

# Package ‘bccp’

May 18, 2021

**Type** Package

**Title** Bias Correction under Censoring Plan

**Author** Mahdi Teimouri

**Maintainer** Mahdi Teimouri <teimouri@aut.ac.ir>

**Description** Developed for the following tasks. Simulating, computing maximum likelihood estimator, computing the Fisher information matrix, computing goodness-of-fit measures, and correcting bias of the ML estimator for a wide range of distributions fitted to units placed on progressive type-I interval censoring and progressive type-II censoring plans. The methods of Cox and Snell (1968) <doi:10.1111/j.2517-6161.1968.tb00724.x> and bootstrap method for computing the bias-corrected maximum likelihood estimator.

**Encoding** UTF-8

**License** GPL (>= 2)

**Depends** R(>= 3.1.0)

**Imports** pracma

**Repository** CRAN

**Version** 0.5.0

**Date** 2021-05-18

**NeedsCompilation** no

**Date/Publication** 2021-05-18 04:10:05 UTC

## R topics documented:

|                       |    |
|-----------------------|----|
| bootbctype1 . . . . . | 2  |
| bootbctype2 . . . . . | 3  |
| coxbctype1 . . . . .  | 5  |
| coxbctype2 . . . . .  | 7  |
| fitype1 . . . . .     | 8  |
| fitype2 . . . . .     | 10 |
| goftype1 . . . . .    | 11 |
| goftype2 . . . . .    | 13 |
| mletype1 . . . . .    | 14 |

|                                   |    |
|-----------------------------------|----|
| mle <span>type</span> 2 . . . . . | 15 |
| plasma . . . . .                  | 17 |
| relief . . . . .                  | 17 |
| r <span>type</span> 1 . . . . .   | 18 |
| r <span>type</span> 2 . . . . .   | 19 |
| simpson . . . . .                 | 20 |
| welcome . . . . .                 | 21 |

## Index 22

---

|             |  |
|-------------|--|
| bootbctype1 | <i>Computing the bias corrected maximum likelihood estimator under progressive type-I interval censoring scheme using the Bootstrap resampling</i> |
|-------------|--|

---

### Description

Computes the bias corrected maximum likelihood estimator (MLE) under progressive type-I interval censoring scheme using the Bootstrap resampling. It works by obtaining the empirical distribution of the MLE using bootstrap approach and then constructing the percentile confidence intervals (PCI) suggested by DiCiccio and Tibshirani (1987).

### Usage

```
bootbctype1(plan, param, mle, cdf, lb = 0, ub = Inf, nboot = 200, coverage = 0.95)
```

### Arguments

|          |   |
|----------|---|
| plan     | Censoring plan for progressive type-I interval censoring scheme. It must be given as a <code>data.frame</code> that includes vector of upper bounds of the censoring times $T$ , vector of number of failed subjects $X$ , vector of removed subjects in each interval $R$ , and percentage of the removed alive items in each interval $P$ . |
| param    | Vector of the of the family parameter's names.  |
| mle      | Vector of the maximum likelihood estimators.  |
| cdf      | Expression of the cumulative distribution function.   |
| lb       | Lower bound of the family support. That is zero by default.   |
| ub       | Upper bound of the family's support. That is <code>Inf</code> by default.   |
| nboot    | Number of Bootstrap resampling.   |
| coverage | Confidence or coverage level for constructing percentile confidence intervals. That is 0.95 by default.   |

### Value

A list of two parts including: 1- bias of MLE with respect to mean, bias of MLE with respect to median, lower bound of the percentile confidence interval (LPCI), and upper bound of the percentile confidence interval (UPCI) at the given coverage level and 2- covariance matrix of the MLE obtained using bootstrapping.

**Author(s)**

Mahdi Teimouri

**References**

T. J. DiCiccio and R. Tibshirani 1987. Bootstrap confidence intervals and bootstrap approximations. *Journal of the American Statistical Association*, 82, 163-170.

A. J. Lemonte, F. Cribari-Neto, and K. L. P. Vasconcellos 2007. Improved statistical inference for the two-parameter Birnbaum-Saunders distribution. *Computational Statistics and Data Analysis*, 51, 4656-4681.

**Examples**

```
data(plasma)
n <- 112
param <- c("lambda", "beta")
mle <- c(1.5, 0.05)
cdf <- quote( (1-exp( -(x*beta)))^lambda )
pdf <- quote( lambda*(1-exp( -(x*beta)))^(lambda-1)*beta*exp( -(x*beta)) )
lb <- 0
plan <- rtype1(n = n, P = plasma$P, T = plasma$upper, param = param, mle = mle,
              cdf.expression = FALSE, pdf.expression = TRUE, cdf = cdf, pdf = pdf,
              lb = lb)
ub <- Inf
nboot <- 200
coverage <- 0.95
bootbctype1(plan = plan, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub, nboot = nboot,
            coverage = coverage)
```

bootbctype2

---

*Computing the bias corrected maximum likelihood estimator under progressive type-I interval censoring scheme using the Bootstrap resampling*

---

**Description**

Computes the bias corrected maximum likelihood estimator under progressive type-I interval censoring scheme using the Bootstrap resampling. It works by obtaining the empirical distribution of the MLE using bootstrap approach and then constructing the percentile confidence intervals (PCI) suggested by DiCiccio and Tibshirani (1987).

