Package ‘OVL.CI’

May 7, 2026

Type Package

Title Inference on the Overlap Coefficient

Version 0.1.1

Maintainer Alba M. Franco-Pereira <albfranc@ucm.es>

Description Provides functions to construct confidence intervals for the Overlap Coefficient (OVL). OVL measures the similarity between two distributions through the overlapping area of their distribution functions. Given its intuitive description and ease of visual representation by the straightforward depiction of the amount of overlap between the two corresponding histograms based on samples of measurements from each one of the two distributions, the development of accurate methods for confidence interval construction can be useful for applied researchers. Implements methods based on the work of Franco-Pereira, A.M., Nakas, C.T., Reiser, B., and Pardo, M.C. (2021) <doi:10.1177/09622802211046386> as well as extensions for multimodal distributions proposed by Alcaraz-Peñalba, A., Franco-Pereira, A., and Pardo, M.C. (2025) <doi:10.1007/s10182-025-00545-2>.

License GPL-2

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 7.3.2

Imports ks, Matrix, mixtools, stats

Depends R (>= 3.5)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Alba M. Franco-Pereira [aut, cre, cph],
Christos T. Nakas [aut],
Benjamin Reiser [aut],
M.Carmen Pardo [aut],
Alba Alcaraz-Peñalba [aut, cph]

Repository CRAN

Date/Publication 2026-01-23 19:50:39 UTC

Contents

`EM` *EM algorithm for a univariate Gaussian mixture*

Description

Fits a univariate Gaussian mixture model using the Expectation-Maximization (EM) algorithm. The function is intended as a lightweight fallback implementation (e.g., when **mixtools** is unavailable or fails).

Usage

```
EM(X, K = 2, max_iter = 100, tol = 1e-05)
```

Arguments

`X`	Numeric vector of observations.
`K`	Integer. Number of mixture components.
`max_iter`	Integer. Maximum number of EM iterations.
`tol`	Positive numeric. Convergence tolerance for the absolute change in the log-likelihood.

Details

The algorithm is initialized using the k-means clustering procedure and then alternates between:

1. E-step: computing the expectation of the complete log-likelihood function.
2. M-step: maximizing the expectation of the complete log-likelihood function.

Value

A list with the following components:

mu Numeric vector of estimated component means (length `K`).

sigma Numeric vector of estimated component standard deviations (length `K`).

pi Numeric vector of estimated mixing proportions (length `K`).

num_iteraciones Number of iterations performed.

posterior Matrix of posterior probabilities (responsibilities) with dimension `length(X)` by `K`.

Examples

```
set.seed(1)
x <- c(rnorm(100, -2, 1), rnorm(100, 2, 1))
fit <- EM(x, K = 2)
fit$mu
fit$pi
```

---

mixnorm_data — *Simulated data with normal and mixture of normal distributions*

Description

Contains control and case samples generated from a normal distribution and a two-component normal mixture distribution, respectively.

Usage

```
data(mixnorm_data)
```

Format

A data frame with 100 rows and 2 variables:

controls Simulated data from a N(5,1) normal distribution.

cases Simulated data from a two-component normal mixture distribution: 0.8N(2,1) + 0.2N(3,1).

References

This dataset was artificially generated for the **OVL.CI** package.

Examples

```
data(mixnorm_data)
```

OVL.BCAN *OVL.BCAN*

Description

Parametric approach using a bootstrap-based approach to estimate the variance.

Usage

```
OVL.BCAN(x, y, alpha = 0.05, B = 100, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.BCAN (controls,cases)
```

OVL.BCbias *OVL.BCbias*

Description

Parametric approach using a bootstrap bias-corrected approach.

Usage

```
OVL.BCbias(x, y, alpha = 0.05, B = 100, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.BCAN (controls,cases)
```

OVL.BCPB *OVL.BCPB*

Description

Parametric approach using a bootstrap percentil approach.

