

# Package ‘LACPD’

June 19, 2020

**Type** Package

**Title** Locally adaptive change-point detection

**Version** 0.2.0

**Author** Authors@R: c(person("Mehdi", "Moradi", email = "m2.moradi@yahoo.com",  
role = c("aut", "cre")),  
person("Manuel", "Montesino-SanMartin", role = "ctb"),  
person("M. Dolores", "Ugarte", role = "ctb"),  
person("Ana F.", "Militino", role = "ctb")  
)

**Maintainer** Mehdi Moradi <m2.moradi@yahoo.com>

**Description** This package provides functions to detect abrupt changes in a set of time-ordered numerical observations using the LACPD technique.

**Depends** R (>= 3.3.0), trend, modifiedmk

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.0

## R topics documented:

lacpd_bcpw . . . . .	1
lacpd_cs . . . . .	4
lacpd_mk . . . . .	6
lacpd_mmkh . . . . .	8
lacpd_mmky . . . . .	10
lacpd_pwmk . . . . .	12
lacpd_tfpwmk . . . . .	14

<b>Index</b>	<b>17</b>
--------------	-----------

lacpd\_bcpw

*Locally change-point detection using the bcpw approach***Description**

Locally change-point detection after accommodating the bcpw approach in LACPD procedure

**Usage**

```
lacpd_bcpw(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

**Arguments**

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

**Details**

This technique accommodates the [bcpw](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector x.

In the tails of x, since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of x is n. The argument k is used to set the number of data in the sides of the target point when looking for change-points. n/k is the number of points on each side we consider. For instance, if n=300 and k=10, this means we consider 30 observations before and 30 after when locally detecting changes.

If leave=TRUE, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If `k` is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the z statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding p.value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained z statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained p-values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and `k` is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given `k`

`attr("allzs")`: a list which retrieves the obtained z statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given `k`

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given `k`

`attr("allps")`: a list which retrieves the obtained p-values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given `k`

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

## See Also

[bcpw](#), [p.adjust](#)

## Examples

```
x <- rnorm(50)
Z <- lacpd_bcpw(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_cs

*Locally change-point detection using the Cox-Stuart test*


---

## Description

Locally change-point detection after accommodating the Cox-Stuart test in LACPD procedure

## Usage

```
lacpd_cs(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

## Arguments

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

## Details

This technique accommodates the [cs.test](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector x.

In the tails of x, since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

**See Also**

[cs.test](#), [p.adjust](#)

**Examples**

```
x <- rnorm(50)
Z <- lacpd_cs(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_mk

*Locally change-point detection using the Mann-Kendall test*


---

**Description**

Locally change-point detection after accommodating the Mann-Kendall test in LACPD procedure

**Usage**

```
lacpd_mk(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

**Arguments**

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

## Details

This technique accommodates the [mk.test](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector  $x$ .

In the tails of  $x$ , since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

## See Also

[mk.test](#), [p.adjust](#)

## Examples

```
x <- rnorm(50)
Z <- lacpd_mk(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_mmkh

*Locally change-point detection using the mmkh approach*


---

## Description

Locally change-point detection after accommodating the mmkh approach in LACPD procedure

## Usage

```
lacpd_mmkh(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

## Arguments

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>



## Details

This technique accommodates the [mmkh](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector  $x$ .

In the tails of  $x$ , since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

## See Also

[mmkh](#), [p.adjust](#)

## Examples

```
x <- rnorm(50)
Z <- lacpd_mmkh(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_mmky

*Locally change-point detection using the mmky approach*

---

## Description

Locally change-point detection after accommodating the mmky approach in LACPD procedure

## Usage

```
lacpd_mmky(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

## Arguments

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

## Details

This technique accommodates the [mmky](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector  $x$ .

In the tails of  $x$ , since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

## See Also

[mmky](#), [p.adjust](#)

## Examples

```
x <- rnorm(50)
Z <- lacpd_mmky(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_pwmk

*Locally change-point detection using the pwmk approach*

---

## Description

Locally change-point detection after accommodating the pwmk approach in LACPD procedure

## Usage

```
lacpd_pwmk(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

## Arguments

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

## Details

This technique accommodates the [pwmk](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector  $x$ .

In the tails of  $x$ , since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

## References

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

## See Also

[pwmk](#), [p.adjust](#)

## Examples

