Package 'triangulr'

October 14, 2022

Title High-Performance Triangular Distribution Functions

Version 1.2.1

Description A collection of high-performance functions for the triangular distribution that consists of the probability density function, cumulative distribution function, quantile function, random variate generator, moment generating function, characteristic function, and expected shortfall function. References: Samuel Kotz, Johan Ren Van Dorp (2004) <doi:10.1142/5720> and Acerbi, Carlo and Tasche, Dirk. (2002) <doi:10.1111/1468-0300.00091>.

License MIT + file LICENSE

URL https://github.com/irkaal/triangulr/
https://irkaal.github.io/triangulr/

BugReports https://github.com/irkaal/triangulr/issues

Depends R (>= 3.3)

Encoding UTF-8

ByteCompile true

RoxygenNote 7.1.1

Collate 'check.R' 'cpp11.R' 'error.R' 'Triangular.R'

LinkingTo cpp11

SystemRequirements C++11

Imports rlang (>= 0.4.11), vctrs (>= 0.3.8)

Suggests testthat (>= 3.0.2)

NeedsCompilation yes

Author Alvin Nursalim [aut, cre]

Maintainer Alvin Nursalim <irkaalv@gmail.com>

Repository CRAN

Date/Publication 2021-05-27 04:40:03 UTC

R topics documented:

`Triangular` *The Triangular Distribution*

Description

These functions provide information about the triangular distribution on the interval from `min` to `max` with mode equal to `mode`. `dtri` gives the density function, `estri` gives the expected shortfall, `mgtri` gives the moment generating function, `ptri` gives the distribution function, `qtri` gives the quantile function, and `rtri` gives the random variate generator.

Usage

```
dtri(x, min = 0, max = 1, mode = 0.5, log = FALSE)

ptri(q, min = 0, max = 1, mode = 0.5, lower_tail = TRUE, log_p = FALSE)

qtri(p, min = 0, max = 1, mode = 0.5, lower_tail = TRUE, log_p = FALSE)

rtri(n, min = 0, max = 1, mode = 0.5)

mgtri(t, min = 0, max = 1, mode = 0.5)

estri(p, min = 0, max = 1, mode = 0.5, lower_tail = TRUE, log_p = FALSE)
```

Arguments

`x, q`	Vector of quantiles.
`min`	Lower limit of the distribution. Must have `min` $<$ `max`.
`max`	Upper limit of the distribution. Must have `max` $>$ `min`.
`mode`	The mode of the distribution. Must have `mode` $\geq$ `min` and `mode` $\leq$ `max`.
`log, log_p`	Logical; if `TRUE`, probabilities `p` are given as `log(p)`.
`lower_tail`	Logical; if `TRUE` (default), probabilities `p` are $P[X \leq x]$, otherwise, $P[X > x]$.
`p`	Vector of probabilities.
`n`	Number of observations. Must have length of one.
`t`	Vector of dummy variables.

Details

If `min`, `max`, or `mode` are not specified they assume the default values of `0`, `1`, and `0.5` respectively.

The triangular distribution has density

$$0$$

for $x < min$ or $x > max$

$$f(x) = \frac{2(x - min)}{(max - min)(mode - min)}$$

for $min \le x < mode$, and

$$f(x) = \frac{2(max - x)}{(max - min)(max - mode)}$$

for $mode < x \le max$.

`rtri` will not generate either of the extreme values unless `max - min` is small compared to `min`, and in particular not for the default arguments.

Value

`dtri` gives the density function, `estri` gives the expected shortfall, `mgtri` gives the moment generating function, `ptri` gives the distribution function, `qtri` gives the quantile function, and `rtri` gives the random variate generator.

The numerical arguments other than `n` with values of size one are recycled to the length of `t` for `mgtri`, the length of `x` for `dtri`, the length of `p` for `estri` and `qtri`, the length of `q` for `ptri`, and `n` for `rtri`. This determines the length of the result.

The logical arguments `log`, `lower_tail`, and `log_p` must be of length one each.

Note

The characteristics of output from pseudo-random number generators (such as precision and periodicity) vary widely. See `.Random.seed` for more information on R's random number generation algorithms.

See Also

`RNG` about random number generation in R.

Distributions for other standard distributions.

Examples

```
# min, max, and mode with lengths equal to the length of x
x <- c(0, 0.5, 1)
d <- dtri(x,
          min = c(0, 0, 0),
          max = c(1, 1, 1),
          mode = c(0.5, 0.5, 0.5))
# min and max will be recycled to the length of x
```

```
rec_d <- dtri(x,
              min = 0,
              max = 1,
              mode = c(0.5, 0.5, 0.5))
all.equal(d, rec_d)

# min, max, and mode with lengths equal to the length of x
n <- 3
set.seed(1)
r <- rtri(n,
          min = c(0, 0, 0),
          max = c(1, 1, 1),
          mode = c(0.5, 0.5, 0.5))
# min and max will be recycled to the length of n
set.seed(1)
rec_r <- rtri(n,
              min = 0,
              max = 1,
              mode = c(0.5, 0.5, 0.5))
all.equal(r, rec_r)

# Log quantiles
x <- c(0, 0.5, 1)
log_d <- dtri(x, log = TRUE)
d <- dtri(x, log = FALSE)
all.equal(log(d), log_d)

# Upper tail probabilities
q <- c(0, 0.5, 1)
upper_p <- ptri(q, lower_tail = FALSE)
p <- ptri(q, lower_tail = TRUE)
all.equal(upper_p, 1 - p)

# Log probabilities
q <- c(0, 0.5, 1)
log_p <- ptri(q, log_p = TRUE)
p <- ptri(q, log_p = FALSE)
all.equal(upper_p, 1 - p)

# The quantile function
p <- c(0, 0.5, 1)
upper_q <- ptri(1 - p, lower_tail = FALSE)
q <- ptri(p, lower_tail = TRUE)
all.equal(upper_q, q)
p <- c(0, 0.5, 1)
log_q <- qtri(log(p), log_p = TRUE)
q <- qtri(p, log_p = FALSE)
all.equal(log_q, q)

# Moment generating function
t <- c(1, 2, 3)
mgtri(t)
```

```
# Expected Shortfall
p <- c(0.1, 0.5, 1)
estri(p)
```

Index