# Package ‘lmom’

February 20, 2015

**Version** 2.5

**Date** 2015-02-01

**Title** L-moments

**Author** J. R. M. Hosking

**Maintainer** J. R. M. Hosking <jrmhosking@gmail.com>

**Description** Functions related to L-moments: computation of L-moments and trimmed L-moments of distributions and data samples; parameter estimation; L-moment ratio diagram; plot vs. quantiles of an extreme-value distribution.

**Imports** stats, graphics



**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2015-02-02 00:52:00

## R topics documented:

---

`lmom-package` *The lmom package*

---

## Description

R functions for use with the method of $L$-moments

## Details

$L$-moments are measures of the location, scale, and shape of probability distributions or data samples. They are based on linear combinations of order statistics. Hosking (1990) and Hosking and Wallis (1997, chap. 2) give expositions of the theory of $L$-moments and $L$-moment ratios. Hosking and Wallis (1997, Appendix) give, for many distributions in common use, expressions for the $L$-moments of the distributions and algorithms for estimating the parameters of the distributions by equating sample and population $L$-moments (the "method of $L$-moments"). This package contains R functions that should facilitate the use of $L$-moment-based methods.

For each of 13 probability distributions, the package contains functions to evaluate the cumulative distribution function and quantile function of the distribution, to calculate the $L$-moments given the parameters and to calculate the parameters given the low-order $L$-moments. These functions are as follows.

`cdf...` computes the cumulative distribution function of the distribution.

`qua...` computes the quantile function (inverse cumulative distribution function) of the distribution.

`lmr...` calculates the $L$-moment ratios of the distribution given its parameters.

`pel...` calculates the parameters of the distribution given its $L$-moments. When the $L$-moments are the sample $L$-moments of a set of data, the resulting parameters are of course the "method of $L$-moments" estimates of the parameters.

Here `...` is a three-letter code used to identify the distribution, as given in the table below. For example the cumulative distribution function of the gamma distribution is `cdfgam`.

| | |
|---|---|
| `exp` | exponential |
| `gam` | gamma |
| `gev` | generalized extreme-value |
| `glo` | generalized logistic |
| `gpa` | generalized Pareto |
| `gno` | generalized normal |

| | |
|---|---|
| gum | Gumbel (extreme-value type I) |
| kap | kappa |
| ln3 | lognormal |
| nor | normal |
| pe3 | Pearson type III |
| wak | Wakeby |
| wei | Weibull |

The following functions are also contained in the package.

samlmu computes the sample $L$-moments of a data vector.

lmrp and lmrq compute the $L$-moments of a probability distribution specified by its cumulative distribution function (for function lmrp) or its quantile function (for function lmrq). The computation uses numerical integration applied to a general expression for the $L$-moments of a distribution. Functions lmrp and lmrq can be used for any univariate distribution. They are slower and usually less accurate than the computations carried out for specific distributions by the lmr... functions.

pelp and pelq compute the parameters of a probability distribution as a function of the $L$-moments. The computation uses function lmrp or lmrq to compute $L$-moments and numerical optimization to find parameter values for which the sample and population $L$-moments are equal. Functions pelp and pelq can be used for any univariate distribution. They are slower and usually less accurate than the computations carried out for specific distributions by the pel... functions.

lmrd draws an $L$-moment ratio diagram.

lmrdpoints and lmrdlines add points, or connected line segments, respectively, to an $L$-moment ratio diagram.

evplot draws an "extreme-value plot", i.e. a quantile-quantile plot in which the horizontal axis is the quantile of an extreme-value type I (Gumbel) distribution.

evpoints, evdistp, and evdistq add, respectively, a set of points, a cumulative distribution function, and a quantile function to an extreme-value plot.

## Trimmed $L$-moments

Some functions support the trimmed $L$-moments defined by Elamir and Seheult (2003). Trimmed $L$-moments are based on linear combinations of order statistics that give zero weight to the most extreme order statistics and thereby can be defined for very heavy-tailed distributions that do not have a finite mean.

Function samlmu can compute sample trimmed $L$-moments. Functions lmrp and lmrq can compute trimmed $L$-moments of probability distributions. Functions pelp and pelq can calculate parameters of a probability distribution given its trimmed $L$-moments.

The distribution-specific functions lmr... and pel... and the functions for $L$-moment ratio diagrams (lmrd, etc.) currently do not support trimmed $L$-moments.

## Parameters of cumulative distribution functions and quantile functions

The functions cdf... (cumulative distribution functions) and qua... (quantile functions) expect the distribution parameters to be specified as a single vector. This differs from the standard R convention, in which each parameter is a separate argument. There are two reasons for this. First, the single-vector parametrization is consistent with the Fortran routines on which these R functions are based. Second, the single-vector parametrization is often easier to use. For example, consider computing the 80th and 90th percentiles of a normal distribution fitted to a set of $L$-moments stored in a vector lmom. In the single-vector parametrization, this is achieved by

```
quanor( c(.8,.9), pelnor(lmom) )
```

The separate-arguments parametrization would need a more complex expression, such as

```
do.call( qnorm, c( list(.8,.9), pelnor(lmom) ) )
```

In functions (lmrp, lmrq, pelp, pelq, evplot, evdistp, evdistq) that take a cumulative distribution function or a quantile function as an argument, the cumulative distribution function or quantile function can use either form of parametrization.

## Relation to the LMOMENTS Fortran package

Functions cdf..., qua..., lmr..., pel..., and samlmu are analogous to Fortran routines from the LMOMENTS package, version 3.04, available from StatLib at http://lib.stat.cmu.edu/general/lmoments. Functions cdfwak and samlmu, and all the lmr... and pel... functions, internally call Fortran code that is derived from the LMOMENTS package.

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## References

Elamir, E. A. H., and Seheult, A. H. (2003). Trimmed L-moments. *Computational Statistics and Data Analysis*, **43**, 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*, **52**, 105-124.

Hosking, J. R. M., and Wallis, J. R. (1997). *Regional frequency analysis: an approach based on L-moments*. Cambridge University Press.

---

cdfexp *Exponential distribution*

---

## Description

Distribution function and quantile function of the exponential distribution.

## Usage

```
cdfexp(x, para = c(0, 1))
quaexp(f, para = c(0, 1))
```

### Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha$ (location, scale). |

### Details

The exponential distribution with parameters $\xi$ (lower bound) and $\alpha$ (mean) has distribution function

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

for $x \geq 0$, and quantile function

$$x(F) = \xi - \alpha \log(1 - F).$$

### Value

`cdfexp` gives the distribution function; `quaexp` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R functions `pexp` and `qexp`.

### See Also

`pexp` for the standard R version of the exponential distribution. `cdfgam` for the gamma distribution, `cdfgpa` for the generalized Pareto distribution, `cdfkap` for the kappa distribution, `cdfpe3` for the Pearson type III distribution, and `cdfwak` for the Wakeby distribution, all of which generalize the exponential distribution.

### Examples

```
# Random sample from the exponential distribution
# with lower bound 0 and mean 3.
quaexp(runif(100), c(0,3))
```

---

## `cdfgam` *Gamma distribution*

---

### Description

Distribution function and quantile function of the gamma distribution.

### Usage

```
cdfgam(x, para = c(1, 1))
quagam(f, para = c(1, 1))
```

### Arguments

x Vector of quantiles.

f Vector of probabilities.

para Numeric vector containing the parameters of the distribution, in the order $\alpha, \beta$ (shape, scale).

### Details

The gamma distribution with shape parameter $\alpha$ and scale parameter $\beta$ has probability density function

$$f(x) = \frac{x^{\alpha-1}\exp(-x/\beta)}{\beta^\alpha \Gamma(\alpha)}$$

for $x \geq 0$, where $\Gamma(.)$ is the gamma function.

### Value

`cdfgam` gives the distribution function; `quagam` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R functions `pgamma` and `qgamma`.

### See Also

`gamma` for the gamma function.

`pgamma` for the standard R version of the gamma distribution.

`cdfpe3` for the Pearson type III distribution, which generalizes the gamma distribution.

### Examples

```
# Random sample from the gamma distribution
# with shape parameter 4 and mean 1.
quagam(runif(100), c(4,1/4))
```

---

## cdfgev *Generalized extreme-value distribution*

---

### Description

Distribution function and quantile function of the generalized extreme-value distribution.

