TITLE: ANOMALY DETECTION

AUTHOR: JOSEPH NJUGUNA

DATE: 10/6/22

1.Defining the question

a) Specifying the question

Check whether there are any anomalies in the given sales dataset. The objective of this task being fraud detection.

b) Defining the metric for success

Identification of anomalies in our dataset.

c) Understanding the context

You are a Data analyst at Carrefour Kenya and are currently undertaking a project that will inform the marketing department on the most relevant marketing strategies that will result in the highest no. of sales (total price including tax). Your project has been divided into four parts where you'll explore a recent marketing dataset by performing various unsupervised learning techniques and later providing recommendations based on your insights.

d) Recording the experimental design

- Exploratory data analysis

- Implementing the solution

2. Reading the data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(anomalize)
```

```
## == Use anomalize to improve your Forecasts by 50%! =============================
## Business Science offers a 1-hour course - Lab #18: Time Series Anomaly Detection!
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

```r
anomalydf <- read.csv("Supermarket_Sales_Forecasting - Sales part(4).csv", header = TRUE, sep = ",")
```

3. Exploring the data

```r
### viewing first 5 rows of our dataset
head(anomalydf)
```

```
##        Date    Sales
## 1  1/5/2019 548.9715
## 2  3/8/2019  80.2200
## 3  3/3/2019 340.5255
## 4 1/27/2019 489.0480
## 5  2/8/2019 634.3785
## 6 3/25/2019 627.6165
```

```r
### viewing last 5 rows of our dataset
tail(anomalydf)
```

```
##           Date     Sales
## 995  2/18/2019   63.9975
## 996  1/29/2019   42.3675
## 997   3/2/2019 1022.4900
## 998   2/9/2019   33.4320
## 999  2/22/2019   69.1110
## 1000 2/18/2019  649.2990
```

```r
### glimpse of unique values
library(dplyr)
glimpse(anomalydf)
```

```
## Rows: 1,000
## Columns: 2
## $ Date  <chr> "1/5/2019", "3/8/2019", "3/3/2019", "1/27/2019", "2/8/2019", "3/~
## $ Sales <dbl> 548.9715, 80.2200, 340.5255, 489.0480, 634.3785, 627.6165, 433.6~
```

```r
### checking data types and their class
str(anomalydf)
```

```
## 'data.frame':    1000 obs. of  2 variables:
##  $ Date : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ Sales: num  549 80.2 340.5 489 634.4 ...
```

Our dataset has one categorical and one numerical column.

```
### dimensions of our dataset
dim(anomalydf)
```

```
## [1] 1000    2
```

Our dataset has 1000 instances and 2 columns

```
### brief statistical summary on our dataset
summary(anomalydf)
```

```
##      Date               Sales        
##  Length:1000        Min.   :  10.68  
##  Class :character   1st Qu.: 124.42  
##  Mode  :character   Median : 253.85  
##                     Mean   : 322.97  
##                     3rd Qu.: 471.35  
##                     Max.   :1042.65  
```

```
### description of our dataset
library(psych)
```

```
## 
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
## 
##     %+%, alpha
```

```
describe(anomalydf)
```

```
##       vars    n   mean     sd median trimmed    mad   min     max   range  skew
## Date*    1 1000  45.58  25.89  47.00   45.63  34.10  1.00   89.00   88.00 -0.03
## Sales    2 1000 322.97 245.89 253.85  293.91 233.78 10.68 1042.65 1031.97  0.89
##       kurtosis   se
## Date*    -1.23 0.82
## Sales    -0.09 7.78
```

Mean sales came to 322.97/=

4. Cleaning the data

Uniformity

```r
### aligning case of our columns to lower case for all
names(anomalydf) <- tolower(names(anomalydf))
```

```r
### lets check for duplicate values
duplicates <- anomalydf[duplicated(anomalydf),]
duplicates
```

```
## [1] date  sales
## <0 rows> (or 0-length row.names)
```

We have no duplicate values.

```r
### detecting missing values
colSums(is.na(anomalydf))
```

```
##  date sales
##     0     0
```

