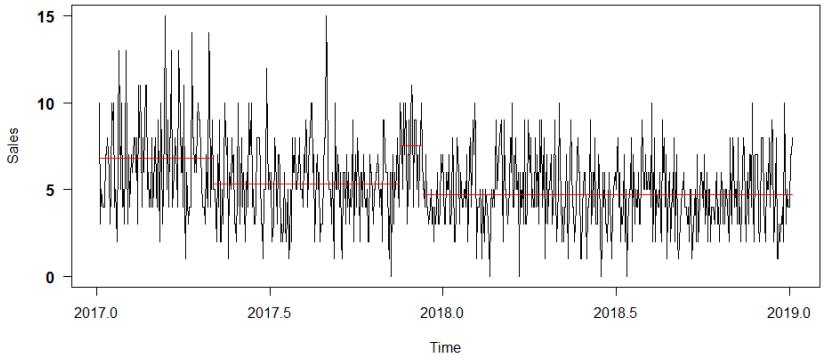



Fig 1.5

However, this general approach to eliminate short segments of cpts is not very effective and efficient because it is time-consuming (requires manual tuning). Also, for some products/location graphs, very short segments of cpts cannot be eliminated due to the sudden spike of data points, despite decreasing the sensitivity of our algorithm.

Hence, detecting cpts through the use of changes in mean is not very accurate as our dataset is largely scattered. This brings us to method 2, to detect cpts through changes in both mean and variance.

***Method 2: Identifying changepoints through changes in both mean and variance, using Bin Seg method, with penalty term SIC and assuming Poisson distribution.***

Bin Seg method: The idea of Bin Seg method is that if a single cpt can be detected in a time-series, then the series can be split around this cpt into two sub-series of either side of the change, then single cpt detection can be then applied to each of the sub-series. This can be

iterated until no further cpt are detected. This recursive approach is one of the most widely used cpt detection methods due to its simplicity and good accuracy. (Maidstone, 2016)

SIC penalty term: The Schwarz information criterion (SIC) is a criterion for model selection among a finite set of models. (Mohamad, 2016) The SIC penalty term is used to determine the model that gives optimal segmentations. SIC outperforms other penalty terms like Akaike's Information Criterion (AIC) and Hannah-Quinn as it is proved to give the most accurate segmentations. (Kaylea Haynes, 2014 )

Poisson Distribution: Assuming Poisson distribution, our parameter estimates for our pre-post change in data is lambda (λ).

***Method 3: Identifying changepoints through changes in both mean and variance, using BinSeg method, with penalty term SIC and assuming Exponential distribution***

We applied the same approach as mentioned in Method 2 previously, but modelled with exponential distribution. However, we found that the algorithm tends to give 0 cpt in most of the locations for each product. The main results for our exponential can be found in the appendix.

We noticed that since exponential distribution is used to simulate the time interval for independent events, it has a memoryless trait. This means that it identifies similar patterns for different time intervals (such as yearly or monthly seasonality). However, these traits are not obvious in our data (due to the short period the data sets include from 2017 to 2019). Also, the majority only shows a monotonous trend. As a result, we mainly draw our conclusion from Method 2 instead of other approaches.

## 1.5 Main results on Change Point Analysis:

As we draw our conclusions from Method 2, Fig 1.6 shows an example of how we present our findings for Product A1 at Location 1 to our client. Our main findings for the remaining 17 graphs will be displayed in the appendix.

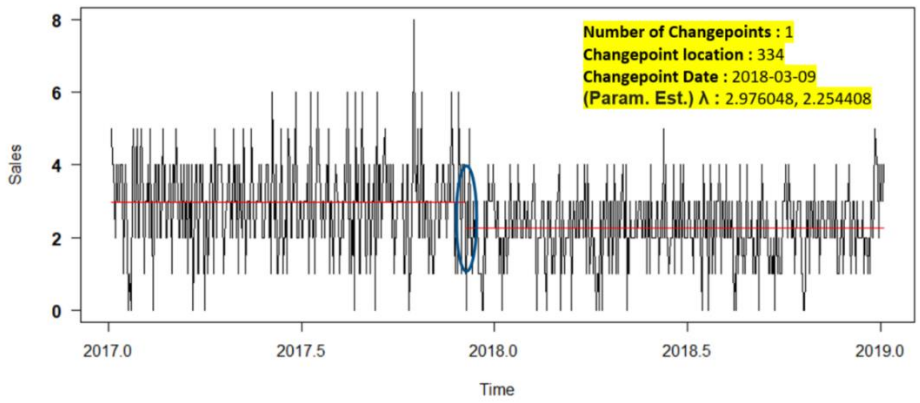



Fig 1.6

### 1.5.1 Main observations across different Products and Locations

With reference to Fig 1.7, our main observations identified from our results are:

1. There is a common cpt corresponding to the date 13 August 2018 for most of the products in Location 1. Sales generally decreased after that cpt, as indicated by the lowered red line on the graphs after the cpt.
2. No cpts were detected from our algorithm for all products in Location 5.
3. There is a common cpt corresponding to date 12 December 2017 for all the products in Location 6. Sales generally increased after that cpt as indicated by the raised red line after the cpt.

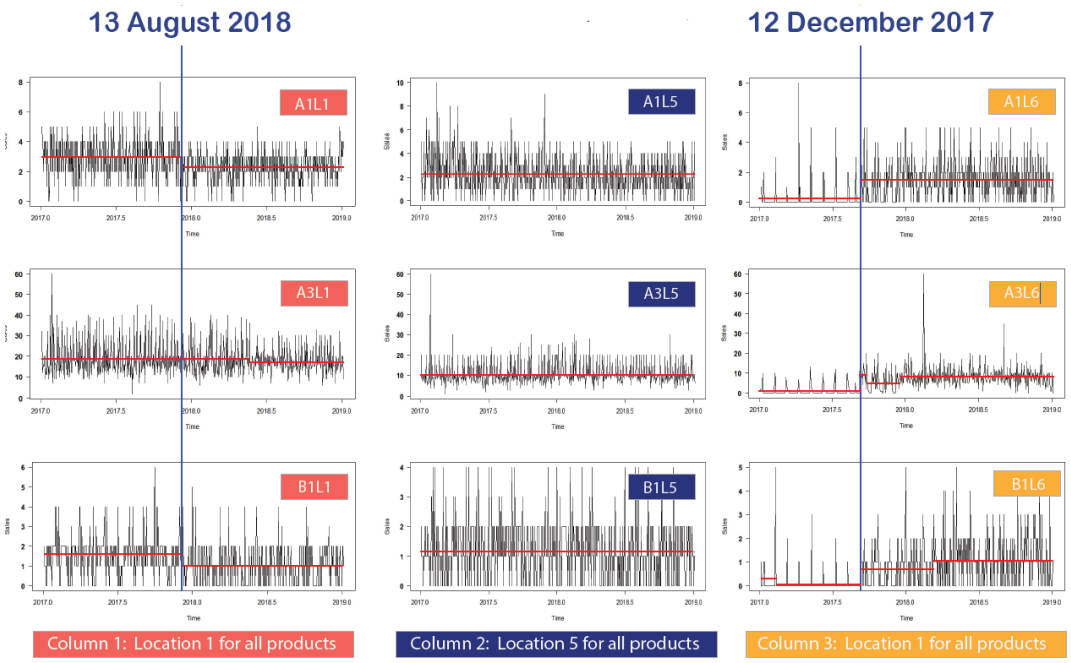



Fig 1.7

## 1.6 Methodology: Predictive analysis and Forecasting

We decided proceeded further and used the results from cpt analysis to predict the future trends of the products' sales. We have found that the prophet library is an effective tool for forecasting time series data. It is based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality. After simple test cases, we find out that Prophet works accurately and quickly in R.

## 1.7 Main Results on Predictive analysis and Forecasting

The majority display 2 distinct trends. First, sales decreased then increased in Year 2018 and second, sales having a consistent decreasing trend.

We utilised other data columns given to us and calculated the average turnover ratio (sales/inventory) for each product and location. Most of them fit well with the expectations we have in mind: a product with high turnover ratio (i.e. most of the bread in inventory is

sold daily) would likely show an increasing trend. Consequently, a product with low turnover ratio would likely show a decreasing trend. Although, there are some results that do not correspond accordingly, we are unable to validate it due to lack of information at this point in time.

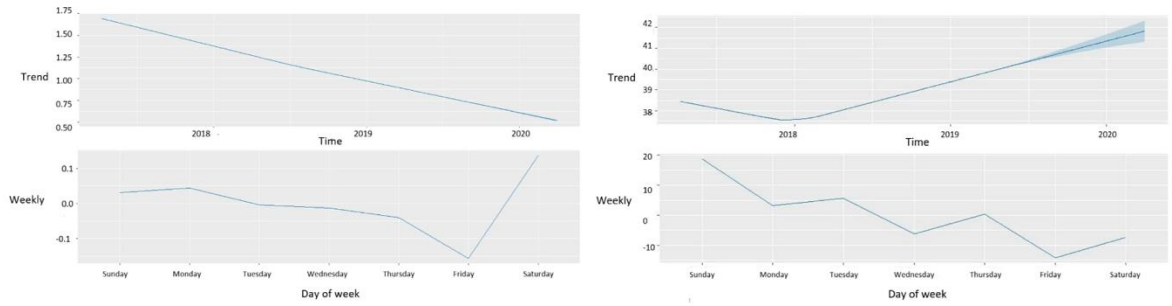



Fig 1.8: Products B1L1 on the left (Turnover ratio:23%) &

A3L3 on the right (Turnover ratio:83%)

**Assumption and Limitations:**

We assumed that the data set given would be sufficient to account for different seasonal trends such that the Prophet library can generate accurate forecasts for each product. A limitations of the project was that were unable to draw a full conclusion after identifying critical change point locations due to lack of information regarding potential factors affecting sales. Additionally, an improvement of results could be obtained if we had a specific range of penalty values the client is looking to further increase the accuracy of our results.