### Usage

```
cdfgev(x, para = c(0, 1, 0))
quagev(f, para = c(0, 1, 0))
```

### Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, k$ (location, scale, shape). |

### Details

The generalized extreme-value distribution with location parameter $\xi$, scale parameter $\alpha$ and shape parameter $k$ has distribution function

$$F(x) = \exp\{-\exp(-y)\}$$

where

$$y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\},$$

with $x$ bounded by $\xi + \alpha/k$ from below if $k < 0$ and from above if $k > 0$, and quantile function

$$x(F) = \xi + \frac{\alpha}{k}\{1 - (-\log F)^k\}.$$

Extreme-value distribution types I, II and III (Gumbel, Frechet, Weibull) correspond to shape parameter values $k = 0$, $k < 0$ and $k > 0$ respectively.

### Value

`cdfgev` gives the distribution function; `quagev` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

### See Also

`cdfgum` for the Gumbel (extreme-value type I) distribution.

`cdfkap` for the kappa distribution, which generalizes the generalized extreme-value distribution.

`cdfwei` for the Weibull distribution,

### Examples

```
# Random sample from the generalized extreme-value distribution
# with parameters xi=0, alpha=1, k=-0.5.
quagev(runif(100), c(0,1,-0.5))
```

`cdfglo` *Generalized logistic distribution*

## Description

Distribution function and quantile function of the generalized logistic distribution.

## Usage

```
cdfglo(x, para = c(0, 1, 0))
quaglo(f, para = c(0, 1, 0))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, k$ (location, scale, shape). |

## Details

The generalized logistic distribution with location parameter $\xi$, scale parameter $\alpha$ and shape parameter $k$ has distribution function

$$F(x) = 1/\{1 + \exp(-y)\}$$

where

$$y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\},$$

with $x$ bounded by $\xi + \alpha/k$ from below if $k < 0$ and from above if $k > 0$, and quantile function

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

The logistic distribution is the special case $k = 0$.

## Value

`cdfglo` gives the distribution function; `quaglo` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## See Also

`cdfkap` for the kappa distribution, which generalizes the generalized logistic distribution.

## Examples

```
# Random sample from the generalized logistic distribution
# with parameters xi=0, alpha=1, k=-0.5.
quaglo(runif(100), c(0,1,-0.5))
```

---

`cdfgno` *Generalized normal distribution*

---

## Description

Distribution function and quantile function of the generalized normal distribution.

## Usage

```
cdfgno(x, para = c(0, 1, 0))
quagno(f, para = c(0, 1, 0))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, k$ (location, scale, shape). |

## Details

The generalized normal distribution with location parameter $\xi$, scale parameter $\alpha$ and shape parameter $k$ has distribution function

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

where

$$y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\}$$

and $\Phi(y)$ is the distribution function of the standard normal distribution, with $x$ bounded by $\xi + \alpha/k$ from below if $k < 0$ and from above if $k > 0$.

The generalized normal distribution contains as special cases the usual three-parameter lognormal distribution, corresponding to $k < 0$, with a finite lower bound and positive skewness; the normal distribution, corresponding to $k = 0$; and the reverse lognormal distribution, corresponding to $k > 0$, with a finite upper bound and negative skewness. The two-parameter lognormal distribution, with a lower bound of zero and positive skewness, is obtained when $k < 0$ and $\xi + \alpha/k = 0$.

## Value

`cdfgno` gives the distribution function; `quagno` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## See Also

`cdfln3` for the **lmom** package's version of the three-parameter lognormal distribution.

`cdfnor` for the **lmom** package's version of the normal distribution.

`pnorm` for the standard R version of the normal distribution.

`plnorm` for the standard R version of the two-parameter lognormal distribution.

## Examples

```
# Random sample from the generalized normal distribution
# with parameters xi=0, alpha=1, k=-0.5.
quagno(runif(100), c(0,1,-0.5))

# The generalized normal distribution with parameters xi=1, alpha=1, k=-1,
# is the standard lognormal distribution.  An illustration:
fval<-seq(0.1,0.9,by=0.1)
cbind(fval, lognormal=qlnorm(fval), g.normal=quagno(fval, c(1,1,-1)))
```

---

`cdfgpa` *Generalized Pareto distribution*

---

## Description

Distribution function and quantile function of the generalized Pareto distribution.

## Usage

```
cdfgpa(x, para = c(0, 1, 0))
quagpa(f, para = c(0, 1, 0))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, k$ (location, scale, shape). |

### Details

The generalized Pareto distribution with location parameter $\xi$, scale parameter $\alpha$ and shape parameter $k$ has distribution function

$$F(x) = 1 - \exp(-y)$$

where

$$y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\},$$

with $x$ bounded by $\xi + \alpha/k$ from below if $k < 0$ and from above if $k > 0$, and quantile function

$$x(F) = \xi + \frac{\alpha}{k}\{1 - (1 - F)^k\}.$$

The exponential distribution is the special case $k = 0$. The uniform distribution is the special case $k = 1$.

### Value

`cdfgpa` gives the distribution function; `quagpa` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

### See Also

`cdfexp` for the exponential distribution.

`cdfkap` for the kappa distribution and `cdfwak` for the Wakeby distribution, which generalize the generalized Pareto distribution.

### Examples

```
# Random sample from the generalized Pareto distribution
# with parameters xi=0, alpha=1, k=-0.5.
quagpa(runif(100), c(0,1,-0.5))
```

---

## cdfgum — *Gumbel (extreme-value type I) distribution*

---

### Description

Distribution function and quantile function of the Gumbel distribution.

### Usage

```
cdfgum(x, para = c(0, 1))
quagum(f, para = c(0, 1))
```

### Arguments

| | |
|---|---|
| x | Vector of quantiles. |
| f | Vector of probabilities. |
| para | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha$ (location, scale). |

### Details

The Gumbel distribution with location parameter $\xi$ and scale parameter $\alpha$ has distribution function

$$F(x) = \exp[-\exp\{-(x-\xi)/\alpha\}]$$

and quantile function

$$x(F) = \xi - \alpha \log(-\log F).$$

### Value

`cdfgum` gives the distribution function; `quagum` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

### See Also

`cdfgev` for the generalized extreme-value distribution, which generalizes the Gumbel distribution.

### Examples

```
# Random sample from the Gumbel distribution with parameters xi=0, alpha=3.
quagum(runif(100), c(0,3))
```

---

## cdfkap — *Kappa distribution*

---

### Description

Distribution function and quantile function of the kappa distribution.

### Usage

```
cdfkap(x, para = c(0, 1, 0, 0))
quakap(f, para = c(0, 1, 0, 0))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, k, h$ (location, scale, shape, shape). |

## Details

The kappa distribution with location parameter $\xi$, scale parameter $\alpha$ and shape parameters $k$ and $h$ has quantile function

$$x(F) = \xi + \frac{\alpha}{k}\left\{1 - \left(\frac{1-F^h}{h}\right)^k\right\}.$$

Its special cases include the generalized logistic ($h = -1$), generalized extreme-value ($h = 0$), generalized Pareto ($h = 1$), logistic ($k = 0$, $h = -1$), Gumbel ($k = 0$, $h = 0$), exponential ($k = 0$, $h = 1$), and uniform ($k = 1$, $h = 1$) distributions.

## Value

`cdfkap` gives the distribution function; `quakap` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## References

Hosking, J. R. M. (1994). The four-parameter kappa distribution. *IBM Journal of Research and Development*, **38**, 251-258.

Hosking, J. R. M., and Wallis, J. R. (1997). *Regional frequency analysis: an approach based on L-moments*, Cambridge University Press, Appendix A.10.

## See Also

`cdfglo` for the generalized logistic distribution, `cdfgev` for the generalized extreme-value distribution, `cdfgpa` for the generalized Pareto distribution, `cdfgum` for the Gumbel distribution, `cdfexp` for the exponential distribution.

## Examples

```
# Random sample from the kappa distribution
# with parameters xi=0, alpha=1, k=-0.5, h=0.25.
quakap(runif(100), c(0,1,-0.5,0.25))
```

## cdfln3 *Three-parameter lognormal distribution*

### Description

Distribution function and quantile function of the three-parameter lognormal distribution.

### Usage

```
cdfln3(x, para = c(0, 0, 1))
qualn3(f, para = c(0, 0, 1))
```

### Arguments

| | |
|---|---|
| x | Vector of quantiles. |
| f | Vector of probabilities. |
| para | Numeric vector containing the parameters of the distribution, in the order $\zeta, \mu, \sigma$ (lower bound, mean on log scale, standard deviation on log scale). |

### Details

The three-parameter lognormal distribution with lower bound $\zeta$, mean on log scale $\mu$, and standard deviation on log scale $\sigma$ has distribution function

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

$x > 0$, where

$$y = \{\log(x - \zeta) - \mu\}/\sigma$$

and $\Phi(y)$ is the distribution function of the standard normal distribution.