We have no missing values

```r
str(anomalydf)
```

```
## 'data.frame':    1000 obs. of  2 variables:
##  $ date : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ sales: num  549 80.2 340.5 489 634.4 ...
```

```r
### converting datatypes
anomalydf$date <- as.Date(Sys.Date() + 1:nrow(anomalydf))
```

5. Implementing the Solution

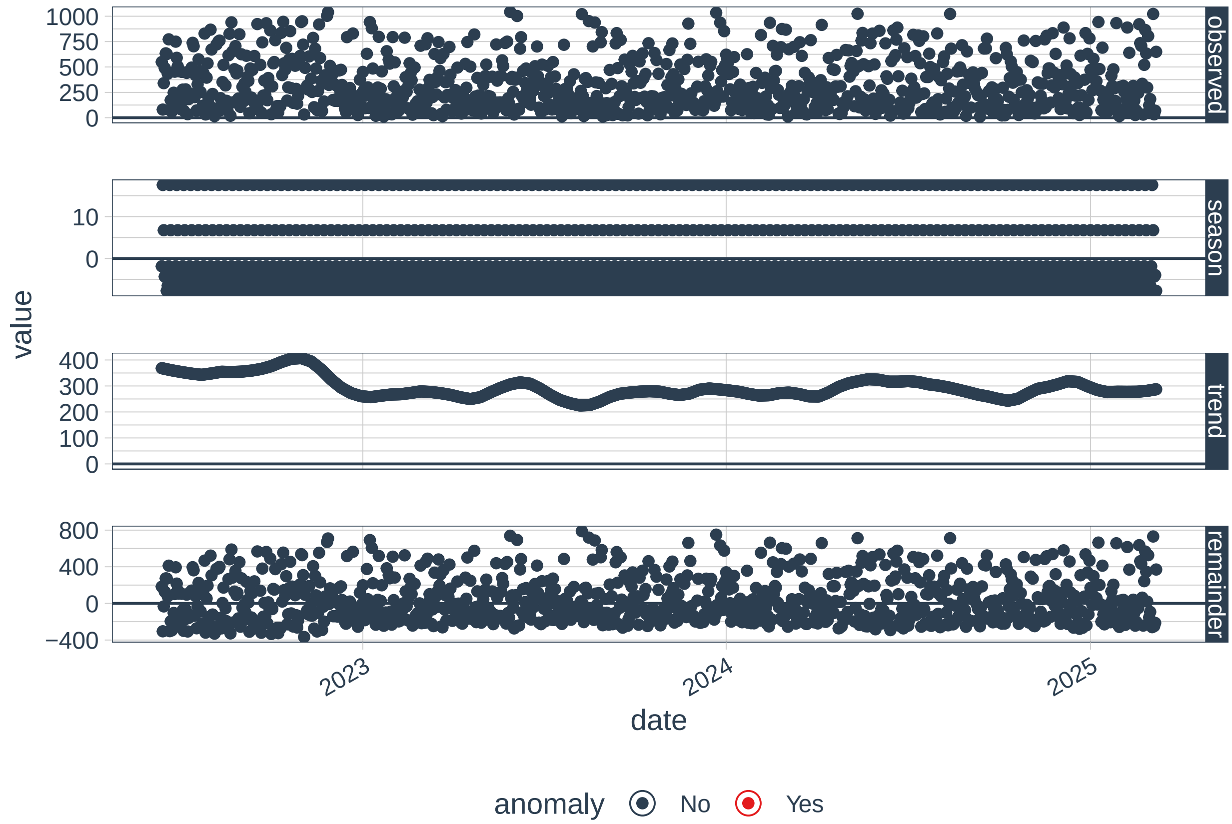
```r
anomalydf %>%
  as_tibble() %>%
