

Package: ksamplesLTRC (via r-universe)

September 18, 2026

Title K-Sample Tests for Truncated and/or Censored Data

Version 0.1.1

Description Nonparametric analysis and comparison of distributions under left truncation and right censoring. The package includes simulation routines for truncated and/or censored survival data, nonparametric distribution comparison methods based on Kolmogorov-Smirnov-type and Cramér-von Mises-type statistics, and bootstrap routines for p-value approximation. For methodological details, see Lago, de Uña-Álvarez and Pardo-Fernández (2025) <[doi:10.1007/s11749-024-00948-4](https://doi.org/10.1007/s11749-024-00948-4)> and Lago, Pardo-Fernández and de Uña-Álvarez (2026) <[doi:10.1007/s10985-026-09713-1](https://doi.org/10.1007/s10985-026-09713-1)>.

License GPL-3

Encoding UTF-8

Suggests knitr, rmarkdown, KMsurv, survival, mvna, twosamples, testthat (>= 3.0.0)

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

URL <https://github.com/adrian-lago/ksamplesLTRC>

BugReports <https://github.com/adrian-lago/ksamplesLTRC/issues>

Repository <https://adrian-lago.r-universe.dev>

Date/Publication 2026-09-18 07:55:21 UTC

RemoteUrl <https://github.com/adrian-lago/ksamplesltrc>

RemoteRef HEAD

RemoteSha 69cf619a83ddb2ec1074960857a5da3ae5662678

Contents

ksample.test	2
lines.estimators	4
lines.t.estimators	5
lt.cvm	5
lt.estimator	6
lt.ks	8
lt.logrank	10
lt.pv.cvm	11
lt.pv.ks	13
lt.sim	15
lt.test	16
lt.truncation.estimator	18
ltrc.cvm	20
ltrc.estimator	21
ltrc.ks	23
ltrc.logrank	24
ltrc.pv.cvm	26
ltrc.pv.ks	28
ltrc.sim	30
ltrc.test	31
ltrc.truncation.estimator	33
plot.estimators	34
plot.ksamples_test	35
plot.t.estimators	36
rc.cvm	36
rc.estimator	37
rc.ks	39
rc.logrank	40
rc.pv.cvm	42
rc.pv.ks	43
rc.sim	45
rc.test	46
summary.ksamples_test	48

Index	49
--------------	-----------

ksample.test	<i>Automatic test selector for survival data</i>
--------------	--

Description

Performs tests for data subject to left truncation, right censoring or both.

Usage

```
ksample.test(
  time,
  entry = NULL,
  status = NULL,
  group,
  weights = NULL,
  tests = c("ks", "cvm", "logrank"),
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  p = 0,
  q = 0,
  seed = NULL,
  keep.boot = FALSE
)
```

Arguments

time	Numeric vector of event or follow-up times.
entry	Optional entry time (for left truncation).
status	Event indicator (1 = event, 0 = censored).
group	Group indicator (factor or vector).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
tests	Character: "ks", "cvm", "logrank".
B	Number of bootstrap replications for approximating the p -value.
plot.curves	Logical, default to FALSE. If TRUE, the function depicts the functions to be compared.
holes	One of 'holes' (default), 'add', or 'conditional'. See ltrc.estimator() for details.
p, q	Weights for the log-rank-type tests to be performed. If NULL, the classical log-rank. test is performed.
seed	Seed for the bootstrap statistics.
keep.boot	Logical, default to FALSE. If TRUE, the statistics computed from the bootstrap resamples are stored.

Value

A list with the following components:

- **statistic**: numerical value of the Kolmogorov–Smirnov and/or Cramér–von Mises statistics.
- **p.value**: approximated p -value of the tests.

- `stat.boot`: bootstrap statistics for the p -value approximation.
- `logrank`: a matrix with weights p and q , test statistics, and p -values.
- `B`: number of bootstrap replications.
- `tests`: tests performed, as specified in the input.
- `curves`: distribution function estimators of the k samples to be compared.
- `groups`: group indicators.
- `holes`: method for handling holes (if appropriate).
- `stat.boot`: bootstrap statistics (if `keep.boot = TRUE`).

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")

  aa <- ksample.test(time = channing$age, entry = channing$ageentry,
    status = channing$death, group = channing$gender,
    weights=rep(1/2,2), B = 200, holes='conditional',
    tests=c('ks','cvm','logrank'),plot.curves=TRUE,
    p=c(0,1,0.5,0),q=c(0,0,0.5,1),keep.boot = FALSE)

  summary(aa)
}
```

lines.estimators	<i>Add estimators to an existing plot</i>
------------------	---

Description

Adds the survival estimator to an existing plot.

Usage

```
## S3 method for class 'estimators'
lines(x, conf.int = FALSE, ...)
```

Arguments

<code>x</code>	An object of class "estimators".
<code>conf.int</code>	Logical, default to FALSE. If TRUE, pointwise confidence bands are depicted for the curves.
<code>...</code>	Additional graphical parameters passed to <code>graphics::lines()</code> .

Value

Invisibly returns `x`.

lines.t.estimators	<i>Add estimators to an existing plot</i>
--------------------	---

Description

Adds the survival estimator to an existing plot.

Usage

```
## S3 method for class 't.estimators'
lines(x, ...)
```

Arguments

x	An object of class "estimators".
...	Additional graphical parameters passed to <code>graphics::lines()</code> .

Value

Invisibly returns x.

lt.cvm	<i>Cramér–von Mises-type statistic for the k-sample problem under left truncation</i>
--------	--

Description

Computes the Cramér–von Mises statistic adapted to left truncation.

Usage

```
lt.cvm(
  p.sample,
  weights = NULL,
  holes = c("holes", "add", "conditional"),
  plot.curves = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (U, X, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
holes	Way to handle holes, if present. See <code>lt.estimator()</code> for more details.
plot.curves	Logical, default to FALSE. If true, the k distributions to be compared are depicted.

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation.

Value

Numerical value of the test statistic computed from the k samples.

References

Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  head(pooled.sample)
  aa <- lt.cvm(p.sample=pooled.sample, weights=rep(1/2,2), plot.curves=TRUE)
  aa
}
```

lt.estimator

Survival function estimator under left truncation

Description

Computes the survival function estimator accounting for left truncation. It also returns pointwise confidence bands and includes the possibility of dealing with holes in the sample.

Usage