**Usage**

```
bootbctype2(plan, param, mle, cdf, pdf, lb = 0, ub = Inf, nboot = 200, coverage = 0.95)
```

**Arguments**

|          |   |
|----------|---|
| plan     | Censoring plan for progressive type-II censoring scheme. It must be given as a <code>data.frame</code> that includes number of failed items $X$ , and vector of removed items $R$ . |
| param    | Vector of the of the family parameter's names.  |
| mle      | Vector of the maximum likelihood estimators.  |
| cdf      | Expression of the cumulative distribution function.   |
| pdf      | Expression for the probability density function.  |
| lb       | Lower bound of the family support. That is zero by default.   |
| ub       | Upper bound of the family's support. That is <code>Inf</code> by default.   |
| nboot    | Number of Bootstrap resampling.   |
| coverage | Confidence or coverage level for constructing percentile confidence intervals. That is 0.95 by default.   |

**Details**

For some families of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., `lb` that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

**Value**

A list of the outputs including a matrix that represents the variance-covariance matrix of the uncorrected MLE, a matrix that represents the variance-covariance matrix of the corrected MLE, the lower LPCI, and upper UPCI, bounds of 95% percentile confidence interval for `param`, the ML estimator, bias value, and bias-corrected estimator. Finally, the goodness-of-fit measures consists of Anderson-Darling (AD), Cramer-von Misses (CVM), and Kolmogorov-Smirnov (KS) statistics.

**Author(s)**

Mahdi Teimouri

**References**

- T. J. DiCiccio and R. Tibshirani 1987. Bootstrap confidence intervals and bootstrap approximations. *Journal of the American Statistical Association*, 82, 163-170.
- A. J. Lemonte, F. Cribari-Neto, and K. L. P. Vasconcellos 2007. Improved statistical inference for the two-parameter Birnbaum-Saunders distribution. *Computational Statistics and Data Analysis*, 51, 4656-4681.

**Examples**

```
n <- 20
R <- c(9, rep(0, 10) )
param <- c("alpha", "beta")
mle <- c(0.80, 12)
```

```

cdf <- quote( 1-exp( -(x/beta)^alpha ) )
pdf <- quote( alpha/beta*(x/beta)^(alpha-1)*exp( -(x/beta)^alpha ) )
lb <- 0
ub <- Inf
nboot <- 200
coverage <- 0.95
plan <- rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
bootctype2(plan = plan, param = param, mle = mle, cdf = cdf, pdf = pdf, lb = lb, ub = ub,
nboot = nboot, coverage = coverage)

```

---

coxctype1

---

*Computing the bias corrected maximum likelihood estimator.*


---

## Description

Computing the bias corrected maximum likelihood estimator (MLE) for the parameters of the general family of distributions under progressive type-I interval censoring scheme. Let  $y_1, y_2, \dots, y_n$  represent the lifetimes of  $n$  items that independently follow the cumulative distribution function (cdf)  $F(\cdot, \theta_j)$  under a progressive type-I interval censoring scheme. We use *bctypei* to compute the bias corrected ML estimator using the method of Cox and Snell (1968). Let  $[T_0 - T_1], [T_1 - T_2], \dots, [T_{m-1} - T_m]$  show a number of  $m$  censoring time intervals,  $\mathbf{X} = (X_1, X_2, \dots, X_m)$  denotes the vector of failed items, and  $\mathbf{R} = (R_1, R_2, \dots, R_m)$  represents the vector of removed alive items in each interval, from  $m \geq 1$ . A schematic, given by the following, displays the progressive type-I interval censoring scheme. We note that the sample size  $n$  is  $n = \sum_{i=1}^m X_i + \sum_{i=1}^m R_i$ . Furthermore,  $R_i$  can be determined by the pre-specified percentage of the remaining surviving items at  $T_i$  or equivalently  $R_i = \lfloor P_i X_i \rfloor$ , for  $i = 1, \dots, m$ . Here,  $\lfloor z \rfloor$  denotes the largest integer less than or equal to  $z$ .

| row     | Time interval        | $\mathbf{X}$ | $\mathbf{R}$ |
|---------|----------------------|--------------|--------------|
| 1       | $[T_0, T_1)$         | $X_1$        | $R_1$        |
| 2       | $[T_1, T_2)$         | $X_2$        | $R_2$        |
| .       | .                    | .            | .            |
| .       | .                    | .            | .            |
| .       | .                    | .            | .            |
| $m - 1$ | $[T_{m-2}, T_{m-1})$ | $X_{m-1}$    | $R_{m-1}$    |
| $m$     | $[T_{m-1}, T_m)$     | $X_m$        | $R_m$        |

