Package: L2hdchange (via r-universe)

August 26, 2024

Type Package

Date 2023-07-19

Title L2 Inference for Change Points in High-Dimensional Time Series

Version 1.0

Maintainer Rui Lin <ruilin1081@gmail.com>

Description Provides a method for detecting multiple change points in high-dimensional time series, targeting dense or spatially clustered signals. See Li et al. (2023) ``L2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM". arXiv preprint <arXiv:2208.13074>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Depends R (>= 2.10)

NeedsCompilation no

Author Jiaqi Li [aut], Likai Chen [aut], Weining Wang [aut], Wei Biao Wu [aut], Rui Lin [cre]

Date/Publication 2023-07-20 11:20:04 UTC

Repository https://rl1081.r-universe.dev

RemoteUrl https://github.com/cran/L2hdchange

RemoteRef HEAD

RemoteSha 0d5b8eb6bed031ba6e18290f3a27f0947a113e38

Contents

`check_nbd` *Check the validity of the neighbourhood specification*

Description

Check the validity of the neighbourhood specification

Usage

```
check_nbd(nbd_info)
```

Arguments

`nbd_info`	A list containing the neighbourhood information. See `ts_hdchange()`.

Value

No return value. Show an error message if nbd_info is invalid.

Examples

```
nbd_info <- list(c(1:10),c(8:20))
check_nbd <- check_nbd(nbd_info)
```

`covid_data` *U.S. COVID-19 Data*

Description

Daily number of COVID-19 cases for 58 areas in the United States (including 50 states, Washington D.C., 5 territories and 2 cruise ships) for 812 days from 22 Jan 2020 to 12 April 2022.

Usage

```
covid_data
```

Format

`covid_data`**:**

A data matrix with p = 58 rows and n = 812 columns.

Source

U.S. CDC https://covid.cdc.gov/covid-data-tracker/#maps_new-admissions-rate-county

---

`covid_nbd_info` *U.S. COVID-19 Data Neighbourhood Information*

Description

U.S. COVID-19 Data Neighbourhood Information

Usage

```
covid_nbd_info
```

Format

`covid_nbd_info`**:**

A list containing five arrays indicating the constituents of five U.S. regions:

Northeast: Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island, Vermont, New Jersey, New York, and Pennsylvania.

Midwest: Illinois, Indiana, Michigan, Ohio, Wisconsin, Iowa, Kansas, Minnesota, Missouri, Nebraska, North Dakota, and South Dakota.

South: Delaware, Florida, Georgia, Maryland, North Carolina, South Carolina, Virginia, District of Columbia, West Virginia, Alabama, Kentucky, Mississippi, Tennessee, Arkansas, Louisiana, Oklahoma, and Texas.

West: Arizona, Colorado, Idaho, Montana, Nevada, New Mexico, Utah, Wyoming, Alaska, California, Hawaii, Oregon, and Washington.

Others: American Samoa, Diamond Princess, Grand Princess, Guam, Northern Mariana Islands, Puerto Rico, and Virgin Islands.

Source

U.S. Census Bureau, W. (2000). *List of regions of the United States*.

`est_hdchange` *Construct an S3 class 'no_nbd' or 'nbd' for change-point estimation*

Description

Construct an S3 class 'no_nbd' or 'nbd' for change-point estimation

Usage

```
est_hdchange(hdobj, test_stats, threshold, stat_all, critical_values)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.
`test_stats`	A list containing the test statistics generated by `get_teststats()`.
`threshold`	The threshold in break estimation.
`stat_all`	An array of test statistics generated by `get_V_l2_MAinf()`.
`critical_values`	An array of quantiles for critical values.

Value

An S3 object of class 'no_nbd' or 'nbd' used as the argument of `get_breaks()`.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

teststats <- get_teststats(ts_no_nbd)
V_12_MAinf <- get_V_l2_MAinf(ts_no_nbd)

estobj <- est_hdchange(hdobj = ts_no_nbd, test_stats = teststats$stat_max,
threshold = 1e-5, stat_all = V_12_MAinf, critical_values = c(0.01, 0.05, 0.1))
```

---

genZ *Generate a random Gaussian vector*

Description

Generate a random Gaussian vector

Usage

```
genZ(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

The Gaussian random vector $\mathcal{Z}$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

Z <- genZ(ts_no_nbd)
```

---

get_breaks *Obtain the time-stamps and spatial locations with breaks*

Description

Obtain the time-stamps and spatial locations with breaks

Usage

```
get_breaks(estobj)
```

Arguments

estobj	An S3 object of class 'no_nbd' or 'nbd' generated by `est_hdchange()`.

