Package: chi (via r-universe)

September 10, 2024

Type Package

Title The Chi Distribution

Version 0.0

URL https://github.com/dkahle/chi

BugReports https://github.com/dkahle/chi/issues

Description Light weight implementation of the standard distribution functions for the chi distribution, wrapping those for the chi-squared distribution in the stats package.

License GPL-2

RoxygenNote 5.0.1

Repository https://dkahle.r-universe.dev

RemoteUrl https://github.com/dkahle/chi

RemoteRef HEAD

RemoteSha ddc32c645ffd4ad8a230ae895a7947c23833bca7

Contents

`chi` *The Chi Distribution*

Description

Density, distribution function, quantile function and random generation for the chi distribution.

Usage

```
dchi(x, df, ncp = 0, log = FALSE)

pchi(q, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)

qchi(p, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)

rchi(n, df, ncp = 0)
```

Arguments

x, q	vector of quantiles.
df	degrees of freedom (non-negative, but can be non-integer).
ncp	non-centrality parameter (non-negative).
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].
p	vector of probabilities.
n	number of observations. If length(n) > 1, the length is taken to be the number required.

Details

The functions (d/p/q/r)chi simply wrap those of the standard (d/p/q/r)chisq R implementation, so look at, say, dchisq for details.

See Also

dchisq; these functions just wrap the (d/p/q/r)chisq functions.

Examples

```
s <- seq(0, 5, .01)
plot(s, dchi(s, 7), type = 'l')

f <- function(x) dchi(x, 7)
q <- 2
integrate(f, 0, q)
(p <- pchi(q, 7))
qchi(p, 7) # = q
mean(rchi(1e5, 7) <= q)


samples <- rchi(1e5, 7)
plot(density(samples))
curve(f, add = TRUE, col = "red")
```

invchi *The Inverse Chi Distribution*

Description

Density, distribution function, quantile function and random generation for the inverse chi distribution.

Usage

```
dinvchi(x, df, ncp = 0, log = FALSE)

pinvchi(q, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)

qinvchi(p, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)

rinvchi(n, df, ncp = 0)
```

Arguments

x, q	vector of quantiles.
df	degrees of freedom (non-negative, but can be non-integer).
ncp	non-centrality parameter (non-negative).
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].
p	vector of probabilities.
n	number of observations. If length(n) > 1, the length is taken to be the number required.

See Also

dchi

Examples

```
s <- seq(0, 2, .01)
plot(s, dinvchi(s, 7), type = 'l')

f <- function(x) dinvchi(x, 7)
q <- .5
integrate(f, 0, q)
(p <- pinvchi(q, 7))
qinvchi(p, 7) # = q
mean(rinvchi(1e5, 7) <= q)


samples <- rinvchi(1e5, 7)
```

```
plot(density(samples))
curve(f, add = TRUE, col = "red")
```

Index