## 1.9 Appendix

***Entire Source Code*:**

```
library(changepoint)
library(prophet)
library(readr)
library("readxl")

daily<-read_excel("A1_L1.xlsx")
b<-ts(daily$Sales,start=c(2017,4),end=c(2019,4),frequency=365)
plot.ts(b)
```

**(changes in mean code)**

```
cptfn <- function(data, pen) {
 b4 <- cpt.mean(data, test.stat="Normal", method = "PELT", penalty = "Manual", pen.value = pen)
 length(cpts(b4)) +1
}

pen.vals <- seq(0, 30,.2)
elbowplotData <- unlist(lapply(pen.vals, function(p)
cptfn(data = b, pen = p)))
plot(pen.vals,elbowplotData,      # plot the number of cpts VS penalty value graph
 xlab = "PELT penalty value",
ylab = " ",
main = " ")

b4=cpt.mean(b,method='PELT',penalty="Manual", pen.value = 80)  # adjust pen.value to adjust sensitivity
ncpts(b4)    # number of changepoints identified
```

```
plot(b4,ncpt(6),yaxt="none",xlab="Time",ylab="Sales")

axis(2, las=2, font=2,cex.axis=1.1)

date = daily$Date

date[position]    # position of cpts identified

summary(b4)

param.est(b4)     # pre post changes in mean value
```

**(Exponential code, meanvar)**

```
b2=cpt.meanvar(b,test.stat='Exponential',method='BinSeg',Q=5,penalty="SIC")

ncpts(b2) #number of cpts detected from exponential test

plot(b2,cpt.width=1,cpt.col='red',yaxt="none",xlab="Time",ylab="Sales") #plot the graph
with the number of cpts

axis(2, las=2, font=2,cex.axis=1.1)

position = cpts(b2)

position          #position of cpts detected

param.est(b2)       # rates values pre post change

summary(b2)

date = daily$Date

date[position]      #returns the dates of changepoints
```

**(Poisson code, meanvar)**

```
b3=cpt.meanvar(b,test.stat='Poisson', method='BinSeg',Q=5, penalty = 'SIC')

ncpts(b3)        #number of cpts detected from poisson test

plot(b3,ncpt(5) ,yaxt="none",xlab="Time",ylab="Sales")  #plot the graph with the number of
cpts indicated (<=# cpts detected)

axis(2, las=2, font=2,cex.axis=1.1)

position = cpts(b3)

position        # position of changepoints

param.est(b3)    # lambda values pre post change

summary(b3)
```

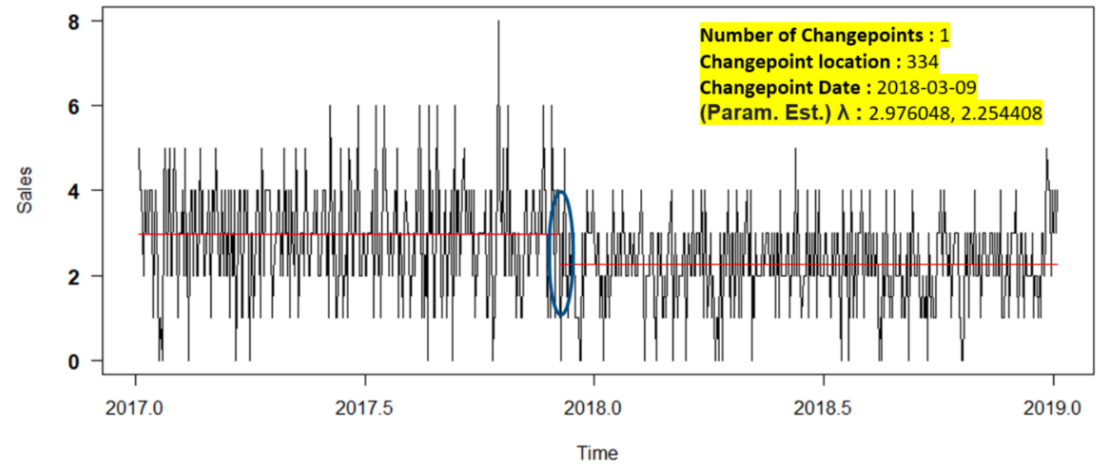
```
date = daily$Date
date[position]   #Returns the dates of changepoint
```

**(prophet library)**

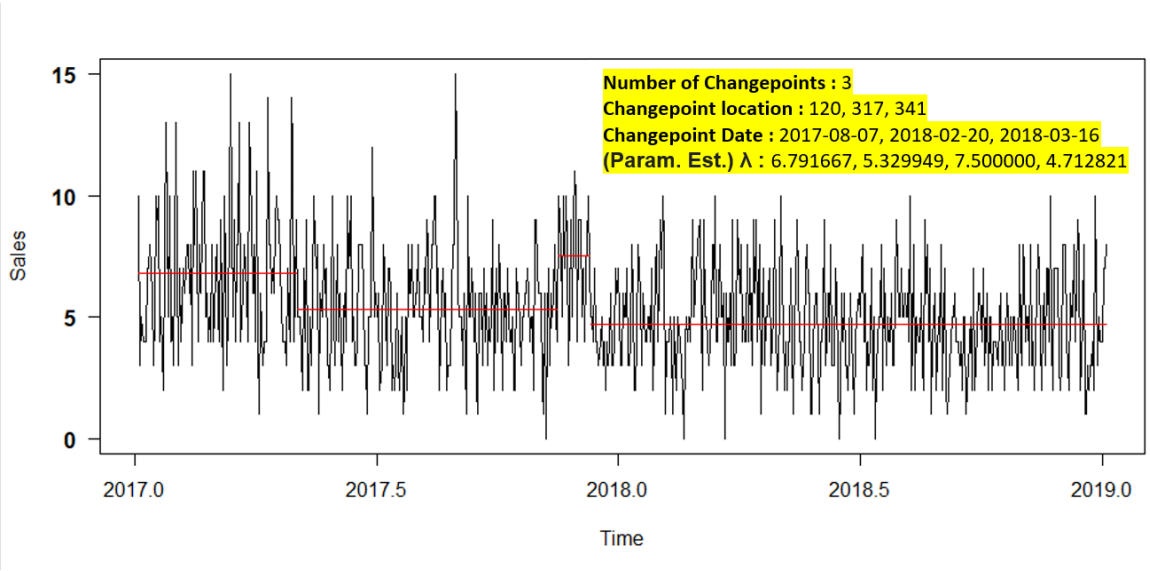
```
df<-read_csv("test.csv"
)
m<-prophet(df)
future <- make_future_dataframe(m, periods = 365)
tail(future)
forecast <- predict(m, future)
tail(forecast[c('ds', 'yhat', 'yhat_lower', 'yhat_upper')])

plot(m, forecast)
prophet_plot_components(m, forecast)
```

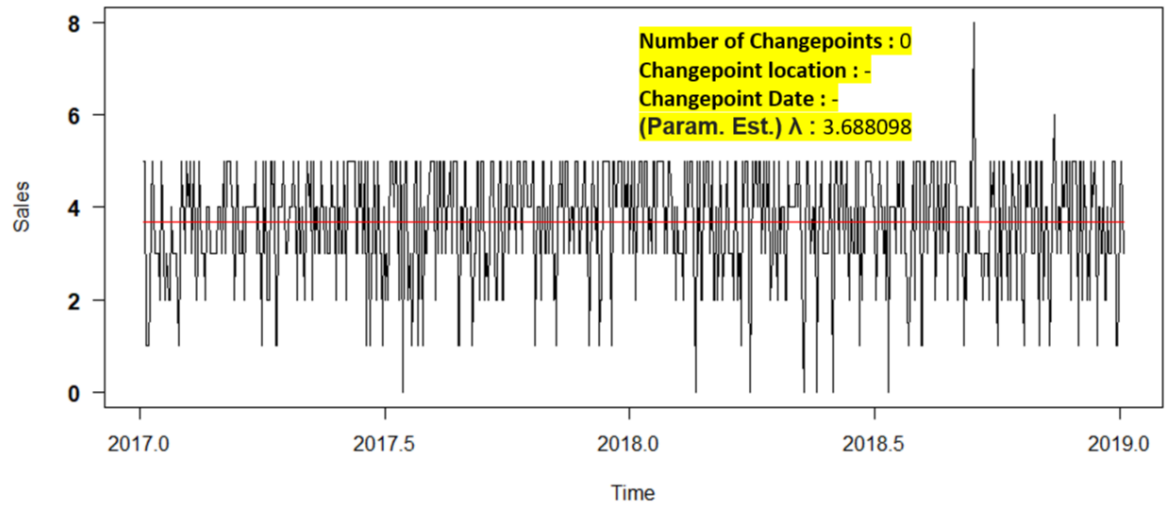
**(Graphs generated using Poisson distribution)**