Value

A list containing the time-stamps and spatial locations with breaks. For S3 class 'no_nbd', it returns the total number of breaks $\widehat{K}$ and the time-stamps $\hat{\tau}_k$. See Algorithm 1 of Li et al. (2023). For S3 class 'nbd', it returns the total number of breaks $\widehat{R}$ and the spatial-temporal location of the break $(\hat{\tau}_r, \hat{s}_r)$. See Algorithm 2 of Li et al. (2023).

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

teststats <- get_teststats(ts_no_nbd)
V_12_MAinf <- get_V_l2_MAinf(ts_no_nbd)

estobj <- est_hdchange(hdobj = ts_no_nbd, test_stats = teststats$stat_max,
threshold = 1e-5, stat_all = V_12_MAinf, critical_values = c(0.01, 0.05, 0.1))

breaks <- get_breaks(estobj)
```

`get_breaks.nbd` *Obtain the time-stamps and spatial locations with breaks*

Description

Obtain the time-stamps and spatial locations with breaks

Usage

```
## S3 method for class 'nbd'
get_breaks(estobj)
```

Arguments

`estobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `est_hdchange()`.

Value

A list containing the total number of breaks $\widehat{R}$ and the spatial-temporal location of the break $(\hat{\tau}_r, \hat{s}_r)$. See Algorithm 2 of Li et al. (2023).

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_breaks.no_nbd` *Obtain the time-stamps and spatial locations without break*

Description

Obtain the time-stamps and spatial locations without break

Usage

```
## S3 method for class 'no_nbd'
get_breaks(estobj)
```

Arguments

`estobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `est_hdchange()`.

Value

A list containing the total number of breaks $\widehat{K}$ and the time-stamps $\hat{\tau}_k$. See Algorithm 1 of Li et al. (2023).

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_cov_x_MAinf` *The covariance matrix for generating random Gaussian vector*

Description

The covariance matrix for generating random Gaussian vector

Usage

```
get_cov_x_MAinf(n, b)
```

Arguments

`n`	Number of time series observations.
`b`	Bandwith parameter $b = window\_size/n$.

Value

The covariance matrix. See section 2.2 of Li et al. (2023).

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

Cov_x_MAinf <- get_cov_x_MAinf(ts_no_nbd$n, ts_no_nbd$b)
```

---

get_critical *Obtain critical values and threshold*

Description

Obtain critical values and threshold

Usage

```
get_critical(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

A list containing the critical values and the threshold parameter ω.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

crit <- get_critical(ts_no_nbd)
```

---

get_critical.nbd *Obtain critical values and threshold*

Description

Obtain critical values and threshold

Usage

```
## S3 method for class 'nbd'
get_critical(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

A list containing the critical values and the threshold parameter ω.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_critical.no_nbd` *Obtain critical values and threshold*

Description

Obtain critical values and threshold

Usage

```
## S3 method for class 'no_nbd'
get_critical(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

A list containing the critical values and the threshold parameter ω.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_GS_MAinf` *Obtain the simulated standardised gap vector*

Description

Obtain the simulated standardised gap vector

Usage

```
get_GS_MAinf(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

An array of the simulated counterpart of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

GS_MAinf <- get_GS_MAinf(ts_no_nbd)
```

---

get_GS_MAinf.nbd *Obtain the simulated standardised gap vector*

Description

Obtain the simulated standardised gap vector

Usage

```
## S3 method for class 'nbd'
get_GS_MAinf(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

An array of the simulated counterpart of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_GS_MAinf.no_nbd` *Obtain the simulated standardised gap vector*

Description

Obtain the simulated standardised gap vector

Usage

```
## S3 method for class 'no_nbd'
get_GS_MAinf(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

An array of the simulated counterpart of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_lr_var` *Compute the long-run variance of the gap vector*

Description

Compute the long-run variance of the gap vector

Usage

```
get_lr_var(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

The covariance matrix of the gap vectors $\hat{J}(.)$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

lr_var <- get_lr_var(ts_no_nbd)
```

---

get_teststats *Obtain the test statistics*

Description

Obtain the test statistics

Usage

```
get_teststats(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

A list containing the test statistics $\boldsymbol{Q}_n$ and a sequence of standardised $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

teststat <- get_teststats(ts_no_nbd)
```

---

get_teststats.nbd *Obtain the test statistics*

Description

Obtain the test statistics

Usage

```
## S3 method for class 'nbd'
get_teststats(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

A list containing the test statistics $\boldsymbol{Q}_n$ and a sequence of standardised $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

get_teststats.no_nbd *Obtain the test statistics*

Description

Obtain the test statistics

Usage

```
## S3 method for class 'no_nbd'
get_teststats(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by ts_hdchange().