  time_decompose(sales, method = "stl", frequency = "auto", trend = "auto") %>%
  anomalize(remainder, method = "gesd", alpha = 0.05, max_anoms = 0.2) %>%
  plot_anomaly_decomposition()
```

```
## Converting from tbl_df to tbl_time.
## Auto-index message: index = date
```

```
## frequency = 7 days
```

```
## trend = 91.5 days
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```



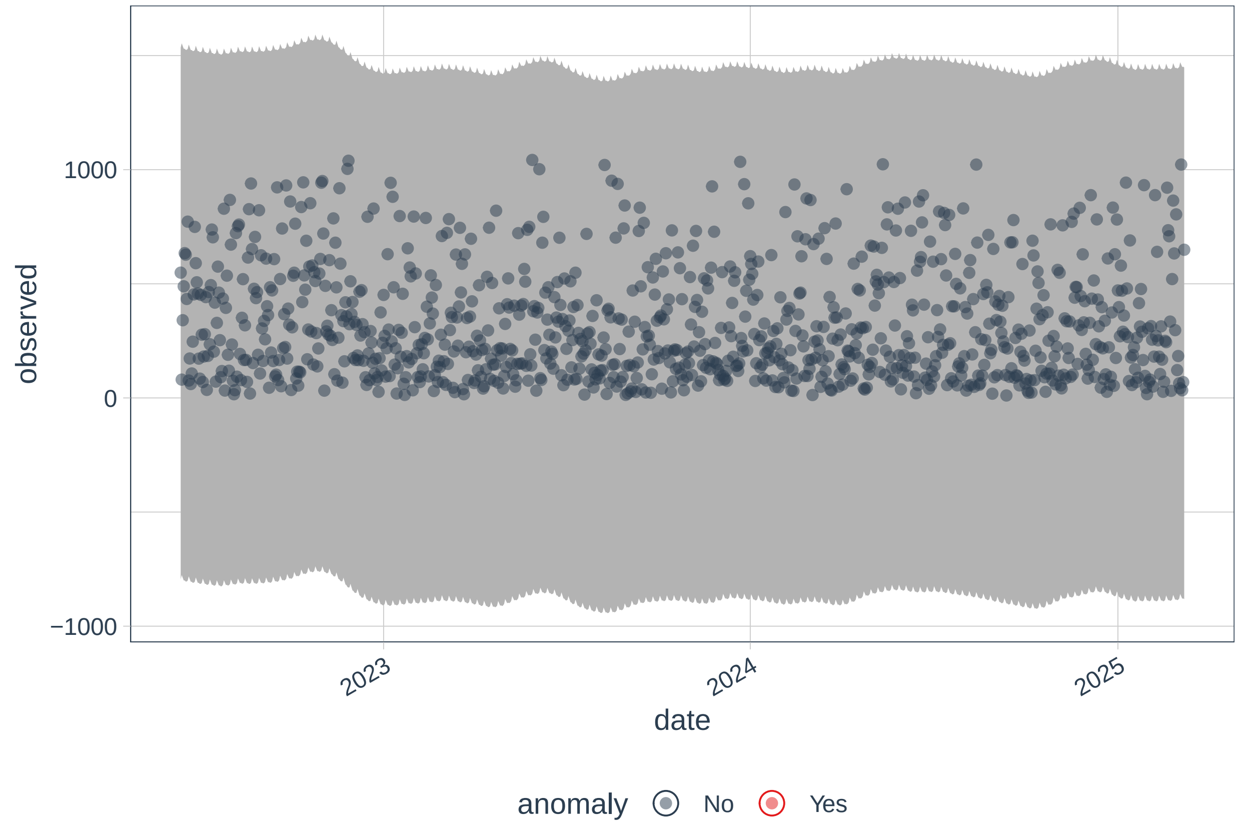
Alpha level of 0.05 does not detect any anomalies.