Usage

```
OVL.BCPB(x, y, alpha = 0.05, B = 100, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.BCPB (controls,cases)
```

OVL.D — *OVL.D*

Description

Parametric approach using the delta method.

Usage

```
OVL.D(x, y, alpha = 0.05)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.D (controls,cases)
```

OVL.DBC — *OVL.DBC*

Description

Parametric approach using the delta method after the Box-Cox transformation.

Usage

```
OVL.DBC(x, y, alpha = 0.05, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.DBC (controls,cases)
```

OVL.DBCL *OVL.DBCL*

Description

Parametric approach using the delta method after the Box-Cox transformation taking into account the variability of the estimated transformation parameter.

Usage

```
OVL.DBCL(x, y, alpha = 0.05, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.DBCL (controls,cases)
```

OVL.Delta.mix *EM-Delta*

Description

Computes a confidence interval for the OVL between two populations under Gaussian and two-component Gaussian mixture models, or both populations modeled as two-component Gaussian mixtures, using EM-based estimation and the delta method.

Usage

```
OVL.Delta.mix(
  x,
  y,
  alpha = 0.05,
  h = 10^(-5),
  interv = c(0, 20),
  all_mix = FALSE
)
```

Arguments

x	Numeric vector. Data from the first group. When `all_mix = FALSE`, this group is modeled as Gaussian.
y	Numeric vector. Data from the second group, modeled as a two-component Gaussian mixture.
alpha	confidence level.
h	Step size used to compute numerical derivatives.
interv	Numeric vector of length 2. Search interval for intersection points between the corresponding densities.
all_mix	Logical. If `TRUE`, both groups are modeled as two-component Gaussian mixtures. If `FALSE`, only `y` is modeled as a mixture and `x` is Gaussian.

Value

A list containing a confidence interval. Additional elements (e.g., `var_OVL`, parameter estimates, `OVL_hat`) may also be returned.

Examples

```
set.seed(1)
x <- ifelse(runif(100) < 0.5, rnorm(100, mean = 0, sd = 1), rnorm(100, mean = 2, sd = 1))
y <- ifelse(runif(100) < 0.5, rnorm(100, mean = 2.5, sd = 1), rnorm(100, mean = 2, sd = 1))
res <- OVL.Delta.mix(x, y, all_mix = TRUE, interv = c(-10, 10))
res$IC1
res$IC2
```

`OVL.GPQ` *OVL.GPQ*

Description

Parametric approach based on generalized inference.

Usage

```
OVL.GPQ(x, y, alpha = 0.05, K = 2500, h_ini = -1.6, BC = FALSE)
```

Arguments

`x`	Numeric vector. Data from the first group.
`y`	Numeric vector. Data from the second group.
`alpha`	confidence level.
`K`	Number of simulated generalized pivotal quantities.
`h_ini`	initial value in the optimization problem.
`BC`	Logical. Indicates whether a Box–Cox transformation is applied to the data.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.GPQ (controls,cases)
```

`OVL.GPQ.mix` *GPQ-Mix*

Description

Computes a confidence interval for the OVL between two populations under Gaussian and two-component Gaussian mixture models, or both populations modeled as two-component Gaussian mixtures, using generalized inference.

Usage

```
OVL.GPQ.mix(x, y, alpha = 0.05, interv = c(0, 20), k = 1000, all_mix = FALSE)
```

Arguments

x	Numeric vector. Data from the first group. When `all_mix = FALSE`, this group is modeled as Gaussian.
y	Numeric vector. Data from the second group, modeled as a two-component Gaussian mixture.
alpha	confidence level.
interv	Numeric vector of length 2. Search interval for intersection points between the corresponding densities.
k	Number of simulated generalized pivotal quantities.
all_mix	Logical. If `TRUE`, both groups are modeled as two-component Gaussian mixtures. If `FALSE`, only `y` is modeled as a mixture and `x` is Gaussian.

Value

confidence interval.

Examples

