Package ‘lmomco’

May 11, 2012

Type Package

Title L-moments, Censored L-moments, Trimmed L-moments, L-comoments,and Many Distributions

Version 1.6.1

Depends R (>= 2.10.0), utils

Suggests copBasic, Lmoments

Date 2012-05-11

Author William H. Asquith

Description The package implements the statistical theory of L-moments in R including L-moment estimation, probability-weighted moment estimation, parameter estimation for numerous familiar and not-so-familiar distributions, and L-moment estimation for the same distributions from the parameters. L-moments are derived from the expectations of order statistics and are linear with respect to the probability-weighted moments; choice of either can be made by mathematical convenience. L-moments are directly analogous to the well-known product moments; however, L-moments have many advantages including unbiasedness, robustness, and consistency with respect to the product moments. The method of L-moments can out perform the method of maximum likelihood. The lmomco package historically is oriented around canonical FORTRAN algorithms of J.R.M. Hosking, and the nomenclature for many of the functions parallels that of the Hosking library,which later became available in the lmom package. However, vast arrays of various extensions and curiosities are added by the author to aid and expand the breadth of L-moment application. Such extensions include venerable statistics as Sen weighted mean, Gini mean difference, plotting positions, and conditional probability adjustment. The plotting of L-moment ratio diagrams is directly supported in this package. Computations of L-moments for right-tail and left-tail censoring by known or unknown censoring threshold and also by indicator variable also are available. E.A.H. Elamir and A.H. Seheult have developed

the trimmed L-moments, which are implemented in this package,and numerical integration of quantile functions is used to
dynamically compute trajectories of select TL-moment ratios for
the construction of TL-moment ratio diagrams. Robert Serfling
and Peng Xiao have extended L-moments into multivariate space;
the so-called sample L-comoments are implemented here and might
have considerable application in copula theory because they
measure asymmetric correlation and higher co-moments or
co-movements of variables. The package supports the following
distributions with moment type shown as ‘‘L’’ (L-moments) or ‘‘TL’’(trimmed L-moments) and additional support for right-tail
censoring ([RC]) include: Asymmetric Exponential Power (L),Cauchy (TL), Exponential (L), Gamma (L), Generalized Extreme
Value (L), Generalized Lambda (L & TL), Generalized Logistic
(L), Generalized Normal (L), Generalized Pareto (L[RC] & TL),Gumbel (L), Kappa (L), Kumaraswamy (L), Laplace (L), Normal
(L), 3-parameter log-Normal (L), Pearson Type III (L), Rayleigh
(L), Reverse Gumbel (L[RC]), Rice/Rician (L), Truncated
Exponential (L), Wakeby (L), and Weibull (L).

Maintainer William H. Asquith <william.asquith@ttu.edu>

Repository CRAN

URL http://www.amazon.com/Distributional-Statistics-Environment-Statistical-Computing/dp/1463508417/

Date/Publication 2012-05-11 15:26:01

R topics documented:

`amarilloprecip` *Annual Maximum Precipitation Data for Amarillo, Texas*

Description

Annual maximum precipitation data for Amarillo, Texas

Usage

```
data(amarilloprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(amarilloprecip)
summary(amarilloprecip)
```

`Apwm2BpwmRC` *Conversion between A- and B-Type Probability-Weighted Moments for Right-Tail Censoring of An Appropriate Distribution*

Description

This function converts "A"-type Probability-Weighted Moments (PWMs, β_r^A) to the "B"-type β_r^B. The β_r^A are the ordinary PWMs for the m left noncensored or observed values. The β_r^B are more complex and use the m observed values and the $m - n$ right-tailed censored values for which the censoring threshold is known. These PWMs are described in the documenation for `pwmRC`.

This function uses the defined relation between to two PWM types when the β_r^A are known along with the parameters (`para`) of a right-tail censored distribution inclusive of the censoring fraction $\zeta = m/n$. The value ζ is the right-tail censor fraction or the probability $\mathrm{Pr}\{\}$ that x is less than the quantile at ζ nonexceedance probability: ($\mathrm{Pr}\{x < X(\zeta)\}$).

$$\beta_{r-1}^B = r^{-1}\{\zeta^r r\beta_{r-1}^A + (1 - \zeta^r)X(\zeta)\},$$

where $1 \leq r \leq n$ and n is the number of moments, and $X(\zeta)$ is the value of the quantile function at nonexceedance probability ζ. Finally, the `RC` in the function name is to denote `R`ight-tail `C`ensoring.

Usage

```
Apwm2BpwmRC(Apwm,para)
```

Arguments

`Apwm`	A vector of A-type PWMs: β_r^A
`para`	The parameters of the distribution from a function such as `pargpaRC` in which the β_r^A are contained in a `list` element titled `betas` and the right-tail censoring fraction ζ is contained in an element titled `zeta`.

Value

An R `list` is returned.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

Bpwm2ApwmRC and pwmRC

Examples

```
# Data listed in Hosking (1995, table 29.2, p. 551)
H <- c(3,4,5,6,6,7,8,8,9,9,9,10,10,11,11,11,13,13,13,13,13,
             17,19,19,25,29,33,42,42,51.9999,52,52,52)
      # 51.9999 was really 52, a real (noncensored) data point.
z <-  pwmRC(H,52)
# The B-type PMWs are used for the parameter estimation of the
# Reverse Gumbel distribution. The parameter estimator requires
# conversion of the PWMs to L-moments by pwm2lmom().
para <- parrevgum(pwm2lmom(z$Bbetas),z$zeta) # parameter object
Bbetas <- Apwm2BpwmRC(z$Abetas,para)
Abetas <- Bpwm2ApwmRC(Bbetas$betas,para)
# Assertion that both of the vectors of B-type PWMs should be the same.
str(Abetas) # A-type PWMs of the distribution
str(z$Abetas) # A-type PWMs of the original data
```

---

are.lmom.valid *Are the L-moments valid*

Description

The second through fifth order L-moments are perhaps the most common in analysis situations. These L-moments have particular constraints on magnitudes and relation to each other. This function evaluates and L-moment object whether the bounds for L-scale ($\lambda_2 > 0$), L-skew ($|\tau_3| < 1$), L-kurtosis ($0.25(5\tau_3^2 - 1) \leq \tau_4 < 1$), and $|\tau_5| < 1$ are satisfied.

Usage

```
are.lmom.valid(lmom)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.

Value

`TRUE`	L-moments are valid.
`FALSE`	L-moments are not valid.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105-124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmoms`, and `pwm2lmom`

Examples

```
lmr <- lmom.ub(rnorm(20))
if(are.lmom.valid(lmr)) print("They are.")
```

`are.par.valid` *Are the Distribution Parameters Consistent with the Distribution*

Description

This function is a dispatcher on top of the `are.parCCC.valid` functions, where `CCC` represents the distribution type: `aep4`, `cau`, `exp`, `gam`, `gev`, `glo`, `gno`, `gpa`, `gum`, `kap`, `kur`, `lap`, `ln3`, `nor`, `pe3`, `ray`, `revgum`, `rice`, `wak`, or `wei`. Internally, this function is called only by `vec2par` in the process of converting a vector into a proper distribution parameter object for this package.

Usage

```
are.par.valid(para,...)
```

Arguments

`para`	A distribution parameter object having at least attributes `type` and `para`.
`...`	Additional arguments for the `are.parCCC.valid` call that is made internally.

Value

`TRUE`	If the parameters are consistent with the distribution specified by the `type` attribute of the parameter object.
`FALSE`	If the parameters are not consistent with the distribution specified by the `type` attribute of the parameter object.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

vec2par

Examples

```
vec  <- c(12,120)           # parameters of exponential distribution
para <- vec2par(vec,'exp')  # build exponential distribution parameter
                            # object
# The following two conditionals are equivalent as are.parexp.valid()
# is called within are.par.valid().
if(are.par.valid(para))    Q <- quaexp(0.5,para)
if(are.parexp.valid(para)) Q <- quaexp(0.5,para)
```

`are.paraep4.valid` *Are the Distribution Parameters Consistent with the 4-p Asymmetric Exponential Power Distribution*

Description

The distribution parameter object returned by functions of this package such as by `paraep4` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfaep4`, `quaaep4`, and `lmomaep4` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.paraep4.valid` function.

Usage

```
are.paraep4.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `paraep4`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are aep4 consistent.
FALSE	If the parameters are not aep4 consistent.

Note

This function calls `is.aep4` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Ayebo, A., and Kozubowski, T.J., 2003, An asymmetric generalization of Gaussian and Laplace laws: Journal of Probability and Statistical Science, v. 1, no. 2, pp. 187-210.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

See Also

`is.aep4`

Examples

```
para <- vec2par(c(0,1, 0.5, 4), type="aep4")
if(are.paraep4.valid(para)) Q <- quaaep4(0.5,para)
```

`are.parcau.valid` *Are the Distribution Parameters Consistent with the Cauchy Distribution*

Description

The distribution parameter object returned by functions of this package such as by `vec2par` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfcau` and `quacau`) require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively.

Usage

```
are.parcau.valid(para,nowarn=FALSE,...)
```

Arguments

para	A distribution parameter list returned by parcau or vec2par.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.
...	Additional arguments for the are.parCCC.valid call that is made internally.

Value

TRUE	If the parameters are cau consistent.
FALSE	If the parameters are not cau consistent.

Note

This function calls is.cau to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Gilchrist, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

is.cau

Examples

```
para <- vec2par(c(12,12),type='cau')
if(are.parcau.valid(para)) Q <- quacau(0.5,para)
```

are.parexp.valid *Are the Distribution Parameters Consistent with the Exponential Distribution*

Description

The distribution parameter object returned by functions of this package such as by parexp are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (cdfexp, quaexp, and lmomexp require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the are.parexp.valid function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parexp.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by parexp.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are exp consistent.
FALSE	If the parameters are not exp consistent.

Note

This function calls is.exp to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.exp

Examples

```
para <- parexp(lmom.ub(c(123,34,4,654,37,78)))
if(are.parexp.valid(para)) Q <- quaexp(0.5,para)
```

---

are.pargam.valid — *Are the Distribution Parameters Consistent with the Gamma Distribution*

Description

The distribution parameter object returned by functions of this package such as by pargam are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (cdfgam, quagam, and lmomgam require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the are.pargam.valid function. The FORTRAN source code of Hosking provides the basis for the function. The parameters are restricted to the following conditions.

$$\alpha > 0 \text{ and } \beta > 0.$$

Usage

```
are.pargam.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by pargam.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are gam consistent.
FALSE	If the parameters are not gam consistent.

Note

This function calls is.gam to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.gam

Examples

```
para <- pargam(lmom.ub(c(123,34,4,654,37,78)))
if(are.pargam.valid(para)) Q <- quagam(0.5,para)
```

`are.pargev.valid` *Are the Distribution Parameters Consistent with the Generalized Extreme Value Distribution*

Description

The distribution parameter object returned by functions of this package such as by `pargev` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfgev`, `quagev`, and `lmomgev` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.pargev.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.pargev.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `pargev`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are gev consistent.
FALSE	If the parameters are not gev consistent.

Note

This function calls `is.gev` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`is.gev`

Examples

```
para <- pargev(lmom.ub(c(123,34,4,654,37,78)))
if(are.pargev.valid(para)) Q <- quagev(0.5,para)
```

`are.pargld.valid` *Are the Distribution Parameters Consistent with the Generalized Lambda Distribution*

Description

The distribution parameter object returned by functions of this package such as by `vec2par` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`quagld`) require consistent parameters to ensure that the Generalized Lambda Distribution is monotonic increasing on $0 \leq F \leq 1$, in which F is nonexceedance probability.

Usage

```
are.pargld.valid(para,verbose=FALSE,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `vec2par`.
`verbose`	A logical switch on additional output to the user—default is `FALSE`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Details

Karian and Dudewicz (2000) outline valid parameter space of the Generalized Lambda distribution. First, according to Theorem 1.3.3 the distribution is valid if and only if

$$\alpha(\kappa F^{\kappa-1} + h(1-F)^{h-1}) \geq 0.$$

for all $F \in [0,1]$. The `are.pargld.valid` function tests against this condition by incrementing through $[0,1]$ by $dF = 0.0001$. This is a brute force method of course. Further, Karian and Dudewicz (2002) provide a diagrammatic representation of regions in κ and h space for suitable α in which the distribution is valid. The `are.pargld.valid` function subsequently checks against the 6 valid regions as a secondary check on Theorem 1.3.3. The regions of the distribution are defined for suitably choosen α by

$$\text{Region 1: } \kappa \leq -1 \text{ and } h \geq 1,$$

$$\text{Region 2: } \kappa \geq 1 \text{ and } h \leq -1,$$

$$\text{Region 3: } \kappa \geq 0 \text{ and } h \geq 0,$$

$$\text{Region 4: } \kappa \leq 0 \text{ and } h \leq 0,$$

$$\text{Region 5: } h \geq (-1/\kappa) \text{ and } -1 \geq \kappa \leq 0, \text{ and}$$

$$\text{Region 6: } h \leq (-1/\kappa) \text{ and } h \geq -1 \text{ and } \kappa \geq 1.$$

Value

`TRUE`	If the parameters are `gld` consistent.
`FALSE`	If the parameters are not `gld` consistent.

Note

This function calls `is.gld` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

is.gld

Examples

```
para <- vec2par(c(123,34,4,3),type='gld')
if(are.pargld.valid(para)) Q <- quagld(0.5,para)
```

---

are.parglo.valid — *Are the Distribution Parameters Consistent with the Generalized Logistic Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parglo` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfglo`, `quaglo`, and `lmomglo` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parglo.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parglo.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parglo`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `glo` consistent.
`FALSE`	If the parameters are not `glo` consistent.

Note

This function calls `is.glo` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.glo

Examples

```
para <- parglo(lmom.ub(c(123,34,4,654,37,78)))
if(are.parglo.valid(para)) Q <- quaglo(0.5,para)
```

---

are.pargno.valid *Are the Distribution Parameters Consistent with the Generalized Normal Distribution*

Description

The distribution parameter object returned by functions of this package such as by `pargno` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfgno`, `quagno`, and `lmomgno` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.pargno.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.pargno.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `pargno`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are gno consistent.
`FALSE`	If the parameters are not gno consistent.

Note

This function calls `is.gno` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`is.gno`

Examples

```
para <- pargno(lmom.ub(c(123,34,4,654,37,78)))
if(are.pargno.valid(para)) Q <- quagno(0.5,para)
```

`are.pargpa.valid` *Are the Distribution Parameters Consistent with the Generalized Pareto Distribution*

Description

The distribution parameter object returned by functions of this package such as by `pargpa` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfgpa`, `quagpa`, and `lmomgpa` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.pargpa.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.pargpa.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by `pargpa`.
nowarn	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are gpa consistent.
FALSE	If the parameters are not gpa consistent.

Note

This function calls `is.gpa` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`is.gpa`

Examples

```
para <- pargpa(lmom.ub(c(123,34,4,654,37,78)))
if(are.pargpa.valid(para)) Q <- quagpa(0.5,para)
```

are.pargum.valid *Are the Distribution Parameters Consistent with the Gumbel Distribution*

Description

The distribution parameter object returned by functions of this package such as by `pargum` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfgum`, `quagum`, and `lmomgum` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.pargum.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.pargum.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `pargum`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `gum` consistent.
`FALSE`	If the parameters are not `gum` consistent.

Note

This function calls `is.gum` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.gum

Examples

```
para <- pargum(lmom.ub(c(123,34,4,654,37,78)))
if(are.pargum.valid(para)) Q <- quagum(0.5,para)
```

---

are.parkap.valid — *Are the Distribution Parameters Consistent with the Kappa Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parkap` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfkap`, `quakap`, and `lmomkap` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parkap.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parkap.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parkap`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `kap` consistent.
`FALSE`	If the parameters are not `kap` consistent.

Note

This function calls `is.kap` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.kap

Examples

```
para <- parkap(lmom.ub(c(123,34,4,654,37,78)))
if(are.parkap.valid(para)) Q <- quakap(0.5,para)
```

---

are.parkur.valid — *Are the Distribution Parameters Consistent with the Kumaraswamy Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parkur` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfkur`, `quakur`, and `lmomkur` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parkur.valid` function.

Usage

```
are.parkur.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parkur`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `kur` consistent.
`FALSE`	If the parameters are not `kur` consistent.

Note

This function calls `is.kur` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

`is.kur`

Examples

```
para <- parkur(lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9)))
if(are.parkur.valid(para)) Q <- quakur(0.5,para)
```

`are.parlap.valid` *Are the Distribution Parameters Consistent with the Laplace Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parlap` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdflap`, `qualap`, and `lmomlap` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parlap.valid` function.

$$\alpha > 0.$$

Usage

```
are.parlap.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parlap`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are `lap` consistent.
FALSE	If the parameters are not `lap` consistent.

Note

This function calls `is.lap` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

`is.lap`

Examples

```
para <- parlap(lmom.ub(c(123,34,4,654,37,78)))
if(are.parlap.valid(para)) Q <- qualap(0.5,para)
```

---

are.parln3.valid *Are the Distribution Parameters Consistent with the 3-Parameter Log-Normal Distribution*

Description

The distribution parameter object returned by functions of this package such as by `pargum` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfln3`, `qualn3`, and `lmomln3` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parln3.valid` function.

Usage

```
are.parln3.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parln3`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `ln3` consistent.
`FALSE`	If the parameters are not `ln3` consistent.

Note

This function calls `is.ln3` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

NEED

See Also

`is.ln3`

Examples

```
para <- parln3(lmom.ub(c(123,34,4,654,37,78)))
if(are.parln3.valid(para)) Q <- qualn3(0.5,para)
```

`are.parnor.valid` *Are the Distribution Parameters Consistent with the Normal Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parnor` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfnor`, `quanor`, and `lmomnor` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parnor.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parnor.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by parnor.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are nor consistent.
FALSE	If the parameters are not nor consistent.

Note

This function calls is.nor to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.nor

Examples

```
para <- parnor(lmom.ub(c(123,34,4,654,37,78)))
if(are.parnor.valid(para)) Q <- quanor(0.5,para)
```

are.parpe3.valid *Are the Distribution Parameters Consistent with the Pearson Type III Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parpe3` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfpe3`, `quape3`, and `lmompe3` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parpe3.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parpe3.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parpe3`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `pe3` consistent.
`FALSE`	If the parameters are not `pe3` consistent.

Note

This function calls `is.pe3` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.pe3

Examples

```
para <- parpe3(lmom.ub(c(123,34,4,654,37,78)))
if(are.parpe3.valid(para)) Q <- quape3(0.5,para)
```

---

are.parray.valid *Are the Distribution Parameters Consistent with the Rayleigh Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parray` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfray`, `quaray`, and `lmomray` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parray.valid` function.

Usage

```
are.parray.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parray`.
`nowarn`	A logical switch on warning suppression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `ray` consistent.
`FALSE`	If the parameters are not `ray` consistent.

Note

This function calls `is.ray` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

is.ray

Examples

```
para <- parray(lmom.ub(c(123,34,4,654,37,78)))
if(are.parray.valid(para)) Q <- quaray(0.5,para)
```

---

are.parrevgum.valid *Are the Distribution Parameters Consistent with the Reverse Gumbel Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parrevgum` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfrevgum`, `quarevgum`, and `lmomrevgum` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parrevgum.valid` function.

Usage

```
are.parrevgum.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parrevgum`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `revgum` consistent.
`FALSE`	If the parameters are not `revgum` consistent.

Note

This function calls `is.revgum` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

is.revgum

Examples

```
para <- vec2par(c(.9252, .1636, .7),type='revgum')
if(are.parrevgum.valid(para)) Q <- quarevgum(0.5,para)
```

---

are.parrice.valid *Are the Distribution Parameters Consistent with the Rice Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parrice` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfrice`, `quarice`, and `lmomrice` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parrice.valid` function.

Usage

```
are.parrice.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parrice`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `rice` consistent.
`FALSE`	If the parameters are not `rice` consistent.

Note

This function calls `is.rice` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

See Also

`is.rice`

Examples

```
#para <- parrice(lmom.ub(c(123,34,4,654,37,78)))
#if(are.parrice.valid(para)) Q <- quarice(0.5,para)
```

`are.partexp.valid` *Are the Distribution Parameters Consistent with the Truncated Exponential Distribution*

Description

The distribution parameter object returned by functions of this package such as by `partexp` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdftexp`, `quatexp`, and `lmomtexp` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.partexp.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.partexp.valid(para,nowarn=FALSE)
```

Arguments

`para`	A distribution parameter list returned by `parexp`.
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

`TRUE`	If the parameters are `texp` consistent.
`FALSE`	If the parameters are not `texp` consistent.

Note

This function calls is.texp to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.texp

Examples

```
para <- parexp(lmom.ub(c(123,34,4,654,37,78)))
if(are.parexp.valid(para)) Q <- quaexp(0.5,para)
```

---

are.parwak.valid *Are the Distribution Parameters Consistent with the Wakeby Distribution*

Description

The distribution parameter object returned by functions of this package such as by parwak are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (cdfwak, quawak, and lmomwak require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the are.parwak.valid function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parwak.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by parwak.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are wak consistent.
FALSE	If the parameters are not wak consistent.

Note

This function calls `is.wak` to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`is.wak`

Examples

```
para <- parwak(lmom.ub(c(123,34,4,654,37,78)))
if(are.parwak.valid(para)) Q <- quawak(0.5,para)
```

---

are.parwei.valid   *Are the Distribution Parameters Consistent with the Weibull Distribution*

Description

The distribution parameter object returned by functions of this package such as by `parwei` are consistent with the corresponding distribution, otherwise a list would not have been returned. However, other functions (`cdfwei`, `quawei`, and `lmomwei` require consistent parameters to return the cumulative probability (nonexceedance), quantile, and L-moments of the distribution, respectively. These functions internally use the `are.parwei.valid` function. The FORTRAN source code of Hosking provides the basis for the function.

Usage

```
are.parwei.valid(para,nowarn=FALSE)
```

Arguments

para	A distribution parameter list returned by parwei.
nowarn	A logical switch on warning surpression. If TRUE then options(warn=-1) is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

Value

TRUE	If the parameters are wei consistent.
FALSE	If the parameters are not wei consistent.

Note

This function calls is.wei to verify consistency between the distribution parameter object and the intent of the user.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

is.wei

Examples

```
para <- parwei(lmom.ub(c(123,34,4,654,37,78)))
if(are.parwei.valid(para)) Q <- quawei(0.5,para)
```

Bpwm2ApwmRC *Conversion between B- and A-Type Probability-Weighted Moments for Right-Tail Censoring of An Appropriate Distribution*

Description

This function converts “B”-type Probability-Weighted Moments (PWMs, β_r^B) to the “A”-type β_r^A. The β_r^A are the ordinary PWMs for the m left noncensored or observed values. The β_r^B are more complex and use the m observed values and the $m - n$ right-tailed censored values for which the censoring threshold is known. These PWMs are described in the documentation for `pwmRC`.

This function uses the defined relation between to two PWM types when the β_r^B are known along with the parameters (`para`) of a right-tail censored distribution inclusive of the censoring fraction $\zeta = m/n$. The value ζ is the right-tail censor fraction or the probability $\mathrm{Pr}\{\}$ that x is less than the quantile at ζ nonexceedance probability: $(\mathrm{Pr}\{x < X(\zeta)\})$.

$$\beta_{r-1}^A = \frac{r\beta_{r-1}^B - (1-\zeta^r)X(\zeta)}{r\zeta^r},$$

where $1 \le r \le n$ and n is the number of moments, and $X(\zeta)$ is the value of the quantile function at nonexceedance probability ζ. Finally, the `RC` in the function name is to denote `R`ight-tail `C`ensoring.

Usage

```
Bpwm2ApwmRC(Bpwm,para)
```

Arguments

`Bpwm`	A vector of B-type PWMs: β_r^B
`para`	The parameters of the distribution from a function such as `pargpaRC` in which the β_r^B are contained in a `list` element titled `betas` and the right-tail censoring fraction ζ is contained in an element titled `zeta`.

Value

An R `list` is returned.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

`Apwm2BpwmRC` and `pwmRC`

Examples

```
# Data listed in Hosking (1995, table 29.2, p. 551)
H <- c(3,4,5,6,6,7,8,8,9,9,9,10,10,11,11,11,13,13,13,13,13,
             17,19,19,25,29,33,42,42,51.9999,52,52,52)
      # 51.9999 was really 52, a real (noncensored) data point.
z <-  pwmRC(H,52)
# The B-type PMWs are used for the parameter estimation of the
# Reverse Gumbel distribution. The parameter estimator requires
# conversion of the PWMs to L-moments by pwm2lmom().
para <- parrevgum(pwm2lmom(z$Bbetas),z$zeta) # parameter object
Abetas <- Bpwm2ApwmRC(z$Bbetas,para)
Bbetas <- Apwm2BpwmRC(Abetas$betas,para)
# Assertion that both of the vectors of B-type PWMs should be the same.
str(Bbetas) # B-type PWMs of the distribution
str(z$Bbetas) # B-type PWMs of the original data
```

---

canyonprecip *Annual Maximum Precipitation Data for Canyon, Texas*

Description

Annual maximum precipitation data for Canyon, Texas

Usage

```
data(canyonprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(canyonprecip)
summary(canyonprecip)
```

`cdfaep4` *Cumulative Distribution Function of the 4-p Asymmetric Exponential Power Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the 4-parameter Asymmetric Exponential Power distribution given parameters (ξ, α, κ, and h) of the distribution computed by `paraep4`. The cumulative distribution function of the distribution is

For $x < \xi$,

$$F(x) = \frac{\kappa^2}{(1+\kappa^2)}\,\gamma([(\xi - x)/(\alpha\kappa)]^h,\ 1/h)$$

and for $x \geq \xi$,

$$F(x) = 1 - \frac{1}{(1+\kappa^2)}\,\gamma([\kappa(x - \xi)/\alpha]^h,\ 1/h)$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, κ is a shape parameter, h is another shape parameter, $\gamma(Z, shape)$ is the upper tail of the incomplete gamma function. The upper tail of the incomplete gamma function is `pgamma(Z, shape, lower.tail=FALSE)` in R and mathematically is

$$\gamma(Z, a) = \int_Z^\infty y^{a-1}\exp(-y)dy\ /\Gamma(a).$$

Usage

```
cdfaep4(x, para, paracheck=TRUE)
```

Arguments

`x`	A real value.
`para`	The parameters from `paraep4` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Ayebo, A., and Kozubowski, T.J., 2003, An asymmetric generalization of Gaussian and Laplace laws: Journal of Probability and Statistical Science, v. 1, no. 2, pp. 187-210.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

See Also

cdfaep4, quaaep4, paraep4

Examples

```
x <- -0.1
para <- vec2par(c(0, 100, 0.5, 4), type="aep4")
F <- cdfaep4(-.1,para)
estx <- quaaep4(F, para)
cat(c("F=",F,"  and estx=",estx,"\n"))

## Not run:
delx <- .1
x <- seq(-20,20, by=delx);
K <- 1;

PAR <- list(para=c(0,1, K, 0.5), type="aep4");
plot(x,cdfaep4(x, PAR), type="n",
     ylab="NONEXCEEDANCE PROBABILITY",
     ylim=c(0,1), xlim=range(x));
lines(x,cdfaep4(x,PAR), lwd=4);
lines(quaaep4(cdfaep4(x,PAR),PAR), cdfaep4(x,PAR), col=2)

PAR <- list(para=c(0,1, K, 1), type="aep4");
lines(x,cdfaep4(x, PAR), lty=2, lwd=4);
lines(quaaep4(cdfaep4(x,PAR),PAR), cdfaep4(x,PAR), col=2)

PAR <- list(para=c(0,1, K, 2), type="aep4");
lines(x,cdfaep4(x, PAR), lty=3, lwd=4);
lines(quaaep4(cdfaep4(x,PAR),PAR), cdfaep4(x,PAR), col=2)

PAR <- list(para=c(0,1, K, 4), type="aep4");
lines(x,cdfaep4(x, PAR), lty=4, lwd=4);
lines(quaaep4(cdfaep4(x,PAR),PAR), cdfaep4(x,PAR), col=2)

## End(Not run)
```

cdfcau *Cumulative Distribution Function of the Cauchy Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Cauchy distribution given parameters (ξ and α) of the distribution provided by parcau or vec2par. The cumulative distribution function of the distribution is

$$F(x) = \frac{\arctan(\frac{x-\xi}{\alpha})}{\pi} + 0.5,$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter and α is a scale parameter.

Usage

```
cdfcau(x, para)
```

Arguments

x	A real value.
para	The parameters from parcau or vec2par.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Gilchrist, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

quacau, parcau, vec2par

Examples

```
para <- c(12,12)
cdfcau(50,vec2par(para,type='cau'))
```

`cdfexp` *Cumulative Distribution Function of the Exponential Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Exponential distribution given parameters (ξ and α) of the distribution computed by `parexp`. The cumulative distribution function of the distribution is

$$F(x) = 1 - e^{\left(\frac{-(x-\xi)}{\alpha}\right)}$$

where $F(x)$ is the nonexceedance probability for the quantile x, ξ is a location parameter and α is a scale parameter.

Usage

```
cdfexp(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parexp` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`quaexp`, `parexp`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfexp(50,parexp(lmr))
```

cdfgam *Cumulative Distribution Function of the Gamma Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Gamma distribution given parameters (α and β) of the distribution computed by pargam. The cumulative distribution function of the distribution has no explicit form, but is expressed as an integral.

$$F(x) = \frac{\beta^{-\alpha}}{\Gamma(\alpha)} \int_0^x t^{\alpha-1} e^{-t/\beta} \mathrm{d}F,$$

where $F(x)$ is the nonexceedance probability for the quantile x. The parameters have the following interpretation in the **R** syntax; α is a shape parameter and β is a scale parameter.

Usage

```
cdfgam(x, para)
```

Arguments

x	A real value.
para	The parameters from pargam or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quagam, pargam

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfgam(50,pargam(lmr))

# A manual demonstration of a gamma parent
G  <- vec2par(c(0.6333,1.579),type='gam') # the parent
F1 <- 0.25         # nonexceedance probability
x  <- quagam(F1,G) # the lower quartile (F=0.25)
a  <- 0.6333       # gamma parameter
b  <- 1.579        # gamma parameter
# compute the integral
xf <- function(t,A,B) { t^(A-1)*exp(-t/B) }
Q  <- integrate(xf,0,x,A=a,B=b)
# finish the math
F2 <- Q$val*b^(-a)/gamma(a)
# check the result
if(abs(F1-F2) < 1e-8) print("yes")
```

---

cdfgev — *Cumulative Distribution Function of the Generalized Extreme Value Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Generalized Extreme Value distribution given parameters (ξ, α, and κ) of the distribution computed by pargev. The cumulative distribution function of the distribution is

$$F(x) = e^{-e^{-y}},$$

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0, \text{ and}$$

$$y = (x-\xi)/\alpha \text{ for } \kappa = 0,$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
cdfgev(x, para)
```

Arguments

x	A real value.
para	The parameters from pargev or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quagev, pargev

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfgev(50,pargev(lmr))
```

---

cdfgld — *Cumulative Distribution Function of the Generalized Lambda Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Generalized Lambda distribution given parameters (ξ, α, κ, and h) of the distribution computed by pargld or similar. The cumulative distribution function of the distribution has no explicit form. The R function `uniroot` is used to root the quantile function quagld to compute the nonexceedance probability. The function returns 0 or 1 if the x argument is at or beyond the limits of the distribution as specified by the parameters.

Usage

```
cdfgld(x, gldpara, paracheck)
```

Arguments

x	A real value.
gldpara	The parameters from pargld or similar.
paracheck	A logical switch as to whether the validity of the parameters should be checked. Default is paracheck=TRUE. This switch is made so that the root solution needed for cdfgld exhibits an extreme speed increase because of the repeated calls to quagld.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

quagld, lmomgld, pargld

Examples

```
P <- vec2par(c(123,340,0.4,0.654),type='gld')
cdfgld(300,P)
```

---

cdfglo — *Cumulative Distribution Function of the Generalized Logistic Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Generalized Logistic distribution given parameters (ξ, α, and κ) of the distribution computed by parglo. The cumulative distribution function of the distribution is

$$F(x) = 1/(1 + e^{-y}),$$

where y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x - \xi)}{\alpha}\right) \text{ for } \kappa \neq 0\text{,and}$$

$$y = (x - \xi)/\alpha \text{ for } \kappa = 0, \text{ and}$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
cdfglo(x, para)
```

Arguments

x	A real value.
para	The parameters from parglo or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quaglo, parglo

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfglo(50,parglo(lmr))
```

`cdfgno` *Cumulative Distribution Function of the Generalized Normal Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Generalized Normal distribution given parameters (ξ, α, and κ) of the distribution computed by `pargno`. The cumulative distribution function of the distribution is

$$F(x) = \Phi(y),$$

where Φ is the cumulative ditribution function of the standard normal distribution and y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0 \text{, and}$$

$$y = (x-\xi)/\alpha \text{ for } \kappa = 0,$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
cdfgno(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `pargno` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quagno, pargno

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfgno(50,pargno(lmr))
```

---

cdfgpa — *Cumulative Distribution Function of the Generalized Pareto Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Generalized Pareto distribution given parameters (ξ, α, and κ) of the distribution computed by pargpa. The cumulative distribution function of the distribution is

$$F(x) = 1 - e^{-y},$$

where y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0, \text{ and}$$

$$y = (x-\xi)/A \text{ for } \kappa = 0,$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
cdfgpa(x, para)
```

Arguments

x	A real value.
para	The parameters from pargpa or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quagpa, pargpa

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfgpa(50,pargpa(lmr))
```

---

cdfgum — *Cumulative Distribution Function of the Gumbel Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Gumbel distribution given parameters (ξ and α) of the distribution computed by pargum. The cumulative distribution function of the distribution is

$$F(x) = e^{-e^{\left(-\frac{(x-\xi)}{\alpha}\right)}},$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, and α is a scale parameter.

Usage

```
cdfgum(x, para)
```

Arguments

x	A real value.
para	The parameters from pargum or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quagum, pargum

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfgum(50,pargum(lmr))
```

---

cdfkap — *Cumulative Distribution Function of the Kappa Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Kappa distribution given parameters (ξ, α, and κ, h) of the distribution computed by parkap. The cumulative distribution function of the distribution is

$$F(x) = \left(1 - h\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right)^{1/\kappa}\right)^{1/h},$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, α is a scale parameter, κ is a shape parameter, and h is another shape parameter.

Usage

```
cdfkap(x, para)
```

Arguments

x	A real value.
para	The parameters from parkap or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quakap, parkap

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78,21,32,231,23))
cdfkap(50,parkap(lmr))
```

`cdfkur` *Cumulative Distribution Function of the Kumaraswamy Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Kumaraswamy distribution given parameters (α and β) of the distribution computed by parkur. The cumulative distribution function of the distribution is

$$F(x) = 1 - (1 - x^{\alpha})^{\beta},$$

where $F(x)$ is the nonexceedance probability for quantile x, α is a shape parameter, and β is a shape parameter.

Usage

```
cdfkur(x, para)
```

Arguments

x	A real value.
para	The parameters from parkur or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

quakur, parkur

Examples

```
lmr <- lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9))
cdfkur(0.5,parkur(lmr))
```

---

cdflap *Cumulative Distribution Function of the Laplace Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Laplace distribution given parameters (ξ and α) of the distribution computed by parlap. The cumulative distribution function of the distribution is

$$F(x) = \frac{1}{2}e^{(x-\xi)/\alpha} \text{ for } x \le \xi \text{, and}$$

$$F(x) = 1 - \frac{1}{2}e^{-(x-\xi)/\alpha} \text{ for } x > \xi \text{,}$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter and α is a scale parameter.

Usage

```
cdflap(x, para)
```

Arguments

x	A real value.
para	The parameters from parlap or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

qualap, parlap

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdflap(50,parlap(lmr))
```

---

cdfln3 *Cumulative Distribution Function of the 3-Parameter Log-Normal Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Log-Normal3 distribution given parameters (ζ, μ, and σ) of the distribution computed by parln3. The cumulative distribution function of the distribution is

$$F(x) = \Phi(y),$$

where Φ is the cumulative ditribution function of the standard normal distribution and y is

Usage

```
cdfln3(x, para)
```

Arguments

x	A real value.
para	The parameters from parln3 or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

NEED

See Also

qualn3, parln3

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfln3(50,parln3(lmr))
```

---

cdfnor *Cumulative Distribution Function of the Normal Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Normal distribution given parameters of the distribution computed by parnor. The cumulative distribution function of the distribution is

$$F(x) = \Phi((x - \mu)/\sigma),$$

where $F(x)$ is the nonexceedance probability for quantile x, μ is the arithmetic mean, and σ is the standard deviation, and Φ is the cumulative distribution function of the standard normal distribution. The R-function `pnorm` is used.

Usage

```
cdfnor(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from parnor or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, pp. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quanor, parnor

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfnor(50,parnor(lmr))
```

`cdfpe3` *Cumulative Distribution Function of the Pearson Type III Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Pearson Type III distribution given parameters (μ, σ, and γ) of the distribution computed by parpe3. These parameters are equal to the product moments: mean, standard deviation, and skew (see pmoms). The cumulative distribution function of the distribution for $\gamma \neq 0$ is

$$F(x) = \frac{G\left(\alpha, \frac{Y}{\beta}\right)}{\Gamma(\alpha)},$$

where $F(x)$ is the nonexceedance probability for quantile x, G is defined below and is related to the incomplete gamma function of R (`pgamma()`), Γ is the complete gamma function, ξ is a location parameter, β is a scale parameter, α is a shape parameter, and $Y = x - \xi$ if $\gamma > 0$ and $Y = \xi - x$ if $\gamma < 0$. These three "new" parameters are related to the product moments by

$$\alpha = 4/\gamma^2,$$

$$\beta = \frac{1}{2}\sigma|\gamma|,$$

$$\xi = \mu - 2\sigma/\gamma.$$

The function $G(\alpha, x)$ is

$$G(\alpha, x) = \int_0^x t^{(a-1)} \mathrm{e}^{-t} \mathrm{d}t.$$

If $\gamma = 0$, the distribution is symmetrical and simply is the normal distribution with mean and standard deviation of μ and σ, respectively. Internally, the $\gamma = 0$ condition is implemented by `pnorm()`. If $\gamma > 0$, the distribution is right-tail heavy, and $F(x)$ is the returned nonexceedance probability. On the other hand if $\gamma < 0$, the distribution is left-tail heavy and $1 - F(x)$ is the actual nonexceedance probability that is returned.

Usage

```
cdfpe3(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parpe3` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`quape3`, `parpe3`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfpe3(50,parpe3(lmr))
```

`cdfray` *Cumulative Distribution Function of the Rayleigh Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Rayleigh distribution given parameters (ξ and α) of the distribution computed by `parray`. The cumulative distribution function of the distribution is

$$F(x) = 1 - e^{-(x-\xi)^2/(2\alpha^2)},$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, and α is a scale parameter.

Usage

```
cdfray(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parray` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

`quaray`, `parray`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfray(50,parray(lmr))
```

`cdfrevgum` *Cumulative Distribution Function of the Reverse Gumbel Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Reverse Gumbel distribution given parameters (ξ and α) of the distribution computed by `parrevgum`. The cumulative distribution function of the distribution is

$$F(x) = 1 - e^{-e^{\left(-\frac{(x-\xi)}{\alpha}\right)}},$$

where $F(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, and α is a scale parameter. Notice that the function has some sign differences and uses the complement of F compared to the cumulative distribution function of the Gumbel distribution in `cdfgum`.

Usage

```
cdfrevgum(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parrevgum` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

`quarevgum`, `parrevgum`

Examples

```
# See p. 553 of Hosking (1995)
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221,  0.245,  0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,threshold=.2960001)
str(z)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -0.516 and lamB2 = 0.523
btypelmoms <- pwm2lmom(z$Bbetas)
# My version of R reports lamB1 = -0.5162 and lamB2 = 0.5218
str(btypelmoms)
rg.pars <- parrevgum(btypelmoms,z$zeta)
str(rg.pars)
# Hosking reports xi=0.1636 and alpha=0.9252 for the sample
# My version of R reports xi = 0.1635 and alpha = 0.9254
F  <- nonexceeds()
PP <- pp(D) # plotting positions of the data
D  <- sort(D)
plot(D,PP)
lines(D,cdfrevgum(D,rg.pars))
```

---

cdfrice *Cumulative Distribution Function of the Rice Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Rice distribution given parameters (ν and SNR) of the distribution computed by parrice. The cumulative distribution function of the distribution is complex and numerical integration of the probability density function is used.

$$F(x) = 1 - Q\left(\frac{\nu}{\alpha}, \frac{x}{\alpha}\right)$$

where $F(x)$ is the nonexceedance probability for quantile x, $Q(a, b)$ is the Marcum Q-function, and ν/α is a form of signal-to-noise ratio SNR. If $\nu = 0$, then the Rayleigh distribution results and pdfray is used. The Marcum Q-function is difficult to work with and the **lmomco** uses the `integrate` function on pdfrice.

Usage

```
cdfrice(x, para)
```

Arguments

x	A real value.
para	The parameters from parrice or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

See Also

pdfrice, quarice, parrice

Examples

```
lmr <- vec2lmom(c(45,0.27), lscale=FALSE)
cdfrice(35,parrice(lmr))
```

---

cdftexp *Cumulative Distribution Function of the Truncated Exponential Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Truncated Exponential distribution given parameters (ξ and α) of the distribution computed by partexp. The cumulative distribution function of the distribution is

$$F(x) = \frac{1 - \exp(-t/\alpha)}{1 - \exp(-\xi/\alpha)},$$

where $F(x)$ is the nonexceedance probability for the quantile x, ξ is a location parameter, α is a scale parameter, and $0 \leq x \leq \xi$. The distribution has $0 < \tau_2 <= 1/2$, $\xi > 0$, and $1/\alpha \neq 0$.

Usage

```
cdftexp(x, para)
```

Arguments

x	A real value.
para	The parameters from partexp or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Vogel, R.M., Hosking, J.R.M., Elphick, C.S., Roberts, D.L., and Reed, J.M., 2008, Goodness of fit of probability distributions for sightings as species approach extinction: Bulletin of Mathematial Biology, v. 71, no. 3, pp. 701–719.

See Also

pdftexp, quatexp, partexp

Examples

```
lmr <- vec2lmom(c(40,0.38), lscale=FALSE)
cdftexp(50,partexp(lmr))

# Vogel and others (2008) example sighting times for the bird
# Eskimo Curlew, inspection shows that these are fairly uniform.
# There is a sighting about every year to two.
T <- c(1946, 1947, 1948, 1950, 1955, 1956, 1959, 1960, 1961,
       1962, 1963, 1964, 1968, 1970, 1972, 1973, 1974, 1976,
       1977, 1980, 1981, 1982, 1982, 1983, 1985)
R <- 1945 # beginning of record
S <- T - R
PARcurlew <- partexp(lmoms(S))
Xmax <- quatexp(1, PARcurlew)
X <- seq(0,Xmax, by=1)
plot(X, cdftexp(X,PARcurlew), type="l")

# Plot looks not curved enough to show the texp?
# Try S <- S^2 and rerunning the last four lines.
```

---

cdfwak *Cumulative Distribution Function of the Wakeby Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Wakeby distribution given parameters (ξ, α, β, γ, and δ) of the distribution computed by parwak. The cumulative distribution function of the distribution has no explicit form.

Usage

```
cdfwak(x, wakpara)
```

Arguments

x	A real value.
wakpara	The parameters from parwak or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

quawak, parwak

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
cdfwak(50,parwak(lmr))
```

---

cdfwei — *Cumulative Distribution Function of the Weibull Distribution*

Description

This function computes the cumulative probability or nonexceedance probability of the Weibull distribution given parameters (ζ, β, and δ) of the distribution computed by parwei. The cumulative distribution function of the distribution is

$$F(x) = 1 - e^{-\frac{x+\zeta}{\beta}^{\delta}},$$

where $F(x)$ is the nonexceedance probability for quantile x, ζ is a location parameter, β is a scale parameter, and δ is a shape parameter.

The Weibull distribution is a reverse Generalized Extreme Value distribution. As result, the Generalized Extreme Value algorithms are used for implementation of the Weibull in this package. The relation between the Generalized Extreme Value parameters (ξ, α, and κ) is

$$\kappa = 1/\delta,$$

$$\alpha = \beta/\delta, \text{ and}$$

$$\xi = \zeta - \beta.$$

These relations are taken from Hosking and Wallis (1997).

In R the cumulative distribution function of the Weibull distribution is `pweibull`. Given a Weibull parameter object `para`, the R syntax is `pweibull(x+para$para[1], para$para[3], scale=para$para[2])`. For the current implementation for this package, the reversed Generalized Extreme Value distribution is used `1-cdfgev(-x,para)`.

Usage

```
cdfwei(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parwei` or similar.

Value

Nonexceedance probability (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`quawei`, `parwei`

Examples

```
# Evaluate Weibull deployed here and within R (pweibull)
lmr <- lmom.ub(c(123,34,4,654,37,78))
WEI <- parwei(lmr)
F1  <- cdfwei(50,WEI)
F2  <- pweibull(50+WEI$para[1],shape=WEI$para[3],scale=WEI$para[2])
if(F1 == F2) EQUAL <- TRUE

# The Weibull is a reversed generalized extreme value
Q <- sort(rlmomco(34,WEI)) # generate Weibull sample
lm1 <- lmoms(Q)    # regular L-moments
lm2 <- lmoms(-Q)   # L-moment of negated (reversed) data
WEI <- parwei(lm1) # parameters of Weibull
GEV <- pargev(lm2) # parameters of GEV
F <- nonexceeds()  # Get a vector of nonexceedance probs
plot(pp(Q),Q)
lines(cdfwei(Q,WEI),Q,lwd=5,col=8)
lines(1-cdfgev(-Q,GEV),Q,col=2) # line over laps previous
```

---

check.fs *Check Vector of Nonexceedance Probabilities*

Description

This function checks that a nonexceedance probability (F) is in the $0 \leq F \leq 1$ range. It does not check that the distribution whether the function as specified by current parameters if valid for $F = 0$ or $F = 1$. End point checking is left to additional internal checks within the functions implementing the distribution. The function is intended for internal use within this library to build logic flow throughout the distribution functions. Users are not expected to need this function themselves. The `check.fs` function is separate because of the heavy use of the logic across a myriad of functions in this package.

Usage

```
check.fs(fs)
```

Arguments

`fs`	A vector of nonexceedance probablity values.

Value

`TRUE`	The nonexceedance probabilities are valid.
`FALSE`	The nonexceedance probabilities are invalid.