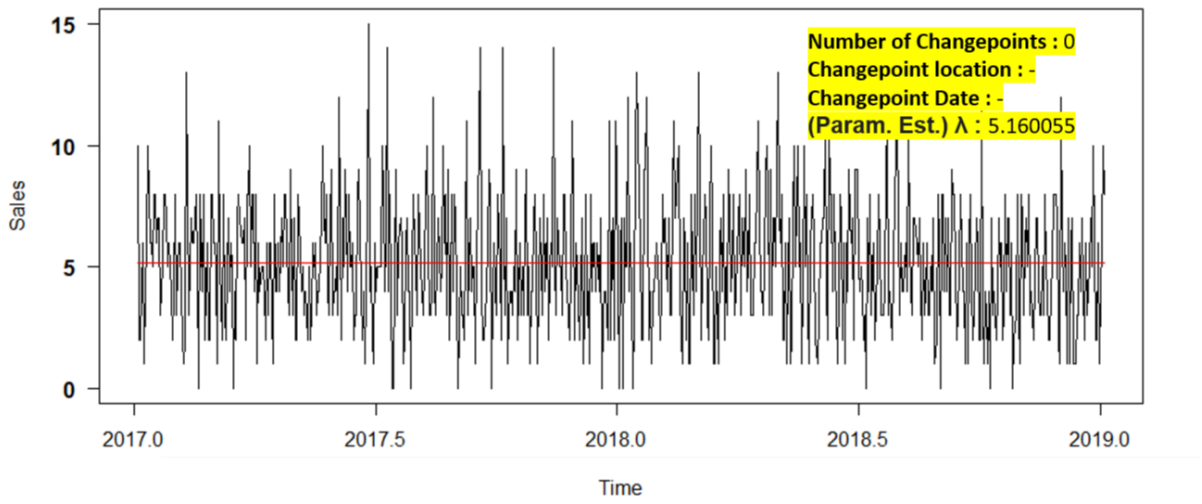

Product A1L1



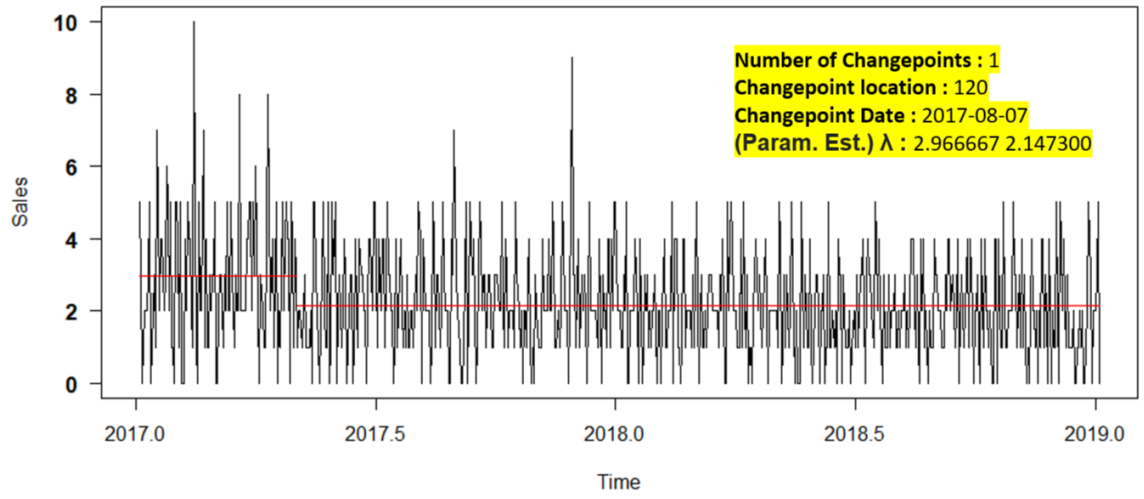

Product A1L2(Poisson)



