Package 'FAdist'

October 12, 2022

Type Package

Title Distributions that are Sometimes Used in Hydrology

Version 2.4

Imports stats

Date 2022-03-02

Author Francois Aucoin

Maintainer Thomas Petzoldt <thomas.petzoldt@tu-dresden.de>

Description Probability distributions that are sometimes useful in hydrology.

License GPL-2

URL https://github.com/tpetzoldt/FAdist

Repository CRAN

NeedsCompilation no

Date/Publication 2022-03-03 22:10:02 UTC

R topics documented:

`FAdist-package` *Distributions that are sometimes used in hydrology*

Description

This package contains several distributions that are sometimes useful in hydrology

Author(s)

Francois Aucoin

Maintainer: Thomas Petzoldt <thomas.petzoldt@tu-dresden.de> in agreement with the original author.

`GAMMA3` *Three-Parameter Gamma Distribution (also known as Pearson type III distribution)*

Description

Density, distribution function, quantile function and random generation for the 3-parameter gamma distribution with shape, scale, and threshold (or shift) parameters equal to `shape`, `scale`, and `thres`, respectively.

Usage

```
dgamma3(x,shape=1,scale=1,thres=0,log=FALSE)
pgamma3(q,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
qgamma3(p,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
rgamma3(n,shape=1,scale=1,thres=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`thres`	threshold or shift parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

If Y is a random variable distributed according to a gamma distribution (with shape and scale parameters), then $X = Y{+}m$ has a 3-parameter gamma distribution with the same shape and scale parameters, and with threshold (or shift) parameter m.

Value

`dgamma3` gives the density, `pgamma3` gives the distribution function, `qgamma3` gives the quantile function, and `rgamma3` generates random deviates.

References

Bobee, B. and F. Ashkar (1991). The Gamma Family and Derived Distributions Applied in Hydrology. Water Resources Publications, Littleton, Colo., 217 p.

See Also

`dgamma`, `pgamma`, `qgamma`, `rgamma`

Examples

```
thres <- 10
x <- rgamma3(n=10,shape=2,scale=11,thres=thres)
dgamma3(x,2,11,thres)
dgamma(x-thres,2,1/11)
```

GenPARETO *Generalized Pareto Distribution*

Description

Density, distribution function, quantile function and random generation for the generalized Pareto distribution with shape and scale parameters equal to `shape` and `scale`, respectively.

Usage

```
dgp(x,shape=1,scale=1,log=FALSE)
pgp(q,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
qgp(p,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
rgp(n,shape=1,scale=1)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.

scale	scale parameter.
log,log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

If *X* is a random variable distributed according to a generalized Pareto distribution, it has density f(x) = 1/scale*(1-shape*x/scale)^((1-shape)/shape)

Value

dgp gives the density, pgp gives the distribution function, qgp gives the quantile function, and rgp generates random deviates.

References

Coles, S. (2001) An introduction to statistical modeling of extreme values. Springer

Examples

```
x <- rgp(1000,-.2,10)
hist(x,freq=FALSE,col='gray',border='white')
curve(dgp(x,-.2,10),add=TRUE,col='red4',lwd=2)
```

GEV *Generalized Extreme Value Distribution (for maxima)*

Description

Density, distribution function, quantile function and random generation for the generalized extreme value distribution (for maxima) with shape, scale, and location parameters equal to shape, scale, and location, respectively.

Usage

```
dgev(x,shape=1,scale=1,location=0,log=FALSE)
pgev(q,shape=1,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
qgev(p,shape=1,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
rgev(n,shape=1,scale=1,location=0)
```

Arguments

x,q	vector of quantiles.
p	vector of probabilities.
n	number of observations.
shape	shape parameter.

`scale`	scale parameter.
`location`	location parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

If *X* is a random variable distributed according to a generalized extreme value distribution, it has density
f(x) = 1/scale*(1+shape*((x-location)/scale))^(-1/shape-1)*exp(-(1+shape*((x-location)/scale))^(-1/shape))

Value

`dgev` gives the density, `pgev` gives the distribution function, `qgev` gives the quantile function, and `rgev` generates random deviates.

References

Coles, S. (2001) An introduction to statistical modeling of extreme values. Springer

Examples

```
x <- rgev(1000,-.1,3,100)
hist(x,freq=FALSE,col='gray',border='white')
curve(dgev(x,-.1,3,100),add=TRUE,col='red4',lwd=2)
```

`GUMBEL` *Gumbel Distribution (for maxima)*

Description

Density, distribution function, quantile function and random generation for the Gumbel distribution (for maxima) with scale and location parameters equal to `scale` and `location`, respectively.