### Value

`cdfln3` gives the distribution function; `qualn3` gives the quantile function.

### Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

### See Also

`cdfgno` for the generalized normal distribution, a more general form of the three-parameter lognormal distribution.

`qlnorm` for the standard R version of the two-parameter lognormal distribution.

## Examples

```
# Random sample from three-parameter lognormal distribution
# with parameters zeta=0, mu=1, sigma=0.5.
qualn3(runif(100), c(0,1,0.5))

## Functions for the three-parameter lognormal distribution can
## also be used with the two-parameter lognormal distribution
# Generate a random sample from a standard lognormal distribution
xx <- qualn3(runif(50))
# Fit 2-parameter LN distribution
pelln3(samlmu(xx), bound=0)
# Fit 2-parameter LN distribution "in log space",
# i.e. fit normal distribution to log-transformed data
pelnor(samlmu(log(xx)))
```

---

`cdfnor` *Normal distribution*

---

## Description

Distribution function and quantile function of the normal distribution.

## Usage

```
cdfnor(x, para = c(0, 1))
quanor(f, para = c(0, 1))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\mu, \sigma$ (location, scale). |

## Details

The normal distribution with location parameter $\mu$ and scale parameter $\sigma$ has probability density function

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\{-(x-\mu)^2/(2\sigma^2)\}.$$

## Value

`cdfnor` gives the distribution function; `quanor` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm` and `qnorm`.

### See Also

pnorm for the standard R version of the normal distribution.

cdfgno for the generalized normal distribution, which generalizes the normal distribution.

### Examples

```
# Random sample from the normal distribution
# with mean 0 and standard deviation 3.
quanor(runif(100), c(0,3))
```

---

## cdfpe3 — *Pearson type III distribution*

---

### Description

Distribution function and quantile function of the Pearson type III distribution

### Usage

```
cdfpe3(x, para = c(0, 1, 0))
quape3(f, para = c(0, 1, 0))
```

### Arguments

| | |
|---|---|
| x | Vector of quantiles. |
| f | Vector of probabilities. |
| para | Numeric vector containing the parameters of the distribution, in the order $\mu, \sigma, \gamma$ (location, scale, shape). |

### Details

The Pearson type III distribution contains as special cases the usual three-parameter gamma distribution (a shifted version of the gamma distribution) with a finite lower bound and positive skewness; the normal distribution, and the reverse three-parameter gamma distribution, with a finite upper bound and negative skewness. The distribution's parameters are the first three (ordinary) moment ratios: $\mu$ (the mean, a location parameter), $\sigma$ (the standard deviation, a scale parameter) and $\gamma$ (the skewness, a shape parameter).

If $\gamma \neq 0$, let $\alpha = 4/\gamma^2$, $\beta = \frac{1}{2}\sigma|\gamma|$, $\xi = \mu - 2\sigma/\gamma$. The probability density function is

$$f(x) = \frac{|x-\xi|^{\alpha-1}\exp(-|x-\xi|/\beta)}{\beta^{\alpha}\Gamma(\alpha)}$$

with $x$ bounded by $\xi$ from below if $\gamma > 0$ and from above if $\gamma < 0$. If $\gamma = 0$, the distribution is a normal distribution with mean $\mu$ and standard deviation $\sigma$.

The Pearson type III distribution is usually regarded as consisting of just the case $\gamma > 0$ given above, and is usually parametrized by $\alpha$, $\beta$ and $\xi$. Our parametrization extends the distribution to include the usual Pearson type III distributions, with positive skewness and lower bound $\xi$, reverse Pearson type III distributions, with negative skewness and upper bound $\xi$, and the Normal distribution, which is included as a special case of the distribution rather than as the unattainable limit $\alpha \to \infty$. This enables the Pearson type III distribution to be used when the skewness of the observed data may be negative. The parameters $\mu$, $\sigma$ and $\gamma$ are the conventional moments of the distribution.

The gamma distribution is obtained when $\gamma > 0$ and $\mu = 2\sigma/\gamma$. The normal distribution is the special case $\gamma = 0$. The exponential distribution is the special case $\gamma = 2$.

## Value

`cdfpe3` gives the distribution function; `quape3` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## References

Hosking, J. R. M. and Wallis, J. R. (1997). *Regional frequency analysis: an approach based on L-moments*, Cambridge University Press, Appendix A.10.

## See Also

`cdfgam` for the gamma distribution.

`cdfnor` for the normal distribution.

## Examples

```
# Random sample from the Pearson type III distribution
# with parameters mu=1, alpha=2, gamma=3.
quape3(runif(100), c(1,2,3))

# The Pearson type III distribution with parameters
# mu=12, sigma=6, gamma=1, is the gamma distribution
# with parameters alpha=4, beta=3.  An illustration:
fval<-seq(0.1,0.9,by=0.1)
cbind(fval, qgamma(fval, shape=4, scale=3), quape3(fval, c(12,6,1)))
```

---

# cdfwak *Wakeby distribution*

---

## Description

Distribution function and quantile function of the Wakeby distribution.

## Usage

```
cdfwak(x, para = c(0, 1, 0, 0, 0))
quawak(f, para = c(0, 1, 0, 0, 0))
```

## Arguments

| | |
|---|---|
| `x` | Vector of quantiles. |
| `f` | Vector of probabilities. |
| `para` | Numeric vector containing the parameters of the distribution, in the order $\xi, \alpha, \beta, \gamma, \delta$. |

## Details

The Wakeby distribution with parameters $\xi$, $\alpha$, $\beta$, $\gamma$ and $\delta$ has quantile function

$$x(F) = \xi + \frac{\alpha}{\beta}\{1 - (1-F)^{\beta}\} - \frac{\gamma}{\delta}\{1 - (1-F)^{-\delta}\}.$$

The parameters are restricted as in Hosking and Wallis (1997, Appendix A.11):

- either $\beta + \delta > 0$ or $\beta = \gamma = \delta = 0$;
- if $\alpha = 0$ then $\beta = 0$;
- if $\gamma = 0$ then $\delta = 0$;
- $\gamma \geq 0$;
- $\alpha + \gamma \geq 0$.

The distribution has a lower bound at $\xi$ and, if $\delta < 0$, an upper bound at $\xi + \alpha/\beta - \gamma/\delta$.

The generalized Pareto distribution is the special case $\alpha = 0$ or $\gamma = 0$. The exponential distribution is the special case $\beta = \gamma = \delta = 0$. The uniform distribution is the special case $\beta = 1$, $\gamma = \delta = 0$.

## Value

`cdfwak` gives the distribution function; `quawak` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## References

Hosking, J. R. M. and Wallis, J. R. (1997). *Regional frequency analysis: an approach based on L-moments*, Cambridge University Press, Appendix A.11.

## See Also

`cdfgpa` for the generalized Pareto distribution.

`cdfexp` for the exponential distribution.

## Examples

```
# Random sample from the Wakeby distribution
# with parameters xi=0, alpha=30, beta=20, gamma=1, delta=0.3.
quawak(runif(100), c(0,30,20,1,0.3))
```

---

# cdfwei *Weibull distribution*

---

## Description

Distribution function and quantile function of the Weibull distribution.

## Usage

```
cdfwei(x, para = c(0, 1, 1))
quawei(f, para = c(0, 1, 1))
```

## Arguments

| | |
|---|---|
| x | Vector of quantiles. |
| f | Vector of probabilities. |
| para | Numeric vector containing the parameters of the distribution, in the order $\zeta, \beta, \delta$ (location, scale, shape). |

## Details

The Weibull distribution with location parameter $\zeta$, scale parameter $\beta$ and shape parameter $\delta$ has distribution function

$$F(x) = 1 - \exp[-\{(x - \zeta)/\beta\}^{\delta}]$$

for $x > \zeta$.

## Value

`cdfwei` gives the distribution function; `quawei` gives the quantile function.

## Note

The functions expect the distribution parameters in a vector, rather than as separate arguments as in the standard R distribution functions `pnorm`, `qnorm`, etc.

## See Also

`cdfgev` for the generalized extreme-value distribution, of which the Weibull (reflected through the origin) is a special case.

### Examples

```
# Random sample from a 2-parameter Weibull distribution
# with scale parameter 2 and shape parameter 1.5.
quawei(runif(100), c(0,2,1.5))

# Illustrate the relation between Weibull and GEV distributions.
# weifit() fits a Weibull distribution to data and returns
#   quantiles of the fitted distribution
# gevfit() fits a Weibull distribution as a "reverse GEV",
#   i.e. fits a GEV distribution to the negated data,
#   then computes negated quantiles
weifit <- function(qval, x) quawei(qval, pelwei(samlmu(x)))
gevfit <- function(qval, x) -quagev(1-qval, pelgev(samlmu(-x)))
# Compare on Ozone data
data(airquality)
weifit(c(0.2,0.5,0.8), airquality$Ozone)
gevfit(c(0.2,0.5,0.8), airquality$Ozone)
```

---

## evplot — *Extreme-value plot*

---

### Description

evplot draws an “extreme-value plot”, i.e. a quantile-quantile plot in which the horizontal axis is the quantile of an extreme-value type I (Gumbel) distribution.

evdistp adds the cumulative distribution function of a distribution to an extreme-value plot.

evdistq adds the quantile function of a distribution to an extreme-value plot.

evpoints adds a set of data points to an extreme-value plot.

### Usage

```
evplot(y, ...)

