Package 'overlapping'

November 2, 2026

Type Package

Title Estimation of Overlapping in Empirical Distributions

Version 2.6

Date 2026-10-31

Author Massimiliano Pastore [aut, cre],
Pierfrancesco Alaimo Di Loro [ctb],
Marco Mingione [ctb],
Antonio Calcagni' [ctb]

Maintainer Massimiliano Pastore <massimiliano.pastore@unipd.it>

Description Functions for estimating the overlapping area of two or more kernel density estimations from empirical data.

Depends R (>= 3.0.0), ggplot2, testthat

License GPL-2

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2026-11-02 10:50:02 UTC

Contents

boot.overlap 2
check.missing 3
final.plot 4
overlap 5
ovmult 7
paired.permutations 9
perm.pairs 10
perm.test 11

Index **13**

`boot.overlap` *Nonparametric Bootstrap to estimate the overlapping area*

Description

Resampling via non-parametric bootstrap to estimate the overlapping area between two or more kernel density estimations from empirical data.

Usage

```
boot.overlap( x, B = 1000, pairsOverlap = FALSE, ... )
```

Arguments

`x`	a list of numerical vectors to be compared (each vector is an element of the list).
`B`	integer, number of bootstrap draws.
`pairsOverlap`	logical, if `TRUE`, available only when the list `x` contains more than two elements, it returns the overlapped area relative to each pair of distributions.
`...`	options, see function `overlap` for details.

Details

If the list `x` contains more than two elements (i.e., more than two distributions) it computes the bootstrap overlapping measure between all the q paired distributions. For example, if `x` contains three elements then $q = 3$; if `x` contains four elements then $q = 6$.

Value

It returns a list containing the following components:

`OVboot_stats`	a data frame $q \times 3$ where each row contains the following statistics: `estOV`, estimated overlapping area, $\hat{\eta}$; `bias`, difference between the expected value over the bootstrap samples and the observed overlapping area: $E(\hat{\eta}^*) - \hat{\eta}$; `se`, bootstrap standard error $\sigma_{\hat{\eta}}$.
`OVboot_dist`	a matrix with `B` rows (bootstrap replicates) and q columns (depending on the number of elements of `x`); each column is a boostrap distribution of the corresponding overlapping measure.

Note

Call function `overlap`.

Thanks to Jeremy Vollen for suggestions.

Author(s)

Massimiliano Pastore

References

Pastore, M. (2018). Overlapping: a R package for Estimating Overlapping in Empirical Distributions. The Journal of Open Source Software, 3 (32), 1023. doi:10.21105/joss.01023

Pastore, M., Calcagnì, A. (2019). Measuring Distribution Similarities Between Samples: A Distribution-Free Overlapping Index. Frontiers in Psychology, 10:1089. doi:10.3389/fpsyg.2019.01089

Examples

```
set.seed(1)
x <- list(Y1=rnorm(100), Y2=rnorm(90,1,3))
## bootstrapping
out <- boot.overlap( x, B = 20, pairsOverlap = TRUE )
out$OVboot_stats

# bootstrap quantile intervals
quantile( out$OVboot_dist, probs = c(.05, .9) )

# plot of bootstrap distributions
Y <- data.frame( values = c(out$OVboot_dist) )
ggplot( Y, aes( values )) + geom_density()

# missing data
x$Y1[1:10] <- NA
out <- boot.overlap( x, B = 20, pairsOverlap = TRUE )
out$OVboot_stats

# ++++++++++++++++++++++++
set.seed(20150605)
x <- list(X1=rnorm(100), X2=rt(50,8), X3=rchisq(80,2))

## bootstrapping
out <- boot.overlap( x, B = 10, pairsOverlap = TRUE )
out$OVboot_stats

# bootstrap quantile intervals
apply( out$OVboot_dist, 2, quantile, probs = c(.05, .9) )

# plot of bootstrap distributions
Y <- stack( data.frame( out$OVboot_dist ))
ggplot( Y, aes( values )) + facet_wrap( ~ind ) + geom_density()
```

`check.missing` *Check for missing data*

Description

Remove `NA`s from the `x` list.

Usage

```
check.missing( x )
```

Arguments

x	a list of numerical vectors to be compared (each vector is an element of the list).

Details

If list x contains NAs, it returns a warning and removes missing cases using na.omit().

Value

It returns the list x without NAs.

Note

Internal function.