Value

A list containing the test statistics $\boldsymbol{Q}_n$ and a sequence of standardised $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

get_V_l2_MAinf *Obtain the standardised gap vector*

Description

Obtain the standardised gap vector

Usage

```
get_V_l2_MAinf(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by ts_hdchange().

Value

An array of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

V_12_MAinf <- get_V_l2_MAinf(ts_no_nbd)
```

---

get_V_l2_MAinf.nbd *Obtain the standardised gap vector*

Description

Obtain the standardised gap vector

Usage

```
## S3 method for class 'nbd'
get_V_l2_MAinf(hdobj)
```

Arguments

hdobj	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

An array of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`get_V_l2_MAinf.no_nbd` *Obtain the standardised gap vector*

Description

Obtain the standardised gap vector

Usage

```
## S3 method for class 'no_nbd'
get_V_l2_MAinf(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

An array of $|V_i|_2^2$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

`hdchange` *Estimate the time-stamps and spatial locations with breaks*

Description

The main function of this package. It performs a test for existence of breaks and estimates the time-stamps and locations of the breaks.

Usage

```
hdchange(hdobj)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.

Value

The return value is an S3 object of class 'no_nbd' or 'nbd' containing a list of the test results and change-point locations.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
############ No neighbourhood case ############

# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))



# Estimate the time-stamps of the breaks
est_result_no_nbd <- hdchange(ts_no_nbd)

# Summarize the results
summary(est_result_no_nbd)

# Plot the results
plot_result(est_result_no_nbd)
axis(1,
  at = est_result_no_nbd$time_stamps,
  labels = c("break 1", "break 2", "break 3")
)
title(main = "Change-points estimation")


############ Neighbourhood case ############

# generate data
data_nbd <- sim_hdchange_nbd(n = 300,
```

```
p = 70,
nbd_info =
 list(
   (1:9), (2:31), (32:41), (42:70),
   (3:15), (16:35), (31:55)
 ),
sp_tp_break = rbind(c(2, 50), c(4, 150), c(2, 250)),
dist_info =
  list(dist = "t", dependence = "iid", param = 5),
jump_max = 1)

# construct nbd object
ts_nbd <- ts_hdchange(data_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1),
nbd_info =
 list(
   (1:9), (2:31), (32:41), (42:70),
   (3:15), (16:35), (31:55)
 ))



# Estimate the time-stamps of the breaks
est_result_nbd <- hdchange(ts_nbd)

# Summarize the results
summary(est_result_nbd)

# Plot the results
plot_result(est_result_nbd, nbd_index = 2)
pairs <- est_result_nbd$nbd_and_stamps_pair
time_stamps <- pairs[pairs[, 1] == 2, 2]
axis(1,
  at = time_stamps,
  labels = c("break 1", "break 2")
)
title(main = "Change-points estimation for neibourhood 2")
```

`plot_result` *Plot the time series and change-points*

Description

Plot the time series and change-points

Usage

```
plot_result(est_result, ...)
```

Arguments

est_result	An S3 object of class 'result_no_nbd' or 'result_nbd' created by get_breaks().
...	Additional arguments.

Details

See hdchange() for examples.

Value

No return value. Presents the plot of the data and breaks.

Examples

```
# generate data
data_nbd <- sim_hdchange_nbd(n = 300,
p = 70,
nbd_info =
 list(
   (1:9), (2:31), (32:41), (42:70),
   (3:15), (16:35), (31:55)
 ),
sp_tp_break = rbind(c(2, 50), c(4, 150), c(2, 250)),
dist_info =
  list(dist = "t", dependence = "iid", param = 5),
jump_max = 1)

# construct nbd object
ts_nbd <- ts_hdchange(data_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1),
nbd_info =
 list(
   (1:9), (2:31), (32:41), (42:70),
   (3:15), (16:35), (31:55)
 ))

