Package ‘GofCens’

May 7, 2026

Type Package

Title Goodness-of-Fit Methods for Right-Censored Data

Version 1.5

Date 2025-05-29

Description Graphical tools and goodness-of-fit tests for right-censored data:

1. Kolmogorov-Smirnov, Cramér-von Mises, and Anderson-Darling tests, which use the empirical distribution function for complete data and are extended for right-censored data.
2. Generalized chi-squared-type test, which is based on the squared differences between observed and expected counts using random cells with right-censored data.
3. A series of graphical tools such as probability or cumulative hazard plots to guide the decision about the most suitable parametric model for the data.

   These functions share several features as they can handle both complete and right-censored data, and they provide parameter estimates for the distributions under study.

License GPL (>= 2)

URL `https://arnaugarciagrbio.github.io/GofCens/`,
`https://github.com/ArnauGarciaGRBIO/GofCens`

Encoding UTF-8

Depends R (>= 3.5.0), survival, actuar

Imports fitdistrplus, grid, ggplot2, gridExtra, boot, methods

NeedsCompilation no

VignetteBuilder knitr

Suggests knitr, rmarkdown, rms

Repository CRAN

Date/Publication 2025-05-29 20:20:02 UTC

Author Klaus Langohr [aut, cre] (ORCID:
<`https://orcid.org/0000-0001-7075-9192`>),
Mireia Besalú [aut] (ORCID: <`https://orcid.org/0000-0003-0473-2404`>),

Matilde Francisco [aut] (ORCID:
<https://orcid.org/0009-0009-1982-0547>),
Arnau Garcia [aut] (ORCID: <https://orcid.org/0009-0009-7370-6980>),
Guadalupe Gómez [aut] (ORCID: <https://orcid.org/0000-0003-4252-4884>)

Maintainer Klaus Langohr <`klaus.langohr@upc.edu`>

Contents

`GofCens-package` *Goodness-of-Fit Methods for Right-Censored Data.*

Description

This package implements both graphical tools and goodness-of-fit tests for right-censored data. All functions can handle complete as well as right-censored data. It has implemented:

1. Kolmogorov-Smirnov (`KScens`), Cramér-von Mises (`CvMcens`), and Anderson-Darling (`ADcens`) tests, which use the empirical distribution function for complete data and are extended for right-censored data.
2. Generalized chi-squared-type (`chisqcens`) test, which is based on the squared differences between observed and expected counts using random cells with right-censored data.
3. A series of graphical tools such as probability (`probPlot`) or cumulative hazard (`cumhazPlot`) plots to guide the decision about the most suitable parametric model for the data.

Details

All functions of the **GofCens** package can be used to check the goodness of fit of the following 8 distributions. The list shows the parametrizations of the survival functions.

1. Exponential Distribution [Exp(β)]

$$S(t) = e^{-\frac{t}{\beta}}$$

2. Weibull Distribution [Wei(α, β)]
$$S(t) = e^{-(\frac{t}{\beta})^{\alpha}}$$
3. Gumbel Distribution [Gum(μ, β)]
$$S(t) = 1 - e^{-e^{-\frac{t-\mu}{\beta}}}$$
4. Log-Logistic Distribution [LLogis(α, β)]
$$S(t) = \frac{1}{1 + \left(\frac{t}{\beta}\right)^{\alpha}}$$
5. Logistic Distribution [Logis(μ, β)]
$$S(t) = \frac{e^{-\frac{t-\mu}{\beta}}}{1 + e^{-\frac{t-\mu}{\beta}}}$$
6. Log-Normal Distribution [LN(μ, β)]
$$S(t) = \int_{\frac{\log t - \mu}{\beta}}^{\infty} \frac{1}{\sqrt{2\pi}}$$
7. Normal Distribution [N(μ, β)]
$$S(t) = \int_{t}^{\infty} \frac{1}{\beta\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\beta^2}} dx$$
8. 4-Param. Beta Distribution [Beta(α, γ, a, b)]
$$S(t) = 1 - \frac{B_{(\alpha,\gamma,a,b)}(t)}{B(\alpha,\gamma)}$$

The list of the parameters of the theoretical distribution can be set manually using the argument `params` of each function. In that case, the correspondence is: α is the `shape` value, γ is the `shape2` value, μ is the `location` value and β is the `scale` value.

In addition, the functions `KScens`, `CvMcens`, `ADcens`, and `chisqcens` can be used with any other distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined.

Package:	GofCens
Type:	Package
Version:	1.5
Date:	2025-05-29
License:	GPL (>= 2)

Author(s)

Klaus Langohr, Mireia Besalú, Matilde Francisco, Arnau Garcia, Guadalupe Gómez

Maintainer: Klaus Langohr <klaus.langohr@upc.edu>

`ADcens` *Anderson-Darling test for complete and right-censored data*

Description

Function `ADcens` computes the Anderson-Darling test statistic and p-value for right-censored data against eight possible predefined or user-specified distributions using bootstrapping. This function also accounts for complete data.