## Usage

```

coxctype1(plan, param, mle, cdf.expression = FALSE, pdf.expression = TRUE, cdf, pdf
, lb = 0)

```

## Arguments

**plan** Censoring plan for progressive type-I interval censoring scheme. It must be given as a data.frame that includes vector of upper bounds of the censoring

|                |   |
|----------------|---|
|                | times T, vector of number of failed items X, and vector of removed items in each interval R.          |
| param          | Vector of the of the family parameter's names.  |
| mle            | A vector that contains MLE of the parameters.   |
| cdf.expression | Logical. That is TRUE, if there is a closed form expression for the cumulative distribution function. |
| pdf.expression | Logical. That is TRUE, if there is a closed form expression for the probability density function.     |
| cdf            | Expression of the cumulative distribution function.   |
| pdf            | Expression of the probability density function.   |
| lb             | Lower bound of the family's support. That is zero by default.   |

### Details

For some families of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., lb that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

### Value

A list of the outputs including: a matrix that represents the variance-covariance matrix of the MLE, a matrix that represents the variance-covariance matrix of the bias corrected MLE, a list of three outputs including MLE, bias of MLE, and bias corrected MLE, a list of goodness-of-fit measures consists of Anderson-Darling (AD), Cramer-von Misses (CVM), and Kolmogorov-Smirnov (KS) statistics.

### Author(s)

Mahdi Teimouri

### References

- Z. Chen 2000. A new two-parameter lifetime distribution with bathtub shape or increasing failure rate function, *Statistics & Probability Letters*, 49 (2), 155-161.
- D. G. Chen and Y. L. Lio 2010. Parameter estimations for generalized exponential distribution under progressive type-I interval censoring, *Computational Statistics and Data Analysis*, 54, 1581-1591.
- D. R. Cox and E. J. Snell, 1968. A general definition of residuals. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2), 248-265.
- M. Teimouri, 2020. Bias corrected maximum likelihood estimators under progressive type-I interval censoring scheme, <https://doi.org/10.1080/03610918.2020.1819320>

**Examples**

```

data(plasma, package="bccp")
plan <- data.frame(T = plasma$upper, X = plasma$X, P = plasma$P, R = plasma$R)
param <- c("lambda", "beta")
mle <- c(1.4, 0.05)
pdf <- quote( lambda*(1-exp( -(x*beta)))^(lambda-1)*beta*exp( -(x*beta)) )
cdf <- quote( (1-exp( -(x*beta)))^lambda )
lb <- 0
coxctype1(plan = plan, param = param, mle = mle, cdf.expression = FALSE, pdf.expression = TRUE,
          cdf = cdf, pdf = pdf, lb = lb)

```

---

coxctype2

*Computing the bias corrected maximum likelihood estimator.*


---

**Description**

Computing the bias corrected maximum likelihood estimator (MLE) for the parameters of the general family of distributions under progressive type-II censoring scheme.

**Usage**

```
coxctype2(plan, param, mle, cdf, pdf, lb = 0, ub = Inf, N = 100)
```

**Arguments**

|       |   |
|-------|---|
| plan  | Censoring plan for progressive type-II censoring scheme. It must be given as a <code>data.frame</code> that includes number of failed items $X$ , and vector of removed items $R$ . |
| param | Vector of the of the family parameter's names.  |
| mle   | Vector of the maximum likelihood estimators.  |
| cdf   | Expression for the cumulative distribution function.  |
| pdf   | Expression for the probability density function.  |
| lb    | Lower bound of the family's support. That is zero by default.   |
| ub    | Upper bound of the family support. That is <code>Inf</code> by default.   |
| N     | An even integer value indicating the number of subdivisions for applying Simpson's integration method.  |

**Details**

For some families of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e.,  $lb$  that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

**Value**

A list of the outputs including: a matrix that represents the variance-covariance matrix of the MLE, a matrix that represents the variance-covariance matrix of the bias corrected MLE, a list of three outputs including MLE, bias of MLE, and bias corrected MLE, a list of goodness-of-fit measures consists of Anderson-Darling (AD), Cramer-von Misses (CVM), and Kolmogorov-Smirnov (KS), statistics.

**Author(s)**

Mahdi Teimouri

**References**

D. R. Cox and E. J. Snell 1968. A general definition of residuals. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2), 248-265.

M. Teimouri and S. Nadarajah 2016. Bias corrected MLEs under progressive type-II censoring scheme, *Journal of Statistical Computation and Simulation*, 86 (14), 2714-2726.