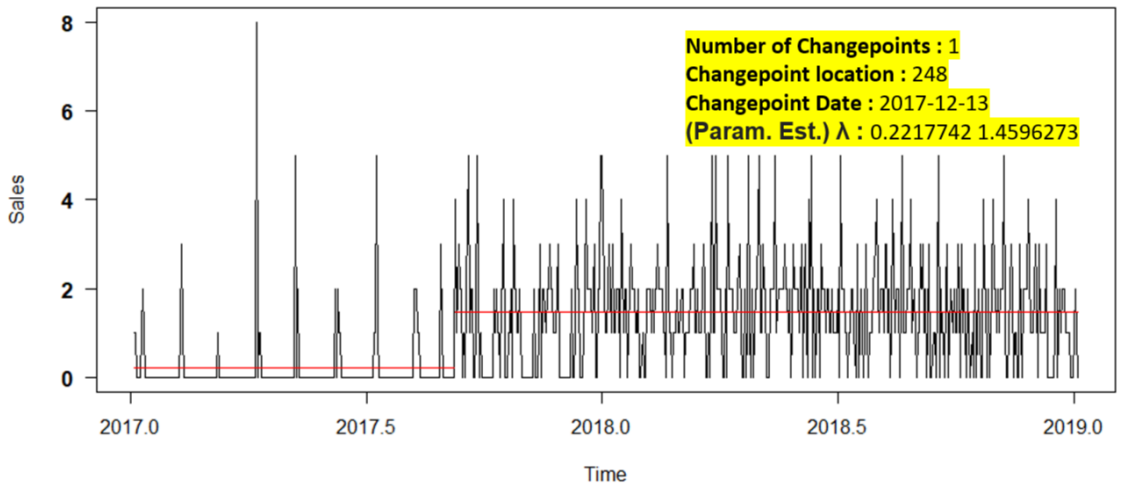

Product A1L3



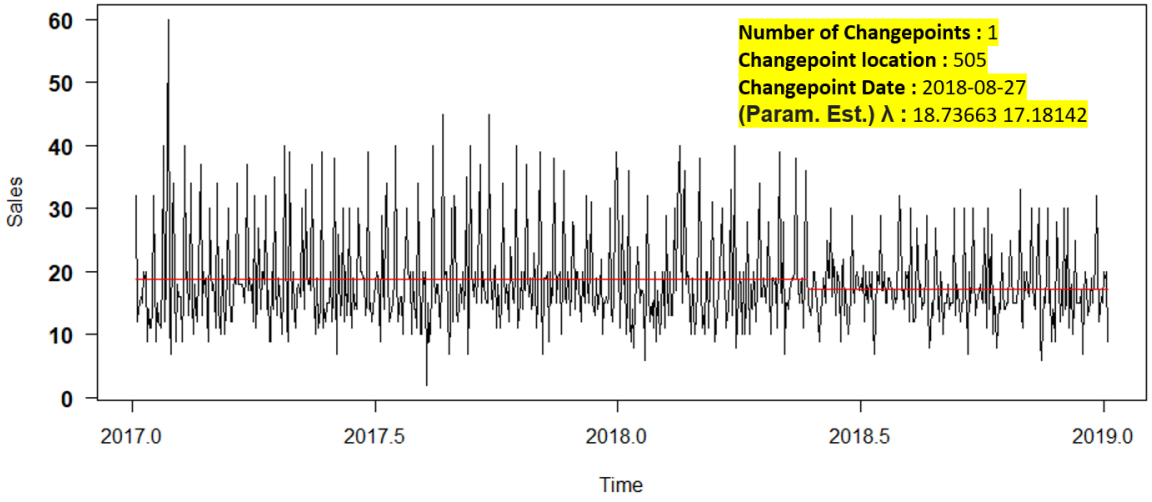

Product A1L4



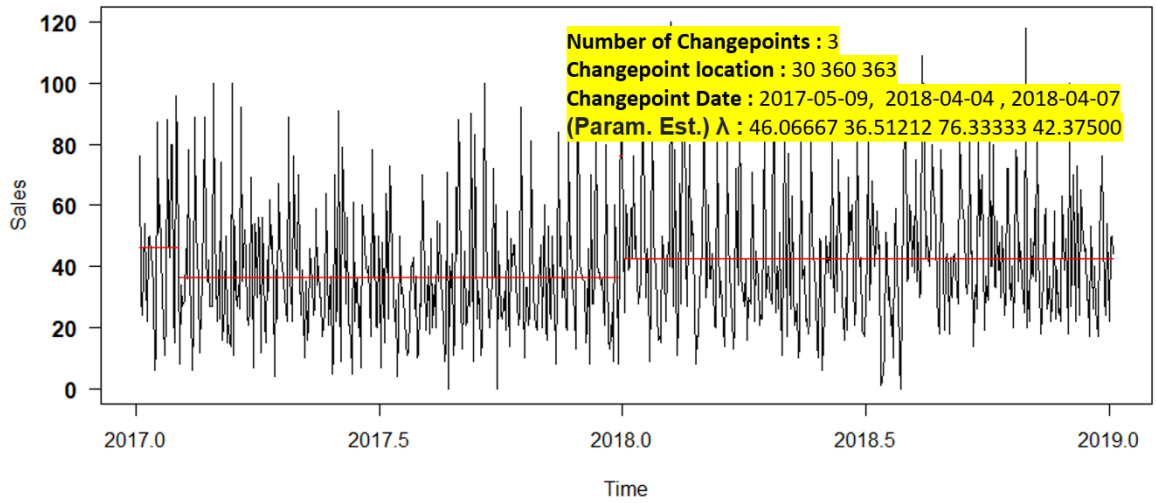

Product A1L5



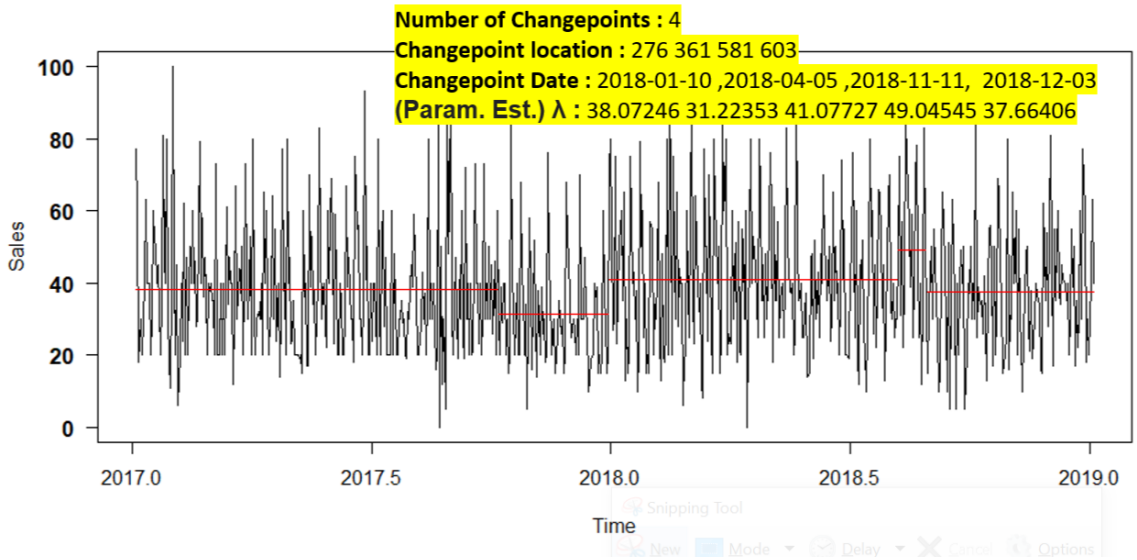

Product A1L6



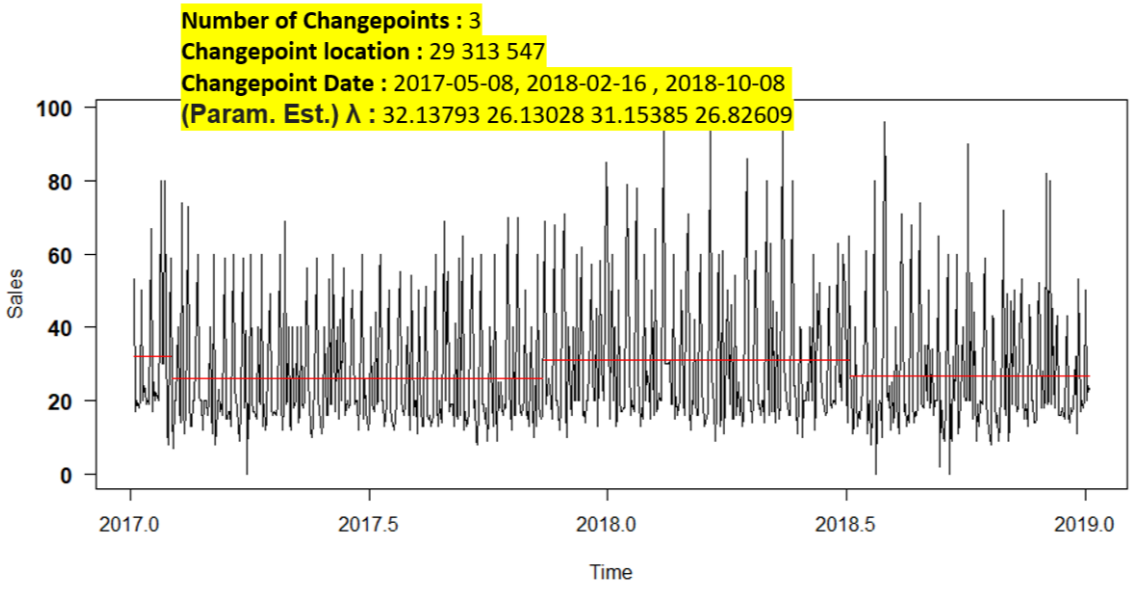
Product A3L1



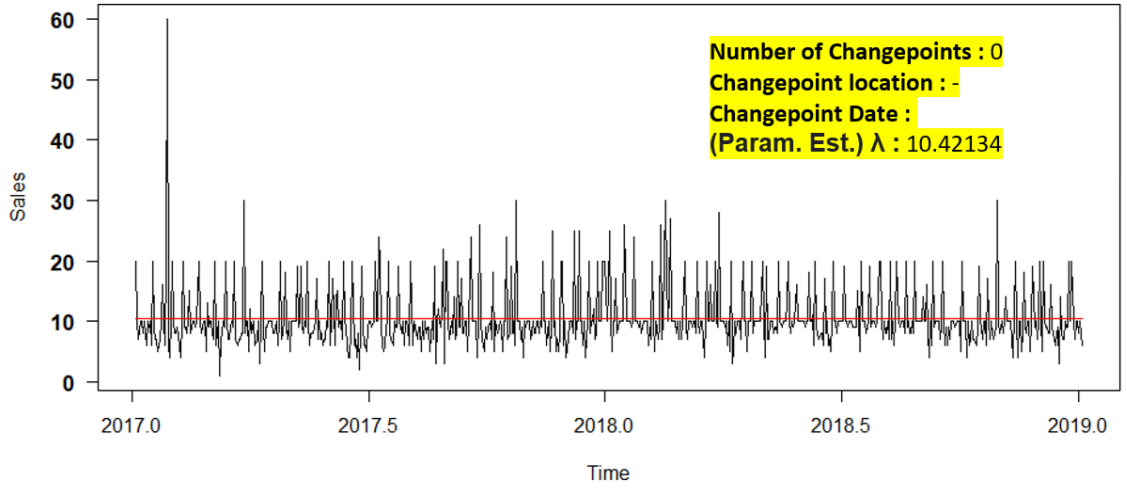

Product A3L2



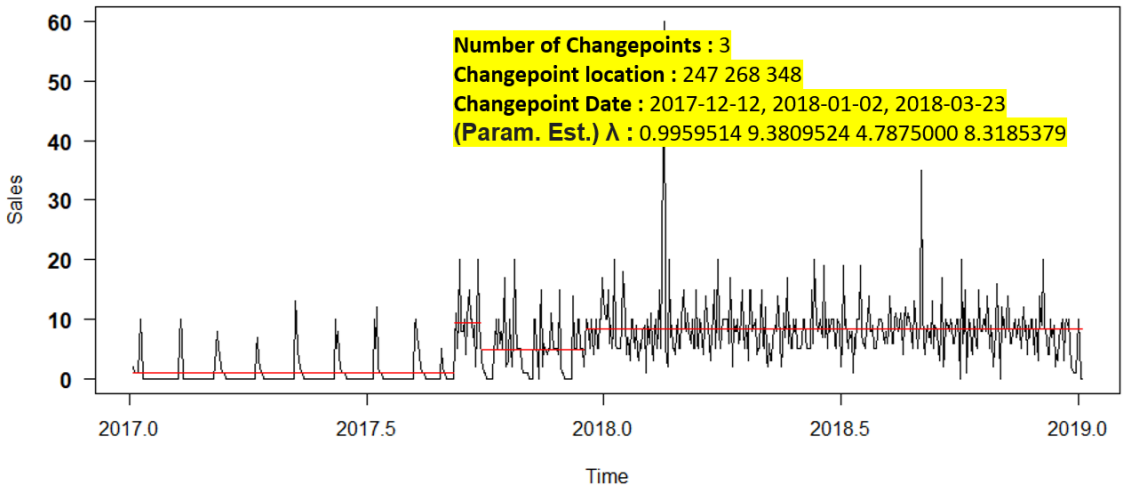