```
lt.estimator(
  o.sample,
  conf.int = FALSE,
  alpha = 0.05,
  conf.type = c("plain", "loglog"),
  holes = c("holes", "add", "conditional")
)
```

Arguments

<code>o.sample</code>	Matrix or data frame with columns (U, X), such that $U \leq X$. If this input contains more columns, they will not be used.
<code>conf.int</code>	Logical, set as default to FALSE. If TRUE, the function returns a pointwise confidence band.
<code>alpha</code>	Pointwise confidence bands with $1 - \alpha$ confidence level
<code>conf.type</code>	One of 'plain' (default) or 'loglog'. Type of pointwise confidence band.
<code>holes</code>	One of 'add', 'conditional', or 'holes' (default) (see details).

Details

The estimator of the survival function when data are subject to left truncation is defined as (Lynden-Bell, 1971)

$$S_n(t) = \prod_{x_i \leq t} \left(1 - \frac{d_i}{r_i}\right),$$

where $x_1, \dots, x_m$ are the distinct observed event times, $d_i = \#\{j : X_j = x_i\}$ and $r_i = \#\{j : U_j \leq x_i \leq X_j\}$. One should note that this definition of the risk sets is not equal to that in `survival::survfit`, where $r_i = \#\{j : U_j < x_i \leq X_j\}$, so these two implementations may not be equal when ties are present in the data.

The variance for the confidence bands follows a Greenwood's-like formula:

$$Var(S_n(t)) = S_n(t)^2 \sum_{x_i \leq t} \frac{d_i}{n_i(n_i - d_i)}.$$

A log-log transformation is also possible. The confidence band is based on the Gaussianity of the process $\sqrt{n}(S_n(t) - S(t))$.

The function also includes a way to deal with possible holes. A hole is an event time with associated unitary risk set, namely $r_i = 1$ for some $i \in \{1, \dots, m - 1\}$. We exclude the largest event time, since it is always a hole. If a hole is present in a sample, the estimator degenerates and takes the value 1 from that time on. Different techniques are available in the literature for the treatment of holes. For instance, in Stute and Wang (2008), authors suggest to increase the risk set by 1 (holes='add'). Other references, like Klein and Moeschberger (2003), suggest conditioning on the largest hole different from the largest event time (holes='conditional'). The possibility of computing the degenerate statistic is also included (holes='holes').

Value

An object with the following content:

- `fail.time`: ordered distinct observed event times
- `estimation`: estimated survival function at each observed event time
- `prob`: jumps of the estimator
- If `conf.int = TRUE`, it also returns the confidence band lower and upper limits for both the distribution and survival function.

References

Klein, J. P. and Moeschberger, M. L. (2003). *Survival Analysis: Techniques for Censored and Truncated Data*. Springer.

Lynden-Bell, D. (1971). A method of allowing for known observational selection in small samples applied to 3CR quasars. *Monthly Notices of the Royal Astronomical Society*, 155:95-118.

Stute, W. and Wang, J. L. (2008). The central limit theorem under random truncation. *Bernoulli*, 14:604–622.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  plot( lt.estimator(pooled.sample[pooled.sample$group==0,],conf.int=TRUE,conf.type='loglog'),
        xlab='Time (in weeks)',ylab='Estimated survival probability' )
  lines( lt.estimator(pooled.sample[pooled.sample$group==1,],conf.int=TRUE,conf.type='loglog'),col=2 )
}
```

lt.ks

Kolmogorov–Smirnov-type statistic for the k-sample problem under left truncation

Description

Computes the Kolmogorov–Smirnov statistic adapted to left truncation.

Usage

```
lt.ks(
  p.sample,
  weights = NULL,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional")
)
```

Arguments

p.sample	Matrix or data frame with columns (U, X, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
plot.curves	Logical, default to FALSE. If true, the k distributions to be compared are depicted.
holes	What to do when a hole is present in a sample. See <code>lt.estimator()</code> for more details.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j (F_{jn_j}(t) - F_n(t))^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation. The two-sample version of this statistic was studied in Lago et al. (2025). Note that, when $k = 2$, one has

$$D_{KS} = (n_1 p_2^2 + n_2 p_1^2) \left(\sup_t |F_{1n_1}(t) - F_{2n_2}(t)| \right)^2.$$

Value

A list with the following components:

- statistic: numerical value of the Kolmogorov–Smirnov statistic.
- time: time when the maximum is attained.

References

- Kiefer, J. (1959). K -sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.
- Lago, A., de Uña-Álvarez, J., & Pardo-Fernández, J. C. (2025). A Kolmogorov–Smirnov-type test for the two-sample problem with left-truncated data. *Test*, 34(1), 69-90.

Examples

```

if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  head(pooled.sample)
  aa <- lt.ks(p.sample=pooled.sample, weights=rep(1/2,2), plot.curves=TRUE)
  aa
}

```

lt.logrank

*Weighted log-rank tests under left truncation***Description**

Computes weighted log-rank-type tests (Fleming–Harrington family) for k groups with left truncation.

Usage

```

lt.logrank(
  p.sample,
  p = 0,
  q = 0,
  plot.curves = FALSE,
  holes = c("holes", "conditional")
)

```

Arguments

p.sample	Matrix or data frame with columns (U, X, group).
p, q	Nonnegative tuning parameters. If one has length 1 and the other is a vector, the scalar is recycled.
plot.curves	Logical, default to FALSE. If true, the the k distribution functions to be compared are depicted.
holes	How the function deals with holes when estimating the survival function with the pooled sample. One of 'add' (default), 'conditional', and 'holes'. If no holes are present in the sample, this argument will not be used.

Details

The test statistic is based on the quantity

$$L_j = \sum_{i=1}^m \omega_i \left(d_{ji} - d_i \frac{r_{ji}}{r_i} \right),$$

for $j \in \{1, \dots, k\}$. Denote by $x_1, \dots, x_m$ the distinct observed event times from the k samples. For a time x_i and a group j , d_{ji} denotes the total count of individuals that experience the event, and r_{ji} is the number of individuals at risk. In addition, $d_i = \sum_{j=1}^k d_{ji}$ and $r_i = \sum_{j=1}^k r_{ji}$, which are the total number of events observed and the total number of individuals at risk at time x_i , respectively.

Finally, the weights $\omega_1, \dots, \omega_m$ are taken as

$$w_i = \hat{S}(x_i)^p (1 - \hat{S}(x_i))^q,$$

with $\hat{S}(x_i)$ the pooled survival just before the event time. The choice $p = 0 = q$ results in the classical version of the log-rank test.

The test statistic is given by

$$(L_1, \dots, L_{k-1}) \hat{\Sigma}^{-1} (L_1, \dots, L_{k-1})^\top,$$

where $\hat{\Sigma}$ is the matrix with the variance-covariance estimators. Each term in the matrix includes the correction factor $(n_j - d_j)/(n_j - 1)$ for tied event times. The test statistic follows an asymptotic χ_{k-1}^2 distribution under the null hypothesis of equal target distributions in the k populations.

Value

A matrix with columns statistic, detailing the choices of p and q, in addition to the corresponding p.value.

References

Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  head(pooled.sample)
  lt.logrank(p.sample=pooled.sample,p=c(0,1,0.5,5),q=c(0,0,0.5,1),plot.curves=TRUE)
}
```

lt.pv.cvm

Cramér–von Mises-type test for the k -sample problem under left truncation

Description

Computes the Cramér–von Mises statistic adapted to left truncation, and the corresponding p -value.

Usage

```
lt.pv.cvm(
  p.sample,
  weights = NULL,
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  seed = NULL,
  keep.boot = FALSE
)
```

Arguments

<code>p.sample</code>	Matrix or data frame with columns (U, Y, Δ , group).
<code>weights</code>	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when <code>tests="logrank"</code> .
<code>B</code>	Number of bootstrap replications to approximate the p -value.
<code>plot.curves</code>	Logical, default to FALSE. If TRUE, the k curves to be compared are depicted.
<code>holes</code>	How to handle holes if present in a sample. One of <code>holes</code> (default), <code>conditional</code> or <code>add</code> . See details in <code>ltrc.estimator()</code> .
<code>seed</code>	Seed for the bootstrap resamples.
<code>keep.boot</code>	Logical, default to FALSE. If TRUE, the bootstrap statistics are returned in the output of the function.

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation.

The p -value is approximated by means of the obvious bootstrap accounting for left truncation, as a particular case of the obvious bootstrap proposed for left-truncated and right-censored data in Bilker and Wang (1997). The exact algorithm for the two-sample problem can be consulted in Lago et al. (2025).

Value

A list with the following elements:

- `statistic`: numerical value of the Cramér–von Mises test statistic
- `pvalue`: p -value of the test
- `curves`: A list with the estimators of the k distribution functions to be compared.
- `groups`: Group indicator
- `stat.boot`: If `keep.boot = TRUE`, the function returns the statistic evaluated in the bootstrap resamples.

References

- Bilker, W.B., and Wang, M.C. (1997). Bootstrapping left truncated and right censored data. *Communications in Statistics - Simulation and Computation*, 26:141-171.
- Kiefer, J. (1959). K -sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.
- Lago, A., de Uña-Álvarez, J., & Pardo-Fernández, J. C. (2025). A Kolmogorov–Smirnov-type test for the two-sample problem with left-truncated data. *TEST*, 34, 69-90.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  head(pooled.sample)
  aa <- lt.pv.cvm(p.sample=pooled.sample, weights=rep(1/2,2), plot.curves=TRUE, B = 200)
  aa$pvalue
  plot(aa)
}
```

lt.pv.ks

Kolmogorov–Smirnov-type test for the k -sample problem under left truncation

Description

Computes the Kolmogorov–Smirnov test adapted to left truncation and the corresponding p -value.