**Examples**

```
n <- 10
R <- c(5, rep(0, n-6) )
param <- c("alpha", "beta")
mle <- c(2,6)
pdf <- quote( alpha/beta*(x/beta)^(alpha-1)*exp( -(x/beta)^alpha ) )
cdf <- quote( 1-exp( -(x/beta)^alpha ) )
lb <- 0
ub <- Inf
N <- 100
plan <- rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
coxctype2(plan = plan, param = param, mle = mle, cdf = cdf, pdf = pdf, lb = lb, ub = ub, N = N)
```

---

fitype1

*Computing the Fisher information matrix under progressive type-I interval censoring scheme*

---

**Description**

Computes the Fisher information matrix under progressive type-I interval censoring scheme. The Fisher information matrix is given by

$$I_{rs} = -E\left(\frac{\partial^2 l(\Theta)}{\partial \theta_r \partial \theta_s}\right),$$

where

$$l(\Theta) = \log L(\Theta) \propto \sum_{i=1}^m X_i \log[F(t_i; \Theta) - F(t_{i-1}; \Theta)] + \sum_{i=1}^m R_i [1 - F(t_i; \Theta)],$$

in which  $F(\cdot; \Theta)$  is the family cumulative distribution function for  $\Theta = (\theta_1, \dots, \theta_k)^T$  and  $r, s = 1, \dots, k$ .



**Usage**

```
fitype1(plan, param, mle, cdf.expression = FALSE, pdf.expression = TRUE, cdf, pdf
, lb = 0)
```

**Arguments**

|                |  |
|----------------|--|
| plan           | Censoring plan for progressive type-I interval censoring scheme. It must be given as a data.frame that includes vector of upper bounds of the censoring times T, vector of number of failed subjects X, vector of removed subjects in each interval R, and percentage of the removed alive items in each interval P. |
| param          | Vector of the of the family parameter's names.   |
| mle            | Vector of the maximum likelihood estimators.   |
| cdf.expression | Logical. That is TRUE, if there is a closed form expression for the cumulative distribution function.  |
| pdf.expression | Logical. That is TRUE, if there is a closed form expression for the probability density function.  |
| cdf            | Expression of the cumulative distribution function.  |
| pdf            | Expression of the probability density function.  |
| lb             | Lower bound of the family support. That is zero by default.  |

**Details**

For some families of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., lb that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

**Value**

Matrices that represent the expected and observed Fisher information matrices.

**Author(s)**

Mahdi Teimouri

**References**

N. Balakrishnan and E. Cramer. 2014. *The art of progressive censoring*. New York: Springer.

D. G. Chen and Y. L. Lio 2010. Parameter estimations for generalized exponential distribution under progressive type-I interval censoring, *Computational Statistics and Data Analysis*, 54, 1581-1591.

M. Teimouri 2020. Bias corrected maximum likelihood estimators under progressive type-I interval censoring scheme, *Communications in Statistics-Simulation and Computation*, doi.org/10.1080/03610918.2020.1819320

**Examples**

```

data(plasma)
  n <- 20
param <- c("alpha", "beta")
mle <- c(0.4, 0.05)
cdf <- quote( 1-exp( beta*(1-exp( x^alpha )) ) )
pdf <- quote( exp( beta*(1-exp( x^alpha )) )*( beta*(exp( x^alpha )*(x^(alpha-1)*alpha) )) )
lb <- 0
plan <- rtype1(n = n, P = plasma$P, T = plasma$upper, param = param, mle = mle, cdf.expression
  = FALSE, pdf.expression = TRUE, cdf = cdf, pdf = pdf, lb = lb)
fitype1(plan = plan, param = param, mle = mle, cdf.expression = FALSE, pdf.expression = TRUE, cdf =
  cdf, pdf = pdf, lb = lb)

```

fitype2

*Computing the Fisher information matrix under progressive type-II censoring scheme*

**Description**

Computes the Fisher information matrix under progressive type-I interval censoring scheme. The Fisher information matrix is given by

$$I_{rs} = -E\left(\frac{\partial^2 l(\Theta)}{\partial \theta_r \partial \theta_s}\right),$$

where

$$l(\Theta) = \log L(\Theta) \propto C \sum_{i=1}^m \log f(x_{i:m:n}; \Theta) + \sum_{i=1}^m R_i \log [1 - F(x_{i:m:n}; \Theta)],$$

in which  $F(\cdot; \Theta)$  is the family cumulative distribution function for  $\Theta = (\theta_1, \dots, \theta_k)^T$  and  $r, s = 1, \dots, k$ , and  $C = n(n - R_1 - 1)(n - R_1 - R_2 - 2) \dots (n - R_1 - R_2 - \dots - R_{m-1} - m + 1)$ .