# Estimate the time-stamps of the breaks
est_result_nbd <- hdchange(ts_nbd)

# Plot the results
plot_result(est_result_nbd, nbd_index = 2)
pairs <- est_result_nbd$nbd_and_stamps_pair
```

```
time_stamps <- pairs[pairs[, 1] == 2, 2]
axis(1,
  at = time_stamps,
  labels = c("break 1", "break 2")
)
title(main = "Change-points estimation for neibourhood 2")
```

`plot_result.result_nbd` *Plot the time series and change-points*

Description

Plot the time series and change-points

Usage

```
## S3 method for class 'result_nbd'
plot_result(est_result, ..., nbd_index)
```

Arguments

`est_result`	An S3 object of class 'result_nbd' created by `get_breaks()`.
`...`	Additional arguments.
`nbd_index`	An integer indicating which neighbourhood to be plotted.

Details

See `hdchange()` for examples.

Value

No return value. Presents the plot of the data and breaks.

`plot_result.result_no_nbd` *Plot the time series and change-points*

Description

Plot the time series and change-points

Usage

```
## S3 method for class 'result_no_nbd'
plot_result(est_result, ...)
```

Arguments

est_result	An S3 object of class 'result_no_nbd' created by get_breaks().
...	Additional arguments.

Details

See hdchange() for examples.

Value

No return value. Presents the plot of the data and breaks.

---

sim_hdchange_nbd *Simulate data with neighbourhood*

Description

Simulate data with neighbourhood

Usage

```
sim_hdchange_nbd(
  n = 300,
  p = 70,
  nbd_info = list((1:9), (2:31), (32:41), (42:70), (3:15), (16:35), (31:55)),
  sp_tp_break = rbind(c(2, 50), c(4, 150), c(2, 250)),
  dist_info = list(dist = "normal", dependence = "iid", param = 1),
  jump_max = 1
)
```

Arguments

n	Number of time series observations.
p	Number of individual.
nbd_info	A list containing the neighbourhood information. See ts_hdchange().
sp_tp_break	A $K \times 2$ matrix indicating the spatial-temporal break location.
dist_info	A list specifying the distribution of the innovation.
jump_max	Maximum jump size of the breaks.

Details

'sp_tp_break' should be a $K \times 2$ matrix with first column indicating the neighbourhoods and the second column indicating the time stamps. For example, 'sp_tp_break = rbind(c(2, 50), c(4, 150), c(2, 250))' means that the second neighbourhood has two breaks taking place at $i = 50, 250$ and the fourth neighbourhood has one break taking place at $i = 150$.

'dist_info' should be a list containing the following items:

- dist: distribution of the innovations, either "normal" or "t".
- dependence: iid or $MA(\infty)$, either "iid" or "MA_inf".
- param = parameter of the distribution, standard deviation for normal distribution and degree of freedom for t distribution

'jump_max' is set equal in nbd case for convenience.

See `ts_hdchange()` for example.

Value

A $p \times n$ simulated data matrix.

Examples

```
data_nbd <- sim_hdchange_nbd(n = 300,
p = 70,
nbd_info =
 list(
   (1:9), (2:31), (32:41), (42:70),
   (3:15), (16:35), (31:55)
 ),
sp_tp_break = rbind(c(2, 50), c(4, 150), c(2, 250)),
dist_info =
  list(dist = "t", dependence = "iid", param = 5),
jump_max = 1)
```

`sim_hdchange_no_nbd` *Simulate data without neighbourhood*

Description

Simulate data without neighbourhood

Usage

```
sim_hdchange_no_nbd(
  n = 200,
  p = 30,
  S = 30,
  tau = c(40, 100, 160),
  dist_info = list(dist = "normal", dependence = "iid", param = 1),
  jump_max = c(2, 2, 1.5)
)
```

Arguments

n	Number of time series observations.
p	Number of individuals.
S	Number of individuals with jumps.
tau	An array of length K for time stamps for breaks.
dist_info	A list specifying the distribution of the innovation.
jump_max	An array of length K for jump sizes of the breaks.

Details

'dist_info' should be a list containing the following items:

- dist: distribution of the innovations, either "normal" or "t".
- dependence: iid or MA(∞), either "iid" or "MA_inf".
- param = parameter of the distribution, standard deviation for normal distribution and degree of freedom for t distribution

See `ts_hdchange()` for example.

Value

A $p \times n$ simulated data matrix.

Examples