Usage

```
## Default S3 method:
ADcens(times, cens = rep(1, length(times)),
       distr = c("exponential", "gumbel", "weibull", "normal",
                 "lognormal", "logistic", "loglogistic", "beta"),
       betaLimits = c(0, 1), igumb = c(10, 10), BS = 999,
       params0 = list(shape = NULL, shape2 = NULL,
                      location = NULL, scale = NULL, theta = NULL),
       tol = 1e-04, start = NULL, ...)
## S3 method for class 'formula'
ADcens(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. In addition, if the character string used is `"name"`, every distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined, can be used.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`BS`	Number of bootstrap samples.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`tol`	Precision of survival times.

`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`start`	A named list giving the initial values of parameters of the named distribution or a function of data computing initial values and returning a named list. This argument may be omitted (default) for the eight prespecified distributions. See more details in mledist.
`...`	Additional arguments for the `boot` function of the **boot** package.

Details

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

To avoid long computation times due to bootstrapping, an alternative with complete data is the function `ad.test` of the **goftest** package.

The precision of the survival times is important mainly in the data generation step of the bootstrap samples.

Value

`ADcens` returns an object of class `"ADcens"`.

An object of class `"ADcens"` is a list containing the following components:

`Distribution`	Null distribution.
`Hypothesis`	Parameters under the null hypothesis (if `params0` is provided).
`Test`	Vector containing the value of the Anderson-Darling statistic (`AD`) and the estimated p-value (`p-value`).
`Estimates`	Vector with the maximum likelihood estimates of the parameters of the distribution under study.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).
`BS`	The number of bootstrap samples used.

Warning

If the amount of data is large, the execution time of the function can be elevated. The parameter `BS` can limit the number of random censored samples generated and reduce the execution time.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

G. Marsaglia and J. Marsaglia. *Evaluating the Anderson-Darling Distribution*. In: Journal of Statistical Software, Articles, 9 (2) (2004), 1-5. URL: https://doi.org/10.18637/jss.v009.i02

See Also

Function `ad.test` (Package **goftest**) for complete data and function gofcens for statistics and p-value of the Kolmogorov-Smirnov, Cramér von-Mises, and Anderson-Darling together for right-censored data.

Examples

```
# Complete data
set.seed(123)
ADcens(times = rweibull(100, 12, scale = 4), distr = "weibull",
       BS = 199)
summary(ADcens(times = rweibull(100, 12, scale = 4), distr = "exponential",
        BS = 199), outp = "table", print.BIC = FALSE, print.infoBoot = TRUE)

## Not run:
# Censored data
set.seed(123)
colonsamp <- colon[sample(nrow(colon), 300), ]
ADcens(Surv(time, status) ~ 1, colonsamp, distr = "normal")

## End(Not run)
```

`chisqcens` *General chi-squared statistics for right-censored data.*

Description

Function `chisqcens` computes the general chi-squared test statistic for right-censored data introduced by Kim (1993) and the respective p-value using bootstrapping.

Usage

```
## Default S3 method:
chisqcens(times, cens = rep(1, length(times)), M,
          distr = c("exponential", "gumbel", "weibull", "normal",
                    "lognormal", "logistic", "loglogistic", "beta"),
          betaLimits=c(0, 1), igumb = c(10, 10), BS = 999,
          params0 = list(shape = NULL, shape2 = NULL,
                       location = NULL, scale = NULL, theta = NULL),
          tol = 1e-04, start = NULL, ...)
## S3 method for class 'formula'
chisqcens(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.

`M`	Number indicating the number of cells that will be considered.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. In addition, if the character string used is `"name"`, every distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined, can be used.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`BS`	Number of bootstrap samples.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`tol`	Precision of survival times.
`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`start`	A named list giving the initial values of parameters of the named distribution or a function of data computing initial values and returning a named list. This argument may be omitted (default) for the eight prespecified distributions. See more details in mledist.
`...`	Additional arguments for the `boot` function of the **boot** package.

Details

The function implements the test introduced by Kim (1993) and returns the value of the test statistic.

The cell boundaries of the test are obtained via the quantiles, which are based on the Kaplan-Meier estimate of the distribution function. In the presence of right-censored data, it is possible that not all quantiles are estimated, and in this case, the value of `M` provided by the user is reduced.

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

The precision of the survival times is important mainly in the data generation step of the bootstrap samples.

Value

`chisqcens` returns an object of class `"chisqcens"`.

An object of class `"chisqcens"` is a list containing the following components:

`Distribution`	Null distribution.
`Hypothesis`	Parameters under the null hypothesis (if `params0` is provided).