Usage

```
dgumbel(x,scale=1,location=0,log=FALSE)
pgumbel(q,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
qgumbel(p,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
rgumbel(n,scale=1,location=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`scale`	scale parameter.
`location`	location parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

If *X* is a random variable distributed according to a Gumbel distribution, it has density
f(x) = 1/scale*exp(-(x-location)/scale-exp(-(x-location)/scale))

Value

`dgumbel` gives the density, `pgumbel` gives the distribution function, `qgumbel` gives the quantile function, and `rgumbel` generates random deviates.

References

Coles, S. (2001) An introduction to statistical modeling of extreme values. Springer

Examples

```
x <- rgumbel(1000,3,100)
hist(x,freq=FALSE,col='gray',border='white')
curve(dgumbel(x,3,100),add=TRUE,col='red4',lwd=2)
```

KAPPA *Kappa Distribution*

Description

Density, distribution function, quantile function and random generation for the kappa distribution with shape and scale parameters equal to `shape` and `scale`, respectively.

Usage

```
dkappa(x,shape=1,scale=1,log=FALSE)
pkappa(q,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
qkappa(p,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
rkappa(n,shape=1,scale=1)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

If *X* is a random variable distributed according to a kappa distribution, it has density
f(x) = shape/scale*(shape+(x/scale)^shape)^(-(shape+1)/shape)

Value

`dkappa` gives the density, `pkappa` gives the distribution function, `qkappa` gives the quantile function, and `rkappa` generates random deviates.

Examples

```
x <- rkappa(1000,12,10)
hist(x,freq=FALSE,col='gray',border='white')
curve(dkappa(x,12,10),add=TRUE,col='red4',lwd=2)
```

`KAPPA4` *Four-Parameter Kappa Distribution*

Description

Density, distribution function, quantile function and random generation for the four-parameter kappa distribution with shape1, shape2, scale, and location parameters equal to `shape1`, `shape2`, `scale`, and `location`, respectively.

Usage

```
dkappa4(x,shape1,shape2,scale=1,location=0,log=FALSE)
pkappa4(q,shape1,shape2,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
qkappa4(p,shape1,shape2,scale=1,location=0,lower.tail=TRUE,log.p=FALSE)
rkappa4(n,shape1,shape2,scale=1,location=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape1`	shape parameter.
`shape2`	shape parameter.
`scale`	scale parameter.
`location`	location parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*,otherwise, *P[X > x]*.

Details

See References

Value

`dkappa4` gives the density, `pkappa4` gives the distribution function, `qkappa4` gives the quantile function, and `rkappa4` generates random deviates.

References

Hosking, J.R.M. (1994). The four-parameter kappa distribution. IBM Journal of Research and Development, 38(3), 251-258.

Examples

```
x <- rkappa4(1000,.1,.2,12,110)
hist(x,freq=FALSE,col='gray',border='white')
curve(dkappa4(x,.1,.2,12,110),add=TRUE,col='red4',lwd=2)
```

`LGAMMA3` *Log-Pearson Type III Distribution*

Description

Density, distribution function, quantile function and random generation for the log-Pearson type III distribution with shape1, shape2, and scale parameters equal to `shape`, `scale`, and `thres`, respectively.

Usage

```
dlgamma3(x,shape=1,scale=1,thres=1,log=FALSE)
plgamma3(q,shape=1,scale=1,thres=1,lower.tail=TRUE,log.p=FALSE)
qlgamma3(p,shape=1,scale=1,thres=1,lower.tail=TRUE,log.p=FALSE)
rlgamma3(n,shape=1,scale=1,thres=1)
```

Arguments

x,q	vector of quantiles.
p	vector of probabilities.
n	number of observations.
shape	shape1 parameter.
scale	shape2 parameter.
thres	scale parameter.
log,log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X <= x]$,otherwise, $P[X > x]$.

Details

If Y is a random variable distributed according to a gamma distribution (with shape and scale parameters), then $X = exp(Y+m)$ has a log-Pearson type III distribution with shape1 and shape2 parameters corresponding to the shape and 1/scale parameteres of Y, and with scale parameter m.

Value

dlgamma3 gives the density, plgamma3 gives the distribution function, qlgamma3 gives the quantile function, and rlgamma3 generates random deviates.

References

BOBEE, B. and F. ASHKAR (1991). The Gamma Family and Derived Distributions Applied in Hydrology. Water Resources Publications, Littleton, Colo., 217 p.

See Also

dgamma, pgamma, qgamma, rgamma, dgamma3, pgamma3, qgamma3, rgamma3

Examples

```
thres <- 10
x <- rlgamma3(n=10,shape=2,scale=11,thres=thres)
dlgamma3(x,2,11,thres)
dgamma3(log(x),2,1/11,thres)/x
dgamma(log(x)-thres,2,11)/x
```

LLOGIS *Log-Logistic Distribution*

Description

Density, distribution function, quantile function and random generation for the log-logistic distribution with shape and scale parameters equal to `shape` and `scale`, respectively.

Usage

```
dllog(x,shape=1,scale=1,log=FALSE)
pllog(q,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
qllog(p,shape=1,scale=1,lower.tail=TRUE,log.p=FALSE)
rllog(n,shape=1,scale=1)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$,otherwise, $P[X > x]$.