Usage

```
lt.pv.ks(
  p.sample,
  weights = NULL,
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  seed = NULL,
  keep.boot = FALSE
)
```

Arguments

<code>p.sample</code>	Matrix or data frame with columns (U, X, group).
<code>weights</code>	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when <code>tests="logrank"</code> .
<code>B</code>	Number of bootstrap replications to approximate the p -value.
<code>plot.curves</code>	Logical, default to FALSE. If TRUE, the k curves to be compared are depicted.
<code>holes</code>	How to handle holes if present in a sample. One of <code>holes</code> (default), <code>conditional</code> or <code>add</code> . See details in ltrc.estimator() .
<code>seed</code>	Seed for the bootstrap resamples.
<code>keep.boot</code>	Logical, default to FALSE. If TRUE, the test statistics evaluated in the bootstrap resamples are included in the output.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j (F_{jn_j}(t) - F_n(t))^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation. The two-sample version of this test was studied in Lago et al. (2025).

The p -value is approximated by means of the obvious bootstrap accounting for left truncation, studied in Bilker and Wang (1997). The exact algorithm for the two-sample problem can be consulted in Lago et al. (2025).

Value

A list with the following components:

- `statistic`: numerical value of the Kolmogorov–Smirnov statistic.
- `pvalue`: approximated p-value of the test.
- `curves`: A list with the estimators of the k distribution functions to be compared.
- `groups`: Group indicator
- `stat.boot`: statistics computed from the bootstrap resamples. Included in the output only if `keep.boot = TRUE`

References

Bilker, W.B., and Wang, M.C. (1997). Bootstrapping left truncated and right censored data. *Communications in Statistics - Simulation and Computation*, 26:141-171.

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026). A Kolmogorov–Smirnov-type test for the two-sample problem with left-truncated data. *TEST*: 34:69-90.

Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  head(pooled.sample)
  aa <- lt.pv.ks(p.sample=pooled.sample, weights=rep(1/2,2), plot.curves=TRUE, B = 200)
  aa$pvalue
  plot(aa)
}
```

lt.sim

Left-truncated data simulation

Description

Simulates data under left truncation from prespecified variables for the truncation and target variables.

Usage

```
lt.sim(
  n,
  rtrunc,
  trunc.args = list(),
  rtarget,
```

```

    target.args = list(),
    seed = NULL
  )

```

Arguments

<code>n</code>	Sample size
<code>rtrunc</code>	A function that simulates data from a distribution
<code>trunc.args</code>	A list with the parameters of the truncation variable
<code>rtarget</code>	A function that simulates data from a distribution
<code>target.args</code>	A list with the parameters of the target distribution
<code>seed</code>	Optional seed.

Value

A data frame with n rows and two columns: the truncation times are in the first column, and the event times, in the second one.

Examples

```

aa <- lt.sim(1000,rnorm,trunc.args=list(4,1),rnorm,target.args = list(4,1))
head(aa)
plot(lt.estimator(aa),col=2,xlab='t',ylab='Estimated survival probiblity')
curve(1-pnorm(x,4,1),add=TRUE)

```

lt.test

Tests for the k -sample problem under left truncation

Description

Computes k -sample tests adapted to left truncation, and the corresponding p -value.

Usage

```

lt.test(
  p.sample,
  weights = NULL,
  tests = c("ks", "cvm", "logrank"),
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  p = 0,
  q = 0,
  seed = NULL,
  keep.boot = FALSE
)

```

Arguments

<code>p.sample</code>	Matrix or data frame with columns (U, X, group).
<code>weights</code>	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when <code>tests="logrank"</code> .
<code>tests</code>	Tests to perform. One of Kolmogorov–Smirnov (ks), Cramér–von Mises (cvm), or weighted log-rank-type tests (logrank).
<code>B</code>	Number of bootstrap replications to approximate the p -values of the Kolmogorov–Smirnov and/or Cramér–von Mises tests.
<code>plot.curves</code>	Logical, FALSE by default. If set to TRUE, the k survival functions to be compared are depicted.
<code>holes</code>	Treatment of possible holes. See details in ltrc.estimator() .
<code>p, q</code>	Weights of the log-rank-type tests. The choice $p = 0 = q$ yields the classical log-rank test.
<code>seed</code>	Seed for the bootstrap resamples.
<code>keep.boot</code>	Logical. If TRUE, bootstrap replicates are included in the returned object as <code>stat.boot</code> . Default is FALSE.

Details

This function collects the three tests implemented in this package: Kolmogorov–Smirnov, Cramér–von Mises and weighted log-rank-type tests, (Klein and Moeschberger, 2003) adapted to left truncation. The p -values of the first two are approximated by bootstrap, and the p -value of the log-rank tests is computed by the χ^2_{k-1} asymptotic distribution of the test statistic, where k is the number of groups.

Value

A list with the following components:

- `statistic`: numerical value of the Kolmogorov–Smirnov and/or Cramér–von Mises statistics.
- `p.value`: approximated p -value of the tests.
- `stat.boot`: bootstrap statistics for the p -value approximation.
- `logrank`: a matrix with weights `p` and `q`, test statistics, and p -values.
- `B`: number of bootstrap replications.
- `tests`: tests performed, as specified in the input.
- `curves`: distribution function estimators of the k samples to be compared.
- `groups`: group indicators.
- `holes`: method for handling holes.

The summary method returns a data frame with one row per test and the following columns:

Test Name of the test. For the weighted log-rank-type tests, the weights `p` and `q` are detailed.

Statistic Observed value of the test statistic.

p.value Corresponding p -value.

References

- Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.
- Lago, A., de Uña-Álvarez, J., and Pardo-Fernández, J.C. (2025). A Kolmogorov–Smirnov-type test for the two-sample problem with left-truncated data. TEST: 34:69-90.

Examples

```
if (requireNamespace("mvna", quietly = TRUE)) {
  data("abortion", package = "mvna")
  head(abortion)
  pooled.sample <- abortion[abortion$cause==3, 2:4]
  aa <- lt.test(pooled.sample, weights=rep(1/2,2), B=200,
    holes='conditional', tests=c('ks', 'cvm', 'logrank'),
    p=c(0,1,0.5,0), q=c(0,0,0.5,1))
  summary(aa)
  plot(aa)
}
```

```
lt.truncation.estimator
```

Estimator of the truncation survival function for data subject to left truncation

Description

It computes the estimator of the truncation survival function for data subject to left truncation

Usage

```
lt.truncation.estimator(o.sample, holes = c("holes", "add", "conditional"))
```

Arguments

<code>o.sample</code>	Matrix or data frame with columns (U, X).
<code>holes</code>	Way to handle holes, if present. One of holes (default), add or conditional. See details in ltrc.estimator() .

Details

This estimator is based on the idea that if X is left-truncated by U , then $-U$ is left-truncated by $-X$. In addition, note that

$$S_U(u) = 1 - S_{-U}(-u),$$

where S_U is the survival function of U and S_{-U} is the survival function of $-U$. Thus, since [lt.estimator\(\)](#) computes the estimator of the survival function, one can apply the Lynden-Bell estimator to the sample formed by reversing the roles of U and X by merely multiplying by -1.

The nontruncation probability is defined as

$$\gamma = \mathbb{P}(U \leq X) = \int G(z) dF(z),$$

where $U \sim G$ and $X \sim F$. Then, an estimator of γ is given by (Woodroffe, 1985)

$$\gamma_n = \sum_{i=1}^m \varphi_i G_n(x_i),$$

where $x_1, \dots, x_m$ are the distinct observed event times, $\varphi_1, \dots, \varphi_m$ are the weights assigned by the Lynden-Bell estimator to the observed event times, and $G_n(x_1), \dots, G_n(x_m)$ are estimated distribution function of the observed variable evaluated at the distinct observed event times.

Value

A list with the following:

- `trunc.time`: observed truncation times.
- `estimation`: estimated survival probability.
- `prob`: jumps of the estimator at each observed truncation time.
- `gamma`: estimator of the nontruncation probability.

References

Lynden-Bell, D. (1971). A method of allowing for known observational selection in small samples applied to 3CR quasars. *Monthly Notices of the Royal Astronomical Society*, 155:95-118.

Woodroffe, M. (1985). Estimating a distribution function with truncated data. *The Annals of Statistics*, 13: 163-177.

Examples