`Test`	Vector containing the value of the test statistic (`Statistic`) and the estimated p-value (`p-value`).
`Estimates`	Vector with the maximum likelihood estimates of the parameters of the distribution under study.
`StdErrors`	Vector containing the estimated standard errors.
`CellNumber`	Vector with two values: the original cell number introduced by the user and the final cell number used.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).
`BS`	The number of bootstrap samples used.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

J. H. Kim. *Chi-Square Goodness-of-Fit Tests for Randomly Censored Data.* In: The Annals of Statistics, 21 (3) (1993), 1621-1639. URL: https://doi.org/10.1214/aos/1176349275

Examples

```
# Complete data
set.seed(123)
chisqcens(times = rgumbel(100, 12, scale = 4), M = 8, distr = "gumbel", BS = 99)
summary(chisqcens(times = rlogis(100, 20, scale = 3), M = 8, distr = "loglogistic",
                  BS = 105), print.AIC = FALSE, print.infoBoot = TRUE)

## Not run:
# Censored data
set.seed(123)
colonsamp <- colon[sample(nrow(colon), 300), ]
chisqcens(Surv(time, status) ~ 1, colonsamp, M = 6, distr = "normal")

## End(Not run)
```

`cumhazPlot` *Cumulative hazard plots to check the goodness of fit of parametric models*

Description

Function `cumhazPlot` uses the cumulative hazard plot to check if a certain distribution is an appropiate choice for the data.

Usage

```
## Default S3 method:
cumhazPlot(times, cens = rep(1, length(times)), distr = "all6", colour = 1,
           betaLimits = c(0, 1), igumb = c(10, 10), ggp = FALSE, m = NULL,
           prnt = FALSE, degs = 3, print.AIC = TRUE, print.BIC = TRUE,
           outp = c("list", "table"), ...)
## S3 method for class 'formula'
cumhazPlot(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the names of the distributions to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. By default, `distr` is set to `"all6"`, which means that the cumulative hazard plots are drawn for the Weibull, loglogistic, lognormal, Gumbel, logistic, and normal distributions.
`colour`	Colour of the points. Default colour: black.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`ggp`	Logical to use or not the **ggplot2** package to draw the plots. Default is `FALSE`.
`m`	Optional layout for the plots to be displayed.
`prnt`	Logical to indicate if the maximum likelihood estimates of the parameters of all distributions considered should be printed. Default is `TRUE`.
`degs`	Integer indicating the number of decimal places of the numeric results of the output.
`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`print.AIC`	Logical to indicate if the AIC of the model should be printed. Default is `TRUE`
`print.BIC`	Logical to indicate if the BIC of the model should be printed. Default is `TRUE`
`outp`	Indicator of how the output will be displayed. The possible formats are `list` and `table`.
`...`	Optional arguments for function `par`, if `ggp = FALSE`.

Details

The cumulative hazard plot is based on transforming the cumulative hazard function Λ in such a way that it becomes linear in t or $\log(t)$. This transformation is specific for each distribution. The function uses the data to compute the Nelson-Aalen estimator of the cumulative hazard function, $\widehat{\Lambda}$, and the maximum likelihood estimators of the parameters of the theoretical distribution under study. If the distribution fits the data, the plot is expected to be a straight line.

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

Value

If `prnt = TRUE`, the following output is returned:

`Distribution`	Distribution under study.
`Estimates`	A list with the maximum likelihood estimates of the parameters of all distributions considered.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).

In addition, a list with the same contents is returned invisibly.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

Examples

```
# Complete data and default distributions
set.seed(123)
x <- rlogis(1000, 50, 5)
cumhazPlot(x, lwd = 2)

# Censored data comparing three distributions
data(nba)
cumhazPlot(Surv(survtime, cens) ~ 1, nba, distr = c("expo", "normal", "gumbel"),
           outp = "table")
```

`CvMcens` *Cramér-von Mises test for complete and right-censored data*

Description

Function `CvMcens` computes the Cramér-von Mises test statistic and p-value for right-censored data against eight possible predefined or user-specified distributions using bootstrapping. This function also accounts for complete data.

Usage

```
## Default S3 method:
CvMcens(times, cens = rep(1, length(times)),
        distr = c("exponential", "gumbel", "weibull", "normal",
                  "lognormal", "logistic", "loglogistic", "beta"),
        betaLimits = c(0, 1), igumb = c(10, 10), BS = 999,
        params0 = list(shape = NULL, shape2 = NULL,
                       location = NULL, scale = NULL, theta = NULL),
        tol = 1e-04, start = NULL, ...)
## S3 method for class 'formula'
CvMcens(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. In addition, if the character string used is `"name"`, every distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined, can be used.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`BS`	Number of bootstrap samples.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`tol`	Precision of survival times.
`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`start`	A named list giving the initial values of parameters of the named distribution or a function of data computing initial values and returning a named list. This argument may be omitted (default) for the eight prespecified distributions. See more details in mledist.
`...`	Additional arguments for the `boot` function of the **boot** package.