Author(s)

W.H. Asquith

See Also

quacau, quaexp, quagam, quagev, quagld, quaglo, quagno, quagpa, quagum, quakap, quanor, quape3, quawak, and quawei

Examples

```
F <- c(0.5,0.7,0.9,1.1)
if(check.fs(F) == FALSE) cat("Bad nonexceedances 0<F<1\n")
```

---

check.pdf *Check and Potentially Graph Probability Density Functions*

Description

This convenience function checks that a given probability density function from **lmomco** appears to workout as mathematically valid. Basically a pdf function must integrate to unity. The `check.fs` function permits some flexibility in the limits of integration and provides a high-level interface from graphical display of the pdf.

Usage

```
check.pdf(pdfunc,  para, lowerF=0.001, upperF=0.999,
eps=0.02, verbose=FALSE, plot=FALSE, plotlowerF=0.001,
plotupperF=0.999, ...)
```

Arguments

`pdfunc`	A probability density function from **lmomco**.
`lowerF`	The lower bounds of nonexceedance probability for the numerical integration.
`upperF`	The upper bounds of nonexceedance probability for the numerical integration.
`para`	The parameters of the distribution.
`eps`	An error term expressing allowable error (deviation) of the numerical integration from unity. (If that is the objective of the call to the `check.pdf` function.)
`verbose`	Is verbose output desired?
`plot`	Should a plot (polygon) of the pdf integration be produce?
`plotlowerF`	Alternative lower limit for the generation of the curve depicting the pdf function.
`plotupperF`	Alternative upper limit for the generation of the curve depicting the pdf function.
`...`	Additional arguments that are passed onto the `integration` function.

Value

An R list structure is returned

`isunity`	Given the `eps` is `F` close enough.
`F`	The numerical integration of the probability density function from `lowerF` to `upperF`.

Author(s)

W.H. Asquith

Examples

```
lmr <- vec2lmom(c(100,40,0.1)) # Arbitrary L-moments
gev <- pargev(lmr) # parameters of Generalized Extreme Value distribution
wei <- parwei(lmr) # parameters of Weibull distribution

# The Weibull is effectively a reversed GEV and the plots in the
# following examples should demonstrate this.

# Two examples that should integrate to "unity" given default parameters.
check.pdf(pdfgev,gev,plot=TRUE)
check.pdf(pdfwei,wei,plot=TRUE)

# Two examples that will not, but the integrated value on the return list
# should be very close to the median (F=0.5) and the resulting plots
# should affirm what this convenience function is actually doing.
check.pdf(pdfgev,upperF=0.5,gev,plot=TRUE)
check.pdf(pdfwei,upperF=0.5,wei,plot=TRUE)
```

---

claudeprecip *Annual Maximum Precipitation Data for Claude, Texas*

Description

Annual maximum precipitation data for Claude, Texas

Usage

```
data(claudeprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(claudeprecip)
summary(claudeprecip)
```

`clearforkporosity` *Porosity Data*

Description

Porosity (fraction of void space) from neutron-density, well log for 5,350–5,400 feet below land surface for Permian Age Clear Fork formation, Ector County, Texas

Usage

```
clearforkporosity
```

Format

A data frame with

POROSITY The pre-sorted porosity data.

Details

Although the porosity data was collected at about 1-foot intervals, these intervals are not provided in the data frame. Further, the porosity data has been sorted to disrupt the specific depth to porosity relation to remove the proprietary nature of the original data.

Examples

```
data(clearforkporosity)
plot(clearforkporosity)
```

`dist.list` *List of Distribution Names*

Description

Return a list of the three character syntax identifying distributions supported within the **lmomco** package. The distributions are `aep`, `cau`, `exp`, `gam`, `gev`, `gld`, `glo`, `gno`, `gpa`, `gum`, `kap`, `kur`, `lap`, `ln3`, `nor`, `pe3`, `ray`, `revgum`, `rice`, `texp`, `wak`, and `wei`.

Usage

```
dist.list()
```

Arguments

No arguments are needed.

Value

A vector of distribution identifiers.

Author(s)

W.H. Asquith

Examples

```
# Build an L-moment object
LM <- vec2lmom(c(10000,1500,0.3,0.1,0.04))
lm2 <- lmorph(LM)  # convert to vectored format
lm1 <- lmorph(lm2) # and back to named format

dist <- dist.list()

# demonstrate that lmom2par internally converts
# to needed L-moment object
for(i in seq(1,length(dist))) {
  # skip Cauchy (needs TL-moments) and GLD (speed)
  # Reverse Gumbel needs censoring . . .
  if(dist[i] == 'cau' | dist[i] == 'gld' | dist[i] == 'aep' |
     dist[i] == 'revgum' | dist[i] == 'kur' |
     dist[i] == 'rice' | dist[i] == 'texp') next
  print(lmom2par(lm1,type=dist[i]))
  print(lmom2par(lm2,type=dist[i]))
}
```

---

dlmomco *Probability Density Function of the Distributions*

Description

This function acts as an alternative front end to par2pdf. The nomenclature of the dlmomco function is to mimic that of built-in **R** functions.

Usage

```
dlmomco(x,para)
```

Arguments

x	A real value.
para	The parameters from lmom2par or similar.

Value

Probability density for x.

Author(s)

W.H. Asquith

See Also

plmomco, rlmomco, qlmomco

Examples

```
para <- vec2par(c(0,1),type='nor') # standard normal parameters
nonexceed <- dlmomco(1,para) # percentile of one standard deviation
```

DrillBitLifetime *Lifetime of Drill Bits*

Description

Hamada (1995, table 9.3) provides a table of lifetime to breakage measured in cycles for drill bits used for producing small holes in printed circuit boards. The data are originally credited to an F. Montmarquet. The data were collected under various control and noise factors to perform reliability assessment to maximize bit reliability with minimization of hole diameter. Smaller holes permit higher density of placed circuitry, and are thus economically attractive. The testing was completed at 3,000 cycles—the right censoring threhold.

Usage

```
data(DrillBitLifetime)
```

Format

A data frame with

LIFETIME Measured in cycles.

References

Hamada, M., 1995, Analysis of experiments for reliability improvement and robust reliability: in Balakrishnan, N. (ed.) Recent Advances in Life-Testing and Reliability: Boca Raton, Fla., CRC Press, ISBN 0–8493–8972–0, pp. 155–172.

Examples

```
data(DrillBitLifetime)
summary(DrillBitLifetime)
```

`expect.max.ostat` *Compute the Expection of a Maximum Order Statistic (or others)*

Description

This function computes

Usage

```
expect.max.ostat(n, para=NULL, cdf=NULL, pdf=NULL,
                 j=NULL, lower=-Inf, upper=Inf)
```

Arguments

`n`	The sample size.
`para`	A distribution parameter list from a function such as `vec2par` or `lmom2par`.
`cdf`	CDF of the distribution for the parameters
`pdf`	PDF of the distribution for the parameters
`j`	The jth value of the order statistic, which defaults to *n=j* if `j=NULL`.
`lower`	The lower limit of the distribution for integration.
`upper`	The upper limit of the distribution for integration.

Value

The expectation of the maximum order statistic, unless j is specified and then the expectation of that order statistic is returned.

Author(s)

W.H. Asquith

See Also

`theoLmoms.max.ostat`

Examples

```
para <- vec2par(c(10,100), type="nor")

# The two output values from these two lines should be
# very similar: (1) theoretical and (2) simulation
expect.max.ostat(10, para=para, cdf=cdfnor, pdf=pdfnor)
mean(sapply(1:1000, function(x) { max(rlmomco(10,para))}))
```

fliplmoms *Flip L-moments by Flip Attribute in L-moment Vector*

Description

This function flips the L-moments by a flip attribute in an L-moment object such as that returned by `lmomsRCmark`. The function will attempt to identify the L-moment object and `lmorph` as necessary, but this support is not guaranteed. The flipping process is used to support left-tail censoring using the right-tail censoring alogrithms of the package. The odd order (`seq(3,n,by2)`) λ_r and τ_r are negated. The mean $\hat{\lambda}_1$ is computed by subtracting the λ_1 from the *lmom* argument from the flip *M*: $\hat{\lambda}_1 = M - \lambda_1$ and the τ_2 is subsequently adjusted by the new mean. This function is written to provide a convenient method to re-transform or back flip the L-moments computed by `lmomsRCmark`. Careful analysis of the example problem listed here should be made.

Usage

```
fliplmoms(lmom, flip=NULL, checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmomsRCmark` or other vectorize L-moment list.
`flip`	`lmomsRCmark` provides the flip, but for other vectorized L-moment list support, the flip can be set by this argument.
`checklmom`	Should the *lmom* be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned that matches the structure of the *lmom* argument (unless an `lmorph` was attempted). The structure is intended to match that coming from `lmomsRCmark`.

Author(s)

W.H. Asquith

References

Wang, Dongliang, Hutson, A.D., Miecznikowski, J.C., 2010, L-moment estimation for parametric survival models given censored data: Statistical Methodology, preprint

Helsel, D.R., 2005, Nondetects and data analysis—Statistics for censored environmental data: Hoboken, New Jersey, John Wiley, 250 p.

See Also

lmomsRCmark

Examples

```
# Create some data with **multiple detection limits**
# This is a left-tail censoring problem--flipping will be required.
fakedat1 <- rnorm(50, mean=16, sd=5);
fake1.left.censor.indicator <- fakedat1 < 14;
fakedat1[fake1.left.censor.indicator]   <- 14;

fakedat2 <- rnorm(50, mean=16, sd=5);
fake2.left.censor.indicator <- fakedat2 < 10;
fakedat2[fake2.left.censor.indicator]   <- 10;

# combine the data sets
fakedat <- c(fakedat1, fakedat2);
fake.left.censor.indicator <- c(fake1.left.censor.indicator,
                                fake2.left.censor.indicator);
ix <- order(fakedat);
fakedat <- fakedat[ix];
fake.left.censor.indicator <- fake.left.censor.indicator[ix];

lmr.usual       <- lmoms(fakedat);
lmr.flipped     <- lmomsRCmark(fakedat, flip=TRUE,
                               rcmark=fake.left.censor.indicator);
lmr.backflipped <- fliplmoms(lmr.flipped); # re-transform
pch <- as.numeric(fake.left.censor.indicator)*15 + 1;
F <- nonexceeds();
plot(pp(fakedat), sort(fakedat), pch=pch,
     xlab="NONEXCEEDANCE PROBABILITY",
     ylab="DATA VALUE");
lines(F, qlmomco(F, parnor(lmr.backflipped)), lwd=2)
lines(F, qlmomco(F, parnor(lmr.usual)), lty=2)
legend(0,20, c("Uncensored", "Left-tail censored"), pch=c(1,16))
# The solid line represented the Normal distribution fit by
# censoring indicator on the multiple left-tail detection limits.


## Not run:
# see example in pwmRC
H <- c(3,4,5,6,6,7,8,8,9,9,9,10,10,11,11,11,13,13,13,13,13,
       17,19,19,25,29,33,42,42,51.9999,52,52,52)
# 51.9999 was really 52, a real (noncensored) data point.
flip <- 100
F <- flip - H #
RCpwm <- pwmRC(H, threshold=52)
lmorph(pwm2lmom(vec2pwm(RCpwm$Bbetas))) # OUTPUT1 STARTS HERE

LCpwm <- pwmLC(F, threshold=(flip - 52))
LClmr <- pwm2lmom(vec2pwm(LCpwm$Bbetas))
LClmr <- lmorph(LClmr)
```

```
#LClmr$flip <- 100; fliplmoms(LClmr) # would also work
fliplmoms(LClmr, flip=flip) # OUTPUT2 STARTS HERE

# The two outputs are the same showing how the flip
# argument works

## End(Not run)
```

---

freq.curve.all *Compute Frequency Curve for All Distributions*

Description

This function is dispatcher on top of the suite of quaCCC functions that compute frequency curves for the L-moments. Frequency curves in hydrologic science is a term typically renaming the more conventional quantile function. The notation CCC represents the three character notation for the distribution: exp, gam, gev, gld, glo, gno, gpa, gum, kap, nor, pe3, wak, and wei. The Cauchy distribution is not used because of its dependency on trimmed L-moments and its general lack of use in applied research problems (at least those familiar to the author). The nonexceedance probabilities to construct the curves are derived from nonexceeds.

Usage

```
freq.curve.all(lmom,aslog10=FALSE,asprob=TRUE,
                      no2para=FALSE,no3para=FALSE,
    no4para=FALSE,no5para=FALSE,
                      step=FALSE,show=FALSE,
                      xmin=NULL,xmax=NULL,xlim=NULL,
                      ymin=NULL,ymax=NULL,ylim=NULL,
                      exp=TRUE,gam=TRUE,gev=TRUE,gld=FALSE,
    glo=TRUE,gno=TRUE,gpa=TRUE,gum=TRUE,
    kap=TRUE,nor=TRUE,pe3=TRUE,wak=TRUE,
    wei=TRUE,...)
```

Arguments

lmom	A L-moment object from lmom.ub or similar.
aslog10	Compute log10 of quantiles—note that NaNs produced in: log(x, base) will be produced for less than zero values.
asprob	The qnorm function is used to convert nonexceedance probabilities, which are produced by nonexceeds, to standard normal deviates. The normal distribution will plot as straight line when this argument is TRUE
no2para	If TRUE, do not run the 2-parameter distributions: exp, gam, gum, and nor.

`no3para`	If `TRUE`, do not run the 3-parameter distributions: `gev`, `glo`, `gno`, `gpa`, `pe3`, and `wei`.
`no4para`	If `TRUE`, do not run the 4-parameter distributions: `kap` and `gld`.
`no5para`	If `TRUE`, do not run the 5-parameter distributions: `wak`.
`step`	Shows incremental processing of each distribution.
`show`	Plots all the frequency curves in a simple (crowded) `plot`.
`xmin`	Minimum x-axis value to use instead of the automatic value determined from the nonexceedance probabilities. This argument is only used is `show=TRUE`.
`xmax`	Maximum x-axis value to use instead of the automatic value determined from the nonexceedance probabilities. This argument is only used is `show=TRUE`.
`xlim`	Both limits of the x-axis. This argument is only used is `show=TRUE`.
`ymin`	Minimum y-axis value to use instead of the automatic value determined from the nonexceedance probabilities. This argument is only used is `show=TRUE`.
`ymax`	Maximum y-axis value to use instead of the automatic value determined from the nonexceedance probabilities. This argument is only used is `show=TRUE`.
`ylim`	Both limits of the y-axis. This argument is only used is `show=TRUE`.
`exp`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gam`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gev`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gld`	A logical switch on computation of corresponding distribution—default is `FALSE`.
`glo`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gno`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gpa`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`gum`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`kap`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`nor`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`pe3`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`wak`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`wei`	A logical switch on computation of corresponding distribution—default is `TRUE`.
`...`	Additional parameters are passed to the parameter estimation routines such as `parexp`.

Value

An extensive R `data.frame` of frequency curves. The nonexceedance probability values, which are provided by `nonexceeds`, are the first item in the `data.frame` under the heading of `nonexceeds`. If a particular distribution could not be fit to the L-moments of the data; this particular function returns zeros so that a `data.frame` can be returned.

Author(s)

W.H. Asquith

See Also

quaexp, quagam, quagev, quagld,
quaglo, quagno, quagpa, quagum,
quakap, quanor, quape3, quawak, and
quawei.

Examples

```
L <- vec2lmom(c(35612,23593,0.48,0.21,0.11))
freq.curve.all(L,gld=FALSE)
freq.curve.all(L,step=TRUE,no2para=TRUE,no4para=TRUE)
```

---

gen.freq.curves — *Plot Randomly Generated Frequency Curves from a Parent Distribution*

Description

This function generates random samples of specified size from a specified parent distribution. Subsequently, the type of parent distribution is fit to the L-moments of the generated sample. The fitted distribution is then plotted with the built-in function `lines`. It is the user's responsibility to have an active `plot` already drawn; unless the `callplot` option is `TRUE`. This function is useful to demonstration of sample size on the uncertainty of a fitted distribution—a motivation for this function is for class room exercise.

Usage

```
gen.freq.curves(n, para, F=NULL, nsim=10, callplot=TRUE,
aslog=FALSE, asprob=FALSE, showsample=FALSE, showparent=FALSE, ...)
```

Arguments

`n`	Sample size to draw from parent as specified by `para`.
`para`	The parameters from `lmom2par` or similar.
`F`	The nonexceedance probabilities for horizontal axis—defaults to `nonexceeds` when the argument is `NULL`.
`nsim`	The number of simulations to perform (frequency curves to draw)—the default is 10.
`callplot`	Calls `plot` to acquire a graphics device—default is `TRUE`, but the called `plot` is left empty.
`aslog`	Compute `log10` of quantiles—note that `NaNs produced in: log(x, base)` will be produced for less than zero values. Otherwise this is a harmless message.

`asprob`	The `qnorm` function is used to convert nonexceedance probabilities, which are produced by `nonexceeds`, to standard normal deviates. The normal distribution will plot as straight line when this argument is `TRUE` and `aslog=FALSE`.
`showsample`	Each simulated sample is drawn through plotting positions (`pp`).
`showparent`	The curve for the parent distribution is plotted on exit from the function if `TRUE`. Further plotting options can not be controlled—unlike the situation with the drawing of the simulated frequency curves.
`...`	Additional parameters are passed to the `lines` call within the function—except for the drawing of the parent distribution (see argument `showparent`.

Value

No value is returned. This function is used for its graphical side effects.

Author(s)

W.H. Asquith

See Also

`lmom2par`, `nonexceeds`, `rlmomco`, `lmoms`

Examples

```
para <- vec2par(c(140,4),type='gam') # build a gamma parent
F    <- nonexceeds() # vector of nonexceedance probabilities
Q    <- quagam(F,para) # the quantiles of parent distribution
# simulated are grey, parent is black
gen.freq.curves(10,para,nsim=15,asprob=TRUE,showparent=TRUE,col=8)
```

`genci` *Generate Confidence Intervals for Quantiles of a Parent Distribution*

Description

This function estimates the lower and upper limits of a specified confidence interval for a vector of quantile value of a specified parent distribution [quantile function $Q(F, \theta)$ with parameters θ] using Monte Carlo simulation. The quantile values, actually specified by a vector of nonexceedance probabilities (F for $0 \le F < 1$) of the values, are specified by the user. The user also provides the parameters of the parent distribution (see `lmom2par`). This function is a wrapper on `qua2ci`; please consult the documentation for that function for further details of the simulation.

Usage

```
genci(para, n, F=NULL, ci=0.90, edist='nor', nsim=1000,
      expand=FALSE, verbose=FALSE, showpar=FALSE, quiet=FALSE)
```

Arguments

`para` The parameters from `lmom2par` or similar.

`n` The sample size that the Monte Carlo simulation will use.

`F` Vector of nonexceedance probabilities ($0 \leq F \leq 1$) of the quantiles for which the confidence interval are needed. If `NULL`, then the vector as returned by `nonexceeds` is used.

`ci` The confidence interval ($0.5 \leq$ `ci` < 1). The interval is specified as the size of the interval. The default is 0.90 or the 90th percentile. The function will return the 5th (1-0.90)/2 and 95th (1-(1-0.90)/2) percentile cumulative probability of the error distribution for the parent quantile as specified by the nonexceedance probability argument (`F`). This argument is passed unused to `qua2ci`.

`edist` The model for the error distribution. Although the normal (the default) is typically assumed in error analyses, it need not be, as support for other distributions supported by the **lmomco** package is available. However, one should seriously consider the values of the simulated L-moments when choosing an error distribution other than the normal. If the L-skew (τ_4) or L-kurtosis (τ_4) values depart considerably from those of the normal ($\tau_3 = 0$ and $\tau_4 = 0.122602$), alternative distributions would likely provide more reliable confidence interval estimation. This argument is passed unused to `qua2ci`.

`nsim` The number of simulations for the sample size `n` to perform. Much larger simulation numbers are highly recommended—see discussion about `qua2ci`. This argument is passed unused to `qua2ci`. Users are encouraged to play with `qua2ci` to get a feel for the value of `edist` and `nsim`.

`expand` Should the returned values be expanded to include information relating to the distribution type and L-moments of the distribution at the corresponding nonexceedance probabilities—in other words the information necessary to reconstruct the reported confidence interval. The default is `FALSE`. If `expand=FALSE` then a single `data.frame` of the lower and upper limits along with the true quantile value of the parent is returned. If `expand=TRUE`, then a more complicated `list` containing multiple `data.frames` is returned.

`verbose` The verbosity of the operation of the function. This argument is passed unused to `qua2ci`.

`showpar` The parameters of the `edist` for each simulation for each F value passed to `qua2ci` are printed. This argument is passed unused to `qua2ci`.

`quiet` Suppress incremental counter for a count down of the F values.

Value

An **R** `data.frame` or `list` is returned (see discussion of argument `expand`). The following elements could be available.

`nonexceed_prob` A vector of F values, which is returned for convenience so that post operations such as plotting are easily coded.

`lower` The lower value of the confidence interval having nonexceedance probability equal to `(1-ci)/2`.

true	The true quantile value from $Q(F, \theta)$ for the corresponding F value.
upper	The upper value of the confidence interval having F equal to 1-(1-ci)/2.
lscale	The second L-moment (L-scale, λ_2) of the distribution of quantiles for the corresponding F. This value is included in the primary returned data.frame because it measures the fundamental sampling variability.
lcv	The ratio of lscale to true. A measure of relative variability
parent	The paraments of the parent distribution if expand=TRUE.
edist	The type of error distribution used to model the confidence interval if the argument expand=TRUE is set.
elmoms	The L-moment of the distribution of quantiles for the corresponding F if the argument expand=TRUE is set.
epara	The parameter list of the error distribution fit to the elmoms if the argument expand=TRUE is set.
ifail	A failure integer.
ifailtext	Text message associated with ifail.

Author(s)

W.H. Asquith

See Also

lmoms, lmom2par, qua2ci, gen.freq.curves

Examples

```
  # For all these examples, nsim is way too small.
  MEAN  <- 0    # mean of zero
  SIGMA <- 100  # standard deviation of 100
  PAR   <- vec2par(c(MEAN,SIGMA),type='nor') # make parameter object
  F     <- c(0.5, 0.8, 0.9, 0.96, 0.98, 0.99) # nonexceed probabilities
  # nsim is small for speed of example not accuracy.
  CI    <- genci(PAR,n=10,F=F,nsim=20)
  plot(CI$nonexceed_prob,CI$true,type='l',lwd=2)
  lines(CI$nonexceed_prob,CI$lower,col=2)
  lines(CI$nonexceed_prob,CI$upper,col=3)

## Not run:
  pdf("twoCIplots.pdf")
  # The qnorm call has been added to produce "normal probability"
  # paper on the horizonal axis. The parent is heavy-tailed.
  GEV <- vec2par(c(10000,1500,-.3),type='gev') # a GEV distribution
  CI  <- genci(GEV,n=20,nsim=200,edist='gno')
  ymin <- log10(min(CI$lower[! is.na(CI$lower)]))
  ymax <- log10(max(CI$upper[! is.na(CI$upper)]))
  plot( qnorm(CI$nonexceed_prob),log10(CI$true),type='l',
               ylim=c(ymin,ymax),lwd=2)
  lines(qnorm(CI$nonexceed_prob),log10(CI$lower),col=2)
```

```
  lines(qnorm(CI$nonexceed_prob),log10(CI$upper),col=3)
  # another error distribution model
  CI  <- genci(GEV,n=20,nsim=200,edist='aep4')
  lines(qnorm(CI$nonexceed_prob),log10(CI$lower),col=2, lty=2)
  lines(qnorm(CI$nonexceed_prob),log10(CI$upper),col=3, lty=2)
  dev.off()

## End(Not run)
```

---

gini.mean.diff *Gini Mean Difference Statistic*

Description

The Gini mean difference statistic $\mathcal{G}$ is a robust estimator of distribution scale and is closely related to the second L-moment $\lambda_2 = \mathcal{G}/2$.

$$\mathcal{G} = \frac{2}{n(n-1)} \sum_{i=1}^{n} (2i - n - 1) X_{i:n}$$

where $X_{i:n}$ are the order statistics.

Usage

```
gini.mean.diff(x)
```

Arguments

`x`	A vector of data values that will be reduced to non-missing values.

Value

An R `list` is returned.

`gini`	The gini mean difference $\mathcal{G}$
`L2`	The L-scale (second L-moment) via $0.5 * \mathcal{G}$
`source`	An attribute identifying the computational source of the Gini's Mean Difference: "gini.mean.diff".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Jurečková, J., and Picek, J., 2006, Robust statistical methods with R: Boca Raton, Fla., Chapman and Hall/CRC, ISBN 1–58488–454–1, 197~p.

See Also

lmoms

Examples

```
fake.dat <- c(123,34,4,654,37,78)
gini <- gini.mean.diff(fake.dat)
lmr <- lmoms(fake.dat)
str(gini)
print(abs(gini$L2 - lmr$lambdas[2]))
```

---

harmonic.mean *The Harmonic Mean with Zero-Value Correction*

Description

Compute the harmonic mean of a vector with a zero-value correction.

$$\breve{\mu} = \left(\frac{\sum_{i=1}^{N_T - N_0} 1/X_i}{N_T - N_0}\right)^{-1} \times \frac{N_T - N_0}{N_T},$$

where $\breve{\mu}$ is harmonic mean, X_i is a nonzero value of the vector, N_T is the (total) sample size, N_0 is the number of zero values.

Usage

```
harmonic.mean(x)
```

Arguments

x	A vector of data values that will be reduced to non-missing values.

Value

An R `list` is returned.

harmean	The harmonic mean with zero-value correction, $\breve{\mu}$.
correction	The zero-value correction, $(N_T - N_0)/N_T$.
source	An attribute identifying the computational source of the harmonic mean: "harmonic.mean".

Note

The harmonic mean can not be computed when zero values are present. This situation is common in surface-water hydrology. As stated in the reference below, in order to calculate water-quality-based effluent limits (WQBELs) for human health protection, a harmonic mean flow is determined for all perennial streams and for streams that are intermittent with perennial pools. Sometimes these streams have days on which measured flow is zero. Because a zero flow cannot be used in the calculation of harmonic mean flow, the second term in the harmonic mean equation is an adjustment factor used to lower the harmonic mean to compensate for days on which the flow was zero. The zero-value correction is the same correction used by the EPA computer program DFLOW.

Author(s)

W.H. Asquith

References

Texas Commission on Environmental Quality, 2003, Procedures to implement the Texas surface-water-quality standards: TCEQ RG–194, p. 47

See Also

pmoms

Examples

```
Q <- c(0,0,5,6,7)
harmonic.mean(Q)
```

---

herefordprecip *Annual Maximum Precipitation Data for Hereford, Texas*

Description

Annual maximum precipitation data for Hereford, Texas

Usage

```
data(herefordprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(herefordprecip)
summary(herefordprecip)
```

---

hmomco *Hazard Functions of the Distributions*

Description

This function acts as a front end to dlmomco and plmomco to compute the hazard function $h(x)$ or conditional failure rate. The function is defined by

$$h(x) = \frac{f(x)}{1 - F(x)},$$

where $f(x)$ is a probability density function and $F(x)$ is the cumulative distribution function.

To help with intuitive understanding of what $h(x)$ means (Ugarte and others, 2008), let dx represent a small unit of measurement. Then the quantity $h(x)dx$ can be conceptualized as the approximate probability that random variable X takes on a value in the interval $[x, x + dx]$.

Ugarte and others (2008) continue by stating that $h(x)$ represents the instantaneous rate of death or failure at time x, given the survival to time x has occurred. Emphasis is needed that $h(x)$ is a rate of probability change and not a probability itself.

Usage

```
hlmomco(x,para)
```

Arguments

x	A real value.
para	The parameters from lmom2par or similar.

Value

Hazard rate for x.

Author(s)

W.H. Asquith

References

Ugarte, M.D., Militino, A.F., and Arnholt, A.T., 2008, Probability and statistics with R: Boca Raton, FL., CRC Press, 700 p.

See Also

plmomco, dlmomco

Examples

```
my.lambda <- 100
para <- vec2par(c(0,my.lambda), type="exp")

x <- seq(40:60)
hlmomco(x,para) # returns vector of 0.01
# because the exponential distribution has a constant
# failure rate equal to 1/scale or 1/100 as in this example.
```

IRSrefunds.by.state *U.S.~Internal Revenue Service Refunds by State for Fiscal Year 2006*

Description

U.S.~Internal Revenue Service refunds by state for fiscal year 2006.

Usage

```
data(IRSrefunds.by.state)
```

Format

A data frame with

STATE State name.

REFUNDS Dollars of refunds.

References

http://www.irs.gov/taxstats/article/0,,id=168593,00.html accessed in December 2007.

Examples

```
data(IRSrefunds.by.state)
summary(IRSrefunds.by.state)
```

is.aep4 — *Is a Distribution Parameter Object Typed as 4-p Asymmetric Exponential Power*

Description

The distribution parameter object returned by functions of this module such as by `paraep4` are typed by an attribute `type`. This function checks that `type` is `aep4` for the 4-parameter Asymmetric Exponential Power distribution.

Usage

```
is.aep4(para)
```

Arguments

`para`	A parameter `list` returned from `paraep4`.

Value

`TRUE`	If the `type` attribute is `aep4`.
`FALSE`	If the `type` is not `aep4`.

Author(s)

W.H. Asquith

See Also

`paraep4`

Examples

```
para <- vec2par(c(0,1, 0.5, 4), type="aep4")
if(is.aep4(para) == TRUE) {
  Q <- quaaep4(0.55,para)
}
```

is.cau *Is a Distribution Parameter Object Typed as Cauchy*

Description

The distribution parameter object returned by functions of this module such as by vec2par are typed by an attribute type. This function checks that type is cau for the Cauchy distribution.

Usage

```
is.cau(para)
```

Arguments

para	A parameter list returned from parcau or vec2par.

Value

TRUE	If the type attribute is cau.
FALSE	If the type is not cau.

Author(s)

W.H. Asquith

See Also

vec2par

Examples

```
para <- vec2par(c(12,12),type='cau')
if(is.cau(para) == TRUE) {
  Q <- quacau(0.5,para)
}
```

is.exp *Is a Distribution Parameter Object Typed as Exponential*

Description

The distribution parameter object returned by functions of this module such as by parexp are typed by an attribute type. This function checks that type is exp for the Exponential distribution.

Usage

```
is.exp(para)
```

Arguments

para	A parameter list returned from parexp.

Value

TRUE	If the type attribute is exp.
FALSE	If the type is not exp.

Author(s)

W.H. Asquith

See Also

parexp

Examples

```
para <- parexp(lmom.ub(c(123,34,4,654,37,78)))
if(is.exp(para) == TRUE) {
  Q <- quaexp(0.5,para)
}
```

---

is.gam *Is a Distribution Parameter Object Typed as Gamma*

Description

The distribution parameter object returned by functions of this module such as by pargam are typed by an attribute type. This function checks that type is gam for the Gamma distribution.

Usage

```
is.gam(para)
```

Arguments

para	A parameter list returned from pargam.

Value

TRUE	If the type attribute is gam.
FALSE	If the type is not gam.

Author(s)

W.H. Asquith

See Also

pargam

Examples

```
para <- pargam(lmom.ub(c(123,34,4,654,37,78)))
if(is.gam(para) == TRUE) {
  Q <- quagam(0.5,para)
}
```

---

is.gev *Is a Distribution Parameter Object Typed as Generalized Extreme Value*

Description

The distribution parameter object returned by functions of this module such as by `pargev` are typed by an attribute `type`. This function checks that `type` is `gev` for the Generalized Extreme Value distribution.

Usage

```
is.gev(para)
```

Arguments

`para`	A parameter `list` returned from `pargev`.

Value

`TRUE`	If the `type` attribute is `gev`.
`FALSE`	If the `type` is not `gev`.

Author(s)

W.H. Asquith

See Also

pargev

Examples

```
para <- pargev(lmom.ub(c(123,34,4,654,37,78)))
if(is.gev(para) == TRUE) {
  Q <- quagev(0.5,para)
}
```

is.gld *Is a Distribution Parameter Object Typed as Generalized Lambda*

Description

The distribution parameter object returned by functions of this module such as by `vec2par` are typed by an attribute `type`. This function checks that `type` is `gld` for the Generalized Lambda distribution.

Usage

```
is.gld(para)
```

Arguments

`para`	A parameter `list` returned from `vec2par`.

Value

`TRUE`	If the `type` attribute is `gld`.
`FALSE`	If the `type` is not `gld`.

Author(s)

W.H. Asquith

See Also

`cdfgld`, `quagld`

Examples

```
para <- vec2par(c(123,120,3,2),type="gld")
if(is.gld(para) == TRUE) {
  Q <- quagld(0.5,para)
}
```

is.glo *Is a Distribution Parameter Object Typed as Generalized Logistic*

Description

The distribution parameter object returned by functions of this module such as by `parglo` are typed by an attribute `type`. This function checks that `type` is `glo` for the Generalized Logistic distribution.

Usage

```
is.glo(para)
```

Arguments

`para`	A parameter `list` returned from `parglo`.

Value

`TRUE`	If the `type` attribute is `glo`.
`FALSE`	If the `type` is not `glo`.

Author(s)

W.H. Asquith

See Also

`parglo`

Examples

```
para <- parglo(lmom.ub(c(123,34,4,654,37,78)))
if(is.glo(para) == TRUE) {
  Q <- quaglo(0.5,para)
}
```

`is.gno` *Is a Distribution Parameter Object Typed as Generalized Normal*

Description

The distribution parameter object returned by functions of this module such as by `pargno` are typed by an attribute `type`. This function checks that `type` is `gno` for the Generalized Normal distribution.

Usage

```
is.gno(para)
```

Arguments

`para`	A parameter `list` returned from `pargno`.

Value

`TRUE`	If the `type` attribute is `gno`.
`FALSE`	If the `type` is not `gno`.

Author(s)

W.H. Asquith

See Also

pargno

Examples

```
para <- pargno(lmom.ub(c(123,34,4,654,37,78)))
if(is.gno(para) == TRUE) {
  Q <- quagno(0.5,para)
}
```

---

is.gpa *Is a Distribution Parameter Object Typed as Generalized Pareto*

Description

The distribution parameter object returned by functions of this module such as by `pargpa` are typed by an attribute `type`. This function checks that `type` is `gpa` for the Generalized Pareto distribution.

Usage

```
is.gpa(para)
```

Arguments

para	A parameter `list` returned from `pargpa`.

Value

TRUE	If the `type` attribute is `gpa`.
FALSE	If the `type` is not `gpa`.

Author(s)

W.H. Asquith

See Also

pargpa

Examples

```
para <- pargpa(lmom.ub(c(123,34,4,654,37,78)))
if(is.gpa(para) == TRUE) {
  Q <- quagpa(0.5,para)
}
```

is.gum — *Is a Distribution Parameter Object Typed as Gumbel*

Description

The distribution parameter object returned by functions of this module such as by `pargum` are typed by an attribute `type`. This function checks that `type` is `gum` for the Gumbel distribution.

Usage

```
is.gum(para)
```

Arguments

`para`	A parameter `list` returned from `pargum`.

Value

`TRUE`	If the `type` attribute is `gum`.
`FALSE`	If the `type` is not `gum`.

Author(s)

W.H. Asquith

See Also

`pargum`

Examples

```
para <- pargum(lmom.ub(c(123,34,4,654,37,78)))
if(is.gum(para) == TRUE) {
  Q <- quagum(0.5,para)
}
```

is.kap — *Is a Distribution Parameter Object Typed as Kappa*

Description

The distribution parameter object returned by functions of this module such as by `parkap` are typed by an attribute `type`. This function checks that `type` is `kap` for the Kappa distribution.

Usage

```
is.kap(para)
```

Arguments

`para`	A parameter `list` returned from `parkap`.

Value

`TRUE`	If the `type` attribute is `kap`.
`FALSE`	If the `type` is not `kap`.

Author(s)

W.H. Asquith

See Also

`parkap`

Examples

```
para <- parkap(lmom.ub(c(123,34,4,654,37,78)))
if(is.kap(para) == TRUE) {
  Q <- quakap(0.5,para)
}
```

`is.kur` *Is a Distribution Parameter Object Typed as Kumaraswamy*

Description

The distribution parameter object returned by functions of this module such as by `parkur` are typed by an attribute `type`. This function checks that `type` is `kur` for the Kumaraswamy distribution.

Usage

```
is.kur(para)
```

Arguments

`para`	A parameter `list` returned from `parkur`.

Value

`TRUE`	If the `type` attribute is `kur`.
`FALSE`	If the `type` is not `kur`.

Author(s)

W.H. Asquith

See Also

parkur

Examples

```
para <- parkur(lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9)))
if(is.kur(para) == TRUE) {
  Q <- quakur(0.5,para)
}
```

---

is.lap *Is a Distribution Parameter Object Typed as Laplace*

Description

The distribution parameter object returned by functions of this module such as by `parlap` are typed by an attribute `type`. This function checks that `type` is `lap` for the Laplace distribution.

Usage

```
is.lap(para)
```

Arguments

para	A parameter `list` returned from `parlap`.

Value

TRUE	If the `type` attribute is `lap`.
FALSE	If the `type` is not `lap`.

Author(s)

W.H. Asquith

See Also

parlap

Examples

```
para <- parlap(lmom.ub(c(123,34,4,654,37,78)))
if(is.lap(para) == TRUE) {
  Q <- qualap(0.5,para)
}
```

is.ln3 *Is a Distribution Parameter Object Typed as 3-Parameter Log-Normal*

Description

The distribution parameter object returned by functions of this module such as by `parln3` are typed by an attribute `type`. This function checks that `type` is `ln3` for the 3-parameter log-Normal distribution.

Usage

```
is.ln3(para)
```

Arguments

`para`	A parameter `list` returned from `parln3`.

Value

`TRUE`	If the `type` attribute is `ln3`.
`FALSE`	If the `type` is not `ln3`.

Author(s)

W.H. Asquith

See Also

`parln3`

Examples

```
para <- vec2par(c(.9252, .1636, .7),type='ln3')
if(is.ln3(para)) {
  Q <- qualn3(0.5,para)
}
```

is.nor — *Is a Distribution Parameter Object Typed as Normal*

Description

The distribution parameter object returned by functions of this module such as by `parnor` are typed by an attribute `type`. This function checks that `type` is `nor` for the Normal distribution.

Usage

```
is.nor(para)
```

Arguments

`para`	A parameter `list` returned from `parnor`.

Value

`TRUE`	If the `type` attribute is `nor`.
`FALSE`	If the `type` is not `nor`.

Author(s)

W.H. Asquith

See Also

`parnor`

Examples

```
para <- parnor(lmom.ub(c(123,34,4,654,37,78)))
if(is.nor(para) == TRUE) {
  Q <- quanor(0.5,para)
}
```

is.pe3 — *Is a Distribution Parameter Object Typed as Pearson Type III*

Description

The distribution parameter object returned by functions of this module such as by `parpe3` are typed by an attribute `type`. This function checks that `type` is `pe3` for the Pearson Type III distribution.

Usage

```
is.pe3(para)
```

Arguments

`para`	A parameter `list` returned from `parpe3`.

Value

`TRUE`	If the `type` attribute is `pe3`.
`FALSE`	If the `type` is not `pe3`.

Author(s)

W.H. Asquith

See Also

`parpe3`

Examples

```
para <- parpe3(lmom.ub(c(123,34,4,654,37,78)))
if(is.pe3(para) == TRUE) {
  Q <- quape3(0.5,para)
}
```

`is.ray` *Is a Distribution Parameter Object Typed as Rayleigh*

Description

The distribution parameter object returned by functions of this module such as by `parray` are typed by an attribute `type`. This function checks that `type` is `ray` for the Rayleigh distribution.

Usage

```
is.ray(para)
```

Arguments

`para`	A parameter `list` returned from `parray`.

Value

`TRUE`	If the `type` attribute is `ray`.
`FALSE`	If the `type` is not `ray`.

Author(s)

W.H. Asquith

See Also

parray

Examples

```
para <- vec2par(c(.9252, .1636, .7),type='ray')
if(is.ray(para)) {
  Q <- quaray(0.5,para)
}
```

---

is.revgum — *Is a Distribution Parameter Object Typed as Reverse Gumbel*

Description

The distribution parameter object returned by functions of this module such as by `parrevgum` are typed by an attribute `type`. This function checks that `type` is `revgum` for the Reverse Gumbel distribution.

Usage

```
is.revgum(para)
```

Arguments

`para`	A parameter `list` returned from `parrevgum`.

Value

`TRUE`	If the `type` attribute is `revgum`.
`FALSE`	If the `type` is not `revgum`.

Author(s)

W.H. Asquith

See Also

parrevgum

Examples

```
para <- vec2par(c(.9252, .1636, .7),type='revgum')
if(is.revgum(para)) {
  Q <- quarevgum(0.5,para)
}
```

is.rice — *Is a Distribution Parameter Object Typed as Rice*

Description

The distribution parameter object returned by functions of this module such as by `parrice` are typed by an attribute `type`. This function checks that `type` is `rice` for the Rice distribution.

Usage

```
is.rice(para)
```

Arguments

`para`	A parameter `list` returned from `parrice`.

Value

`TRUE`	If the `type` attribute is `rice`.
`FALSE`	If the `type` is not `rice`.

Author(s)

W.H. Asquith

See Also

`parrice`

Examples

```
para <- vec2par(c(3, 4),type='rice')
if(is.rice(para)) {
  Q <- quarice(0.5,para)
}
```

is.texp — *Is a Distribution Parameter Object Typed as Truncated Exponential*

Description

The distribution parameter object returned by functions of this module such as by `partexp` are typed by an attribute `type`. This function checks that `type` is `texp` for the Truncated Exponential distribution.

Usage

```
is.texp(para)
```

Arguments

para	A parameter list returned from partexp.

Value

TRUE	If the type attribute is texp.
FALSE	If the type is not texp.

Author(s)

W.H. Asquith

See Also

partexp

Examples

```
para <- parexp(lmom.ub(c(123,34,4,654,37,78)))
if(is.exp(para) == TRUE) {
  Q <- quaexp(0.5,para)
}
```

---

is.wak *Is a Distribution Parameter Object Typed as Wakeby*

Description

The distribution parameter object returned by functions of this module such as by parwak are typed by an attribute type. This function checks that type is wak for the Wakeby distribution.

Usage

```
is.wak(para)
```

Arguments

para	A parameter list returned from parwak.

Value

TRUE	If the type attribute is wak.
FALSE	If the type is not wak.

Author(s)

W.H. Asquith

See Also

parwak

Examples

```
para <- parwak(lmom.ub(c(123,34,4,654,37,78)))
if(is.wak(para) == TRUE) {
  Q <- quawak(0.5,para)
}
```

---

is.wei *Is a Distribution Parameter Object Typed as Weibull*

Description

The distribution parameter object returned by functions of this module such as by `parwei` are typed by an attribute `type`. This function checks that `type` is `wei` for the Weibull distribution.

Usage

```
is.wei(para)
```

Arguments

para	A parameter `list` returned from `parwei`.

Value

TRUE	If the `type` attribute is `wei`.
FALSE	If the `type` is not `wei`.

Author(s)

W.H. Asquith

See Also

parwei

Examples

```
para <- parwei(lmom.ub(c(123,34,4,654,37,78)))
if(is.wei(para) == TRUE) {
  Q <- quawei(0.5,para)
}
```

LaguerreHalf *Laguerre Polynomial (Half)*

Description

This function estimates the Laguerre polynomial, which is useful in applications involving the variance of the Rice distribution.

Usage

```
LaguerreHalf(x)
```

Arguments

x	A value.

Value

The value for the Laguerre polynomial is returned.

Author(s)

W.H. Asquith

See Also

pdfrice

Examples

```
LaguerreHalf(-100^2/(2*10^2))
```

Lcomoment.coefficients *L-comoment Coefficient Matrix*

Description

Compute the L-comoment coefficients from an L-comoment matrix of order $k \geq 2$ and the $k = 2$ (2nd order) L-comoment matrix. However, if the first argument is 1st-order then the coefficients of L-covariation are computed. The function requires that each matrix has already computed by the function `Lcomoment.matrix`.

Usage

```
Lcomoment.coefficients(Lk,L2)
```

Arguments

`Lk`	A $k \geq 2$ L-comoment matrix from `Lcomoment.matrix`.
`L2`	A $k = 2$ L-comoment matrix from `Lcomoment.matrix(Dataframe,k=2)`.

Details

The coefficient of L-variation is computed by `Lcomoment.coefficients(L1,L2)` where `L1` is a 1st-order L-moment matrix and `L2` is a $k = 2$ L-comoment matrix. Symbolically, the coefficient of L-covariation is

$$\hat{\tau}_{[12]} = \frac{\hat{\lambda}_{2[12]}}{\hat{\lambda}_{1[12]}}.$$

The higher L-comoment coefficients (L-coskew, L-cokurtosis, ...) are computed by the function `Lcomoment.coefficients(L3,L2)` ($k = 3$), `Lcomoment.coefficients(L4,L2)` ($k = 4$), and so on. Symbolically, the higher L-comoment coefficients are

$$\hat{\tau}_{k[12]} = \frac{\hat{\lambda}_{k[12]}}{\hat{\lambda}_{2[12]}}, \text{ for } k \geq 3.$$

Finally, the usual univariate L-moment ratios as seen from `lmom.ub` or `lmoms` are along the diagonal. The `Lcomoment.coefficients` function does not make use of `lmom.ub` or `lmoms`.

Value

An R `list` is returned.

`type`	The type of L-comoment representation in the matrix: “Lcomoment.coefficients”.
`order`	The order of the coefficients. `order=2` L-covariation, `order=3` L-coskew, ...
`matrix`	A $k \geq 2$ L-comoment coefficient matrix.

Note

The function begins with a capital letter. This is intentionally done so that lower case namespace is preserved. L-comoments are new in the literature and experimental in this package. By using a capital letter now, then `lcomoment.coefficients` remains an available name in future releases.

Author(s)

W.H. Asquith

References

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

lmom.ub, Lcomoment.matrix, Lcomoment.coefficients

Examples

```
D      <- data.frame(X1=rnorm(30),X2=rnorm(30),X3=rnorm(30))
L1     <- Lcomoment.matrix(D,k=1)
L2     <- Lcomoment.matrix(D,k=2)
L3     <- Lcomoment.matrix(D,k=3)
LkLCV  <- Lcomoment.coefficients(L1,L2)
LkTAU3 <- Lcomoment.coefficients(L3,L2)
```

Lcomoment.correlation — *L-correlation Matrix (L-correlation through Sample L-comoments)*

Description

Compute the L-correlation from an L-comoment matrix of order $k = 2$. This function assumes that each matrix is already computed by the function Lcomoment.matrix.

Usage