**Usage**

```
fitype2(plan, param, mle, cdf, pdf, lb = 0, ub = Inf, N = 100)
```

**Arguments**

|       |  |
|-------|--|
| plan  | Censoring plan for progressive type-II censoring scheme. It must be given as a data.frame that includes number of failed items X, and vector of removed items R. |
| param | Vector of the of the family parameter's names.   |
| mle   | Vector of the maximum likelihood estimators.   |
| cdf   | Expression for the cumulative distribution function.   |
| pdf   | Expression for the probability density function.   |
| lb    | Lower bound of the family support. That is zero by default.  |
| ub    | Upper bound of the family support. That is Inf by default.   |
| N     | An even integer value indicating the number of subdivisions for applying Simpson's integration method.   |

**Details**

For some families of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., lb that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

**Value**

Matrices that represent the expected and observed Fisher information matrices.

**Author(s)**

Mahdi Teimouri

**References**

N. Balakrishnan and AHMED Hossain 2007. Inference for the Type II generalized logistic distribution under progressive Type II censoring, *Journal of Statistical Computation and Simulation*, 77(12), 1013-1031.

M. Teimouri and S. Nadarajah 2016. Bias corrected MLEs under progressive type-II censoring scheme, *Journal of Statistical Computation and Simulation*, 86 (14), 2714-2726.

**Examples**

```
n <- 20
R <- c(5, rep(0, n-6) )
param <- c("alpha", "beta")
mle <- c(2,6)
pdf <- quote( alpha/beta*(x/beta)^(alpha-1)*exp( -(x/beta)^alpha ) )
cdf <- quote( 1-exp( -(x/beta)^alpha ) )
lb <- 0
ub <- Inf
N <- 100
plan <- rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
fitype2(plan = plan, param = param, mle = mle, cdf = cdf, pdf = pdf, lb = lb, ub = ub, N = N)
```

---

goftype1

---

*Computing goodness-of-fit (GOF) measures under progressive type-I interval censoring scheme.*


---

**Description**

The goodness-of-fit (GOF) measures consist of Anderson-Darling (AD) and Cram er-von Misses (CVM) statistics for progressive type-I interval censoring scheme are given, respectively, by

$$AD = n \sum_{i=1}^m \gamma_i^2 \log \left[ \frac{A_{i+1}(1 - A_i)}{A_i(1 - A_{i+1})} \right] + 2n \sum_{i=1}^m \gamma_i \log \left( \frac{1 - A_{i+1}}{1 - A_i} \right) - n(A_{m+1} - A_1)$$

$$-n \log\left(\frac{1 - A_{m+1}}{1 - A_1}\right) + n(1 - A_{m+1} - \log A_{m+1}),$$

$$CVM = n \sum_{i=1}^m \gamma_i^2 (A_{i+1} - A_i) - n \sum_{i=1}^m \gamma_i (A_{i+1}^2 - A_i^2) + \frac{n}{3} (A_{m+1}^3 - A_1^3) + \frac{n}{3} (1 - A_{m+1})^3,$$

where  $R_0 = 0$ ,  $\gamma_i = (\sum_{j=1}^i X_j + \sum_{j=1}^{i-1} R_j)/n$ , and  $A_i = G(T_{i-1} | \hat{\Theta})$ , for  $i = 1, \dots, m$ .

### Usage

```
goftype1(plan, param, mle, cdf.expression = FALSE, pdf.expression = TRUE, cdf, pdf, lb = 0)
```

### Arguments

|                |  |
|----------------|--|
| plan           | Censoring plan for progressive type-I interval censoring scheme. It must be given as a data.frame that includes vector of upper bounds of the censoring times T, vector of number of failed subjects X, and vector of removed subjects in each interval R. |
| param          | Vector of the of the family parameter's names.   |
| mle            | Vector of the estimated parameters.  |
| cdf.expression | Logical. That is TRUE, if there is a closed form expression for the cumulative distribution function.  |
| pdf.expression | Logical. That is TRUE, if there is a closed form expression for the probability density function.  |
| cdf            | Expression of the cumulative distribution function.  |
| pdf            | Expression of the probability density function.  |
| lb             | Lower bound of the family support. That is zero by default.  |

### Details

We note that for lifetime distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., lb that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf. Theoretically, for lifetime distribution, we have  $lb = T_0 = 0$ .

### Value

A vector of goodness-of-fit measures consist of Anderson-Darling (AD) and Cramer-von Misses (CVM) statistics.

### Author(s)

Mahdi Teimouri

## References

M. Teimouri 2020. Bias corrected maximum likelihood estimators under progressive type-I interval censoring scheme, *Communications in Statistics-Simulation and Computation*, <https://doi.org/10.1080/03610918.2020.1819320>.