Details

Koziol and Green (1976) proposed a Cramér-von Mises statistic for randomly censored data. This function reproduces this test for a given survival data and a theorical distribution. In presence of ties, different authors provide slightly different definitions of the product-limit estimator, what might provide different values of the test statistic.

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

To avoid long computation times due to bootstrapping, an alternative with complete data is the function `cvm.test` of the **goftest** package.

The precision of the survival times is important mainly in the data generation step of the bootstrap samples.

Value

`CvMcens` returns an object of class `"CvMcens"`.

An object of class `"CvMcens"` is a list containing the following components:

`Distribution`	Null distribution.
`Hypothesis`	Parameters under the null hypothesis (if `params0` is provided).
`Test`	Vector containing the value of the Cramér-von Mises statistic (`CvM`) and the estimated p-value (`p-value`).
`Estimates`	Vector with the maximum likelihood estimates of the parameters of the distribution under study.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).
`BS`	The number of bootstrap samples used.

Warning

If the amount of data is large, the execution time of the function can be elevated. The parameter `BS` can limit the number of random censored samples generated and reduce the execution time.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

J. A. Koziol and S. B. Green. *A Cramér-von Mises statistic for randomly censored data.* In: Biometrika, 63 (3) (1976), 465-474. URL: https://doi.org/10.1093/biomet/63.3.465

A. N. Pettitt and M. A. Stephens. *Modified Cramér-von Mises statistics for censored data.* In: Biometrika, 63 (2) (1976), 291-298. URL: https://doi.org/10.1093/biomet/63.2.291

See Also

Function `cvm.test` (Package **goftest**) for complete data and gofcens for statistics and p-value of Kolmogorov-Smirnov, Cramér von-Mises and Anderson-Darling together for right-censored data.

Examples

```
# Complete data
set.seed(123)
CvMcens(times = rweibull(100, 12, scale = 4), distr = "weibull",
        BS = 199)
summary(CvMcens(times = rweibull(100, 12, scale = 4), distr = "normal",
        BS = 99), degs = 4, print.AIC = FALSE, print.BIC = FALSE)

## Not run:
# Censored data
set.seed(123)
colonsamp <- colon[sample(nrow(colon), 300), ]
CvMcens(Surv(time, status) ~ 1, colonsamp, distr = "normal")

## End(Not run)
```

`gofcens` *Kolmogorov-Smirnov, Cramér-von Mises, and Anderson-Darling tests for complete and right-censored data*

Description

Function `gofcens` computes the Kolmogorov-Smirnov, Cramér-von Mises, and Anderson-Darling statistics and p-values for right-censored data against eight possible predefined or user-specified distributions using bootstrapping. This function also accounts for complete data.

Usage

```
## Default S3 method:
gofcens(times, cens = rep(1, length(times)),
        distr = c("exponential", "gumbel", "weibull", "normal",
                  "lognormal", "logistic", "loglogistic", "beta"),
        betaLimits = c(0, 1), igumb = c(10, 10), BS = 999,
        params0 = list(shape = NULL, shape2 = NULL,
                       location = NULL, scale = NULL, theta = NULL),
        tol = 1e-04, start = NULL, ...)
## S3 method for class 'formula'
gofcens(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. In addition, if the character string used is `"name"`, every distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined, can be used.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`BS`	Number of bootstrap samples.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`tol`	Precision of survival times.
`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`start`	A named list giving the initial values of parameters of the named distribution or a function of data computing initial values and returning a named list. This argument may be omitted (default) for the eight prespecified distributions. See more details in mledist.
`...`	Additional arguments for the `boot` function of the **boot** package.

Details

All p-values are calculated via bootstrapping methods. For the three hypothesis tests, the same data generated with the bootstrapping method are used.

The precision of the survival times is important mainly in the data generation step of the bootstrap samples.

When dealing with complete data, we recommend the use of functions `ks.test` of the **stats** package and `cvm.test` and `ad.test` of the **goftest** package.

Value

`gofcens` returns an object of class `"gofcens"`.

An object of class `"gofcens"` is a list containing the following components:

`Distribution`	Null distribution.

`Hypothesis`	Parameters under the null hypothesis (if `params0` is provided).
`Test`	Vector containing the values of the Kolmogovor-Smirnov (`KS`), Cramér-von Mises (`CvM`), and Anderson-Darling (`AD`) test statistics and the estimated p-value (`p-value`).
`Estimates`	Vector with the maximum likelihood estimates of the parameters of the distribution under study.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).
`BS`	The number of bootstrap samples used.

Warning

If the amount of data is large, the execution time of the function can be elevated. The parameter `BS` can limit the number of random censored samples generated and reduce the execution time.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

J. A. Koziol and S. B. Green. *A Cramér-von Mises statistic for randomly censored data.* In: Biometrika, 63 (3) (1976), 465-474. URL: https://doi.org/10.1093/biomet/63.3.465

A. N. Pettitt and M. A. Stephens. *Modified Cramér-von Mises statistics for censored data.* In: Biometrika, 63 (2) (1976), 291-298. URL: https://doi.org/10.1093/biomet/63.2.291

See Also

ks.test (Package **stats**), `cvm.test` (Package **goftest**), and `ad.test` (Package **goftest**) for complete data, and KScens for the Kolmogorov-Smirnov test for right-censored data, which returns the p-value.

Examples

