

Package: StructuralDecompose (via r-universe)

October 18, 2024

Type Package

Title Decomposes a Level Shifted Time Series

Version 0.1.1

Description Explains the behavior of a time series by decomposing it into its trend, seasonality and residuals. It is built to perform very well in the presence of significant level shifts. It is designed to play well with any breakpoint algorithm and any smoothing algorithm. Currently defaults to 'lowess' for smoothing and 'strucchange' for breakpoint identification. The package is useful in areas such as trend analysis, time series decomposition, breakpoint identification and anomaly detection.

License MIT + file LICENSE

URL <https://allen-1242.github.io/StructuralDecompose/>

Depends R (>= 2.10)

Imports changepoint, segmented, strucchange

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Repository <https://allen-1242.r-universe.dev>

RemoteUrl <https://github.com/allen-1242/structuraldecompose>

RemoteRef HEAD

RemoteSha e442fd2d5f527d6c2f20d65e4665ebaac9fa5159

Contents

| | |
|-------------------------------|----------|
| AnomalyDetection | 2 |
| BreakPoints | 3 |
| LevelCheck | 3 |
| MeanCleaning | 4 |
| MedianCleaning | 5 |
| Nile_dataset | 5 |
| Smoothing | 6 |
| StructuralDecompose | 6 |
| Index | 8 |

| | |
|------------------|------------------------------------|
| AnomalyDetection | <i>Automatic Anomaly detection</i> |
|------------------|------------------------------------|

Description

Automatic Anomaly detection

Usage

```
AnomalyDetection(  
  timeseries,  
  frequency = 52,  
  conf_level = 1.5,  
  breaks,  
  window_len = 14  
)
```

Arguments

- timeseries Given time series
- frequency Timeseries frequency, defaults to 12 points
- conf_level Confidence level for Anomaly detection
- breaks breakpoints identified
- window_len Window length for anomaly detection

Value

the list of anomalies in the time series, along with the time series plot

Examples

```
AnomalyDetection(timeseries = StructuralDecompose::Nile_dataset[,1], breaks = c(4, 50, 80))  
  
AnomalyDetection(timeseries = runif(n = 50, min = 1, max = 10), breaks = c(4, 20, 30))
```

| | |
|-------------|----------------------------------|
| BreakPoints | <i>Generation of breakpoints</i> |
|-------------|----------------------------------|

Description

Generation of breakpoints

Usage

```
BreakPoints(
  timeseries,
  frequency = 52,
  break_algorithm = "strucchange",
  break_level = 0.05
)
```

Arguments

| | |
|-----------------|--|
| timeseries | Given time series |
| frequency | Timeseries frequency, defaults to 12 points |
| break_algorithm | Breakpoint algorithm to be used. Defaults to strucchange |
| break_level | Additional parameters for breakpoint algorithm |

Value

A list of breakpoints

Examples

```
BreakPoints(timeseries = seq(100), frequency = 52, break_level = 0.05)
BreakPoints(timeseries = StructuralDecompose::Nile_dataset[,1], frequency = 52)
```

| | |
|------------|------------------------------------|
| LevelCheck | <i>Minimum level length checks</i> |
|------------|------------------------------------|

Description

Minimum level length checks

Usage

```
LevelCheck(timeseries, level_length = 10, breaks)
```

Arguments

| | |
|--------------|----------------------------------|
| timeseries | Given time series |
| level_length | Mean distance between two levels |
| breaks | breakpoints returned |

Value

The series cleaned with the minimum level check

Examples

```
LevelCheck(timeseries = StructuralDecompose::Nile_dataset[,1], breaks = c(1,4,5))  
  
LevelCheck(timeseries = runif(n = 50, min = 1, max = 10), breaks = c(1,4,5))
```

| | |
|--------------|--------------------------|
| MeanCleaning | <i>Mean level checks</i> |
|--------------|--------------------------|

Description

Mean level checks

Usage

```
MeanCleaning(timeseries, mean_level = 0.5, breaks, frequency = 52)
```