## Default S3 method:
evplot(y, qfunc, para, npoints = 101, plim, xlim = c(-2, 5),
       ylim, type,
       xlab = expression("Reduced variate,  " * -log(-log(italic(F)))),
       ylab = "Quantile", rp.axis = TRUE, ...)

evdistp(pfunc, para, npoints = 101, ...)
evdistq(qfunc, para, npoints = 101, ...)

evpoints(y, ...)
```

## Arguments

| | |
|---|---|
| `y` | Numeric vector. The data values in the vector are plotted on the extreme-value plot. |
| `qfunc` | A quantile function. The function is drawn as a curve on the extreme-value plot. |
| `pfunc` | A cumulative distribution function. The function is drawn as a curve on the extreme-value plot. |
| `para` | Distribution parameters for the quantile function `qfunc` or cumulative distribution function `pfunc`.<br>If `pfunc` or `qfunc` is the standard R form of quantile function, `para` should be a list.<br>If `pfunc` or `qfunc` is the `qua...` form of quantile function used throughout the **lmom** package, `para` should be a numeric vector.<br>In `evplot`, `para` is not used if `qfunc` is omitted. |
| `npoints` | Number of points to use in drawing the quantile function. The points are equally spaced along the x axis. Not used if `qfunc` is omitted. |
| `plim` | X axis limits, specified as probabilities. |
| `xlim` | X axis limits, specified as values of the Gumbel reduced variate $-\log(-\log F)$, where $F$ is the nonexceedance probability. Not used if `plim` is specified. |
| `ylim` | Y axis limits. |
| `type` | Plot type. Determines how the data values in `y` are plotted. Interpreted in the same way as argument `type` of function `plot`, i.e. `"p"` for points, `"b"` for points connected by lines, etc. |
| `xlab` | X axis label. |
| `ylab` | Y axis label. |
| `rp.axis` | Logical. Whether to draw the “Return period” axis, a secondary horizontal axis. |
| `...` | Additional arguments are passed to the plotting routine. |

## Arguments of cumulative distribution functions and quantile functions

`pfunc` and `qfunc` can be either the standard R form of cumulative distribution function or quantile function (i.e. for a distribution with $r$ parameters, the first argument is the variate $x$ or the probability $p$ and the next $r$ arguments are the parameters of the distribution) or the `cdf...` or `qua...` forms used throughout the **lmom** package (i.e. the first argument is the variate $x$ or probability $p$ and the second argument is a vector containing the parameter values).

## Note

Data points are plotted at the Gringorten plotting position, i.e. the $i$th smallest of $n$ data points is plotted at the horizontal position corresponding to nonexceedance probability $(i-0.44)/(n+0.12)$.

## Author(s)

J. R. M. Hosking `<jrmhosking@gmail.com>`

## Examples

```
# Extreme-value plot of Ozone from the airquality data
data(airquality)
evplot(airquality$Ozone)

# Fit a GEV distribution and add it to the plot
evdistq(quagev, pelgev(samlmu(airquality$Ozone)))

# Not too good -- try a kappa distribution instead
evdistq(quakap, pelkap(samlmu(airquality$Ozone)), col="red")
```

---

`lmr-functions` *L-moments of specific probability distributions*

---

## Description

Computes the $L$-moments of a probability distribution given its parameters. The following distributions are recognized:

| | |
|---|---|
| `lmrexp` | exponential |
| `lmrgam` | gamma |
| `lmrgev` | generalized extreme-value |
| `lmrglo` | generalized logistic |
| `lmrgpa` | generalized Pareto |
| `lmrgno` | generalized normal |
| `lmrgum` | Gumbel (extreme-value type I) |
| `lmrkap` | kappa |
| `lmrln3` | three-parameter lognormal |
| `lmrnor` | normal |
| `lmrpe3` | Pearson type III |
| `lmrwak` | Wakeby |
| `lmrwei` | Weibull |

## Usage

```
lmrexp(para = c(0, 1), nmom = 2)
lmrgam(para = c(1, 1), nmom = 2)
lmrgev(para = c(0, 1, 0), nmom = 3)
lmrglo(para = c(0, 1, 0), nmom = 3)
lmrgno(para = c(0, 1, 0), nmom = 3)
lmrgpa(para = c(0, 1, 0), nmom = 3)
lmrgum(para = c(0, 1), nmom = 2)
lmrkap(para = c(0, 1, 0, 0), nmom = 4)
lmrln3(para = c(0, 0, 1), nmom = 3)
lmrnor(para = c(0, 1), nmom = 2)
lmrpe3(para = c(0, 1, 0), nmom = 3)
```

```
lmrwak(para = c(0, 1, 0, 0, 0), nmom = 5)
lmrwei(para = c(0, 1, 1), nmom = 3)
```

## Arguments

| | |
|---|---|
| para | Numeric vector containing the parameters of the distribution. |
| nmom | The number of $L$-moments to be calculated. |

## Details

Numerical methods and accuracy are as described in Hosking (1996, pp. 8–9).

## Value

Numeric vector containing the $L$-moments.

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## References

Hosking, J. R. M. (1996). Fortran routines for use with the method of $L$-moments, Version 3. Research Report RC20525, IBM Research Division, Yorktown Heights, N.Y.

## See Also

lmrp to compute $L$-moments of a general distribution specified by its cumulative distribution function or quantile function.

samlmu to compute $L$-moments of a data sample.

pelexp, etc., to compute the parameters of a distribution given its $L$-moments.

For individual distributions, see their cumulative distribution functions:

| | |
|---|---|
| cdfexp | exponential |
| cdfgam | gamma |
| cdfgev | generalized extreme-value |
| cdfglo | generalized logistic |
| cdfgpa | generalized Pareto |
| cdfgno | generalized normal |
| cdfgum | Gumbel (extreme-value type I) |
| cdfkap | kappa |
| cdfln3 | three-parameter lognormal |
| cdfnor | normal |
| cdfpe3 | Pearson type III |
| cdfwak | Wakeby |
| cdfwei | Weibull |

### Examples

```
# Compare sample L-moments of Ozone from the airquality data
# with the L-moments of a GEV distribution fitted to the data
data(airquality)
smom <- samlmu(airquality$Ozone, nmom=6)
gevpar <- pelgev(smom)
pmom <- lmrgev(gevpar, nmom=6)
print(smom)
print(pmom)
```

---

## lmrd *L-moment ratio diagram*

---

### Description

Draws an $L$-moment ratio diagram.

### Usage