Product A3L3



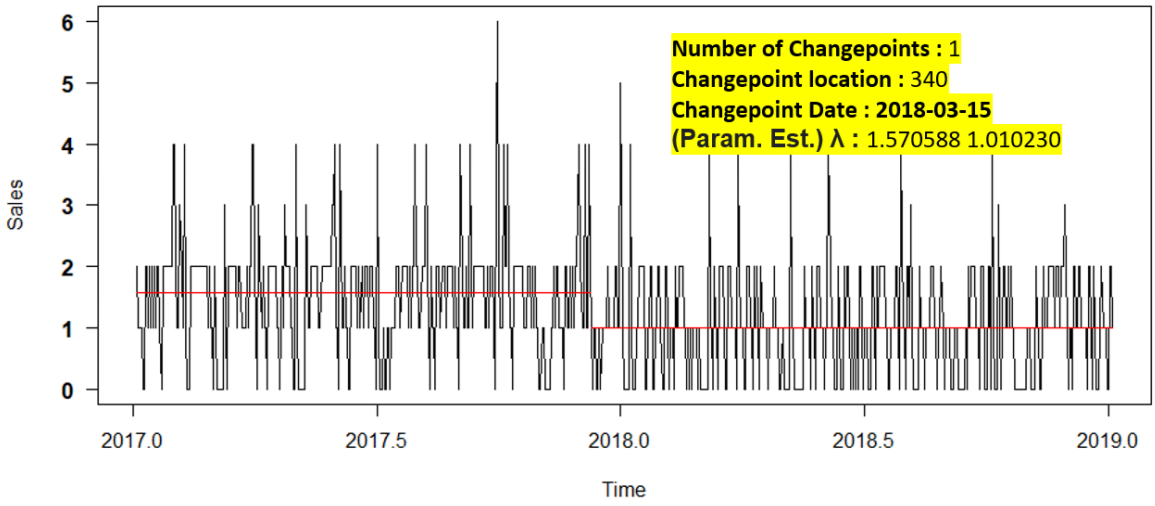

Product A3L4



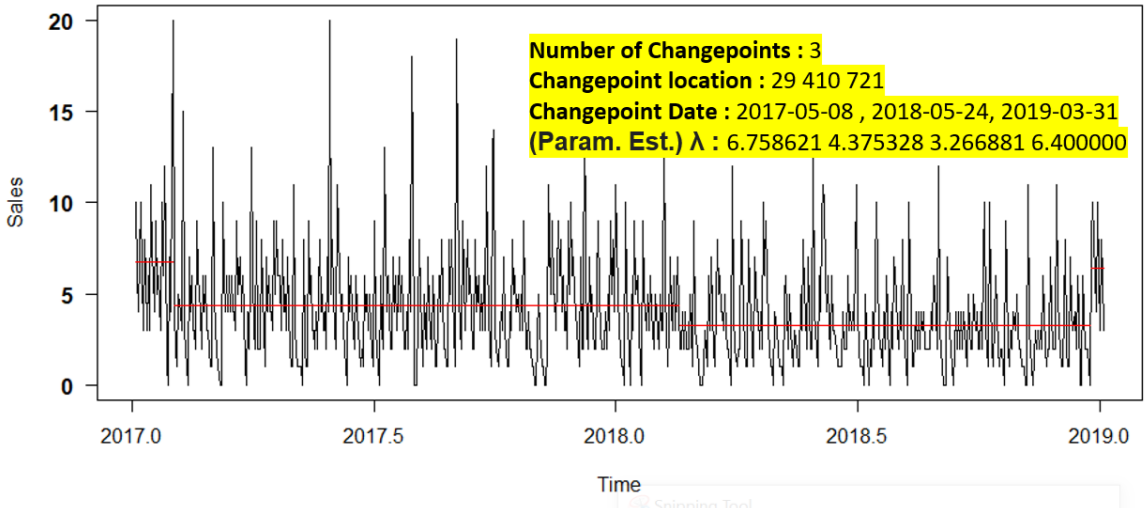

Product A3L5



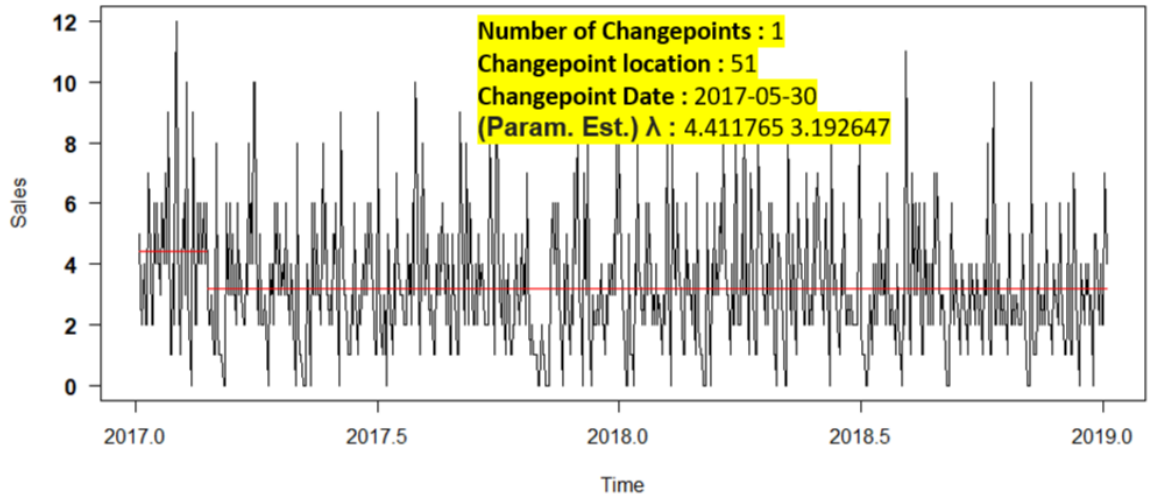

Product A3L6



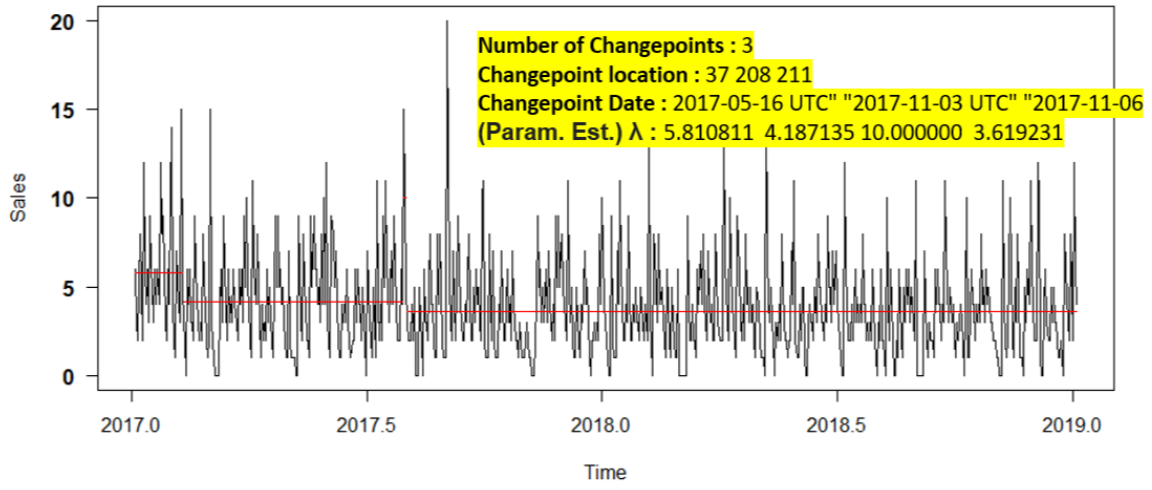

Product B1L1



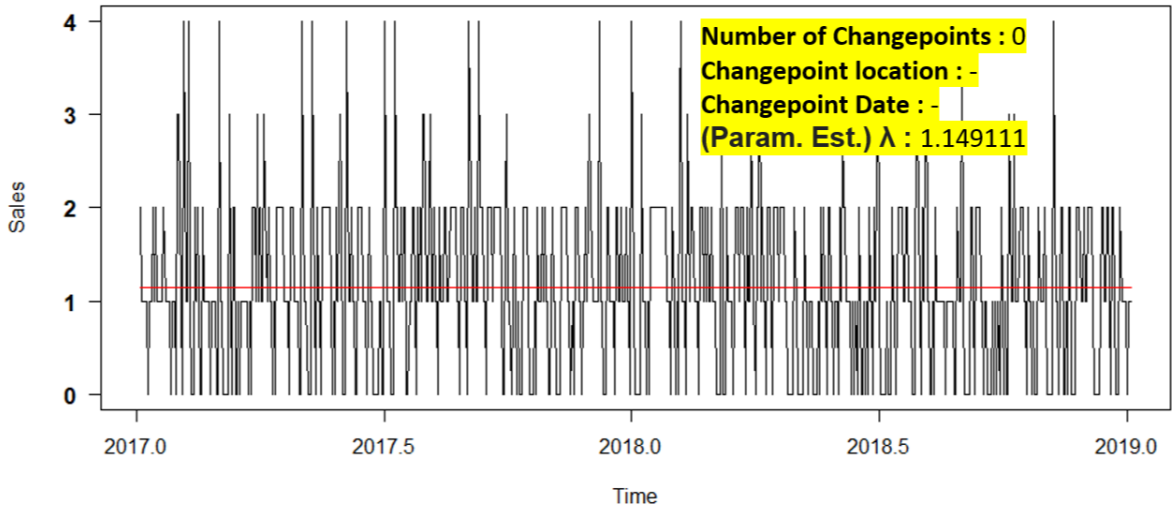
Product B1L2



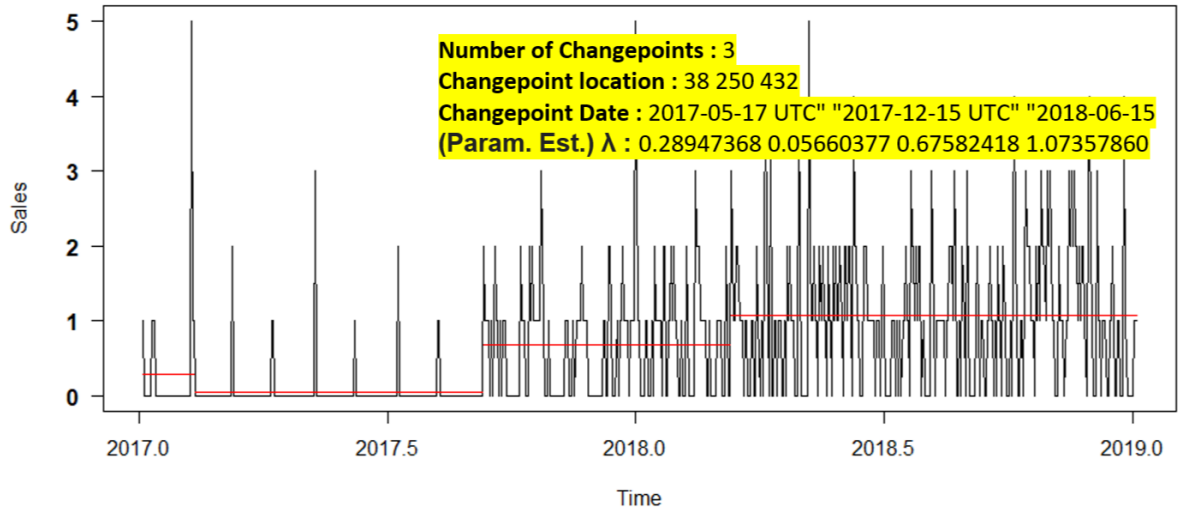

Product B1L3




Product B1L4