```
aa <- lt.sim(500,rnorm,trunc.args=list(4,1),rnorm,target.args = list(4,1))
g.est <- lt.truncation.estimator(o.sample=aa)
plot(g.est$trunc.time,g.est$estimation,type='s',lwd=2,col=2)
curve(1-pnorm(x,4,1),add=TRUE,lwd=2,
xlab='t',ylab='Estimated distribution function')
# nontruncation probability estimator:
g.est$gamma
```

ltrc.cvm	<i>Cramér–von Mises-type statistic for the k-sample problem under left truncation and right censoring</i>
----------	--

Description

Computes the Cramér–von Mises statistic adapted to left truncation and right censoring.

Usage

```
ltrc.cvm(
  p.sample,
  weights = NULL,
  holes = c("holes", "add", "conditional"),
  plot.curves = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Delta, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
holes	How to handle holes if present in a sample. One of holes (default), conditional or add. See ltrc.estimator() for details.
plot.curves	Logical, default to FALSE. If TRUE, the k curves to be compared are depicted.

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation and right censoring, as studied in Lago et al. (2026).

Value

Numerical value of the test statistic computed from the k samples.

References

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026). Kolmogorov–Smirnov and Cramér–von Mises tests for the k-sample problem for left-truncated and right-censored data. *Lifetime Data Analysis*: 32(2).

Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")
  data.channing <- data.frame(channing$ageentry, channing$age,
    channing$death, channing$gender)
  ltrc.cvm(data.channing, weights=rep(1/2,2), holes='conditional', plot.curves = TRUE)
}
```

ltrc.estimator

Survival function estimator under left truncation and right censoring

Description

Computes the survival function estimator accounting for late entry times and censored observations. It also returns pointwise confidence bands.

Usage

```
ltrc.estimator(
  o.sample,
  conf.int = FALSE,
  alpha = 0.05,
  conf.type = c("plain", "loglog"),
  holes = c("holes", "add", "conditional")
)
```

Arguments

<code>o.sample</code>	Matrix or data frame with columns (U, Y, Delta, group).
<code>conf.int</code>	Logical, set as default to FALSE. If set to TRUE, the function returns a pointwise confidence band.
<code>alpha</code>	Pointwise confidence bands with $1 - \alpha$ confidence level
<code>conf.type</code>	Type of pointwise confidence band. Set as default to <i>plain</i> .
<code>holes</code>	One of 'add', 'conditional', or 'holes' (default) (see details).

Details

The estimator is defined as

$$S_n(t) = \prod_{y_i \leq t} \left(1 - \frac{d_i}{n_i}\right),$$

where $y_1, \dots, y_m$ are the distinct observed event times, $d_i = \#\{j : Y_j = y_i, \Delta_j = 1\}$ and $n_i = \#\{j : U_j \leq y_i \leq Y_j\}$. It reduces to the Kaplan–Meier estimator if data are not subject to truncation, and to the Lynden-Bell estimator if no censoring is present.

The variance for the confidence bands follows a Greenwood’s like formula:

$$Var(S_n(t)) = S_n(t)^2 \sum_{y_i \leq t} \frac{d_i}{n_i(n_i - d_i)}.$$

The confidence intervals are based on the asymptotic Gaussianity of the process $\sqrt{n}(S_n(t) - S(t))$. Different transformations are also possible.

The function also includes a way to deal with possible holes. A hole is an event time with associated unitary risk set. If a hole is present in a sample, the estimator degenerates and takes the value 1 from that time on. Different techniques are available in the literature for the treatment of holes. For instance, in Stute and Wang (2008), authors suggest to increase the risk set by 1 (holes=’add’). Other references, like Klein and Moeschberger (2003), suggest conditioning on the largest hole different from the largest event time (holes=’conditional’). The possibility of computing the degenerate statistic is also included (holes=’holes’).

Value

An object with the following content:

fail.time: ordered distinct observed event times

estimation: estimated survival function at each observed event time

prob: jumps of the estimator

If conf.int is set to TRUE, it also returns the confidence band lower and upper limits

References

Klein, J. P. and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer

Wang, M.C. (1991). Nonparametric estimation from cross-sectional survival data. Journal of the American Statistical Association, 86:130-143.

Stute, W. and Wang, J. L. (2008). The central limit theorem under random truncation. Bernoulli, 14:604–622.

Examples

```
observed.sample <- ltrc.sim(n=1000,rtrunc=rnorm,rtarget=rnorm,rcens=rexp,
trunc.args=list(4,1),target.args = list(4,1),cens.args = list(1))
est <- ltrc.estimator(observed.sample,conf.int=TRUE,alpha=0.05,conf.type='loglog')
str(est)
plot(est,col='firebrick',xlab='t', ylab='Estimated survival probability')
```

```

if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")
  o.sample <- channing[channing$gender==1,c(3,4,2)] # subgroup of men
  head(o.sample)
  plot( ltrc.estimator(o.sample,holes='holes'),col=2,xlab='t',
        ylab='Estimated survival probability' ,
        main = 'Estimated survival probability for men') # no correction
  lines( ltrc.estimator(o.sample,holes='add'),col=3 ) # +1 to risk set
  lines( ltrc.estimator(o.sample,holes='conditional'),col=4 )
  legend('topright',legend=c('holes="holes" ', 'holes="add" ',
                             'holes="conditional" '),col=2:4,lwd=rep(1,3))
}

```

ltrc.ks	<i>Kolmogorov–Smirnov-type statistic for the k-sample problem under left truncation and right censoring</i>
---------	--

Description

Computes the Kolmogorov–Smirnov statistic adapted to left truncation and right censoring.

Usage

```

ltrc.ks(
  p.sample,
  weights = NULL,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional")
)

```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Delta, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
plot.curves	Logical, default to FALSE. If TRUE, the k curves to be compared are depicted.
holes	How to handle holes if present in a sample. One of holes (default), conditional or add. See ltrc.estimator() for details.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j \left(F_{jn_j}(t) - F_n(t) \right)^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation and right censoring, as studied in Lago et al. (2026).

Value

A list with the following components:

- **statistic**: numerical value of the Kolmogorov–Smirnov statistic.
- **time**: time when the maximum is attained.

References

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026) Kolmogorov–Smirnov and Cramér–von Mises tests for the k-sample problem for left-truncated and right-censored data. *Lifetime Data Analysis*: 32(2).

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")

  data.channing <- data.frame(channing$ageentry, channing$age,
    channing$death, channing$gender)
  ltrc.ks(data.channing, weights=rep(1/2, 2), holes='conditional')
}
```

ltrc.logrank

Weighted log-rank tests under left truncation and right censoring

Description

Computes weighted log-rank-type tests (Fleming–Harrington family) for k groups with left truncation and right censoring.

Usage

```
ltrc.logrank(
  p.sample,
  p = 0,
  q = 0,
  plot.curves = FALSE,
  holes = c("holes", "conditional")
)
```

Arguments

<code>p.sample</code>	Matrix or data frame with columns (U, Y, Delta, group).
<code>p, q</code>	Nonnegative weight parameters. If one has length 1 and the other is a vector, the scalar is recycled.
<code>plot.curves</code>	Logical, default to FALSE. If true, the the k distribution functions to be compared are depicted.
<code>holes</code>	One of 'holes' (default) or 'conditional'. In contrast to other functions of this package the possibility of adding 1 to the risk set is not considered.