```
set.seed(1)
x <- ifelse(runif(100) < 0.5,
            rnorm(100, mean = 0, sd = 1),
            rnorm(100, mean = 2, sd = 1))
y <- ifelse(runif(100) < 0.5,
            rnorm(100, mean = 2.5, sd = 1),
            rnorm(100, mean = 2, sd = 1))
res <- OVL.GPQ.mix(x, y, all_mix = TRUE, interv = c(-10, 10))
res$IC1
res$IC2
```

OVL.K *OVL.K*

Description

Kernel approach estimating the variance via bootstrap.

Usage

```
OVL.K(x, y, alpha = 0.05, B = 100, k = 1, h = 1)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.

k	kernel. When k=1 (default value) the kernel used in the estimation is the Gaussian kernel. Otherwise, the Epanechnikov kernel is used instead.
h	bandwidth. When h=1 (default value) the cross-validation bandwidth is chosen. Otherwise, the bandwidth considered by Schmid and Schmidt (2006) is used instead.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.K (controls,cases)
```

OVL.KPB *OVL.KPB*

Description

Kernel approach using a bootstrap percentile approach.

Usage

```
OVL.KPB(x, y, alpha = 0.05, B = 100, k = 1, h = 1)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
k	kernel. When k=1 (default value) the kernel used in the estimation is the Gaussian kernel. Otherwise, the Epanechnikov kernel is used instead.
h	bandwidth. When h=1 (default value) the cross-validation bandwidth is chosen. Otherwise, the bandwidth considered by Schmid and Schmidt (2006) is used instead.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.KPB (controls,cases)
```

OVL.LogitBCAN *OVL.LogitBCAN*

Description

BCAN procedure carried out in the logit scale and back-transformed.

Usage

```
OVL.LogitBCAN(x, y, alpha = 0.05, B = 100, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.LogitBCAN (controls,cases)
```

OVL.LogitD *OVL.LogitD*

Description

Parametric approach using the delta method after switching to a logit scale and then transforming back.

Usage

```
OVL.LogitD(x, y, alpha = 0.05)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.LogitD (controls,cases)
```

OVL.LogitDBC *OVL.LogitDBC*

Description

Parametric approach using the delta method after the Box-Cox transformation after switching to a logit scale and then transforming back.

Usage

```
OVL.LogitDBC(x, y, alpha = 0.05, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.LogitDBC (controls,cases)
```

OVL.LogitDBCL *OVL.LogitDBCL*

Description

OVL.LogitDBCL

Usage

```
OVL.LogitDBCL(x, y, alpha = 0.05, h_ini = -0.6)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
h_ini	initial value in the optimization problem.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.LogitDBCL (controls,cases)
```

OVL.LogitK *OVL.LogitK*

Description

Kernel approach estimating the variance via bootstrap in the logit scale and back-transformed.

Usage

```
OVL.LogitK(x, y, alpha = 0.05, B = 100, k = 1, h = 1)
```

Arguments

x	Numeric vector. Data from the first group.
y	Numeric vector. Data from the second group.
alpha	confidence level.
B	bootstrap size.
k	kernel. When k=1 (default value) the kernel used in the estimation is the Gaussian kernel. Otherwise, the Epanechnikov kernel is used instead.
h	bandwidth. When h=1 (default value) the cross-validation bandwidth is chosen. Otherwise, the bandwidth considered by Schmid and Schmidt (2006) is used instead.

Value

confidence interval.

Examples

```
controls = rnorm(50,6,1)
cases = rnorm(100,6.5,0.5)
OVL.LogitK (controls,cases)
```

---

test_data *Simulated data with normal distributions*

Description

Contains controls and cases data from normal distributions.

Usage

```
data(test_data)
```

Format

A data frame with 100 rows and 2 variables:

controls Simulated data from a N(10,1)distribution for the control group.

cases Simulated data from a N(10.5,0.5)distribution for the case group.

References

This data set was artificially created for the **OVL.CI** package.

Examples

```
data(test_data)
```

Index