Arguments

| | |
|------------|---|
| timeseries | Given time series |
| mean_level | Mean distance between two levels |
| breaks | breakpoints returned |
| frequency | Timeseries frequency, defaults to 12 points |

Value

The series cleaned with the mean check

Examples

```
MeanCleaning(timeseries = StructuralDecompose::Nile_dataset[,1], breaks = c(1,4,5), frequency = 1)  
  
MeanCleaning(timeseries = runif(n = 50, min = 1, max = 10), breaks = c(1,4,5), frequency = 12)
```

| | |
|----------------|----------------------------|
| MedianCleaning | <i>Median level checks</i> |
|----------------|----------------------------|

Description

Median level checks

Usage

MedianCleaning(timeseries, median_level = 0.5, breaks, frequency = 52)

Arguments

- timeseries Given time series
- median_level Median distance between two levels
- breaks Breaks identified
- frequency Timeseries frequency, defaults to 12 points

Value

The series cleaned with the median check

Examples

```
MedianCleaning(timeseries = StructuralDecompose::Nile_dataset[,1], breaks = c(1,4,5))  
  
MedianCleaning(timeseries = runif(n = 50, min = 1, max = 10), breaks = c(1,4,5))
```

| | |
|--------------|---------------------------|
| Nile_dataset | <i>Nile River Dataset</i> |
|--------------|---------------------------|

Description

Nile River Dataset

| | |
|-----------|---------------------------------------|
| Smoothing | <i>Smoothening of the time series</i> |
|-----------|---------------------------------------|

Description

Smoothening of the time series

Usage

```
Smoothing(timeseries, frequency = 52, smoothening_algorithm = "lowess", breaks)
```

Arguments

- timeseries Given time series
- frequency Timeseries frequency, defaults to 12 points
- smoothening_algorithm
 Smoothening algorithm required
- breaks Breakpoints identified by the previous algorithm
- lowess Lowess smoothener

Value

The smoothened time series

Examples

```
Smoothing(timeseries = StructuralDecompose::Nile_dataset[,1], breaks = c(4, 50, 80))  
  
Smoothing(timeseries = runif(n = 50, min = 1, max = 10), breaks = c(4, 20, 30))
```

| | |
|---------------------|-------------------------------------|
| StructuralDecompose | <i>Main decomposition algorithm</i> |
|---------------------|-------------------------------------|

Description

Main decomposition algorithm

Usage

```
StructuralDecompose(
  Data,
  frequency = 12,
  break_algorithm = "strucchange",
  smoothening_algorithm = "lowess",
  break_level = 0.05,
  median_level = 0.5,
  mean_level = 0.5,
  level_length = 12,
  conf_level = 0.5,
  window_len = 12,
  plot = FALSE
)
```

Arguments

| | |
|-----------------------|--|
| Data | Time series required |
| frequency | Frequency of the time series |
| break_algorithm | breakpoints algorithm used. Defaults to strucchange |
| smoothening_algorithm | Smoothing algorithm used. Defaults to lowess |
| break_level | Break level for the breakpoints algorithm |
| median_level | Average median distance between two level |
| mean_level | Average mean distance between a group of points near breakpoints |
| level_length | Minimum number of points required to determine a level |
| conf_level | Confidence level for Anomaly detection, best to keep this a static value |
| window_len | Length of the Moving window for Anomaly Detection |
| plot | True or False indicating if you want the internal plots to be generated |

Value

The decomposed time series along with a host of other metrics

Examples

```
StructuralDecompose(Data = StructuralDecompose::Nile_dataset[,1])
```

```
StructuralDecompose(Data = runif(n = 50, min = 1, max = 10))
```

Index

* **datasets**

Nile_dataset, [5](#)

AnomalyDetection, [2](#)

BreakPoints, [3](#)

LevelCheck, [3](#)

MeanCleaning, [4](#)

MedianCleaning, [5](#)

Nile_dataset, [5](#)

Smoothing, [6](#)

StructuralDecompose, [6](#)