```
lmrd(x, y, distributions = "GLO GEV GPA GNO PE3", twopar,
     xlim, ylim, pch=3, cex, col, lty, lwd=1,
     legend.lmrd = TRUE, xlegend, ylegend,
     xlab = expression(italic(L) * "-skewness"),
     ylab = expression(italic(L) * "-kurtosis"), ...)
```

### Arguments

`x` Numeric vector of $L$-skewness values.

Alternatively, if argument `y` is omitted, `x` can be an object that contains both $L$-skewness and $L$-kurtosis values. It can be a vector with elements named `"t_3"` and `"t_4"` (or `"tau_3"` and `"tau_4"`), a matrix or data frame with columns named `"t_3"` and `"t_4"` (or `"tau_3"` and `"tau_4"`), or an object of class `"regdata"` (as defined in package **lmomRFA**).

`y` Numeric vector of $L$-kurtosis values.

`distributions` Indicates the three-parameter distributions whose $L$-skewness–$L$-kurtosis relations are to be plotted as lines on the diagram. The following distribution identifiers are recognized, in upper or lower case:

| | |
|---|---|
| GLO | generalized logistic |
| GEV | generalized extreme-value |
| GPA | generalized Pareto |
| GNO | generalized normal |
| PE3 | Pearson type III |
| WAK.LB | lower bound of $L$-kurtosis for given $L$-skewness, for the Wakeby distribution. |
| ALL.LB | lower bound of $L$-kurtosis for given $L$-skewness, for all distributions. |

The argument should be either a character vector each of whose elements is one of the above abbreviations or a character string containing one or more of the abbreviations separated by blanks. The specified $L$-skewness–$L$-kurtosis curves will be plotted.

If no three-parameter distributions are to be plotted, specify `distributions` to be `FALSE` or the empty string, `""`.

`twopar` Two-parameter distributions whose ($L$-skewness, $L$-kurtosis) values are to be plotted as points on the diagram. The following distribution identifiers are recognized, in upper or lower case:

| | |
|---|---|
| `E` or `EXP` | exponential |
| `G` or `GUM` | Gumbel |
| `L` or `LOG` | logistic |
| `N` or `NOR` | normal |
| `U` or `UNI` | uniform |

The argument should be either a character vector each of whose elements is one of the above abbreviations or a character string containing one or more of the abbreviations separated by blanks. $L$-skewness–$L$-kurtosis points for the specified distributions will be plotted and given one-character labels.

The default is to plot the two-parameter distributions that are special cases of the three-parameter distributions specified in argument `distributions`. Thus for example if `distributions="GPA PE3"`, the default for `twopar` is `"EXP NOR UNI"`: NOR is a special case of PE3, UNI of GPA, EXP of both GPA and PE3.

If no two-parameter distributions are to be plotted, specify `twopar` to be `FALSE` or the empty string, `""`.

`xlim` x axis limits. Default: `c(0, 0.6)`, expanded if necessary to cover the range of the data.

`ylim` y axis limits. Default: `c(0, 0.4)`, expanded if necessary to cover the range of the data.

`pch` Plotting character to be used for the plotted ($L$-skewness, $L$-kurtosis) points.

`cex` Symbol size for plotted points, like graphics parameter `cex`.

`col` Vector specifying the colors. If it is of length 1 and `x` is present, it will be used for the plotted points. Otherwise it will be used for the lines on the plot. For the default colors for the lines, see the description of argument `lty` below.

`lty` Vector specifying the line types to be used for the lines on the plot.

By default, colors and line types are matched to the distributions given in argument `distributions`, as follows:

| | |
|---|---|
| `GLO` | blue, solid line |
| `GEV` | green, solid line |
| `GPA` | red, solid line |
| `GNO` | black, solid line |
| `PE3` | cyan, solid line |
| `WAK.LB` | red, dashed line |
| `ALL.LB` | black, dashed line |

| | |
|---|---|
| | The green and cyan colors are less bright than the standard "green" and "cyan"; they are defined to be "#00C000" and "#00E0E0", respectively. |
| lwd | Vector specifying the line widths to be used for the lines on the plot. |
| legend.lmrd | Controls whether a legend, identifying the $L$-skewness–$L$-kurtosis relations of the three-parameter distributions, is plotted. Either logical, indicating whether a legend is to be drawn, or a list specifying arguments to the legend function. Default arguments include bty="n", which must be overridden if a legend box is to be drawn; other arguments set by default are x, y, legend, col, lty, and lwd.<br>Not used if distributions=FALSE. |
| xlegend | x coordinate of the upper left corner of the legend. Default: the minimum x value. Not used if distributions=FALSE or legend.lmrd=FALSE. |
| ylegend | y coordinate of the upper left corner of the legend. Default: the maximum y value. Not used if distributions=FALSE or legend.lmrd=FALSE. |
| xlab | X axis label. |
| ylab | Y axis label. |
| ... | Additional arguments are passed to the matplot function that draws the axis box and the lines for three-parameter distributions. |

## Details

lmrd calls a sequence of graphics functions: matplot for the axis box and the curves for three-parameter distributions; points for the points for two-parameter distributions and text for their labels; legend for the legend; and points for the $(x, y)$ data points.

Note that the only graphics parameters passed to points are col (if of length 1), cex, and pch. If more complex features are required, such as different colors for different points, follow lmrd by an additional call to points, e.g. follow lmrd(t3, t4) by points(t3, t4, col=c("red", "green")).

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## Examples

```
data(airquality)
lmrd(samlmu(airquality$Ozone))

# Tweaking a few graphics parameters makes the graph look better
# (in the author's opinion)
lmrd(samlmu(airquality$Ozone), xaxs="i", yaxs="i", las=1)

# An example that illustrates the sampling variability of L-moments
#
# Generate 50 random samples of size 30 from the Gumbel distribution
# - stored in the rows of matrix mm
mm <- matrix(quagum(runif(1500)), nrow=50)
#
# Compute the first four sample L-moments of each sample
```

```
# - stored in the rows of matrix aa
aa <- apply(mm, 1, samlmu)
#
# Plot the L-skewness and L-kurtosis values on an L-moment ratio
# diagram that also shows (only) the population L-moment ratios
# of the Gumbel distribution
lmrd(t(aa), dist="", twopar="G", col="red")
```

---

`lmrdpoints` *Add points or lines to an L-moment ratio diagram*

---

## Description

`lmrdpoints` adds points, and `lmrdlines` adds connected line segments, to an $L$-moment ratio diagram.

## Usage

```
lmrdpoints(x, y=NULL, type="p", ...)
lmrdlines(x, y=NULL, type="l", ...)
```

## Arguments

| | |
|---|---|
| `x` | Numeric vector of $L$-skewness values. |
| `y` | Numeric vector of $L$-kurtosis values. May be omitted: see “Details” below. |
| `type` | Character indicating the type of plotting. Can be any valid value for the `type` argument of `plot.default`. |
| `...` | Further arguments (graphics parameters), passed to `points` or `lines`. |

## Details

The functions `lmrdpoints` and `lmrdlines` are equivalent to `points` and `lines` respectively, except that if argument `y` is omitted, `x` is assumed to be an object that contains both $L$-skewness and $L$-kurtosis values. As in `lmrd`, it can be a vector with elements named `"t_3"` and `"t_4"` (or `"tau_3"` and `"tau_4"`), a matrix or data frame with columns named `"t_3"` and `"t_4"` (or `"tau_3"` and `"tau_4"`), or an object of class `"regdata"` (as defined in package **lmomRFA**).

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## See Also

`lmrd`, `points`, `lines`.

## Examples

```
# Plot L-moment ratio diagram of Wind from the airquality data set
data(airquality)
lmrd(samlmu(airquality$Wind), xlim=c(-0.2, 0.2))
# Sample L-moments of each month's data
( lmom.monthly <- with(airquality,
  t(sapply(5:9, function(mo) samlmu(Wind[Month==mo])))) )
# Add the monthly values to the plot
lmrdpoints(lmom.monthly, pch=19, col="blue")


# Draw an L-moment ratio diagram and add a line for the
# Weibull distribution
lmrd(xaxs="i", yaxs="i", las=1)
weimom <- sapply( seq(0, 0.9, by=0.01),
  function(tau3) lmrwei(pelwei(c(0,1,tau3)), nmom=4) )
lmrdlines(t(weimom), col='darkgreen', lwd=2)
```

---

`lmrp` *L-moments of a general probability distribution*

---

## Description

Computes the $L$-moments or trimmed $L$-moments of a probability distribution given its cumulative distribution function (for function `lmrp`) or quantile function (for function `lmrq`).

## Usage