```
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))
```

summary.result_nbd *Summarize the estimation results*

Description

Summarize the estimation results

Usage

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

Arguments

object	An S3 object of class 'result_nbd' created by get_breaks().
...	Additional arguments.

Details

See hdchange() for examples.

Value

No return value. Presents the summary of the test and estimation results.

summary.result_no_nbd *Summarize the estimation results*

Description

Summarize the estimation results

Usage

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

Arguments

object	An S3 object of class 'result_no_nbd' created by get_breaks().
...	Additional arguments.

Details

See hdchange() for examples.

Value

No return value. Presents the summary of the test and estimation results.

`test_existence` *Test the existence of change-points in the data*

Description

Test the existence of change-points in the data

Usage

```
test_existence(hdobj, display = TRUE)
```

Arguments

`hdobj`	An S3 object of class 'no_nbd' or 'nbd' generated by `ts_hdchange()`.
`display`	A logical. If 'display = TRUE', the test statistics and critical values will be printed.

Details

See `hdchange()` for examples.

Value

A list containing the following elements:

- 'test_stats' The test statistics $\boldsymbol{Q}_n$.
- 'critical_values' The critical values.
- 'stat_all' An array of $|V_i|_2^2$.
- 'critical_value_alpha' The threshold value ω depending on alpha.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
# generate data
data_no_nbd <- sim_hdchange_no_nbd(n = 200,
p = 30,
S = 30,
tau = c(40, 100, 160),
dist_info =
  list(dist = "normal", dependence = "MA_inf", param = 1),
jump_max = c(2, 2, 1.5))

# construct no_nbd object
ts_no_nbd <- ts_hdchange(data_no_nbd,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

test <- test_existence(ts_no_nbd, display = TRUE)
```

`ts_hdchange` *'no_nbd' or 'nbd' object construction*

Description

This function creates an S3 object of class 'no_nbd' or 'nbd' containing the initialising information supplied to the main function `hdchange()`. 'no_nbd' or 'nbd' are constructed depending on whether the neighbourhood information is provided. The resulting object will be used in the test and estimation functions.

Usage

```
ts_hdchange(
  data,
  window_size = 30,
  m = 8,
  h = 1,
  N_rep = 999,
  alpha = 1e-05,
  quantiles = c(0.01, 0.05, 0.1),
  nbd_info = NULL
)
```

Arguments

data	p by n data matrix, n = number of time series observations, p = cross-sectional dimension.
window_size	$window\_size = b \times n$, e.g. $n = 100$, $b = 30$.
m	Number of blocks in long-run variance estimation, 8 by default.
h	Parameter in long-run variance estimation, 1 by default.
N_rep	Number of repetitions in MC simulation.
alpha	A small positive number controlling for the threshold in break estimation.
quantiles	An array of quantiles for critical values.
nbd_info	A list containing the neighbourhood information, NULL by default indicating no neighbourhoods.

Details

'nbd_info' indicates the location of individuals in the data matrix. For example, 'nbd_info = list(c(1:10), c(25:35), c(7:18))' means that there are three neighbourhoods. The first neighbourhood contains from the 1st to 10th individuals and the same rule applies to the rest of neighbourhoods. The neighbourhoods are allowed to be overlapped. See also the illustrating example in hdchange().

Value

The return value is an S3 object of class 'no_nbd' or 'nbd'. It contains a list of the following items:

- data, m, h, N_rep, alpha, quantiles, and nbd_info are the same as in the arguments.
- n = number of time series observations.
- p = cross-sectional dimension.
- b = bandwith parameter $b = window\_size/n$.

References

Li, J., Chen, L., Wang, W. and Wu, W.B., 2022. ℓ^2 Inference for Change Points in High-Dimensional Time Series via a Two-Way MOSUM. *arXiv preprint arXiv:2208.13074*.

Examples

```
data <- covid_data

# No neighbourhood case
ts_no_nbd <- ts_hdchange(data,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1))

# Neighbourhood case
```

```
ts_nbd <- ts_hdchange(data,
window_size = 30,
m = 8,
h = 1,
N_rep = 999,
alpha = 1e-5,
quantiles = c(0.01, 0.05, 0.1),
nbd_info = list(c(1:10), c(25:35), c(7:18)))
```

Index