## Examples

```
data(plasma)
  n <- 20
param <- c("alpha", "beta")
mle <- c(0.4, 0.05)
cdf <- quote( 1-exp( beta*(1-exp( x^alpha )) ) )
pdf <- quote( exp( beta*(1-exp( x^alpha )) )*( beta*(exp( x^alpha ))*(x^(alpha-1)*alpha) ) )
lb <- 0
plan <- rtype1(n = n, P = plasma$P, T = plasma$upper, param = param, mle = mle, cdf.expression
  = FALSE, pdf.expression = TRUE, cdf = cdf, pdf = pdf, lb = lb)
goftype1(plan = plan, param = param, mle = mle, cdf.expression=TRUE, pdf.expression = FALSE, cdf =
  cdf, pdf = pdf, lb = lb)
```

---

goftype2

*Computing goodness-of-fit (GOF) measures under progressive type-II censoring scheme.*

---

## Description

The goodness-of-fit (GOF) measures consist of Anderson-Darling (AD), Cramer-von Misses (CVM), and log-likelihood statistics for progressive type-II censoring scheme.

## Usage

```
goftype2(plan, param, mle, cdf, pdf)
```

## Arguments

|       |   |
|-------|---|
| plan  | Censoring plan for progressive type-II censoring scheme. It must be given as a <code>data.frame</code> that includes number of failed items $X$ , and vector of removed items $R$ . |
| param | Vector of the of the family parameter's names.  |
| mle   | Vector of the maximum likelihood estimators.  |
| cdf   | Expression for the cumulative distribution function.  |
| pdf   | Expression for the probability density function.  |

## Author(s)

Mahdi Teimouri

## References

R. Pakyari and N. Balakrishnan 2012. A general purpose approximate goodness-of-fit for progressively Type-II censored data, *IEEE Transaction on Reliability*, 61, 238-244.

## Examples

```
n <- 20
R <- c(5, rep(0, n-6) )
param <- c("alpha", "beta")
mle <- c(2,6)
pdf <- quote( alpha/beta*(x/beta)^(alpha-1)*exp( -(x/beta)^alpha ) )
cdf <- quote( 1-exp( -(x/beta)^alpha ) )
lb <- 0
ub <- Inf
plan <- rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
goftype2(plan = plan, param = param, mle = mle, cdf = cdf, pdf = pdf)
```

---

|         |   |
|---------|---|
| mleype1 | <i>Computing the maximum likelihood estimator (MLE) for the parameters of the statistical model fitted to a progressive type-I interval censoring scheme.</i> |
|---------|---|

---

## Description

Computes the MLE of for the parameters of the model fitted to a progressive type-I interval censoring scheme with likelihood function

$$l(\Theta) = \log L(\Theta) \propto \sum_{i=1}^m X_i \log [F(t_i; \Theta) - F(t_{i-1}; \Theta)] + \sum_{i=1}^m R_i [1 - F(t_i; \Theta)],$$

in which  $F(\cdot; \Theta)$  is the family cumulative distribution function for  $\Theta = (\theta_1, \dots, \theta_k)^T$  provided that  $F(t_0; \Theta) = 0$ .

## Usage

```
mleype1(plan, param, start, cdf.expression = FALSE, pdf.expression = TRUE, cdf, pdf,
method = "Nelder-Mead", lb = 0, ub = Inf, level = 0.05)
```

## Arguments

|       |  |
|-------|--|
| plan  | Censoring plan for progressive type-I interval censoring scheme. It must be given as a data.frame that includes vector of upper bounds of the censoring times T, vector of number of failed items X, and vector of removed items in each interval R. |
| param | Vector of the of the family parameter's names.   |
| start | Vector of the initial values.  |

|                |  |
|----------------|--|
| cdf.expression | Logical. That is TRUE, if there is a closed form expression for the cumulative distribution function.                                  |
| pdf.expression | Logical. That is TRUE, if there is a closed form expression for the probability density function.                                      |
| cdf            | Expression of the cumulative distribution function.  |
| pdf            | Expression of the probability density function.  |
| method         | The method for the numerical optimization that includes one of CG, Nelder-Mead, BFGS, L-BFGS-B, SANN.                                  |
| lb             | Lower bound of the family's support. That is zero by default.  |
| ub             | Upper bound of the family's support. That is Inf by default.   |
| level          | Significance level for constructing asymptotic confidence interval That is 0.05 by default for constructing a 95% confidence interval. |

**Value**

MLE, standard error of MLE, and asymptotic confidence interval for MLE.

**Author(s)**

Mahdi Teimouri

**Examples**

```
data(plasma, package="bccp")
plan <- data.frame(T = plasma$upper, X = plasma$X, P = plasma$P, R = plasma$R)
param <- c("lambda", "beta")
mle <- c(1.4, 0.05)
pdf <- quote( lambda*(1-exp( -(x*beta)))^(lambda-1)*beta*exp( -(x*beta)) )
cdf <- quote( (1-exp( -(x*beta)))^lambda )
lb <- 0
ub <- Inf
level <- 0.05
mletype1(plan = plan, param = param, start = mle, cdf.expression = FALSE, pdf.expression = TRUE,
         cdf = cdf, pdf = pdf, method = "Nelder-Mead", lb = lb, ub = ub, level = level)
```