```
anomalydf %>%
  as_tibble() %>%
  time_decompose(sales) %>%
  anomalize(remainder) %>%
  time_recompose() %>%
  plot_anomalies(time_recomposed = TRUE, ncol = 3, alpha_dots = 0.5)
```

```
## Converting from tbl_df to tbl_time.
## Auto-index message: index = date
```

```
## frequency = 7 days
```

```
## trend = 91.5 days
```



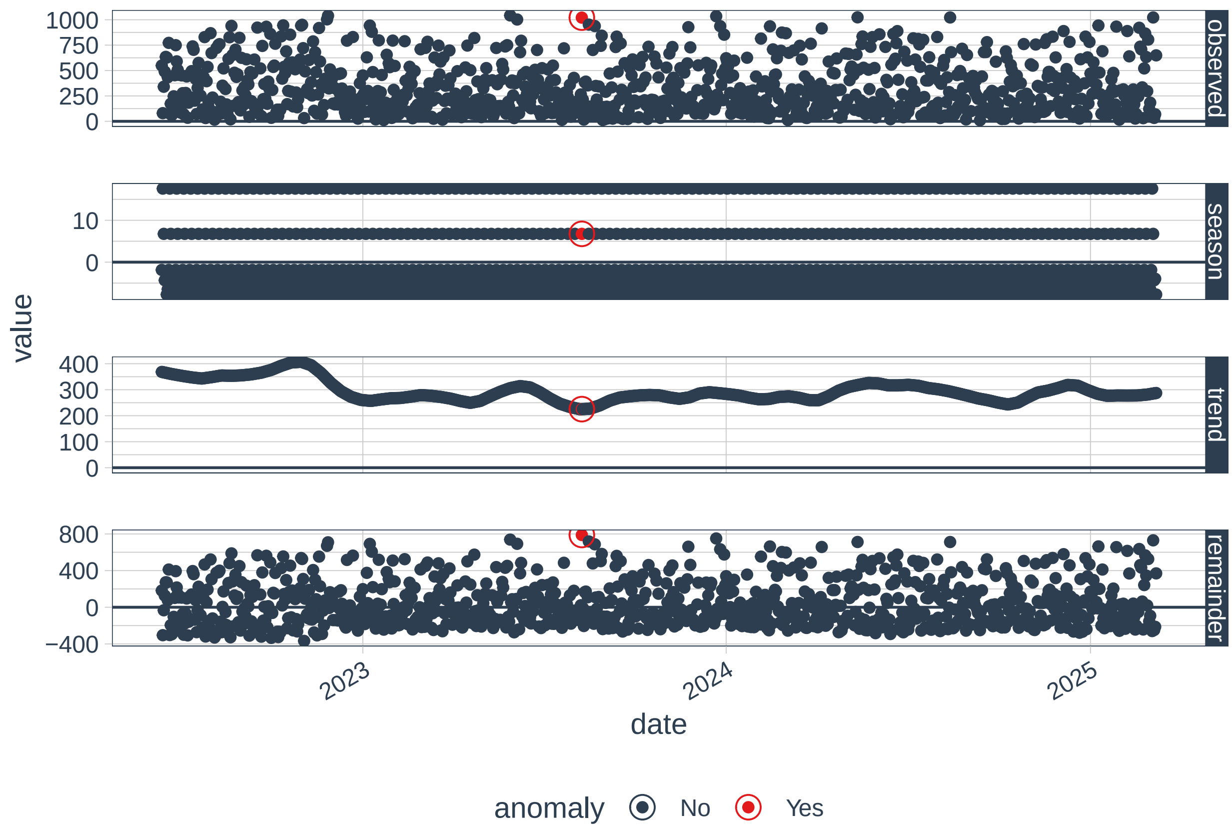
Hyperparameter tuning

```
### alpha level of 0.25
anomalydf %>%
  as_tibble() %>%
  time_decompose(sales, method = "stl", frequency = "auto", trend = "auto") %>%
  anomalize(remainder, method = "gesd", alpha = 0.35, max_anoms = 0.2) %>%
  plot_anomaly_decomposition()
```

```
## Converting from tbl_df to tbl_time.
## Auto-index message: index = date
```

```
## frequency = 7 days
```

```
## trend = 91.5 days
```



An alpha level of 0.35 detects anomalies.

5. Conclusion

Alpha level of 0.5 doesn't detect any anomalies.

Alpha level of 0.35 detects anomalies.

The year 2023 has most anomalies.

6. Recommendation

Sales in the year 2023 could be increased by selling in bulk.

7. Follow up questions

a) Did we have right data?

```
Yes.
```

b) Do we need other data to answer our question?

```
No.
```

c) Did we have the right question?

`Yes.`