```
## Not run:
# Complete data
set.seed(123)
gofcens(times = rweibull(100, 12, scale = 4), distr = "weibull", BS = 499)
summary(gofcens(times = rweibull(100, 12, scale = 4), distr = "exponential"),
        outp = "table", print.infoBoot = TRUE)

# Censored data
data(colon)
set.seed(123)
colonsamp <- colon[sample(nrow(colon), 300), ]
gofcens(Surv(time, status) ~ 1, colonsamp, distr = "normal")

## End(Not run)
```

`kmPlot` *Plot of the Kaplan-Meier and parametric estimations*

Description

Function `kmPlot` is a function that generates a plot that combines a Kaplan-Meier survival curve and a parametric survival curve in the same graph. It is useful for comparing non-parametric survival estimates with the fitted parametric survival model.

Usage

```
## Default S3 method:
kmPlot(times, cens = rep(1, length(times)), distr = "all6",
       colour = c("black", "blue", "cornflowerblue"),
       betaLimits = c(0, 1), igumb = c(10, 10), ggp = FALSE, m = NULL,
       prnt = FALSE, degs = 3, print.AIC = TRUE, print.BIC = TRUE, ...)
## S3 method for class 'formula'
kmPlot(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), the exponential (`"exponential"`) and the beta (`"beta"`) distribution. Default is `"all6"` and includes the fisrt 6 listed which are the most used distributions.
`colour`	Vector with three components indicating the colours of the displayed plots. The first element is for the survival curve, the second for the Kaplan-Meier curve, and the last one for the confidence intervals.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`ggp`	Logical to use or not the **ggplot2** package to draw the plots. Default is `FALSE`.
`m`	Optional layout for the plots to be displayed.
`prnt`	Logical to indicate if the maximum likelihood estimates of the parameters should be printed. Default is `TRUE`.
`degs`	Integer indicating the number of decimal places of the numeric results of the output.

`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`print.AIC`	Logical to indicate if the AIC of the model should be printed. Default is `TRUE`
`print.BIC`	Logical to indicate if the BIC of the model should be printed. Default is `TRUE`
`...`	Optional arguments for function `par`, if `ggp = FALSE`.

Details

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

Value

If `prnt = TRUE`, the following output is returned:

`Distribution`	Distribution under study.
`Estimates`	A list with the maximum likelihood estimates of the parameters of all distributions considered.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).

In addition, a list with the same contents is returned invisibly.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

Peterson Jr, Arthur V. *Expressing the Kaplan-Meier estimator as a function of empirical subsurvival functions*. In: Journal of the American Statistical Association 72.360a (1977): 854-858. URL: https://doi.org/10.1080/01621459.1977.10479970

Examples

```
# Plots for complete data and default distributions
set.seed(123)
x <- rexp(1000, 0.1)
kmPlot(x)

# Plots for censored data using ggplot2
kmPlot(Surv(time, status) ~ 1, colon, distr= "lognormal", ggp = TRUE)

# Plots for censored data from three distributions
data(nba)
kmPlot(Surv(survtime, cens) ~ 1, nba, distr = c("normal", "weibull", "lognormal"),
       prnt = FALSE)
```

`KScens` *Kolmogorov-Smirnov test for complete and right-censored data*

Description

Function `KScens` computes the Kolmogorov-Smirnov statistic and p-value for right-censored data against eight possible predefined or user-specified distributions using either bootstrapping or a modified test. This function also accounts for complete data.