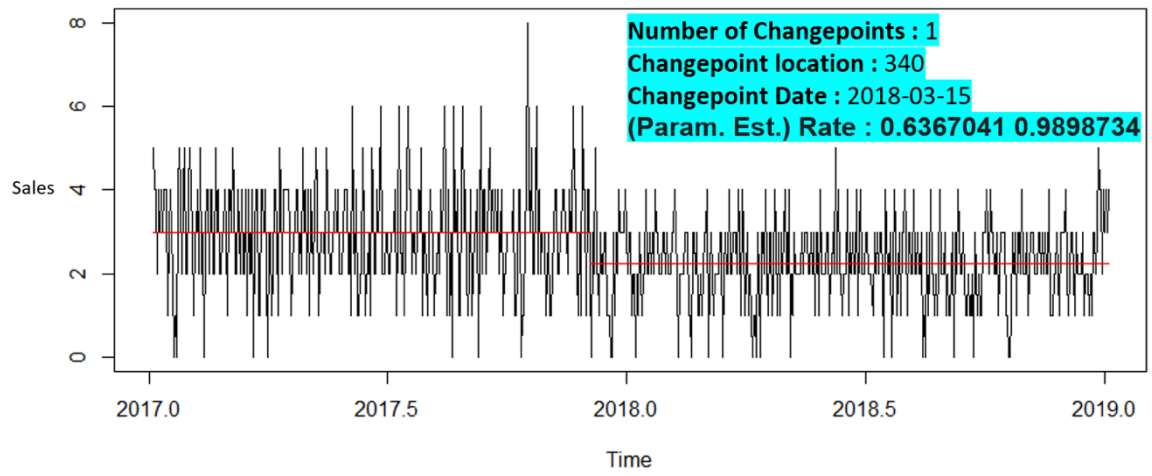

Product B1L5



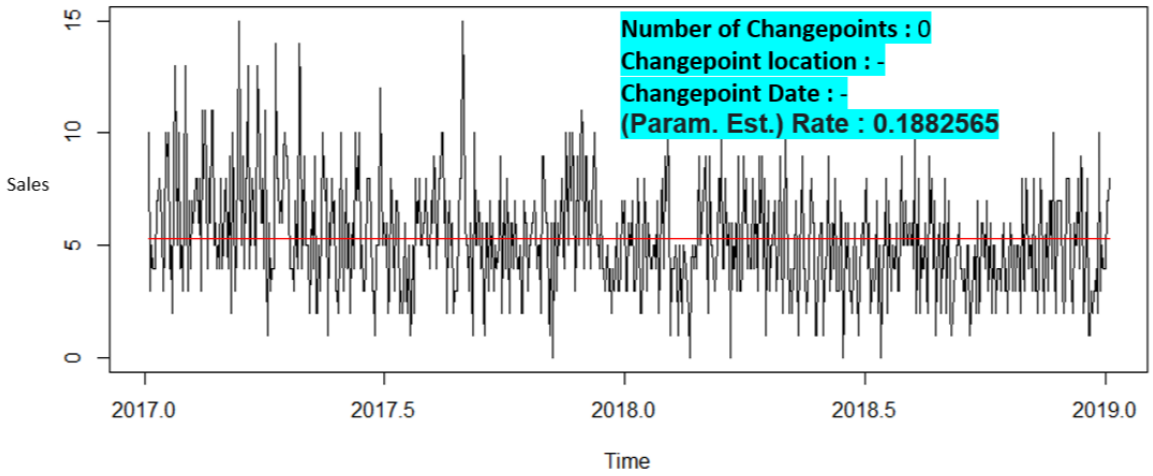

Product B1L6

**(Graphs generated using Exponential distribution)**

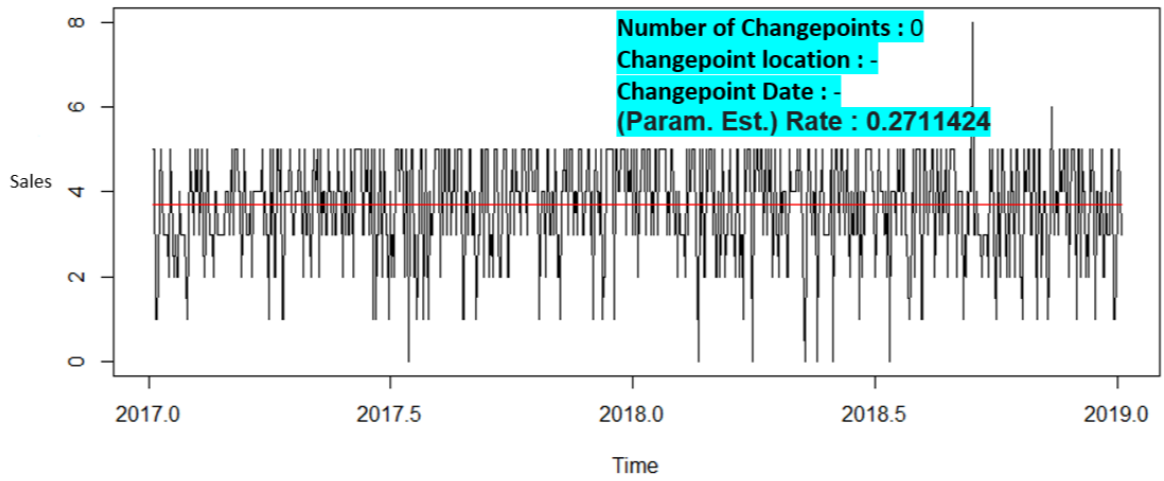
**Code:**



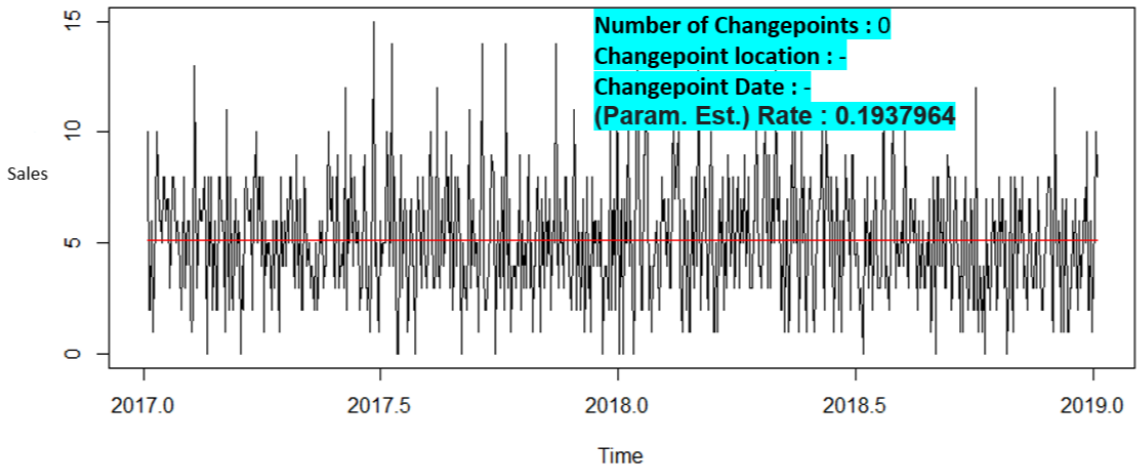

Product A1L1



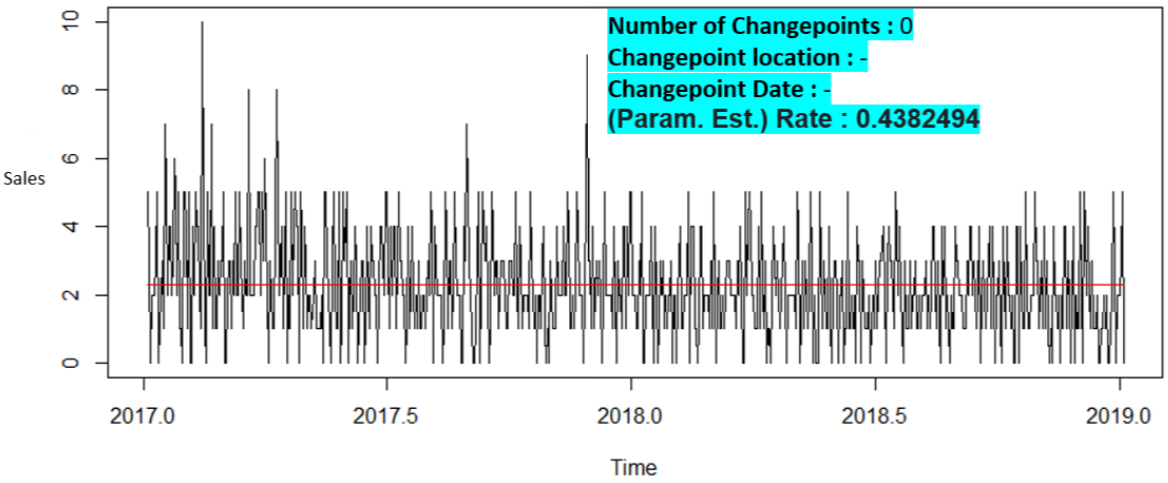

Product A1L2



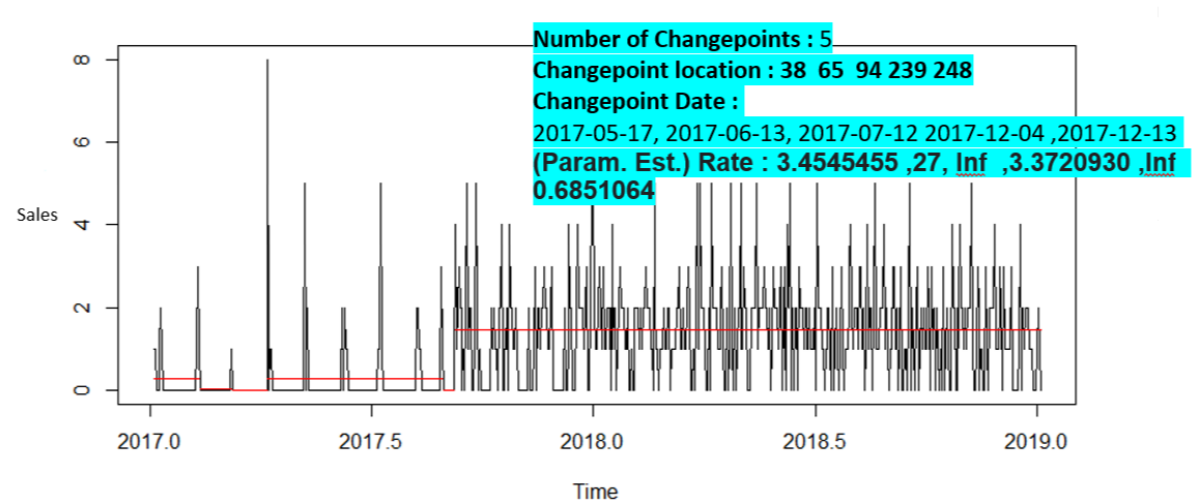