---

mle<sub>type2</sub>

*Computing the maximum likelihood estimator (MLE) for the parameters of the statistical model fitted to a progressive type-II censoring scheme.*

---

**Description**

Computes the MLE of for the parameters of the model fitted to a progressive type-II censoring scheme with likelihood function

$$l(\Theta) = \log L(\Theta) \propto C \sum_{i=1}^m \log f(x_{i:m:n}; \Theta) + \sum_{i=1}^m R_i \log [1 - F(x_{i:m:n}; \Theta)],$$

in which  $F(\cdot; \Theta)$  is the family cumulative distribution function for  $\Theta = (\theta_1, \dots, \theta_k)^T$  and  $r, s = 1, \dots, k$ , and  $C = n(n - R_1 - 1)(n - R_1 - R_2 - 2) \dots (n - R_1 - R_2 - \dots - R_{m-1} - m + 1)$ .

**Usage**

```
mletype2(plan, param, start, cdf, pdf, method = "Nelder-Mead", lb = 0, ub = Inf, N = 100,
          level = 0.05)
```

**Arguments**

|        |   |
|--------|---|
| plan   | Censoring plan for progressive type-II censoring scheme. It must be given as a data.frame including: number of items placed on the test at time zero and a vector that contains number R, of the removed alive items. |
| param  | Vector of the of the family parameter's names.  |
| start  | Vector of the initial values.   |
| pdf    | Expression of the probability density function.   |
| cdf    | Expression of the cumulative distribution function.   |
| method | The method for the numerically optimization that includes one of CG, Nelder-Mead, BFGS, L-BFGS-B, SANN.   |
| lb     | Lower bound of the family's support. That is zero by default.   |
| ub     | Upper bound of the family's support. That is Inf by default.  |
| N      | An even integer number indicating the number of subdivisions for applying Simpson's integration method.   |
| level  | Significance level for constructing asymptotic confidence interval That is 0.05 by default for constructing a 95% confidence interval.  |

**Value**

MLE, standard error of MLE, and asymptotic confidence interval for MLE.

**Author(s)**

Mahdi Teimouri

**References**

M. Teimouri and S. Nadarajah 2016. Bias corrected MLEs under progressive type-II censoring scheme, *Journal of Statistical Computation and Simulation*, 86 (14), 2714-2726.

**Examples**

```
n <- 10
R <- c(5, rep(0, n-6) )
param <- c("alpha", "beta")
mle <- c(2,6)
pdf <- quote( alpha/beta*(x/beta)^(alpha-1)*exp( -(x/beta)^alpha ) )
cdf <- quote( 1-exp( -(x/beta)^alpha ) )
lb <- 0
ub <- Inf
N <- 100
level <- 0.05
```



```
plan <- rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
mletype2(plan = plan, param = param, start = mle, cdf = cdf, pdf = pdf, method = "Nelder-Mead",
         lb = lb, ub = ub, N = N, level = level)
```

---

plasma

*Plasma survival data*

---

### Description

The plasma survival data contains the Survival times of plasma cell myeloma for 112 patients, see Carbone et al. (1967).

### Usage

```
data(plasma)
```

### Format

A text file with 4 columns.

### References

P. P. Carbone, L. E. Kellerhouse, and E. A. Gehan 1967. Plasmacytic myeloma: A study of the relationship of survival to various clinical manifestations and anomalous protein type in 112 patients. *The American Journal of Medicine*, 42 (6), 937-48.

### Examples

```
data(plasma)
```

---

relief

*Wingo's pain relief data*

---

### Description

In order to investigate the effectiveness of an anesthetic antibiotic ointment in relieving pain caused by wounds, a number of 30 patients take part in a test where 10 of them are removed from the test under censoring plan  $R = (5, 1, 4, 0)$  at times  $T_1 = 0.25, T_2 = 0.50, T_3 = 0.75$  and  $T_4 = 3.50$ . We note that the termination time is not fixed, but it is assumed that  $T_4 = 3.50$  sufficiently large to ensure that no observation is right censored, see Balakrishnan and Cramer (2014).

### Usage

```
data(relief)
```

**Format**

A text file with 20 observations.

**References**

N. Balakrishnan and E. Cramer 2014. *The Art of Progressive Censoring*. New York, Springer.

**Examples**

```
data(relief)
```

---

 rtype1
 

---



---

*Simulating a progressive type-I interval censoring scheme*


---

**Description**

Simulates a progressive type-I interval censoring scheme when censoring times and percentage of removed items in end of each interval are given in a `data.frame` structure.

