Package 'binnednp'

June 7, 2019

Type Package

Title Nonparametric Estimation for Interval-Grouped Data

Version 0.4.0

Author Daniel Barreiro Ures, Basilio B. Fraguela, Ramón Doallo Biempica, Ricardo Cao, Mario Francisco-Fernández, Miguel Reyes

Maintainer Daniel Barreiro Ures <dbu5293@gmail.com>

Description Kernel density and distribution estimation for interval-grouped data (Reyes, Francisco-Fernandez and Cao 2016, 2017) <doi:10.1080/10485252.2016.1163348>, <doi:10.1007/s11749-017-0523-9>, (Gonzalez-Andujar, Francisco-Fernandez, Cao, Reyes, Urbano, Forcella and Bastida 2016) <doi:10.1111/wre.12216> and nonparametric estimation of seedling emergence indices (Cao, Francisco-Fernandez, Anand, Bastida and Gonzalez-Andujar 2011) <doi:10.1017/S002185961100030X>.

Depends R (>= 2.10)

License GPL-3

Encoding UTF-8

LazyData true

Imports Rcpp (>= 0.12.12), kedd, nor1mix, fitdistrplus, grDevices, graphics, stats, mclust, Rdpack, parallel, doParallel, foreach

RdMacros Rdpack

LinkingTo Rcpp

RoxygenNote 6.1.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-06-07 17:50:03 UTC

R topics documented:

`anv.binned` *ANOVA in kernel distribution estimation with binned data using bootstrap.*

Description

ANOVA in kernel distribution estimation with binned data using bootstrap.

Usage

```
anv.binned(n, y, trt.w, abs.values = FALSE, B = 500)
```

Arguments

`n`	Vector of positive integers. Sizes of the complete samples corresponding to each treatment.
`y`	Vector. Observed values. They define the extremes of the sequence of intervals in which data is binned.
`trt.w`	Matrix. Proportion of observations within each interval. Each column corresponds to a different treatment.
`abs.values`	Logical. Indicates if the values of trt.w are given in absolute (TRUE) or relative (FALSE) format.
`B`	Positive integer. Number of bootstrap replicates used to compute the confidence bands.

Details

ANOVA for interval-grouped data.

Value

p-value of the test.

References

Reyes M (2015). *Statistical methods for studying emergence curves in weed science*. Ph.D. thesis, Universidade da Coruna, Spain.

bw.dens.binned *Bootstrap and plug-in bandwidth selectors for kernel density estimation with binned data.*

Description

Bootstrap and plug-in bandwidth selectors for kernel density estimation with binned data.

Usage

```
bw.dens.binned(n, y, w, ni, gboot, pilot.type = 3, hn = 100,
  plugin.type = "N", confband = FALSE, alpha = 0.05, B = 1000,
  plot = TRUE, print = TRUE, model, parallel = FALSE,
  pars = new.env())
```

Arguments

n	Positive integer. Size of the complete sample.
y	Vector. Observed values. They define the extremes of the sequence of intervals in which data is binned.
w	Vector. Proportion of observations within each interval.
ni	Vector. Number of observations within each interval.
gboot	Positive real number. Pilot bandwidth for the bootstrap bandwidth selector.
pilot.type	1, 2 or 3. If gboot is missing, pilot bandwidth for the bootstrap bandwidth selector is automatically selected using methods 1, 2 or 3. Defaults to 3. See details for more information.
hn	Positive integer. Size of the grid of bandwidths for which MISE will be approximated. Defaults to 100.
plugin.type	Character. If plugin.type = "N", normality is assumed at the last step when calculating the plug-in bandwidth. If plugin.type = "A", parameter at last step is estimated nonparametrically using gplugin as bandwidth. Otherwise, the unknown parameter is estimated fitting a normal mixture. Defaults to type = "N".
confband	Logical. If TRUE, bootstrap confidence bands are constructed for the density function. Defaults to FALSE.
alpha	Real number between 0 and 1. Significance level for the bootstrap confidence bands. Defaults to 0.05.
B	Positive integer. Number of bootstrap resamples used when constructing confidence bands. Defaults to 1000.
plot	Logical. If TRUE, kernel density estimators are plotted along with (optional) bootstrap confidence bands. Defaults to TRUE.
print	Logical. If TRUE and confband is TRUE, the percentage of bootstrap resamples already evaluated is printed. Defaults to TRUE.

model	Character. Name of the parametric family of distributions to be fitted for the grouped sample. Parameters are estimated by maximum likelihood.
parallel	Logical. If TRUE, confidence bands are estimated using parallel computing with sockets.
pars	Environment. Needed for the well functioning of the script. DO NOT modify this argument.

Details

If pilot.type = 1, an heuristic rule is used for calculating the pilot bandwidth. It's not recommended when population's density function is suspected to be highly multimodal.

If pilot.type = 2, the pilot bandwidth is such that the kernel density estimator with bandwidth gboot approximates the histogram of the grouped sample minimizing the residual sum of squares. If pilot.type = 3, a penalty is imposed on the curvature of the kernel density estimator with bandwidth gboot. The penalty parameter is selected as to best approximate the curvature of the true density.