Author(s)

Massimiliano Pastore

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(10), X2=rt(10,8))
check.missing( x )

x$X1[1:5] <- NA
check.missing( x )
```

---

final.plot *Final plot*

Description

Graphical representation of the estimated densities along with the overlapping area.

Usage

```
final.plot( x, pairs = FALSE, boundaries = NULL )
```

Arguments

x	a list of numerical vectors to be compared; each vector is an element of the list, see overlap.
pairs	logical, if TRUE (and x contains more than two elements) produces pairwise plots.
boundaries	an optional vector indicating the minimum and the maximum over a predefined subset of the support of the empirical densities.

Details

It requires the package ggplot2.

Note

The output plot can be customized using the ggplot2 rules, see example below.

Author(s)

Massimiliano Pastore

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(100),X2=rt(50,8),X3=rchisq(80,2))
final.plot(x)
final.plot(x, pairs = TRUE)

# customizing plot
final.plot(x) + scale_fill_brewer() + scale_color_brewer()
final.plot(x) + theme(text=element_text(size=15))
```

---

overlap *Estimate the overlapping measure.*

Description

It returns the overlapped estimated area between two or more kernel density estimations from empirical data. The overlapping measure can be computed either as the integral of the minimum between two densities (type = "1") or as the proportion of overlapping area between two densities (type = "2"). In the last case, the integral of the minimum between two densities is divided by the integral of the maximum of the two densities.

Usage

```
overlap( x, nbins = 1024, type = c( "1", "2" ),
    pairsOverlap = TRUE, plot = FALSE, boundaries = NULL,
    get_xpoints = FALSE, ... )
```

Arguments

x	a list of numerical vectors to be compared (each vector is an element of the list).
nbins	number of equally spaced points through which the density estimates are compared; see density for details.
type	character, type of index. If type = "2" returns the proportion of the overlapped area between two or more densities, see Details.
pairsOverlap	logical, if TRUE (default) returns the overlapped area relative to each pair of distributions.
plot	logical, if TRUE, the final plot of estimated densities and overlapped areas is produced.
boundaries	an optional vector indicating the minimum and the maximum over a predefined subset of the support of the empirical densities, see Details.
get_xpoints	logical, if TRUE returns a vector where the abscissas represent the points of intersection among the densities. Note: it works only if pairsOverlap = FALSE.
...	optional arguments to be passed to the function density.

Details

When dealing with two densities: type = "1" corresponds to the integral of the minimum between the two densities; type = "2" corresponds to the proportion of the overlapped area over the total area.

If the list x contains more than two elements (i.e. more than two distributions) it computes both the multiple and the pairwise overlapping among all distributions.

If plot = TRUE all the overlapped areas are plotted. It requires ggplot2.

The optional vector boundaries has to contain two numbers for the empirical minimum and maximum of the overlapped area. See examples below.

Value

It returns a list containing the following components:

OV	estimate of the overlapped area; if x contains more than two elements then a vector of estimates is returned.
xpoints	a list of intersection points (in abscissa) among the densities (if get_xpoints = TRUE).
OVpairs	the estimates of overlapped areas for each pair of densities (only if x contains more than two elements).

Note

Call function ovmult.

If list x contains NAs it returns a warning and deletes missing cases using na.omit().

Author(s)

Massimiliano Pastore, Pierfrancesco Alaimo Di Loro, Marco Mingione

References

Pastore, M. (2018). Overlapping: a R package for Estimating Overlapping in Empirical Distributions. The Journal of Open Source Software, 3 (32), 1023. doi:10.21105/joss.01023

Pastore, M., Calcagnì, A. (2019). Measuring Distribution Similarities Between Samples: A Distribution-Free Overlapping Index. Frontiers in Psychology, 10:1089. doi:10.3389/fpsyg.2019.01089

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(100), X2=rt(50,8), X3=rchisq(80,2))
overlap(x, plot=TRUE)

# including boundaries
x <- list(X1=runif(100), X2=runif(100,.5,1))
overlap(x, plot=TRUE, boundaries=c(.5,1))

x <- list(X1=runif(100), X2=runif(50), X3=runif(30))
overlap(x, plot=TRUE, boundaries=c(.1,.9))

# changing kernel
overlap(x, plot=TRUE, kernel="rectangular")

# normalized overlap
N <- 1e5
x <- list(X1=runif(N),X2=runif(N,.5))
overlap(x)
overlap(x, type = "2")

# missing data
x$X1[1:10] <- NA
overlap(x)
```

`ovmult` *Multiple overlapping estimation*

Description

It gives the overlap area between two or more kernel density estimations from empirical data.

Usage