Details

The test statistic is based on the quantity

$$L_j = \sum_{i=1}^m \omega_i \left(d_{ji} - d_i \frac{r_{ji}}{r_i} \right),$$

for $j \in \{1, \dots, k\}$. Denote by $y_1, \dots, y_m$ the distinct observed event times from the k samples. For a time y_i and a group j , d_{ji} denotes the total count of individuals that experience the event, and r_{ji} is the number of individuals at risk. In addition, $d_i = \sum_{j=1}^k d_{ji}$ and $r_i = \sum_{j=1}^k r_{ji}$, which are the total number of events observed and the total number of individuals at risk at time y_i , respectively.

Finally, the weights $\omega_1, \dots, \omega_m$ are taken as

$$w_i = \hat{S}(y_i)^p (1 - \hat{S}(y_i))^q,$$

with $\hat{S}(x_i)$ the pooled survival just before the event time. The choice $p = 0 = q$ results in the classical version of the log-rank test.

The test statistic is given by

$$(L_1, \dots, L_{k-1}) \hat{\Sigma}^{-1} (L_1, \dots, L_{k-1})^\top,$$

where $\hat{\Sigma}$ is the matrix with the variance-covariance estimators. Each term in the matrix includes the correction factor $(n_j - d_j)/(n_j - 1)$ for tied event times. The test statistic follows an χ_{k-1}^2 distribution under the null hypothesis of equal target distributions in the k populations.

Value

A matrix with columns statistic and p.value.

References

Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")
}
```

```
data.channing <- data.frame(channing$ageentry, channing$age,
channing$death, channing$gender)
ltrc.logrank(data.channing, plot.curves=TRUE,
p = c(0,1,0.5,0),q = c(0,0,0.5,1),
holes = 'conditional')
}
```

ltrc.pv.cvm

Cramér–von Mises-type test for the k -sample problem under left truncation and right censoring

Description

Computes the Cramér–von Mises test adapted to left truncation and right censoring and the corresponding p -value.

Usage

```
ltrc.pv.cvm(
  p.sample,
  weights = NULL,
  B = 500,
  holes = c("holes", "add", "conditional"),
  plot.curves = FALSE,
  keep.boot = FALSE
)
```

Arguments

<code>p.sample</code>	Matrix or data frame with columns (U, Y, Δ , group).
<code>weights</code>	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when <code>tests="logrank"</code> .
<code>B</code>	Number of bootstrap replications to approximate the p -value
<code>holes</code>	How to deal with holes when estimating the distribution function of the target variable. One of 'holes' (default), 'add', and 'conditional'. See ltrc.estimator() for details.
<code>plot.curves</code>	Logical, default to FALSE. If TRUE, the k samples to be compared are depicted.
<code>keep.boot</code>	Logical, default to FALSE. If TRUE, the function returns the statistics computed from the bootstrap resamples.

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation and right censoring, as studied in Lago et al. (2026).

The p -value is approximated by means of the obvious bootstrap accounting for left truncation and right censoring, studied in Bilker and Wang (1997). The exact algorithm for the k -sample problem can be consulted in Lago et al. (2026).

Value

A list with the following:

- `statistic`: Evaluation of the Cramér–von Mises statistic on the samples.
- `pvalue`: p -value approximated by bootstrap.
- `curves`: A list with the estimators of the k distribution functions to be compared.
- `groups`: Group indicator
- `stat.boot`: Evaluation of the Cramér–von Mises statistic on the bootstrap resamples. Only included in the output if `keep.boot = TRUE`.

References

Bilker, W.B., and Wang, M.C. (1997). Bootstrapping left truncated and right censored data. *Communications in Statistics - Simulation and Computation*, 26:141-171.

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026). Kolmogorov–Smirnov and Cramér–von Mises tests for the k -sample problem for left-truncated and right-censored data. *Lifetime Data Analysis*: 32(2).

Kiefer, J. (1959). K -sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")

  data.channing <- data.frame( channing$ageentry, channing$age,
    channing$death, channing$gender )
  aa <- ltrc.pv.cvm(data.channing, holes = 'conditional',
    plot.curves = FALSE, B = 200)
```

```
aa$pvalue
}
```

ltrc.pv.ks	<i>Kolmogorov–Smirnov-type test for the k-sample problem under left truncation and right censoring</i>
------------	---

Description

Computes the Kolmogorov–Smirnov test adapted to left truncation and right censoring and the corresponding p -value.

Usage

```
ltrc.pv.ks(
  p.sample,
  weights = NULL,
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  keep.boot = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Δ , group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
B	Number of bootstrap replications to approximate the p -value.
plot.curves	Logical, default to FALSE. If TRUE, the k curves to be compared are depicted.
holes	How to handle holes if present in a sample. One of holes (default), conditional or add. See ltrc.estimator() for details.
keep.boot	Logical, default to FALSE. If TRUE, the function returns the statistics evaluated in the bootstrap resamples.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j \left(F_{jn_j}(t) - F_n(t) \right)^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for left truncation and right censoring, as studied in Lago et al. (2026).

The p -value is approximated by means of the obvious bootstrap accounting for left truncation and right censoring, studied in Bilker and Wang (1997). The exact algorithm for the k -sample problem can be consulted in Lago et al. (2026).

Value

A list with the following components:

- `statistic`: numerical value of the Kolmogorov–Smirnov statistic.
- `p.value`: approximated p -value of the tests.
- `curves`: A list with the k estimators of the distribution functions to be compared.
- `groups`: Group indicator
- `stat.boot`: the statistics computed from the bootstrap resamples. Included in the output only if `keep.boot = TRUE`.

References

Bilker, W.B., and Wang, M.C. (1997). Bootstrapping left truncated and right censored data. *Communications in Statistics - Simulation and Computation*, 26:141-171.

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026). Kolmogorov–Smirnov and Cramér–von Mises tests for the k -sample problem for left-truncated and right-censored data. *Lifetime Data Analysis*: 32(2).

Kiefer, J. (1959). K -sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")
  data.channing <- data.frame(channing$ageentry, channing$age,
    channing$death, channing$gender)
  aa <- ltrc.pv.ks(data.channing, weights=rep(1/2, 2),
    holes = 'conditional', plot.curves = FALSE, B = 200)
  plot(aa)
  aa$pvalue
}
```

ltrc.sim

*Simulate left-truncated and right-censored data***Description**

Simulates data under left truncation and right censoring.

Usage

```
ltrc.sim(
  n,
  rtrunc,
  trunc.args = list(),
  rtarget,
  target.args = list(),
  rcens,
  cens.args = list(),
  seed = NULL
)
```

Arguments

n	Sample size.
rtrunc	A function generating truncation times.
trunc.args	A list of arguments passed to rtrunc.
rtarget	A function generating target/event times.
target.args	A list of arguments passed to rtarget.
rcens	A function generating residual censoring times.
cens.args	A list of arguments passed to rcens.
seed	Optional seed.

Value

A data frame with columns u, y, and delta.

Examples

```
aa <- ltrc.sim(n=500,rtrunc=rnorm,trunc.args=list(4,1),
  rtarget=rnorm,target.args = list(4,1),
  rcens=rexp,cens.args=list(1),seed=12)
head(aa)
```

ltrc.test	<i>Tests for the k-sample problem under left truncation and right censoring</i>
-----------	--

Description

Computes k -sample tests adapted to left truncation and right censoring and the corresponding p -value.