```
Lcomoment.correlation(L2)
```

Arguments

L2	A $k = 2$ L-comoment matrix from Lcomoment.matrix(Dataframe,k=2).

Details

L-correlation is computed by Lcomoment.coefficients(L2,L2) where L2 is second order L-comoment matrix. The usual L-scale values as seen from lmom.ub or lmoms are along the diagonal. This function does not make use of lmom.ub or lmoms and can be used to verify computation of τ (coefficient of L-variation).

Value

An R list is returned.

type	The type of L-comoment representation in the matrix: “Lcomoment.coefficients”.
order	The order of the matrix—extracted from the first matrix in arguments.
matrix	A $k \geq 2$ L-comoment coefficient matrix.

Note

The function begins with a capital letter. This is intentionally done so that lower case namespace is preserved. L-comoments are new in the literature and experimental in this package. By using a capital letter now, then lcomoment.correlation remains an available name in future releases.

Author(s)

W.H. Asquith

References

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

`lmom.ub`, `Lcomoment.matrix`, `Lcomoment.correlation`

Examples

```
D   <- data.frame(X1=rnorm(30),X2=rnorm(30),X3=rnorm(30))
L2  <- Lcomoment.matrix(D,k=2)
RHO <- Lcomoment.correlation(L2)
```

`Lcomoment.Lk12` *Compute a Single Sample L-comoment*

Description

Compute the $k \geq 1$ order L-comoment ($\lambda_{k[r:n]}$) for a given pair of random variables. The order of the L-comoments is specified.

Usage

```
Lcomoment.Lk12(X1,X2,k=1)
```

Arguments

`X1`	An vector of random variables (a sample of random variable 1).
`X2`	Another vector of random variables (a sample of random variable 2).
`k`	The order of the L-comoment to compute. The default is 1.

Details

L-comoments of random variable $X1$ are computed from the concomitants of $X2$. That is, $X2$ is sorted in ascending order to create the order statistics of $X2$. $X1$ is in turn reshuffled to the order of $X2$ for form the concomitants of $X2$ (denoted as $X^{(12)}$). The concomitants inturn are used in a weighted summation and expectation calculation to compute the L-comoment of $X1$ with respect to $X2$. The inverse can also be done (`Lcomoment.Lk12(X2,X1,k=1)`) and is not necessarily equal to (`Lcomoment.Lk12(X1,X2,k=1)`). The notation of `Lk12` is to read “Lambda for kth order L-comoment”, where the `12` portion of the notation reflects that of Serfling and Xiao (2006). The weights for the computation are derived from calls by `Lcomoment.Lk12` to `Lcomoment.Wk`.

$$\hat{\lambda}_{k[12]} = \frac{1}{n}\sum_{r=1}^{n} w^{(k)}_{r:n} x^{(12)}_{[r:n]}$$

The L-comoments of $X2$ are computed from the concomitants of $X1$ ($X^{(21)}$) are formed by sorting $X1$ in ascending order and in turn shuffling $X2$ by the order of $X1$. The sample concomitants are thus formed ($x^{(12)}_{[r:n]}$). By symmetry the L-comoment is

$$\hat{\lambda}_{k[21]} = \frac{1}{n}\sum_{r=1}^{n} w^{(k)}_{r:n} x^{(21)}_{[r:n]}$$

Value

A single L-comoment.

Note

The function begins with a capital letter. This is intentionally done so that lower case namespace is preserved. L-comoments are new in the literature and experimental in this package. By using a capital letter now, then `lcomoment.Lk12` or similar remains an available name in future releases.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

`Lcomoment.matrix`, `Lcomoment.Wk`

Examples

```
X1   <- rnorm(20)
X2   <- rnorm(20)
Lk12 <- Lcomoment.Lk12(X1,X2,k=1)
```

`Lcomoment.matrix` *Compute Sample L-comoment Matrix*

Description

Compute the L-comoments from a rectangular `data.frame` contain arrays of random variables. The order of the L-comoments is specified.

Usage

```
Lcomoment.matrix(DATAFRAME,k=1)
```

Arguments

`DATAFRAME`	A convential `data.frame` that is rectangular
`k`	The order of the L-comoments to compute. Default is $k = 1$

Details

L-comoments are computed for each item in the `data.frame`. L-comoments of order $k = 1$ are means and comeans. L-coments of order $k = 2$ are L-scale and L-coscale values. L-comoments of order $k = 3$ are L-skew and L-coskews. L-comoments of order $k = 4$ are L-kurtosis and L-cokurtosis, and so on. The usual univariate L-moments of order k as seen from `lmom.ub` are along the diagonal. This function does not make use of `lmom.ub`. The `Lcomoment.matrix` function calls `Lcomment.Lk12` for each cell in the matrix. The L-comoment matrix for d-random variables is

$$\boldsymbol{\Lambda_k} = (\hat{\lambda}_{k[ij]})$$

computed over the pairs $(X^{(i)}, X^{(j)})$ where $1 \leq i \leq j \leq d$.

Value

An R `list` is returned.

`type`	The type of L-comoment representation in the matrix: “Lcomoments”.
`order`	The order of the matrix—specified by k in the argument list.
`matrix`	A kth order L-comoment matrix.

Note

The function begins with a capital letter. This is intentionally done so that lower case namespace is preserved. L-comoments are new in the literature and experimental in this package. By using a capital letter now, then `lcomoment.matrix` remains an available name in future releases.

Author(s)

W.H. Asquith

References

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

Lcomoment.Lk12, Lcomoment.coefficients, lmom.ub

Examples

```
D  <- data.frame(X1=rnorm(30),X2=rnorm(30),X3=rnorm(30))
L1 <- Lcomoment.matrix(D,k=1)
L2 <- Lcomoment.matrix(D,k=2)
```

Lcomoment.Wk *Weighting Coefficient for Sample L-comoment*

Description

Compute the weight factors for computation of an L-comoment for order k, order statistic r, and sample size n.

Usage

```
Lcomoment.Wk(k,r,n)
```

Arguments

k	Order of L-comoment being computed by parent calls to Lcomoment.Wk.
r	Order statistic index involved.
n	Sample size.

Details

This function computes the weight factors needed to calculation L-comoments and is interfaced or used by Lcomoment.Lk12. This function is not necessarily for end users. The weight factor $w^{(k)}_{r:n}$ is the discrete Legendre polynomial. The weight factors are well illustrated in figure 2.6 of Hosking and Wallis (1997).

$$w^{(k)}_{r:n} = \sum_{j=0}^{min\{r-1,k-1\}} (-1)^{k-1-j} \frac{\binom{k-1}{j}\binom{k-1+j}{j}\binom{r-1}{j}}{\binom{n-1}{j}}.$$

Value

A single L-comoment weight factor.

Note

The function begins with a capital letter. This is intentionally done so that lower case namespace is preserved. L-comoments are new in the literature and experimental in this package. By using a capital letter now, then `lcomoment.Wk` remains an available name in future releases.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

`Lcomoment.Lk12`

Examples

```
Wk <- Lcomoment.Wk(2,3,5)
print(Wk)

## Not run:
# To compute the weight factors for L-skew and L-coskew (k=3) computation
# for a sample of size 20.
Wk <- matrix(nrow=20,ncol=1)
for(r in seq(1,20)) Wk[r] <- Lcomoment.Wk(3,r,20)
plot(seq(1,20),Wk, type="b")

## End(Not run)

# The following shows the actual weights used for computation of
# the first four L-moments. The sum of the each sample times the
# corresponding weight equals the L-moment.
fakedat <- sort(c(-10, 20, 30, 40));  n <- length(fakedat)
Wk1 <- Wk2 <- Wk3 <- Wk4 <- vector(mode="numeric", length=n);
for(i in 1:n) {
   Wk1[i] <- Lcomoment.Wk(1,i,n)/n
   Wk2[i] <- Lcomoment.Wk(2,i,n)/n
   Wk3[i] <- Lcomoment.Wk(3,i,n)/n
   Wk4[i] <- Lcomoment.Wk(4,i,n)/n
}
cat(c("Weights for mean",         round(Wk1, digits=4), "\n"))
cat(c("Weights for L-scale",      round(Wk2, digits=4), "\n"))
cat(c("Weights for 3rd L-moment", round(Wk3, digits=4), "\n"))
cat(c("Weights for 4th L-moment", round(Wk4, digits=4), "\n"))
my.lams <- c(sum(fakedat*Wk1), sum(fakedat*Wk2),
             sum(fakedat*Wk3), sum(fakedat*Wk4))
```

```
cat(c("Manual L-moments:", my.lams, "\n"))
cat(c("lmomco L-moments:", lmoms(fakedat, nmom=4)$lambdas,"\n"))
# The last two lines of output should be the same---note that lmoms()
# does not utilize Lcomoment.Wk(). So a double check is provided.
```

`lcomoms2` *The Sample L-comoments for Two Variables*

Description

Compute the sample L-moments for two variable `data.frame`. The "2" in the function name is to refer to fact that this function operates on only two variables. The length of the variables must be greater than the number of L-comoments requested.

Usage

```
lcomoms2(DATAFRAME, nmom=3, asdiag=FALSE, opdiag=FALSE, ...)
```

Arguments

`DATAFRAME`	A vector of data values.
`nmom`	The number of moments to compute. Default is 3.
`asdiag`	Return the `diagonal` of the matrices. Default is `FALSE`.
`opdiag`	Return the opposing diagonal of the matrices. Default is `FALSE`. This function returns the opposing diagonal from first two to second.
`...`	Additional arguments to pass.

Value

An R `list` is returned of the first

`L1`	Matrix or diagonals of first L-comoment.
`L2`	Matrix or diagonals of second L-comoment.
`T2`	Matrix or diagonals of L-comoment correlation.
`T3`	Matrix or diagonals of L-comoment skew.
`T4`	Matrix or diagonals of L-comoment kurtosis.
`T5`	Matrix or diagonals of L-comoment Tau5.
`source`	An attribute identifying the computational source of the L-comoments: "lcomoms2".

Note

This function computes the L-comoments through the generalization of the `Lcomoment.matrix` and `Lcomoment.coefficients` functions.

Author(s)

W.H. Asquith

References

Serfling, R., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v.~98, pp.~1765–1781.

See Also

`Lcomoment.matrix` and `Lcomoment.coefficients`

Examples

```
# Random simulation of standard normal and then combine with
# a random standard exponential distribution
X <- rnorm(200); Y <- X + rexp(200)

z <- lcomoms2(data.frame(X=X, Y=Y))
print(z)


z <- lcomoms2(data.frame(X=X, Y=Y), diag=TRUE)
print(z$T3) # the L-skew values of the margins

z <- lcomoms2(data.frame(X=X, Y=Y), opdiag=TRUE)
print(z$T3) # the L-coskew values
```

`lmom.ub` *Unbiased Sample L-moments by Direct Sample Estimators*

Description

Unbiased sample L-moments are computed for a vector using the direct sample estimation method as opposed to the use of sample probability-weighted moments. The L-moments are the ordinary L-moments and not the trimmed L-moments (see `TLmoms`). The mean, L-scale, coefficient of L-variation (τ, LCV, L-scale/mean), L-skew (τ_3, TAU3, L3/L2), L-kurtosis (τ_4, TAU4, L4/L2), and τ_5 (TAU5, L4/L2) are computed. In conventional nomenclature, the L-moments are

$$\hat{\lambda}_1 = \mathrm{L1} = \text{mean},$$

$$\hat{\lambda}_2 = \mathrm{L2} = \text{L-scale},$$

$$\hat{\lambda}_3 = \mathrm{L3} = \text{third L-moment},$$

$$\hat{\lambda}_4 = \mathrm{L4} = \text{fourth L-moment},$$

$$\hat{\lambda}_5 = \mathrm{L5} = \text{fifth L-moment},$$

$$\hat{\tau} = \mathrm{LCV} = \lambda_2/\lambda_1 = \text{coefficient of L-variation},$$

$$\hat{\tau}_3 = \text{TAU3} = \lambda_3/\lambda_2 = \text{L-skew},$$

$$\hat{\tau}_4 = \text{TAU4} = \lambda_4/\lambda_2 = \text{L-kurtosis, and}$$

$$\hat{\tau}_5 = \text{TAU5} = \lambda_5/\lambda_2 = \text{not named.}$$

Usage

```
lmom.ub(x)
```

Arguments

`x`	A vector of data values.

Details

The L-moment ratios (τ, τ_3, τ_4, and τ_5) are the primary higher L-moments for application, such as for distribution parameter estimation. However, the actual L-moments (λ_3, λ_4, and λ_5) are also reported. This implementation of L-moment calculation requires a minimum of five data points. If you want to compute more or fewer L-moments, then see `lmoms`.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew—analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: “lmom.ub”.

Author(s)

W.H. Asquith

Source

The Perl code base of W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

Wang, Q.J., 1996b, Direct sample estimators of L-moments: Water Resources Research, vol. 32, no. 12., pp. 3617–3619.

See Also

`lmom2pwm`, `pwm.ub`, `pwm2lmom`, `lmoms`, and `lmorph`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmorph(lmr)
lmom.ub(rnorm(100))
```

`lmom2par` *Convert L-moments to the the Parameters of a Distribution*

Description

This function converts the L-moments of the data to the parameters of a distribution. The type of distribution is specified in the argument list: `aep4`, `exp`, `gam`, `gev`, `gld`, `glo`, `gno`, `gpa`, `gum`, `kap`, `kur`, `ln3`, `nor`, `pe3`, `ray`, `revgum`, `wak`, or `wei`. There is no Cauchy distribution support by the `lmom2par` function because Cauchy uses TL-moments and not the ordinary L-moments. For Cauchy functions such as `parcau` and `lmomcau` should be used in accordance with their documentation.

Usage

```
lmom2par(lmom, type, ...)
```

Arguments

`lmom`	An L-moment object such as that returned by `lmom.ub` or `pwm2lmom`.
`type`	Three character distribution type (for example, `type='gev'`).
`...`	Additional arguments for the `parCCC` functions.

Value

An R `list` is returned. This list should contain at least the following items, but some distributions such as the `revgum` have extra.

`type`	The type of distribution in three character format.
`para`	The parameters of the distribution.
`source`	Attribute specifying source of the parameters.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105-124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis–An approach based on L-moments: Cambridge University Press.

See Also

`lmom2par`

Examples

```
lmr      <- lmom.ub(rnorm(20))
para     <- lmom2par(lmr,type='nor')
```

`lmom2pwm` *L-moments to Probability-Weighted Moments*

Description

Converts the L-moments to the Probability-Weighted Moments (PWMs) given the L-moments. The conversion is linear so procedures based on L-moments are identical to those based on PWMs. The relation between L-moments and PWMs is shown with `pwm2lmom`.

Usage

```
lmom2pwm(lmom)
```

Arguments

`lmom`	An L-moment object created by `lmom.ub` or similar. The function also supports `lmom` as a vector of L-moments (λ_1, λ_2, τ_3, τ_4, and τ_5).

Details

PWMs are linear combinations of the L-moments and therefore contain the same statistical information of the data as the L-moments. However, the PWMs are harder to interpret as measures of probability distributions. The PWMs are included here for theoretical completeness and are not intended for use with the majority of the other functions implementing the various probability distributions. The relation between L-moments (λ_r) and PWMs (β_{r-1}) for $1 \le r \le 5$ order is

$$\lambda_1 = \beta_0,$$

$$\lambda_2 = 2\beta_1 - \beta_0,$$

$$\lambda_3 = 6\beta_2 - 6\beta_1 + \beta_0,$$

$$\lambda_4 = 20\beta_3 - 30\beta_2 + 12\beta_1 - \beta_0, \text{ and}$$

$$\lambda_5 = 70\beta_4 - 140\beta_3 + 90\beta_2 - 20\beta_1 + \beta_0.$$

The linearity between L-moments and PWMs means that procedures based on one are equivalent to the other. This function only accomodates the first five L-moments and PWMs. Therefore, at least five L-moments are required in the passed argument.

Value

An R `list` is returned.

`betas`	The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	Source of the PWMs: "pwm"

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, pwm.ub, pwm2lmom

Examples

```
pwm <- lmom2pwm(lmom.ub(c(123,34,4,654,37,78)))

lmom2pwm(lmom.ub(rnorm(100)))

lmom2pwm(lmoms(rnorm(100)))


lmomvec1 <- c(1000,1300,0.4,0.3,0.2,0.1)
pwmvec   <- lmom2pwm(lmomvec1)
print(pwmvec)
#$betas
#[1] 1000.0000 1150.0000 1070.0000  984.5000  911.2857
#
#$source
#[1] "lmom2pwm"


lmomvec2 <- pwm2lmom(pwmvec)
print(lmomvec2)
#$lambdas
#[1] 1000 1300  520  390  260
#
#$ratios
#[1]  NA 1.3 0.4 0.3 0.2
#
#$source
#[1] "pwm2lmom"


pwm2lmom(lmom2pwm(list(L1=25,L2=20,TAU3=.45,TAU4=0.2,TAU5=0.1)))
```

---

lmom2vec — *Convert an L-moment object to a Vector of L-moments*

Description

This function converts an L-moment object in the structure used by **lmomco** into a simple vector of λ_1, λ_2, τ_3, τ_4, τ_5.

Usage

```
lmom2vec(lmom, ...)
```

Arguments

`lmom`	L-moment object as from functions such as `lmoms`, `lmom.ub`, and `vec2lmom`.
`...`	Not presently used.

Value

A vector of the first five L-moments (λ_1, λ_2, τ_3, τ_4, τ_5) or mean, L-scale, L-skew, L-kurtosis, and Tau5.

Author(s)

W.H. Asquith

See Also

`lmom.ub`, `lmoms`, `lmorph`, and `vec2lmom`

Examples

```
lmr <- lmom.ub(rnorm(40))
lmom2vec(lmr)


lmr <- vec2lmom(c(140,150,.3,.2,-.1))
lmom2vec(lmr)
```

`lmomaep4` *L-moments of the 4-p Asymmetric Exponential Power Distribution*

Description

This function computes the L-moments of the 4-parameter Asymmetric Exponential Power distribution given the parameters (ξ, α, κ, and h) from `paraep4`. The first four L-moments of the distribution are complex.

The mean λ_1 is

$$\lambda_1 = \xi + \alpha(1/\kappa - \kappa)\frac{\Gamma(2/h)}{\Gamma(1/h)}$$

where $\Gamma(x)$ is the complete gamma function or `gamma()` in R.

The L-scale λ_2 is

$$\lambda_2 = -\frac{\alpha\kappa(1/\kappa - \kappa)^2\Gamma(2/h)}{(1+\kappa^2)\Gamma(1/h)} + 2\frac{\alpha\kappa^2(1/\kappa^3 + \kappa^3)\Gamma(2/h)I_{1/2}(1/h, 2/h)}{(1+\kappa^2)^2\Gamma(1/h)}$$

where $I_{1/2}(1/h, 2/h)$ is the cumulative distribution function of the beta distribution ($I_x(a,b)$) or `pbeta(1/2, shape1=1/h, shape2=2/h)` in R. This function is also referred to as the normalized incomplete beta function (Delicado and Goria, 2008) and defined as

$$I_x(a,b) = \frac{\int_0^x t^{a-1}(1-t)^{b-1}\,\mathrm{d}t}{\beta(a,b)}$$

where $\beta(1/h, 2/h)$ is the complete beta function or `beta(1/h, 2/h)` in R.

The third L-moment λ_3 is

$$\lambda_3 = A_1 + A_2 + A_3$$

where the A_i are

$$A_1 = \frac{\alpha(1/\kappa - \kappa)(\kappa^4 - 4\kappa^2 + 1)\Gamma(2/h)}{(1+\kappa^2)^2\Gamma(1/h)}$$

$$A_2 = -6\frac{\alpha\kappa^3(1/\kappa - \kappa)(1/\kappa^3 + \kappa^3)\Gamma(2/h)I_{1/2}(1/h, 2/h)}{(1+\kappa^2)^3\Gamma(1/h)}$$

$$A_3 = 6\frac{\alpha(1+\kappa^4)(1/\kappa - \kappa)\Gamma(2/h)\Delta}{(1+\kappa^2)^2\Gamma(1/h)}$$

and where Δ is

$$\Delta = \frac{1}{\beta(1/h, 2/h)}\int_0^{1/2} t^{1/h-1}(1-t)^{2/h-1} I_{(1-t)/(2-t)}(1/h, 3/h)\,\mathrm{d}t$$

The fourth L-moment λ_4 is

$$\lambda_4 = B_1 + B_2 + B_3 + B_4$$

where the B_i are

$$B_1 = -\frac{\alpha\kappa(1/\kappa - \kappa)^2(\kappa^4 - 8\kappa^2 + 1)\Gamma(2/h)}{(1+\kappa^2)^3\Gamma(1/h)}$$

$$B_2 = 12\frac{\alpha\kappa^2(\kappa^3 + 1/\kappa^3)(\kappa^4 - 3\kappa^2 + 1)\Gamma(2/h)I_{1/2}(1/h, 2/h)}{(1+\kappa^2)^4\Gamma(1/h)}$$

$$B_3 = -30\frac{\alpha\kappa^3(1/\kappa - \kappa)^2(1/\kappa^2 + \kappa^2)\Gamma(2/h)\Delta}{(1+\kappa^2)^3\Gamma(1/h)}$$

$$B_4 = 20\frac{\alpha\kappa^4(1/\kappa^5 + \kappa^5)\Gamma(2/h)\Delta_1}{(1+\kappa^2)^4\Gamma(1/h)}$$

and where Δ_1 is

$$\Delta_1 = \frac{\int_0^{1/2}\int_0^{(1-y)/(2-y)} y^{1/h-1}(1-y)^{2/h-1}z^{1/h-1}(1-z)^{3/h-1}\, I'\,\mathrm{d}z\,\mathrm{d}y}{\beta(1/h,2/h)\beta(1/h,3/h)}$$

for which $I' = I_{(1-z)(1-y)/(1+(1-z)(1-y))}(1/h, 2/h)$ is the cumulative distribution function of the beta distribution ($I_x(a,b)$) or `pbeta((1-z)(1-y)/(1+(1-z)(1-y)), shape1=1/h, shape2=2/h)` in R.

Usage

```
lmomaep4(para, paracheck=TRUE, t3t4only=FALSE)
```

Arguments

`para`	The parameters of the distribution.
`paracheck`	Should the parameters be checked for validity by the `are.paraep4.valid` function.
`t3t4only`	Return only the τ_3 and τ_4 for the parameters κ and h. The λ_1 and λ_2 are not explicitly used although numerical values for these two L-moments are required only to avoid computational errors. Care is made so that the α parameter that is in numerator of $\lambda_{2,3,4}$ is not used in the computation of τ_3 and τ_4. Hence, this option permits the computation of τ_3 and τ_4 when α is unknown or potentially known with great uncertainty.

Value

An R list is returned.

`lambdas`	Vector of the L-moments. First element is λ_1, second element is λ_2, and so on.
`ratios`	Vector of the L-moment ratios. Second element is τ, third element is τ_3 and so on.
`trim`	Trim level = 0
`leftrim`	Left trimming level = 0
`rightrim`	Right trimming level = 0
`source`	An attribute identifying the computational source of the L-moments: "lmomaep4".

or an alternative R list is returned if `t3t4only=TRUE`

`T3`	L-skew, τ_3
`T4`	L-kurtosis, τ_4

Author(s)

W.H. Asquith

References

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

See Also

paraep4, quaaep4, cdfaep4

Examples

```
## Not run:
para <- vec2par(c(0, 1, 0.5, 4), type="aep4")
lmomaep4(para)

## End(Not run)
```

---

lmomcau *Trimmed L-moments of the Cauchy Distribution*

Description

This function estimates the trimmed L-moments of the Cauchy distribution given the parameters (ξ and α) from parcau. The trimmed L-moments in terms of the parameters are

$$\lambda_1^{(1)} = \xi,$$

$$\lambda_2^{(1)} = 0.698\alpha,$$

$$\tau_3^{(1)} = 0, \text{ and}$$

$$\tau_4^{(1)} = 0.343.$$

Usage

```
lmomcau(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

`lambdas`	Vector of the trimmed L-moments. First element is $\lambda_1^{(1)}$, second element is $\lambda_2^{(1)}$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\tau^{(1)}$, third element is $\tau_3^{(1)}$ and so on.
`source`	An attribute identifying the computational source of the L-moments: "lmomcau"
`trim`	Level of symmetrical trimming used—`trim=1`.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Gilchrist, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

`parcau`, `quacau`, `cdfcau`

Examples

```
X1 <- rcauchy(20)
lmomcau(parcau(TLmoms(X1,trim=1)))
```

`lmomexp` *L-moments of the Exponential Distribution*

Description

This function estimates the L-moments of the Exponential distribution given the parameters (ξ and α) from `parexp`. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \alpha,$$

$$\lambda_2 = \alpha/2,$$

$$\tau_3 = 1/3,$$

$$\tau_4 = 1/6, \text{ and}$$

$$\tau_5 = 1/10.$$

Usage

```
lmomexp(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation–analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew–analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis–analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: "lmomexp".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`parexp`, `quaexp`, `cdfexp`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomexp(parexp(lmr))
```

lmomgam *L-moments of the Gamma Distribution*

Description

This function estimates the L-moments of the Gamma distribution given the parameters (α and β) from pargam. The L-moments in terms of the parameters are complicated and solved numerically.

Usage

```
lmomgam(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomgam”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pargam, quagam, cdfgam

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomgam(pargam(lmr))
```

`lmomgev` *L-moments of the Generalized Extreme Value Distribution*

Description

This function estimates the L-moments of the Generalized Extreme Value distribution given the parameters (ξ, α, and κ) from pargev. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \frac{\alpha}{\kappa}(1 - \Gamma(1+\kappa)),$$

$$\lambda_2 = \frac{\alpha}{\kappa}(1 - 2^{-\kappa})\Gamma(1+\kappa),$$

$$\tau_3 = \frac{2(1 - 3^{-\kappa})}{1 - 2^{-\kappa}} - 3, \text{ and}$$

$$\tau_4 = \frac{5(1 - 4^{-\kappa}) - 10(1 - 3^{-\kappa}) + 6(1 - 2^{-\kappa})}{1 - 2^{-\kappa}}.$$

Usage

```
lmomgev(para)
```

Arguments

`para`	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coefficient of variation.
`TAU3`	The third L-moment ratio or L-skew—analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: “lmomgev”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pargev, quagev, cdfgev

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomgev(pargev(lmr))
```

---

lmomgld *L-moments of the Generalized Lambda Distribution*

Description

This function estimates the L-moments of the Generalized Lambda distribution given the parameters (ξ, α, κ, and h) from vec2par. The L-moments in terms of the parameters are complicated; however, there are analytical solutions. There are no simple expressions of the parameters in terms of the L-moments. The first L-moment or the mean of the distribution is

$$\lambda_1 = \xi + \alpha\left(\frac{1}{\kappa+1} - \frac{1}{h+1}\right).$$

The second L-moment or L-scale in terms of the parameters and the mean is

$$\lambda_2 = \xi + \frac{2\alpha}{(\kappa+2)} - 2\alpha\left(\frac{1}{h+1} - \frac{1}{h+2}\right) - \xi.$$

The third L-moment in terms of the parameters, the mean, and L-scale is

$$\boldsymbol{Y} = 2\xi + \frac{6\alpha}{(\kappa+3)} - 3(\alpha+\xi) + \xi \text{ and}$$

$$\lambda_3 = \boldsymbol{Y} + 6\alpha\left(\frac{2}{h+2} - \frac{1}{h+3} - \frac{1}{h+1}\right).$$

The fourth L-moment in termes of the parameters and the first three L-moments is

$$\boldsymbol{Y} = \frac{-3}{h+4}\left(\frac{2}{h+2} - \frac{1}{h+3} - \frac{1}{h+1}\right),$$

$$\boldsymbol{Z} = \frac{20\xi}{4} + \frac{20\alpha}{(\kappa+4)} - 20\boldsymbol{Y}\alpha, \text{ and}$$

$$\lambda_4 = \boldsymbol{Z} - 5(\kappa + 3(\alpha+\xi) - \xi) + 6(\alpha+\xi) - \xi.$$

It is conventional to express L-moments in terms of only the parameters and not the other L-moments. Lengthy algebra and further manipulation yields such a system of equations. The L-moments of the distribution are

$$\lambda_1 = \xi + \alpha\left(\frac{1}{\kappa+1} - \frac{1}{h+1}\right),$$

$$\lambda_2 = \alpha\left(\frac{\kappa}{(\kappa+2)(\kappa+1)} + \frac{h}{(h+2)(h+1)}\right),$$

$$\lambda_3 = \alpha\left(\frac{\kappa(\kappa-1)}{(\kappa+3)(\kappa+2)(\kappa+1)} - \frac{h(h-1)}{(h+3)(h+2)(h+1)}\right), \text{ and}$$

$$\lambda_4 = \alpha\left(\frac{\kappa(\kappa-2)(\kappa-1)}{(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1)} + \frac{h(h-2)(h-1)}{(h+4)(h+3)(h+2)(h+1)}\right).$$

The L-moment ratios are

$$\tau_3 = \frac{\kappa(\kappa-1)(h+3)(h+2)(h+1) - h(h-1)(\kappa+3)(\kappa+2)(\kappa+1)}{(\kappa+3)(h+3)\times[\kappa(h+2)(h+1) + h(\kappa+2)(\kappa+1)]} \text{ and}$$

$$\tau_4 = \frac{\kappa(\kappa-2)(\kappa-1)(h+4)(h+3)(h+2)(h+1) + h(h-2)(h-1)(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1)}{(\kappa+4)(h+4)(\kappa+3)(h+3)\times[\kappa(h+2)(h+1) + h(\kappa+2)(\kappa+1)]}.$$

The pattern being established through symmetry, even higher L-moment ratios are readily obtained. Note the alternating substraction and addition of the two terms in the numerator of the L-moment ratios (τ_r). For odd $r \geq 3$ substraction is seen and for even $r \geq 3$ addition is seen. For example, the fifth L-moment ratio is

$$N1 = \kappa(\kappa-3)(\kappa-2)(\kappa-1)(h+5)(h+4)(h+3)(h+2)(h+1),$$

$$N2 = h(h-3)(h-2)(h-1)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1),$$

$$D1 = (\kappa+5)(h+5)(\kappa+4)(h+4)(\kappa+3)(h+3),$$

$$D2 = [\kappa(h+2)(h+1) + h(\kappa+2)(\kappa+1)], \text{ and}$$

$$\tau_5 = \frac{N1 - N2}{D1 \times D2}.$$

By inspection the τ_r equations are not applicable for negative integer values $k = \{-1, -2, -3, -4, \ldots\}$ and $h = \{-1, -2, -3, -4, \ldots\}$ as division by zero will result. There are additional, but difficult to formulate, restrictions on the parameters both to define a valid Generalized Lambda distribution as well as valid L-moments. Verification of the parameters is conducted through `are.pargld.valid`, and verification of the L-moment validity is conducted through `are.lmom.valid`.

Usage

```
lmomgld(para)
```

Arguments

para	The parameters of the distribution.

Value

An R list is returned.

lambdas	Vector of the TL-moments. First element is $\lambda_1^{(1)}$, second element is $\lambda_2^{(1)}$, and so on.
ratios	Vector of the TL-moment ratios. Second element is $\tau^{(1)}$, third element is $\tau_3^{(1)}$ and so on.
trim	Trim level = 0
leftrim	Left trimming level = 0
rightrim	Right trimming level = 0
source	An attribute identifying the computational source of the TL-moments: "lmomgld".

Author(s)

W.H. Asquith

Source

Derivations conducted by W.H. Asquith on February 11 and 12, 2006.

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Karvanen, J., Eriksson, J., and Koivunen, V., 2002, Adaptive score functions for maximum likelihood ICA: Journal of VLSI Signal Processing, vol. 32, p. 82–92.

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distibutions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

pargld, cdfgld, quagld

Examples

```
lmomgld(vec2par(c(10,10,0.4,1.3),type='gld'))

PARgld <- vec2par(c(0,1,1,.5), type="gld")
theoTLmoms(PARgld, nmom=6)
lmomgld(PARgld)
```

lmomglo *L-moments of the Generalized Logistic Distribution*

Description

This function estimates the L-moments of the Generalized Logistic distribution given the parameters (ξ, α, and κ) from parglo. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \alpha\left(\frac{1}{\kappa} - \frac{\pi}{\sin(\kappa\pi)}\right),$$

$$\lambda_2 = \frac{\alpha\kappa\pi}{\sin(\kappa\pi)},$$

$$\tau_3 = -\kappa, \text{ and}$$

$$\tau_4 = \frac{(1+5\kappa^2)}{6}.$$

Usage

```
lmomglo(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomglo”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

parglo, quaglo, cdfglo

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomglo(parglo(lmr))
```

---

lmomgno — *L-moments of the Generalized Normal Distribution*

Description

This function estimates the L-moments of the Generalized Normal (log-Normal) distribution given the parameters (ξ, α, and κ) from pargno. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \frac{\alpha}{\kappa}(1 - e^{\kappa^2/2}) \text{ and}$$

$$\lambda_2 = \frac{\alpha}{\kappa}(e^{\kappa^2/2})(1 - 2\Phi(-\kappa/\sqrt{2})),$$

where Φ is the cumulative distribution of the standard normal distribution. There are no simple expressions for τ_3, τ_4, and τ_5. Log transformation of the data prior to fitting of the Generalized Normal distribution is not required.

Usage

```
lmomgno(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew—analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: "lmomgno".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`pargno`, `quagno`, `cdfgno`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomgno(pargno(lmr))
```

lmomgpa *L-moments of the Generalized Pareto Distribution*

Description

This function estimates the L-moments of the Generalized Pareto distribution given the parameters (ξ, α, and κ) from pargpa. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \frac{\alpha}{\kappa + 1},$$

$$\lambda_2 = \frac{\alpha}{(\kappa + 2)(\kappa + 1)},$$

$$\tau_3 = \frac{(1 - \kappa)}{(\kappa + 3)}, \text{ and}$$

$$\tau_4 = \frac{(1 - \kappa)(2 - \kappa)}{(\kappa + 4)(\kappa + 3)}.$$

Usage

```
lmomgpa(para)
```

Arguments

para	The parameters of the distribution.

Value

An R list is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomgpa”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pargpa, quagpa, cdfgpa

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomgpa(pargpa(lmr))
```

---

lmomgpaRC *B-type L-moments of the Generalized Pareto Distribution with Right-Tail Censoring*

Description

This function computes the "B"-type L-moments of the Generalized Pareto distribution given the parameters (ξ, α, and κ) from pargpaRC and the right-tail censoring fraction ζ. The B-type L-moments in terms of the parameters are

$$\lambda^B_1 = \xi + \alpha m_1,$$

$$\lambda^B_2 = \alpha(m_1 - m_2),$$

$$\lambda^B_3 = \alpha(m_1 - 3m_2 + 2m_3),$$

$$\lambda^B_4 = \alpha(m_1 - 6m_2 + 10m_3 - 5m_4), \text{ and}$$

$$\lambda^B_5 = \alpha(m_1 - 10m_2 + 30m_3 - 35m_4 + 14m_5),$$

where $m_r = \{1-(1-\zeta)^{r+\kappa}\}/(r+\kappa)$ and ζ is the right-tail censor fraction or the probability $\mathrm{Pr}\{\}$ that x is less than the quantile at ζ nonexceedance probability: ($\mathrm{Pr}\{x < X(\zeta)\}$). Finally, the RC in the function name is to denote Right-tail Censoring.

Usage

```
lmomgpaRC(para)
```

Arguments

para — The parameters of the distribution. Note that if the ζ part of the parameters (see pargpaRC) is not present then zeta=1 is assumed.

Value

An R list is returned.

lambdas — Vector of the L-moments. First element is λ_1, second element is λ_2, and so on.

ratios — Vector of the L-moment ratios. Second element is τ, third element is τ_3 and so on.

source — An attribute identifying the computational source of the L-moments: "lmomgpa2".

message — For clarity, this function adds the unusual message to an L-moment object that the lambdas and ratios are B-type L-moments.

zeta — The censoring fraction. Assumed equal to unity if not present in the gpa parameter object.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

pargpa, pargpaRC, lmomgpa, quagpa, cdfgpa

Examples

```
para <- vec2par(c(1500,160,.3),type="gpa") # build a GPA parameter set
lmorph(lmomgpa(para))
lmomgpaRC(para) # zeta = 1 is internally assumed if not available
# The previous two commands should output the same parameter values from
# independent code bases.

# Now assume that we have the sample parameters, but the zeta is nonunity.
para$zeta = .8
lmomgpaRC(para) # The B-type L-moments.
```

`lmomgum` *L-moments of the Gumbel Distribution*

Description

This function estimates the L-moments of the Gumbel distribution given the parameters (ξ and α) from pargum. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + (0.5722\ldots)\alpha,$$

$$\lambda_2 = \alpha\log(2),$$

$$\tau_3 = 0.169925,$$

$$\tau_4 = 0.150375, \text{ and}$$

$$\tau_5 = 0.055868.$$

Usage

```
lmomgum(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew–analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomgum”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pargum, quagum, cdfgum

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomgum(pargum(lmr))
```

`lmomkap` *L-moments of the Kappa Distribution*

Description

This function estimates the L-moments of the Kappa distribution given the parameters (ξ, α, κ, and h) from parkap. The L-moments in terms of the parameters are complicated and are solved numerically.

Usage

```
lmomkap(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.

L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: "lmomkap".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

parkap, quakap, cdfkap

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomkap(parkap(lmr))
```

---

lmomkur *L-moments of the Kumaraswamy Distribution*

Description

This function estimates the L-moments of the Kumaraswamy distribution given the parameters (α and β) from parkur. The L-moments in terms of the parameters with $\eta = 1 + 1/\alpha$ are

$$\lambda_1 = \beta B(\eta, \beta),$$

$$\lambda_2 = \beta[B(\eta, \beta) - 2B(\eta, 2\beta)],$$

$$\tau_3 = \frac{B(\eta, \beta) - 6B(\eta, 2\beta) + 6B(\eta, 3\beta)}{B(\eta, \beta) - 2B(\eta, 2\beta)},$$

$$\tau_4 = \frac{B(\eta, \beta) - 12B(\eta, 2\beta) + 30B(\eta, 3\beta) - 40B(\eta, 4\beta)}{B(\eta, \beta) - 2B(\eta, 2\beta)}, \text{ and}$$

$$\tau_5 = \frac{B(\eta, \beta) - 20B(\eta, 2\beta) + 90B(\eta, 3\beta) - 140B(\eta, 4\beta) + 70B(\eta, 5\beta)}{B(\eta, \beta) - 2B(\eta, 2\beta)}.$$

where $B(a, b)$ is the complete beta function or `beta()`.

Usage

```
lmomkur(para)
```

Arguments

`para`	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew–analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: "lmomkur".

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

`parkur`, `quakur`, `cdfkur`

Examples

```
lmr <- lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9))
lmomkur(parkur(lmr))

## Not run:
A <- B <- exp(seq(-3,5, by=.05))
logA <- logB <- T3 <- T4 <- c();
i <- 0
for(a in A) {
  for(b in B) {
    i <- i + 1
    parkur <- list(para=c(a,b), type="kur");
```

```
    lmr <- lmomkur(parkur)
    logA[i] <- log(a); logB[i] <- log(b)
    T3[i] <- lmr$TAU3; T4[i] <- lmr$TAU4
  }
}
library(lattice)
contourplot(T3~logA+logB, cuts=20, lwd=0.5, label.style="align",
            xlab="LOG OF ALPHA", ylab="LOG OF BETA",
            xlim=c(-3,5), ylim=c(-3,5),
            main="L-SKEW FOR KUMARASWAMY DISTRIBUTION")
contourplot(T4~logA+logB, cuts=10, lwd=0.5, label.style="align",
            xlab="LOG OF ALPHA", ylab="LOG OF BETA",
            xlim=c(-3,5), ylim=c(-3,5),
            main="L-KURTOSIS FOR KUMARASWAMY DISTRIBUTION")

## End(Not run)
```

---

lmomlap *L-moments of the Laplace Distribution*

Description

This function estimates the L-moments of the Laplace distribution given the parameters (ξ and α) from parlap. The L-moments in terms of the parameters are

$$\lambda_1 = \xi,$$

$$\lambda_2 = \frac{3\alpha}{4},$$

$$\tau_3 = 0,$$

$$\tau_4 = \frac{17}{22},$$

$$\tau_5 = 0,$$

$$\tau_6 = \frac{31}{360}.$$

For r odd and $r \geq 3$, $\lambda_r = 0$, and for r even and $r \geq 4$, the L-moments using the hypergeometric function ${}_2F_1()$ are

$$\lambda_r = \frac{2\alpha}{r(r-1)}[1 - {}_2F_1(-r, r-1, 1, 1/2)]$$

Usage

```
lmomlap(para)
```

Arguments

`para`	The parameters of the distribution.

Value

An R `list` is returned.

`lambdas`	The L-moments
`ratios`	The L-moment ratios.
`source`	An attribute identifying the computational source of the L-moments: "lmomlap".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

`parlap`, `qualap`, `cdflap`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomlap(parlap(lmr))
```

`lmomln3` *L-moments of the 3-Parameter Log-Normal Distribution*

Description

This function estimates the L-moments of the Log-Normal3 distribution given the parameters (ξ, α, and κ) from `parln3`. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \frac{\alpha}{\kappa}(1 - e^{\kappa^2/2}) \text{ and}$$

$$\lambda_2 = \frac{\alpha}{\kappa}(e^{\kappa^2/2})(1 - 2\Phi(-\kappa/\sqrt{2})),$$

where Φ is the cumulative distribution of the standard normal distribution. There are no simple expressions for τ_3, τ_4, and τ_5. Log transformation of the data prior to fitting of the Generalized Normal distribution is not required.

