

# Package ‘argus’

September 15, 2023

**Type** Package

**Title** Random Variate Generator for the Argus Distribution

**Version** 0.1.1

**Maintainer** Wolfgang Hormann <hormanngw@yahoo.com>

**Imports** Runuran

**Description** Random variate generation, density, CDF and quantile function for the Argus distribution. Especially, it includes for random variate generation a flexible inversion method that is also fast in the varying parameter case. A Ratio-of-Uniforms method is provided as second alternative.

**License** GPL (>= 2)

**LazyLoad** yes

**NeedsCompilation** yes

**Repository** CRAN

**Author** Wolfgang Hormann [aut, cre],  
Christoph Baumgarten [aut]

**Date/Publication** 2023-09-15 08:02:29 UTC

## R topics documented:

argus-package . . . . .	1
rargus . . . . .	3

<b>Index</b>	<b>5</b>
--------------	----------

---

argus-package	<i>Generator and density for the Argus distribution</i>
---------------	---

---

## Description

Random variate generation, density, CDF and quantile function for the Argus distribution. Especially, it includes for random variate generation a flexible inversion method that is also fast in the varying parameter case. A Ratio-of-Uniforms method is provided as second alternative.

## Details

The Argus distribution has the density (pdf) which is proportional to

$$f(x) = x\sqrt{1-x^2} \exp(-0.5\chi^2(1-x^2)), \text{ for } 0 \leq x \leq 1 \text{ and } \chi > 0$$

Package: argus  
License: GPL 2 or later

Package **argus** provides the routines:

**rargus** generates argus distributed random variates.

**dargus** computes the density of the argus distribution.

**pargus** computes the CDF of the argus distribution.

**qargus** computes the quantile function (ie. the inverse CDF) of the argus distribution.

## Author(s)

Wolfgang Hörmann & Christoph Baumgarten

## References

Christoph Baumgarten: Random Variate Generation by Fast Numerical Inversion in the Varying Parameter Case.

## Examples

```
## Evaluate the pdf (density)
dargus(c(0.1,0.5,0.9), chi=0.3)
## Evaluate the CDF for different chi values
pargus(c(0.1,0.5,0.9), chi=c(0.3,1.3,2.3))
## Evaluate the quantile function
qargus(c(0.1,0.5,0.9), chi=4.5)
## Draw a random sample
rargus(n=10, chi=0.3)
## compare histogram and density
system.time(y<-rargus(1.e5,chi=2.5))
hist(y,breaks=100,freq=FALSE)
lines(x<-seq(0,1,1.e-3),dargus(x,2.5),col=2,lwd=2)
```

---

rargus	<i>The Argus distribution</i>
--------	-------------------------------

---

### Description

Density, distribution function, quantile function and random generation for the Argus distribution with parameter  $\chi$ . `rargus()` uses very efficient random variate generation methods. Inversion or the Ratio of Uniforms method can be selected.

### Usage

```
rargus(n=length(chi) , chi, method = c("inversion","RoU") )
dargus(x, chi, log = FALSE)
pargus(x, chi, lower = TRUE, log.p = FALSE)
qargus(p, chi, lower = TRUE, log.p = FALSE)
```

### Arguments

n	Number of observations
chi	parameter vector $\chi > 0$
method	random variate generation method to be used (details are below)
x	x-value for the density or CDF
log	If TRUE the logarithm of the density is returned.
lower	If FALSE 1-CDF or its inverse is returned
log.p	$\log(p)$ the logarithm of the probability is used or returned
p	probability of the quantile

### Details

The Argus distribution has the density (pdf) which is proportional to

$$f(x) = x\sqrt{1-x^2} \exp(-0.5\chi^2(1-x^2)), \text{ for } 0 \leq x \leq 1 \text{ and } \chi > 0$$

The generators used for `rargus()` are very efficient also for the varying parameter case, i.e. when the vector `chi` has length `n`.

`method = "inversion"`: transforms the uniform variate into an argus variate using a close to exact approximation of the inverse CDF. It is also well suited for the varying parameter case as the theorem that an argus variate can be represented as a transformed truncated Gamma(1.5) variate allows to express the inverse CDF of the argus distribution as a simple transform of the inverse CDF of the Gamma(1.5) distribution. The use of the `pinv.new()` function of the `Runuran` package makes that evaluation very fast. When loading the `argus`-package `pinv.new()` calculates and stores the required tables.

`method = "RoU"`: uses the Ratio of Uniforms method which requires 2 or more uniform variates to generate one argus variate. It is also a bit slower than the inversion method and is added here mainly for the case that the `Runuran`-package is not available.

**Value**

rargus creates a random sample of size n.

dargus gives the density, dargus give the CDF and qargus gives the quantile function. The length of the result for these 3 functions is the maximum of the lengths of the numerical arguments that are recycled to the length of the result. Only the first elements of the logical arguments are used.

**Author(s)**

Wolfgang Hörmann and Christoph Baumgarten

**References**

Christoph Baumgarten: Random Variate Generation by Fast Numerical Inversion in the Varying Parameter Case.

**Examples**

```
library(argus)
# pdf, cdf and quantile function
dargus((0:10)/10,.6)
qargus(0.9,2)
pargus(qargus(0.9,2),2)

y<-rargus(n=1.e5,chi=0.3,method="inversion")
hist(y,breaks=100,main="Argus pdf with chi=0.3",freq=FALSE)
lines(xv<-seq(0,1,0.001),dargus(xv,chi=0.3),lwd=0.2,col=2)

n=1.e5
chiv <- 1. +runif(n)*5 # chi-values in (1,5)
system.time({set.seed(123);y<-rargus(chi=chiv)})
set.seed(123);u<- runif(n)
mean(abs(pargus(y,chiv)-u))

# random variate generation with the inversion method
y<-rargus(n=1.e5,chi=0.3,method="inversion")
hist(y,breaks=100,main="Argus pdf with chi=0.3",freq=FALSE)
lines(xv<-seq(0,1,0.001),dargus(xv,chi=0.3),lwd=0.2,col=2)
# using Ratio of Uniforms
y<-rargus(n=1.e5,chi=3,method="RoU")
hist(y,breaks=100,main="Argus pdf with chi=3",freq=FALSE)
lines(xv<-seq(0,1,0.001),dargus(xv,chi=3),lwd=2,col=2)
# generating for different chi values
y<-rargus(n=100,chi=runif(100)*2,method="RoU")
```

# Index

\* **datagen**

argus-package, 1

rargus, 3

\* **distribution**

argus-package, 1

rargus, 3

\* **package**

argus-package, 1

rargus, 3

argus (argus-package), 1

argus-package, 1

dargus, 2

dargus (rargus), 3

pargus, 2

pargus (rargus), 3

qargus, 2

qargus (rargus), 3

rargus, 2, 3