Details

If Y is a random variable distributed according to a logistic distribution (with location and scale parameters), then $X = exp(Y)$ has a log-logistic distribution with shape and scale parameters corresponding to the scale and location parameteres of Y, respectively.

Value

`dllog` gives the density, `pllog` gives the distribution function, `qllog` gives the quantile function, and `rllog` generates random deviates.

See Also

`dlogis`, `plogis`, `qlogis`, `rlogis`

Examples

```
x <- rllog(10,1,0)
dllog(x,1,0)
dlogis(log(x),0,1)/x
```

LLOGIS3 *Three-Parameter Log-Logistic Distribution*

Description

Density, distribution function, quantile function and random generation for the 3-parameter log-logistic distribution with shape, scale, and threshold (or shift) parameters equal to `shape`, `scale`, and `thres`, respectively.

Usage

```
dllog3(x,shape=1,scale=1,thres=0,log=FALSE)
pllog3(q,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
qllog3(p,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
rllog3(n,shape=1,scale=1,thres=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`thres`	threshold (or shift) parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$,otherwise, $P[X > x]$.

Details

If Y is a random variable distributed according to a logistic distribution (with location and scale parameters), then $X = exp(Y)+m$ has a 3-parameter log-logistic distribution with shape and scale parameters corresponding to the scale and location parameteres of Y, respectively; and threshold parameter m.

Value

`dllog3` gives the density, `pllog3` gives the distribution function, `qllog3` gives the quantile function, and `rllog3` generates random deviates.

See Also

`dlogis`, `plogis`, `qlogis`, `rlogis`, `dllog`, `pllog`, `qllog`, `rllog`

Examples

```
m <- 100
x <- rllog3(10,1,0,m)
dllog3(x,1,0,m)
dllog(x-m,1,0)
dlogis(log(x-m),0,1)/(x-m)
```

LNORM3 *Three-Parameter Lognormal Distribution*

Description

Density, distribution function, quantile function and random generation for the 3-parameter lognormal distribution with shape, scale, and threshold (or shift) parameters equal to `shape`, `scale`, and `thres`, respectively.

Usage

```
dlnorm3(x,shape=1,scale=1,thres=0,log=FALSE)
plnorm3(q,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
qlnorm3(p,shape=1,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
rlnorm3(n,shape=1,scale=1,thres=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`thres`	threshold (or shift) parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$,otherwise, $P[X > x]$.

Details

If Y is a random variable distributed according to a normal distribution (with location(mean) and scale(standard deviation) parameters), then $X = exp(Y)+m$ has a 3-parameter lognormal distribution with shape and scale parameters corresponding to the scale and location parameteres of Y, respectively; and threshold parameter m.

Value

`dlnorm3` gives the density, `plnorm3` gives the distribution function, `qlnorm3` gives the quantile function, and `rlnorm3` generates random deviates.

See Also

dnorm, pnorm, qnorm, rnorm, dlnorm, plnorm, qlnorm, rlnorm

Examples

```
m <- 100
x <- rlnorm3(10,1,0,m)
dlnorm3(x,1,0,m)
dlnorm(x-m,0,1)
dnorm(log(x-m),0,1)/(x-m)
```

WEIBULL3 *Three-Parameter Weibull Distribution*

Description

Density, distribution function, quantile function and random generation for the 3-parameter Weibull distribution with shape, scale, and threshold (or shift) parameters equal to `shape`, `scale`, and `thres`, respectively.

Usage

```
dweibull3(x,shape,scale=1,thres=0,log=FALSE)
pweibull3(q,shape,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
qweibull3(p,shape,scale=1,thres=0,lower.tail=TRUE,log.p=FALSE)
rweibull3(n,shape,scale=1,thres=0)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations.
`shape`	shape parameter.
`scale`	scale parameter.
`thres`	threshold (or shift) parameter.
`log,log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$,otherwise, $P[X > x]$.

Details

If Y is a random variable distributed according to a Weibull distribution (with shape and scale parameters), then $X = Y+m$ has a 3-parameter Weibull distribution with shape and scale parameters corresponding to the shape and scale parameteres of Y, respectively; and threshold parameter m.

Value

dweibull3 gives the density, pweibull3 gives the distribution function, qweibull3 gives the quantile function, and rweibull3 generates random deviates.

See Also

dweibull, pweibull, qweibull, rweibull

Examples

```
m <- 100
x <- rweibull3(10,3,1,m)
dweibull3(x,3,1,m)
dweibull(x-m,3,1)
```

Index