Package 'trdist'

July 21, 2024

Type Package

Title Univariate Proability Distributions with Truncation

Version 1.0.1

Date 2024-07-17

Author Jared Studyvin [aut, cre]

Maintainer Jared Studyvin <studyvinstat@gmail.com>

Depends R (>= 3.6.0)

Description Truncation of univariate probability distributions. The probability distribution can come from other packages so long as the function names follow the standard d, p, q, r naming format. Also other univariate probability distributions are included.

License GNU General Public License

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2024-07-21 09:00:02 UTC

Contents

getDistributionFunction *Get Distribution Functions*

Description

Determines if the distribution functions are available. This is intended for internal use only.

Usage

```
getDistributionFunction(type, distr, ...)
```

Arguments

type	Character, typically either 'r', 'q', 'p', or 'd'.
distr	Character, typically something like 'norm', 'gamma', etc.
...	Currently ignored.

Details

It is determined that `paste0(type, dist)` is a function and returns that function. The nature of the returned function is not verified.

Value

Function, the first function in the search path that matches the name `paste0(type, dist)`.

Examples

```
fun <- getDistributionFunction(type="q",distr="norm")
```

LogLogistic *Log-Logistic Distribution*

Description

The probability density function, cumulative density function, inverse cumulative density function, random generation for the log logistic distribution.

Usage

```
dllog(x, shape = 1, scale = 1, log = FALSE, ...)

llogSummaryStats(shape, scale, ...)

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	Vector of quantiles.
shape	Shape parameter.
scale	Scale parameter.
log	Logical; if TRUE, log densities are returned.
...	Currently ignored.
q	Vector of quantiles.
lower.tail	Logical; if TRUE (default), probabilities are P(X <= x) otherwise, P(X > x).
log.p	Logical; if TRUE, probabilities p are given as log(p).
p	Vector of probabilities.
n	Number of observations. If length(n) > 1, the length is taken to be the number required.

Details

If X is a random variable distributed according to a logistic distribution, then Y = exp(X) has a log-logistic distribution.

The log-logistic distribution with parameters shape = a and scale = s has density

$$f(x) = \frac{(\frac{1}{a*exp(s))})(\frac{x}{\exp s})^{\frac{1}{a}-1}}{(1+(\frac{x}{\exp s})^{1/a})^2}$$

for x >= 0, a > 1, and s > 0.

The median is exp(s), mean is

$$\frac{a\pi * exp(s)}{sin(a*\pi)}$$

for 1/a > 1. The variance is

$$(exp(s))^2(\frac{2*\pi*a}{(sin(2*pi*a))} - \frac{(a*\pi)^2}{(sin^2(a*\pi))})$$

for 1/a > 2. The mode is

$$exp(s)(\frac{(1/a)-1}{(1/a)+1})^a$$

for 1/a > 1 otherwise it is zero.

Value

`dllog` returns vector of the densities.

`llogSummaryStats` returns a data frame of summary statistics.

`pllog` returns a vector of probabilities.

`qllog` returns a vector of quantiles.

`rllog` returns a vector of random log-logistic variates.

See Also

`Logistic`

Examples

```
y <- rllog(5,shape=1,scale=1/3)
dllog(x=y,shape=1,scale=1/3)
dlogis(x=log(y),location=1/3,scale=1)/y

pllog(q=y,shape=1,scale=1/3)
qllog(p=seq(0,1,by=.25),shape=1,scale=1/3)
```

---

trdist *Univariate Proability Distributions with Truncation*

Description

Truncation of univariate probability distributions. The probability distribution can come from other packages so long as the function names follow the standard d, p, q, r naming format. Also other univariate probability distributions are included.

Author(s)

Maintainer: Jared Studyvin <studyvinstat@gmail.com>

`truncatedDistribution` *Truncated Distributions*

Description

Truncated probability density function, truncated cumulative density function, inverse truncated cumulative density function, and random variates from a truncated distribution.

Usage

```
dtrunc(x, distr, ..., low = -Inf, high = Inf, log = FALSE)

ptrunc(q, distr, ..., low = -Inf, high = Inf, lower.tail = TRUE, log.p = FALSE)

qtrunc(p, distr, ..., low = -Inf, high = Inf, lower.tail = TRUE, log.p = FALSE)

rtrunc(n, distr, ..., low = -Inf, high = Inf)
```

Arguments

`x`	Vector of quantiles.
`distr`	Character value specifying the desired probability distribution.
`...`	Additional arguments passed to the non-truncated distribution functions.
`low`	Numeric value specifying the lower truncation bound.
`high`	Numeric value specifying the upper truncation bound.
`log`	Logical; if TRUE, log densities are returned.
`q`	Vector of quantiles.
`lower.tail`	Logical; if TRUE (default), probabilities are P(X <= x) otherwise, P(X > x).
`log.p`	Currently ignored.
`p`	Vector of probabilities.
`n`	A positive integer specifying the desired number of random variates.

Details

The non truncated distribution functions are assumed to be available. For example if the normal distribution is desired then used `distr='norm'`, the functions then look for 'qnorm', 'pnorm', etc.

The truncation interval is (low, high], which only matters for discrete distribution.

The random variates are produced using the direct method (see Casella and Berger 2002).

Value

`dtrunc` returns a vector of densities.

`ptrunc` returns a vector of probabilities.

`qtrunc` returns a vector of quantiles.

`rtrunc` returns a vector of random variates.

References

G. Casella and R. L. Berger. Statistical inference. Vol. 2. Duxbury Pacific Grove, CA, 2002.

Examples

```
## dtrunc
# not truncted
dnorm(5,mean=5)
dtrunc(x=5,distr='norm',mean=5)
# truncated
dtrunc(x=5,distr='norm',mean=5,low=4, high=5.5)



## ptrunc
#not truncated
pgamma(2, shape=3, rate=2)
ptrunc(2, distr = 'gamma', shape=3, rate=2)
# truncated
ptrunc(2, distr = 'gamma', shape=3, rate=2, low=1, high=5)

## upper tail
# not truncated
pgamma(2, shape=3, rate=2,lower.tail=FALSE)
ptrunc(2, distr='gamma', shape=3, rate=2, lower.tail=FALSE)
# truncated
ptrunc(2, distr='gamma', shape=3, rate=2, low=1, high=5, lower.tail=FALSE)



## qtrunc
#not truncated
qnorm(p=.975)
qtrunc(p=.975,distr='norm')
# truncted
qtrunc(p=.975,distr='norm', low=0, high=1)

## upper tail
# not truncted
qnorm(p=.975,lower.tail=FALSE)
qtrunc(p=.975, distr='norm', lower.tail=FALSE)
# truncated
qtrunc(p=.975, distr='norm', low=0, high=1, lower.tail=FALSE)

## rtrunc
rtrunc(n=5, distr = 'gamma', shape=3, rate=2, low=2, high=5)
```

Index