Value

A list with components

h_boot	Bootstrap bandwidth selector.
h_plugin	Plug-in bandwidth selector.

References

Reyes M (2015). *Statistical methods for studying emergence curves in weed science*. Ph.D. thesis, Universidade da Coruna, Spain.

Reyes M, Francisco-Fernandez M, Cao R (2016). "Nonparametric kernel density estimation for general grouped data." *Journal of Nonparametric Statistics*, **2**, 235–249.

Reyes M, Francisco-Fernandez M, Cao R (2017). "Bandwidth selection in kernel density estimation for interval-grouped data." *TEST*, **26**, 527–545.

Examples

```
set.seed(1)
n <- 200 #complete sample size
k <- 30 #number of intervals
x <- rnorm(n,6,1) #complete sample
y <- seq(min(x)-0.2,max(x)+0.2,len=k+1) #intervals
w <- c(sapply(2:k,function(i)sum( x<y[i]&x>=y[i-1] )), sum(x<=y[k+1]&x>=y[k]) )/n #proportions
bw.dens.binned(n,y,w,plot=FALSE)
```

bw.dist.binned *Plug-in bandwidth selector for kernel distribution estimation and binned data.*

Description

Plug-in bandwidth selector for kernel distribution estimation and binned data.

Usage

```
bw.dist.binned(n, y, w, ni, gplugin, type = "N", confband = F,
  B = 1000, alpha = 0.05, plot = TRUE, print = TRUE, model,
  parallel = FALSE, pars = new.env())
```

Arguments

n	Positive integer. Size of the complete sample.
y	Vector. Observed values. They define the extremes of the sequence of intervals in which data is binned.
w	Vector. Proportion of observations within each interval.
ni	Vector. Number of observations within each interval.
gplugin	Positive real number. Pilot bandwidth. If missing, rule-of-thumb bandwidth is considered.
type	Character. If type = "N", normality is assumed at the last step when calculating the plug-in bandwidth. If type = "A", parameter at last step is estimated non-parametrically using gplugin as bandwidth. Otherwise, the unknown parameter is estimated fitting a normal mixture. Defaults to type = "N".
confband	Logical. If TRUE, bootstrap confidence bands for the distribution are constructed. Defaults to FALSE.
B	Number of bootstrap resamples. Defaults to 1000.
alpha	Significance level for the bootstrap confidence bands. Defaults to 0.05.
plot	Logical. If TRUE, results are plotted. Defaults to TRUE.
print	Logical. If TRUE, script current status is printed. Defaults to TRUE.
model	Character. Name of the parametric family of distributions to be fitted for the grouped sample. Parameters are estimated by maximum likelihood.
parallel	Logical. If TRUE, confidence bands are estimated using parallel computing with sockets.
pars	Environment. Needed for the well functioning of the script. DO NOT modify this argument.

Value

A list with components

h	Plug-in bandwidth.
Fh	Function. Kernel distribution estimator with bandwidth h.
confband	(Optional) Bootstrap confidence bands for the distribution function.

References

Reyes M (2015). *Statistical methods for studying emergence curves in weed science*. Ph.D. thesis, Universidade da Coruna, Spain.

Gonzalez-Andujar J, Francisco-Fernandez M, Cao R, Reyes M, Urbano J, Forcella F, Bastida F (2016). "A comparative study between nonlinear regression and nonparametric approaches for modeling Phalaris paradoxa seedling emergence." *Weed Research*, **56**, 367–376.

Examples

```
set.seed(1)
n <- 200 #complete sample size
k <- 30 #number of intervals
x <- rnorm(n,6,1) #complete sample
y <- seq(min(x)-0.2,max(x)+0.2,len=k+1) #intervals
w <- c(sapply(2:k,function(i)sum( x<y[i]&x>=y[i-1] )), sum(x<=y[k+1]&x>=y[k]) )/n #proportions
bw.dist.binned(n,y,w,plot=FALSE)
```

---

bw.dist.binned.boot *Bootstrap bandwidth selector for kernel distribution estimation and binned data.*

Description

Bootstrap bandwidth selector for kernel distribution estimation and binned data.

Usage

```
bw.dist.binned.boot(n, y, w, ni, g, pilot.type = 2, nit = 10,
  confband = FALSE, B = 1000, alpha = 0.05, print = TRUE,
  plot = TRUE, parallel = FALSE, pars = new.env())
```

Arguments

n	Positive integer. Size of the complete sample.
y	Vector. Observed values. They define the extremes of the sequence of intervals in which data is binned.
w	Vector. Proportion of observations within each interval.

ni	Vector. Number of observations within each interval.
g	Positive real number. Pilot bandwidth. If missing, plug-in N bandwidth for the distribution is considered.
pilot.type	1 or 2. If g is missing, pilot bandwidth for the bootstrap bandwidth selector is automatically selected using methods 1 or 2. Defaults to 1. See details for more information.
nit	Positive integer. Number of iterations in the dichotomy algorithm for the estimation of the bootstrap bandwidth.
confband	Logical. If TRUE, bootstrap confidence bands are constructed for the estimator. Defaults to FALSE.
B	Positive integer. Number of bootstrap resamples used for the construction of the confidence bands. Defaults to 1000.
alpha	Real number between 0 and 1. Significance level for the confidence bands. Defaults to 0.05
print	Logical. If TRUE, script current status is printed. Defaults to TRUE.
plot	Logical. If TRUE, results are plotted. Defaults to FALSE.
parallel	Logical. If TRUE, confidence bands are estimated using parallel computing with sockets.
pars	Environment. Needed for the well functioning of the script. DO NOT modify this argument.