Usage

```
ltrc.test(
  p.sample,
  weights = NULL,
  tests = c("ks", "cvm", "logrank"),
  B = 500,
  plot.curves = FALSE,
  holes = c("holes", "add", "conditional"),
  p = 0,
  q = 0,
  seed = NULL,
  keep.boot = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Δ , group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
tests	Tests to perform. One of Kolmogorov–Smirnov (ks), Cramér–von Mises (cvm), or weighted log-rank-type tests (logrank).
B	Number of bootstrap replications to approximate the p -values of the Kolmogorov–Smirnov and/or Cramér–von Mises tests.
plot.curves	Logical, FALSE by default. If set to TRUE, the k survival functions to be compared are depicted.
holes	One of holes (default), conditional or add. If holes = "add" and logrank is requested, the log-rank test is computed with holes = "holes" and a warning is issued. See ltrc.estimator() for details.
p, q	Weights of the log-rank-type tests. The choice $p = 0 = q$ yields the classical log-rank test.
seed	Seed for the bootstrap resamples.
keep.boot	Logical. If TRUE, bootstrap replicates are included in the returned object as stat.boot. Default is FALSE.

Details

This function collects the three tests implemented in this package: Kolmogorov–Smirnov, Cramér–von Mises (Lago et al., 2026) and weighted log-rank-type tests, (Klein and Moeschberger, 2003) adapted to left truncation and right censoring. The p -values of the first two are approximated by bootstrap, and the p -value of the log-rank tests is computed by the χ^2_{k-1} asymptotic distribution of the test statistic, where k is the number of groups.

Value

A list with the following components:

- **statistic**: numerical value of the Kolmogorov–Smirnov and/or Cramér–von Mises statistics.
- **p.value**: approximated p -value of the tests.
- **stat.boot**: bootstrap statistics for the p -value approximation.
- **logrank**: a matrix with weights p and q , test statistics, and p -values.
- **B**: number of bootstrap replications.
- **tests**: tests performed, as specified in the input.
- **curves**: distribution function estimators of the k samples to be compared.
- **groups**: group indicators.
- **holes**: method for handling holes.

The summary method returns a data frame with one row per test and the following columns:

Test Name of the test. For the weighted log-rank-type tests, the weights p and q are detailed.

Statistic Observed value of the test statistic.

p.value Corresponding p -value.

References

Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.

Lago, A., Pardo-Fernández, J.C., and de Uña-Álvarez, J. (2026). Kolmogorov–Smirnov and Cramér–von Mises tests for the k -sample problem for left-truncated and right-censored data. Lifetime Data Analysis: 32(2).

Examples

```
if (requireNamespace("KMsurv", quietly = TRUE)) {
  data("channing", package = "KMsurv")

  data.channing <- data.frame(channing$ageentry, channing$age,
    channing$death, channing$gender)
  aa <- ltrc.test(data.channing, weights=rep(1/2,2),
    holes='conditional', tests=c('ks', 'cvm', 'logrank'), B = 200,
    p=c(0,1,0.5,0), q=c(0,0,0.5,1))
  plot(aa)
```

```
summary(aa)
}
```

```
ltrc.truncation.estimator
```

Estimator of the truncation distribution function for data subject to left truncation and right censoring

Description

It computes the estimator of the truncation distribution function for data subject to left truncation and right censoring

Usage

```
ltrc.truncation.estimator(o.sample, holes = c("holes", "add", "conditional"))
```

Arguments

<code>o.sample</code>	Matrix or data frame with columns (U, Y, Delta).
<code>holes</code>	One of 'add', 'conditional' or 'holes' (default). How to deal with holes in the estimation of the target variable survival function. See ltrc.estimator() for details.

Details

The estimator of the distribution function of the truncation variable is given by

$$G_n(t) = \sum_i \frac{S_n(u_i)^{-1}}{\sum_j S_n(u_j)^{-1}} \mathbb{I}[u_i \leq t],$$

where $u_1, \dots, u_m$ are the distinct truncation times and S_n is the survival function estimator of the target variable.

The nontruncation probability is defined as

$$\gamma = \mathbb{P}(U \leq X) = \int G(z) dF(z),$$

where F is the distribution function of the target variable. Then, it can be estimated as

$$\gamma_n = \sum_{i=1}^m \varphi_i G_n(y_i),$$

where $y_1, \dots, y_m$ are the distinct observed event times and $\varphi_1, \dots, \varphi_m$ are the weights that the estimator S_n assigns to $y_1, \dots, y_m$.

In practice, it is possible to observe truncation times larger than the largest event time. This would break the formula of the estimator G_n defined above. In such a case, following the recommendations

in Wang (1991), we reduce the distribution function of the truncation variable to a conditional distribution on the identifiable region, by trimming a small amount of data. This is analogous to conditioning to times largest than the largest hole when estimating the distribution function of the target variable.

Value

An object with the observed truncation times, the estimator at those times, the jumps of the estimator and an estimator of the nontruncation probability

References

Wang, M.C. (1991). Nonparametric Estimation from Cross-Sectional Survival Data. Journal of the American Statistical Association, 86:130-143.

Examples

```
aa <- ltrc.sim(n=5000,rtrunc=rnorm,trunc.args = list(4,1),rtarget=rnorm,
target.args=list(4,1),rcens=rexp,cens.args=list(1))
head(aa)
plot(ltrc.truncation.estimator(aa))
curve(pnorm(x,4,1),add=TRUE,col=2)

if (requireNamespace("KMsurv", quietly = TRUE)) {
data("channing", package = "KMsurv")

o.sample <- channing[channing$gender==1,c(3,4,2)] # subgroup of men
plot(ltrc.estimator(o.sample)) # hole in the second event time
plot(ltrc.truncation.estimator(o.sample,holes='add'))
lines(ltrc.truncation.estimator(o.sample,holes='conditional'),col=2)
}
```

plot.estimators

Plot method for estimators objects

Description

Produces a step plot of the survival estimator.

Usage

```
## S3 method for class 'estimators'
plot(x, conf.int = FALSE, ...)
```

Arguments

x	An object of class "estimators".
conf.int	Logical, default to FALSE. If TRUE, pointwise confidence bands are depicted for the curves.
...	Additional graphical parameters passed to <code>graphics::plot()</code> .

Value

Invisibly returns x.

plot.ksamples_test	<i>Plot method for k-sample test objects</i>
--------------------	--

Description

Plots the estimated survival curves for the groups involved in `ltrc.test()`.

Usage

```
## S3 method for class 'ksamples_test'
plot(
  x,
  y = NULL,
  col = NULL,
  lwd = 2,
  lty = 1,
  xlab = "Time",
  ylab = "Survival probability",
  main = "Estimated survival curves",
  legend.pos = "topright",
  ...
)
```

Arguments

x	An object of class "ksamples_test".
y	Ignored.
col	Vector of colours for the groups.
lwd	Line width.
lty	Line type.
xlab	X-axis label.
ylab	Y-axis label.
main	Plot title.
legend.pos	Position of the legend.
...	Additional graphical parameters passed to <code>graphics::plot()</code> .

Value

Invisibly returns x.

plot.t.estimators	<i>Plot method for estimators objects</i>
-------------------	---

Description

Produces a step plot of the survival estimator.

Usage

```
## S3 method for class 't.estimators'
plot(x, ...)
```

Arguments

x	An object of class "estimators".
...	Additional graphical parameters passed to graphics::plot() .

Value

Invisibly returns x.

rc.cvm	<i>Cramér–von Mises-type statistic for the k-sample problem under right censoring</i>
--------	---

Description

Computes the Cramér–von Mises statistic adapted to left truncation and right censoring.

Usage

```
rc.cvm(p.sample, weights = NULL, plot.curves = FALSE)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Delta, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
plot.curves	Logical, default to FALSE. If TRUE, the <i>k</i> distributions to be compared are depicted

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for right censoring.

Value

Numerical value of the test statistic computed from the k samples.