```
lmrp(pfunc, ..., bounds=c(-Inf,Inf), symm=FALSE, order=1:4,
     ratios=TRUE, trim=0, acc=1e-6, subdiv=100, verbose=FALSE)

lmrq(qfunc, ..., symm=FALSE, order=1:4, ratios=TRUE, trim=0,
     acc=1e-6, subdiv=100, verbose=FALSE)
```

## Arguments

| | |
|---|---|
| `pfunc` | Cumulative distribution function. |
| `qfunc` | Quantile function. |
| `...` | Arguments to `pfunc` or `qfunc`. |
| `bounds` | Either a vector of length 2, containing the lower and upper bounds of the distribution, or a function that calculates these bounds given the distribution parameters as inputs. |
| `symm` | For `lmrq`, a logical value indicating whether the distribution is symmetric about its median. |

For lmrp, either the logical value FALSE to indicate that the distribution is not symmetric, or a numeric value to indicate that the distribution is symmetric and that the specified value is the center of symmetry.

If the distribution is symmetric, odd-order $L$-moments are exactly zero and the symmetry is used to slightly speed up the computation of even-order $L$-moments.

| | |
|---|---|
| order | Orders of the $L$-moments and $L$-moment ratios to be computed. |
| ratios | Logical. If FALSE, $L$-moments are computed; if TRUE (the default), $L$-moment ratios are computed. |
| trim | Degree of trimming. If a single value, symmetric trimming of the specified degree will be used. If a vector of length 2, the two values indicate the degrees of trimming at the lower and upper ends of the "conceptual sample" (Elamir and Seheult, 2003) of order statistics that is used to define the trimmed $L$-moments. |
| acc | Requested accuracy. The function will try to achieve this level of accuracy, as relative error for $L$-moments and absolute error for $L$-moment ratios. |
| subdiv | Maximum number of subintervals used in numerical integration. |
| verbose | Logical. If FALSE, only the values of the $L$-moments and $L$-moment ratios are returned. If TRUE, more details of the numerical integration are returned: see "Value" section below. |

## Details

Computations use expressions in Hosking (2007): eq. (7) for lmrp, eq. (5) for lmrq. Integrals in those expressions are computed by numerical integration.

## Value

If verbose=FALSE and ratios=FALSE, a numeric vector containing the $L$-moments.

If verbose=FALSE and ratios=TRUE, a numeric vector containing the $L$-moments (of orders 1 and 2) and $L$-moment ratios (of orders 3 and higher).

If verbose=TRUE, a data frame with columns as follows:

| | |
|---|---|
| value | $L$-moments (if ratios=FALSE), or $L$-moments and $L$-moment ratios (if ratios=TRUE). |
| abs.error | Estimate of the absolute error in the computed value. |
| message | "OK" or a character string giving the error message resulting from the numerical integration. |

## Arguments of cumulative distribution functions and quantile functions

pfunc and qfunc can be either the standard R form of cumulative distribution function or quantile function (i.e. for a distribution with $r$ parameters, the first argument is the variate $x$ or the probability $p$ and the next $r$ arguments are the parameters of the distribution) or the cdf... or qua... forms used throughout the **lmom** package (i.e. the first argument is the variate $x$ or probability $p$ and the second argument is a vector containing the parameter values). Even for the R form, however, starting values for the parameters are supplied as a vector start.

If bounds is a function, its arguments must match the distribution parameter arguments of pfunc: either a single vector, or a separate argument for each parameter.

## Warning

Arguments `bounds`, `symm`, `order`, `ratios`, `trim`, `acc`, `subdiv`, and `verbose` cannot be abbreviated and must be specified by their full names (if abbreviated, the names would be matched to the arguments of `pfunc` or `qfunc`).

## Note

In package **lmom** versions 1.6 and earlier, the "Details" section stated that "Integrals in those expressions are computed by numerical integration, using the R function `integrate`". As of version 2.0, numerical integration uses an internal function that directly calls (slightly modified versions of) Fortran routines in QUADPACK (Piessens et al. 1983). R's own `integrate` function uses C code "based on" the QUADPACK routines, but in R versions 2.12.0 through 3.0.1 did not in every case reproduce the results that would have been obtained with the Fortran code (this is R bug PR#15219).

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## References

Elamir, E. A. H., and Seheult, A. H. (2003). Trimmed L-moments. *Computational Statistics and Data Analysis*, **43**, 299-314.

Hosking, J. R. M. (2007). Some theory and practical uses of trimmed L-moments. *Journal of Statistical Planning and Inference*, **137**, 3024-3039.

Piessens, R., deDoncker-Kapenga, E., Uberhuber, C., and Kahaner, D. (1983). *Quadpack: a Subroutine Package for Automatic Integration*. Springer Verlag.

## See Also

`lmrexp` to compute (untrimmed) $L$-moments of specific distributions.

`samlmu` to compute (trimmed or untrimmed) $L$-moments of a data sample.

`pelp` and `pelexp`, to compute the parameters of a distribution given its (trimmed or untrimmed) $L$-moments.

## Examples

```
## Generalized extreme-value (GEV) distribution
## - three ways to get its L-moments
lmrp(cdfgev, c(2,3,-0.2))
lmrq(quagev, c(2,3,-0.2))
lmrgev(c(2,3,-0.2), nmom=4)

## GEV bounds specified as a vector
lmrp(cdfgev, c(2,3,-0.2), bounds=c(-13,Inf))

## GEV bounds specified as a function -- single vector of parameters
gevbounds <- function(para) {
  k <- para[3]
  b <- para[1]+para[2]/k
```

```
  c(ifelse(k<0, b, -Inf), ifelse(k>0, b, Inf))
}
lmrp(cdfgev, c(2,3,-0.2), bounds=gevbounds)

## GEV bounds specified as a function -- separate parameters
pgev <- function(x, xi, alpha, k)
  pmin(1, pmax(0, exp(-((1-k*(x-xi)/alpha)^(1/k)))))
pgevbounds <- function(xi,alpha,k) {
  b <- xi+alpha/k
  c(ifelse(k<0, b, -Inf), ifelse(k>0, b, Inf))
}
lmrp(pgev, xi=2, alpha=3, k=-0.2, bounds=pgevbounds)

## Normal distribution
lmrp(pnorm)
lmrp(pnorm, symm=0)
lmrp(pnorm, mean=2, sd=3, symm=2)
# For comparison, the exact values
lmrnor(c(2,3), nmom=4)

# Many L-moment ratios of the exponential distribution
# This may warn that "the integral is probably divergent"
lmrq(qexp, order=3:20)

# ... nonetheless the computed values seem accurate:
# compare with the exact values, tau_r = 2/(r*(r-1)):
cbind(exact=2/(3:20)/(2:19), lmrq(qexp, order=3:20, verbose=TRUE))

# Of course, sometimes the integral really is divergent
## Not run:
lmrq(function(p) (1-p)^(-1.5))

## End(Not run)

# And sometimes the integral is divergent but that's not what
# the warning says (at least on the author's system)
lmrp(pcauchy)

# Trimmed L-moments for Cauchy distribution are finite
lmrp(pcauchy, symm=0, trim=1)

# Works for discrete distributions too, but often requires
# a larger-than-default value of 'subdiv'
lmrp(ppois, lambda=5, subdiv=1000)
```

---

## pel-functions — *Parameter estimation for specific distributions by the method of L-moments*

### Description

Computes the parameters of a probability distribution as a function of the $L$-moments. The following distributions are recognized:

| | |
|---|---|
| pelexp | exponential |
| pelgam | gamma |
| pelgev | generalized extreme-value |
| pelglo | generalized logistic |
| pelgpa | generalized Pareto |
| pelgno | generalized normal |
| pelgum | Gumbel (extreme-value type I) |
| pelkap | kappa |
| pelln3 | three-parameter lognormal |
| pelnor | normal |
| pelpe3 | Pearson type III |
| pelwak | Wakeby |
| pelwei | Weibull |

### Usage

```
pelexp(lmom)
pelgam(lmom)
pelgev(lmom)
pelglo(lmom)
pelgno(lmom)
pelgpa(lmom, bound = NULL)
pelgum(lmom)
pelkap(lmom)
pelln3(lmom, bound = NULL)
pelnor(lmom)
pelpe3(lmom)
pelwak(lmom, bound = NULL, verbose = FALSE)
pelwei(lmom, bound = NULL)
```

### Arguments

| | |
|---|---|
| lmom | Numeric vector containing the $L$-moments of the distribution or of a data sample. |
| bound | Lower bound of the distribution. If NULL (the default), the lower bound will be estimated along with the other parameters. |
| verbose | Logical: whether to print a message when not all parameters of the distribution can be computed. |

### Details

Numerical methods and accuracy are as described in Hosking (1996, pp. 10–11). Exception: if pelwak is unable to fit a Wakeby distribution using all 5 $L$-moments, it instead fits a generalized

Pareto distribution to the first 3 $L$-moments. (The corresponding routine in the LMOMENTS Fortran package would attempt to fit a Wakeby distribution with lower bound zero.)

## Value

A numeric vector containing the parameters of the distribution.

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## References

Hosking, J. R. M. (1996). Fortran routines for use with the method of $L$-moments, Version 3. Research Report RC20525, IBM Research Division, Yorktown Heights, N.Y.

## See Also

`pelp` for parameter estimation of a general distribution specified by its cumulative distribution function or quantile function.

`lmrexp`, etc., to compute the $L$-moments of a distribution given its parameters.

For individual distributions, see their cumulative distribution functions:

| | |
|---|---|
| `cdfexp` | exponential |
| `cdfgam` | gamma |
| `cdfgev` | generalized extreme-value |
| `cdfglo` | generalized logistic |
| `cdfgpa` | generalized Pareto |
| `cdfgno` | generalized normal |
| `cdfgum` | Gumbel (extreme-value type I) |
| `cdfkap` | kappa |
| `cdfln3` | three-parameter lognormal |
| `cdfnor` | normal |
| `cdfpe3` | Pearson type III |
| `cdfwak` | Wakeby |
| `cdfwei` | Weibull |

## Examples