Details

If pilot.type = 1, plug-in bandwidth for the distribution is considered as pilot bandwidth for the bootstrap selector.

If pilot.type = 2, the pilot bandwidth is such that the kernel distribution estimator with bandwidth g approximates the empirical distribution of the grouped sample minimizing the residual sum of squares. Also, a penalty is imposed on the global slope of the kernel density estimator with bandwidth g. The penalty parameter is selected as to best approximate the global slope of the true density.

Value

A list with components:

h	Bootstrap bandwidth for the distribution function.
Fh	Function. Kernel distribution estimator with bandwidth h.
confband (optional)	Matrix. Its columns contain the bootstrap confidence bands for the estimator.

References

Reyes M (2015). *Statistical methods for studying emergence curves in weed science*. Ph.D. thesis, Universidade da Coruna, Spain.

Examples

```
set.seed(1)
n <- 200 #complete sample size
k <- 30 #number of intervals
x <- rnorm(n,6,1) #complete sample
y <- seq(min(x)-0.2,max(x)+0.2,len=k+1) #intervals
w <- c(sapply(2:k,function(i)sum( x<y[i]&x>=y[i-1] )), sum(x<=y[k+1]&x>=y[k]) )/n #proportions
bw.dist.binned.boot(n,y,w,plot=FALSE)
```

`emergence.indices` *Nonparametric estimates of indices measuring the global slope and curvature of the density function for binned data.*

Description

Nonparametric estimates of indices measuring the global slope and curvature of the density function for binned data.

Usage

```
emergence.indices(n, y, w, ni, hseq, hn = 200, nmix = 4, B = 500,
  method = "np", last.iter.np = F, confint = FALSE, B.conf = 1000,
  alpha = 0.05, print = TRUE, parallel = FALSE, pars = new.env())
```

Arguments

`n`	Positive integer. Size of the complete sample.
`y`	Vector. Observed values. They define the extremes of the sequence of intervals in which data is binned.
`w`	Vector. Proportion of observations within each interval.
`ni`	Vector. Number of observations within each interval.
`hseq`	Vector. Grid of bandwidths for which MSE is approximated through bootstrap. If missing, a grid of size hn is considered.
`hn`	Positive integer. Size of the grid of bandwidths for which MSE will be approximated. Defaults to 200.
`nmix`	Positive integer. Maximum number of components for the normal mixture model. Defaults to 4.
`B`	Positive integer. Number of bootstrap resamples used to find the bandwidth that minimizes MSE. Defaults to 500.
`method`	Character. If method="np", resamples are taken from kernel density estimator with pilot bandwidth. If method="mix", a normal mixture pilot model is considered. If method="plugin", plug-in estimates are returned. Defaults to "np".

last.iter.np	Logical. If FALSE, normality is assumed at the last step when calculating the plug-in bandwidth. Otherwise, rule-of-thumb selector is used. Defaults to FALSE.
confint	Logical. If TRUE, bootstrap confidence intervals are constructed for the indices. Defaults to FALSE.
B.conf	Positive integer. Number of bootstrap resamples considered to construct the confidence intervals.
alpha	Real number between 0 and 1. Significance level considered for the confidence intervals.
print	Logical. If TRUE, status of the script and results are printed. Defaults to TRUE.
parallel	Logical. If TRUE, confidence bands are estimated using parallel computing with sockets.
pars	Environment. Needed for the well functioning of the script. DO NOT modify this argument.

Value

Nonparametric estimates of the indices and (optional) confidence intervals.

References

Cao R, Francisco-Fernandez M, Anand A, Bastida F, Gonzalez-Andujar J (2011). "Computing statistical indices for hydrothermal times using weed emergence data." *Journal of Agricultural Science*, **149**, 701–712.

Reyes M, Francisco-Fernandez M, Cao R (2017). "Bandwidth selection in kernel density estimation for interval-grouped data." *TEST*, **26**, 527–545.

Examples

```
set.seed(1)
n <- 200 #complete sample size
k <- 30 #number of intervals
x <- rnorm(n,6,1) #complete sample
y <- seq(min(x)-0.2,max(x)+0.2,len=k+1) #intervals
w <- c(sapply(2:k,function(i)sum( x<y[i]&x>=y[i-1] )), sum(x<=y[k+1]&x>=y[k]) )/n #proportions
emergence.indices(n,y,w)
```

Index