References

Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Schumacher, M. (1984). Two-Sample Tests of Cramér–von Mises-and Kolmogorov–Smirnov-Type for Randomly Censored Data. *International Statistical Review/Revue Internationale de Statistique*, 263-281.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)

  data <- data.frame(y=stanford2$time,
    delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age>=47))
  rc.cvm(data,weights=table(data$group)/nrow(data))
}
```

rc.estimator

Survival function estimator under left truncation and right censoring

Description

Computes the survival function estimator accounting for censored observations. It also returns pointwise confidence bands.

Usage

```
rc.estimator(
  o.sample,
  conf.int = FALSE,
  alpha = 0.05,
  conf.type = c("plain", "loglog")
)
```

Arguments

<code>o.sample</code>	Matrix or data frame with columns (U, Y, Delta, group).
<code>conf.int</code>	Logical, set as default to FALSE. If set to TRUE, the function returns a pointwise confidence band.
<code>alpha</code>	Pointwise confidence bands with $1 - \alpha$ confidence level
<code>conf.type</code>	Type of pointwise confidence band. Set as default to <i>plain</i> .

Details

The estimator is defined as

$$S_n(t) = \prod_{y_i \leq t} \left(1 - \frac{d_i}{n_i}\right),$$

where $y_1, \dots, y_m$ are the distinct observed event times, $d_i = \#\{j : Y_j = y_i, \Delta_j = 1\}$ and $n_i = \#\{j : U_j \leq y_i \leq Y_j\}$. It reduces to the Kaplan–Meier estimator if data are not subject to truncation, and to the Lynden-Bell estimator if no censoring is present.

The variance for the confidence bands follows a Greenwood’s like formula:

$$\text{Var}(S_n(t)) = S_n(t)^2 \sum_{y_i \leq t} \frac{d_i}{n_i(n_i - d_i)}.$$

Different transformations are also possible.

Value

An object with the following content:

fail.time: ordered distinct observed event times

estimation: estimated survival function at each observed event time

prob: jumps of the estimator

If `conf.int` is set to TRUE, it also returns the confidence band lower and upper limits

References

- Klein, J. P. and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer
- Wang, M.C. (1991). Nonparametric estimation from cross-sectional survival data. Journal of the American Statistical Association, 86:130-143.
- Stute, W. and Wang, J. L. (2008). The central limit theorem under random truncation. Bernoulli, 14:604–622.

Examples

```
x <- rnorm(500,4,1)
c <- rnorm(500,4,1)
o.sample <- data.frame(y = pmin(x,c),delta = as.numeric(x <= c))
plot(rc.estimator(o.sample,conf.int=TRUE,conf.type='loglog'))
curve(1-pnorm(x,4,1),add=TRUE,col=2)
```

rc.ks	<i>Kolmogorov–Smirnov-type statistic for the k-sample problem under right censoring</i>
-------	---

Description

Computes the Kolmogorov–Smirnov statistic adapted to right censoring.

Usage

```
rc.ks(p.sample, weights = NULL, plot.curves = FALSE)
```

Arguments

p.sample	Matrix or data frame with columns (Y, Delta, group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
plot.curves	Logical, FALSE by default. Otherwise, it represents the k distribution functions to be compared.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j (F_{jn_j}(t) - F_n(t))^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for right censoring.

For the particular case of $k = 2$, the test statistic reduces to

$$D_{KS} = (n_1 p_2^2 + n_2 p_1^2) \left(\sup_t |F_{1n_1}(t) - F_{2n_2}(t)| \right)^2,$$

thus the test indeed extends the classical two-sample Kolmogorov–Smirnov test. A two-sample Kolmogorov–Smirnov-type test for right-censored data was proposed in Schumacher (1984).

Value

A list with the following components:

- **statistic**: numerical value of the Kolmogorov–Smirnov statistics.
- **time**: time when the maximum is attained.

References

Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Schumacher, M. (1984). Two-Sample Tests of Cramér–von Mises-and Kolmogorov–Smirnov-Type for Randomly Censored Data. *International Statistical Review/Revue Internationale de Statistique*, 263-281.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)

  data <- data.frame(y=stanford2$time,
    delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age>=47))
  rc.ks(p.sample=data,weights=table(data$group)/nrow(data),plot.curves=TRUE)
}
```

rc.logrank

Weighted log-rank tests under left truncation

Description

Computes weighted log-rank-type tests (Fleming–Harrington family) for k groups with left truncation and right censoring.

Usage

```
rc.logrank(p.sample, p = 0, q = 0, plot.curves = FALSE)
```

Arguments

- | | |
|-------------|---|
| p.sample | Matrix or data frame with columns (U, Y, Delta, group). |
| p, q | Nonnegative weight parameters. If one has length 1 and the other is a vector, the scalar is recycled. |
| plot.curves | Logical, default to FALSE. If TRUE, the k distributions to be compared are depicted |

Details

The test statistic is based on the quantity

$$L_j = \sum_{i=1}^m \omega_i \left(d_{ji} - d_i \frac{r_{ji}}{r_i} \right),$$

for $j \in \{1, \dots, k\}$. Denote by $y_1, \dots, y_m$ the distinct observed event times from the k samples. For a time y_i and a group j , d_{ji} denotes the total count of individuals that experience the event, and r_{ji} is the number of individuals at risk. In addition, $d_i = \sum_{j=1}^k d_{ji}$ and $r_i = \sum_{j=1}^k r_{ji}$, which are the total number of events observed and the total number of individuals at risk at time y_i , respectively.

Finally, the weights $\omega_1, \dots, \omega_m$ are taken as

$$w_i = \hat{S}(y_i)^p (1 - \hat{S}(y_i))^q,$$

with $\hat{S}(x_i)$ the pooled survival just before the event time. The choice $p = 0 = q$ results in the classical version of the log-rank test.

The test statistic is given by

$$(L_1, \dots, L_{k-1}) \hat{\Sigma}^{-1} (L_1, \dots, L_{k-1})^\top,$$

where $\hat{\Sigma}$ is the matrix with the variance-covariance estimators. Each term in the matrix includes the correction factor $(n_j - d_j)/(n_j - 1)$ for tied event times. The test statistic follows an χ_{k-1}^2 distribution under the null hypothesis of equal target distributions in the k populations.

Value

A matrix with columns statistic and p.value.

References

Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)
  data <- data.frame(y=stanford2$time,
    delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age>=47))
  rc.logrank(data, p = c(0, 1, 2), q = 0)
}
```

rc.pv.cvm	<i>Cramér–von Mises-type test for the k-sample problem under right censoring</i>
-----------	---

Description

Computes the Kolmogorov–Smirnov test adapted to left truncation and right censoring and the corresponding p -value.

Usage

```
rc.pv.cvm(p.sample, weights = NULL, B = 500, keep.boot = FALSE)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Δ , group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
B	Number of bootstrap replications to approximate the p -value.
keep.boot	Logical, default to FALSE. If TRUE, the function returns the evaluation of the Cramér–von Mises statistic on the bootstrap resamples.

Details

In Kiefer (1959), the k -sample version of the Cramér–von Mises test is defined as

$$D_{CvM} = \sum_{j=1}^k n_j \int (F_{jn_j}(t) - F_n(t))^2 dF_n(t),$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for right censoring.

The p -value is approximated by means of the obvious bootstrap accounting for left truncation and right censoring, studied in Efron (1981).

Value

A list with the following:

- statistic: Cramér–von Mises statistic evaluated on the the samples.
- pvalue: p -value of the test approximated via bootstrap.

- curves: A list with the estimators of the k distribution functions to be compared.
- groups: Group indicator.
- stat.boot: The Cramér–von Mises statistic evaluated from the bootstrap resamples.

References

- Efron, B. (1981). Censored data and the bootstrap. *Journal of the American Statistical Association*, 76: 312-319.
- Kiefer, J. (1959). K-sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)
  data <- data.frame(y=stanford2$time,
    delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age<=47))
  aa <- rc.pv.cvm(data,weights=table(data$group)/nrow(data), B = 200)
  plot(aa)
  aa$pvalue
}
```

rc.pv.ks	<i>Kolmogorov–Smirnov-type test for the k-sample problem under right censoring</i>
----------	---

Description

Computes the Kolmogorov–Smirnov test adapted to right censoring and the corresponding p -value.

Usage

```
rc.pv.ks(
  p.sample,
  weights = NULL,
  B = 500,
  plot.curves = FALSE,
  keep.boot = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (U, Y, Δ , group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".

B	Number of bootstrap replications to approximate the p -value
plot.curves	logical, set to FALSE by default. If TRUE, the function depicts the k functions to be compared.
keep.boot	Logical, default to FALSE. If TRUE, the statistics evaluated on the bootstrap resamples are included in the output.