Product A1L3



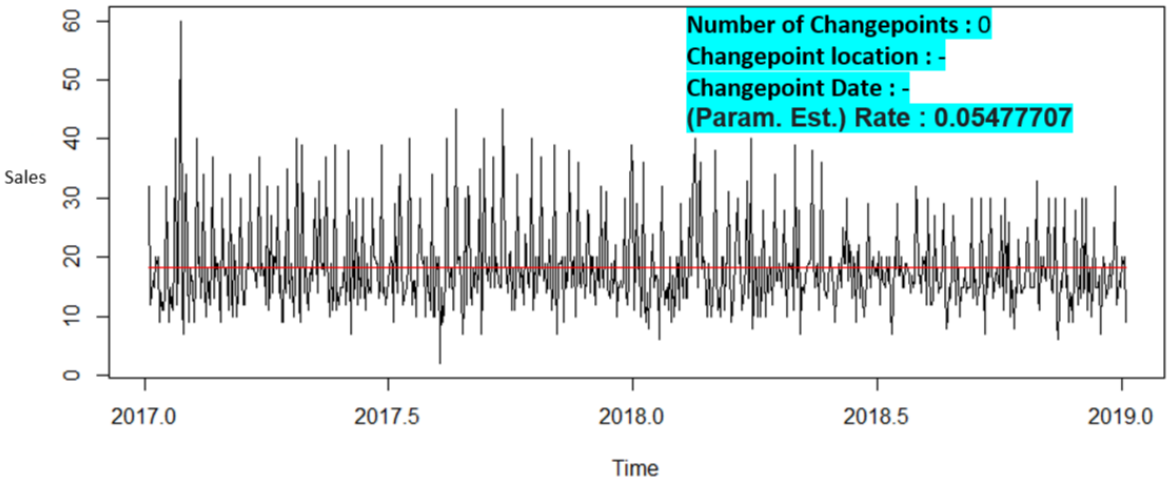

Product A1L4



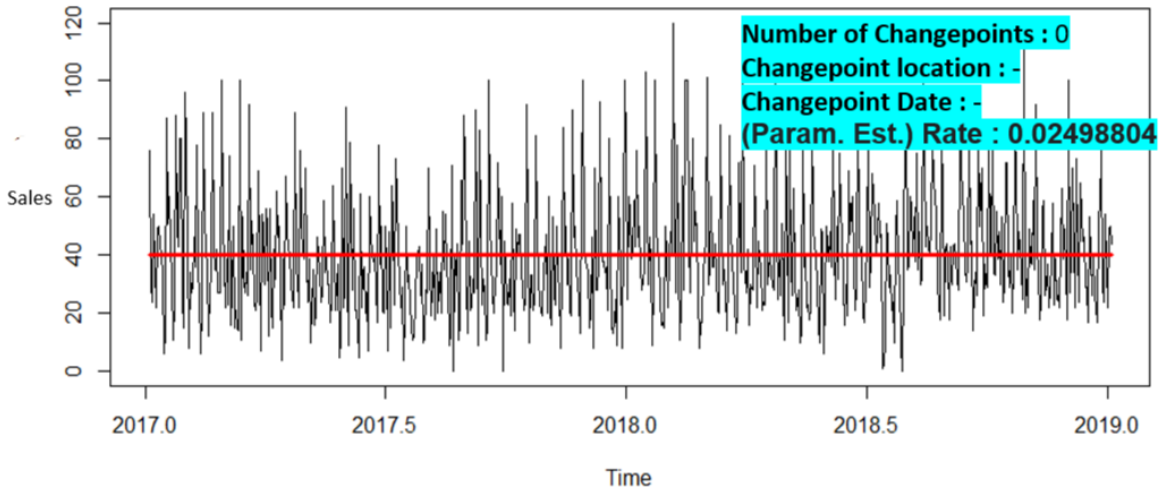

Product A1L5



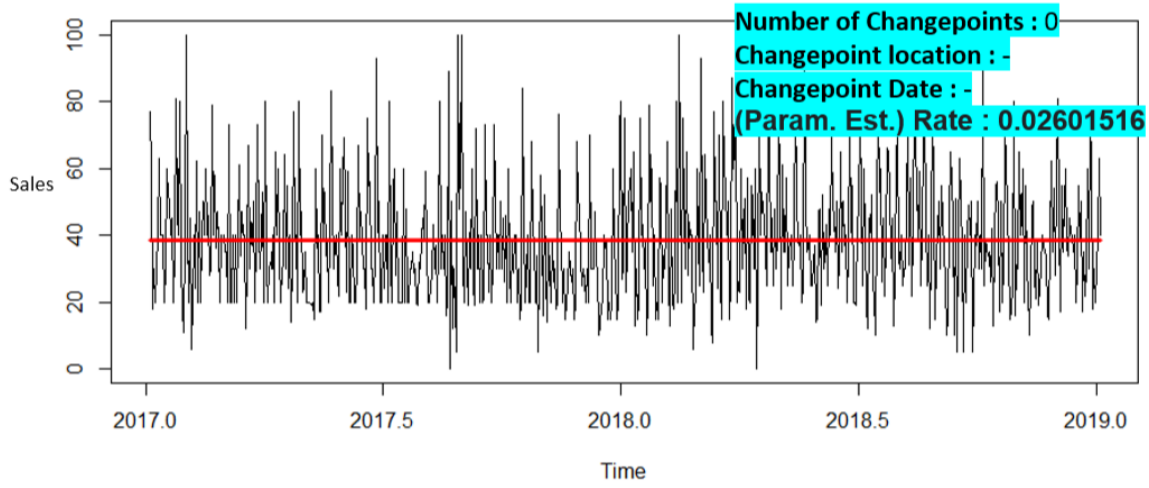

Product A1L6



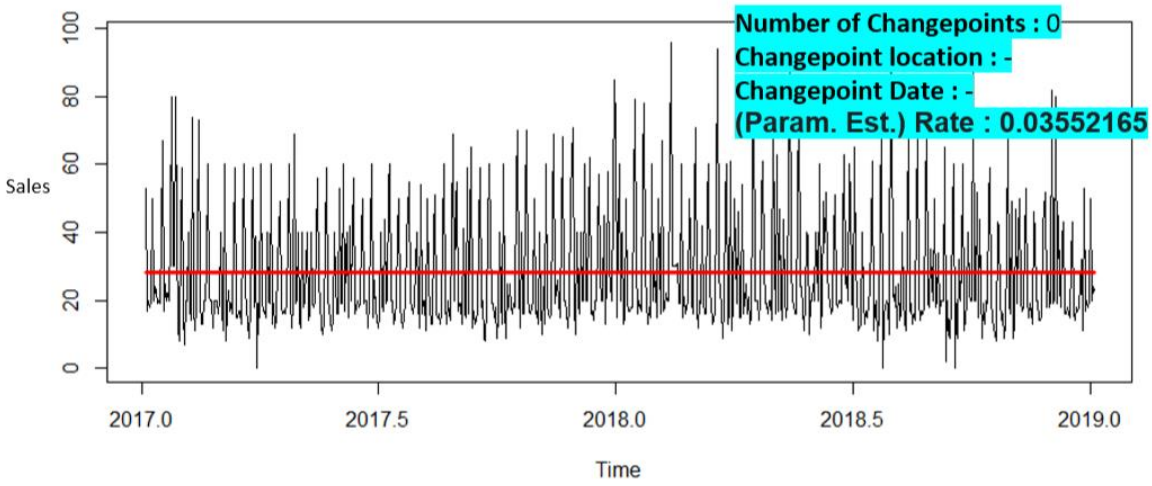

Product A3L1



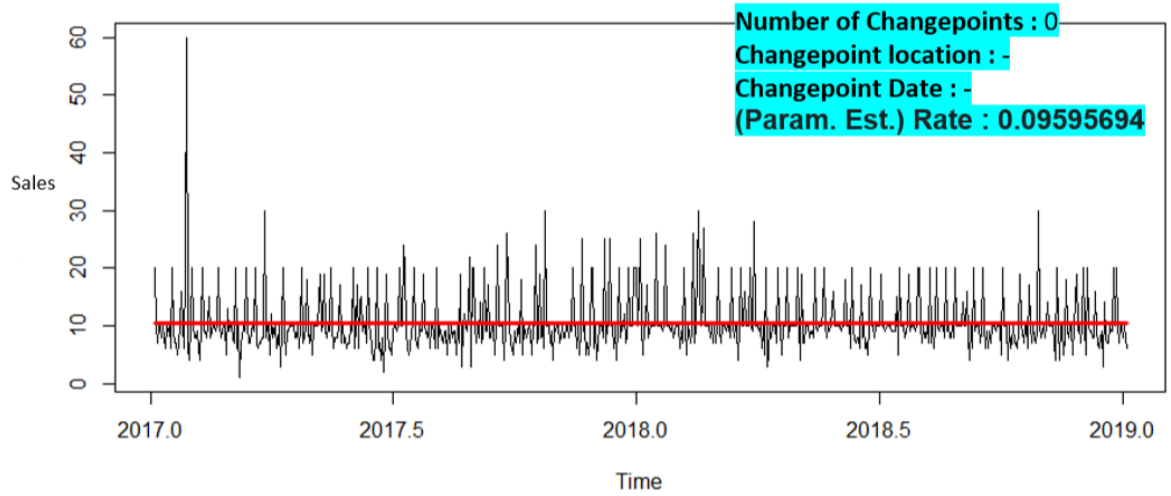

Product A3L2



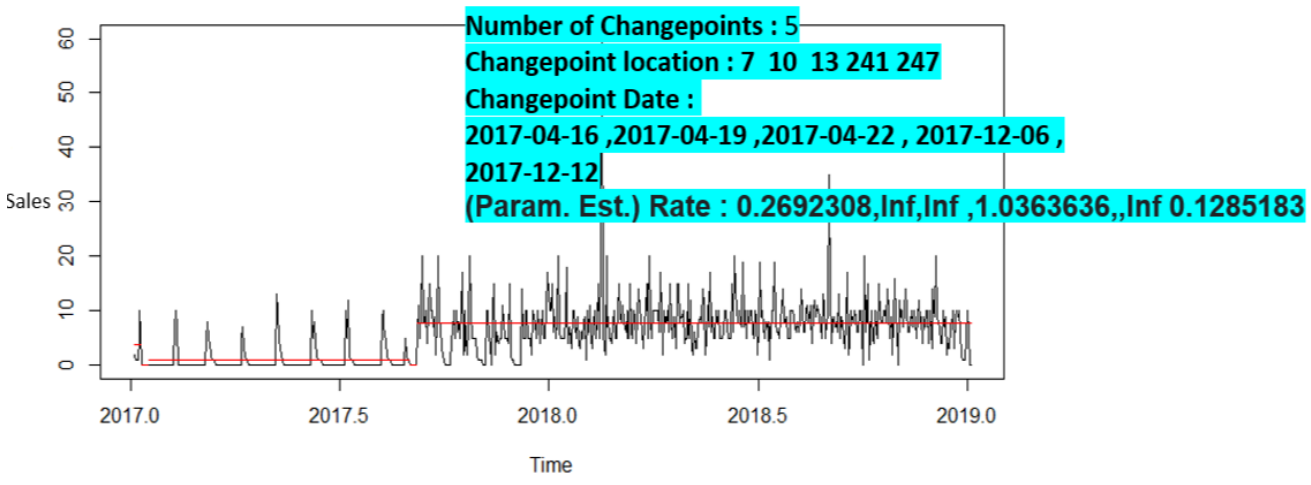
Product A3L3