```
x <- rnorm(50)
Z <- lacpd_pwmk(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```

---

lacpd\_tfpwmk

*Locally change-point detection using the tfpwmk approach*

---

## Description

Locally change-point detection after accommodating the tfpwmk approach in LACPD procedure

## Usage

```
lacpd_tfpwmk(
  x,
  m = 1,
  k = 2,
  blow = 0.1,
  bup = (1 - blow),
  leave = FALSE,
  adjust = FALSE,
  history = FALSE,
  ...
)
```

## Arguments

x	a numeric vector
m	number of times to sub-sample
k	single number or numeric vector proportional to the number of points on each side of the target point. See details
blow	fraction of observations (0-1) at the beginning of the time-series not considered for change detection
bup	similar to blow, but for the end of the time series. Default is 1-blow
leave	if TRUE, the function uses the leave-one-out technique when looking for changes
adjust	if TRUE, p-value will be adjusted by methods in <a href="#">p.adjust</a>
history	if TRUE, it maintains the stepwise results when k is a vector
...	arguments passed to <a href="#">p.adjust</a>

## Details

This technique accommodates the [tfpwmk](#) trend detection method in the LACPD procedure of Moradi et al. (2020) to look for potential change-points in the numerical vector  $x$ .

In the tails of  $x$ , since there are not enough data points before/after the target points, the method uses sub-sampling, wherein it moderates the effect of sub-sampling by iteration.

Assume the length of  $x$  is  $n$ . The argument  $k$  is used to set the number of data in the sides of the target point when looking for change-points.  $n/k$  is the number of points on each side we consider. For instance, if  $n=300$  and  $k=10$ , this means we consider 30 observations before and 30 after when locally detecting changes.

If `leave=TRUE`, the function removes the target point when checking for possible changes at the target point.

If `adjust=TRUE`, methods such as "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none" can be passed through function, e.g. `method="BY"`.

If  $k$  is a vector of numbers, then the function returns a result based on adaptive sliding windows which is an average result obtained from different windows.

## Value

`cp`: the index of the most probable change point in the time series

`z`: the  $z$  statistics of LACPD

`magnitude`: the magnitude of change

`p.value`: the corresponding  $p$ -value

`s`: the time-periods that the function has looked for potential changes

Attributes:

`attr("zs")`: retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes

`attr("ps")`: retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes

`attr("mags")`: retrieves the magnitude of change at the time-periods the function has looked for potential changes

if `history=TRUE` and  $k$  is a numeric vector, then the following attributes can also be retrieved.

`attr("history")`: a dataframe containing the results (`cp`, `magnitude`, `Z`, and `p.value`) based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allzs")`: a list which retrieves the obtained  $z$  statistics at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allmags")`: a list which retrieves the magnitude of change at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

`attr("allps")`: a list which retrieves the obtained  $p$ -values at the time-periods the function has looked for potential changes based on different adaptive sliding windows which are permutations of the given  $k$

## Author(s)

Mehdi Moradi <m2.moradi@yahoo.com>, Manuel Montesino-SanMartin <manuel.montesino@unavarra.es>

**References**

Moradi, M., Montesino-SanMartin, M., Ugarte, M. D., and Militino, A. F. (2020). Locally adaptive change-point detection with applications to remote sensing and land use changes.

**See Also**

[tfpwmk](#), [p.adjust](#)

**Examples**

```
x <- rnorm(50)
Z <- lacpd_tfpwmk(x,m=10,k=3)
plot(Z$s,attr(Z,"zs"),type="l",ylab = "Z",xlab = "time")
plot(Z$s,attr(Z,"mags"),type="l",ylab = "Z",xlab = "time")
```



# Index

bcpw, [2](#), [3](#)

cs.test, [4](#), [5](#)

lacpd\_bcpw, [1](#)

lacpd\_cs, [4](#)

lacpd\_mk, [6](#)

lacpd\_mmkh, [8](#)

lacpd\_mmky, [10](#)

lacpd\_pwmk, [12](#)

lacpd\_tfpwmk, [14](#)

mk.test, [6](#), [8](#)

mmkh, [9](#), [10](#)

mmky, [11](#), [12](#)

p.adjust, [2–6](#), [8](#), [10](#), [12](#), [14](#), [16](#)

pwmk, [13](#), [14](#)

tfpwmk, [15](#), [16](#)