Usage

```
lmomln3(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomln3”.

Author(s)

W.H. Asquith

References

NEED

See Also

parln3, qualn3, cdfln3

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomln3(parln3(lmr))
```

`lmomnor` *L-moments of the Normal Distribution*

Description

This function estimates the L-moments of the Normal distribution given the parameters (μ and σ) from `parnor`. The L-moments in terms of the parameters are

$$\lambda_1 = \mu,$$

$$\lambda_2 = \sqrt{\pi}\sigma,$$

$$\tau_3 = 0,$$

$$\tau_4 = 0.122602, \text{ and}$$

$$\tau_5 = 0.$$

Usage

```
lmomnor(para)
```

Arguments

`para`	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew—analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: "lmomnor".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

parnor, quanor, cdfnor

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomnor(parnor(lmr))
```

---

lmompe3 *L-moments of the Pearson Type III Distribution*

Description

This function estimates the L-moments of the Pearson Type III distribution given the parameters (μ, σ, and γ) from parpe3. The L-moments in terms of the parameters are complicated and solved numerically.

For the implementation in the **lmomco** package, the three parameters are μ, σ, and γ for the mean, standard deviation, and skew, respectively. Therefore, the Pearson Type III distribution is of considerable theoretical interest to this package because the parameters, which are estimated via the L-moments, are in fact the product moments. Although, these values fitted by the method of L-moments will not be numerically equal to the sample product moments. Further details are provided in the examples section of the pmoms function documentation.

Usage

```
lmompe3(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew—analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`source`	An attribute identifying the computational source of the L-moments: "lmompe3".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`parpe3`, `quape3`, `cdfpe3`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmompe3(parpe3(lmr))
```

lmomray — *L-moments of the Rayleigh Distribution*

Description

This function estimates the L-moments of the Rayleigh distribution given the parameters (ξ and α) from parray. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \alpha\sqrt{\pi/2},$$

$$\lambda_2 = \frac{1}{2}\alpha(\sqrt{2}-1)\sqrt{\pi},$$

$$\tau_3 = \frac{1-3/\sqrt{2}+2/\sqrt{3}}{1-1/\sqrt{2}} = 0.1140, \text{ and}$$

$$\tau_4 = \frac{1-6/\sqrt{2}+10/\sqrt{3}-5\sqrt{4}}{1-1/\sqrt{2}} = 0.1054.$$

Usage

```
lmomray(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew–analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomray”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

parray, quaray, cdfray

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomray(parray(lmr))
```

---

lmomRCmark *Sample L-moment for Right-Tail Censoring by a Marking Variable*

Description

Compute the sample L-moments for right-tail censored data set in which censored data values are identified by a marking variable.

Usage

```
lmomRCmark(x, rcmark=NULL, r=1, sort=TRUE)
```

Arguments

x	A vector of data values.
rcmark	The right-tail censoring (upper) marking variable for unknown threshold: 1 is uncensored, 0 is censored.
r	The L-moment order to return, default is the mean.
sort	Does the data need sorting?

Value

An R list is returned.

lambdas	Vector of the L-moments. First element is $\hat{\lambda}_1^{(0,0)}$, second element is $\hat{\lambda}_2^{(0,0)}$, and so on.
ratios	Vector of the L-moment ratios. Second element is $\hat{\tau}^{(0,0)}$, third element is $\hat{\tau}_3^{(0,0)}$ and so on.
trim	Level of symmetrical trimming used in the computation, which will equal NULL if asymmetrical trimming was used.
leftrim	Level of left-tail trimming used in the computation.
rightrim	Level of right-tail trimming used in the computation.
source	An attribute identifying the computational source of the L-moments: "lmomsRCmark".

Author(s)

W.H. Asquith

References

Wang, Dongliang, Hutson, A.D., Miecznikowski, J.C., 2010, L-moment estimation for parametric survival models given censored data: Statistical Methodology, v. 7, no. 6, pp. 655–667.

See Also

lmomsRCmark

Examples

```
# See example under lmomsRCmark
```

---

lmomrevgum *L-moments of the Reverse Gumbel Distribution*

Description

This function estimates the L-moments of the Reverse Gumbel distribution given the parameters (ξ and α) from parrevgum. The type-B L-moments in terms of the parameters are

$$\lambda^B_1 = \xi - (0.5722\ldots)\alpha - \alpha\{\mathrm{Ei}(-\log(1-\zeta))\},$$

$$\lambda^B_2 = \alpha\{\log(2) + \mathrm{Ei}(-2\log(1-\zeta)) - \mathrm{Ei}(-\log(1-\zeta))\},$$

$$\tau_3 = ,$$

$$\tau_4 = \text{, and}$$

$$\tau_5 = .$$

where ζ is the right-tail censoring fraction of the sample or the nonexceedance probability of the right-tail censoring threshold, and $\mathrm{Ei}(x)$ is the exponential integral defined as

$$\mathrm{Ei}(X) = \int_X^{\infty} x^{-1} e^{-x} \mathrm{d}x,$$

where $\mathrm{Ei}(-\log(1-\zeta)) \rightarrow 0$ as $\zeta \rightarrow 1$ and $\mathrm{Ei}(-\log(1-\zeta))$ can not be evaluated as $\zeta \rightarrow 0$.

Usage

```
lmomrevgum(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

`L1`	Arithmetic mean.
`L2`	L-scale—analogous to standard deviation.
`LCV`	coefficient of L-variation—analogous to coe. of variation.
`TAU3`	The third L-moment ratio or L-skew–analogous to skew.
`TAU4`	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
`TAU5`	The fifth L-moment ratio.
`L3`	The third L-moment.
`L4`	The fourth L-moment.
`L5`	The fifth L-moment.
`zeta`	Number of samples observed (noncensored) divided by the total number of samples.
`source`	An attribute identifying the computational source of the L-moments: "lmomrevgum".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

`parrevgum`, `quarevgum`, `cdfrevgum`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
rev.para <- lmom2par(lmr,type='revgum')
lmomrevgum(rev.para)
```

lmomrice *L-moments of the Rice Distribution*

Description

This function estimates the L-moments of the Rice distribution given the parameters (ν and α) from parrice. The L-moments in terms of the parameters are complex. They are computed here by the system of maximum order statistic expectations from the theoLmoms.max.ostat, which uses the expect.max.ostat. The connection between τ_2 and ν/α and a special function (Laguerre polynomial) of ν^2/α^2 and additional algebraic terms is tabulated in the data frame located in `.lmomcohash$RiceTable`. The file 'SysDataBuilder.R' provides additional technical details.

Usage

```
lmomrice(para, ...)
```

Arguments

`para`	The parameters of the distribution.
`...`	Additional arguments passed to the theoLmoms.max.ostat.

Value

An R `list` is returned.

`lambdas`	Vector of the L-moments. First element is λ_1, second element is λ_2, and so on.
`ratios`	Vector of the L-moment ratios. Second element is τ, third element is τ_3 and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` until trimming support is made.
`leftrim`	Level of left-tail trimming used in the computation, which will equal `NULL` until trimming support is made.
`rightrim`	Level of right-tail trimming used in the computation, which will equal `NULL` until trimming support is made.
`source`	An attribute identifying the computational source of the L-moments: "lmomrice", but the exact contents of the remainder of the string might vary as limiting distributions of Normal and Rayleigh can be involved for $\nu/\alpha > 52$ (super high SNR, Normal) or $24 < \nu/\alpha \leq 52$ (high SNR, Normal) or $\nu/\alpha < 0.08$ (very low SNR, Rayleigh).

Author(s)

W.H. Asquith

See Also

parrice, quarice, cdfrice, theoLmoms.max.ostat

Examples

```
rice <- vec2par(c(65,34), type="rice")
lmomrice(rice)

## Not run:
# Use the additional arguments to show how to avoid
# unnecessary overhead when using the Rice, which only
# has two parameters.
  rice <- vec2par(c(15,14), type="rice")
  system.time(lmomrice(rice, nmom=2))
  system.time(lmomrice(rice, nmom=6))


  lcvs <- vector(mode="numeric"); i <- 0
  SNR  <- c(seq(7,0.25, by=-0.25), 0.1)
  for(snr in SNR) {
    i <- i + 1
    rice    <- vec2par(c(10,10/snr), type="rice")
    lcvs[i] <- lmomrice(rice, nmom=2)$ratios[2]
  }
  plot(lcvs, SNR,
       xlab="COEFFICIENT OF L-VARIATION",
       ylab="LOCAL SIGNAL TO NOISE RATIO (NU/ALPHA)")
  lines(.lmomcohash$RiceTable$LCV,
        .lmomcohash$RiceTable$SNR)
  abline(1,0, lty=2)
  mtext("Rice Distribution")
  text(0.15,0.5, "More noise than signal")
  text(0.15,1.5, "More signal than noise")

## End(Not run)
```

`lmoms` *The Sample L-moments and L-moment Ratios*

Description

Compute the sample L-moments for a vector.

Usage

```
lmoms(x,nmom=5)
```

Arguments

x	A vector of data values.
nmom	The number of moments to compute. Default is 5.

Value

An R `list` is returned.

`lambdas`	Vector of the L-moments. First element is $\hat{\lambda}^{(0,0)}_1$, second element is $\hat{\lambda}^{(0,0)}_2$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\hat{\tau}^{(0,0)}$, third element is $\hat{\tau}^{(0,0)}_3$ and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` if asymmetrical trimming was used.
`leftrim`	Level of left-tail trimming used in the computation.
`rightrim`	Level of right-tail trimming used in the computation.
`source`	An attribute identifying the computational source of the L-moments: "lmoms".

Note

This function computes the L-moments through the generalization of the TL-moments (`TLmoms`). In fact, this function calls the default TL-moments with no trimming of the sample. This function is equivalent to `lmom.ub`, but returns a different data structure.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational statistics and data analysis, vol. 43, pp. 299-314.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

See Also

`lmom.ub`, `TLmoms`, and `lmorph`

Examples

```
X1 <- rnorm(30)
L5 <- lmoms(X1)
L4 <- lmoms(X1,nmom=4)
```

lmomsf01 *The Sample L-moments and L-moment Ratios from Nonexceedance Probabilities on the Bounds of 0 to 1*

Description

Compute the sample L-moments for a vector given another vector of nonexceedance probabilities F according to the following

$$\lambda_r = \int_0^1 x(F) P^{\star}_{r-1}(F)\, \mathrm{d}F$$

where $P^{\star}_r(F)$ is

$$P^{\star}_r(F) = \sum_{k=0}^{r} (-1)^{r-k} \binom{r}{k} \binom{r+k}{k} F^k.$$

The quantile function $x(F)$ is approximated by linear interpolation using the `approx` function with flat line (constant) extrapolation to the left and right of the end points of the input vector `x`. The extrapolation however can be controlled by functional arguments.

Usage

```
lmomsf01(x, f=NULL, a=0, nmom=5, subdivisionscale=50,
       sort=TRUE, constantails=TRUE, efs=1e-3)
```

Arguments

`x`	A vector of data values.
`f`	A vector of nonexceedance probabilities for each value in `x`.
`a`	A plotting position coefficient as also used by the `pp` function for estimation of nonexceedance probabilities in `f` if `f=NULL`. The default is for the Weibull plotting positions.
`nmom`	The number of moments to compute. Default is 5.
`subdivisionscale`	A multiplicative factor on the sample size (`length(x)`) for the `subdivision` argument of the `integrate` function. The default value is not optimal but determined by some limited trial and error by the author.
`sort`	Does the data need sorting? The computations require sorted data. This option is provided to optimize processing speed if presorted data already exists.
`constantails`	During the linear interpolation, is `rule=2` of the `approx` function to be used if `constantails=TRUE` or will tail extrapolation using the last left and right tail slopes of the x and F values be used.
`efs`	A near zero value for a difference between the pair of left and right edge F values that are used for computation of the demoninator of a slope computation. The default value is relatively ad hoc determined but provides protection from slopes becoming too large. This argument only has effect if `constantails=FALSE`.

Value

An R `list` is returned.

`lambdas`	Vector of the L-moments. First element is $\hat{\lambda}_1^{(0,0)}$, second element is $\hat{\lambda}_2^{(0,0)}$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\hat{\tau}^{(0,0)}$, third element is $\hat{\tau}_3^{(0,0)}$ and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` if asymmetrical trimming was used.
`leftrim`	Level of left-tail trimming used in the computation.
`rightrim`	Level of right-tail trimming used in the computation.
`source`	An attribute identifying the computational source of the L-moments: "lmomsf".

Note

Trimming of the L-moments is not yet supported.

Author(s)

W.H. Asquith

References

Asquith, W.H., 2011, Distributional analysis with L-moment statistics using the R environment for statistical computing: CreateSpace, ISBN 978–1–46350841–8, 344 p. [eqs. 6.2 and 6.3, p. 91]

See Also

`lmoms`

Examples

```
set.seed(11)
x <- rnorm(35)
lmomsf01(x)$lambdas # linear approx via weibull
lmomsf01(sort(x), f=pp(x))$lambdas # same
lmomsf01(x, f=pp(x, sort=FALSE))$lambdas # same
pwm2lmom(pwm.pp(x, pp=pp(x, sort=FALSE)))$lambdas # directly by weibull
pwm2lmom(pwm.pp(x, B=1))$lambdas # same
lmoms(x)$lambdas # unbiased estimates

## Not run:
  lmr0 <- lmoms(x)
  lmr1 <- lmomsf01(sort(x), f=pp(x))
  lmr2 <- pwm2lmom(pwm.pp(x, B=1))
  Fs <- seq(0.001, 0.999, by=0.001)
  F <- pp(x, sort=FALSE)
  xofF <- approx(x=F, y=x, xout=Fs, rule=2)$y
  plot(qnorm(Fs), xofF, type="l",
```

```
      xlab="STANDARD NORMAL VARIATE",
      ylab="VALUE")
  points(qnorm(F), x)
  lines(qnorm(Fs), qlmomco(Fs, parnor(lmr0)), col=2)
  lines(qnorm(Fs), qlmomco(Fs, parnor(lmr1)))
  lines(qnorm(Fs), qlmomco(Fs, parnor(lmr2)), col=3)

## End(Not run)
```

`lmomsRCmark` *Sample L-moments Moments for Right-Tail Censoring by a Marking Variable*

Description

Compute the sample L-moments for right-tail censored data set in which censored data values are identified by a marking variable. Extension of left-tail censoring can be made using `fliplmoms` and the example therein.

Usage

```
lmomsRCmark(x, rcmark=NULL, nmom=5, sort=TRUE, flip=NA,
               flipfactor=1.1)
```

Arguments

`x`	A vector of data values.
`rcmark`	The right-tail censoring (upper) marking variable for unknown threshold: 0 is uncensored, 1 is censored.
`nmom`	Number of L-moments to return.
`sort`	Does the data need sorting?
`flip`	Does the data require flipping so that left-censored data can be processed as such. If the flip is a logical and TRUE, then flipfactor times the maximum of *x* is used. If the flip is a numeric, then it is used as the flip. The mean is NOT unflipped during the population of this statistics.
`flipfactor`	The value that is greater than 1, which is multiplied on the maximum of *x* to determine the flip, if the flip is not otherwise provided.

Value

An R `list` is returned.

`lambdas`	Vector of the L-moments. First element is $\hat{\lambda}^{(0,0)}_1$, second element is $\hat{\lambda}^{(0,0)}_2$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\hat{\tau}^{(0,0)}$, third element is $\hat{\tau}^{(0,0)}_3$ and so on.

trim	Level of symmetrical trimming used in the computation, which will equal NULL if asymmetrical trimming was used. This is not currently implemented as no one has done the derivations.
leftrim	Level of left-tail trimming used in the computation. This is not currently implemented as no one has done the derivations.
rightrim	Level of right-tail trimming used in the computation. This is not currently implemented as no one has done the derivations.
n	The complete sample size.
n.cen	The number of right-censored data values.
flip	The flip used in the computations for support of left-tail censoring.
source	An attribute identifying the computational source of the L-moments: “lmomsRCmark”.

Author(s)

W.H. Asquith

References

Wang, Dongliang, Hutson, A.D., Miecznikowski, J.C., 2010, L-moment estimation for parametric survival models given censored data: Statistical Methodology, v. 7, no. 6, pp. 655–667.

Helsel, D.R., 2005, Nondetects and data analysis—Statistics for censored environmental data: Hoboken, New Jersey, John Wiley, 250 p.

See Also

lmomRCmark, fliplmoms

Examples

```
# Efron, B., 1988, Logistic regression, survival analysis, and the
# Kaplan-Meier curve: Journal of the American Statistical Association,
# v.83, no.402, pp.414--425
# Survival time measured in days for 51 patients with a marking
# variable in the "time,mark" ensemble. If marking variable is 1,
# then the time is right-censored by an unknown censoring threshold.
Efron <-
c(7,0,  34,0,  42,0,  63,0,  64,0,  74,1,  83,0,  84,0,  91,0,
108,0,  112,0,  129,0,  133,0,  133,0,  139,0,  140,0,  140,0,
146,0,  149,0,  154,0,  157,0,  160,0,  160,0,  165,0,  173,0,
176,0,  185,1,  218,0,  225,0,  241,0,  248,0,  273,0,  277,0,
279,1,  297,0,  319,1,  405,0,  417,0,  420,0,  440,0,  523,1,
523,0,  583,0,  594,0,  1101,0,  1116,1,  1146,0,  1226,1,
1349,1,  1412,1, 1417,1);

# Break up the ensembles into to vectors
ix <- seq(1,length(Efron),by=2)
T  <- Efron[ix]
```

```
Efron.data <- T;
Efron.rcmark <- Efron[(ix+1)]

lmr.RC <- lmomsRCmark(Efron.data, rcmark=Efron.rcmark)
lmr.ub <- lmoms(Efron.data)
lmr.noRC <- lmomsRCmark(Efron.data)
PP <- pp(Efron.data)
plot(PP, Efron.data, col=(Efron.rcmark+1), ylab="DATA")
lines(PP, qlmomco(PP, lmom2par(lmr.noRC, type="kap")), lwd=3, col=8)
lines(PP, qlmomco(PP, lmom2par(lmr.ub, type="kap")))
lines(PP, qlmomco(PP, lmom2par(lmr.RC, type="kap")), lwd=2, col=2)
legend(0,1000,c("uncensored L-moments by indicator (Kappa distribution)",
                "unbiased L-moments (Kappa)",
           "right-censored L-moments by indicator (Kappa distribution)"),
                lwd=c(3,1,2), col=c(8,1,2))
```

---

lmomtexp *L-moments of the Truncated Exponential Distribution*

Description

This function estimates the L-moments of the Truncated Exponential distribution given the parameters (ξ and α) from partexp. The L-moments in terms of the parameters are

$$\lambda_1 = \xi + \alpha,$$

$$\lambda_2 = \alpha/2,$$

$$\tau_3 = 1/3,$$

$$\tau_4 = 1/6, \text{ and}$$

$$\tau_5 = 1/10.$$

Usage

```
lmomtexp(para)
```

Arguments

para	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation–analogous to coe. of variation.

TAU3	The third L-moment ratio or L-skew–analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis–analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: "lmomt-exp".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

partexp, quatexp, cdftexp

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomexp(parexp(lmr))
```

---

lmomTLgld *Trimmed L-moments (t=1) of the Generalized Lambda Distribution*

Description

This function estimates the symmetrical trimmed L-moments (TL-moments) for $t = 1$ of the Generalized Lambda distribution given the parameters (ξ, α, κ, and h) from vec2par. The TL-moments in terms of the parameters are complicated; however, there are analytical solutions. There are no simple expressions of the parameters in terms of the L-moments. The first four TL-moments of the distribution are

$$\lambda_1^{(1)} = \xi + 6\alpha\left(\frac{1}{(\kappa+3)(\kappa+2)} - \frac{1}{(h+3)(h+2)}\right),$$

$$\lambda^{(1)}_2 = 6\alpha\left(\frac{\kappa}{(\kappa+4)(\kappa+3)(\kappa+2)} + \frac{h}{(h+4)(h+3)(h+2)}\right),$$

$$\lambda^{(1)}_3 = \frac{20\alpha}{3}\left(\frac{\kappa(\kappa-1)}{(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} - \frac{h(h-1)}{(h+5)(h+4)(h+3)(h+2)}\right),$$

$$\lambda^{(1)}_4 = \frac{15\alpha}{2}\left(\frac{\kappa(\kappa-2)(\kappa-1)}{(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} + \frac{h(h-2)(h-1)}{(h+6)(h+5)(h+4)(h+3)(h+2)}\right), \text{ and}$$

$$\lambda^{(1)}_5 = \frac{42\alpha}{5}\,(N1 - N2)\,,$$

where

$$N1 = \frac{\kappa(\kappa-3)(\kappa-2)(\kappa-1)}{(\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} \text{ and}$$

$$N2 = \frac{h(h-3)(h-2)(h-1)}{(h+7)(h+6)(h+5)(h+4)(h+3)(h+2)}.$$

The TL-moment ($t=1$) for $\tau^{(1)}_3$ is

$$\tau^{(1)}_3 = \frac{10}{9}\left(\frac{\kappa(\kappa-1)(h+5)(h+4)(h+3)(h+2) - h(h-1)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)}{(\kappa+5)(h+5)\times[\kappa(h+4)(h+3)(h+2) + h(\kappa+4)(\kappa+3)(\kappa+2)]}\right).$$

The TL-moment ($t=1$) for $\tau^{(1)}_4$ is

$$N1 = \kappa(\kappa-2)(\kappa-1)(h+6)(h+5)(h+4)(h+3)(h+2),$$
$$N2 = h(h-2)(h-1)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2),$$
$$D1 = (\kappa+6)(h+6)(\kappa+5)(h+5),$$
$$D2 = [\kappa(h+4)(h+3)(h+2) + h(\kappa+4)(\kappa+3)(\kappa+2)], \text{ and}$$
$$\tau^{(1)}_4 = \frac{5}{4}\left(\frac{N1+N2}{D1\times D2}\right).$$

Finally the TL-moment ($t=1$) for $\tau^{(1)}_5$ is

$$N1 = \kappa(\kappa-3)(\kappa-2)(\kappa-1)(h+7)(h+6)(h+5)(h+4)(h+3)(h+2),$$
$$N2 = h(h-3)(h-2)(h-1)(\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2),$$
$$D1 = (\kappa+7)(h+7)(\kappa+6)(h+6)(\kappa+5)(h+5),$$
$$D2 = [\kappa(h+4)(h+3)(h+2) + h(\kappa+4)(\kappa+3)(\kappa+2)], \text{ and}$$
$$\tau^{(1)}_5 = \frac{7}{5}\left(\frac{N1-N2}{D1\times D2}\right).$$

By inspection the τ_r equations are not applicable for negative integer values $k = \{-2,-3,-4,\dots\}$ and $h = \{-2,-3,-4,\dots\}$ as division by zero will result. There are additional, but difficult to formulate, restrictions on the parameters both to define a valid Generalized Lambda distribution as well as valid L-moments. Verification of the parameters is conducted through `are.pargld.valid`, and verification of the L-moment validity is conducted through `are.lmom.valid`.

Usage

```
lmomTLgld(para, nmom=6, trim=0, leftrim=NULL, rightrim=NULL, tau34=FALSE)
```

Arguments

`para`	The parameters of the distribution.
`nmom`	Number of L-moments to compute.
`trim`	Symmetrical trimming level.
`leftrim`	Left trimming level, t_1.
`rightrim`	Right trimming level, t_2.
`tau34`	A logical controlling the level of L-moments returned by the function. If true, then this function returns only τ_3 and τ_4; this feature might be useful in certain research applications of the generalized lambda distribution associated with the multiple solutions possible for the distribution.

Value

An R `list` is returned.

`lambdas`	Vector of the TL-moments. First element is $\lambda_1^{(t_1,t_2)}$, second element is $\lambda_2^{(t_1,t_2)}$, and so on.
`ratios`	Vector of the TL-moment ratios. Second element is $\tau^{(1)}$, third element is $\tau_3^{(1)}$ and so on.
`trim`	Trim level = left or right values if they are equal
`leftrim`	Left trimming level
`rightrim`	Right trimming level
`source`	An attribute identifying the computational source of the TL-moments: "lmomTL-gld".

Author(s)

W.H. Asquith

Source

Derivations conducted by W.H. Asquith on February 18 and 19, 2006 and others in early March 2011.

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational statistics and data analysis, vol. 43, pp. 299-314.

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105-124.

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions–The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

lmomgld, pargld, cdfgld, quagld

Examples

```
lmomgld(vec2par(c(10,10,0.4,1.3), type='gld'))

## Not run:
PARgld <- vec2par(c(15,12,1,.5), type="gld")
theoTLmoms(PARgld, leftrim=0, rightrim=0, nmom=6)
lmomTLgld(PARgld, leftrim=0, rightrim=0)

theoTLmoms(PARgld, trim=2, nmom=6)
lmomTLgld(PARgld, trim=2)

theoTLmoms(PARgld, trim=3, nmom=6)
lmomTLgld(PARgld, leftrim=3, rightrim=3)

theoTLmoms(PARgld, leftrim=10, rightrim=2, nmom=6)
lmomTLgld(PARgld, leftrim=10, rightrim=2)

## End(Not run)
```

---

lmomTLgpa *Trimmed L-moments of the Generalized Pareto Distribution*

Description

This function estimates the symmetrical trimmed L-moments (TL-moments) for $t = 1$ of the Generalized Pareto distribution given the parameters (ξ, α, and κ) from parTLgpa. The TL-moments in terms of the parameters are

$$\lambda^{(1)}_1 = \xi + \frac{\alpha(\kappa+5)}{(\kappa+3)(\kappa+2)},$$

$$\lambda^{(1)}_2 = \frac{6\alpha}{(\kappa+4)(\kappa+3)(\kappa+2)},$$

$$\tau^{(1)}_3 = \frac{10(1-\kappa)}{9(\kappa+5)}, \text{ and}$$

$$\tau^{(1)}_4 = \frac{5(\kappa-1)(\kappa-2)}{4(\kappa+6)(\kappa+5)}.$$

Usage

```
lmomTLgpa(para)
```

Arguments

`para` The parameters of the distribution.

Value

An R `list` is returned.

`lambdas` Vector of the TL-moments. First element is $\lambda_1^{(1)}$, second element is $\lambda_2^{(1)}$, and so on.

`ratios` Vector of the L-moment ratios. Second element is $\tau^{(1)}$, third element is $\tau_3^{(1)}$ and so on.

`trim` Trim level = 1

`source` An attribute identifying the computational source of the TL-moments: "lmomTL-gpa".

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`parTLgpa`, `quagpa`, `cdfgpa`

Examples

```
TL <- TLmoms(c(123,34,4,654,37,78,21,3400),trim=1)
TL
lmomTLgpa(parTLgpa(TL))
```

lmomwak — *L-moments of the Wakeby Distribution*

Description

This function estimates the L-moments of the Wakeby distribution given the parameters (ξ, α, β, γ, and δ) from parwak. The L-moments in terms of the parameters are complicated and solved numerically.

Usage

```
lmomwak(wakpara)
```

Arguments

wakpara	The parameters of the distribution.

Value

An R `list` is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coe. of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomwak”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

parwak, quawak, cdfwak

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomwak(parwak(lmr))
```

---

lmomwei *L-moments of the Weibull Distribution*

Description

This function estimates the L-moments of the Weibull distribution given the parameters (ζ, β, and δ) from parwei. The Weibull distribution is a reverse Generalized Extreme Value distribution. As result, the Generalized Extreme Value algorithms (lmomgev) are used for computation of the L-moments of the Weibull in this package (see parwei).

Usage

```
lmomwei(para)
```

Arguments

para	The parameters of the distribution.

Value

An R list is returned.

L1	Arithmetic mean.
L2	L-scale—analogous to standard deviation.
LCV	coefficient of L-variation—analogous to coefficient of variation.
TAU3	The third L-moment ratio or L-skew—analogous to skew.
TAU4	The fourth L-moment ratio or L-kurtosis—analogous to kurtosis.
TAU5	The fifth L-moment ratio.
L3	The third L-moment.
L4	The fourth L-moment.
L5	The fifth L-moment.
source	An attribute identifying the computational source of the L-moments: “lmomwei”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

parwei, quawei, cdfwei

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmr
lmomwei(parwei(lmr))
```

---

lmorph *Morph an L-moment Object*

Description

Morph or change one L-moment object type into another. The first L-moment object created for **lmomco** used an R list with named L-moment values such as `L1` or `TAU3`. This object was bounded for L-moment orders less than or equal to five. However, subsequent package development in early 2006 that was related to the trimmed L-moments suggested that an alternative L-moment object structure be used that utilized two vectors for the L-moments and the L-moment ratios. This second object type is not bounded by L-moment order. In turn it became important to seemlessly morph from one object structure to the other and back again. The canonical structure of the first L-moment object type is documented under `lmom.ub`; whereas, the canonical structure for the second L-moment object type is documented under `lmoms` (actually, `TLmoms`). Because the first L-moment object is bounded by five, L-moment order larger than this will be ignored in the morphing process.

Usage

```
lmorph(lmom)
```

Arguments

`lmom`	An L-moment object of type like `lmom.ub` or `lmoms`.

Value

A two different R `lists` (L-moment objects), which are the opposite of the argument type—see the documentation for `lmom.ub` and `lmoms`.

Note

If any of the trimming characteristics of the second type of L-moment object (`trim`, `leftrim`, or `rightrim`) have a greater than zero value, then conversion to the L-moment object with named values will not be performed. A message will be provided that the conversion was not performed.

Author(s)

W.H. Asquith

See Also

lmom.ub, lmoms, TLmoms

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lmorph(lmr)
lmorph(lmorph(lmr))
```

---

lmrdia *L-moment Ratio Diagram Components*

Description

This function returns a list of the L-skew and L-kurtosis (τ_3 and τ_4, respectively) ordinates for construction of L-moment Ratio (L-moment diagrams) that are useful in selecting a distribution to model the data.

Usage

```
lmrdia()
```

Value

An R `list` is returned.

`limits`	The theoretical limits of τ_3 and τ_4–below τ_4 are theoretically not possible.
`aep4`	τ_3 and τ_4 lower limits of the Asymmetric Exponential Power distribution.
`cau`	τ_3 and τ_4 (limiting conditions) of the Cauchy distribution.
`exp`	τ_3 and τ_4 of the Exponential distribution.
`gev`	τ_3 and τ_4 of the Generalized Extreme Value distribution.
`glo`	τ_3 and τ_4 of the Generalized Logistic distribution.
`gpa`	τ_3 and τ_4 of the Generalized Pareto distribution.
`gum`	τ_3 and τ_4 of the Gumbel distribution.
`ray`	τ_3 and τ_4 of the Rayleigh distribution.
`lognormal`	τ_3 and τ_4 of the Lognormal distribution.
`nor`	τ_3 and τ_4 of the Normal distribution.
`pe3`	τ_3 and τ_4 of the Pearson Type III distribution.
`uniform`	τ_3 and τ_4 of the uniform distribution.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`plotlmrdia`

Examples

```
lratios <- lmrdia()
```

`lmrdiscord` *Compute Discordance on L-CV, L-skew, and L-kurtosis*

Description

This function computes the Hosking and Wallis discordancy of the first three L-moment ratios (L-CV, L-skew, and L-kurtosis) according to their implementation in Hosking and Wallis (1997) and earlier. Discordancy triplets of these L-moment ratios is heuristically measured by effectively locating the triplet from the mean center of the 3-dimensional cloud of values. The **lmomRFA** provides for discordancy embedded in the "L-moment method" of regional frequency analysis. The author of **lmomco** chooses to have a separate "high level" implementation for emergent ideas of his in evaluating unusual sample distributions outside of the `regdata` object class envisioned by Hosking in the **lmomRFA** package.

Let $\boldsymbol{\mu_i}$ be a row vector of the values of $\tau_2^{[i]}$, $\tau_3^{[i]}$, $\tau_4^{[i]}$ L-moment ratios for the ith group or site out of n sites. Let $\overline{\boldsymbol{\mu}}$ be a row vector of mean values of all the n sites. Defining a sum of squares and cross products 3-by-3 matrix as

$$\boldsymbol{S} = \sum_{i}^{n} (\boldsymbol{\mu} - \overline{\boldsymbol{\mu}})(\boldsymbol{\mu} - \overline{\boldsymbol{\mu}})^T$$

compute the discorancy of the ith site as

$$D_i = \frac{n}{3} (\boldsymbol{\mu} - \overline{\boldsymbol{\mu}})^T \boldsymbol{S}^{-1} (\boldsymbol{\mu} - \overline{\boldsymbol{\mu}})$$

A site is judged to be discordance if D_i exceeds a critical value. The critical value is a function of sample size. Hosking and Wallis (1997, p. 47) provide a table for general application. By about $n = 14$, the critical value is taken as $D_c = 3$, although the D_{max} increases with sample size. Specifically, the D_i has an upper limit of

$$D_i \leq (n-1)/3$$

However, Hosking and Wallis (1997, p. 47) recommend "that any site with $D_i > 3$ be regarded as discordant." A statistical test of D_i can be constructed. Hosking and Wallis (1997, p. 47) report that the $D_{critical}$ is

$$D_{critical,n,\alpha} = \frac{(n-1)Z}{n-4+3Z}$$

where

$$Z = F(\alpha/n, 3, n-4)$$

upper-tail quantile of the F distribution with degrees of freedom 3 and $n-4$. A table of critical values is preloaded into the `lmrdiscord()` function as this mimics the table of Hosking and Wallis (1997, table 3.1) as a means for cross verification. This table corresponds to the $\alpha = 0.1$ significance.

Usage

```
lmrdiscord(site=NULL, t2=NULL, t3=NULL, t4=NULL,
           Dcrit=NULL, digits=4, lmrdigits=4, sort=TRUE,
           alpha1=0.10, alpha2=0.01, ...)
```

Arguments

- `site` An optional group or site identification; it will be sequenced from 1 to n if `NULL`.
- `t2` L-CV values; emphasis that L-scale is not used.
- `t3` L-skew values.
- `t4` L-kurtosis values.
- `Dcrit` An optional (user specified) critical value for discordance. This value will override the Hosking and Wallis (1997, table 3.1) critical values.
- `digits` The number of digits in rounding operations.
- `lmrdigits` The numer of digits in rounding operation for the echo of the L-moment ratios.
- `sort` A logical on the sort status of the returned data frame.
- `alpha1` A significane level that is greater than `alpha2`.
- `alpha2` A signficance level that is less than `alpha1`.
- `...` Other arguments that might be used. The author added these because it was found that the function was often called by higher level functions that aggregated much of the discordance computations.

Value

An R `data.frame` is returned.

`site`	The group or site identification as used by the function.
`t2`	L-CV values.
`t3`	L-skew values.
`t4`	L-kurtosis.
`Dmax`	The maximum discordancy $D_{max} = (n-1)/3$.
`Dalpha1`	The critical value of D for $\alpha_1 = 0.10$ (default) significance as set by `alpha1` argument.
`Dalpha2`	The critical value of D for $\alpha_2 = 0.01$ (default) significance as set by `alpha1` argument.
`Dcrit`	The critical value of discordancy (user or tabled).
`D`	The discordancy of the L-moment ratios used to trigger the logical in `isD`.
`isD`	Are the L-moment ratios discordant (if starred).
`signif`	A hyphen, star, or double star based on the `Dalpha1` and `Dalpha2` values.

Author(s)

W.H. Asquith

Source

Consultation of the `lmomRFA.f` and `regtst()` function of the **lmomRFA** R package by J.R.M. Hosking. Thanks Jon and Jim Wallis for such a long advocation of the discordancy issue begin at least as early as the 1993 Water Resources Research Paper (-wha).

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmoms`

Examples

```
## Not run:
# This is the canonical test of lmrdiscord().
library(lmomRFA) # Import lmomRFA, needs lmom package too
data(Cascades)   # Extract Hosking's data use in his examples
data <- as.regdata(Cascades) # A "regional" data structure
Dhosking <- sort(regtst(data)$D, decreasing=TRUE) # Discordancy

Dlmomco <- lmrdiscord(site=data$name,
```

```
                       t2=data$t, t3=data$t_3, t4=data$t_4)

Dasquith <- Dlmomco$D
# Now show the site id, and the two discordancy computations
print(data.frame(NAME=data$name, Dhosking=Dhosking,
                                 Dasquith=Dasquith))
# The Dhosking and Dasquith columns had better match!

## End(Not run)

set.seed(3) # This seed produces a "*" and "**", but users
# are strongly encouraged to repeat the folowing code block
# over and over with an unspecified seed and look at the table.
n <- 30 # simulation sample size
par1 <- lmom2par(vec2lmom(c(1, .23, .2, .1)), type="kap")
par2 <- lmom2par(vec2lmom(c(1, .5, -.1)),      type="gev")
name <- t2 <- t3 <- t4 <- vector(mode="numeric")
for(i in 1:20) {
  X <- rlmomco(n, par1); lmr <- lmoms(X)
  t2[i] <- lmr$ratios[2]
  t3[i] <- lmr$ratios[3]
  t4[i] <- lmr$ratios[4]
  name[i] <- "kappa"
}
j <- length(t2)
for(i in 1:3) {
  X <- rlmomco(n, par2); lmr <- lmoms(X)
  t2[j + i] <- lmr$ratios[2]
  t3[j + i] <- lmr$ratios[3]
  t4[j + i] <- lmr$ratios[4]
  name[j + i] <- "gev"
}
D <- lmrdiscord(site=name, t2=t2, t3=t3, t4=t4)
print(D)

plotlmrdia(lmrdia(), xlim=c(-.2,.6), ylim=c(-.1, .4),
           autolegend=TRUE, xleg=0.1, yleg=.4)
points(D$t3,D$t4)
text(D$t3,D$t4,D$site, cex=0.75, pos=3)
text(D$t3,D$t4,D$D, cex=0.75, pos=1)
```

---

nonexceeds *Common Nonexceedance Probabilities*

Description

This function returns a vector nonexceedance probabilities.

Usage

```
nonexceeds()
```

Value

A vector of selected nonexceedance probabilities useful in assessing the "frequency curve" in hydrologic applications (noninclusive). The nonexceedance probabilities extend further into the right-hand tail of the "distribution" to the 0.996 and 0.998 nonexceedance probability values as these are equivalent to the 250- and 500-year events respectively. The relation between annual recurrence interval and nonexceedance probability (when annual data are analyzed) is

$$F = 1 - \frac{1}{T},$$

where T is the T-year event.

Author(s)

W.H. Asquith

See Also

quaexp, quagam, quagev, quaglo, quagno, quagpa, quagum, quakap, quanor, quape3, quawak, and check.fs

Examples

```
lmr <- lmom.ub(rnorm(20))
para <- parnor(lmr)
quanor(nonexceeds(),para)
```

---

par2cdf *Cumulative Distribution Function of the Distributions*

Description

This function acts as a front end of dispatcher to the distribution-specific cumulative distribution functions.

Usage

```
par2cdf(x,para,...)
```

Arguments

x	A real value.
para	The parameters from lmom2par or similar.
...	The additional arguments are passed to the cumulative distribution function such as paracheck=FALSE for the Generalized Lambda distribution (cdfgld).

Value

Nonexceedance probability ($0 \leq F \leq 1$) for x.

Author(s)

W.H. Asquith

See Also

par2qua, lmom2par

Examples

```
lmr       <- lmom.ub(rnorm(20))
para      <- parnor(lmr)
nonexceed <- par2cdf(0,para)
```

---

par2cdf2 *Equivalent Cumulative Distribution Function of Two Distributions*

Description

This function computes the nonexceedance probability of a given quantile from the weighted combination of two quantile functions. The distributions are specified by the two parameter object arguments. The left-tail parameter object is the distribution governing the left tail; the right-tail parameter object is the distribution governing the right tail. The quantile function algebra is

$$Q(F) = (1 - F) \times Q_{lefttail}(F) + F \times Q_{righttail}(F)$$

where $Q(F)$ is the equivalent quantile for nonexceedance probability F computed by the tail weigthing. $Q_{lefttail}(F)$ is the left-tail quantile function; $Q_{righttail}$ is the right-tail quantile function. The par2cdf2 function inverses the above equation for F. Parameter objects are discussed in vec2par and lmom2par functions. If the optional weight argument is provided, then the multiplication of 1-F or F is replaced by 1-weight or weight, respectively. If weight=0, then the quantiles for the right tail are returned, and if weight=1, then the quantiles for the left tail are returned.

A word of caution. The resulting weighted quantile function is not checked for monotonic increase with F. This is a required property of quantile functions.

Usage

```
par2cdf2(x,leftpara,rightpara,weight=NULL,...)
```

Arguments

`x`	A real value.
`leftpara`	The left-tail parameters from `lmom2par` or similar.
`rightpara`	The right-tail parameters from `lmom2par` or similar.
`weight`	An optional weighting argument to use in lieu of `F`.
`...`	The additional arguments are passed to the cumulative distribution function such as `paracheck=FALSE` for the generalized Lambda distribution (`cdfgld`).

Value

Nonexceedance probability ($0 \leq F \leq 1$) for `x` from the two distributions.

Author(s)

W.H. Asquith

See Also

`par2cdf`, `lmom2par`, and `par2qua2`

Examples

```
# Contrived example
lmr       <- lmom.ub(rnorm(20))
leftpara  <- parnor(lmr)
rightpara <- pargev(lmr)
combined.median <- par2qua2(0.5,leftpara,rightpara)
combined.nonexceed <- par2cdf2(combined.median,leftpara,rightpara)
```

---

par2lmom *Convert the Parameters of a Distribution to the L-moments*

Description

This function converts the parameters of a distribution to the L-moment as represented in an L-moment object. This function dispatches to `lmomCCC` where CCC represents the three character distribution identifier: `aep4`, `cau`, `exp`, `gam`, `gev`, `gld`, `glo`, `gno`, `gpa`, `gum`, `kap`, `kur`, `lap`, `ln3`, `nor`, `pe3`, `ray`, `revgum`, `rice`, `texp`, `wak`, and `wei`. The conversion of parameters to TL-moments (`TLmoms`) is not supported. Specific use of functions such as `lmomTLgld` or `lmomTLgpa` for the TL-moments of the Generalized Lambda and Generalized Pareto distributions is required.

Usage

```
par2lmom(para)
```

Arguments

`para`	A parameter object of a distribution.

Value

An L-moment object (an R `list`) is returned.

Author(s)

W.H. Asquith

See Also

`lmom.ub`, `lmom2par`

Examples

```
lmr      <- lmom.ub(rnorm(20))
para     <- parnor(lmr)
frompara <- par2lmom(para)
```

`par2pdf` *Probability Density Function of the Distributions*

Description

This function acts as a front end or dispatcher to the distribution-specific probability density functions.

Usage

```
par2pdf(f,para,...)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `lmom2par` or similar.
`...`	The additional arguments are passed to the quantile function such as `paracheck = FALSE` for the Generalized Lambda distribution (`quagld`).

Value

Quantile value for F.

Author(s)

W.H. Asquith

See Also

par2pdf, par2cdf, lmom2par

Examples

```
lmr     <- lmom.ub(rnorm(20))
para    <- parnor(lmr)
median  <- par2qua(0.5,para)
```

---

par2qua *Quantile Function of the Distributions*

Description

This function acts as a front end or dispatcher to the distribution-specific quantile functions.

Usage

```
par2qua(f,para,...)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from lmom2par or similar.
...	The additional arguments are passed to the quantile function such as paracheck = FALSE for the Generalized Lambda distribution (quagld).

Value

Quantile value for F.

Author(s)

W.H. Asquith

See Also

par2cdf, lmom2par

Examples

```
lmr     <- lmom.ub(rnorm(20))
para    <- parnor(lmr)
median  <- par2qua(0.5,para)
```

par2qua2 *Equivalent Quantile Function of Two Distributions*

Description

This function combines two distributions into one by weighting of the two quantile functions by the nonexceedance probability. The distributions are specified by the parameter arguments. The left-tail parameter object is the distribution governing the left tail; the right-tail parameter object is the distribution governing the right tail. The quantile function algebra is

$$Q(F) = (1 - F) \times Q_{lefttail}(F) + F \times Q_{righttail}(F)$$

where $Q(F)$ is the equivalent quantile for nonexceedance probability F computed by the tail weigthing. $Q_{lefttail}(F)$ is the left-tail quantile function; $Q_{righttail}$ is the right-tail quantile function. Parameter objects are discussed in `vec2par` and `lmom2par` functions. If the optional `weight` argument is provided, then the multiplication of `1-F` or `F` is replaced by `1-weight` or `weight`, respectively. If `weight=0`, then the quantiles for the right tail are returned, and if `weight=1`, then the quantiles for the left tail are returned.

A word of caution. The resulting weighted quantile function is not checked for monotonic increase with F. This is a required property of quantile functions.

Usage

```
par2qua2(f,leftpara,rightpara,weight=NULL,...)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`leftpara`	The left-tail parameters from `lmom2par` or similar.
`rightpara`	The right-tail parameters from `lmom2par` or similar.
`weight`	An optional weighting argument to use in lieu of `F`.
`...`	The additional arguments are passed to the quantile function such as `paracheck = FALSE` for the generalized Lambda distribution (`quagld`).

Value

Quantile value for F from the two distributions.

Author(s)

W.H. Asquith

See Also

`par2qua`, `lmom2par`, and `par2cdf2`

Examples

```
# Contrived example
lmr       <- lmom.ub(rnorm(20))
leftpara  <- parnor(lmr)
rightpara <- pargev(lmr)
combined.median <- par2qua2(0.5,leftpara,rightpara)

# Bigger example--using Kappa fit to whole sample
# for the right tail and Normal fit to whole sample
# for the left tail
D   <- c(123,523,345,356,2134,345,2365,235,12,235,61)
LM  <- lmom.ub(D)
KAP <- parkap(LM)
NOR <- parnor(LM)
PP  <- pp(D)
plot(PP,sort(D),ylim=c(-500,2300))
lines(PP,par2qua(PP,KAP),col=2)
lines(PP,par2qua(PP,NOR),col=3)
lines(PP,par2qua2(PP,NOR,KAP))
```

---

par2vec — *Convert a Parameter Object to a Vector of Parameters*

Description

This function converts a parameter object to a vector of parameters using the `$para` component of the parameter list such as returned by `vec2par`.

Usage

```
par2vec(para, ...)
```

Arguments

`para`	A parameter object of a distribution.
`...`	Additional arguments should they even be needed.

Value

An R `vector` is returned in moment order.

Author(s)

W.H. Asquith

See Also

`vec2par`

Examples

```
para <- vec2par(c(12,123,0.5), type="gev")
par2vec(para)
#   xi alpha kappa
# 12.0 123.0   0.5
```

---

paraep4 — *Estimate the Parameters of the 4-p Asymmetric Exponential Power Distribution*

Description

This function estimates the parameters of the 4-parameter Asymmetric Exponential Power distribution given the L-moments of the data in an L-moment object such as that returned by `lmoms`. The relation between distribution parameters and L-moments is seen under `lmomaep4`. Relatively straightforward, but difficult to numerically achieve, optimization is needed to extract the parameters from the L-moments.

Delicado and Goria (2008) make argue for numerical methods to use the following objective function

$$\epsilon(\alpha,\kappa,h) = \log(1 + \sum_{r=2}^{4}(\hat{\lambda}_r - \lambda_r)^2)$$

and subsequently solve directly for ξ.

This objective function was chosen by Delicado and Goria because the solution surface can become quite flat for away from the minimum. The author of **lmomco** agrees with the findings of those authors from limited exploratory analysis and the development of the algorithms used here under the rubic of the "DG" method. This exploration resulted in an alternative algorithm using tabulated initial guesses described below. An evident drawback of the Delicado-Goria algorithm, is that precision in α is may be lost according to the observation that this parameter can be analytically computed given λ_2, κ, and h.

It is established practice in L-moment theory of four (and similarly three) parameter distributions to see expressions for τ_3 and τ_4 used for numerical optimization to obtain the two higher parameters (α and h) first and then see analytical expressions directly compute the two lower parameters (ξ and α). The author made various exploratory studies by optimizing on τ_3 and τ_4 through a least squares objective function. Such a practice seems to perform acceptably when compared to that recommended by Delicado and Goria (2008) when the initial guesses for the parameters are drawn from pretabulation of the relation between $\{\alpha, h\}$ and $\{\tau_3, \tau_4\}$.

Another optimization, referred to here as the "A" method, is available for parameter estimation using the following objective function

$$\epsilon(\kappa,h) = \sqrt{(\hat{\tau}_3 - \tau_3)^2 + (\hat{\tau}_4 - \tau_4)^2}$$

and subsequently solve directly for α and then ξ. The "A" method appears to perform slightly better in κ and h estimation and quite a bit better in α and and ξ as seemingly expected because these last two are analytically computed. The objective function of the A method defaults to use of the $\sqrt{x}$ but this can be removed by setting `sqrt.t3t4=FALSE`.

The initial guesses for the κ and h parameters derives from a hashed environment in in file '`sysdata.rda`' (`.lmomcohash$AEPkh2lmrTable`) in which the $\{\kappa, h\}$ pair having the minimum $\epsilon(\kappa, h)$ in which τ_3 and τ_4 derive from the table as well. The file '`SysDataBuilder.R`' provides additional technical details on how the `AEPkh2lmrTable` was generated.

The table represents a systematic double-loop sweep through `lmomaep4` for

$$-3 \leq \log(\kappa) \leq 3, \Delta = 0.05$$

and

$$-3 \leq \log(h) \leq 3, \Delta = 0.05$$

The function will not return parameters if the following lower bounds of τ_4 is not met: $\tau_4 \geq 0.77555(|\tau_3|) - 3.3355(|\tau_3|)^2 + 14.196(|\tau_3|)^3 - 29.909(|\tau_3|)^4 + 37.214(|\tau_3|)^5 - 24.741(|\tau_3|)^6 + 6.7998(|\tau_3|)^7$. For this polynomial, the residual standard error is RSE = 0.0003125 and the maximum absolute error for τ_3:$[0, 1] < 0.0015$. The actual coefficients in `paraep4` have additional significant figures.

Usage

```
paraep4(lmom, checklmom=TRUE,
        method=c("A", "DG", "ADG"),
        sqrt.t3t4=TRUE, eps=1e-4,
        checkbounds=TRUE, kapapproved=TRUE,
        A.guess=NULL, K.guess=NULL, H.guess=NULL)
```

Arguments

`lmom` A L-moment object created by `lmoms` or `pwm2lmom`.

`checklmom` Should the L-moments be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

`method` Which method for parameter estimation should be used. The "A" or "DG" methods. The "ADG" method will run both methods and retains the salient optimization results of each but the official parameters in `para` are those from the "A" method. Lastly, all minimization is based on the `optim` function using the Nelder-Mead method and default arguments.

`sqrt.t3t4` If true and the method is "A", then the square root of the sum of square errors in τ_3 and τ_4 are used instead of sum of square differences alone.