Details

In Kiefer (1959), the k -sample version of the Kolmogorov–Smirnov test is defined as

$$D_{KS} = \sup_t \sum_{j=1}^k n_j (F_{jn_j}(t) - F_n(t))^2,$$

where F_{jn_j} is the cumulative distribution function estimator of calculated only with the j -th sample, and F_n is given by

$$F_n(t) = \sum_{j=1}^k p_j F_{jn_j}(t),$$

where $p_1, \dots, p_k$ correspond to the input weights. This function implements the previous test statistic accounting for right censoring.

For the particular case of $k = 2$, the test statistic reduces to

$$D_{KS} = (n_1 p_2^2 + n_2 p_1^2) \left(\sup_t |F_{1n_1}(t) - F_{2n_2}(t)| \right)^2,$$

thus the test indeed extends the classical two-sample Kolmogorov–Smirnov test. A two-sample Kolmogorov–Smirnov-type test for right-censored data was proposed in Schumacher (1984).

The p -value is approximated by means of the obvious bootstrap accounting for right censoring, studied in Efron (1981).

Value

A list with the following components:

- statistic: numerical value of the Kolmogorov–Smirnov statistic.
- pvalue: approximated p -value of the test.
- curves: A list with the estimators of the k distribution functions to be compared.
- groups: Group indicator
- stat.boot: statistics computed from the bootstrap resamples.

References

- Efron, B. (1981). Censored data and the bootstrap. *Journal of the American Statistical Association*, 76: 312-319.
- Kiefer, J. (1959). K -sample analogues of the Kolmogorov–Smirnov and Cramér–von Mises tests. *The Annals of Mathematical Statistics*, 30:420-447.

Examples

```

if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)

  data <- data.frame(y=stanford2$time,
    delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age<=47))
  aa <- rc.pv.ks(data,weights=table(data$group)/nrow(data), B = 200)
  plot(aa)
  aa$pvalue
}

```

rc.sim

*Right-censored data simulation***Description**

Simulates right-censored survival data from prespecified event-time and censoring-time distributions.

Usage

```

rc.sim(
  n,
  rtarget,
  target.args = list(),
  rcens,
  cens.args = list(),
  seed = NULL
)

```

Arguments

<code>n</code>	Sample size.
<code>rtarget</code>	A function that simulates event/survival times.
<code>target.args</code>	A list with the parameters of the target distribution.
<code>rcens</code>	A function that simulates censoring times.
<code>cens.args</code>	A list with the parameters of the censoring distribution.
<code>seed</code>	Optional seed for reproducibility.

Value

A data frame with n rows and two columns:

y Observed follow-up time, i.e. $\min(X, C)$.

delta Event indicator, equal to 1 if the event is observed and 0 if censored.

Examples

```
aa <- rc.sim(n = 1000, rtarget = rweibull,
target.args = list(shape = 2, scale = 3),
rcens = rexp, cens.args = list(rate = 0.2),
seed = 123)
head(aa)
mean(aa$delta == 0) # empirical censoring rate
```

rc.test

Tests for the k -sample problem under right censoring

Description

Computes k -sample tests adapted to right censoring and the corresponding p -value.

Usage

```
rc.test(
  p.sample,
  weights = NULL,
  tests = c("ks", "cvm", "logrank"),
  B = 500,
  plot.curves = FALSE,
  p = 0,
  q = 0,
  seed = NULL,
  keep.boot = FALSE
)
```

Arguments

p.sample	Matrix or data frame with columns (Y, Δ , group).
weights	Vector of weights of the estimator under the null hypothesis. If no vector is provided, weights proportional to each sample size are assigned automatically by the function. Not used when tests="logrank".
tests	Tests to perform. One or more of Kolmogorov–Smirnov ('ks'), Cramér–von Mises ('cvm'), or weighted log-rank-type tests ('logrank').
B	Number of bootstrap replications to approximate the p -values of the Kolmogorov–Smirnov and/or Cramér–von Mises tests.
plot.curves	Logical, FALSE by default. If set to TRUE, the k survival functions to be compared are depicted.
p, q	Weights of the log-rank-type tests. The choice $p = 0 = q$ yields the classical log-rank test.
seed	Seed for the bootstrap resamples.
keep.boot	Logical. If TRUE, bootstrap replicates are included in the returned object as stat.boot. Default is FALSE.

Details

This function collects the three tests implemented in this package: Kolmogorov–Smirnov, Cramér–von Mises and weighted log-rank-type tests, (Klein and Moeschberger, 2003) adapted to right censoring. The p -values of the first two are approximated by the obvious bootstrap (Efron, 1981), and the p -value of the log-rank tests is computed by the χ^2_{k-1} asymptotic distribution of the test statistic, where k is the number of groups.

Value

A list with the following components:

- **statistic**: numerical value of the Kolmogorov–Smirnov and/or Cramér–von Mises statistics.
- **p.value**: approximated p -value of the tests.
- **logrank**: a matrix with weights p and q , test statistics, and p -values.
- **B**: number of bootstrap replications.
- **tests**: tests performed, as specified in the input.
- **curves**: distribution function estimators of the k samples to be compared.
- **groups**: group indicators.
- **stat.boot**: evaluation of the tests in the bootstrap resamples. Only when `keep.boot = TRUE`.

The summary method returns a data frame with one row per test and the following columns:

Test Name of the test. For the weighted log-rank-type tests, the weights p and q are detailed.

Statistic Observed value of the test statistic.

p.value Corresponding p -value.

References

- Klein, J.P., and Moeschberger, M. L. (2003). Survival Analysis: Techniques for Censored and Truncated Data. Springer, New York.
- Efron, B. (1981). Censored data and the bootstrap. Journal of the American Statistical Association, 76: 312-319.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  library(survival)

  data <- data.frame(y=stanford2$time,delta=stanford2$status,
    group=as.numeric(stanford2$age<=35)+as.numeric(stanford2$age>=47))
  aa <- rc.test(p.sample=data,weights=table(data$group)/nrow(data),
    plot.curves=TRUE,tests=c('ks','cvm','logrank'),
    B = 200, p=c(0,1,2), q=0 )

  plot(aa)
  summary(aa)
}
```

summary.ksamples_test *Summary method for k-sample test objects*

Description

Produces a table with the test name, test statistic and p-value for the procedures computed by `ltrc.test()`.

Usage

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

Arguments

<code>object</code>	An object of class "ksamples_test".
<code>...</code>	Additional arguments passed to methods.

Value

A data frame with columns:

Test Name of the test ("KS", "CvM", or the corresponding (p, q) values for weighted rank tests).

Statistic Observed test statistic.

p-value Associated p-value.

Index

graphics::lines(), [4](#), [5](#)
graphics::plot(), [35](#), [36](#)

ksample.test, [2](#)

lines.estimators, [4](#)
lines.t.estimators, [5](#)
lt.cvm, [5](#)
lt.estimator, [6](#)
lt.estimator(), [5](#), [9](#), [18](#)
lt.ks, [8](#)
lt.logrank, [10](#)
lt.pv.cvm, [11](#)
lt.pv.ks, [13](#)
lt.sim, [15](#)
lt.test, [16](#)
lt.truncation.estimator, [18](#)
ltrc.cvm, [20](#)
ltrc.estimator, [21](#)
ltrc.estimator(), [3](#), [12](#), [14](#), [17](#), [18](#), [20](#), [23](#),
 [26](#), [28](#), [31](#), [33](#)
ltrc.ks, [23](#)
ltrc.logrank, [24](#)
ltrc.pv.cvm, [26](#)
ltrc.pv.ks, [28](#)
ltrc.sim, [30](#)
ltrc.test, [31](#)
ltrc.test(), [35](#), [48](#)
ltrc.truncation.estimator, [33](#)

plot.estimators, [34](#)
plot.ksamples_test, [35](#)
plot.t.estimators, [36](#)

rc.cvm, [36](#)
rc.estimator, [37](#)
rc.ks, [39](#)
rc.logrank, [40](#)
rc.pv.cvm, [42](#)
rc.pv.ks, [43](#)

rc.sim, [45](#)
rc.test, [46](#)

summary.ksamples_test, [48](#)