Usage

```
## Default S3 method:
KScens(times, cens = rep(1, length(times)),
       distr = c("exponential", "gumbel", "weibull", "normal",
                 "lognormal", "logistic", "loglogistic", "beta"),
       betaLimits = c(0, 1), igumb = c(10, 10), BS = 999,
       params0 = list(shape = NULL, shape2 = NULL, location = NULL,
                      scale = NULL, theta = NULL),
       tol = 1e-04, boot = TRUE, start = NULL, ...)
## S3 method for class 'formula'
KScens(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution. In addition, if the character string used is `"name"`, every distribution for which the corresponding density (`dname`), distribution (`pname`), and random generator (`rname`) functions are defined, can be used.
`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`BS`	Number of bootstrap samples.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`tol`	Precision of survival times.

`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`boot`	Logical to indicate if the p-value is computed using bootstrapping or using the the modified Kolmogorov-Smirnov test (see details). Default is `TRUE`.
`start`	A named list giving the initial values of parameters of the named distribution or a function of data computing initial values and returning a named list. This argument may be omitted (default) for the eight prespecified distributions. See more details in mledist.
`...`	Additional arguments for the `boot` function of the **boot** package.

Details

By default the p-value is computed via bootstrapping methods.

The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

To avoid long computation times due to bootstrapping, an alternative with complete data is the function `ks.test` of the **stats** package.

The precision of the survival times is important mainly in the data generation step of the bootstrap samples.

If `boot = FALSE` a modified test is used to compute the p-value. Fleming et al. (1980) proposed a modified Kolmogorov-Smirnov test to use with right-censored data. This function reproduces this test for a given survival data and a theorical distribution. The approximation for the p-value is acceptable when it is smaller than 0.8 and excellent when it is smaller than 0.2. The output of the function follows the notation of Fleming et al. (1980).

In presence of ties, different authors provide slightly different definitions of $\widehat{F}_n(t)$, with which other values of the test statistic might be obtained.

Value

`KScens` returns an object of class `"KScens"`.

An object of class `"KScens"` is a list containing the following components:

`Distribution`	Null distribution.
`Hypothesis`	Parameters under the null hypothesis (if `params0` is provided).
`Test`	Vector containing the value of the modified Kolmogorov-Smirnov statistic (`A`), the estimated p-value (`p-value`), the estimation of the image of the last recorded time (`F(y_m)`) and the last recorded time (`y_m`).
`Estimates`	Vector with the maximum likelihood estimates of the parameters of the distribution under study.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).
`BS`	The number of bootstrap samples used. If the modified test is used, a 0 is returned.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

T. R. Fleming et al. *Modified Kolmogorov-Smirnov test procedure with application to arbitrarily right-censored data*. In: Biometrics 36 (1980), 607-625. URL: https://doi.org/10.2307/2556114

See Also

Function ks.test (Package **stats**) for complete data and gofcens for statistics and p-value of Kolmogorov-Smirnov, Cramér von-Mises and Anderson-Darling together for right-censored data.

Examples

```
# Complete data with bootstrapping
set.seed(123)
KScens(times = rweibull(100, 12, scale = 4), distr = "weibull", BS = 99)

# Censored data with bootstrapping
KScens(Surv(time, status) ~ 1, colon, distr = "norm", BS = 99)

# Censored data using the modified test
KScens(Surv(time, status) ~ 1, colon, distr = "norm", boot = FALSE)

data(nba)
summary(KScens(Surv(survtime, cens) ~ 1, nba, "logis", boot = FALSE), degs = 2)
KScens(Surv(survtime, cens) ~ 1, nba, "beta", betaLimits = c(0, 80),
       boot = FALSE)
```

`Methods tests` *Methods for* `ADcens`*,* `chisqcens`*,* `CvMcens`*,* `gofcens` *and* `KScens` *objects.*

Description

Print, summary and print summary methods for `ADcens`, `chisqcens`, `CvMcens`, `gofcens` and `KScens` objects.

Usage

```
## S3 method for class 'ADcens'
print(x, ...)

## S3 method for class 'ADcens'
summary(object, outp = c("list", "table"),
        print.AIC = TRUE, print.BIC = TRUE, print.infoBoot = FALSE, ...)
```

```
## S3 method for class 'ADcens'
print.summary(x, degs = 3, ...)

## S3 method for class 'chisqcens'
print(x, ...)

## S3 method for class 'chisqcens'
summary(object, outp = c("list", "table"),
        print.AIC = TRUE, print.BIC = TRUE, print.infoBoot = FALSE, ...)

## S3 method for class 'chisqcens'
print.summary(x, degs = 3, ...)

## S3 method for class 'CvMcens'
print(x, ...)

## S3 method for class 'CvMcens'
summary(object, outp = c("list", "table"),
        print.AIC = TRUE, print.BIC = TRUE, print.infoBoot = FALSE, ...)

## S3 method for class 'CvMcens'
print.summary(x, degs = 3, ...)

## S3 method for class 'gofcens'
print(x, ...)

## S3 method for class 'gofcens'
summary(object, outp = c("list", "table"),
        print.AIC = TRUE, print.BIC = TRUE, print.infoBoot = FALSE, ...)

## S3 method for class 'gofcens'
print.summary(x, degs = 3, ...)

## S3 method for class 'KScens'
print(x, ...)

## S3 method for class 'KScens'
summary(object, outp = c("list", "table"),
        print.AIC = TRUE, print.BIC = TRUE, print.infoBoot = FALSE, ...)