```
# Sample L-moments of Ozone from the airquality data
data(airquality)
lmom <- samlmu(airquality$Ozone)

# Fit a GEV distribution
pelgev(lmom)
```

pelp *Parameter estimation for a general distribution by the method of L-moments*

## Description

Computes the parameters of a probability distribution as a function of the $L$-moments or trimmed $L$-moments.

## Usage

```
pelp(lmom, pfunc, start, bounds = c(-Inf, Inf),
     type = c("n", "s", "ls", "lss"),
     ratios = NULL, trim = NULL, method = "nlm", acc = 1e-5,
     subdiv = 100, ...)

pelq(lmom, qfunc, start, type = c("n", "s", "ls", "lss"),
     ratios = NULL, trim = NULL, method = "nlm", acc = 1e-5,
     subdiv = 100, ...)
```

## Arguments

`lmom` Numeric vector containing the $L$-moments of the distribution or of a data sample.

`pfunc` Cumulative distribution function of the distribution.

`qfunc` Quantile function of the distribution.

`start` Vector of starting values for the parameters.

`bounds` Either a vector of length 2, containing the lower and upper bounds of the distribution, or a function that calculates these bounds given the distribution parameters as inputs.

`type` Type of distribution, i.e. how it is parametrized. Must be one of the following:

- `"ls"` The distribution has a location parameter and a scale parameter.
- `"lss"` The distribution has a location parameter and a scale parameter, and is symmetric about its median.
- `"s"` The distribution has a scale parameter but not a location parameter.
- `"n"` The distribution has neither a location parameter nor a scale parameter.

For more details, see the “Distribution type” section below.

`ratios` Logical or `NULL`. If `FALSE`, `lmom` should contain $L$-moments; if `TRUE`, `lmom` should contain $L$-moment ratios. If `NULL` and `lmom` has names, the contents of `lmom` will be inferred from these names - see section “Inferring ‘ratios’ and ‘trim’” below. The default value (if `ratios` is `NULL` and `lmom` has no names) is `TRUE`.

`trim` The degree of trimming corresponding to the $L$-moments in `lmom`. Can be a single value or a vector length 2, as for `samlmu`. Can also be `NULL`: in this case if `lmom` has names, the degree of trimming will be inferred from these names - see section "Inferring 'ratios' and 'trim'" below. The default value (if `trim` is `NULL` and `lmom` has no names) is `0`.

`method` Method used to estimate the shape parameters (i.e. all parameters other than the location and scale parameters, if any). Valid values are `"nlm"` (the default), `"uniroot"` (which is valid only if the distribution has at most one shape parameter), and any of the values of the `method` argument of function `optim`. See the "Details" section below.

`acc` Requested accuracy for the estimated parameters. This will be absolute accuracy for shape parameters, relative accuracy for a scale parameter, and absolute accuracy of the location parameter divided by the scale parameter for a location parameter.

`subdiv` Maximum number of subintervals used in the numerical integration that computes $L$-moments of the distribution. Passed to functions `lmrp` or `lmrq`, which perform this integration.

`...` Further arguments will be passed to the optimization function (`nlm`, `uniroot`, or `optim`).

## Details

For shape parameters, numerical optimization is used to find parameter values for which the population $L$-moments or $L$-moment ratios are equal to the values supplied in `lmom`. Computation of $L$-moments or $L$-moment ratios uses functions `lmrp` (for `pelp`) or `lmrq` (for `pelq`). Numerical optimization uses R functions `nlm` (if `method="nlm"`), `uniroot` (if `method="uniroot"`), or `optim` with the specified method (for the other values of `method`). Function `uniroot` uses one-dimensional root-finding, while functions `nlm` and `optim` try to minimize a criterion function that is the sum of squared differences between the population $L$-moments or $L$-moment ratios and the values supplied in `lmom`. Location and scale parameters are then estimated noniteratively. In all cases, the calculation of population $L$-moments and $L$-moment ratios is performed by function `lmrp` or `lmrq` (when using `pelp` or `pelq` respectively).

This approach is very crude. Nonetheless, it is often effective in practice. As in all numerical optimizations, success may depend on the way that the distribution is parametrized and on the particular choice of starting values for the parameters.

## Value

A list with components:

`para` Numeric vector containing the estimated parameters of the distribution.

`code` An integer indicating the result of the numerical optimization used to estimate the shape parameters. It is `0` if there are no shape parameters. In general, values `1` and `2` indicate successful convergence of the iterative procedure, a value of `3` indicates that the iteration may not have converged, and values of `4` or more indicate that the iteration did not converge. Specifically, `code` is:

For method `"nlm"`, the `code` component of the return value from `nlm`.

For method "uniroot", 1 if the estimated precision of the shape parameter is less than or equal to acc, and 4 otherwise.

For the other methods, the convergence component of the return value from optim.

**Further details of arguments**

The length of lmom should be (at least) the highest order of $L$-moment used in the estimation procedure. For a distribution with $r$ parameters this is $2r - 2$ if type="lss" and $r$ otherwise.

pfunc and qfunc can be either the standard R form of cumulative distribution function or quantile function (i.e. for a distribution with $r$ parameters, the first argument is the variate $x$ or the probability $p$ and the next $r$ arguments are the parameters of the distribution) or the cdf... or qua... forms used throughout the **lmom** package (i.e. the first argument is the variate $x$ or probability $p$ and the second argument is a vector containing the parameter values). Even for the R form, however, starting values for the parameters are supplied as a vector start.

If bounds is a function, its arguments must match the distribution parameter arguments of pfunc: either a single vector, or a separate argument for each parameter.

It is assumed that location and scale parameters come first in the set of parameters of the distribution. Specifically: if type="ls" or type="lss", it is assumed that the first parameter is the location parameter and that the second parameter is the scale parameter; if type="s" it is assumed that the first parameter is the scale parameter.

It is important that the length of start be equal to the number of parameters of the distribution. Starting values for location and scale parameters should be included in start, even though they are not used. If start has the wrong length, it is possible that meaningless results will be returned without any warning being issued.

**Distribution type**

The type argument affects estimation as follows. We assume that location and scale parameters are $\xi$ and $\alpha$ respectively, and that the shape parameters (if there are any) are collectively designated by $\theta$.

If type="ls", then the $L$-moment ratios $\tau_3, \tau_4, \ldots$ depend only on the shape parameters. If there are any shape parameters, they are estimated by equating the sample $L$-moment ratios of orders 3, 4, etc., to the population $L$-moment ratios and solving the resulting equations for the shape parameters (using as many equations as there are shape parameters). The $L$-moment $\lambda_2$ is a multiple of $\alpha$, the multiplier being a function only of $\theta$. $\alpha$ is estimated by dividing the second sample $L$-moment by the multiplier function evaluated at the estimated value of $\theta$. The $L$-moment $\lambda_1$ is $\xi$ plus a function of $\alpha$ and $\theta$. $\xi$ is estimated by subtracting from the first sample $L$-moment the function evaluated at the estimated values of $\alpha$ and $\theta$.

If type="lss", then the $L$-moment ratios of odd order, $\tau_3, \tau_5, \ldots$, are zero and the $L$-moment ratios of even order, $\tau_4, \tau_6, \ldots$, depend only on the shape parameters. If there are any shape parameters, they are estimated by equating the sample $L$-moment ratios of orders 4, 6, etc., to the population $L$-moment ratios and solving the resulting equations for the shape parameters (using as many equations as there are shape parameters). Parameters $\alpha$ and $\xi$ are estimated as in case when type="ls".

If type="s", then the $L$-moments divided by $\lambda_1$, i.e. $\lambda_2/\lambda_1, \lambda_3/\lambda_1, \ldots$, depend only on the shape parameters. If there are any shape parameters, they are estimated by equating the sample

$L$-moments (divided by the first sample $L$-moment) of orders 2, 3, etc., to the corresponding population $L$-moments (divided by the first population $L$-moment) and solving the resulting equations (as many equations as there are shape parameters). The $L$-moment $\lambda_1$ is a multiple of $\alpha$, the multiplier being a function only of $\theta$. $\alpha$ is estimated by dividing the first sample $L$-moment by the multiplier function evaluated at the estimated value of $\theta$.

If `type="n"`, then all parameters are shape parameters. They are estimated by equating the sample $L$-moments of orders 1, 2, etc., to the population $L$-moments and solving the resulting equations for the parameters (using as many equations as there are parameters).

## Inferring 'ratios' and 'trim'

If `ratios` or `trim` is `NULL`, appropriate values will be inferred by inspecting the names of `lmom`. It is assumed that `lmom` was generated by a call to `samlmu`, `lmrp`, or `lmrq`; in this case its names will reflect the values of `ratios` and `trim` used in that call. If in this case `lmom` has no names, default values `ratios=TRUE` and `trim=0` will be used.

This inference is made in order to reduce the need to specify the orders of trimming repetitively. For example, a distribution with quantile function `qfunc` can be fitted to (1,1)-trimmed $L$-moments of data in `x` by

