Package 'npwbs'

May 9, 2026

Type Package

Title Nonparametric Multiple Change Point Detection Using Wild Binary Segmentation

Version 0.3.0

Description Implements a procedure for detecting multiple location-scale change points in a sequence of univariate observations, as described in Ross (2026) ``Nonparametric Detection of Multiple Location-Scale Change Points via Wild Binary Segmentation'' <doi:10.48550/arXiv.2107.01742>. The method combines Wild Binary Segmentation with a rank-based statistic and provides calibrated thresholds for controlling the probability of incorrectly detecting a change point in a homogeneous sequence.

Depends R (>= 3.6.0)

License GPL-3

Encoding UTF-8

NeedsCompilation no

Author Gordon J. Ross [aut, cre]

Maintainer Gordon J. Ross <`gordon.ross@ed.ac.uk`>

Repository CRAN

Date/Publication 2026-05-02 18:30:02 UTC

Contents

`detectChanges` *Nonparametric detection of multiple change points using Wild Binary Segmentation*

Description

Returns the estimated number and locations of the change points in a sequence of univariate observations. For full details of how this procedure works, please see G. J. Ross (2021) - "Nonparametric Detection of Multiple Location-Scale Change Points via Wild Binary Segmentation" at `https://arxiv.org/abs/2107.01742`

Usage

```
detectChanges(y,alpha=0.05,prune=TRUE,M=10000,d=2,displayOutput=FALSE)
```

Arguments

`y`	The sequence to test for change points
`alpha`	Required Type I error (i.e. false positive) rate. Can be set to either 0.05 or 0.01
`prune`	Whether to prune potential excess change points via post-processing. Most likely should be left as TRUE.
`M`	Number of subsequences to sample during WBS and pruning. The built-in thresholds are calibrated for `M = 10000`, so other values are not supported by the packaged calibration.
`d`	Minimum number of observations between change points. The built-in thresholds currently support only `d = 2`.
`displayOutput`	If TRUE then will print some information while searching for change points

Details

The built-in thresholds are precomputed for the Lepage statistic using `d = 2`, `M = 10000`, and `alpha` in `{0.05, 0.01}`. Thresholds are calibrated up to segment length `n = 10000`. For `n > 10000`, the implementation uses the `n = 10000` threshold and issues a warning. For `n > 1000`, the built-in thresholds use a smooth tail approximation fitted to additional Monte Carlo calibration points.

If `prune = TRUE`, a post-processing pruning step retests merged neighbouring segments and removes changepoints that are no longer supported.

Value

A vector containing the detected changepoint locations. A returned value `k` denotes a split between observations `k` and `k + 1`, i.e. the change occurs after observation `k`.

Author(s)

Gordon J. Ross `<gordon@gordonjross.co.uk>`

Examples

```
set.seed(100)
y <- c(rnorm(30,0,1),rnorm(30,3,1), rnorm(30,0,1),rnorm(30,0,3))
detectChanges(y)
```

Index