

Package ‘BayesProject’

May 6, 2026

Type Package

Title Fast Projection Direction for Multivariate Changepoint Detection

Version 1.0

Date 2020-09-24

Author Georg Hahn [aut,cre], Paul Fearnhead [ctb], Idris A. Eckley [ctb]

Maintainer Georg Hahn <ghahn@hsph.harvard.edu>

Description

Implementations in 'cpp' of the BayesProject algorithm (see G. Hahn, P. Fearnhead, I.A. Eckley (2020) <[doi:10.1007/s11222-020-09966-2](https://doi.org/10.1007/s11222-020-09966-2)>) which implements a fast approach to compute a projection direction for multivariate changepoint detection, as well as the sum-cusum and max-cusum methods, and a wild binary segmentation wrapper for all algorithms.

License GPL (>= 2)

Imports Rcpp (>= 0.12.13), Rdpack

RdMacros Rdpack

LinkingTo Rcpp, RcppEigen

RoxygenNote 7.1.0

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-09-29 09:30:16 UTC

Contents

bayes	2
summaxcusum	3
testdata	4
wildBinSeg	4

Index	6
--------------	----------

bayes	<i>Cpp implementation of the Bayesian projection algorithm to detect single multivariate changepoints.</i>
-------	------------------------------------------------------------------------------------------------------------

Description

Detects one multivariate changepoint in a dataset using the fast projection direction algorithm of Hahn et al. (2019). Solely required is the dataset as first parameter. The testing threshold ("threshold"), the number of timepoints to calculate a projection ("nTimePoints") and the regularisation parameter ("K") are chosen automatically.

Usage

```
bayes(x, threshold, nTimePoints = NULL, K = 1/sqrt(2), rescale.var = TRUE)
```

Arguments

x	A $p \times n$ matrix representing p data series having n observations each.
threshold	The testing threshold to detect the single changepoint. If missing, parameter will be calibrated automatically.
nTimePoints	The number of equidistant timepoints at which the projection direction is calculated. If no value (NULL) is given, timepoints are chosen automatically.
K	The regularisation parameter for the Bayesian projection direction. Default is $1/\sqrt{2}$.
rescale.var	A boolean flag to indicate if the variance should be rescaled before detecting a changepoint. Default is TRUE.

References

Hahn, G., Fearnhead, P., Eckley, I.A. (2020). Fast computation of a projection direction for multivariate changepoint detection. Stat Comput.

Examples

```
library(BayesProject)
data(testdata)
res <- bayes(testdata,nTimePoints=100)
print(res$cpt)
```

summaxcusum	<i>Cpp implementation of sum-cusum and max-cusum for single change-point detection.</i>
-------------	-----------------------------------------------------------------------------------------

Description

Detects one multivariate changepoint in a dataset using the sum-cusum or max-cusum technique. Solely required is the dataset as first parameter. The testing threshold ("threshold") is chosen automatically if missing. The parameter "sum_cusum" (default TRUE) indicates if sum-cusum or max-cusum is used.

Usage

```
summaxcusum(x, threshold, sum_cusum = TRUE, rescale.var = TRUE)
```

Arguments

x	A $p \times n$ matrix representing p data series having n observations each.
threshold	The testing threshold to detect the single changepoint. If missing, parameter will be calibrated automatically.
sum_cusum	A boolean flag to indicate if sum cusum (sum_cusum=T) or max cusum (sum_cusum=F) is used. Default is TRUE.
rescale.var	A boolean flag to indicate if the variance should be rescaled before detecting a changepoint. Default is TRUE.

References

Hahn, G., Fearnhead, P., Eckley, I.A. (2020). Fast computation of a projection direction for multivariate changepoint detection. Stat Comput.

Examples

```
library(BayesProject)
data(testdata)
resSumCusum <- summaxcusum(testdata, sum_cusum=TRUE)
print(resSumCusum$cpt)
resMaxCusum <- summaxcusum(testdata, sum_cusum=FALSE)
print(resMaxCusum$cpt)
```

testdata	<i>Simulated test data.</i>
----------	-----------------------------

Description

A dataset containing time series for 100 variates with 1000 data points for each variate. The dataset contains 5 changepoints with each one being shared independently by 10 variates. The observations prior to each changepoint are IID Gaussian, distributed with unit variance and random mean drawn from $N(0,1)$ Gaussians. The mean after each changepoint, for each variable that is selected to have a change in mean, is changed by size=1 at each changepoint location with the sign of change chosen uniformly at random.

Usage

```
data(testdata)
```

Format

A matrix with 100 rows and 2000 columns.

References

Hahn, G., Fearnhead, P., Eckley, I.A. (2020). Fast computation of a projection direction for multivariate changepoint detection. Stat Comput.

wildBinSeg	<i>Wild Binary Segmentation Wrapper for the functions "bayes" and "summaxcusum".</i>
------------	--------------------------------------------------------------------------------------

Description

Detects multivariate changepoints in a dataset using the Wild Binary Segmentation framework of Fryzlewicz (2014). The dataset is supplied as the first parameter. The second parameter is a calibrated function "cusumFct(x)" which takes a multivariate data matrix x and returns a cusum vector for it. The threshold is supplied with parameter "threshold", parameter "m" specifies the number of random WBS intervals on each recursion level, and "minwindow" is the minimal window size up to which the dataset is further divided recursively to find more changepoints.

Usage

```
wildBinSeg(x, cusumFct, threshold, m = 100, minwindow = 10, rescale.var = TRUE)
```

Arguments

<code>x</code>	A $p \times n$ matrix representing p data series having n observations each.
<code>cusumFct</code>	A calibrated function which returns a cusum vector for a dataset supplied as its single input parameter. Note that rescaling of the variance should be deactivated inside "cusumFct". When using the function "bayes" as in the example below, it is advised to set the threshold to e.g. zero in order to deactivate the time-consuming (and unnecessary) threshold computation inside "bayes".
<code>threshold</code>	The testing threshold to detect the single changepoint. The threshold must be specified.
<code>m</code>	The number of random WBS intervals on each recursion level.
<code>minwindow</code>	The minimal window size up to which the dataset is further divided recursively to find more changepoints.
<code>rescale.var</code>	A boolean flag to indicate if the variance should be rescaled before detecting a changepoint. Default is TRUE.

References

Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *Ann Statist*, 42(6):2243–2281.

Examples

```
library(BayesProject)
data(testdata)
bayes_cusum <- function(x) bayes(x, threshold=0, rescale.var=FALSE)$cusum
res <- wildBinSeg(testdata, cusumFct=bayes_cusum, threshold=1)
print(res)
```

Index

- * **dataset**
 - testdata, 4
- bayes, 2
- summaxcusum, 3
- testdata, 4
- wildBinSeg, 4