

Package ‘decp’

August 22, 2024

Type Package

Title Complete Change Point Analysis

Version 0.1.2

Description Provides a comprehensive approach for identifying and estimating change points in multivariate time series through various statistical methods. Implements the multiple change point detection methodology from Ryan & Killick (2023) <doi:10.1080/00401706.2023.2183261> and a novel estimation methodology from Fotopoulos et al. (2023) <doi:10.1007/s00362-023-01495-0> generalized to fit the detection methodologies. Performs both detection and estimation of change points, providing visualization and summary information of the estimation process for each detected change point.

Depends R (>= 3.1)

License GPL-3

Encoding UTF-8

Imports geigen, ggplot2, magrittr, matrixcalc, purrr, rlang, stats

RoxygenNote 7.3.1

Maintainer Vasileios Pavlopoulos <vasileios.pavlopoulos@uah.edu>

NeedsCompilation no

Author Vasileios Pavlopoulos [cre, aut],

Hieu Pham [aut, ctb],

Paras Bhatt [aut, ctb],

Yi Tan [aut, ctb],

Ravi Patnayakuni [aut, ctb]

Repository CRAN

Date/Publication 2024-08-22 09:10:02 UTC

Contents

adjusted_ratio_bin_seg	2
decp	3
mle_change_point_detection	4
plot.decp_result	5

plot.mle_change_point_result	5
print.decp_result	6
print.mle_change_point_result	6
simulate_estimation	7
summary.decp_result	8

Index	9
--------------	----------

adjusted_ratio_bin_seg

Adjusted Ratio Binary Segmentation

Description

Adjusted ratio binary segmentation.

Usage

```
adjusted_ratio_bin_seg(input_data, minseglen, alpha)
```

Arguments

input_data	A numeric matrix of observations for multivariate time series data where the dimension is not greater than the observations. Date columns should not be inputted.
minseglen	Minimum segment length for detecting change points.
alpha	Level of significance for calculating the confidence intervals.

Value

A list with change points and segments.

Examples

```
# Example usage
data <- matrix(rnorm(1000), ncol = 10)
result <- adjusted_ratio_bin_seg(data, minseglen = 30, alpha = 0.05)
```

decp *Detect and Estimate Change Points*

Description

Detect and estimate change points.

Usage

```
decp(  
  input_data,  
  alpha = 0.05,  
  num_simulations = 10000,  
  num_iterations = 100,  
  verbose = TRUE  
)
```

Arguments

input_data	A numeric matrix of observations for multivariate time series data where the dimension is not greater than the observations. Date columns should not be inputted.
alpha	Level of significance for calculating the confidence intervals
num_simulations	Specifies the number of simulations to be conducted during the estimation process. It is recommended to set num_simulations to a large value to ensure greater certainty and reliability of the results. A higher number of simulations helps in capturing the variability and improves the accuracy of the estimation.
num_iterations	Determines the size of the two-sided random walk in the estimation process (each path). If the jump size of the change point is small, num_iterations should be set to higher values to achieve accurate results. For jump size ≥ 1 , the default value is 100.
verbose	Logical value indicating whether to print messages during the function execution. Default is TRUE.

Value

An object of class 'decp_result' containing the ordered change points, the summary of the jump sizes for each pair of segments, the Confidence Interval (C.I.) of each detected change point, the maximum zhta, confidence interval level, and warnings in case that the C.I. of two adjacent change points overlap.

Examples

```
# Example usage
data_part1 <- matrix(rnorm(1500, mean = 0, sd = 1), ncol = 5)
data_part2 <- matrix(rnorm(1500, mean = 3, sd = 1), ncol = 5)
data <- rbind(data_part1, data_part2)
result <- decp(data, alpha = 0.05, num_simulations = 100, num_iterations = 50)
print(result)
```

mle_change_point_detection

MLE Change Point Detection

Description

Maximum likelihood estimation change point detection.

Usage

```
mle_change_point_detection(input_data, verbose = TRUE)
```

Arguments

input_data	A numeric matrix of observations for multivariate time series data where the dimension is not greater than the observations. Date columns should not be inputted.
verbose	Logical value indicating whether to print messages during the function execution. Default is TRUE.

Value

An object of class 'mle_change_point_result' containing the index of the change point estimate, its MLE value, and the MLE data.

Examples

```
# Example usage
data <- matrix(rnorm(1000), ncol = 10)
tau_range <- 30:(nrow(data) - 30)
result <- mle_change_point_detection(data)
print(result)
```

`plot.decp_result` *Plot method for decp_result*

Description

Plot method for `decp_result`

Usage

```
## S3 method for class 'decp_result'  
plot(x, ...)
```

Arguments

`x` An object of class 'decp_result'
`...` Additional arguments passed to the plotting function

`plot.mle_change_point_result`
 Plot method for mle_change_point_result

Description

Plot method for `mle_change_point_result`

Usage

```
## S3 method for class 'mle_change_point_result'  
plot(x, ...)
```

Arguments

`x` An object of class 'mle_change_point_result'
`...` Additional arguments passed to the plotting function

```
print.decp_result      Print method for decp_result
```

Description

Print method for decp_result

Usage

```
## S3 method for class 'decp_result'  
print(x, ...)
```

Arguments

x	An object of class 'decp_result'
...	Additional arguments (not used)

```
print.mle_change_point_result  
      Print method for 'mle_change_point_result' class
```

Description

Print method for 'mle_change_point_result' class

Usage

```
## S3 method for class 'mle_change_point_result'  
print(x, ...)
```

Arguments

x	An object of class 'mle_change_point_result'.
...	Additional arguments (not used).

simulate_estimation *Simulate Estimation*

Description

The estimation of the detected change point.

Usage

```
simulate_estimation(  
  lambda1,  
  lambda2,  
  term1,  
  term2,  
  num_simulations,  
  num_iterations  
)
```

Arguments

lambda1	Eigenvalues of the first segment.
lambda2	Eigenvalues of the second segment.
term1	The negative drift term of the left hand side of the random walk.
term2	The negative drift term of the right hand side of the random walk.
num_simulations	Specifies the number of simulations to be conducted during the estimation process. It is recommended to set num_simulations to a large value to ensure greater certainty and reliability of the results. A higher number of simulations helps in capturing the variability and improves the accuracy of the estimation.
num_iterations	Determines the size of the two-sided random walk in the estimation process (each path). If the jump size of the change point is small, num_iterations should be set to higher values to achieve accurate results. For jump size ≥ 1 , the default value is 100.

Value

A numeric vector of the estimation results centered around zero. The spike of the histogram is represents estimated change point, and it is expected to be at zero.

Examples

```
# Example usage  
lambda1 <- rnorm(10)  
lambda2 <- rnorm(10)  
term1 <- -1  
term2 <- -2  
result <- simulate_estimation(lambda1, lambda2, term1, term2,  
                             num_iterations = 100, num_simulations = 100)
```

summary.decp_result *Summary method for decp_result*

Description

Summary method for decp_result

Usage

```
## S3 method for class 'decp_result'  
summary(object, ...)
```

Arguments

object	An object of class 'decp_result'
...	Additional arguments (not used)

Index

`adjusted_ratio_bin_seg`, 2

`decp`, 3

`mle_change_point_detection`, 4

`plot.decp_result`, 5

`plot.mle_change_point_result`, 5

`print.decp_result`, 6

`print.mle_change_point_result`, 6

`simulate_estimation`, 7

`summary.decp_result`, 8