`eps` A small term or threshold for which the square root of the sum of square errors in τ_3 and τ_4 is compared to to judge "good enough" for the alogrithm to set the `ifail` on return in addition to convergence flags coming from the `optim` function. Note that eps is only used if the "A" or "ADG" methods are triggered because the other method uses the scale parameter which in reality could be quite large relative to the other 2 shape parameters, and a reasonable default for such a secondary error threshold check would be ambiguous.

checkbounds — Should the lower bounds of τ_4 be verified and if sample $\hat{\tau}_3$ and $\hat{\tau}_4$ are outside of these bounds, then NA are returned for the solutions.

kapapproved — Should the Kappa distribution be fit by parkap if $\hat{\tau}_4$ is below the lower bounds of τ_4? This fitting is only possible if checkbounds is true. The Kappa and AEP4 overlap partially. The AEP4 extends τ_4 above Generalized Logistic and Kappa extends τ_4 below the lower bounds of τ_4 for AEP4 and extends all the way to the theoretical limits as used within are.lmom.valid.

A.guess — A user specified guess of the α parameter to provide to the optimization of any of the methods. This argument just superceeds the simple initial guess of $\alpha = 1$.

K.guess — A user specified guess of the κ parameter to supercede that derived from the .lmomcohash$AEPkh2lmrTable in file 'sysdata.rda'.

H.guess — A user specified guess of the h parameter to supercede that derived from the .lmomcohash$AEPkh2lmrTable in file 'sysdata.rda'.

Value

An R list is returned.

type — The type of distribution: aep4.

para — The parameters of the distribution.

source — The source of the parameters: "paraep4".

ifail — A numeric failure code.

ifailtext — A text message for the failure code.

method — The method as specified by the method.

L234 — Optional and dependent on method "DG" or "ADG". Another R list containing the optimization details by the "DG" method along with the estimated parameters in para_L234. The "_234" is to signify that optimization is being conducted using λ_2, λ_3, and λ_4. The parameter values in para are those only when the "DG" method is used.

T34 — Optional and dependent on method "A" or "ADG". Another R list containing the optimization details by the "A" method along with the estimated parameters in para_T34. The "_T34" is to signify that opimization is being conducted using τ_3 and τ_4 only. The parameter values in para are those by the "A" method.

The values for ifail or produced by three mechanisms. First, the convergence number emanating from the optim function itself. Second, the integer 1 is used when the failure is attributable to the optim function. Second, the interger 2 is a general attempt to have a singular failure by sometype of eps outside of optim. Fourth, the integer 3 is used to show that the parameters fail against a parameter validity check in are.paraep4.valid. And fifth, the integer 4 is used to show that the sample L-moments are below the lower bounds of the τ_4 polynomial shown here.

Additional and self explanatory elements on the returned list will be present if the Kappa distribution was fit instead.

Author(s)

W.H. Asquith

References

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

Asquith, W.H., 2012, Parameter Estimation for the 4-Parameter Asymmetric Exponential Power Distribution by the Method of L-moments using R, A Journal Article in Progress.

See Also

lmomaep4, cdfaep4, quaaep4

Examples

```
PAR <- list(para=c(100, 1000, 1.7, 1.4), type="aep4");
lmr <- lmomaep4(PAR)

PARdg  <- paraep4(lmr, method="DG")
PARasq <- paraep4(lmr, method="A")

# print(PARdg); print(PARasq)

F <- c(0.001, 0.005, seq(0.01,0.99, by=0.01), 0.995, 0.999)
qF <- qnorm(F)
ylim <- range( quaaep4(F, PAR), quaaep4(F, PARdg), quaaep4(F, PARasq) )
plot(qF, quaaep4(F, PARdg), type="n", ylim=ylim,
     xlab="STANDARD NORMAL VARIATE", ylab="QUANTILE")
lines(qF, quaaep4(F, PAR), col=8, lwd=10) # the true curve
lines(qF, quaaep4(F, PARdg),  col=2, lwd=3)
lines(qF, quaaep4(F, PARasq), col=3, lwd=2, lty=2)
# See how the red curve deviates, Delicado-Goria failed
# and the ifail attribute in PARdg is TRUE.

print(PAR$para)
print(PARdg$para)
print(PARasq$para)

ePAR1dg <- abs((PAR$para[1] - PARdg$para[1])/PAR$para[1])
ePAR2dg <- abs((PAR$para[2] - PARdg$para[2])/PAR$para[2])
ePAR3dg <- abs((PAR$para[3] - PARdg$para[3])/PAR$para[3])
ePAR4dg <- abs((PAR$para[4] - PARdg$para[4])/PAR$para[4])

ePAR1asq <- abs((PAR$para[1] - PARasq$para[1])/PAR$para[1])
ePAR2asq <- abs((PAR$para[2] - PARasq$para[2])/PAR$para[2])
ePAR3asq <- abs((PAR$para[3] - PARasq$para[3])/PAR$para[3])
ePAR4asq <- abs((PAR$para[4] - PARasq$para[4])/PAR$para[4])

MADdg  <- mean(ePAR1dg,  ePAR2dg,  ePAR3dg,  ePAR4dg)
```

```
MADasq <- mean(ePAR1asq, ePAR2asq, ePAR3asq, ePAR4asq)

# We see that the Asquith method performs better for the example
# parameters in PAR and inspection of the graphic will show that
# the Delicado-Goria solution is obviously off.
print(MADdg)
print(MADasq)

# Repeat the above with this change in parameter to
# PAR <- list(para=c(100, 1000, .7, 1.4), type="aep4")
# and the user will see that all three methods converged on the
# correct values.
```

---

parcau *Estimate the Parameters of the Cauchy Distribution*

Description

This function estimates the parameters of the Cauchy distribution from the trimmed L-moments (TL-moments) having trim level 1.

Usage

```
parcau(lmom)
```

Arguments

`lmom`	TL-moments from `TLmoms` with `trim=1`.

Details

Unlike many of the other distributions in this package, the parameter estimation occurs by passing the data into the function and not from passing of an L-moment object (see `lmom.ub`). Contrast this practice with `pargum` for example.) The reason this is so is because the usual L-moments are undefined for the Cauchy distribution, but the trimmed L-moments with a symmetrical trimming parameter are defined. Specifically, the L-moments by trimming the smallest and largest order statistic expections of the Cauchy are defined by Elamir and Seheult (2003). The function `parcau` calls `TLlmoms(x,trim=1))` internally to compute the trimmed L-moments. The relation between the parameters and the trimmed L-moments is

$$\xi = \lambda^{(1)}_1 \text{and}$$

$$\alpha = \frac{\lambda^{(1)}_2}{0.698}.$$

Value

An R `list` is returned.

`type`	The type of distribution: `cau`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "parcau".

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Gilchrist, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

`TLmoms`, `lmomcau`

Examples

```
X1 <- rcauchy(20)
parcau(TLmoms(X1,trim=1))
```

`parexp` *Estimate the Parameters of the Exponential Distribution*

Description

This function estimates the parameters of the Exponential distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomexp`.

Usage

```
parexp(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `exp`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “parexp”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomexp`, `cdfexp`, `quaexp`

Examples

```
lmr <- lmom.ub(rnorm(20))
parexp(lmr)
```

---

pargam — *Estimate the Parameters of the Gamma Distribution*

Description

This function estimates the parameters of the Gamma distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomgam`. The parameters can not be estimated if $\lambda_1 \leq \lambda_2$ or $\lambda_2 \leq 0$.

Usage

```
pargam(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `gam`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "pargam".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomgam`, `cdfgam`, `quagam`

Examples

```
lmr <- lmom.ub(rnorm(20))
pargam(lmr)
```

pargev *Estimate the Parameters of the Generalized Extreme Value Distribution*

Description

This function estimates the parameters of the Generalized Extreme Value distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomgev`.

Usage

```
pargev(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `gev`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "pargev".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomgev`, `cdfgev`, `quagev`

Examples

```
lmr <- lmom.ub(rnorm(20))
pargev(lmr)
```

---

pargld *Estimate the Parameters of the Generalized Lambda Distribution*

Description

This function estimates the parameters of the Generalized Lambda distribution given the L-moments of the data in an ordinary L-moment object (`lmom.ub` or a trimmed L-moment object (`TLmoms` for `t=1`. The relation between distribution parameters and L-moments is seen under `lmomgld`. There are no simple expressions for the parameters in terms of the L-moments. This function is experimental and general details of the algorithm are provided below. Further, consider that multiple parameter solutions are possible with the Generalized Lambda so some expertise in the distribution and other aspects are needed.

Usage

```
pargld(lmom, verbose=FALSE, initkh=NULL, eps=1e-3,
       aux=c("tau5", "tau6"), checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub`, `pwm2lmom`, or `TLmoms` with `trim=0`.
`verbose`	A logical switch on the verbosity of output. Default is `verbose=FALSE`.
`initkh`	A vector of the initial guess of the κ and h parameters. No other regions of parameter space are consulted.
`eps`	A small term or threshold for which the square root of the sum of square errors in τ_3 and τ_4 is compared to to judge "good enough" for the alogrithm to order solutions based on smallest error as explained in next argument.
`aux`	Control the algorithm to order solutions based on smallest error in $\Delta\tau_5$ or $\Delta\tau_6$.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Details

Karian and Dudewicz (2000) summarize six regions of the κ and h space in which the Generalized Lambda distribution is valid for suitably choosen α. Numerical experimentation suggestions that the L-moments are not valid in Regions 1 and 2. However, initial guesses of the parameters within each region are used for numerous separate `optim` (the R function) efforts to perform a least sum-of-square errors on the following objective function.

$$(\hat{\tau}_3 - \tilde{\tau}_3)^2 + (\hat{\tau}_4 - \tilde{\tau}_4)^2,$$

where $\hat{\tau}_r$ is the L-moment ratio of the data, $\tilde{\tau}_r$ is the estimated value of the L-moment ratio for the fitted distribution κ and h and τ_r is the actual value of the L-moment ratio.

For each optimization a check on the validity of the parameters so produced is made–are the parameters consistent with the Generalized Lambda distribution and a second check is made on the validity of τ_3 and τ_4. If both validity checks return `TRUE` then the optimization is retained if its sum-of-square error is less than the previous optimum value. It is possible for a given solution to be found outside the starting region of the initial guesses. The surface generated by the τ_3 and τ_4 equations seen in `lmomgld` is complex–different initial guesses within a given region can yield what appear to be radically different κ and h. Users are encouraged to "play" with alternative solutions (see the `verbose` argument). A quick double check on the L-moments from the solved parameters using `lmomgld` is encouraged as well. Karvanen and others (2002, eq. 25) provide an equation expressing κ and h as equal (a symmetrical Generalized Lambda distribution) in terms of τ_4 and suggest that the equation be used to determine initial values for the parameters. This equation is used on an experimental basis for the final optimization attempt by this function.

Value

An R `list` is returned if `result='best'`.

`type`	The type of distribution: `gld`.
`para`	The parameters of the distribution.
`delTau5`	Difference between the $\tilde{\tau}_5$ of the fitted distribution and true $\hat{\tau}_5$.
`error`	Smallest sum of square error found.
`source`	The source of the parameters: "pargld".
`rest`	An R `data.frame` of other solutions if found.

The rest of the solutions have the following:

`xi`	The location parameter of the distribution.
`alpha`	The scale parameter of the distribution.
`kappa`	The 1st shape parameter of the distribution.
`h`	The 2nd shape parameter of the distribution.
`attempt`	The attempt number that found valid TL-moments and parameters of GLD.
`delTau5`	The absolute difference between $\hat{\tau}^{(1)}_5$ of data to $\tilde{\tau}^{(1)}_5$ of the fitted distribution.
`error`	The sum of square error found.
`initial_k`	The starting point of the κ parameter.
`initial_h`	The starting point of the h parameter.
`valid.gld`	Logical on validity of the GLD—`TRUE` by this point.
`valid.lmr`	Logical on validity of the L-moments—`TRUE` by this point.
`lowerror`	Logical on whether error was less than `eps`—`TRUE` by this point.

Note

Although the underlying derivations of the L-moments in terms of the parameters are sound and this function uses built-in functions of R for the optimization, this function should be considered experimental until further notice. The "preferred" solution might not be appropriate. This function is not for production code. This is a cumbersome method of parameter solution and enhancements or total redesign of the algorithm is expected.

Author(s)

W.H. Asquith

Source

W.H. Asquith in Feb. 2006 with copy of Karian and Dudewicz (2000) and again Feb. 2011.

References

Karvanen, J., Eriksson, J., and Koivunen, V., 2002, Adaptive score functions for maximum likelihood ICA: Journal of VLSI Signal Processing, vol. 32, p. 82–92.

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

lmom.ub, lmomgld, cdfgld, quagld, parTLgld

Examples

```
X <- rgamma(202,2) # simulate a skewed distribution
lmr <- lmoms(X) # compute trimmed L-moments
PARgld <- pargld(lmr) # fit the GLD

F <- pp(X)
plot(F,sort(X), col=8, cex=0.25)
lines(F, qlmomco(F,PARgld)) # show the best estimate
if(! is.null(PARgld$rest)) { #$
  n <- length(PARgld$rest$xi)
  other <- unlist(PARgld$rest[n,1:4]) #$ # show alternative
  lines(F, qlmomco(F,vec2par(other, type="gld")), col=2)
}
# Note in the extraction of other solutions that no testing for whether
# additional solutions were found is made. Also, it is quite possible
# that the other solutions "[n,1:4]" is effectively another numerical
# convergence on the primary solution. Some users of this example thus
# might not see two separate lines. Users are encouraged to inspect the
# rest of the solutions: print(PARgld$rest); #$

# For one run of the above example, the GLD results follow
#print(PARgld)
#$type
#[1] "gld"
```

```
#$para
#      xi    alpha    kappa        h
#3.144379 2.943327 7.420334 1.050792
#$delTau5
#[1] -0.0367435
#$error
#[1] 5.448016e-10
#$source
#[1] "pargld"
#$rest
#         xi    alpha       kappa         h attempt     delTau5        error
#1 3.1446434 2.943469 7.421131671 1.0505370      14 -0.03675376 6.394270e-10
#2 0.4962471 8.794038 0.008295796 0.2283519       4 -0.04602541 8.921139e-10


## Not run:
F <- seq(.01,.99,.01)
plot(F,qlmomco(F, vec2par(c(3.1446434, 2.943469,
                            7.4211316, 1.050537), type="gld")),
     type="l")
lines(F,qlmomco(F, vec2par(c(0.4962471, 8.794038,
                             0.0082958, 0.228352), type="gld")))

## End(Not run)
```

parglo *Estimate the Parameters of the Generalized Logistic Distribution*

Description

This function estimates the parameters of the Generalized Logistic distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomglo`.

Usage

```
parglo(lmom,checklmom=TRUE)
```

Arguments

`lmom` A L-moment object created by `lmom.ub` or `pwm2lmom`.

`checklmom` Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `glo`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "parglo".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomglo`, `cdfglo`, `quaglo`

Examples

```
lmr <- lmom.ub(rnorm(20))
parglo(lmr)
```

`pargno` *Estimate the Parameters of the Generalized Normal Distribution*

Description

This function estimates the parameters of the Generalized Normal (log-Normal) distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomgno`.

Usage

```
pargno(lmom,checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R list is returned.

type	The type of distribution: gno.
para	The parameters of the distribution.
source	The source of the parameters: "pargno".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmomgno, cdfgno, quagno

Examples

```
lmr <- lmom.ub(rnorm(20))
pargno(lmr)
```

pargpa	*Estimate the Parameters of the Generalized Pareto Distribution*

Description

This function estimates the parameters of the Generalized Pareto distribution given the L-moments of the data in an ordinary L-moment object (`lmom.ub` or a trimmed L-moment object (`TLmoms` for `t=1`. The relation between distribution parameters and L-moments is seen under `lmomgpa` or `lmomgpa2`.

Usage

```
pargpa(lmom,zeta=1,xi=NULL,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub`, `pwm2lmom`, or `TLmoms` with `trim=0`.
`zeta`	The right censoring fraction. If less than unity then a dispatch to the `pargpaRC` is made and the `lmom` argument must contain the B-type L-moments. If the data is not right, then this value must be left alone to the default of unity.
`xi`	The lower limit of the distribution. If ξ is known then alternative algorithms are used.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `gpa`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “pargpa”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmomgpa, cdfgpa, quagpa

Examples

```
X   <- rexp(200)
lmr <- lmom.ub(X)
P1  <- pargpa(lmr)
P2  <- pargpa(lmr, xi=0.25)

## Not run:
F <- nonexceeds()
plot(pp(X), sort(X))
lines(F,quagpa(F,P1))  # black line
lines(F, quagpa(F,P2), col=2)   # red line

## End(Not run)
```

---

pargpaRC *Estimate the Parameters of the Generalized Pareto Distribution with Right-Tail Censoring*

Description

This function estimates the parameters (ξ, α, and κ) of the Generalized Pareto distribution given the "B"-type L-moments (through the B-type Probability-Weighted Moments) of the data under right censoring conditions (see pwmRC). The "B"-type L-moments in terms of the parameters are

$$\lambda^B_1 = \xi + \alpha m_1,$$
$$\lambda^B_2 = \alpha(m_1 - m_2),$$
$$\lambda^B_3 = \alpha(m_1 - 3m_2 + 2m_3),$$
$$\lambda^B_4 = \alpha(m_1 - 6m_2 + 10m_3 - 5m_4), \text{ and}$$
$$\lambda^B_5 = \alpha(m_1 - 10m_2 + 30m_3 - 35m_4 + 14m_5),$$

where $m_r = \{1-(1-\zeta)^{r+\kappa}\}/(r+\kappa)$ and ζ is the right-tail censor fraction or the probability $\mathrm{Pr}\{\}$ that x is less than the quantile at ζ nonexceedance probability: ($\mathrm{Pr}\{x < X(\zeta)\}$). Finally, the RC in the function name is to denote Right-tail Censoring.

Usage

```
pargpaRC(lmom,zeta=1,xi=NULL,lower=-1,upper=20,checklmom=TRUE)
```

Arguments

`lmom`	A B-type L-moment object created by a function such as `pwm2lmom` from B-type Probability-Weighted Moments from `pwmRC`.
`zeta`	The censoring fraction. The number of samples observed (noncensored) divided by the total number of samples.
`xi`	The lower limit of the distribution. If ξ is known then alternative algorithms are used.
`lower`	The lower value for κ for a call to the `optimize` function. For the L-moments of the distribution to be valid $\kappa > -1$.
`upper`	The upper value for κ for a call to the `optimize` function. Hopefully, a large enough default is chosen for real-world data sets.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Details

The `optimize` function is used to numerically solve for the shape parameter κ. No test or evaluation is made on the quality of the minimization. Users should consult the contents of the `optim` portion of the returned list. Finally, this function should return the same parameters if $\zeta = 1$ as the `pargpa` function.

Value

An R `list` is returned.

`type`	The type of distribution: `gpa`.
`para`	The parameters of the distribution.
`zeta`	The right-tail censoring fraction.
`source`	The source of the parameters: "pargpaRC".
`optim`	The `list` returned by the `optimize` function.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

lmomgpa, lmomgpaRC, pargpa, cdfgpa, quagpa

Examples

```
n         <- 60 # samplesize
para      <- vec2par(c(1500,160,.3),type="gpa") # build a GPA parameter set
fakedata  <- quagpa(runif(n),para) # generate n simulated values
threshold <- 1700 # a threshold to apply the simulated censoring
fakedata  <- sapply(fakedata,function(x) { if(x > threshold)
                                          return(threshold) else return(x)
                                        })
lmr       <- lmoms(fakedata) # Ordinary L-moments without considering
                             # that the data is censored
estpara   <- pargpa(lmr) # Estimated parameters of parent

pwm2     <- pwmRC(fakedata,threshold=threshold) # compute censored PWMs
typeBpwm <- pwm2$Bbetas # the B-type PWMs
zeta     <- pwm2$zeta # the censoring fraction

cenpara <- pargpaRC(pwm2lmom(typeBpwm),zeta=zeta) # Estimated parameters
F       <- nonexceeds() # nonexceedance probabilities for plotting purposes

# Visualize some data
plot(F,quagpa(F,para), type='l', lwd=3) # The true distribution
lines(F,quagpa(F,estpara), col=3) # Green estimated in the ordinary fashion
lines(F,quagpa(F,cenpara), col=2) # Red, consider that the data is censored
# now add in what the drawn sample looks like.
PP <- pp(fakedata) # plotting positions of the data
points(PP,sort(fakedata)) # sorting is needed!
# Interpretation. You should see that the red line more closely matches
# the heavy black line. The green line should be deflected to the right
# and pass through the values equal to the threshold, which reflects the
# much smaller L-skew of the ordinary L-moments compared to the type-B
# L-moments.


# Assertion, given some PWMs or L-moments, if zeta=1 then the parameter
# estimates must be identical. The following provides a demonstration.
para1 <- pargpaRC(pwm2lmom(typeBpwm),zeta=1)
para2 <- pargpa(pwm2lmom(typeBpwm))
str(para1)
str(para2)
```

```
# Assertion as previous assertion, let us trigger different optimizer
# algorithms with a non-NULL xi parameter and see if the two parameter
# lists are the same.
para1 <- pargpaRC(pwm2lmom(typeBpwm), zeta=zeta)
para2 <- pargpaRC(pwm2lmom(typeBpwm), xi=para1$para[1], zeta=zeta)
str(para1)
str(para2)
```

---

pargum *Estimate the Parameters of the Gumbel Distribution*

Description

This function estimates the parameters of the Gumbel distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomgum`.

Usage

```
pargum(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `gum`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "pargum".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomgum`, `cdfgum`, `quagum`

Examples

```
lmr <- lmom.ub(rnorm(20))
pargum(lmr)
```

`parkap` *Estimate the Parameters of the Kappa Distribution*

Description

This function estimates the parameters of the Kappa distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomkap`.

Usage

```
parkap(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `kap`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "parkap".

support	The support (or range) of the fitted distribution.
ifail	A numeric failure code.
ifailtext	A text message for the failure code.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmomkap, cdfkap, quakap

Examples

```
lmr <- lmom.ub(rnorm(20))
parkap(lmr)
```

---

parkur *Estimate the Parameters of the Kumaraswamy Distribution*

Description

This function estimates the parameters of the Kumaraswamy distribution given the L-moments of the data in an L-moment object such as that returned by lmom.ub. The relation between distribution parameters and L-moments is seen under lmomkur.

Usage

```
parkur(lmom,checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `kur`.
`para`	The parameters of the distribution.
`err`	The convergence error.
`convergence`	Logical showing whether error convergence occurred.
`source`	The source of the parameters: “parkur”.

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy’s distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

`lmom.ub`, `lmomkur`, `cdfgum`, `quakur`

Examples

```
lmr <- lmom.ub(runif(20)^2)
parkur(lmr)

kurpar <- list(para=c(1,1), type="kur");
lmr <- lmomkur(kurpar)
parkur(lmr)

kurpar <- list(para=c(0.1,1), type="kur");
lmr <- lmomkur(kurpar)
parkur(lmr)

kurpar <- list(para=c(1,0.1), type="kur");
lmr <- lmomkur(kurpar)
parkur(lmr)

kurpar <- list(para=c(0.1,0.1), type="kur");
lmr <- lmomkur(kurpar)
parkur(lmr)
```

parlap *Estimate the Parameters of the Laplace Distribution*

Description

This function estimates the parameters of the Generalized Logistic distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and sample L-moments using the best available combination of the lower L-moments are

$$\xi = \lambda_1 - 50/31 \times \lambda_3, \text{ and}$$

$$\alpha = 1.4741\lambda_2 - 0.5960\lambda_4.$$

Although by simplist definition, one would have $\xi = \lambda_1$ and $\alpha = \frac{4}{3}\lambda_2$.

Usage

```
parlap(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `lap`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “parlap”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

`lmom.ub`, `lmomlap`, `cdflap`, `qualap`

Examples

```
lmr <- lmom.ub(rnorm(20))
parlap(lmr)
```

---

parln3 *Estimate the Parameters of the 3-Parameter Log-Normal Distribution*

Description

This function estimates the parameters of the Log-Normal3 distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomln3`.

Usage

```
parln3(lmom, zeta=NULL, checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`zeta`	Lower bounds, if `NULL` then solved for.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `ln3`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "parln3".

Author(s)

W.H. Asquith

References

NEED

See Also

`lmom.ub`, `lmomln3`, `cdfln3`, `qualn3`

Examples

```
lmr <- lmom.ub(rnorm(20))
parln3(lmr)
```

---

parnor *Estimate the Parameters of the Normal Distribution*

Description

This function estimates the parameters of the Normal distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomnor`. There are interesting parallels between λ_2 (L-scale) and σ (standard deviation). The σ estimated from this function will not necessarily equal the output of the `sd()` function of R. See the extended example for further illustration.

Usage

```
parnor(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `nor`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: "parnor".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmomnor, cdfnor, quanor

Examples

```
lmr <- lmom.ub(rnorm(20))
parnor(lmr)

# A more extended example to explore the differences between an
# L-moment derived estimate of the standard deviation and R's sd()
true.std <- 15000 # select a large standard deviation
std         <- vector(mode = "numeric") # vector of sd()
std.by.lmom <- vector(mode = "numeric") # vector of L-scale values
sam <- 7   # number of samples to simulate
sim <- 100 # perform simulation sim times
for(i in seq(1,sim)) {
  Q <- rnorm(sam,sd=15000) # draw random normal deviates
  std[i] <- sd(Q) # compute standard deviation
  lmr <- lmoms(Q) # compute the L-moments
  std.by.lmom[i] <- lmr$lambdas[2] # save the L-scale value
}
# convert L-scale values to equivalent standard deviations
std.by.lmom      <- sqrt(pi)*std.by.lmom

# compute the two biases and then output
# see how the standard deviation estimated through L-scale
# has a smaller bias than the usual (product moment) standard
# deviation. The unbiasness of L-moments is demonstrated.
std.bias         <- true.std - mean(std)
std.by.lmom.bias <- true.std - mean(std.by.lmom)
cat(c(std.bias,std.by.lmom.bias,"\n"))
```

---

parpe3 *Estimate the Parameters of the Pearson Type III Distribution*

Description

This function estimates the parameters of the Pearson Type III distribution given the L-moments of the data in an L-moment object such as that returned by lmom.ub. The L-moments in terms of the parameters are complicated and solved numerically.

For the implementation in the **lmomco** package, the three parameters are μ, σ, and γ for the mean, standard deviation, and skew, respectively. Therefore, the Pearson Type III distribution is of considerable theoretical interest to this package because the parameters, which are estimated via the L-moments, are in fact the product moments. Although, these values fitted by the method of L-moments will not be numerically equal to the sample product moments. Further details are provided in the examples section of the pmoms function documentation.

Usage

```
parpe3(lmom,checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R list is returned.

type	The type of distribution: pe3.
para	The parameters of the distribution.
source	The source of the parameters: "parpe3".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmompe3, cdfpe3, quape3

Examples

```
lmr <- lmom.ub(rnorm(20))
parpe3(lmr)
```

parray *Estimate the Parameters of the Rayleigh Distribution*

Description

This function estimates the parameters of the Rayleigh distribution given the L-moments of the data in an L-moment object such as that returned by lmom.ub. The relations between distribution parameters and L-moments are

$$\alpha = \frac{2\lambda_2\sqrt{\pi}}{\sqrt{2}-1},$$

and

$$\xi = \lambda_1 - \alpha\sqrt{\pi/2}.$$

Usage

```
parray(lmom, xi=NULL, checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
xi	The lower limit of the distribution. If ξ is known then alternative algorithms are used.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R list is returned.

type	The type of distribution: ray.
para	The parameters of the distribution.
source	The source of the parameters: "parray".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

lmom.ub, lmomray, cdfray, quaray

Examples

```
lmr <- lmom.ub(rnorm(20))
parray(lmr)
```

---

parrevgum *Estimate the Parameters of the Reverse Gumbel Distribution*

Description

This function estimates the parameters of the Reverse Gumbel distribution given the type-B L-moments of the data in an L-moment object such as that returned by pwmRC using pwm2lmom. This distribution is important in the analysis of censored data. It is the distribution of a logarithmically transformed two-parameter Weibull distribution. The relation between distribution parameters and L-moments is

$$\alpha = \lambda_2^B / \{\log(2) + \mathrm{Ei}(-2\log(1-\zeta)) - \mathrm{Ei}(-\log(1-\zeta))\} \text{ and}$$

$$\xi = \lambda_1^B + \alpha\{\mathrm{Ei}(-\log(1-\zeta))\},$$

where ζ is the right-tail censoring fraction of the sample o the nonexceedance probability of the right-tail censoring threshold, and $\mathrm{Ei}(x)$ is the exponential integral defined as

$$\mathrm{Ei}(X) = \int_X^\infty x^{-1} e^{-x} \mathrm{d}x,$$

where $\mathrm{Ei}(-\log(1-\zeta)) \rightarrow 0$ as $\zeta \rightarrow 1$ and $\mathrm{Ei}(-\log(1-\zeta))$ can not be evaluated as $\zeta \rightarrow 0$.

Usage

```
parrevgum(lmom,zeta=1,checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by pwm2lmom through pwmRC or other L-moment type object. The user intervention of the zeta differentiates this distribution and hence this function from similar parameter estimation functions in the **lmomco** package.
zeta	The right censoring fraction. Number of samples observed (noncensored) divided by the total number of samples.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `revgum`.
`para`	The parameters of the distribution.
`zeta`	The right censoring fraction. Number of samples observed (noncensored) divided by the total number of samples.
`source`	The source of the parameters: "parrevgum".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

`pwm2lmom`, `pwmRC`, `cdfrevgum`, `quarevgum`

Examples

```
# See p. 553 of Hosking (1995)
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221,  0.245,  0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,threshold=.2960001)
str(z)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -.516 and lamB2 = 0.523
btypelmoms <- pwm2lmom(z$Bbetas)
# My version of R reports lamB1 = -0.5162 and lamB2 = 0.5218
str(btypelmoms)
rg.pars <- parrevgum(btypelmoms,z$zeta)
str(rg.pars)
# Hosking reports xi = 0.1636 and alpha = 0.9252 for the sample
# My version of R reports xi = 0.1635 and alpha = 0.9254
```

parrice *Estimate the Parameters of the Rice Distribution*

Description

This function estimates the parameters (ν and α) of the Rice distribution giventhe L-moments of the data in an L-moment object such as that returned bylmom.ub. The relations between distribution parameters and L-moments are complex and tabular lookup is made using a relation between τ and a form of signal-to-noise ratio SNR defined as ν/α and a relation between τ and precomputed Laguerre polynomial, which in turn requires modified Bessel functions of the first kind. The Laguerre polynomial is

$$L_{1/2}(x) = \mathrm{e}^{x/2} \times [(1-x)I_0(-x/2) - xI_1(-x/2)]$$

where the modified Bessel function of the first kind is $I_k(x)$.

The λ_1 (mean) is most straightforward

$$\lambda_1 = \alpha \times \sqrt{\pi/2} L_{1/2}(-\nu^2/[2\alpha^2])$$

for which the terms to the right of the multiplication symbol are uniquely a function of τ and precomputed for tabular lookup and interpolation from 'sysdata.rdb' (.lmomcohash$RiceTable). Parameter estimation also relies directly on tabular lookup and interpolation to convert τ to SNR. The file 'SysDataBuilder.R' provides additional technical details.

Usage

```
parrice(lmom, checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check. However, the end point of the Rice distribution for high ν/α is not determined here, so it is recommended to leave checklmom turned on.

Value

An R list is returned.

type	The type of distribution: rice.
para	The parameters of the distribution.
source	The source of the parameters: "parrice".
ifail	A numeric failure mode.
ifailtext	A helpful message on the failure.

Author(s)

W.H. Asquith

See Also

lmom.ub, lmomrice, cdfrice, quarice

Examples

```
parrice(lmomrice(vec2par(c(100,11), type="rice")))
parrice(lmomrice(vec2par(c(10,50), type="rice")))

## Not run:  # Beyond limits of the Rice
  parrice(lmomrice(vec2par(c(100,0.1), type="rice")))

## End(Not run)

plotlmrdia(lmrdia(), xlim=c(0,0.2), ylim=c(-0.1,0.22),
           autolegend=TRUE, xleg=0.05, yleg=0.05)
lines(.lmomcohash$RiceTable$TAU3, .lmomcohash$RiceTable$TAU4,
      lwd=5, col=8)
legend(0.1,0, "RICE DISTRIBUTION", lwd=5, col=8, bty="n")
text(0.14,-0.04, "Normal distribution limit on left end point")
text(0.14,-0.055, "Rayleigh distribution limit on right end point")


## Not run:  # check parrice against a Maximum Likelihood method in VGAM
set.seed(1)
library(VGAM) # now example from riceff() of VGAM
vee <- exp(2); sigma <- exp(1); y <- rrice(n <- 1000, vee, sigma)
fit <- vglm(y ~ 1, riceff, trace=TRUE, crit="c")
Coef(fit)
# NOW THE MOMENT OF TRUTH, USING L-MOMENTS
parrice(lmoms(y))
# VGAM 0.8-1 reports
#     vee    sigma
# 7.344560 2.805877
# lmomco 1.2.2 reports
#      nu    alpha
# 7.348784 2.797651

## End(Not run)
```

---

partexp   *Estimate the Parameters of the Truncated Exponential Distribution*

Description

This function estimates the parameters of the Truncated Exponential distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomtexp`.

Usage

```
partexp(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `texp`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “partexp”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomtexp`, `cdftexp`, `quatexp`

Examples

```
lmr <- lmom.ub(rnorm(20))
parexp(lmr)
```

parTLgld *Estimate the Parameters of the Generalized Lambda Distribution using Trimmed L-moments (t=1)*

Description

This function estimates the parameters of the Generalized Lambda distribution given the trimmed L-moments (TL-moments) for $t = 1$ of the data in a TL-moment object with a trim level of unity (`trim=1`). The relation between distribution parameters and TL-moments is seen under `lmomTLgld`. There are no simple expressions for the parameters in terms of the L-moments. This function is experimental and general details of the algorithm are provided below. Further, consider that multiple parameter solutions are possible with the Generalized Lambda so some expertise in the distribution and other aspects are needed.

Usage

```
parTLgld(lmom, verbose=FALSE, initkh=NULL, eps=1e-3,
         aux=c("tau5", "tau6"), checklmom=TRUE)
```

Arguments

`lmom`	A TL-moment object created by `TLmoms`.
`verbose`	A logical switch on the verbosity of output. Default is `verbose=FALSE`.
`initkh`	A vector of the initial guess of the κ and h parameters. No other regions of parameter space are consulted.
`eps`	A small term or threshold for which the square root of the sum of square errors in τ_3 and τ_4 is compared to to judge "good enough" for the alogrithm to order solutions based on smallest error as explained in next argument.
`aux`	Control the algorithm to order solutions based on smallest error in trimmed $\Delta\tau_5$ or $\Delta\tau_6$.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Details

Karian and Dudewicz (2000) summarize six regions of the κ and h space in which the Generalized Lambda distribution is valid for suitably choosen α. Numerical experimentation suggestions that the L-moments are not valid in Regions 1 and 2. However, initial guesses of the parameters within each region are used numerous six separate `optim` (the R function) efforts to perform a least sum-of-square errors on the following objective function.

$$(\hat{\tau}_3^{(1)} - \tilde{\tau}_3^{(1)})^2 + (\hat{\tau}_4^{(1)} - \tilde{\tau}_4^{(1)})^2,$$

where $\tilde{\tau}_r^{(1)}$ is the L-moment ratio of the data, $\hat{\tau}_r^{(1)}$ is the estimated value of the TL-moment ratio for the current pairing of κ and h and $\tau_r^{(1)}$ is the actual value of the L-moment ratio.

For each optimization a check on the validity of the parameters so produced is made–are the parameters consistent with the Generalized Lambda distribution and a second check is made on the validity of $\tau_3^{(1)}$ and $\tau_4^{(1)}$. If both validity checks return TRUE then the optimization is retained if its sum-of-square error is less than the previous optimum value. It is possible for a given solution to be found outside the starting region of the initial guesses. The surface generated by the $\tau_3^{(1)}$ and $\tau_4^{(1)}$ equations seen in lmomTLgld is complex–different initial guesses within a given region can yield what appear to be radically different κ and h. Users are encouraged to "play" with alternative solutions (see the verbose argument). A quick double check on the L-moments (not TL-moments) from the solved parameters using lmomTLgld is encouraged as well.

Value

An R list is returned if result='best'.

type	The type of distribution: gld.
para	The parameters of the distribution.
delTau5	Difference between $\tilde{\tau}_5^{(1)}$ of the fitted distribution and true $\hat{\tau}_5^{(1)}$.
error	Smallest sum of square error found.
source	The source of the parameters: "parTLgld".
rest	An R data.frame of other solutions if found.

The rest of the solutions have the following:

xi	The location parameter of the distribution.
alpha	The scale parameter of the distribution.
kappa	The 1st shape parameter of the distribution.
h	The 2nd shape parameter of the distribution.
attempt	The attempt number that found valid TL-moments and parameters of GLD.
delTau5	The absolute difference between $\hat{\tau}_5^{(1)}$ of data to $\tilde{\tau}_5^{(1)}$ of the fitted distribution.
error	The sum of square error found.
initial_k	The starting point of the κ parameter.
initial_h	The starting point of the h parameter.
valid.gld	Logical on validity of the GLD—TRUE by this point.
valid.lmr	Logical on validity of the L-moments—TRUE by this point.
lowerror	Logical on whether error was less than eps—TRUE by this point.

Note

Although the underlying derivations of the TL-moments in terms of the parameters are sound and this function uses built-in functions of R for the optimization, this function should be considered experimental until further notice. The "preferred" solution might not be appropriate. This function is not for production code. This is a cumbersome method of parameter solution and enhancements or total redesign of the algorithm is expected.

Author(s)

W.H. Asquith

Source

W.H. Asquith in Feb. 2006 with copy of Karian and Dudewicz (2000) and again Feb. 2011.

References

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

TLmoms, lmomTLgld, cdfgld, quagld, pargld

Examples

```
X <- rgamma(202,2) # simulate a skewed distribution
lmr <- TLmoms(X, trim=1) # compute trimmed L-moments
PARgldTL <- parTLgld(lmr) # fit the GLD

F <- pp(X) # plotting positions for graphing
plot(F,sort(X), col=8, cex=0.25)
lines(F, qlmomco(F,PARgldTL)) # show the best estimate
if(! is.null(PARgldTL$rest)) { #$
  n <- length(PARgldTL$rest$xi)
  other <- unlist(PARgldTL$rest[n,1:4]) #$ # show alternative
  lines(F, qlmomco(F,vec2par(other, type="gld")), col=2)
}
# Note in the extraction of other solutions that no testing for whether
# additional solutions were found is made. Also, it is quite possible
# that the other solutions "[n,1:4]" is effectively another numerical
# convergence on the primary solution. Some users of this example thus
# might not see two separate lines. Users are encouraged to inspect the
# rest of the solutions: print(PARgld$rest); #$

# For one run of the above example, the GLD results follow
#print(PARgldTL)
#$type
#[1] "gld"
#$para
#         xi       alpha       kappa           h
# 1.02333964 -3.86037875 -0.06696388 -0.22100601
#$delTau5
#[1] -0.02299319
#$error
#[1] 7.048409e-08
#$source
#[1] "pargld"
#$rest
#         xi     alpha       kappa          h attempt     delTau5        error
```

```
#1  1.020725 -3.897500 -0.06606563 -0.2195527       6 -0.02302222 1.333402e-08
#2  1.021203 -3.895334 -0.06616654 -0.2196020       4 -0.02304333 8.663930e-11
#3  1.020684 -3.904782 -0.06596204 -0.2192197       5 -0.02306065 3.908918e-09
#4  1.019795 -3.917609 -0.06565792 -0.2187232       2 -0.02307092 2.968498e-08
#5  1.023654 -3.883944 -0.06668986 -0.2198679       7 -0.02315035 2.991811e-07
#6 -4.707935 -5.044057  5.89280906 -0.3261837      13  0.04168800 2.229672e-10

## Not run:
F <- seq(.01,.99,.01)
plot(F,qlmomco(F, vec2par(c( 1.02333964, -3.86037875,
                            -0.06696388, -0.22100601), type="gld")),
     type="l")
lines(F,qlmomco(F, vec2par(c(-4.707935, -5.044057,
                              5.89280906, -0.3261837), type="gld")))

## End(Not run)
```

parTLgpa — *Estimate the Parameters of the Generalized Pareto Distribution using Trimmed L-moments*

Description

This function estimates the parameters of the Generalized Pareto distribution given the the trimmed L-moments (TL-moments) for $t = 1$ of the data in TL-moment object with a trim level of unity (`trim=1`). The parameters are computed as

$$\kappa = \frac{10 - 45\tau_3^{(1)}}{9\tau_3^{(1)} + 10},$$

$$\alpha = \frac{1}{6}\lambda_2^{(1)}(\kappa + 2)(\kappa + 3)(\kappa + 4), \text{ and}$$

$$\xi = \lambda_1^{(1)} - \frac{\alpha(\kappa + 5)}{(\kappa + 2)(\kappa + 3)}.$$

Usage

```
parTLgpa(lmom)
```

Arguments

`lmom`	A TL-moment object created by `TLmoms`.

Value

An R `list` is returned.

`type`	The type of distribution: `gpa`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “parTLgpa”.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`TLmoms`, `lmomTLgpa`, `cdfgpa`, `quagpa`

Examples

```
TL <- TLmoms(rnorm(20),trim=1)
parTLgpa(TL)
```

---

parwak *Estimate the Parameters of the Wakeby Distribution*

Description

This function estimates the parameters of the Wakeby distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The relation between distribution parameters and L-moments is seen under `lmomwak`.

Usage

```
parwak(lmom,checklmom=TRUE)
```

Arguments

`lmom`	A L-moment object created by `lmom.ub` or `pwm2lmom`.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R `list` is returned.

`type`	The type of distribution: `wak`.
`para`	The parameters of the distribution.
`source`	The source of the parameters: “parwak”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `lmomwak`, `cdfwak`, `quawak`

Examples

```
lmr <- lmom.ub(rnorm(20))
parwak(lmr)
```

`parwei` *Estimate the Parameters of the Weibull Distribution*

Description

This function estimates the parameters of the Weibull distribution given the L-moments of the data in an L-moment object such as that returned by `lmom.ub`. The Weibull distribution is a reverse Generalized Extreme Value distribution. As result, the Generalized Extreme Value algorithms are used for implementation of the Weibull in this package. The relation between the Generalized Extreme Value parameters (ξ, α, and κ) is

$$\kappa = 1/\delta,$$

$$\alpha = \beta/\delta, \text{ and}$$

$$\xi = \zeta - \beta.$$

These relations are taken from Hosking and Wallis (1997).

Usage

```
parwei(lmom,checklmom=TRUE)
```

Arguments

lmom	A L-moment object created by lmom.ub or pwm2lmom.
checklmom	Should the lmom be checked for validity using the are.lmom.valid function. Normally this should be left as the default and it is very unlikely that the L-moments will not be viable (particularly in the τ_4 and τ_3 inequality). However, for some circumstances or large simulation exercises then one might want to bypass this check.

Value

An R list is returned.

type	The type of distribution: wei.
para	The parameters of the distribution.
source	The source of the parameters: “parwei”.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

lmom.ub, lmomwei, cdfwei, quawei

Examples

```
lmr <- lmom.ub(rnorm(20))
parwei(lmr)
lmr <- parwei(lmom.ub(rweibull(3000,1.3,scale=340)-1200))
str(lmr)
```

pdfaep4 *Probability Density Function of the 4-p Asymmetric Exponential Power Distribution*

Description

This function computes the probability density of the 4-parameter Asymmetric Exponential Power distribution given parameters (ξ, α, κ, and h) of the distribution computed by paraep4. The probability density function of the distribution is

$$f(x) = \frac{\kappa\, h}{\alpha(1+\kappa^2)\,\Gamma(1/h)} \exp[-(\kappa^{\mathrm{sign}(x-\xi)}\,(\,|x-\xi|/\alpha)\,)^h]$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, α is a scale parameter, κ is a shape parameter, and h is another shape parameter. The range of the distribution is $-\infty < x < \infty$.

Usage

```
pdfaep4(x, para, paracheck=TRUE)
```

Arguments

`x`	A real value.
`para`	The parameters from paraep4 or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Ayebo, A., and Kozubowski, T.J., 2003, An asymmetric generalization of Gaussian and Laplace laws: Journal of Probability and Statistical Science, v. 1, no. 2, pp. 187-210.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

See Also

cdfaep4, quaaep4, paraep4

Examples

```
aep4 <- vec2par(c(1000,15000,0.5,0.4), type='aep4');
F <- nonexceeds();
x <- quaaep4(F,aep4);
check.pdf(pdfaep4,aep4,plot=TRUE);

## Not run:
delx <- .01;
x <- seq(-10,10, by=delx);
K <- 3;
PAR <- list(para=c(0,1, K, 0.5), type="aep4");
plot(x,pdfaep4(x, PAR), type="n",
     ylab="PROBABILITY DENSITY",
     ylim=c(0,0.6), xlim=range(x));
lines(x,pdfaep4(x,PAR), lwd=2);

PAR <- list(para=c(0,1, K, 1), type="aep4");
lines(x,pdfaep4(x, PAR), lty=2, lwd=2);

PAR <- list(para=c(0,1, K, 2), type="aep4");
lines(x,pdfaep4(x, PAR), lty=3, lwd=2);

PAR <- list(para=c(0,1, K, 4), type="aep4");
lines(x,pdfaep4(x, PAR), lty=4, lwd=2);

## End(Not run)
```

---

pdfcau *Probability Density Function of the Cauchy Distribution*

Description

This function computes the probability density of the Cauchy distribution given parameters (ξ and α) of the distribution provided by parcau or vec2par. The probability density function of the distribution is

$$f(x) = \left(\pi\alpha[1 + \left(\frac{x-\xi}{\alpha}\right)^2]\right)^{-1},$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter and α is a scale parameter.

Usage

```
pdfcau(x, para)
```

Arguments

x	A real value.
para	The parameters from parcau or vec2par.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Evans, Merran, Hastings, Nicholas, Peacock, J.B., 2000, Statistical distributions: 3rd ed., Wiley, New York.

Gilchrist, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

cdfcau, quacau, parcau, vec2par

Examples

```
cau <- vec2par(c(12,12),type='cau')
x <- quacau(0.5,cau)
pdfcau(x,cau)
```

---

pdfexp *Probability Density Function of the Exponential Distribution*

Description

This function computes the probability density of the Exponential distribution given parameters (ξ and α) of the distribution computed by parexp. The probability density function of the distribution is

$$f(x) = \alpha^{-1} e^{\left(\frac{-(x-\xi)}{\alpha}\right)}$$

where $f(x)$ is the probability density for the quantile x, ξ is a location parameter and α is a scale parameter.

Usage

```
pdfexp(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parexp` or similar.