```
ovmult( x, nbins = 1024, type = c( "1", "2" ),
    boundaries = NULL, get_xpoints = FALSE, ... )
```

Arguments

x	a list of numerical vectors to be compared (each vector is an element of the list).
nbins	number of equally spaced points through which the density estimates are compared; see density for details.
type	character, type of index. If type = "2" returns the proportion of the overlapped area between two or more densities, see overlap.
boundaries	an optional vector indicating the minimum and the maximum over a predefined subset of the support of the empirical densities.
get_xpoints	logical, if TRUE returns a vector where the abscissas represent the points of intersection among the densities. Note: it works only if pairsOverlap = FALSE.
...	optional arguments to be passed to the function density.

Details

If the list x contains more than two elements (i.e. more than two distributions) it computes multiple overlap measures.

The optional vector boundaries has to contain two numbers for the empirical minimum and maximum of the overlapped area. See examples below.

Value

It returns the value of overlapped area.

Note

Called from the function overlap.

If list x contains NAs it returns a warning and deletes missing cases using na.omit().

Author(s)

Pierfrancesco Alaimo Di Loro, Marco Mingione, Massimiliano Pastore

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(100), X2=rt(50,8), X3=rchisq(80,2))
ovmult(x)
ovmult(x, type = "2" )

# including boundaries
x <- list(X1=runif(100), X2=runif(100,.5,1))
ovmult(x, boundaries=c( 0, .8 ))

x <- list(X1=runif(100), X2=runif(50), X3=runif(30))
ovmult(x, boundaries=c( .2, .8 ))

# changing kernel
ovmult(x, kernel="rectangular")
```

```
# missing data
x$X3[1:10] <- NA
ovmult(x)
```

`paired.permutations` *Permutation for non-independent data*

Description

Randomly permute the paired data.

Usage

```
paired.permutations( x )
```

Arguments

`x`	a list of two numerical vectors to be compared (each vector is an element of the list).

Details

When the data are paired (i.e. non-independent observations), the exchange takes place within each pair. Note: if the data are paired, the elements in the list must be of the same length and in the same order.

Value

It returns a list with paired elements of `x` randomly permuted.

Note

Internal function called by `perm.test`.

Author(s)

Massimiliano Pastore

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(10), X2=rt(10,8))
paired.permutations( x )
```

perm.pairs *Paired permutation*

Description

Perform a random permutation of the data list.

Usage

```
perm.pairs( x, paired = FALSE )
```

Arguments

x	a list of numerical vectors to be compared (each vector is an element of the list).
paired	logical; set to TRUE if the data are paired (i.e. non-independent samples).

Value

It returns a list with paired elements of x randomly permuted.

Note

Internal function called by perm.test.

Author(s)

Massimiliano Pastore

Examples

```
set.seed(20150605)
# independent samples
x <- list(X1=rnorm(10), X2=rt(15,8))
perm.pairs( x )

x <- list(X1=rnorm(10), X2=rt(15,8), X3=rchisq(12,3))
perm.pairs( x )

# non-independent samples
x <- list(X1=rnorm(10), X2=rt(10,8))
perm.pairs( x, paired = TRUE )
```

perm.test *Permutation test on the (non-)overlapping area*

Description

Perform a permutation test on the overlapping index.

Usage

```
perm.test( x, paired = FALSE, B = 1000,
          return.distribution = FALSE, ... )
```

Arguments

x	a list of numerical vectors to be compared (each vector is an element of the list).
paired	logical; set to TRUE if the data are paired (i.e. non-independent samples), see details.
B	integer, number of permutation replicates.
return.distribution	logical, if TRUE it returns the distribution of permuted Z statistics.
...	options, see function overlap for details.

Details

It performs a permutation test of the null hypothesis that there is no difference between the two distributions, i.e. the overlapping index (η) is one, or the non-overlapping index ($1 - \eta = \zeta$) is zero.

By default (paired = FALSE), it is assumed that the values in the list are independent observations. If the data are paired (i.e. non-independent observations; paired = TRUE), the exchange takes place within each pair. Note: if the data are paired, the elements in the list must be of the same length and in the same order.

Value

It returns a list containing the following components:

Zobs	the observed values of non-overlapping index, i.e. 1-η.
pval	p-values.
Zperm	the permutation distributions.

Warning

Currently, it only runs the permutation test on two groups at a time. If x contains more than 2 elements, it performs all paired permutation tests.

Note

Call function overlap.

Author(s)

Massimiliano Pastore

References

Calignano, G., Perugini, A., Nucci, M., Finos, L., Pastore M. (2026). How do my distributions differ? Significance testing for the Overlapping Index using Permutation Test. Psychonomic Bulletin & Review, 33:189. doi:10.3758/s1342302602946z

Examples

```
set.seed(20150605)
x <- list(X1=rnorm(100), X2=rt(50,8))

## not run: this example take several minutes
## permutation test
# out <- perm.test( x, return.distribution = TRUE )
# out$pval
# plot( density( out$Zperm ) )
# abline( v = out$Zobs )

x <- list(X1=rnorm(100), X2=rt(50,8), X3=rchisq(75,3))
# out <- perm.test( x )
# out$pval
```

Index

∗ **utility**
boot.overlap, 2
check.missing, 3
final.plot, 4
overlap, 5
ovmult, 7
paired.permutations, 9
perm.pairs, 10
perm.test, 11

boot.overlap, 2

check.missing, 3

density, *6*, *8*

final.plot, 4

overlap, *2*, *5*, 5, *8*, *11*
ovmult, *6*, 7

paired.permutations, 9
perm.pairs, 10
perm.test, *9*, *10*, 11