**Usage**

```
rtype1(n, P, T, param, mle, cdf.expression = FALSE, pdf.expression = TRUE, cdf, pdf,
       lb = 0)
```

**Arguments**

|                |   |
|----------------|---|
| n              | Number of items placed on the test at time zero.  |
| P              | A vector of upper bounds of the censoring times T.  |
| T              | A vector that contains percentage P, of the removed alive items in each interval.                     |
| param          | A character vector that contains the name of family's parameters.                                     |
| mle            | A vector that contains of the maximum likelihood estimators of the parameters.                        |
| cdf.expression | Logical. That is TRUE, if there is a closed form expression for the cumulative distribution function. |
| pdf.expression | Logical. That is TRUE, if there is a closed form expression for the probability density function.     |
| cdf            | Expression of the cumulative distribution function.   |
| pdf            | Expression of the probability density function.   |
| lb             | Lower bound of the family's support. That is zero by default.   |

**Details**

For a family of distributions whose support is the positive semi-axis, i.e.,  $x > 0$ , the cumulative distribution function (cdf) may not be differentiable. In this case, the lower bound of the support of random variable, i.e., lb that is zero by default, must be chosen some positive small value to ensure the differentiability of the cdf.

**Value**

A data frame with four columns including censoring times, number of failed items in each interval, number of removed subjects in end of each interval, and percentage of removed items in end of each interval.

**Author(s)**

Mahdi Teimouri

**References**

R. Aggarwala 2001. Progressive interval censoring: some mathematical results with applications to inference. *Communications in Statistics-Theory and Methods*, 30(8&9), 1921–1935.

**Examples**

```
data(plasma)
n <- 112
param <- c("alpha", "beta")
mle <- c(0.4, 0.05)
cdf <- quote( 1-exp( beta*(1-exp( x^alpha )) ) )
pdf <- quote( exp( beta*(1-exp( x^alpha )) )*( beta*(exp( x^alpha ))*( x^(alpha-1)*alpha ) ) )
lb <- 0
rtype1(n = n, P = plasma$P, T = plasma$upper, param = param, mle = mle, cdf.expression = FALSE,
pdf.expression = TRUE, cdf = cdf, pdf = pdf, lb = lb)
```

---

rtype2

*Simulating a progressive type-II censoring scheme*


---

**Description**

Simulates a progressive type-II censoring scheme.

**Usage**

```
rtype2(n, R, param, mle, cdf, lb = 0, ub = Inf)
```

**Arguments**

|       |   |
|-------|---|
| n     | Number of items placed on the test at time zero.                |
| R     | A vector that contains number R, of the removed alive items.    |
| param | Character vector that contains the name of family's parameters. |
| mle   | The maximum likelihood estimators of the parameters.            |
| cdf   | Expression for the cumulative distribution function.            |
| lb    | Lower bound of the family's support. That is zero by default.   |
| ub    | Upper bound of the family's support. That is Inf by default.    |

**Value**

A data frame with two columns including observed lifetimes  $X$ , and number of censored items  $R$ .

**Author(s)**

Mahdi Teimouri

**References**

N. Balakrishnan and R. A. Sandhu 1995. A Simple Simulational Algorithm for Generating Progressive Type-II Censored Samples. *The American Statistician*, 49(2), 229-230.

**Examples**

```
n <- 20
R <- c(9, rep(0, 10) )
param <- c("alpha", "beta")
mle <- c(0.80, 12)
cdf <- quote( 1-exp( beta*(1-exp( x^alpha )) ) )
lb <- 0
ub <- Inf
rtype2(n = n, R = R, param = param, mle = mle, cdf = cdf, lb = lb, ub = ub)
```

---

simpson

---

*Computing integration numerically through the Simpson's method*


---

**Description**

Computes the integration for a real-valued function.

**Usage**

```
simpson(fun, lb, ub, N = 100)
```

**Arguments**

|     |  |
|-----|--|
| fun | Integrand expression   |
| lb  | Lower bound of integration.  |
| ub  | Upper bound of integration.  |
| N   | An even integer value indicating the number of subdivisions for applying Simpson's integration method. |

**Author(s)**

Mahdi Teimouri

**References**

E. Suli and D. Mayers 2003. *An Introduction to Numerical Analysis*, Cambridge University Press.

**Examples**

```
fun <- function(x) 1/sqrt( 2*pi*sigma^2 )*exp( -.5*(x-mu)^2/sigma^2 )
mu <- 0
sigma <- 1
lb <- 0
ub <- Inf
N <- 100
simpson(fun = fun, lb = lb, ub = ub, N = N)
```

---

welcome

*Starting message when loading bccp*

---

**Description**

It contains a welcome message for user of package bccp.

# Index

## \* datasets

plasma, [17](#)

relief, [17](#)

bootbctype1, [2](#)

bootbctype2, [3](#)

coxbctype1, [5](#)

coxbctype2, [7](#)

fitype1, [8](#)

fitype2, [10](#)

goftype1, [11](#)

goftype2, [13](#)

mletype1, [14](#)

mletype2, [15](#)

plasma, [17](#)

relief, [17](#)

rtype1, [18](#)

rtype2, [19](#)

simpson, [20](#)

welcome, [21](#)