## S3 method for class 'KScens'
print.summary(x, degs = 3, ...)
```

Arguments

`object, x`	An object of class `ADcens` `chisqcens`, `CvMcens`, `gofcens` or `KScens`.
`outp`	Indicator of how the output will be displayed. The possible formats are `list`

	and `table`.
`print.AIC`	Logical to indicate if the AIC of the model should be printed. Default is `TRUE`
`print.BIC`	Logical to indicate if the BIC of the model should be printed. Default is `TRUE`
`print.infoBoot`	Logical to indicate if the number of bootstrap samples used should be printed. Default is `FALSE`
`degs`	Integer indicating the number of decimal places of the numeric results of the output. By default is 3.
`...`	Optional arguments.

Details

`print()` Prints only essential information of the object.

`summary()` Produces result summaries, printing information in a more extensive form and with different formats available.

`print.summary()` Prints the summary output.

Value

`print()` Basic information is returned on the screen.

`summary()` A list with the elements:

- `Distribution`: Null distribution.
- `Hypothesis`: Parameters under the null hypothesis (if `params0` is provided).
- `Test`: Vector containing the value of the Anderson-Darling statistic (`AD`) and the estimated p-value (`p-value`).
- `Estimates`: Vector with the maximum likelihood estimates of the parameters of the distribution under study.
- `StdErrors`: Vector containing the estimated standard errors.
- `aic`: The Akaike information criterion.
- `bic`: The so-called BIC or SBC (Schwarz Bayesian criterion).
- `BS`: The number of bootstrap samples used.

`print.summary()` `x`, with the invisible flag set to prevent printing.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

Examples

```
# Generating data
set.seed(123)
survt <- round(rlnorm(300, 2, 1), 2)
censt <- round(rexp(300, 1 / 20), 2)
times <- pmin(survt, censt)
cens <- as.numeric(survt <= censt)
```

```
# Print method
set.seed(123)
CvMcens(times, cens, distr = "weibull", BS = 99)

# List output from summary method
set.seed(123)
summary(ADcens(times = rweibull(100, 12, scale = 4), distr = "weibull",
        BS = 149))

# Table output from summary method
set.seed(123)
summary(ADcens(times = rweibull(100, 12, scale = 4), distr = "weibull",
               BS = 99), outp = "table")
```

`nba` *Survival times of former NBA players.*

Description

Survival times of former NBA players after their NBA career.

Usage

```
data("nba")
```

Format

A data frame with 3962 observations on the following 3 variables.

`id` Player ID

`survtime` Time (in years) from end of NBA career until either death or July 31, 2019.

`cens` Death indicator (1, exact survival time; 0, right-censored survival time).

Details

The survival times of former NBA players were analyzed by Martínez et al. (2022).

Source

J. A. Martínez, K. Langohr, J. Felipo, L. Consuegra and M. Casals. *Data set on mortality of national basketball association (NBA) players.* In: Data in Brief, 45 (2022). URL: https://doi.org/10.1016/j.dib.2022.108615

Examples

```
data(nba)
cumhazPlot(Surv(survtime, cens) ~ 1, nba)
```

---

probPlot *Probability plots to check the goodness of fit of parametric models*

Description

`probPlot` provides four types of probability plots: P-P plot, Q-Q plot, Stabilised probability plot, and Empirically Rescaled plot to check if a certain distribution is an appropiate choice for the data.

Usage