Value

Probability density (F) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfexp`, `quaexp`, `parexp`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
expp <- parexp(lmr)
x <- quaexp(.5,expp)
pdfexp(x,expp)
```

`pdfgam` *Probability Density Function of the Gamma Distribution*

Description

This function computes the probability density function of the Gamma distribution given parameters (α, shape, and β, scale) of the distribution computed by pargam. The probability density function of the distribution has no explicit form, but is expressed as an integral.

$$f(x) = \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha - 1} e^{-x/\beta},$$

where $f(x)$ is the probability density for the quantile x. The parameters have the following interpretation in the **R** syntax; α is a shape parameter and β is a scale parameter.

Usage

```
pdfgam(x, para)
```

Arguments

x	A real value.
para	The parameters from pargam or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgam, quagam, pargam

Examples

```
lmr <- lmoms(c(123,34,4,654,37,78))
gam <- pargam(lmr)
x <- quagam(0.5,gam)
pdfgam(x,gam)
```

`pdfgev` *Probability Density Function of the Generalized Extreme Value Distribution*

Description

This function computes the probability density of the Generalized Extreme Value distribution given parameters (ξ, α, and κ) of the distribution computed by `pargev`. The probability density function of the distribution is

$$f(x) = \alpha^{-1} e^{-(1-\kappa)y - e^{-y}},$$

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0, \text{ and}$$

$$y = (x-\xi)/\alpha \text{ for } \kappa = 0,$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
pdfgev(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `pargev` or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pdfgev, quagev, pargev

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
gev <- pargev(lmr)
x <- quagev(0.5,gev)
pdfgev(x,gev)
```

---

pdfgld *Probability Density Function of the Generalized Lambda Distribution*

Description

This function computes the probability density function of the Generalized Lambda distribution given parameters (ξ, α, κ, and h) of the distribution computed by pargld or similar. The probability density function is

$$f(x) = [(\kappa(F^{\kappa-1}) + h(1-F)^{h-1}) \times \alpha]^{-1} \text{at } x = Q(F)$$

where $f(x)$ is the probability density function at x, $Q(F)$ is the quantile function for nonexceedance probability F.

Usage

```
pdfgld(x, gldpara, paracheck)
```

Arguments

x	A real value.
gldpara	The parameters from pargld or similar.
paracheck	A logical switch as to whether the validity of the parameters should be checked. Default is paracheck=TRUE. This switch is made so that the root solution needed for cdfgld exhibits an extreme speed increase because of the repeated calls to quagld.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

cdfgld, quagld, lmomgld, pargld

Examples

```
# Using Karian and Dudewicz, 2000, p. 10
gld <- vec2par(c(0.0305,1/1.3673,0.004581,0.01020),type='gld')
quagld(0.25,gld) # which equals about 0.028013 as reported by K&D
pdfgld(0.028013,gld) # which equals about 43.04 as reported by K&D
## Not run:
F <- seq(.001,.999,by=.001)
x <- quagld(F,gld)
plot(x, pdfgld(x,gld), type='l', xlim=c(0,.1))

## End(Not run)
```

---

pdfglo *Probability Density Function of the Generalized Logistic Distribution*

Description

This function computes the probability density of the Generalized Logistic distribution given parameters (ξ, α, and κ) of the distribution computed by parglo. The probability density function of the distribution is

$$f(x) = \frac{\alpha^{-1} e^{-(1-\kappa)y}}{(1+e^{-y})^2},$$

where y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0\text{,and}$$

$$y = (x-\xi)/\alpha \text{ for } \kappa = 0\text{, and}$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
pdfglo(x, para)
```

Arguments

x	A real value.
para	The parameters from parglo or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pdfglo, quaglo, parglo

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
glo <- parglo(lmr)
x <- quaglo(0.5,glo)
pdfglo(x,glo)
```

---

pdfgno — *Probability Density Function of the Generalized Normal Distribution*

Description

This function computes the probability density of the Generalized Normal distribution given parameters (ξ, α, and κ) of the distribution computed by pargno. The probability density function function of the distribution is

$$f(x) = \frac{\exp(\kappa y - y^2/2)}{\alpha\sqrt{2\pi}},$$

where Φ is the cumulative distribution function of the standard normal distribution and y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x - \xi)}{\alpha}\right) \text{ for } \kappa \neq 0, \text{ and}$$

$$y = (x - \xi)/\alpha \text{ for } \kappa = 0,$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
pdfgno(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `pargno` or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfgno`, `quagno`, `pargno`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
gno <- pargno(lmr)
x <- quagno(0.5,gno)
pdfgno(x,gno)
```

pdfgpa *Probability Density Function of the Generalized Pareto Distribution*

Description

This function computes the probability density of the Generalized Pareto distribution given parameters (ξ, α, and κ) of the distribution computed by pargpa. The probability density function of the distribution is

$$f(x) = \alpha^{-1} e^{-(1-\kappa)y},$$

where y is

$$y = -\kappa^{-1} \log\left(1 - \frac{\kappa(x-\xi)}{\alpha}\right) \text{ for } \kappa \neq 0, \text{ and}$$

$$y = (x-\xi)/A \text{ for } \kappa = 0,$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
pdfgpa(x, para)
```

Arguments

x	A real value.
para	The parameters from pargpa or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgpa, quagpa, pargpa

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
gpa <- pargpa(lmr)
x <- quagpa(0.5,gpa)
pdfgpa(x,gpa)
```

---

pdfgum *Probability Density Function of the Gumbel Distribution*

Description

This function computes the probability density of the Gumbel distribution given parameters (ξ and α) of the distribution computed by pargum. The probability density function of the distribution is

$$f(x) = \alpha^{-1} e^{\left(-\frac{(x-\xi)}{\alpha}\right)} e^{-e^{\left(-\frac{(x-\xi)}{\alpha}\right)}},$$

where $f(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, and α is a scale parameter.

Usage

```
pdfgum(x, para)
```

Arguments

x	A real value.
para	The parameters from pargum or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgum, quagum, pargum

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
gum <- pargum(lmr)
x <- quagum(0.5,gum)
pdfgum(x,gum)
```

---

pdfkap   *Probability Density Function of the Kappa Distribution*

Description

This function computes the probability density of the Kappa distribution given parameters (ξ, α, κ, and h) of the distribution computed by parkap. The probability density function of the distribution is

$$f(x) = \alpha^{-1}[1 - \kappa(x - \xi)/\alpha]^{1/k-1} \times [F(x)]^{1-h}$$

where $f(x)$ is the probability density for quantile x, $F(x)$ is the cumulative distribution function of the distribution (cdfkap)), ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
pdfkap(x, para)
```

Arguments

x	A real value.
para	The parameters from parkap or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

Sourced from written communication with Dr. Hosking in October 2007.

See Also

cdfkap, quakap, parkap

Examples

```
kap <- vec2par(c(1000,15000,0.5,-0.4),type='kap')
F <- nonexceeds()
x <- quakap(F,kap)
check.pdf(pdfkap,kap,plot=TRUE)
```

---

pdfkur *Probability Density Function of the Kumaraswamy Distribution*

Description

This function computes the probability density of the Kumaraswamy distribution given parameters (α and β) of the distribution computed by parkur. The probability density function of the distribution is

$$f(x) = \alpha\beta x^{\alpha-1}(1 - x^{\alpha})^{\beta-1},$$

where $f(x)$ is the nonexceedance probability for quantile x, α is a shape parameter, and β is a shape parameter.

Usage

```
pdfkur(x, para)
```

Arguments

x	A real value.
para	The parameters from parkur or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

cdfkur, quakur, parkur

Examples

```
lmr <- lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9))
kur <- parkur(lmr)
x <- quakur(0.5,kur)
pdfkur(x,kur)
```

---

pdflap *Probability Density Function of the Laplace Distribution*

Description

This function computes the probability density of the Laplace distribution given parameters (ξ and α) of the distribution computed by parlap. The probability density function of the distribution is

$$f(x) = \frac{1}{2}\alpha^{-1}e^{-|x-\xi|/\alpha}.$$

Usage

```
pdflap(x, para)
```

Arguments

x	A real value.
para	The parameters from parlap or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

pdflap, qualap, parlap

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
lap <- parlap(lmr)
x <- qualap(0.5,lap)
pdflap(x,lap)
```

---

pdfln3 *Probability Density Function of the 3-Parameter Log-Normal Distribution*

Description

This function computes the probability density of the Log-Normal3 distribution given parameters (ζ, μ, and σ) of the distribution computed by parln3. The probability density function function of the distribution is

$$f(x) = \frac{exp\kappa y - y^2/2}{\alpha\sqrt{2\pi}},$$

where Φ is the cumulative ditribution function of the standard normal distribution and y is

Usage

```
pdfln3(x, para)
```

Arguments

x	A real value.
para	The parameters from parln3 or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

NEED

See Also

cdfln3, qualn3, parln3

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
ln3 <- parln3(lmr)
x <- qualn3(0.5,ln3)
pdfgno(x,ln3)
```

---

pdfnor — *Probability Density Function of the Normal Distribution*

Description

This function computes the probability density function of the Normal distribution given parameters of the distribution computed by parnor. The probability density function of the distribution is

$$f(x) = \frac{1}{\sigma sqrt 2\pi} e^{-(x-\mu)^2/(2\sigma^2)},$$

where $f(x)$ is the probability density for quantile x, μ is the arithmetic mean, and σ is the standard deviation, and Φ is the cumulative distribution function of the standard normal distribution. The R-function pnorm is used.

Usage

```
pdfnor(x, para)
```

Arguments

x	A real value.
para	The parameters from parnor or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfnor, quanor, parnor

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
pdfnor(50,parnor(lmr))
```

---

pdfpe3 *Probability Density Function of the Pearson Type III Distribution*

Description

This function computes the probability density of the Pearson Type III distribution given parameters (μ, σ, and γ) of the distribution computed by parpe3. These parameters are equal to the product moments: mean, standard deviation, and skew (see pmoms). The probability density function of the distribution for $\gamma \neq 0$ is

$$f(x) = \frac{Y^{\alpha-1} exp{-Y/\beta}}{\beta^{\alpha}\Gamma(\alpha)},$$

where $f(x)$ is the probability density for quantile x, G is defined below and is related to the incomplete gamma function of R (pgamma()), Γ is the complete gamma function, ξ is a location parameter, β is a scale parameter, α is a shape parameter, and $Y = x - \xi$ if $\gamma > 0$ and $Y = \xi - x$ if $\gamma < 0$ These three "new" parameters are related to the product moments by

$$\alpha = 4/\gamma^2,$$

$$\beta = \frac{1}{2}\sigma|\gamma|,$$

$$\xi = \mu - 2\sigma/\gamma.$$

The function $G(\alpha, x)$ is

$$G(\alpha, x) = \int_0^x t^{(a-1)} \mathrm{e}^{-t} \mathrm{d}t.$$

If $\gamma = 0$, the distribution is symmetrical and simply is the probability density normal distribution with mean and standard deviation of μ and σ, respectively. Internally, the $\gamma = 0$ condition is implemented by pnorm().

Usage

```
pdfpe3(x, para)
```

Arguments

x	A real value.
para	The parameters from parpe3 or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfpe3, quape3, parpe3

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
pe3 <- parpe3(lmr)
x <- quape3(0.5,pe3)
pdfpe3(x,pe3)
```

---

pdfray *Probability Density Function of the Rayleigh Distribution*

Description

This function computes the probability density of the Rayleigh distribution given parameters (ξ and α) of the distribution computed by parray. The probability density function of the distribution is

$$f(x) = \frac{x - \xi}{\alpha^2} e^{-(x-\xi)^2/(2\alpha^2)},$$

where $f(x)$ is the nonexceedance probability for quantile x, ξ is a location parameter, and α is a scale parameter.

Usage

```
pdfray(x, para)
```

Arguments

x	A real value.
para	The parameters from parray or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

cdfray, quaray, parray

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
ray <- parray(lmr)
x <- quaray(0.5,ray)
pdfray(x,ray)
```

---

pdfrevgum *Probability Density Function of the Reverse Gumbel Distribution*

Description

This function computes the probability density of the Reverse Gumbel distribution given parameters (ξ and α) of the distribution computed by parrevgum. The probability density function of the distribution is

$$f(x) = \alpha^{-1}[-e^{-e^{-y}}][-e^{y}],$$

where

$$y = \frac{x - \xi}{\alpha},$$

where $f(x)$ is the probability density for quantile x, ξ is a location parameter, and α is a scale parameter. Notice that the function has some sign differences and uses the complement of f compared to the probability density function of the Gumbel distribution in pdfgum.

Usage

```
pdfrevgum(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parrevgum` or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

`cdfrevgum`, `quarevgum`, `parrevgum`

Examples

```
# See p. 553 of Hosking (1995)
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221,  0.245,  0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,threshold=.2960001)
str(z)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -0.516 and lamB2 = 0.523
btypelmoms <- pwm2lmom(z$Bbetas)
# My version of R reports lamB1 = -0.5162 and lamB2 = 0.5218
str(btypelmoms)
rg.pars <- parrevgum(btypelmoms,z$zeta)
str(rg.pars)
# Hosking reports xi=0.1636 and alpha=0.9252 for the sample
# My version of R reports xi = 0.1635 and alpha = 0.9254
# Now one can continue one with a plotting example.
```

```
## Not run:
F  <- nonexceeds()
PP <- pp(D) # plotting positions of the data
D  <- sort(D)
plot(D,PP)
lines(D,cdfrevgum(D,rg.pars))
# Now finally do the PDF
F <- seq(0.01,0.99,by=.01)
x <- quarevgum(F,rg.pars)
plot(x,pdfrevgum(x,rg.pars),type='l')

## End(Not run)
```

---

pdfrice *Probability Density Function of the Rice Distribution*

Description

This function computes the probability density of the Rice distribution given parameters (ν and SNR) of the distribution computed by parrice. The probability density function of the distribution is

$$f(x) = \frac{x}{\alpha^2}\exp(-(x^2+\nu^2)/(2\alpha^2))I_o(x\nu/\alpha^2)$$

where $f(x)$ is the nonexceedance probability for quantile x, ν is a parameter, and ν/α is a form of signal-to-noise ratio SNR. If $\nu = 0$, then the Rayleigh distribution results and pdfray is used. If $24 < \mathrm{SNR} < 52$ is used, then the Normal distribution functions are used with appropriate parameter estimation for μ and σ that include the Laguerre polynomial LaguerreHalf. If $\mathrm{SNR} > 52$, then the Normal distribution functions continue to be used with $\mu = \alpha * \mathrm{SNR}$ and $\sigma = A$.

Usage

```
pdfrice(x, para)
```

Arguments

x	A real value.
para	The parameters from parrice or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

See Also

cdfrice, quarice, parrice

Examples

```
lmr <- lmoms(c(10, 43, 27, 26, 49, 26, 62, 39, 51, 14))
rice <- parrice(lmr)
x <- quarice(nonexceeds(),rice)
plot(x,pdfrice(x,rice), type="b")

## Not run:
drice = function(x, vee, sigma, log = FALSE) { # From the VGAM package
    if (!is.logical(log.arg <- log))
        stop("bad input for argument 'log'")
    rm(log)


    N = max(length(x), length(vee), length(sigma))
    x = rep(x, len=N); vee = rep(vee, len=N); sigma = rep(sigma, len=N)

    logdensity = rep(log(0), len=N)
    xok = (x > 0)
    x.abs = abs(x[xok]*vee[xok]/sigma[xok]^2)
    logdensity[xok] = log(x[xok]) - 2 * log(sigma[xok]) +
                      (-(x[xok]^2+vee[xok]^2)/(2*sigma[xok]^2)) +
                      log(besselI(x.abs, nu=0, expon.scaled = TRUE)) +
                      x.abs
    logdensity[sigma <= 0] = NaN
    logdensity[vee < 0] = NaN
    if (log.arg) logdensity else exp(logdensity)
}


## End(Not run)

# For SNR=v/a > 24 or 240.001/10 > 24, the Normal distribution is
# used by the Rice as implemented here.
rice1 <- vec2par(c(239.9999,10), type="rice")
rice2 <- vec2par(c(240.0001,10), type="rice")
x <- 200:280
plot( x, pdfrice(x, rice1), type="l", lwd=5, lty=3) # still RICIAN code
lines(x, dnorm(  x, mean=240.0001, sd=10), lwd=3, col=2) # NORMAL obviously
lines(x, pdfrice(x, rice2), lwd=1, col=3) # NORMAl code triggered


# For SNR=v/a > 52 or 521/10 > 52, the Normal distribution is
# used by the Rice as implemented here with simple parameter estimation
# because this high of SNR is beyond limits of Bessel function in Laguerre
# polynomial
rice1 <- vec2par(c(520,10), type="rice")
rice2 <- vec2par(c(521,10), type="rice")
x <- 10^(log10(520)-.01):10^(log10(520)+.01)
plot( x, pdfrice(x, rice1), type="l", lwd=5, lty=3)
lines(x, pdfrice(x, rice2), lwd=1, col=3) # NORMAl code triggered
```

pdftexp *Probability Density Function of the Truncated Exponential Distribution*

Description

This function computes the probability density of the Truncated Exponential distribution given parameters (ξ and α) of the distribution computed by partexp. The probability density function of the distribution is

$$f(x) = \frac{\alpha^{-1}\exp(-t/\alpha)}{1-\exp(-\xi/\alpha)},$$

where $f(x)$ is the probability density for the quantile x, ξ is a location parameter, α is a scale parameter, and $0 \le x \le \xi$. The distribution has $0 < \tau_2 <= 1/2$, $\xi > 0$, and $1/\alpha \ne 0$.

Usage

```
pdftexp(x, para)
```

Arguments

x	A real value.
para	The parameters from partexp or similar.

Value

Probability density (F) for x.

Author(s)

W.H. Asquith

References

Vogel, R.M., Hosking, J.R.M., Elphick, C.S., Roberts, D.L., and Reed, J.M., 2008, Goodness of fit of probability distributions for sightings as species approach extinction: Bulletin of Mathematatial Biology, v. 71, no. 3, pp. 701–719.

See Also

cdftexp, quatexp, partexp

Examples

```
lmr <- vec2lmom(c(40,0.38), lscale=FALSE)
expp <- partexp(lmr)
x <- quatexp(.5,expp)
pdftexp(x,expp)
```

pdfwak *Probability Density Function of the Wakeby Distribution*

Description

This function computes the probability density of the Wakeby distribution given parameters (ξ, α, β, γ, and δ) of the distribution computed by pargev. The probability density function of the distribution is

$$f(x) = [\alpha(1-F)^{\beta-1} + \gamma(1-F)^{-\delta-1}]^{-1},$$

where $f(x)$ is the probability density for quantile x,ξ is a location parameter, α and β are scale parameters, and γ, and δ are shape parameters. The five returned parameters from parwak in order are ξ, α, β, γ, and δ.

Usage

```
pdfwak(x, para)
```

Arguments

x	A real value.
para	The parameters from parwak or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

Sourced from written communication with Dr. Hosking in October 2007.

See Also

cdfwak, quawak, parwak

Examples

```
lmr <- vec2lmom(c(1,0.5,.4,.3,.15))
wak <- parwak(lmr)
F <- nonexceeds()
x <- quawak(F,wak)
check.pdf(pdfwak,wak,plot=TRUE)
```

`pdfwei` *Probability Density Function of the Weibull Distribution*

Description

This function computes the probability density of the Weibull distribution given parameters (ζ, β, and δ) of the distribution computed by `parwei`. The probability density function of the distribution is

$$f(x) = \delta Y^{\delta - 1} \exp(-Y^{\delta})/\beta$$

where $f(x)$ is the probability density, $Y = (x - \zeta)/\beta$, quantile x, ζ is a location parameter, β is a scale parameter, and δ is a shape parameter.

The Weibull distribution is a reverse Generalized Extreme Value distribution. As result, the Generalized Extreme Value algorithms are used for implementation of the Weibull in this package. The relation between the Generalized Extreme Value parameters (ξ, α, and κ) is: $\kappa = 1/\delta$, $\alpha = \beta/\delta$, and $\xi = \zeta - \beta$. These relations are taken from Hosking and Wallis (1997).

In R the probability distribution function of the Weibull distribution is `pweibull`. Given a Weibull parameter object `para`, the R syntax is `pweibull(x+para$para[1], para$para[3], scale=para$para[2])`. For the current implementation for this package, the reversed Generalized Extreme Value distribution is used `pdfgev(-x,para)`.

Usage

```
pdfwei(x, para)
```

Arguments

`x`	A real value.
`para`	The parameters from `parwei` or similar.

Value

Probability density (f) for x.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`quawei`, `parwei`

Examples

```
  # Evaluate Weibull deployed here and within R (pweibull)
  lmr <- lmom.ub(c(123,34,4,654,37,78))
  WEI <- parwei(lmr)
  F1  <- cdfwei(50,WEI)
  F2  <- pweibull(50+WEI$para[1],shape=WEI$para[3],scale=WEI$para[2])
  if(F1 == F2) EQUAL <- TRUE

## Not run:
  # The Weibull is a reversed generalized extreme value
  Q <- sort(rlmomco(34,WEI)) # generate Weibull sample
  lm1 <- lmoms(Q)    # regular L-moments
  lm2 <- lmoms(-Q)   # L-moment of negated (reversed) data
  WEI <- parwei(lm1) # parameters of Weibull
  GEV <- pargev(lm2) # parameters of GEV
  F <- nonexceeds()  # Get a vector of nonexceedance probs
  plot(pp(Q),Q)
  lines(cdfwei(Q,WEI),Q,lwd=5,col=8)
  lines(1-cdfgev(-Q,GEV),Q,col=2) # line over laps previous

## End(Not run)
```

---

plmomco *Cumulative Distribution Function of the Distributions*

Description

This function acts as an alternative front end to `par2cdf`. The nomenclature of the `plmomco` function is to mimic that of built-in **R** functions.

Usage

```
plmomco(x,para)
```

Arguments

`x`	A real value.
`para`	The parameters from `lmom2par` or similar.

Value

Nonexceedance probability ($0 \leq F \leq 1$) for `x`.

Author(s)

W.H. Asquith

See Also

dlmomco, rlmomco, qlmomco

Examples

```
para <- vec2par(c(0,1),type='nor') # standard normal parameters
nonexceed <- plmomco(1,para) # percentile of one standard deviation
```

---

plotlmrdia — *Plot L-moment Ratio Diagram*

Description

Plot the L-moment ratio diagram of L-skew and L-kurtosis from an L-moment ratio diagram object returned by `lmrdia`. This diagram is useful for selecting a distribution to model the data. The application of L-moment diagrams is well documented in the literature. This function is intended to function as a demonstration of L-moment diagram plotting. It is expected that users will "roll their own" version of this function for their own specific purposes.

Usage

```
plotlmrdia(lmr, nopoints=FALSE, nolines=FALSE, nolimits=FALSE,
           noaep4=FALSE, nogev=FALSE, noglo=FALSE, nogpa=FALSE,
           nope3=FALSE,  nogno=FALSE, nocau=FALSE, noexp=FALSE,
           nonor=FALSE,  nogum=FALSE, noray=FALSE, nouni=FALSE,
           xlab="L-SKEW", ylab="L-KURTOSIS",
           autolegend=FALSE, xleg=NULL, yleg=NULL, ...)
```

Arguments

`lmr`	L-moment diagram object from `lmrdia`.
`nopoints`	If `TRUE` then point distributions are not drawn.
`nolines`	If `TRUE` then line distributions are not drawn.
`nolimits`	If `TRUE` then theoretical limits of L-moments are not drawn.
`noaep4`	If `TRUE` then line of Asymmetric Exponential Power distribution is not drawn.
`nogev`	If `TRUE` then line of Generalized Extreme Value distribution is not drawn.
`noglo`	If `TRUE` then line of Generalized Logistic distribution is not drawn.
`nogno`	If `TRUE` then line of Generalized Normal (log-Normal) distribution is not drawn.
`nogpa`	If `TRUE` then line of Generalized Pareto distribution is not drawn.
`nope3`	If `TRUE` then line of Pearson Type III distribution is not drawn.
`nocau`	If `TRUE` then point (limiting) of the Cauchy distribution is not drawn.
`noexp`	If `TRUE` then point of Exponential distribution is not drawn.
`nonor`	If `TRUE` then point of Normal distribution is not drawn.

`nogum`	If `TRUE` then point of Gumbel distribution is not drawn.
`noray`	If `TRUE` then point of Rayleigh distribution is not drawn.
`nouni`	If `TRUE` then point of Uniform distribution is not drawn.
`xlab`	Horizonal axis label passed to `xlab` of the `plot` function.
`ylab`	Vertical axis label passed to `ylab` of the `plot` function.
`autolegend`	Generate the legend by built-in algorithm.
`xleg`	X-coordinate of the legend.
`yleg`	Y-coordinate of the legend.
`...`	Additional arguments passed onto the `plot` function.

Note

This function provides hardwired calls to `lines` and `points` to produce the diagram. The plot symbology for the shown distributions is summarized here. The Kappa (four parameter) and Wakeby (five parameter) distributions are not well represented on the diagram as each constitute an area (Kappa) or hyperplane (Wakeby) and not a line (three-parameter distributions) or a point (two-parameter distributions). However, the Kappa demarks the area bounded by the Generalized Logistic (`glo`) on the top and the theoretical L-moment limits on the bottom.

GRAPHIC TYPE	GRAPHIC NATURE
L-moment Limits	line width 2 and color 8 (grey)
Generalized Extreme Value	line width 1, line type 2 (dash), and color 2 (red)
Generalized Logistic	line width 1 and color 3 (green)
Generalized Normal	line width 1, line type 2 (dash), and color 4 (blue)
Generalized Pareto	line width 1 and color 4 (blue)
Pearson Type III	line width 1 and color 6 (purple)
Exponential	symbol 16 (filled circle) and color 2 (red)
Normal	symbol 15 (filled square) and color 2 (red)
Gumbel	symbol 17 (filled triangle) and color 2 (red)
Rayleigh	symbol 18 (filled diamond) and color 2 (red)
Uniform	symbol 12 (square and a plus sign) and color 2 (red)

Author(s)

W.H. Asquith

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98-4044, 107 p.

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, pp. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis–An approach based on L-moments: Cambridge University Press.

Vogel, R.M., and Fennessey, N.M., 1993, L moment diagrams should replace product moment diagrams: Water Resources Research, vol. 29, no. 6, pp. 1745–1752.

See Also

lmrdia

Examples

```
plotlmrdia(lmrdia())

# A more complex example follows.

# For a given mean, L-scale, L-skew and L-kurtosis, a sample size
# of 30 and using 50 simulations, set the L-moments in lmr and fit
# a Kappa distribution
T3 <- 0.34; T4 <- 0.21; n <- 30; nsim <- 50;
lmr <- vec2lmom(c(10000,7500,T3,T4)); kap <- parkap(lmr)

# Next, create vectors for storage of simulated L-skew (t3)
# and L-kurtosis (t4)
t3 <- vector(mode = "numeric"); t4 <- t3;

# Next, perform nsim simulations by randomly drawing from the Kappa
# distribution and compute the L-moments in sim.lmr and store the
# t3 and t4 values of each simulated sample.
for(i in 1:nsim) {
  sim.lmr <- lmoms(rlmomco(n,kap))
  t3[i] <- sim.lmr$ratios[3]; t4[i] <- sim.lmr$ratios[4]
}

# Finally, plot the diagram with a legend at a specified location,
# and "zoom" into the diagram by setting the axis limits.
plotlmrdia(lmrdia(), autolegend=TRUE, xleg=0.1, yleg=.41,
           xlim=c(-.1,.5), ylim=c(-.1,.4), nopoints=TRUE)


# Follow up with plotting of the t3,t4 values and the mean of these.
points(t3,t4)
points(mean(t3),mean(t4),pch=16,cex=3)

# A complete the example by plotting crossing dashed lines at the
# population values of L-skew and L-kurtosis
lines(c(T3,T3),c(-1,1),col=8, lty=2)
lines(c(-1,1),c(T4,T4),col=8, lty=2)
```

`pmoms` *The Sample Product Moments: Mean, Standard Deviation, Skew, and Excess Kurtosis*

Description

Compute the four sample product moments for a vector.

Usage

```
pmoms(x)
```

Arguments

`x`	A vector of data values.

Value

An R `list` is returned.

`moments`	Vector of the product moments. First element is the mean (`mean()`), second is standard deviation, and the higher values typically are not used, but the ratios[3] and ratios[4] are.
`ratios`	Vector of the product moment ratios. Second element is the coefficient of variation, ratios[3] is skew, and ratios[4] is excess kurtosis.
`sd`	Nearly unbiased standard deviation [well at least unbiased variance (`unbiased.sd^2`)] computed by R `sd()`.
`umvu.sd`	Uniformly-minimum variance unbiased estimator of standard deviation.
`skew`	Nearly unbiased skew, same as ratios[3].
`kurt`	Nearly nbiased excess kurtosis, same as ratios[4].
`classic.sd`	Classical (theoretical) definition of standard deviation.
`classic.skew`	Classical (theoretical) definition of skew.
`classic.kurtosis`	Classical (theoretical) definition of excess kurtosis
`message`	The product moments are confusing in terms of definition because they are not naturally unbiased. Your author thinks that it is informative to show the biased versions on the output from the `pmoms` function. Therefore, this `message` includes several clarifications of the output.
`source`	An attribute identifying the computational source (the function name) of the product moments: "pmoms".

Note

This function is primarily available for gamesmanship playing with the Pearson Type III distribution as its parameterization in the **lmomco** package returns the product moments as the parameters. See the example below. Another reason for having this function is that it demonstrates application of unbiased product moments and permits comparisons to the L-moments.

The `umvu.sd` is computed by

$$\hat{\sigma}' = \frac{\Gamma[(n-1)/2]}{\Gamma(n/2)\sqrt{2}} \sqrt{\sum_{i=1}^{n}(x_i - \hat{\mu})^2}$$

Author(s)

W.H. Asquith

References

Dingman, S.L., 2002, Physical hydrology, 2nd ed: Prentice Hall, Upper Saddle River, N.J., appendix C.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmoms`

Examples

```
# A simple example
PM <- pmoms(rnorm(1000)) # n standard normal values as a fake data set.
cat(c(PM$moments[1],PM$moments[2],PM$ratios[3],PM$ratios[4],"\n"))
# As sample size gets very large the four values returned should be
# 0,1,0,0 by definition of the standard normal distribution.

# A more complex example
para <- vec2par(c(100,500,3),type='pe3') # mean=100, sd=500, skew=3
# The Pearson type III distribution is implemented here such that
# the "parameters" are equal to the mean, standard deviation, and skew.
simDATA <- rlmomco(100,para) # simulate 100 observations
PM <- pmoms(simDATA) # compute the product moments

p.tmp <- c(PM$moments[1],PM$moments[2],PM$ratios[3])
cat(c("Sample P-moments:",p.tmp,"\n"))
# This distribution has considerable variation and large skew. Stability
# of the sample product moments requires LARGE sample sizes (too large
# for a builtin example)

# Continue the example through the L-moments
lmr <- lmoms(simDATA) # compute the L-moments
epara <- parpe3(lmr) # estimate the Pearson III parameters. This is a
```

```
# hack to back into comparative estimates of the product moments. This
# can only be done because we know that the parent distribution is a
# Pearson Type III

l.tmp <- c(epara$para[1],epara$para[2],epara$para[3])
cat(c("PearsonIII by L-moments:",l.tmp,"\n"))
# The first values are the means and will be identical and close to 100.
# The second values are the standard deviations and the L-moment to
#   PearsonIII will be closer to 500 than the product moment (this
#   shows the raw power of L-moment based analysis---they work).
# The third values are the skew. Almost certainly the L-moment estimate
#   of skew will be closer to 3 than the product moment.
```

---

pp *Plotting-Position Formula*

Description

The plotting positions of a data vector (x) are returned in ascending order. The plotting-position formula is

$$pp_i = \frac{i - a}{n + 1 - 2a},$$

where pp_i is the nonexceedance probability F of the ith ascending data value. The parameter a specifies the plotting-position type, and n is the sample size (`length(x)`). Alternatively, the plotting positions can be computed by

$$pp_i = \frac{i + A}{n + B},$$

where A and B can obviously be expressed in terms of a. The criteria $A > B > -1$ must be satisfied.

Usage

```
pp(x, A=NULL, B=NULL, a=0, sort=TRUE)
```

Arguments

x	A vector of data values. The vector is used to get sample size through `length()`;
A	A value for the plotting-position coefficient A;
B	A value for the plotting-position coefficient B;
a	A value for the plotting-position formula from which A and B are computed, default is a=0, which returns the Weibull plotting positions; and
sort	A logical whether the ranks of the data are sorted prior to F computation.

Value

An R `vector` is returned.

Note

Various plotting positions have been suggested in the literature. Stedinger and others (1992, p. 18.25) comment that "all plotting positions give crude estimates of the unknown [non]exceedance probabilities associated with the largest (and smallest) events." The various plotting positions are summarized in the follow table.

Weibull $a = 0$, Unbiased exceedance probability for all distributions

Median $a = 0.3175$, Median exceedance probabilities for all distributions

APL ≈ 0.35, Often used with probability-weighted moments

Blom $a = 0.375$, Nearly unbiased quantiles for normal distribution

Cunnane $a = 0.40$, Approximately quantile unbiased

Gringorten $a = 0.44$, Optimized for Gumbel distribution

Hazen $a = 0.50$, A traditional choice

The function uses the `rank()` function, which has specific settings to handle tied data. For implementation here, the `ties.method="first"` method to `rank()` is used.

Author(s)

W.H. Asquith

References

Stedinger, J.R., Vogel, R.M., and Foufoula-Georgiou, E., 1992, Frequency analysis of extreme events, in Handbook of Hydrology, chapter 18, editor-in-chief D. A. Maidment: McGraw-Hill, New York.

See Also

nonexceeds, pwm.pp

Examples

```
Q <- rnorm(20)
PP <- pp(Q)
plot(PP,sort(Q))

Q <- rweibull(30,1.4,scale=400)
WEI <- parwei(lmom.ub(Q))
PP <- pp(Q)
plot(PP,sort(Q))
lines(PP,quawei(PP,WEI))

# This plot looks similar, but when connecting lines are added
# the nature of the sorting is obvious.
```

```
plot(pp(Q,sort=FALSE), Q)
lines(pp(Q,sort=FALSE), Q, col=2)
```

---

prettydist *A Pretty List of Distribution Names*

Description

Return a full name of one or more distributions from the abbreviation for the distribution.

Usage

```
prettydist(x)
```

Arguments

x	A vector of **lmomco** distribution abbreviations.

Value

A vector of distribution identifiers.

Author(s)

W.H. Asquith

Examples

```
the.lst <- dist.list() # the authoritative list of abbreviations
prettydist(the.lst)
```

---

prob2T *Convert a Vector of Annual Nonexceedance Probabilities to T-year Return Periods*

Description

This function converts a vector of annual nonexceedance probabilities F to T-year return periods.

$$T = \frac{1}{1 - F}$$

Usage

```
prob2T(F)
```

Arguments

`F`	A vector of annual nonexceedance probabilities.

Value

A vector of T-year return periods.

Author(s)

W.H. Asquith

See Also

`T2prob`, `nonexceeds`

Examples

```
F <- nonexceeds()
T <- prob2T(F)
```

`pwm` *Unbiased Sample Probability-Weighted Moments*

Description

Unbiased sample Probability-Weighted Moments (PWMs) are computed from a sample. The first five β_r's are computed by default.

$$\beta_r = n^{-1} \sum_{j=1}^{n} \binom{j-1}{r} x_{j:n}$$

Usage

```
pwm(x,nmom=5,sort=TRUE)
```

Arguments

`x`	A vector of data values.
`nmom`	Number of PWMs to return.
`sort`	Does the data need sorting? The computations require sorted data. This option is provided to optimize processing speed if presorted data already exists.

Value

An R `list` is returned.

`betas`	The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	Source of the PWMs: "pwm"

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmoms`, `pwm2lmom`, `pwm`

Examples

```
# Data listed in Hosking (1995, table 29.2, p. 551)
H <- c(3,4,5,6,6,7,8,8,9,9,9,10,10,11,11,11,13,13,13,13,13,
       17,19,19,25,29,33,42,42,51.9999,52,52,52)
# 51.9999 was really 52, but a real non censored data point.
z <-  pwmRC(H,52,checkbetas=TRUE)
str(z)
# Hosking(1995) reports that A-type L-moments for this sample are
# lamA1=15.7 and lamAL-CV=.389, and lamAL-skew=.393
pwm2lmom(z$Abetas)
# My version of R reports 15.666, 0.3959, and 0.4030


# See p. 553 of Hosking (1995)
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221, 0.245, 0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,.2960001)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -.516 and lamB2 = 0.523
```

```
pwm2lmom(z$Bbetas)
# My version of R reports -.5162 and 0.5218
```

---

pwm.gev *Generalized Extreme Value Plotting-Position Probability-Weighted Moments*

Description

Generalized Extreme Value plotting-position Probability-Weighted Moments (PWMs) are computed from a sample. The first five β_r's are computed by default. The plotting-position formula for the Generalized Extreme Value distribution is

$$p_i = \frac{i - 0.35}{n},$$

where p_i is the nonexceedance probability F of the ith ascending values of the sample of size n. The PWMs are computed by

$$\beta_r = n^{-1} \sum_{i=1}^{n} p_i^r \times x_{j:n},$$

where $x_{j:n}$ is the jth order statistic $x_{1:n} \leq x_{2:n} \leq x_{j:n} \ldots \leq x_{n:n}$ of random variable X, and r is $0, 1, 2, \ldots$. Finally, `pwm.gev` dispatches to `pwm.pp(data,A=-0.35,B=0)` and does not have its own logic.

Usage

```
pwm.gev(x,nmom=5,sort=TRUE)
```

Arguments

`x`	A vector of data values.
`nmom`	Number of PWMs to return.
`sort`	Does the data need sorting? The computations require sorted data. This option is provided to optimize processing speed if presorted data already exists.

Value

An R `list` is returned.

`betas`	The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	Source of the PWMs: "pwm.gev"

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pwm.ub, pwm.pp, pwm2lmom

Examples

```
pwm <- pwm.gev(rnorm(20))
```

---

pwm.pp *Plotting-Position Sample Probability-Weighted Moments*

Description

The sample Probability-Weighted Moments (PWMs) are computed from the plotting positions of the data. The first five β_r's are computed by default. The plotting-position formula for a sample size of n is

$$p_i = \frac{i + A}{n + B},$$

where pp_i is the nonexceedance probability F of the ith ascending data values. An alternative form of the plotting position equation is

$$p_i = \frac{i + a}{n + 1 - 2a},$$

where a is a single plotting position coefficient. Having a provides A and B, therefore the parameters A and B together specify the plotting-position type. The PWMs are computed by

$$\beta_r = n^{-1} \sum_{i=1}^{n} p_i^r \times x_{j:n},$$

where $x_{j:n}$ is the jth order statistic $x_{1:n} \leq x_{2:n} \leq x_{j:n} \ldots \leq x_{n:n}$ of random variable X, and r is $0, 1, 2, \ldots$.

Usage

```
pwm.pp(x, pp=NULL, A=NULL, B=NULL, a=0, nmom=5, sort=TRUE)
```

Arguments

- `x` A vector of data values;
- `pp` An optional vector of nonexceedance probabilities. If present then `A` and `B` or `a` are ignored;
- `A` A value for the plotting-position formula. If `A` and `B` are both zero then the unbiased PWMs are computed through `pwm.ub`;
- `B` Another value for the plotting-position formula. If `A` and `B` are both zero then the unbiased PWMs are computed through `pwm.ub`;
- `a` A single plotting position coefficient from which, if not `NULL`, A and B will be internally computed;
- `nmom` Number of PWMs to return;
- `sort` Does the data need sorting? The computations require sorted data. This option is provided to optimize processing speed if presorted data already exists.

Value

An R `list` is returned.

- `betas` The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
- `source` Source of the PWMs: "pwm.pp"

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`pwm.ub`, `pwm.gev`, `pwm2lmom`

Examples

```
pwm <- pwm.pp(rnorm(20), A=-0.35, B=0)

X <- rnorm(20)
pwm <- pwm.pp(X, pp=pp(X)) # weibull plotting positions
```

pwm.ub *Unbiased Sample Probability-Weighted Moments*

Description

Unbiased sample Probability-Weighted Moments (PWMs) are computed from a sample. The first five β_r's are computed by default.

$$\beta_r = n^{-1} \binom{n-1}{r}^{-1} \sum_{j=1}^{n} \binom{j-1}{r} x_{j:n}$$

Usage

```
pwm.ub(x,nmom=5,sort=TRUE)
```

Arguments

`x`	A vector of data values.
`nmom`	Number of PWMs to return.
`sort`	Does the data need sorting? The computations require sorted data. This option is provided to optimize processing speed if presorted data already exists.

Value

An R `list` is returned.

`betas`	The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	Source of the PWMs: "pwm.ub"

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

pwm.pp, pwm.gev, pwm2lmom

Examples

```
pwm <- pwm.ub(rnorm(20))
```

---

pwm2lmom *Probability-Weighted Moments to L-moments*

Description

Converts the Probability-Weighted Moments (PWM) to the L-moments given the PWM. The conversion is linear so procedures based on PWMs and identical to those based on L-moments.

$$\lambda_1 = \beta_0,$$

$$\lambda_2 = 2\beta_1 - \beta_0,$$

$$\lambda_3 = 6\beta_2 - 6\beta_1 + \beta_0,$$

$$\lambda_4 = 20\beta_3 - 30\beta_2 + 12\beta_1 - \beta_0,$$

$$\lambda_5 = 70\beta_4 - 140\beta_3 + 90\beta_2 - 20\beta_1 + \beta_0,$$

$$\tau = \lambda_2/\lambda_1,$$

$$\tau_3 = \lambda_3/\lambda_2,$$

$$\tau_4 = \lambda_4/\lambda_2, \text{ and}$$

$$\tau_5 = \lambda_5/\lambda_2.$$

The general expression and the expression used for computation if the argument is a vector of PWMs is

$$\lambda_{r+1} = \sum_{k=0}^{r} (-1)^{r-k} \binom{r}{k} \binom{r+k}{k} \beta_{k+1}$$

Usage

```
pwm2lmom(pwm)
```

Arguments

pwm	A PWM object created by pwm.ub or similar.

Details

The Probability Weighted Moments (PWMs) are linear combinations of the L-moments and therefore contain the same statistical information of the data as the L-moments. However, the PWMs are harder to interpret as measures of probability distributions. The linearity between L-moments and PWMs means that procedures base on one are equivalent to the other.

The function can take a variety of PWM argument types in `pwm`. The function checks whether the argument is a list and if so attempts to extract the β_r's from `list` names such as `BETA0`, `BETA1`, and so on. If the extraction is successful, then a list of L-moments similar to `lmom.ub` is returned. If the extraction was not successful, then a `list` name `betas` is checked; if `betas` is found then this vector of PWMs is used to compute the L-moments. If `pwm` is a `list` but can not be routed in the function, a `warnings` is made and `NULL` returned. If the `pwm` argument is a `vector`, then this vector of PWMs is used to compute the L-moments are returned.

Value

One of two R `list` are returned.

`Version I`
: **L1** Arithmetic mean
: **L2** L-scale—analogous to standard deviation
: **LCV** coefficient of L-variation—analogous to coe. of variation
: **TAU3** The third L-moment ratio or L-skew—analogous to skew
: **TAU4** The fourth L-moment ratio or L-kurtosis—analogous to kurtosis
: **TAU5** The fifth L-moment ratio
: **L3** The third L-moment
: **L4** The fourth L-moment
: **L5** The fifth L-moment

`Version II`
: **lambdas** The L-moments
: **ratios** The L-moment ratios
: **source** Source of the L-moments (`pwm2lmom`)

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`lmom.ub`, `pwm.ub`, `pwm`, `lmom2pwm`

Examples

```
D <- c(123,34,4,654,37,78)
pwm2lmom(pwm.ub(D))
pwm2lmom(pwm(D))

pwm2lmom(pwm(rnorm(100)))
```

---

pwm2vec *Convert Probability-Weighted Moment object to a Vector of Probability-Weighted Moments*

Description

This function converts a Probability-Weighted Moment object in the structure used by **lmomco** into a simple vector of β_0, β_1, β_2, β_3, β_4.

Usage

```
pwm2vec(pwm, ...)
```

Arguments

pwm	Probability-Weighted Moment object as from functions such as pwm and vec2pwm.
...	Not presently used.

Value

A vector of the first five Probability-Weighted Moments if available. The $betas field of the pwm argument is simply returned by this function.

Author(s)

W.H. Asquith

See Also

pwm and vec2pwm

Examples

```
pmr <- pwm(rnorm(40))
pwm2vec(pmr)


pmr <- vec2pwm(c(140,150,45,21))
pwm2vec(pmr)
```

pwmLC — *Sample Probability-Weighted Moments for Left-Tail Censoring*

Description

Compute the sample Probability-Weighted Moments (PWMs) for left-tail censored data set—that is a data set censored from below. The censoring threshold is denoted as T.

Usage

```
pwmLC(x, threshold=NULL, nmom=5, sort=TRUE)
```

Arguments

`x`	A vector of data values.
`threshold`	The left-tail censoring (lower) threshold.
`nmom`	Number of PWMs to return.
`sort`	Does the data need sorting? Note that convention is the have a β'_0, but this is placed in the first index `i=1` of the `betas` vector.

Details

There is some ambiguity if the threshold also numerically equals valid data in the data set. In the data for the examples below, which are taken from elsewhere, there are real observations at the censoring level. One can see how a hack is made to marginally decrease or increase the data or the threshold for the computations. This is needed because the code uses `sapply(x,function(v) { if(v >= T) return(T); re` to reset the data vector `x`. By operating on the data in this fashion one can toy with various levels of the threshold for experimental purposes; this seemed a more natural way for general implementation. The code sets n=`length(x)` and m=`n - length(x[x == T])`, which also seems natural. The β^A_r are computed by dispatching to `pwm`.

Value

An R `list` is returned.

`Aprimebetas`	The A'-type PWMs. These should be same as `pwm()` returns if there is no censoring. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`Bprimebetas`	The B'-type PWMs. These should be `NA` if there is no censoring. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	Source of the PWMs: "pwmLC"
`threshold`	The upper censoring threshold.
`zeta`	The right censoring fraction: `numbelowthreshold/samplesize`

numbelowthreshold
: Number of data points equal to or above the threshold.

observedsize
: Number of real data points in the sample (above the threshold).

samplesize
: Number of actual sample values.

Author(s)

W.H. Asquith

References

Zafirakou-Koulouris, A., Vogel, R.M., Craig, S.M., and Habermeier, J., 1998, L-moment diagrams for censored observations: Water Resources Research, v. 34, no. 5, pp. 1241-1249.

See Also

lmoms, pwm2lmom, pwm, pwmRC

Examples

```
#
```

---

pwmRC — *Sample Probability-Weighted Moments for Right-Tail Censoring*

Description

Compute the sample Probability-Weighted Moments (PWMs) for right-tail censored data set—that is a data set censored from above. The censoring threshold is denoted as T. The data possess m values that are observed (noncensored, $< T$) out of a total of n samples. The ratio of m to n is defined as $\zeta = m/n$, which will play an important role in parameter estimation. The ζ is interpreted as the probability $\mathrm{Pr}\{\}$ that x is less than the quantile at ζ nonexceedance probability: $(\mathrm{Pr}\{x < X(\zeta)\})$. Two types of PWMs are computed for right-tail censored situations. The "A"-type PWMs and "B"-type PWMs. The A-type PWMs are defined by

$$\beta_r^A = m^{-1} \sum_{j=1}^{m} \binom{j-1}{r} x_{[j:n]},$$

which are the PWMs of the uncensored sample of m observed values. The B-type PWMs are computed from the "complete" sample, in which the $n - m$ censored values are replaced by the censoring threshold T. The B-type PWMs are defined by

$$\beta_r^B = n^{-1} \left(\sum_{j=1}^{m} \binom{j-1}{r} x_{[j:n]} + \sum_{j=m+1}^{n} \binom{j-1}{r} T \right).$$

The two previous expressions are used in the function. These PWMs are readily converted to L-moments by the usual methods (pwm2lmom). When there are more than a few censored values, the PWMs are readily computed by computing β_r^A and using the expression

$$\beta_r^B = Z\beta_r^A + \frac{1-Z}{r+1}T,$$

where

$$Z = \frac{m}{n}\frac{\binom{m-1}{r}}{\binom{n-1}{r}}.$$

The two expressions above are consulted when the `checkbetas=TRUE` argument is present. Both sequences of B-type are `cat`ed to the terminal. This provides a check on the implementation of the algorithm. The functions Apwm2BpwmRC and Bpwm2ApwmRC can be used to switch back and forth between the two PWM types given fitted parameters for a distribution in the **lmomco** package that supports right-tail censoring. Finally, the `RC` in the function name is to denote `Right`-tail `Censoring`.

Usage

```
pwmRC(x,threshold=NULL,nmom=5,sort=TRUE,checkbetas=FALSE)
```

Arguments

`x`	A vector of data values.
`threshold`	The right-tail censoring (upper) threshold.
`nmom`	Number of PWMs to return.
`sort`	Does the data need sorting? Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`checkbetas`	A cross relation between β_r^A and β_r^B exists—display the results of the secondary computation of the β_r^B. The two displayed vectors should be numerically equal.

Details

There is some ambiguity if the threshold also numerically equals valid data in the data set. In the data for the examples below, which are taken from elsewhere, there are real observations at the censoring level. One can see how a hack is made to marginally decrease or increase the data or the threshold for the computations. This is needed because the code uses `sapply(x,function(v) { if(v >= T) return(T); re` to reset the data vector `x`. By operating on the data in this fashion one can toy with various levels of the threshold for experimental purposes; this seemed a more natural way for general implementation. The code sets n=`length(x)` and m=`n - length(x[x == T])`, which also seems natural. The β_r^A are computed by dispatching to pwm.

Value

An R `list` is returned.

Abetas	The A-type PWMs. These should be same as pwm() returns if there is no censoring. Note that convention is the have a β_0, but this is placed in the first index i=1 of the betas vector.
Bbetas	The B-type PWMs. These should be NA if there is no censoring. Note that convention is the have a β_0, but this is placed in the first index i=1 of the betas vector.
source	Source of the PWMs: "pwmRC"
threshold	The upper censoring threshold.
zeta	The right censoring fraction: numabovethreshold/samplesize
numabovethreshold	Number of data points equal to or above the threshold.
observedsize	Number of real data points in the sample (below the threshold).
samplesize	Number of actual sample values.

Author(s)

W.H. Asquith

References

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressable in inverse form: Water Resources Research, vol. 15, p. 1,049–1,054.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

lmoms, pwm2lmom, pwm, pwmLC

Examples

```
# Data listed in Hosking (1995, table 29.2, p. 551)
H <- c(3,4,5,6,6,7,8,8,9,9,9,10,10,11,11,11,13,13,13,13,13,
       17,19,19,25,29,33,42,42,51.9999,52,52,52)
# 51.9999 was really 52, a real (noncensored) data point.
z <-  pwmRC(H,threshold=52,checkbetas=TRUE)
str(z)
# Hosking(1995) reports that A-type L-moments for this sample are
# lamA1=15.7 and lamAL-CV=.389, and lamAL-skew=.393
pwm2lmom(z$Abetas)
# My version of R reports 15.666, 0.3959, and 0.4030


# See p. 553 of Hosking (1995)
```

```
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221,  0.245,  0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,.2960001)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -.516 and lamB2 = 0.523
pwm2lmom(z$Bbetas)
# My version of R reports -.5162 and 0.5218
```

---

qlmomco *Quantile Function of the Distributions*

Description

This function acts as an alternative front end to `par2qua`. The nomenclature of the `qlmomco` function is to mimic that of built-in **R** functions.

Usage

```
qlmomco(f,para)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `lmom2par` or similar.