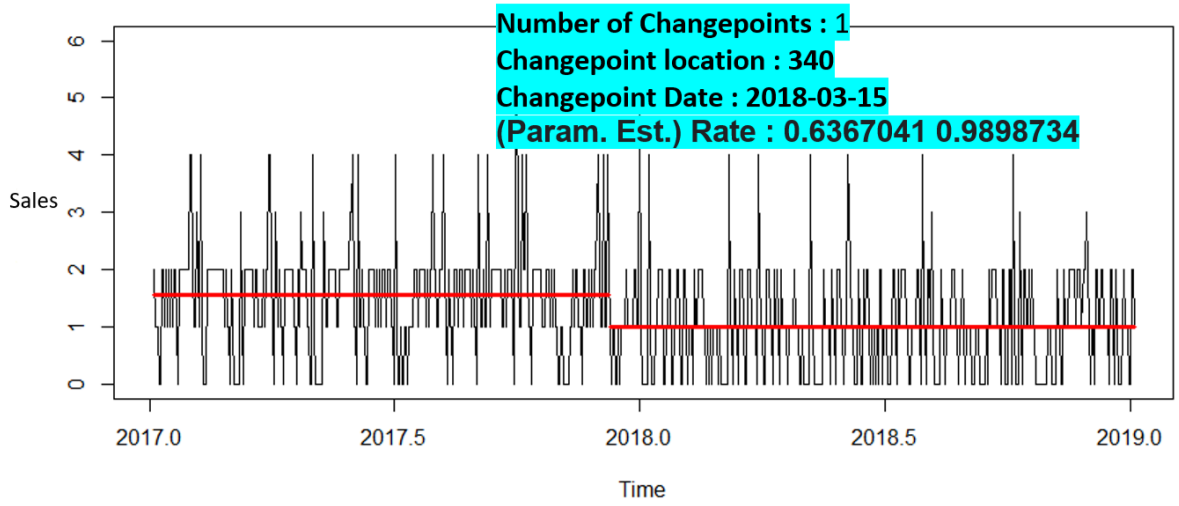

Product A3L4



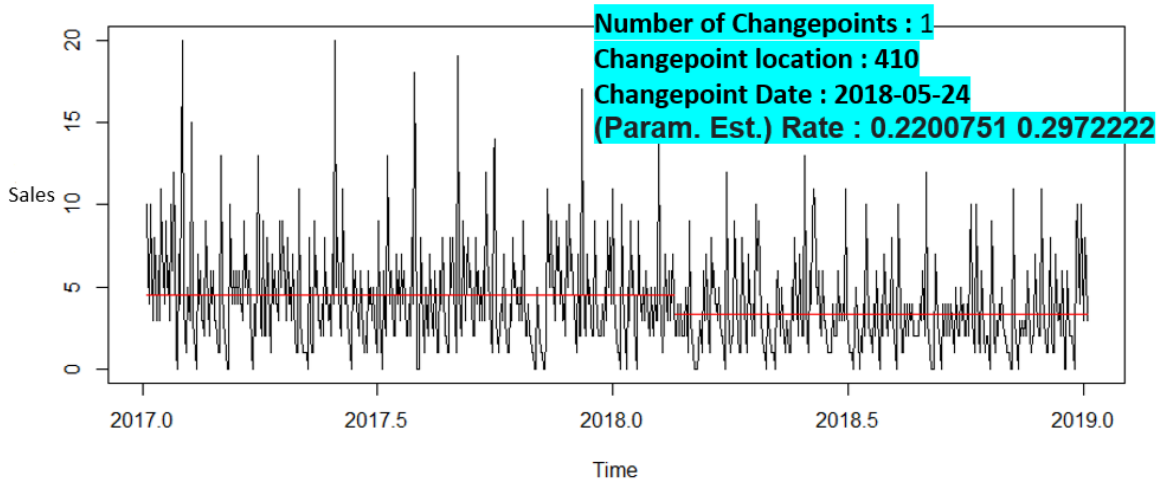

Product A3L5



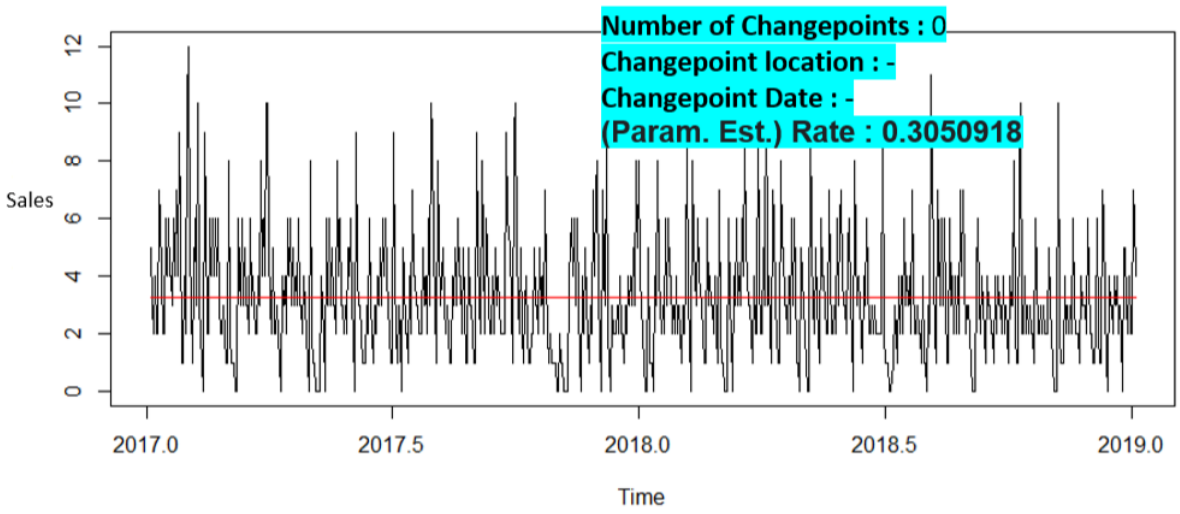
Product A3L6



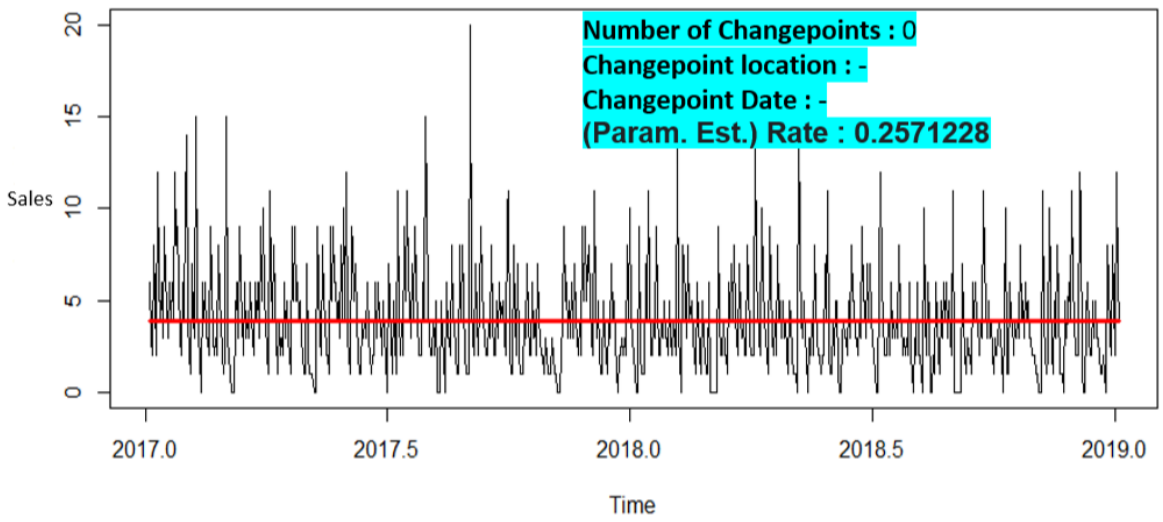

Product B1L1



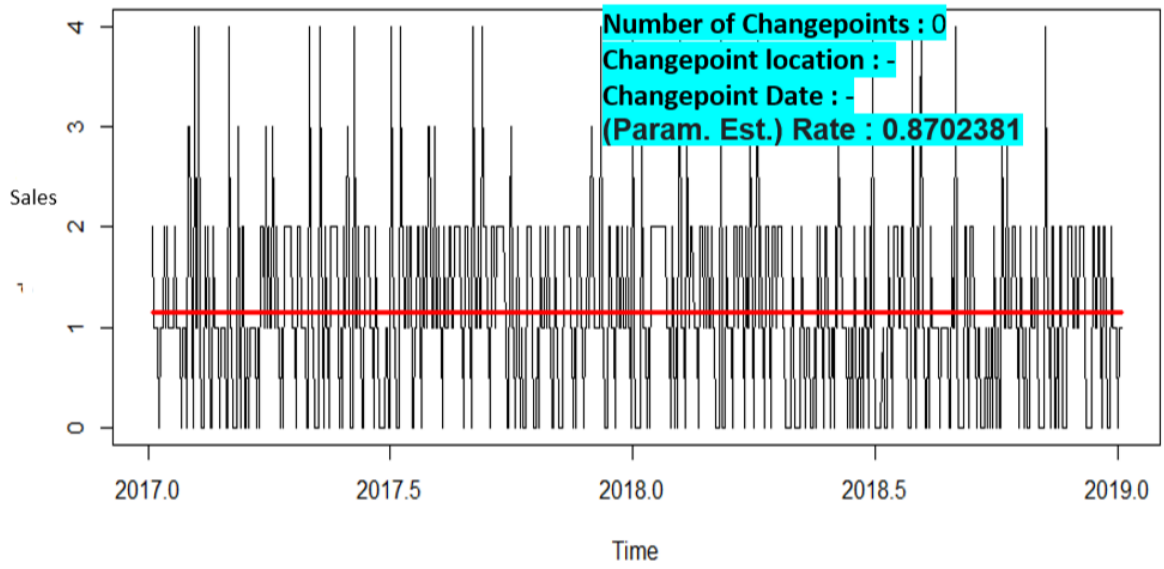

Product B1L2



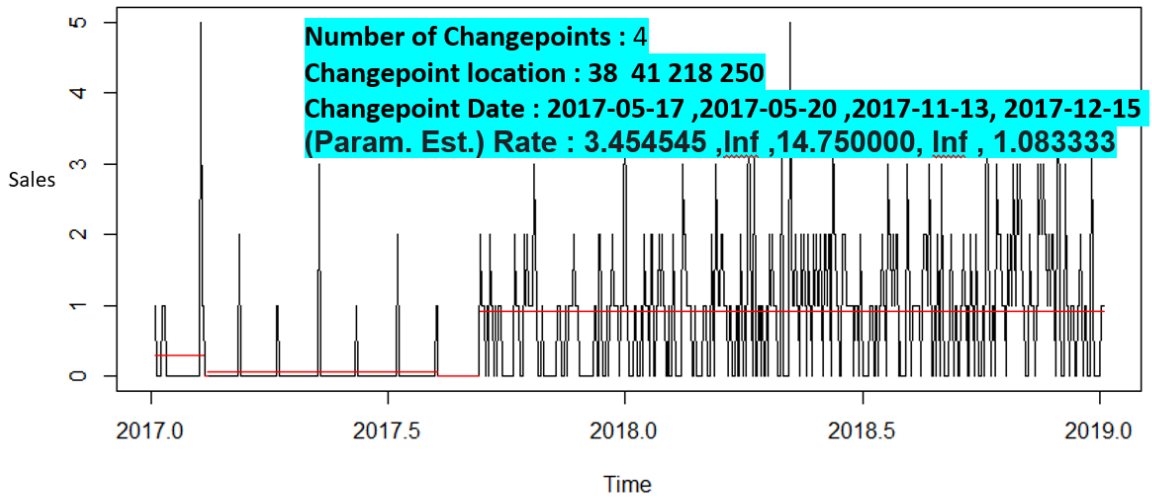
Product B1L3




Product B1L4




Product B1L5




Product B1L6