```
## Default S3 method:
probPlot(times, cens = rep(1, length(times)),
         distr = c("exponential", "gumbel", "weibull", "normal",
                   "lognormal", "logistic", "loglogistic", "beta"),
         plots = c("PP", "QQ", "SP", "ER"),
         colour = c("green4", "deepskyblue4", "yellow3",
                    "mediumvioletred"), mtitle = TRUE, ggp = FALSE,
         m = NULL, betaLimits = c(0, 1), igumb = c(10, 10),
         prnt = FALSE, degs = 3,
         params0 = list(shape = NULL, shape2 = NULL,
                        location = NULL, scale = NULL), print.AIC = TRUE,
                        print.BIC = TRUE, ...)
## S3 method for class 'formula'
probPlot(formula, data, ...)
```

Arguments

`times`	Numeric vector of times until the event of interest.
`cens`	Status indicator (1, exact time; 0, right-censored time). If not provided, all times are assumed to be exact.
`distr`	A string specifying the name of the distribution to be studied. The possible distributions are the exponential (`"exponential"`), the Weibull (`"weibull"`), the Gumbel (`"gumbel"`), the normal (`"normal"`), the lognormal (`"lognormal"`), the logistic (`"logistic"`), the loglogistic (`"loglogistic"`), and the beta (`"beta"`) distribution.
`plots`	Vector stating the plots to be displayed. Possible choices are the P-P plot (`"PP"`), the Q-Q plot (`"QQ"`), the Stabilised Probability plot (`"SP"`), and the Empirically Rescaled plot (`"ER"`). By default, all four plots are displayed.
`colour`	Vector indicating the colours of the displayed plots. The vector will be recycled if its length is smaller than the number of plots to be displayed.
`mtitle`	Logical to add or not the title "Probability plots for a `distr` distribution" to the plot. Default is `TRUE`.
`ggp`	Logical to use or not the **ggplot2** package to draw the plots. Default is `FALSE`.
`m`	Optional layout for the plots to be displayed.

`betaLimits`	Two-components vector with the lower and upper bounds of the Beta distribution. This argument is only required, if the beta distribution is considered.
`igumb`	Two-components vector with the initial values for the estimation of the Gumbel distribution parameters.
`prnt`	Logical to indicate if the maximum likelihood estimates of the parameters should be printed. Default is `TRUE`.
`degs`	Integer indicating the number of decimal places of the numeric results of the output.
`params0`	List specifying the parameters of the theoretical distribution. By default, parameters are set to `NULL` and estimated with the maximum likelihood method. This argument is only considered, if all parameters of the studied distribution are specified.
`formula`	A formula with a numeric vector as response (which assumes no censoring) or `Surv` object.
`data`	Data frame for variables in `formula`.
`print.AIC`	Logical to indicate if the AIC of the model should be printed. Default is `TRUE`
`print.BIC`	Logical to indicate if the BIC of the model should be printed. Default is `TRUE`
`...`	Optional arguments for function `par`, if `ggp = FALSE`.

Details

By default, function `probPlot` draws four plots: P-P plot, SP plot, Q-Q plot, and EP plot. Following, a description is given for each plot.

The **Probability-Probability plot** (P-P plot) depicts the empirical distribution, $\widehat{F}(t)$, which is obtained with the Kaplan-Meier estimator if data are right-censored, versus the theoretical cumulative distribution function (cdf), $\widehat{F_0}(t)$. If the data come from the chosen distribution, the points of the resulting graph are expected to lie on the identity line.

The **Stabilised Probability plot** (SP plot), proposed by Michael (1983), is a transformation of the P-P plot. It stabilises the variance of the plotted points. If $F_0 = F$ and the parameters of F_0 are known, $\widehat{F_0}(t)$ corresponds to the cdf of a uniform order statistic, and the arcsin transformation stabilises its variance. If the data come from distribution F_0, the SP plot will resemble the identity line.

The **Quartile-Quartile plot** (Q-Q plot) is similar to the P-P plot, but it represents the sample quantiles versus the theoretical ones, that is, it plots t versus $\widehat{F}_0^{-1}(\widehat{F}(t))$. Hence, if F_0 fits the data well, the resulting plot will resemble the identity line.

A drawback of the Q-Q plot is that the plotted points are not evenly spread. Waller and Turnbull (1992) proposed the **Empirically Rescaled plot** (EP plot), which plots $\widehat{F}_u(t)$ against $\widehat{F}_u(\widehat{F}_0^{-1}(\widehat{F}(t)))$, where $\widehat{F}_u(t)$ is the empirical cdf of the points corresponding to the uncensored observations. Again, if $\widehat{F}_0$ fits the data well, the ER plot will resemble the identity line.

By default, all four probability plots are drawn and the maximum likelihood estimates of the parameters of the chosen parametric model are returned. The parameter estimation is acomplished with the `fitdistcens` function of the **fitdistrplus** package.

Value

If `prnt = TRUE`, the following output is returned:

`Distribution`	Distribution under study.
`Parameters`	Parameters used to draw the plots (if `params0` is provided).
`Estimates`	A list with the maximum likelihood estimates of the parameters of all distributions considered.
`StdErrors`	Vector containing the estimated standard errors.
`aic`	The Akaike information criterion.
`bic`	The so-called BIC or SBC (Schwarz Bayesian criterion).

In addition, a list with the same contents is returned invisibly.

Author(s)

K. Langohr, M. Besalú, M. Francisco, A. Garcia, G. Gómez.

References

J. R. Michael. *The Stabilized Probability Plot*. In: Biometrika 70 (1) (1983), 11-17. URL: https://doi.org/10.1093/biomet/70.1.11

L.A. Waller and B.W. Turnbull. *Probability Plotting with Censored Data*. In: American Statistician 46 (1) (1992), 5-12. URL: https://doi.org/10.1080/00031305.1992.10475837

Examples

```
# P-P, Q-Q, SP, and EP plots for complete data
set.seed(123)
x <- rlnorm(1000, 3, 2)
probPlot(x)
probPlot(x, distr = "lognormal")

# P-P, Q-Q, SP, and EP plots for censored data using ggplot2
probPlot(Surv(time, status) ~ 1, colon, "weibull", ggp = TRUE)

# P-P, Q-Q and SP plots for censored data and lognormal distribution
data(nba)
probPlot(Surv(survtime, cens) ~ 1, nba, "lognorm", plots = c("PP", "QQ", "SP"),
         ggp = TRUE, m = matrix(1:3, nr = 1))
```

Index