Value

Quantile value for F for the specified parameters.

Author(s)

W.H. Asquith

See Also

`dlmomco`, `plmomco`, `rlmomco`

Examples

```
para <- vec2par(c(0,1),type='nor') # standard normal parameters
p75  <- qlmomco(.75,para) # 75th percentile of one standard deviation
```

qua.ostat *Compute the Quantiles of the Distribution of an Order Statistic*

Description

This function computes a specified quantile by nonexceedance probability F for the jth-order statistic of a sample of size n for a given distribution. Let the quantile function (inverse distribution) of the Beta distribution be

$$\mathrm{B}^{-1}(F, j, n - j + 1),$$

and let $x(F, \Theta)$ represent the quantile function of the given distribution and Θ represents a vector of distribution parameters. The quantile function of the distribution of the jth-order statistic is

$$x(\mathrm{B}^{-1}(F, j, n - j + 1), \Theta).$$

Usage

```
qua.ostat(f,j,n,para=NULL)
```

Arguments

`f`	The nonexceedance probability F for the quantile.
`j`	The jth-order statistic $x_{1:n} \leq x_{2:n} \leq \ldots \leq x_{j:n} \leq x_{n:n}$.
`n`	The sample size.
`para`	A distribution parameter list from a function such as `vec2par` or `lmom2par`.

Value

The quantile of the distribution of the jth-order statistic is returned.

Author(s)

W.H. Asquith

References

Gilchrist, W.G., 2000, Statistical modelling with quantile functions: Chapman and Hall/CRC, Boca Raton, Fla.

See Also

`lmom2par`, `vec2par`

Examples

```
gpa <- vec2par(c(100,500,0.5),type='gpa')
n <- 20   # the sample size
j <- 15   # the 15th order statistic
F <- 0.99 # the 99th percentile
theoOstat <- qua.ostat(F,j,n,gpa)

# Let us test this value against a brute force estimate.
Jth <- vector(mode = "numeric")
for(i in seq(1,10000)) {
  Q <- sort(rlmomco(n,gpa))
  Jth[i] <- Q[j]
}
bruteOstat <- quantile(Jth,F) # estimate by built-in function
theoOstat <- signif(theoOstat,digits=5)
bruteOstat <- signif(bruteOstat,digits=5)
cat(c("Theoretical=",theoOstat,"  Simulated=",bruteOstat,"\n"))
```

---

qua2ci — *Estimate Confidence Interval for a Single Quantile of a Parent Distribution*

Description

This function estimates the lower and upper limits of a specified confidence interval for an aribitrary quantile value of a specified parent distribution [quantile function $Q(F,\theta)$ with parameters θ] using Monte Carlo simulation. The quantile value, actually the nonexceedance probability (F for $0 \leq F < 1$) of the value, is specified by the user. The user also provides the parameters of the parent distribution (see `lmom2par`).

For `nsim` simulation runs (ideally a very large number), samples of size n are drawn from $Q(F,\theta)$ using the R-function `runif(n)` to generate the simulated F values and the **lmomco** function `par2qua` function to compute the quantile. The L-moments of the simulated sample are computed using `lmoms` and a distribution of the same type as the parent is fit to these simulated L-moments. The F-quantile of the just-fitted distribution is computed and placed into a vector. The process of simulating the sample, computing the L-moments, computing the parameters, and solving for the F-quantile is repeated for the specified number of simulation runs.

To estimate the confidence interval, the L-moments of the vector simulated quantiles are computed. Subsequently, the parameters of a user-specified distribution "error" distribution are computed. The two quantiles of this error distribution for the specified confidence interval are computed. These two quantiles represent the estimated lower and upper limits for the confidence interval of the parent distribution for samples of size n.

Finally, it is often useful to have vectors of lower and upper limits for confidence intervals for a vector of F values. The function `genci` does just that and uses the `qua2ci` function as the computational engine.

Usage

```
qua2ci(f, para, n, ci=0.90, edist='nor', nsim=1000,
verbose=FALSE, showpar=FALSE, maxlogdiff=6)
```

Arguments

- `f` — Nonexceedance probability ($0 \leq F \leq 1$) of the quantile for which the confidence interval is needed.
- `para` — The parameters from `lmom2par` or similar.
- `n` — The sample size that the Monte Carlo simulation will use.
- `ci` — The confidence interval ($0.5 \leq \mathtt{ci} < 1$). The interval is specified as the size of the interval. The default is 0.90 or the 90th percentile. The function will return the 5th (1-0.90)/2 and 95th (1-(1-0.90)/2) percentile cumulative probability of the error distribution for the parent quantile as specified by the nonexceedance probability argument. The arguments `ci` and `f` therefore are separate features.
- `edist` — The model for the error distribution. Although the normal (the default) is typically assumed in error analyses, it need not be, as support for other distributions supported by the **lmomco** package is available. However, one should seriously consider the values of the simulated L-moments when choosing an error distribution other than the normal. If the L-skew (τ_4) or L-kurtosis (τ_4) values depart considerably from those of the normal ($\tau_3 = 0$ and $\tau_4 = 0.122602$), alternative distributions would likely provide more reliable confidence interval estimation.
- `nsim` — The number of simulations for the sample size `n` to perform. Large numbers produce more refined confidence limit estimates at the cost of CPU time. The default is anticipated to be large enough for evaluative-useage without too much computational delay. Much larger simulation numbers are highly recommended.
- `verbose` — The verbosity of the operation of the function.
- `showpar` — The parameters of the `edist` for each simulation are printed.
- `maxlogdiff` — The maximum permitted difference in log10 space between a simulated quantile and the true value. It is possible that a well fit simulated sample to the parent distribution type provides crazy quantile estimates in the far reaches of either tail. The default value of 6 was chosen based on experience with the Kappa distribution fit to a typical heavy-right tail flood magnitude data set. However, the choice of 6-log cycles is ad hoc at best; users are encouraged to do their own exploration. If `verbose=TRUE` then a message will be printed when the `maxlogdiff` condition is tripped.

Value

An R `list` is returned.

- `lower` — The lower value of the confidence interval having nonexceedance probability equal to (1-`ci`)/2.
- `true` — The value returned by `par2qua(f,para)`.
- `upper` — The upper value of the confidence interval having nonexceedance probability equal to 1-(1-`ci`)/2.

elmoms	The L-moments of the distribution of simulated of quantiles.
epara	The parameters of the error distribution fit using the elmoms.
ifail	A diagnostic value. A value of zero means that successful exit was made.
ifailtext	A descriptive message related to the ifail value.

Author(s)

W.H. Asquith

See Also

lmoms, lmom2par, par2qua, genci

Examples

```
# It is well known that standard deviation (sigma) of the
# sample mean is equal to sigma/sample_size.
MEAN  <- 0
SIGMA <- 100
PAR   <- vec2par(c(MEAN,SIGMA),type='nor')
CI    <- qua2ci(0.5,PAR,n=10,nsim=20) # F=0.5-->median=mean
# Theoretrical sample mean sigma = 100/10 = 10
# L-moment theory: L-scale*sqrt(pi) = sigma
# Thus, it follows that
CI$elmoms$lambdas[2]/sqrt(pi)
# approaches 10 as nsim --> Inf.


# Another example.
D   <- c(123,34,4,654,37,78, 93, 95, 120) # fake sample
lmr <- lmoms(D)    # compute the lmoments of the data
WEI <- parwei(lmr) # estimate parameters of the weibull
CI  <- qua2ci(0.75,WEI,20,nsim=20,ci=0.95)
# CI contains the estimate 95-percent
# confidence interval for the 75th-percent of the parent
# weibull distribution for size 20 samples from the parent
```

---

quaaep4 *Quantile Function of the 4-p Asymmetric Exponential Power Distribution*

Description

This function computes the quantiles of the 4-parameter Asymmetric Exponential Power distribution given parameters (ξ, α, κ, and h) of the distribution computed by paraep4. The quantile function of the distribution given the cumulative distribution function $F(x)$ is

For $F < F(\xi)$,

$$x(F) = \xi - \alpha\kappa[\gamma^{(-1)}((1+\kappa^2)F/\kappa^2,\ 1/h)]^{1/h}$$

and for $F \geq F(\xi)$,

$$x(F) = \xi + \frac{\alpha}{\kappa}[\gamma^{(-1)}((1+\kappa^2)(1-F),\ 1/h)]^{1/h}$$

where $x(F)$ is the quantile x for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, κ is a shape parameter, h is another shape parameter, $\gamma^{(-1)}(Z, shape)$ is the inverse of the upper tail of the incomplete gamma function. The range of the distribution is $-\infty < x < \infty$. The inverse upper tail of the incomplete gamma function is `qgamma(Z, shape, lower.tail=FALSE)` in R. The mathematical definition of the upper tail of the incomplete gamma function shown in documentation for `cdfaep4`.

Usage

```
quaaep4(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `paraep4` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Ayebo, A., and Kozubowski, T.J., 2003, An asymmetric generalization of Gaussian and Laplace laws: Journal of Probability and Statistical Science, v. 1, no. 2, pp. 187-210.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661-1673.

See Also

`pdfaep4`, `cdfaep4`, `paraep4`

Examples

```
para <- vec2par(c(0,1, 0.5, 2), type="aep4");
IQR <- quaaep4(0.75,para) - quaaep4(0.25,para);
cat("Interquartile Range=",IQR,"\n")

## Not run:
F <- c(0.00001, 0.0001, 0.001, seq(0.01, 0.99, by=0.01),
       0.999, 0.9999, 0.99999);
delx <- 0.1;
x <- seq(-10,10, by=delx);
K <- .67

PAR <- list(para=c(0,1, K, 0.5), type="aep4");
plot(x,cdfaep4(x, PAR), type="n",
     ylab="NONEXCEEDANCE PROBABILITY",
     ylim=c(0,1), xlim=c(-20,20));
lines(x,cdfaep4(x,PAR), lwd=3);
lines(quaaep4(F, PAR), F, col=4);

PAR <- list(para=c(0,1, K, 1), type="aep4");
lines(x,cdfaep4(x, PAR), lty=2, lwd=3);
lines(quaaep4(F, PAR), F, col=4, lty=2);

PAR <- list(para=c(0,1, K, 2), type="aep4");
lines(x,cdfaep4(x, PAR), lty=3, lwd=3);
lines(quaaep4(F, PAR), F, col=4, lty=3);

PAR <- list(para=c(0,1, K, 4), type="aep4");
lines(x,cdfaep4(x, PAR), lty=4, lwd=3);
lines(quaaep4(F, PAR), F, col=4, lty=4);

## End(Not run)
```

---

quacau *Quantile Function of the Cauchy Distribution*

Description

This function computes the quantiles of the Cauchy distribution given parameters (ξ and α) of the distribution provided by vec2par. The quantile function of the distribution is

$$x(F) = \xi + \alpha \times \tan(\pi(F - 0.5)),$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter and α is a scale parameter. R supports the quantile function of the Cauchy distribution through qcauchy. This function does not use qcauchy because qcauchy does not return Inf for $F = 1$ although it returns -Inf for $F = 0$.

Usage

```
quacau(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \le F \le 1$).
para	The parameters from parcau or vec2par.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

Gilchirst, W.G., 2000, Statistical modeling with quantile functions: Chapman and Hall/CRC, Boca Raton, FL.

See Also

cdfcau, parcau, vec2par

Examples

```
para <- c(12,12)
quacau(.5,vec2par(para,type='cau'))
```

---

quaexp *Quantile Function of the Exponential Distribution*

Description

This function computes the quantiles of the Exponential distribution given parameters (ξ and α) of the distribution computed by parexp. The quantile function of the distribution is

$$x(F) = \xi - \alpha \log(1 - F),$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter and α is a scale parameter.

Usage

```
quaexp(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parexp or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfexp, parexp

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quaexp(0.5,parexp(lmr))
```

quagam *Quantile Function of the Gamma Distribution*

Description

This function computes the quantiles of the Gamma distribution given parameters (α and β) of the distribution computed by pargam. The quantile function has no explicit form. See the qgamma function and cdfgam. The parameters have the following interpretations: α is a shape parameter and β is a scale parameter in the R syntax.

Usage

```
quagam(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from pargam or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgam, pargam

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
g <- pargam(lmr)
quagam(0.5,g)

# generate 50 random samples from this fitted parent
Qsim <- rlmomco(5000,g)
# compute the apparent gamma parameter for this parent
gsim <- pargam(lmoms(Qsim))
```

---

quagev *Quantile Function of the Generalized Extreme Value Distribution*

Description

This function computes the quantiles of the Generalized Extreme Value distribution given parameters (ξ, α, and κ) of the distribution computed by pargev. The quantile function of the distribution is

$$x(F) = \xi + \frac{\alpha}{\kappa}\left(1 - (-\log(F))^{\kappa}\right) \text{ for } \kappa \neq 0 \text{ and}$$

$$x(F) = \xi - \alpha\log(-\log(F)) \text{ for } \kappa = 0,$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
quagev(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \le F \le 1$).
para	The parameters from pargev or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgev, pargev

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quagev(0.5,pargev(lmr))
```

---

quagld *Quantile Function of the Generalized Lambda Distribution*

Description

This function computes the quantiles of the Generalized Lambda distribution given parameters (ξ, α, κ, and h) of the distribution computed by vec2par. The quantile function of the distribution is

$$x(F) = \xi + \alpha(F^{\kappa} - (1 - F)^{h}),$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, and κ, and h are shape parameters. Note that in this parameterization the scale term is shown in the numerator and not the denominator. This is done in this package as part of the parallel nature between distributions. The scale parameter is often shown have same units and the location parameter.

Usage

```
quagld(f, gldpara, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
gldpara	The parameters from vec2par or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Karian, Z.A., and Dudewicz, E.J., 2000, Fitting statistical distributions—The generalized lambda distribution and generalized bootstrap methods: CRC Press, Boca Raton, FL, 438 p.

See Also

cdfgld, pargld, lmomgld, lmomTLgld,
pargld, parTLgld

Examples

```
para <- vec2par(c(123,34,4,3),type="gld")
quagld(0.5,para)
```

---

quaglo *Quantile Function of the Generalized Logistic Distribution*

Description

This function computes the quantiles of the Generalized Logistic distribution given parameters (ξ, α, and κ) of the distribution computed by parglo. The quantile function of the distribution is

$$x(F) = \xi + \frac{\alpha}{\kappa}\left(1 - \left(\frac{1-F}{F}\right)^{\kappa}\right) \text{ for } \kappa \neq 0 \text{ and}$$

$$x(F) = \xi - \alpha \log\left(\frac{1-F}{F}\right) \text{ for } \kappa = 0,$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
quaglo(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parglo or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfglo, parglo

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quaglo(0.5,parglo(lmr))
```

---

quagno *Quantile Function of the Generalized Normal Distribution*

Description

This function computes the quantiles of the Generalized Normal (log-Normal) distribution given parameters (xi, α, and κ) of the distribution computed by pargno. The quantile function of the distribution has no explicit form. The parameters have the following interpretations: ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
quagno(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `pargno` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfgno`, `pargno`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quagno(0.5,pargno(lmr))
```

---

quagpa — *Quantile Function of the Generalized Pareto Distribution*

Description

This function computes the quantiles of the Generalized Pareto distribution given parameters (ξ, α, and κ) of the distribution computed by `pargpa`. The quantile function of the distribution is

$$x(F) = \xi + \frac{\alpha}{\kappa}\left(1 - (1-F)^{\kappa}\right) \text{ for } \kappa \neq 0 \text{ and}$$

$$x(F) = \xi - \alpha \log(1-F) \text{ for } \kappa = 0,$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, and κ is a shape parameter.

Usage

```
quagpa(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \le F \le 1$).
para	The parameters from pargpa or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfgpa, pargpa

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quagpa(0.5,pargpa(lmr))
```

`quagum` *Quantile Function of the Gumbel Distribution*

Description

This function computes the quantiles of the Gumbel distribution given parameters (ξ and α) of the distribution computed by `pargum`. The quantile function of the distribution is

$$x(F) = \xi - \alpha \log(-\log(F)),$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, and α is a scale parameter.

Usage

```
quagum(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `pargum` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfgum`, `pargum`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quagum(0.5,pargum(lmr))
```

---

quakap *Quantile Function of the Kappa Distribution*

Description

This function computes the quantiles of the Kappa distribution given parameters (ξ, α, κ, and h) of the distribution computed by parkap. The quantile function of the distribution is

$$x(F) = \xi + \frac{\alpha}{\kappa}\left(1 - \left(\frac{1-F^h}{h}\right)^{\kappa}\right),$$

where $x(F)$ is the quantile for nonexceedance probability f, ξ is a location parameter, α is a scale parameter, κ is a shape parameter, and h is another shape parameter.

Usage

```
quakap(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parkap or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfkap, parkap

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78,21,32,231,23))
quakap(0.5,parkap(lmr))
```

---

quakur *Quantile Function of the Kumaraswamy Distribution*

Description

This function computes the quantiles $0 < x < 1$ of the Kumaraswamy distribution given parameters (α and β) of the distribution computed by parkur. The quantile function of the distribution is

$$x(F) = (1 - (1 - F)^{1/\beta})^{1/\alpha},$$

where $x(F)$ is the quantile for nonexceedance probability F, α is a shape parameter, and β is a shape parameter.

Usage

```
quakur(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parkur or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v.6, pp. 70–81.

See Also

cdfkur, parkur

Examples

```
lmr <- lmom.ub(c(0.25, 0.4, 0.6, 0.65, 0.67, 0.9))
quakur(0.5,parkur(lmr))
```

---

qualap *Quantile Function of the Laplace Distribution*

Description

This function computes the quantiles of the Laplace distribution given parameters (ξ and α) of the distribution computed by parlap. The quantile function of the distribution is

$$x(F) = \xi + \alpha \times \log(2F) \text{ for } F \leq 0.5\text{, and}$$

$$x(F) = \xi - \alpha \times \log(2(1-F)) \text{ for } F > 0.5\text{,}$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter and α is a scale parameter.

Usage

```
qualap(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parlap or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: IBM Research Report RC12210, T.J. Watson Research Center, Yorktown Heights, New York.

See Also

`cdflap`, `parlap`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
qualap(0.5,parlap(lmr))
```

---

qualn3 *Quantile Function of the 3-Parameter Log-Normal Distribution*

Description

This function computes the quantiles of the Log-Normal3 distribution given parameters ($zeta$, μ, and σ) of the distribution computed by `parln3`. The quantile function of the distribution has no explicit form. The parameters have the following interpretations: ζ is a lower bounds parameter, μ is a location parameter, and σ is a scale parameter.

Usage

```
qualn3(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `parln3` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

NEED

See Also

`cdfln3`, `parln3`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
qualn3(0.5,parln3(lmr))
```

---

quanor *Quantile Function of the Normal Distribution*

Description

This function computes the quantiles of the Normal distribution given parameters ($mean$ and sd) of the distribution computed by `parnor`. The quantile function of the distribution has no explicit form (see `cdfnor` and `qnorm`). The parameters have the following interpretations: $mean$ is the arithmetic mean and sd is the standard deviation. The R-function `qnorm` is used.

Usage

```
quanor(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`para`	The parameters from `parnor` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfnor`, `parnor`, `quagno`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quanor(0.5,parnor(lmr))
```

---

quape3 *Quantile Function of the Pearson Type III Distribution*

Description

This function computes the quantiles of the Pearson Type III distribution given parameters (μ, σ, and γ) of the distribution computed by `parpe3`. The quantile function of the distribution has no explicit form (see `cdfpe3`).

For the implementation in the **lmomco** package, the three parameters are μ, σ, and γ for the mean, standard deviation, and skew, respectively. Therefore, the Pearson Type III distribution is of considerable theoretical interest to this package because the parameters, which are estimated via the L-moments, are in fact the product moments. Although, these values fitted by the method of L-moments will not be numerically equal to the sample product moments. Further details are provided in the examples section of the `pmoms` function documentation.

Usage

```
quape3(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \le F \le 1$).
`para`	The parameters from `parpe3` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

cdfpe3, parpe3

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quape3(0.5,parpe3(lmr))
```

---

quaray *Quantile Function of the Rayleigh Distribution*

Description

This function computes the quantiles of the Rayleigh distribution given parameters (ξ and α) of the distribution computed by parray. The quantile function of the distribution is

$$x(F) = \xi + \sqrt{-2\alpha^2 \log(1 - F)},$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, and α is a scale parameter.

Usage

```
quaray(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parray or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorkton Heights, N.Y.

See Also

cdfray, parray

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quaray(0.5,parray(lmr))
```

---

quarevgum *Quantile Function of the Reverse Gumbel Distribution*

Description

This function computes the quantiles of the Reverse Gumbel distribution given parameters (ξ and α) of the distribution computed by parrevgum. The quantile function of the distribution is

$$x(F) = \xi + \alpha \log(-\log(1 - F)),$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, and α is a scale parameter. Notice that the function has some sign differences and uses the complement of F compared to the Gumbel quantile function in quagum.

Usage

```
quarevgum(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parrevgum or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data, in Recent Advances in Life-Testing and Reliability, edited by N. Balakrishnan, chapter 29, CRC Press, Boca Raton, Fla., pp. 546–560.

See Also

cdfrevgum, parrevgum

Examples

```
# See p. 553 of Hosking (1995)
# Data listed in Hosking (1995, table 29.3, p. 553)
D <- c(-2.982, -2.849, -2.546, -2.350, -1.983, -1.492, -1.443,
       -1.394, -1.386, -1.269, -1.195, -1.174, -0.854, -0.620,
       -0.576, -0.548, -0.247, -0.195, -0.056, -0.013,  0.006,
        0.033,  0.037,  0.046,  0.084,  0.221,  0.245,  0.296)
D <- c(D,rep(.2960001,40-28)) # 28 values, but Hosking mentions
                              # 40 values in total
z <-  pwmRC(D,threshold=.2960001)
str(z)
# Hosking reports B-type L-moments for this sample are
# lamB1 = -.516 and lamB2 = 0.523
btypelmoms <- pwm2lmom(z$Bbetas)
# My version of R reports lamB1 = -0.5162 and lamB2 = 0.5218
str(btypelmoms)
rg.pars <- parrevgum(btypelmoms,z$zeta)
str(rg.pars)
# Hosking reports xi = 0.1636 and alpha = 0.9252 for the sample
# My version of R reports xi = 0.1635 and alpha = 0.9254
F  <- nonexceeds()
PP <- pp(D) # plotting positions of the data
plot(PP,sort(D),ylim=range(quarevgum(F,rg.pars)))
lines(F,quarevgum(F,rg.pars))
# In the plot notice how the data flat lines at the censoring level,
# but the distribution continues on.  Neat.
```

---

quarice *Quantile Function of the Rice Distribution*

Description

This function computes the quantiles of the Rice distribution given parameters (ν and α) of the distribution computed by parrice. The quantile function of the distribution is complex and numerical rooting of the cumulative distribution function is used.

$$x(F, \nu, \alpha)$$

where $x(F)$ is the quantile for nonexceedance probability F, ν is a parameter, and ν/α is a form of signal-to-noise ratio SNR. If $\nu = 0$, then the Rayleigh distribution results and quaray is used.

Usage

```
quarice(f, para, xmax=NULL, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from parrice or similar.
xmax	The maximum x value used for integeration.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

See Also

cdfrice, parrice

Examples

```
lmr <- vec2lmom(c(125,0.20), lscale=FALSE)
quarice(0.75,parrice(lmr))


# The quantile function of the Rice as implemented in lmomco
# is slow because of rooting the CDF, which is created by
# integration of the PDF. Rician random variates are easily created.
# Thus, in speed applications the rlmomco() with a Rice parameter
# object could be bypassed by the following function, rrice().
## Not run:
"rrice" = function(n, nu, alpha) { # from the VGAM package
    theta = 1 # any number
    X = rnorm(n, mean=nu * cos(theta), sd=alpha)
    Y = rnorm(n, mean=nu * sin(theta), sd=alpha)
    return(sqrt(X^2 + Y^2))
}
n <- 500; # suggest making it about 10,000
nu <- 100; alpha <- 10
lmoms(rrice(n, nu, alpha))
lmoms(rlmomco(n, vec2par(c(nu,alpha), type='rice')))
```

```
## End(Not run)
```

quatexp *Quantile Function of the Truncated Exponential Distribution*

Description

This function computes the quantiles of the Truncated Exponential distribution given parameters (ξ and α) of the distribution computed by partexp. The quantile function of the distribution is

$$x(F) = -\alpha \log[1 - F(1 - \exp(-\xi/\alpha))],$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α is a scale parameter, $0 \leq x \leq \xi$. The distribution has $0 < \tau_2 <= 1/2$, $\xi > 0$, and $1/\alpha \neq 0$.

Usage

```
quatexp(f, para, paracheck=TRUE)
```

Arguments

f	Nonexceedance probability ($0 \leq F \leq 1$).
para	The parameters from partexp or similar.
paracheck	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Vogel, R.M., Hosking, J.R.M., Elphick, C.S., Roberts, D.L., and Reed, J.M., 2008, Goodness of fit of probability distributions for sightings as species approach extinction: Bulletin of Mathematial Biology, v. 71, no. 3, pp. 701–719.

See Also

pdftexp, cdftexp, partexp

Examples

```
  lmr <- vec2lmom(c(40,0.38), lscale=FALSE)
  quatexp(0.5,partexp(lmr))

## Not run:
  F <- nonexceeds()
  L1 <- 50; T2 <- seq(0.51,0.005,by=-.001)
  PAR <- partexp(vec2lmom(c(L1,1/3), lscale=FALSE))
  plot(F, quatexp(F, PAR),
       type="l", lwd=2, col=2,
       xlab="NONEXCEEDANCE PROBABILITY",
       ylab="SIGHTING TIMES",
       ylim=c(0,300)) # uniform distribution

  for(t2 in T2) {
    PAR <- partexp(vec2lmom(c(L1,t2), lscale=FALSE))
    if(is.null(PAR)) next
    if(PAR$is.uni) {
      # For the T2 near 1/3 a kick over to uniform solution is
      # needed.  For the -0.001 steps shown above no uniform
      # distribution solutions will be "used" and no output.
      print(PAR$para) # by this print() will be seen.
    }
    lines(F, quatexp(F,PAR), col=rgb(0,0,0,.1))
  }
  # Because T2 started at > 1/2, ten warnings of LCV > 1/2
  # will result during execution of the for() loop.

## End(Not run)
```

---

quawak — *Quantile Function of the Wakeby Distribution*

Description

This function computes the quantiles of the Wakeby distribution given parameters (ξ, α, β, γ, and δ) of the distribution computed by parwak. The quantile function of the distribution is

$$x(F) = \xi + \frac{\alpha}{\beta}(1 - (1-F)^{\beta}) - \frac{\gamma}{\delta}(1 - (1-F))^{-\delta},$$

where $x(F)$ is the quantile for nonexceedance probability F, ξ is a location parameter, α and β are scale parameters, and γ, and δ are shape parameters. The five returned parameters from parwak in order are ξ, α, β, γ, and δ.

Usage

```
quawak(f, wakpara, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`wakpara`	The parameters from `parwak` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

Hosking, J.R.M., 1996, FORTRAN routines for use with the method of L-moments: Version 3, IBM Research Report RC20525, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfwak`, `parwak`

Examples

```
lmr <- lmom.ub(c(123,34,4,654,37,78))
quawak(0.5,parwak(lmr))
```

`quawei` *Quantile Function of the Weibull Distribution*

Description

This function computes the quantiles of the Weibull distribution given parameters (ζ, β, and δ) of the distribution computed by `parwei`. The quantile function of the distribution is

$$x(F) = \beta[-\log(1-F)]^{1/\delta} - \zeta,$$

where $x(F)$ is the quantile for nonexceedance probability F, ζ is a location parameter, β is a scale parameter, and δ is a shape parameter.

The Weibull distribution is a reverse Generalized Extreme Value distribution. As result, the Generalized Extreme Value algorithms are used for implementation of the Weibull in this package. The relation between the Generalized Extreme Value distribution parameters (ξ, α, and κ) is

$$\kappa = 1/\delta,$$

$$\alpha = \beta/\delta, \text{ and}$$

$$\xi = \zeta - \beta.$$

These relations are taken from Hosking and Wallis (1997).

In R the quantile function of the Weibull distribution is `qweibull`. Given a Weibull parameter object `p`, the R syntax is `qweibull(f, p$para[3], scale=p$para[2]) - p$para[1]`. For the current implementation for this package, the reversed Generalized Extreme Value distribution is used `-quagev((1-f),para)`.

Usage

```
quawei(f, para, paracheck=TRUE)
```

Arguments

`f`	Nonexceedance probability ($0 \le F \le 1$).
`para`	The parameters from `parwei` or similar.
`paracheck`	A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

Value

Quantile value for nonexceedance probability F.

Author(s)

W.H. Asquith

References

Hosking, J.R.M. and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge University Press.

See Also

`cdfwei`, `parwei`

Examples

```
# Evaluate Weibull deployed here and within R (qweibull)
lmr <- lmom.ub(c(123,34,4,654,37,78))
WEI <- parwei(lmr)
Q1  <- quawei(0.5,WEI)
Q2  <- qweibull(0.5,shape=WEI$para[3],scale=WEI$para[2])-WEI$para[1]
if(Q1 == Q2) EQUAL <- TRUE

# The Weibull is a reversed generalized extreme value
Q <- sort(rlmomco(34,WEI)) # generate Weibull sample
lm1 <- lmoms(Q)    # regular L-moments
lm2 <- lmoms(-Q)   # L-moment of negated (reversed) data
WEI <- parwei(lm1) # parameters of Weibull
GEV <- pargev(lm2) # parameters of GEV
F <- nonexceeds()  # Get a vector of nonexceedance probs
plot(pp(Q),Q)
lines(F,quawei(F,WEI))
lines(F,-quagev(1-F,GEV),col=2) # line over laps previous
```

---

rlmomco   *Random Deviates of a Distribution*

Description

This function generates random deviates for the specified distribution in the parameter object argument. See documentation about the parameter object is seen in `lmom2par` or `vec2par`. The prepended `r` in the function name is to parallel the built-in distribution syntax of R but of course reflects the **lmomco** name in the function. An assumption is made that the user knows that they are feeding appropirate (valid) parameters for the distribution. This is seen by the

`paracheck = FALSE`

argument passed to the `par2qua` function.

Usage

```
rlmomco(n,para)
```

Arguments

`n`	Number of samples to generate
`para`	The parameters from `lmom2par` or similar.

Value

Vector of quantile values.

Author(s)

W.H. Asquith

See Also

dlmomco, plmomco, qlmomco, lmom2par, vec2par

Examples

```
lmr      <- lmom.ub(rnorm(20)) # generate 20 standard normal deviates
para     <- parnor(lmr) # estimate parameters of the normal
simulate <- rlmomco(20,para) # simulate 20 samples using lmomco package

lmr  <- vec2lmom(c(1000,500,.3)) # first three lmoments are known
para <- lmom2par(lmr,type="gev") # est. parameters of GEV distribution
Q    <- rlmomco(45,para) # simulate 45 samples
PP   <- pp(Q)            # compute the plotting positions
plot(PP,sort(Q))         # plot the data up
```

---

sen.mean *Sen's Weighted Mean Statistic*

Description

The Sen weighted mean statistic $\mathcal{S}_{n,k}$ is a robust estimator of the mean of a distribution.

$$\mathcal{S}_{n,k} = \binom{n}{2k+1}^{-1} \sum_{i=1}^{n} \binom{i-1}{k} \binom{n-i}{k} X_{i:n}$$

where $X_{i:n}$ are the order statistics and k is a weighting or trimming parameter.

Note that $\mathcal{S}_{n,0} = \mu = \overline{X}_n$ or the arithmetic mean and $\mathcal{S}_{n,k}$ is the sample median if either n is even and $k = (n/2) - 1$ or n is odd and $k = (n-1)/2$.

Usage

```
sen.mean(x,k=0)
```

Arguments

x	A vector of data values that will be reduced to non-missing values.
k	A weighting or trimming parameter $0 < k < (n-1)/2$.

Value

An R list is returned.

sen	The sen mean $\mathcal{S}_{n,k}$
source	An attribute identifying the computational source of Sen's Weighted Mean: "sen.mean".

Author(s)

W.H. Asquith

References

Jurečková, J., and Picek, J., 2006, Robust statistical methods with R: Boca Raton, Fla., Chapman and Hall/CRC, ISBN 1–58488–454–1, 197~p.

Sen, P.K., 1964, On some properties of the rank-weighted means: Journal Indian Society of Agricultural Statistics: vol.~16, pp.~51–61.

See Also

TLmoms, gini.mean.diff

Examples

```
fake.dat <- c(123,34,4,654,37,78)
sen.mean(fake.dat); mean(fake.dat) # These should be the same values

sen.mean(fake.dat, k=(length(fake.dat)/2) - 1); median(fake.dat)
# Again, same values

# Finally, the sen.mean() is like a symmetrically trimmed TL-moment
# Let us demonstrate by computed a two sample trimming for each side
# for a Normal distribution having a mean of 100.
fake.dat <- rnorm(20, mean=100)
lmr <- TLmoms(fake.dat, trim=2)
sen <- sen.mean(fake.dat, k=2)

print(abs(lmr$lambdas[1] - sen$sen)) # zero is returned
```

T2prob *Convert a Vector of T-year Return Periods to Annual Nonexceedance Probabilities*

Description

This function converts a vector of T-year return periods to annual nonexceedance probabilities F.

$$F = 1 - 1/T$$

Usage

```
T2prob(T)
```

Arguments

T	A vector of T-year return periods.

Value

A vector of annual nonexceedance probabilities.

Author(s)

W.H. Asquith

See Also

prob2T, nonexceeds

Examples

```
T <- c(1, 2, 5, 10, 25, 50, 100, 250, 500)
F <- T2prob(T)
```

---

theoLmoms *The Theoretical L-moments and L-moment Ratios using Integration of the Quantile Function*

Description

Compute the theoretical L-moments for a vector. A theoretical L-moment in integral form is

$$\lambda_r = \frac{1}{r}\sum_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\frac{r!\,I_r}{(r-k-1)!k!}\text{, in which}$$

$$I_r = \int_0^1 X(F)\times F^{r-k-1}(1-F)^k\,\mathrm{d}F,$$

where $X(F)$ is the quantile function of the random variable X for nonexceedance probability F, and r represents the order of the L-moments. This function actually dispatches to theoTLmoms with `trim=0` argument.

Usage

```
theoLmoms(para,nmom=5,verbose=FALSE,minF=0,maxF=1)
```

Arguments

`para`	A distribution parameter object of this package `vec2par`.
`nmom`	The number of moments to compute. Default is 5.
`verbose`	Toggle verbose output. Because the R function `integrate` is used to perform the numerical integration, it might be useful to see selected messages regarding the numerical integration.
`minF`	The end point of nonexceedance probability in which to perform the integration. Try setting to non-zero (but small) if you have a divergent integral.
`maxF`	The end point of nonexceedance probability in which to perform the integration. Try setting to non-unity (but close) if you have a divergent integral.

Value

An R `list` is returned.

`lambdas`	Vector of the TL-moments. First element is λ_1, second element is λ_2, and so on.
`ratios`	Vector of the L-moment ratios. Second element is τ_2, third element is τ_3 and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal zero (the ordinary L-moments).
`source`	An attribute identifying the computational source of the L-moments: "theoTL-moms".

Note

The actual function used is `theoLmoms(para,nmom=nmom,trim=0,verbose=verbose`.

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105–124.

See Also

`theoTLmoms`, `par2qua`, `TLmoms`, `lmom.ub`

Examples

```
para <- vec2par(c(0,1),type='nor') # standard normal
TL00 <- theoLmoms(para) # compute ordinary L-moments
```

theoLmoms.max.ostat — *Compute the Theoretical L-moments of a Distribution Distribution based on System of Maximum Order Statistic Expectations*

Description

This function computes

Usage

```
theoLmoms.max.ostat(para=NULL, cdf=NULL, pdf=NULL, nmom=4, ...)
```

Arguments

`para`	A distribution parameter list from a function such as `vec2par` or `lmom2par`.
`cdf`	CDF of the distribution for the parameters.
`pdf`	PDF of the distribution for the parameters.
`nmom`	The number of L-moments to compute.
`...`	Optional, but likely, arguments to pass to `expect.max.ostat`. Such arguments will likely tailor the integrations limits specific for the distribution in question.

Value

An R `list` is returned.

`lambdas`	Vector of the L-moments. First element is λ_1, second element is λ_2, and so on.
`ratios`	Vector of the L-moment ratios. Second element is τ, third element is τ_3 and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` until trimming support is made.
`leftrim`	Level of left-tail trimming used in the computation, which will equal `NULL` until trimming support is made.
`rightrim`	Level of right-tail trimming used in the computation, which will equal `NULL` until trimming support is made.
`source`	An attribute identifying the computational source of the L-moments: "theoLmoms.max.ostat".

Author(s)

W.H. Asquith

See Also

`theoLmoms`, `expect.max.ostat`

Examples

```
para <- vec2par(c(40,20), type='nor')
theoLmoms.max.ostat(para=para, cdf=cdfnor, pdf=pdfnor)

# The Rice distribution is complex and tailoring of the integration
# limits is needed to effectively trapped errors, the limits for the
# Normal distribution above are infinite so no granular control is needed.
para <- vec2par(c(30,10), type="rice")
theoLmoms.max.ostat(para=para, cdf=cdfrice, pdf=pdfrice,
                    lower=0, upper=.Machine$double.max)

## Not run:
# The Normal distribution is used on the fly for the Rice for
# high signal to noise ratios (SNR=nu/alpha > some threshold). This
# application here will error out.
nu <- 30; alpha <- 0.5
para <- vec2par(c(nu,alpha), type="rice")
theoLmoms.max.ostat(para=para, cdf=cdfrice, pdf=pdfrice,
                    lower=0, upper=.Machine$double.max)

## End(Not run)
```

`theopwms` *The Theoretical Probability-Weighted Moments using Integration of the Quantile Function*

Description

Compute the theoretical probability-weighted moments (PWMs) for a distribution. A theoretical PWM in integral form is

$$\beta_r = \int_0^1 X(F)\, F^r \,\mathrm{d}F,$$

where $X(F)$ is the quantile function of the random variable X for nonexceedance probability F and r represents the order of the PWM. This function loops across the above equation for each `nmom` set in the argument list. The function $X(F)$ is computed through the `par2qua` function. The distribution type is determined using the `type` attribute of the `para` argument—the parameter object.

Usage

```
theopwms(para,nmom=5,verbose=FALSE)
```

Arguments

`para`	A distribution parameter object of this package `vec2par`.
`nmom`	The number of moments to compute. Default is 5.
`verbose`	Toggle verbose output. Because the R function `integrate` is used to perform the numerical integration, it might be useful to see selected messages regarding the numerical integration.

Value

An R `list` is returned.

`betas`	The PWMs. Note that convention is the have a β_0, but this is placed in the first index `i=1` of the `betas` vector.
`source`	An attribute identifying the computational source of the probability-weighted moments: "theopwms".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105-124.

See Also

`theoLmoms`, `par2qua`, `pwm`

Examples

```
para     <- vec2par(c(0,1),type='nor') # standard normal
the.pwms <- theopwms(para) # compute PWMs
str(the.pwms)
```

`theoTLmoms` *The Theoretical Trimmed L-moments and TL-moment Ratios using Integration of the Quantile Function*

Description

Compute the theoretical trimmed L-moments (TL-moments) for a vector. The level of symmetrical or asymmetrical trimming is specified. A theoretical TL-moment in integral form is

$$\lambda_r^{(t_1,t_2)} = \underbrace{\frac{1}{r}}_{\substack{\text{average}\\ \text{of terms}}} \sum_{k=0}^{r-1} \overbrace{(-1)^k}^{\text{differences}} \underbrace{\binom{r-1}{k}}_{\text{combinations}} \frac{\overbrace{(r+t_1+t_2)!}^{\text{sample size}}\; I_r^{(t_1,t_2)}}{\underbrace{(r+t_1-k-1)!}_{\text{left tail}}\underbrace{(t_2+k)!}_{\text{right tail}}}, \text{ in which}$$

$$I_r^{(t_1,t_2)} = \int_0^1 \underbrace{X(F)}_{\substack{\text{quantile}\\ \text{function}}} \times \overbrace{F^{r+t_1-k-1}}^{\text{left tail}} \overbrace{(1-F)^{t_2+k}}^{\text{right tail}} \,\mathrm{d}F,$$

where $X(F)$ is the quantile function of the random variable X for nonexceedance probability F, t_1 represents the trimming level of the t_1-smallest, t_2 represents the trimming level of the t_2-largest values, r represents the order of the L-moments. This function loops across the above equation for each nmom set in the argument list. The function $X(F)$ is computed through the par2qua function. The distribution type is determined using the type attribute of the para argument—the parameter object.