```
lmom <- samlmu(x, trim=1)
fit <- pelq(lmom, qfunc, start=...)
```

There is no need to specify `trim` both in the call to `samlmu` and the call to `pelq`.

## Author(s)

J. R. M. Hosking <jrmhosking@gmail.com>

## See Also

`pelexp` for parameter estimation of specific distributions.

## Examples

```
## Gamma distribution -- rewritten so that its first parameter
## is a scale parameter
my.pgamma <- function(x, scale, shape) pgamma(x, shape=shape, scale=scale)
pelp(c(5,2), my.pgamma, start=c(1,1), bounds=c(0,Inf), type="s")
# We can also do the estimation suppressing our knowledge
# that one parameter is a shape parameter.
pelp(c(5,2), my.pgamma, start=c(1,1), bounds=c(0,Inf), type="n")
rm(my.pgamma)

## Kappa distribution -- has location, scale and 2 shape parameters
# Estimate via pelq
pel.out <- pelq(c(10,5,0.3,0.15), quakap, start=c(0,1,0,0), type="ls")
pel.out
# Check that L-moments of estimated distribution agree with the
# L-moments input to pelq()
lmrkap(pel.out$para)
```

```
# Compare with the distribution-specific routine pelkap
pelkap(c(10,5,0.3,0.15))
rm(pel.out)

# Similar results -- what's the advantage of the specific routine?
system.time(pelq(c(10,5,0.3,0.15), quakap, start=c(0,1,0,0), type="ls"))
system.time(pelkap(c(10,5,0.3,0.15)))

# Caution -- pelq() will not check that estimates are reasonable
lmom <- c(10,5,0.2,0.25)
pel.out <- pelq(lmom, quakap, start=c(0,1,0,0), type="ls")
pel.out
lmrkap(pel.out$para) # should be close to lmom, but tau_3 and tau_4 are not
# What happened? pelkap will tell us
try(pelkap(lmom))
rm(lmom, pel.out)

## Inverse Gaussian -- don't have explicit estimators for this
## distribution, but can use numerical methods
#
# CDF of inverse gaussian distribution
pig <- function(x, mu, lambda) {
  temp <- suppressWarnings(sqrt(lambda/x))
  xx <- pnorm(temp*(x/mu-1))+exp(2*lambda/mu+pnorm(temp*(x/mu+1),
              lower.tail=FALSE, log.p=TRUE))
  out <- ifelse(x<=0, 0, xx)
  out
}
# Fit to ozone data
data(airquality)
(lmom<-samlmu(airquality$Ozone))
pel.out <- pelp(lmom[1:2], pig, start=c(10,10), bounds=c(0,Inf))
pel.out
# First four L-moments of fitted distribution,
# for comparison with sample L-moments
lmrp(pig, pel.out$para[1], pel.out$para[2], bounds=c(0,Inf))
rm(pel.out)

## A Student t distribution with location and scale parameters
#
qstu <- function(p, xi, alpha, df) xi + alpha * qt(p, df)
# Estimate parameters.  Distribution is symmetric: use type="lss"
pelq(c(3,5,0,0.2345), qstu, start=c(0,1,10), type="lss")
# Doesn't converge (at least on the author's system) --
# try a different parametrization
qstu2 <- function(p, xi, alpha, shape) xi + alpha * qt(p, 1/shape)
# Now it converges
pelq(c(3,5,0,0.2345), qstu2, start=c(0,1,0.1), type="lss")
# Or try a different optimization method
pelq(c(3,5,0,0.2345), qstu, start=c(0,1,10), type="lss",
    method="uniroot", lower=2, upper=100)

## With trimmed L-moments, we can fit this distribution even when
```

```
## it does not have a finite mean ('df' less than 1)
set.seed(123456)
dat <- qstu(runif(1000), xi=3, alpha=5, df=0.75)
lmom <- samlmu(dat, trim=1)
lmom
# Note that pelq() infers 'trim=1' from the names of 'lmom'
pelq(lmom, qstu, start=c(0,1,10), type="lss",  method="uniroot",
  lower=0.51, upper=100)

rm(qstu, qstu2, dat, lmom)
```

---

samlmu *Sample L-moments*

---

## Description

Computes the "unbiased" sample (trimmed) $L$-moments and $L$-moment ratios of a data vector.

## Usage

```
samlmu(x, nmom=4, sort.data=TRUE, ratios=sort.data, trim=0)
samlmu.s(x, nmom=4, sort.data=TRUE, ratios=sort.data, trim=0)
.samlmu(x, nmom=4)
```

## Arguments

| | |
|---|---|
| x | A numeric vector. |
| nmom | Number of $L$-moments to be found. |
| sort.data | Logical: whether the x vector needs to be sorted. |
| ratios | Logical. If FALSE, $L$-moments are computed; if TRUE (the default), $L$-moment ratios are computed. |
| trim | Degree of trimming. If a single value, symmetric trimming of the specified degree will be used. If a vector of length 2, the two values indicate the degrees of trimming at the lower and upper ends of the "conceptual sample" (Elamir and Seheult, 2003) of order statistics that is used to define the trimmed $L$-moments. |

## Details

samlmu and samlmu.s are functionally identical. samlmu calls a Fortran routine internally, and is usually faster. samlmu.s is written entirely in the S language; it is provided so that users can conveniently see how the calculations are done.

.samlmu is a "bare-bones" version for use in programming. It gives an error if x contains missing values, computes $L$-moment ratios and not $L$-moments, does not give a warning if all the elements of x are equal, and returns its result in an unnamed vector.

Sample $L$-moments are defined in Hosking (1990). Calculations use the algorithm given in Hosking (1996, p.14).

Trimmed sample $L$-moments are defined as in Hosking (2007), eq. (15) (a small extension of Elamir and Seheult (2003), eq. (16)). They are calculated from the untrimmed sample $L$-moments using the recursions of Hosking (2007), eqs. (12)-(13).

## Value

If `ratios` is `TRUE`, a numeric vector containing the $L$-moments and $L$-moment ratios, in the order $\ell_1$, $\ell_2$, $t_3$, $t_4$, etc. If `ratios` is `FALSE`, a numeric vector containing the $L$-moments in the order $\ell_1$, $\ell_2$, $\ell_3$, $\ell_4$, etc.

## Note

The term "trimmed" is used in a different sense from its usual meaning in robust statistics. In particular, the first trimmed $L$-moment is in general not equal to any trimmed mean of the data sample.

## Author(s)

J. R. M. Hosking `<jrmhosking@gmail.com>`

## References

Elamir, E. A. H., and Seheult, A. H. (2003). Trimmed L-moments. *Computational Statistics and Data Analysis*, **43**, 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*, **52**, 105-124.

Hosking, J. R. M. (1996). Fortran routines for use with the method of $L$-moments, Version 3. Research Report RC20525, IBM Research Division, Yorktown Heights, N.Y.

Hosking, J. R. M. (2007). Some theory and practical uses of trimmed L-moments. *Journal of Statistical Planning and Inference*, **137**, 3024-3039.

## Examples

```
data(airquality)
samlmu(airquality$Ozone, 6)

# Trimmed L-moment ratios
samlmu(airquality$Ozone, trim=1)
```

# Index