As of version 1.5.2 of **lmomco**, there exists enhanced error trapping on integration failures in theoTLmoms. The function now abandons operations should any of the integrations for the rth L-moment fail for reasons such as divergent integral or round off problems. The function returns NAs for all L-moments in lambdas and ratios.

Usage

```
theoTLmoms(para,nmom=5,trim=NULL,leftrim=NULL,
           rightrim=NULL,verbose=FALSE,minF=0,maxF=1)
```

Arguments

para	A distribution parameter object of this package vec2par.
nmom	The number of moments to compute. Default is 5.
trim	Level of symmetrical trimming to use in the computations. Although NULL in the argument list, the default is 0—the usual L-moment is returned.
leftrim	Level of trimming of the left-tail of the sample.
rightrim	Level of trimming of the right-tail of the sample.
verbose	Toggle verbose output. Because the R function integrate is used to perform the numerical integration, it might be useful to see selected messages regarding the numerical integration.
minF	The end point of nonexceedance probability in which to perform the integration. Try setting to non-zero (but small) if you have a divergent integral.
maxF	The end point of nonexceedance probability in which to perform the integration. Try setting to non-unity (but close) if you have a divergent integral.

Value

An R list is returned.

lambdas	Vector of the TL-moments. First element is $\lambda_1^{(t_1,t_2)}$, second element is $\lambda_2^{(t_1,t_2)}$, and so on.
ratios	Vector of the L-moment ratios. Second element is $\tau^{(t_1,t_2)}$, third element is $\tau_3^{(t_1,t_2)}$ and so on.
trim	Level of symmetrical trimming used in the computation, which will equal NULL if asymmetrical trimming was used.
leftrim	Level of left-tail trimming used in the computation.
rightrim	Level of right-tail trimming used in the computation.
source	An attribute identifying the computational source of the L-moments: "theoTL-moms".

Author(s)

W.H. Asquith

References

Hosking, J.R.M., 1990, L-moments–Analysis and estimation of distributions using linear combinations of order statistics: Journal of the Royal Statistical Society, Series B, vol. 52, p. 105-124.

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, vol. 43, pp. 299–314.

See Also

theoLmoms, par2qua, TLmoms, lmom.ub

Examples

```
para <- vec2par(c(0,1),type='nor') # standard normal
TL00 <- theoTLmoms(para) # compute ordinary L-moments
TL30 <- theoTLmoms(para,leftrim=3,rightrim=0) # trim 3 smallest samples

# Lets look at the difference from simulation to theoretrical using
# L-kurtosis and asymmetrical trimming for generalized Lambda dist.
P <- vec2par(c(10000,10000,6,.4),type='gld')
Lkurt <- TLmoms(quagld(runif(100),P),rightrim=3,leftrim=0)$ratios[4]
theoLkurt <- theoTLmoms(P,rightrim=3,leftrim=0)$ratios[4]
Lkurt - theoLkurt # as the number for runif goes up, this
                  # difference goes to zero

# Example using the Generalized Pareto Distribution
# to verify computations from theoretical and sample stand point.
n      <- 100 # really a much larger sample should be used---for speed
P      <- vec2par(c(12,34,4),type='gpa')
theoTL <- theoTLmoms(P,rightrim=2,leftrim=4)
samTL  <- TLmoms(quagpa(runif(n),P),rightrim=2,leftrim=4)
del    <- samTL$ratios[3] - theoTL$ratios[3] # if n is large difference
                                             # is small
str(del)
```

TLmom *A Sample Trimmed L-moment*

Description

A sample trimmed L-moment (TL-moment) is computed for a vector. The $r \geq 1$ order of the L-moment is specified as well as the level of symmetrical trimming. A trimmed TL-moment is

$$\hat{\lambda}^{(t)}_r = \frac{1}{r}\sum_{i=t+1}^{n-t}\left[\frac{\sum\limits_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\binom{i-1}{r+t-1-k}\binom{n-i}{t+k}}{\binom{n}{r+2t}}\right]x_{i:n},$$

where t represents the trimming level of the t-largest or t-smallest values, r represents the order of the L-moments, n represents the sample size, and $x_{i:n}$ represents the ith sample order statistic ($x_{1:n} \leq x_{2:n} \leq \ldots \leq x_{n:n}$).

Usage

```
TLmom(x,order,trim=NULL,leftrim=NULL,rightrim=NULL,sortdata)
```

Arguments

`x`	A vector of data values.
`order`	L-moment order to use in the computations. Default is 1 (the mean).
`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment is returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`sortdata`	A logical switch on whether the data should be sorted. The default is TRUE.

Value

An R `list` is returned.

`lambda`	The TL-moment of order=`order` and trimming=`trim`.
`order`	L-moment order computed. Default is 1 (the mean).
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` if asymmetrical trimming was used.
`leftrim`	Level of left-tail trimming used in the computation.
`rightrim`	Level of right-tail trimming used in the computation.

Note

The presence of the `sortdata` switch can be dangerous. L-moment computation requires that the data be sorted into the "order statistics". Thus the default behavior of `sortdata=TRUE` is required when the function is called on its own. In practice, this function would not be used as multiple trimmed L-moments would be needed. Multiple trimmed L-moments are best computed by `TLmoms`, which calls `TLmom` multiple times. The function `TLmoms` takes over the sort operation on the data and passes `sortdata=FALSE` to `TLmom` for efficiency. (The point of this discussion is that one does not want to waste CPU time sorting the data more than once.)

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational statistics and data analysis, vol. 43, pp. 299-314.

See Also

TLmoms

Examples

```
X1 <- rcauchy(30)
TL <- TLmom(X1,order=2,trim=1)
```

TLmoms *The Sample Trimmed L-moments and L-moment Ratios*

Description

Compute the sample trimmed L-moments (TL-moments) for a vector. The level of symmetrical trimming is specified. A TL-moment is

$$\hat{\lambda}^{(t_1,t_2)}_r = \frac{1}{r}\sum_{i=t_1+1}^{n-t_2}\left[\frac{\sum\limits_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\binom{i-1}{r+t_1-1-k}\binom{n-i}{t_2+k}}{\binom{n}{r+t_1+t_2}}\right]x_{i:n},$$

where t_a represents the trimming level of the t_2-largest or t_1-smallest values, r represents the order of the L-moments, n represents the sample size, and $x_{i:n}$ represents the ith sample order statistic ($x_{1:n} \le x_{2:n} \le \ldots \le x_{n:n}$). This function loops across the above equation for each `nmom` set in the argument list.

Usage

```
TLmoms(x,nmom,trim=NULL,leftrim=NULL,rightrim=NULL)
```

Arguments

`x`	A vector of data values.
`nmom`	The number of moments to compute. Default is 5.
`trim`	Level of symmetrical trimming to use in the computations, which will equal `NULL` if asymmetrical trimming was used. Although `NULL` in the argument list, the default is 0—the usual L-moment is returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.

Value

An R `list` is returned.

`lambdas`	Vector of the TL-moments. First element is $\hat{\lambda}_1^{(t_1,t_2)}$, second element is $\hat{\lambda}_2^{(t_1,t_2)}$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\hat{\tau}^{(t_1,t_2)}$, third element is $\hat{\tau}_3^{(t_1,t_2)}$ and so on.
`trim`	Level of symmetrical trimming used in the computation, which will equal `NULL` if asymmetrical trimming was used.
`leftrim`	Level of left-tail trimming used in the computation.
`rightrim`	Level of right-tail trimming used in the computation.
`source`	An attribute identifying the computational source of the L-moments: "TLmoms".

Note

It is important to note that the "L-moment object" returned by `TLmoms` is different in structure to that returned by `lmom.ub`, `pwm2lmom`, and similar as the TL-moments should not be confused with the ordinary L-moments. Implementation in the package might change.

Author(s)

W.H. Asquith

References

Elamir, E.A.H., and Seheult, A.H., 2003, Trimmed L-moments: Computational statistics and data analysis, vol. 43, pp. 299-314.

See Also

`TLmom`, `lmoms`, and `lmorph`

Examples

```
X1 <- rcauchy(30)
TL <- TLmoms(X1,nmom=6,trim=1)
```

tlmrcau — *Compute Select TL-moment ratios of the Cauchy Distribution*

Description

This function computes select TL-moment ratios of the Cauchy distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrcau(trim=NULL, leftrim=NULL, rightrim=NULL, xi=0, alpha=1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quacau`, `theoTLmoms`

Examples

```
tlmrcau(trim=2)
tlmrcau(trim=2, xi=2)
```

---

tlmrexp *Compute Select TL-moment ratios of the Exponential Distribution*

Description

This function computes select TL-moment ratios of the Exponential distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrexp(trim=NULL, leftrim=NULL, rightrim=NULL, xi=0, alpha=1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

quaexp, theoTLmoms

Examples

```
tlmrexp(trim=2)
tlmrexp(trim=2, xi=2)
```

---

tlmrgev *Compute Select TL-moment ratios of the Generalized Extreme Value Distribution*

Description

This function computes select TL-moment ratios of the Generalized Extreme Value distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter κ parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the theoTLmoms function.

Usage

```
tlmrgev(trim=NULL, leftrim=NULL, rightrim=NULL,
        xi=0, alpha=1, kbeg=-.99, kend=10, by=.1)
```

Arguments

trim	Level of symmetrical trimming to use in the computations. Although NULL in the argument list, the default is 0—the usual L-moment ratios are returned.
leftrim	Level of trimming of the left-tail of the sample.
rightrim	Level of trimming of the right-tail of the sample.
xi	Location parameter of the distribution.
alpha	Scale parameter of the distribution.
kbeg	The beginning κ value of the distribution.
kend	The ending κ value of the distribution.
by	The increment for the seq() between kbeg and kend.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quagev`, `theoTLmoms`

Examples

```
tlmrgev(leftrim=12, rightrim=1, xi=0, alpha=2)
tlmrgev(leftrim=12, rightrim=1, xi=100, alpha=20)

## Not run:
  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the GEV distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, noglo=TRUE, nogpa=TRUE, nope3=TRUE,
             nogno=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
             nogum=TRUE, noray=TRUE, nouni=TRUE)

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right. Notice the
  # expansion of the kappa parameter space from > -1 to
  # something near -5.
  J <- tlmrgev(kbeg=-4.99, leftrim=1, rightrim=4)
  lines(J$tau3, J$tau4, lwd=2, col=3) # BLUE CURVE

  # Compute the TL-moment ratios for trimming of four
  # values on the left and one on the right.
  J <- tlmrgev(kbeg=-1.99, leftrim=4, rightrim=1)
  lines(J$tau3, J$tau4, lwd=2, col=4) # GREEN CURVE

  # The kbeg and kend can be manually changed to see how
  # the resultant curve expands or contracts on the
  # extent of the L-moment ratio diagram.

## End(Not run)
## Not run:
```

```
  # Following up, let us plot the two quantile functions
  LM  <- vec2par(c(0,1,-0.99), type='gev', paracheck=FALSE)
  TLM <- vec2par(c(0,1,-4.99), type='gev', paracheck=FALSE)
  F <- nonexceeds()
  plot(qnorm(F),  quagev(F, LM), type="l")
  lines(qnorm(F), quagev(F, TLM, paracheck=FALSE), col=2)
  # Notice how the TLM parameterization runs off towards
  # infinity much much earlier than the conventional
  # near limits of the GEV.

## End(Not run)
```

`tlmrglo` *Compute Select TL-moment ratios of the Generalized Logistic Distribution*

Description

This function computes select TL-moment ratios of the Generalized Logistic distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter κ parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrglo(trim=NULL, leftrim=NULL, rightrim=NULL,
        xi=0, alpha=1, kbeg=-.99, kend=0.99, by=.1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.

kbeg	The beginning κ value of the distribution.
kend	The ending κ value of the distribution.
by	The increment for the `seq()` between `kbeg` and `kend`.

Value

An R `list` is returned.

tau2	A vector of the $\tau_2^{(t_1,t_2)}$ values.
tau3	A vector of the $\tau_3^{(t_1,t_2)}$ values.
tau4	A vector of the $\tau_4^{(t_1,t_2)}$ values.
tau5	A vector of the $\tau_5^{(t_1,t_2)}$ values.
tau6	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

quaglo, theoTLmoms

Examples

```
tlmrglo(leftrim=1, rightrim=3, xi=0, alpha=4)
tlmrglo(leftrim=1, rightrim=3, xi=32, alpha=83)

## Not run:
  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the GLO distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, nogev=TRUE, nogpa=TRUE, nope3=TRUE,
             nogno=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
             nogum=TRUE, noray=TRUE, nouni=TRUE)

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right. Notice the
  # expansion of the kappa parameter space from
  # -1 < k < -1 to something larger based on manual
  # adjustments until blue curve encompassed the plot.
  J <- tlmrglo(kbeg=-2.5, kend=1.9, leftrim=1, rightrim=4)
  lines(J$tau3, J$tau4, lwd=2, col=2) # RED CURVE

  # Compute the TL-moment ratios for trimming of four
  # values on the left and one on the right.
  J <- tlmrglo(kbeg=-1.65, kend=3, leftrim=4, rightrim=1)
  lines(J$tau3, J$tau4, lwd=2, col=4) # BLUE CURVE
```

```
  # The kbeg and kend can be manually changed to see how
  # the resultant curve expands or contracts on the
  # extent of the L-moment ratio diagram.

## End(Not run)
## Not run:

  # Following up, let us plot the two quantile functions
  LM  <- vec2par(c(0,1,0.99), type='glo', paracheck=FALSE)
  TLM <- vec2par(c(0,1,3.00), type='glo', paracheck=FALSE)
  F <- nonexceeds()
  plot(qnorm(F),  quaglo(F, LM), type="l")
  lines(qnorm(F), quaglo(F, TLM, paracheck=FALSE), col=2)
  # Notice how the TLM parameterization runs off towards
  # infinity much much earlier than the conventional
  # near limits of the GLO.

## End(Not run)
```

---

tlmrgno *Compute Select TL-moment ratios of the Generalized Normal Distribution*

Description

This function computes select TL-moment ratios of the Generalized Normal distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter κ parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the theoTLmoms function.

Usage

```
tlmrgno(trim=NULL, leftrim=NULL, rightrim=NULL,
        xi=0, alpha=1, kbeg=-3, kend=3, by=.1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.
`kbeg`	The beginning κ value of the distribution.
`kend`	The ending κ value of the distribution.
`by`	The increment for the `seq()` between `kbeg` and `kend`.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quagno`, `theoTLmoms`

Examples

```
tlmrgno(leftrim=3, rightrim=2, xi=0, alpha=2)
tlmrgno(leftrim=3, rightrim=2, xi=120, alpha=55)

## Not run:
  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the GNO distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, nogev=TRUE, nogpa=TRUE, nope3=TRUE,
             noglo=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
             nogum=TRUE, noray=TRUE, nouni=TRUE)

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right.
  J <- tlmrgno(kbeg=-3.5, kend=3.9, leftrim=1, rightrim=4)
```

```
  lines(J$tau3, J$tau4, lwd=2, col=2) # RED CURVE

  # Compute the TL-moment ratios for trimming of four
  # values on the left and one on the right.
  J <- tlmrgno(kbeg=-4, kend=4, leftrim=4, rightrim=1)
  lines(J$tau3, J$tau4, lwd=2, col=4) # BLUE CURVE

  # The kbeg and kend can be manually changed to see how
  # the resultant curve expands or contracts on the
  # extent of the L-moment ratio diagram.

## End(Not run)
## Not run:

  # Following up, let us plot the two quantile functions
  LM  <- vec2par(c(0,1,0.99), type='gno', paracheck=FALSE)
  TLM <- vec2par(c(0,1,3.00), type='gno', paracheck=FALSE)
  F <- nonexceeds()
  plot(qnorm(F),  quagno(F, LM), type="l")
  lines(qnorm(F), quagno(F, TLM, paracheck=FALSE), col=2)
  # Notice how the TLM parameterization runs off towards
  # infinity much much earlier than the conventional
  # near limits of the GNO.

## End(Not run)
```

---

tlmrgpa *Compute Select TL-moment ratios of the Generalized Pareto*

Description

This function computes select TL-moment ratios of the Generalized Pareto distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter κ parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the theoTLmoms function.

Usage

```
tlmrgpa(trim=NULL, leftrim=NULL, rightrim=NULL,
        xi=0, alpha=1, kbeg=-.99, kend=10, by=.1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.
`kbeg`	The beginning κ value of the distribution.
`kend`	The ending κ value of the distribution.
`by`	The increment for the `seq()` between `kbeg` and `kend`.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quagpa`, `theoTLmoms`

Examples

```
tlmrgpa(leftrim=7, rightrim=2, xi=0, alpha=31)
tlmrgpa(leftrim=7, rightrim=2, xi=143, alpha=98)

## Not run:
  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the GPA distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, nogev=TRUE, noglo=TRUE, nope3=TRUE,
             nogno=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
             nogum=TRUE, noray=TRUE, nouni=TRUE)

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right. Notice the
  # expansion of the kappa parameter space from k > -1.
```

```
  J <- tlmrgpa(kbeg=-3.2, kend=50, by=.05, leftrim=1, rightrim=4)
  lines(J$tau3, J$tau4, lwd=2, col=2) # RED CURVE
  # Notice the gap in the curve near tau3 = 0.1

  # Compute the TL-moment ratios for trimming of four
  # values on the left and one on the right.
  J <- tlmrgpa(kbeg=-1.6, kend=8, leftrim=4, rightrim=1)
  lines(J$tau3, J$tau4, lwd=2, col=3) # GREEN CURVE

  # The kbeg and kend can be manually changed to see how
  # the resultant curve expands or contracts on the
  # extent of the L-moment ratio diagram.

## End(Not run)
## Not run:

  # Following up, let us plot the two quantile functions
  LM  <- vec2par(c(0,1,0.99), type='gpa', paracheck=FALSE)
  TLM <- vec2par(c(0,1,3.00), type='gpa', paracheck=FALSE)
  F <- nonexceeds()
  plot(qnorm(F),  quagpa(F, LM), type="l")
  lines(qnorm(F), quagpa(F, TLM, paracheck=FALSE), col=2)
  # Notice how the TLM parameterization runs off towards
  # infinity much much earlier than the conventional
  # near limits of the GPA.

## End(Not run)
```

---

tlmrgum *Compute Select TL-moment ratios of the Gumbel Distribution*

Description

This function computes select TL-moment ratios of the Gumbel distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrgum(trim=NULL, leftrim=NULL, rightrim=NULL, xi=0, alpha=1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.

rightrim	Level of trimming of the right-tail of the sample.
xi	Location parameter of the distribution.
alpha	Scale parameter of the distribution.

Value

An R `list` is returned.

tau2	A vector of the $\tau_2^{(t_1,t_2)}$ values.
tau3	A vector of the $\tau_3^{(t_1,t_2)}$ values.
tau4	A vector of the $\tau_4^{(t_1,t_2)}$ values.
tau5	A vector of the $\tau_5^{(t_1,t_2)}$ values.
tau6	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

quagum, theoTLmoms

Examples

```
tlmrgum(trim=2)
tlmrgum(trim=2, xi=2)
```

---

tlmrln3 — *Compute Select TL-moment ratios of the 3-Parameter Log-Normal Distribution*

Description

This function computes select TL-moment ratios of the Log-Normal3 distribution for defaults of $\zeta = 0$ and $\mu_{\mathrm{log}} = 0$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, , $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ζ and μ_{log}.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter σ_{log} parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the theoTLmoms function.

Usage

```
tlmrln3(trim=NULL, leftrim=NULL, rightrim=NULL,
        zeta=0, mulog=0, sbeg=0.01, send=3.5, by=.1)
```

Arguments

trim	Level of symmetrical trimming to use in the computations. Although NULL in the argument list, the default is 0—the usual L-moment ratios are returned.
leftrim	Level of trimming of the left-tail of the sample.
rightrim	Level of trimming of the right-tail of the sample.
zeta	Location parameter of the distribution.
mulog	Mean of the logarithms of the distribution.
sbeg	The beginning $\sigma_{\log}$ value of the distribution.
send	The ending $\sigma_{\log}$ value of the distribution.
by	The increment for the seq() between sbeg and send.

Value

An R list is returned.

tau2	A vector of the $\tau_2^{(t_1,t_2)}$ values.
tau3	A vector of the $\tau_3^{(t_1,t_2)}$ values.
tau4	A vector of the $\tau_4^{(t_1,t_2)}$ values.
tau5	A vector of the $\tau_5^{(t_1,t_2)}$ values.
tau6	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

qualn3, theoTLmoms

Examples

```
## Not run:
  # Recalling that generalized Normal and log-Normal3 are
  # the same with the GNO being the more general.

  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the GNO distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, noglo=TRUE, nogpa=TRUE, nope3=TRUE,
             nogev=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
```

```
            nogum=TRUE, noray=TRUE, nouni=TRUE)

  LN3 <- tlmrln3(sbeg=.001, mulog=-1)
  lines(LN3$tau3, LN3$tau4) # See how it overplots the GNO
  # for right skewness. So only part of the GNO is covered.

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right.
  J <- tlmrgno(kbeg=-3.5, kend=3.9, leftrim=1, rightrim=4)
  lines(J$tau3, J$tau4, lwd=2, col=2) # RED CURVE

  LN3 <- tlmrln3(, leftrim=1, rightrim=4, sbeg=.001)
  lines(LN3$tau3, LN3$tau4) # See how it again over plots
  # only part of the GNO

## End(Not run)
```

---

tlmrnor *Compute Select TL-moment ratios of the Normal Distribution*

Description

This function computes select TL-moment ratios of the Normal distribution for defaults of $\mu = 0$ and $\sigma = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for μ and σ.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrnor(trim=NULL, leftrim=NULL, rightrim=NULL, mu=0, sigma=1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`mu`	Location parameter (mean) of the distribution.
`sigma`	Scale parameter (standard deviation) of the distribution.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quanor`, `theoTLmoms`

Examples

```
tlmrnor(leftrim=2, rightrim=1)
tlmrnor(leftrim=2, rightrim=1, mu=100, sigma=1000)
```

---

tlmrpe3 *Compute Select TL-moment ratios of the Pearson Type III*

Description

This function computes select TL-moment ratios of the Pearson Type III distribution for defaults of $\xi = 0$ and $\beta = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

If the message

```
Error in integrate(XofF, 0, 1) : the integral is probably divergent
```

occurs then careful adjustment of the shape parameter β parameter range is very likely required. Remember that TL-moments with nonzero trimming permit computation of TL-moments into parameter ranges beyond those recognized for the usual L-moments.

The function uses numerical integration of the quantile function of the distribution through the `theoTLmoms` function.

Usage

```
tlmrpe3(trim=NULL, leftrim=NULL, rightrim=NULL,
        xi=0, beta=1, abeg=-.99, aend=0.99, by=.1)
```

Arguments

trim	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
leftrim	Level of trimming of the left-tail of the sample.
rightrim	Level of trimming of the right-tail of the sample.
xi	Location parameter of the distribution.
beta	Scale parameter of the distribution.
abeg	The beginning α value of the distribution.
aend	The ending α value of the distribution.
by	The increment for the `seq()` between `abeg` and `aend`.

Value

An R `list` is returned.

tau2	A vector of the $\tau_2^{(t_1,t_2)}$ values.
tau3	A vector of the $\tau_3^{(t_1,t_2)}$ values.
tau4	A vector of the $\tau_4^{(t_1,t_2)}$ values.
tau5	A vector of the $\tau_5^{(t_1,t_2)}$ values.
tau6	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

quape3, theoTLmoms

Examples

```
tlmrpe3(leftrim=2, rightrim=4, xi=0, beta=2)
tlmrpe3(leftrim=2, rightrim=4, xi=100, beta=20)

## Not run:
  # Plot and L-moment ratio diagram of Tau3 and Tau4
  # with exclusive focus on the PE3 distribution.
  plotlmrdia(lmrdia(), autolegend=TRUE, xleg=-.1, yleg=.6,
             xlim=c(-.8, .7), ylim=c(-.1, .8),
             nolimits=TRUE, nogev=TRUE, nogpa=TRUE, noglo=TRUE,
             nogno=TRUE, nocau=TRUE, noexp=TRUE, nonor=TRUE,
             nogum=TRUE, noray=TRUE, nouni=TRUE)

  # Compute the TL-moment ratios for trimming of one
  # value on the left and four on the right. Notice the
```

```
  # expansion of the alpha parameter space from
  # -1 < a < -1 to something larger based on manual
  # adjustments until blue curve encompassed the plot.
  J <- tlmrpe3(abeg=-15, aend=6, leftrim=1, rightrim=4)
  lines(J$tau3, J$tau4, lwd=2, col=2) # RED CURVE

  # Compute the TL-moment ratios for trimming of four
  # values on the left and one on the right.
  J <- tlmrpe3(abeg=-6, aend=10, leftrim=4, rightrim=1)
  lines(J$tau3, J$tau4, lwd=2, col=4) # BLUE CURVE

  # The abeg and aend can be manually changed to see how
  # the resultant curve expands or contracts on the
  # extent of the L-moment ratio diagram.

## End(Not run)
## Not run:

  # Following up, let us plot the two quantile functions
  LM  <- vec2par(c(0,1,0.99), type='pe3', paracheck=FALSE)
  TLM <- vec2par(c(0,1,3.00), type='pe3', paracheck=FALSE)
  F <- nonexceeds()
  plot(qnorm(F),  quape3(F, LM), type="l")
  lines(qnorm(F), quape3(F, TLM, paracheck=FALSE), col=2)
  # Notice how the TLM parameterization runs off towards
  # infinity much much earlier than the conventional
  # near limits of the PE3.

## End(Not run)
```

tlmrray *Compute Select TL-moment ratios of the Rayleigh Distribution*

Description

This function computes select TL-moment ratios of the Rayleigh distribution for defaults of $\xi = 0$ and $\alpha = 1$. This function can be useful for plotting the trajectory of the distribution on TL-moment ratio diagrams of $\tau_2^{(t_1,t_2)}$, $\tau_3^{(t_1,t_2)}$, $\tau_4^{(t_1,t_2)}$, $\tau_5^{(t_1,t_2)}$, and $\tau_6^{(t_1,t_2)}$. In reality, $\tau_2^{(t_1,t_2)}$ is a dependent on the values for ξ and α.

The function uses numerical integration of the quantile function of the distribution through the theoTLmoms function.

Usage

```
tlmrray(trim=NULL, leftrim=NULL, rightrim=NULL, xi=0, alpha=1)
```

Arguments

`trim`	Level of symmetrical trimming to use in the computations. Although `NULL` in the argument list, the default is 0—the usual L-moment ratios are returned.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`xi`	Location parameter of the distribution.
`alpha`	Scale parameter of the distribution.

Value

An R `list` is returned.

`tau2`	A vector of the $\tau_2^{(t_1,t_2)}$ values.
`tau3`	A vector of the $\tau_3^{(t_1,t_2)}$ values.
`tau4`	A vector of the $\tau_4^{(t_1,t_2)}$ values.
`tau5`	A vector of the $\tau_5^{(t_1,t_2)}$ values.
`tau6`	A vector of the $\tau_6^{(t_1,t_2)}$ values.

Author(s)

W.H. Asquith

See Also

`quaray`, `theoTLmoms`

Examples

```
tlmrray(leftrim=2, rightrim=1, xi=0, alpha=2)
tlmrray(leftrim=2, rightrim=1, xi=10, alpha=2)
```

`tulia6Eprecip` *Annual Maximum Precipitation Data for Tulia 6E, Texas*

Description

Annual maximum precipitation data for Tulia 6E, Texas

Usage

```
data(tulia6Eprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(tulia6Eprecip)
summary(tulia6Eprecip)
```

`tuliaprecip` *Annual Maximum Precipitation Data for Tulia, Texas*

Description

Annual maximum precipitation data for Tulia, Texas

Usage

```
data(tuliaprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(tuliaprecip)
summary(tuliaprecip)
```

`USGSsta01515000peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 01515000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 01515000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta01515000peaks)
```

Format

A data frame with

Date The date of the annual peak streamflow.

Streamflow Annual peak streamflow data in cubic feet per second.

Flags Qualification flags on the streamflow data.

Stage Annual peak stage (gage height, river height) in feet.

Flags.1 Qualification flags on the gage height data.

Examples

```
data(USGSsta01515000peaks)
## Not run: plot(USGSsta01515000peaks)
```

`USGSsta02366500peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 02366500*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 02366500. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta02366500peaks)
```

Format

A data frame with

Date The date of the annual peak streamflow.

Streamflow Annual peak streamflow data in cubic feet per second.

Flags Qualification flags on the streamflow data.

Stage Annual peak stage (gage height, river height) in feet.

Flags.1 Qualification flags on the gage height data.

Examples

```
data(USGSsta02366500peaks)
## Not run: plot(USGSsta02366500peaks)
```

`USGSsta05405000peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 05405000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 05405000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta05405000peaks)
```

Format

A data frame with

agency_cd Agency code.

site_no Agency station number.

peak_dt The date of the annual peak streamflow.

peak_tm Time of the peak streamflow.

peak_va Annual peak streamflow data in cubic feet per second.

peak_cd Qualification flags on the streamflow data.

gage_ht Annual peak stage (gage height, river height) in feet.

gage_ht_cd Qualification flags on the gage height data.

year_last_pk Peak streamflow reported is the highest since this year.

ag_dt Date of maximum gage-height for water year (if not concurrent with peak).

ag_tm Time of maximum gage-height for water year (if not concurrent with peak).

ag_gage_ht Maximum gage height for water year in feet (if not concurrent with peak).

ag_gage_ht_cd Maximum gage height code.

Examples

```
data(USGSsta05405000peaks)
## Not run: plot(USGSsta05405000peaks)
```

`USGSsta06766000dvs` *Daily Mean Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 06766000*

Description

Daily mean streamflow data for U.S. Geological Survey streamflow-gaging station 06766000 PLATTE RIVER AT BRADY, NE. The qualification code `X01_00060_00003_cd` values are:

A Approved for publication – Processing and review completed.

1 Daily value is write protected without any remark code to be printed.

Usage

```
data(USGSsta06766000dvs)
```

Format

A data frame with

agency_cd The agency code `USGS`.

site_no The station identification number.

datetime The date and time of the data.

X01_00060_00003 The daily mean streamflow data in cubic feet per second.

X01_00060_00003_cd A code on the data value.

Examples

```
data(USGSsta06766000dvs)
## Not run: plot(USGSsta06766000dvs)
```

`USGSsta08151500peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 08151500*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08151500. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta08151500peaks)
```

Format

A data frame with

Date The date of the annual peak streamflow.

Streamflow Annual peak streamflow data in cubic feet per second.

Flags Qualification flags on the streamflow data.

Stage Annual peak stage (gage height, river height) in feet.

Examples

```
data(USGSsta08151500peaks)
## Not run: plot(USGSsta08151500peaks)
```

USGSsta08167000peaks *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 08167000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08167000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta08167000peaks)
```

Format

A data frame with

agency_cd Agency code.

site_no Agency station number.

peak_dt The date of the annual peak streamflow.

peak_tm Time of the peak streamflow.

peak_va Annual peak streamflow data in cubic feet per second.

peak_cd Qualification flags on the streamflow data.

gage_ht Annual peak stage (gage height, river height) in feet.

gage_ht_cd Qualification flags on the gage height data.

year_last_pk Peak streamflow reported is the highest since this year.

ag_dt Date of maximum gage-height for water year (if not concurrent with peak).

ag_tm Time of maximum gage-height for water year (if not concurrent with peak).

ag_gage_ht Maximum gage height for water year in feet (if not concurrent with peak).

ag_gage_ht_cd Maximum gage height code.

Examples

```
data(USGSsta08167000peaks)
## Not run: plot(USGSsta08167000peaks)
```

`USGSsta08190000peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 08190000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08190000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta08190000peaks)
```

Format

A data frame with

agency_cd Agency code.

site_no Agency station number.

peak_dt The date of the annual peak streamflow.

peak_tm Time of the peak streamflow.

peak_va Annual peak streamflow data in cubic feet per second.

peak_cd Qualification flags on the streamflow data.

gage_ht Annual peak stage (gage height, river height) in feet.

gage_ht_cd Qualification flags on the gage height data.

year_last_pk Peak streamflow reported is the highest since this year.

ag_dt Date of maximum gage-height for water year (if not concurrent with peak).

ag_tm Time of maximum gage-height for water year (if not concurrent with peak).

ag_gage_ht Maximum gage height for water year in feet (if not concurrent with peak).

ag_gage_ht_cd Maximum gage height code.

Examples

```
data(USGSsta08190000peaks)
## Not run: plot(USGSsta08190000peaks)
```

`USGSsta09442000peaks` *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 09442000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 09442000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta09442000peaks)
```

Format

A data frame with

Date The date of the annual peak streamflow.

Streamflow Annual peak streamflow data in cubic feet per second.

Flags Qualification flags on the streamflow data.

Stage Annual peak stage (gage height, river height) in feet.

Examples

```
data(USGSsta09442000peaks)
## Not run: plot(USGSsta09442000peaks)
```

USGSsta14321000peaks *Annual Peak Streamflow Data for U.S. Geological Survey Streamflow-Gaging Station 14321000*

Description

Annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 14321000. The peak streamflow-qualification codes `Flag` are:

1 Discharge is a Maximum Daily Average

2 Discharge is an Estimate

3 Discharge affected by Dam Failure

4 Discharge less than indicated value, which is Minimum Recordable Discharge at this site

5 Discharge affected to unknown degree by Regulation or Diversion

6 Discharge affected by Regulation or Diversion

7 Discharge is an Historic Peak

8 Discharge actually greater than indicated value

9 Discharge due to Snowmelt, Hurricane, Ice-Jam or Debris Dam breakup

A Year of occurrence is unknown or not exact

B Month or Day of occurrence is unknown or not exact

C All or part of the record affected by Urbanization, Mining, Agricultural changes, Channelization, or other

D Base Discharge changed during this year

E Only Annual Maximum Peak available for this year

The gage height qualification codes `Flag.1` are:

1 Gage height affected by backwater

2 Gage height not the maximum for the year

3 Gage height at different site and(or) datum

4 Gage height below minimum recordable elevation

5 Gage height is an estimate

6 Gage datum changed during this year

Usage

```
data(USGSsta14321000peaks)
```

Format

A data frame with

Date The date of the annual peak streamflow.

Streamflow Annual peak streamflow data in cubic feet per second.

Flags Qualification flags on the streamflow data.

Stage Annual peak stage (gage height, river height) in feet.

Flags.1 Qualification flags on the gage height data.

Examples

```
data(USGSsta14321000peaks)
## Not run: plot(USGSsta14321000peaks)
```

---

vec2lmom *Convert a Vector of L-moments to a L-moment Object*

Description

This function converts a vector of L-moments to a L-moment object of this package. The object is an R `list`. This function is intended to facilitate the use of L-moments (and Tl-moments) that the user might have from other sources. L-moments and L-moment ratios of arbitrary length are supported.

Because in typical practice, the $k \geq 3$ order L-moments are dimensionless ratios (τ_3, τ_4, and τ_5), this function computes λ_3, λ_4, λ_5 from λ_2 from the ratios. However, typical practice is not set on the use of λ_2 or τ as measure of dispersion. Therefore, this function takes an `lscale` optional logical (`TRUE|FALSE`) argument–if λ_2 is provided and `lscale=TRUE`, then τ is computed by the function and if τ is provided, then λ_2 is computed by the function.

Usage

```
vec2lmom(vec, lscale=TRUE, trim=NULL, leftrim=NULL, rightrim=NULL, checklmom=TRUE)
```

Arguments

`vec`	A vector of L-moment values in λ_1, λ_2 or τ, τ_3, τ_4, and τ_5 order.
`lscale`	A logical switch on the type of the second value of first argument. L-scale (λ_2) or LCV (τ). Default is `TRUE`, the second value in the first argument is λ_2.
`trim`	Level of symmetrical trimming, which should equal `NULL` if asymmetrical trimming is used.
`leftrim`	Level of trimming of the left-tail of the sample.

`rightrim`	Level of trimming of the right-tail of the sample.
`checklmom`	Should the `lmom` be checked for validity using the `are.lmom.valid` function. Normally this should be left as the default unless TL-moments are being constructed in liew of `vec2TLmom`.

Value

An R `list` is returned.

Author(s)

W.H. Asquith

See Also

`lmom.ub`, `vec2pwm`

Examples

```
lmr <- vec2lmom(c(12,0.6,0.34,0.20,0.05),lscale=FALSE)
```

`vec2par` *Convert a Vector of Parameters to a Parameter Object of a Distribution*

Description

This function converts a vector of parameters to a parameter object of a distribution. The type of distribution is specified in the argument list: `aep4`, `cau`, `exp`, `gam`, `gev`, `glo`, `gno`, `gpa`, `gum`, `kap`, `kur`, `lap`, `ln3`, `nor`, `pe3`, `ray`, `revgum`, `rice`, `texp`, `wak`, and `wei`. If the distribution type is not identified, then the function issues a warning, but goes ahead and creates the parameter list and of course can not check for the validity of the parameters.

Usage

```
vec2par(vec, type, nowarn=FALSE, paracheck=TRUE, ...)
```

Arguments

`vec`	A vector of parameter values for the distribution specified by type.
`type`	Three character distribution type (for example, `type='gev'`).
`nowarn`	A logical switch on warning surpression. If `TRUE` then `options(warn=-1)` is made and restored on return. This switch is to permit calls in which warnings are not desired as the user knows how to handle the returned value—say in an optimization algorithm.

paracheck — A logical controlling whether the parameters and checked for validity. Overriding of this check might be extremely important and needed for use of the distribution quantile function in the context of TL-moments with nonzero trimming.

... — Additional arguments for the are.par.valid call that is made internally.

Details

If the distribution is a Reverse Gumbel (type=revgum)) or Generalized Pareto (type=gpa) , which are two-parameter or three-parameter distributions, the third or fourth value in the vector is the ζ of the distribution. ζ represents the fraction of the sample that is noncensored, or number of observed (noncensored) values divided by the sample size. The ζ represents censoring on the right, that is there are unknown observations above a threshold or the largest observed sample. Consultation of parrevgum or pargpaRC should elucidate the censoring discussion.

Value

An R list is returned. This list should contain at least the following items, but some distributions such as the revgum have extra.

type — The type of distribution in three character format.

para — The parameters of the distribution.

source — Attribute specifying source of the parameters—“vec2par”.

Author(s)

W.H. Asquith

See Also

lmom2par

Examples

```
para <- vec2par(c(12,123,0.5),'gev')
Q <- quagev(0.5,para)

my.custom <- vec2par(c(2,2),'myowndist') # Rice distribution
```

---

vec2pwm *Convert a Vector of Probability-Weighted Moments to a Probability-Weighted Moments Object*

Description

This function converts a vector of Probability-Weighted Moments (PWM) to a PWM object of this package. The object is an R list. This function is intended to facilitate the use of PWM that the user might have from other sources. The first five PWMs are supported (β_0, β_1, β_2, β_3, β_4).

Usage

```
vec2pwm(vec, as.list=FALSE)
```

Arguments

`vec`	A vector of PWM values in (β_0, β_1, β_2, β_3, β_4) order.
`as.list`	A logical controlling the returned data structure.

Value

An R `list` is returned if `as.list=TRUE`.

`BETA0`	The first PWM–equal to the arithmetic mean.
`BETA1`	The second PWM.
`BETA2`	The third PWM.
`BETA3`	The fourth PWM.
`BETA4`	The fifth PWM.
`source`	Source of the PWMs: "vec2pwm"

Another R `list` is returned if `as.list=FALSE`.

`betas`	The PWMs
`source`	Source of the PWMs: "vec2pwm"

Author(s)

W.H. Asquith

See Also

`vec2lmom`

Examples

```
pwm <- vec2pwm(c(12,123,12,12,54))
```

vec2TLmom *Convert a Vector of TL-moments to a TL-moment Object*

Description

This function converts a vector of trimmed L-moments (TL-moments) to a TL-moment object of this package. The object is an R `list`. This function is intended to facilitate the use of TL-moments that the user might have from other sources. The trimming on the left-tail is denoted by t and the trimming on the right-tail is denoted as s. The first five TL-moments are $\lambda_1^{(t,s)}$, $\lambda_2^{(t,s)}$, $\lambda_3^{(t,s)}$, λ_4, $\lambda_5^{(t,s)}$, $\tau^{(t,s)}$, $\tau_3^{(t,s)}$, $\tau_4^{(t,s)}$, and $\tau_5^{(t,s)}$. The function supports TL-moments and TL-moment ratios of arbitrary length. Because in typical practice, the $k \geq 3$ order L-moments are dimensionless ratios ($\tau_3^{(t,s)}$, $\tau_4^{(t,s)}$, and $\tau_5^{(t,s)}$), this function computes $\lambda_3^{(t,s)}$, $\lambda_4^{(t,s)}$, $\lambda_5^{(t,s)}$ from $\lambda_2^{(t,s)}$ and the ratios. However, typical practice is not set on the use of $\lambda_2^{(t,s)}$ or $\tau^{(t,s)}$ as measure of dispersion. Therefore, this function takes an `lscale` optional logical argument—if $\lambda_2^{(t,s)}$ is provided and `lscale=TRUE`, then τ is computed by the function and if τ is provided, then $\lambda_2^{(t,s)}$ is computed by the function. The trim level of the TL-moment is required. Lastly, it might be common for $t = s$ and hence symmetrical trimming is used.

Usage

```
vec2TLmom(vec, ...)
```

Arguments

`vec`	A vector of L-moment values in $\lambda_1^{(t,s)}$, $\lambda_2^{(t,s)}$ or $\tau^{(t,s)}$, $\tau_3^{(t,s)}$, $\tau_4^{(t,s)}$, and $\tau_5^{(t,s)}$ order.
`...`	The arguments used by `vec2lmom`.

Value

An R `list` is returned where t represents the `trim` level.

`lambdas`	Vector of the TL-moments. First element is $\lambda_1^{(t,s)}$, second element is $\lambda_2^{(t,s)}$, and so on.
`ratios`	Vector of the L-moment ratios. Second element is $\tau^{(t,s)}$, third element is $\tau_3^{(t,s)}$ and so on.
`trim`	Level of symmetrical trimming, which should equal `NULL` if asymmetrical trimming is used.
`leftrim`	Level of trimming of the left-tail of the sample.
`rightrim`	Level of trimming of the right-tail of the sample.
`source`	An attribute identifying the computational source of the L-moments: “TLmoms”.

Author(s)

W.H. Asquith

See Also

TLmoms, vec2lmom

Examples

```
TL <- vec2TLmom(c(12,0.6,0.34,0.20,0.05),lscale=FALSE,trim=1)
```

---

vegaprecip *Annual Maximum Precipitation Data for Vega, Texas*

Description

Annual maximum precipitation data for Vega, Texas

Usage

```
data(vegaprecip)
```

Format

A data frame with

YEAR The calendar year of the annual maxima.

DEPTH The depth of 7-day annual maxima rainfall in inches.

References

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Examples

```
data(vegaprecip)
summary(vegaprecip)
```

z.par2cdf *Cumulative Distribution Function of Blipped Distributions*

Description

This function acts as a front end of dispatcher to the distribution-specific cumulative distribution functions.

$$x(F) = 0$$

for $0 \le F \le p$ and

$$x_G\left(\frac{F-p}{1-p}\right)$$

for $F > p$.

Usage

```
z.par2cdf(x,p,para,z=0,...)
```

Arguments

`x`	A real value.
`p`	Nonexceedance probability of the `z` value. This probability could simply be the portion of record having zero values if `z=0`.
`para`	The parameters from `lmom2par` or similar.
`z`	Threshold value.
`...`	The additional arguments are passed to the cumulative distribution function such as `paracheck=FALSE` for the Generalized Lambda distribution (`cdfgld`).

Value

Nonexceedance probability ($0 \le F \le 1$) for `x`.

Author(s)

W.H. Asquith

See Also

`z.par2qua`, `par2cdf`

Examples

```
# see the example for z.par2qua for more context
## define the real parent (or close)
the.gpa <- vec2par(c(100,1000,0.1),type='gpa')
fake.data <- rlmomco(30,the.gpa) # simulate some data
fake.data <- sort(c(fake.data,0,0,0,0)) # add of zero observations

# next compute the parameters for the positive data
gpa <- pargpa(lmoms(fake.data[fake.data > 0]))
n <- length(fake.data) # sample size
p <- length(fake.data[fake.data == 0])/n # est. prob of zero value
F <- nonexceeds() # handy values, to get nice range of x
x <- z.par2qua(F,p,gpa) # x are now computed

PP <- pp(fake.data) # compute plotting positions of sim. sample

plot(PP,fake.data,ylim=c(0,5000)) # plot the sample
lines(cdfgpa(x,the.gpa),x) # the parent (without zeros)
lines(z.par2cdf(x,p,gpa),x,lwd=3) # fitted model with zero conditional

# now repeat the above code over and over again and watch the results
```

`z.par2qua` *Quantile Function of the Blipped Distributions*

Description

This function acts as a front end or dispatcher to the distribution-specific quantile functions.

$$F(x) = 0$$

for $x \leq 0$ and

$$F(x) = p + (1-p)G(x)$$

for $x > 0$.

Usage

```
z.par2qua(f,p,para,z=0,...)
```

Arguments

`f`	Nonexceedance probability ($0 \leq F \leq 1$).
`p`	Nonexceedance probability of `z` value.
`para`	The parameters from `lmom2par` or similar.
`z`	Threshold value.
`...`	The additional arguments are passed to the quantile function such as `paracheck = FALSE` for the Generalized Lambda distribution (`quagld`).

Value

Quantile value for f.

Author(s)

W.H. Asquith

See Also

z.par2cdf, par2qua

Examples

```
# define the real parent (or close)
the.gpa <- vec2par(c(100,1000,0.1),type='gpa')
fake.data <- rlmomco(30,the.gpa) # simulate some data
fake.data <- sort(c(fake.data,0,0,0,0)) # add of zero observations

# next compute the parameters for the positive data
gpa <- pargpa(lmoms(fake.data[fake.data > 0]))
n <- length(fake.data) # sample size
p <- length(fake.data[fake.data == 0])/n # est. prob of zero value
F <- nonexceeds() # handy values
PP <- pp(fake.data) # compute plotting positions of sim. sample

plot(PP,fake.data,ylim=c(0,5000)) # plot the sample
lines(F,quagpa(F,the.gpa)) # the parent (without zeros)
lines(F,z.par2qua(F,p,gpa),lwd=3) # fitted model with zero conditional

# now repeat the above code over and over again and watch the results
```

Index