Package 'occupancy'

October 14, 2022

Type Package

Title Probability Functions for Occupancy Distributions

Version 1.2

Date 2021-06-23

Maintainer Ben O'Neill <ben.oneill@hotmail.com>

Description The classical and extended occupancy distributions occur in cases where balls are randomly allocated
to bins. The PDF, CDF, quantile functions, generation of random variates, and calculating the first four central moments of the distributions are implemented as described in O'Neill (2019) <doi:10.1080/00031305.2019.1699445>.

License MIT + file LICENSE

Encoding UTF-8

Imports matrixStats

Suggests VGAM, ggplot2, gridExtra

RoxygenNote 7.1.1

NeedsCompilation no

Author Ben O'Neill [aut, cre]

Repository CRAN

Date/Publication 2021-06-24 11:00:02 UTC

R topics documented:

dmaxcount *The Maximum-Count Occupancy Distribution*

Description

Density, distribution function, quantile function and random generation for the maximum count occupancy distribution with size and shape parameters.

Usage

```
dmaxcount(x, size, space, prob = 1, log = FALSE)

dmaxcount.all(max.x, max.size, space, prob = 1, log = FALSE)

pmaxcount(q, size, space, prob = 1, log.p = FALSE, lower.tail = TRUE)

qmaxcount(p, size, space, prob = 1, log.p = FALSE, lower.tail = TRUE)

rmaxcount(n, size, space, prob = 1)
```

Arguments

x	vector of quantiles.
size	The size parameter for the maximum-count distribution (number of balls)
space	The space parameter for the maximum-count distribution (number of bins)
prob	The probability parameter for the occupancy distribution (probability of ball occupying its bin)
log	logical; if TRUE, probabilities p are given as log(p).
max.x	A vector of numeric values to be used as arguments for the probability mass function
max.size	The maximum size parameter for the maximum-count distribution (number of balls)
q	vector of quantiles.
log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ 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

`dmaxcount.all` returns the entire PMF.

This function computes probabilities or log-probabilities from the probability mass function of the maximum-count distribution, which is the distribution for the maximum of the counts for the number of balls in a bin in the extended occupancy problem. Details of the algorithm in the classical case can be found in the papers below. The extension to include the probability parameter is done using the binomial mixture representation of the extended occupancy problem.

Value

If all inputs are correctly specified (i.e., parameters are in allowable range) then the output will be a vector of probabilities/log-probabilities up to the maximum argument values

References

Bonetti, M., Corillo, P. and Ogay, A. (2019) Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics.

Rappeport, M,A. (1968) Algorithms and computational procedures for the application of order statistics to queuing problems. PhD thesis, New York University.

Examples

```
x <- rmaxcount(10, 2, 2)
p <- pmaxcount(x, 2, 2)
stopifnot(x == qmaxcount(p, 2, 2))
dmaxcount.all(2,2,2)
```

---

dnegocc *The Negative Occupancy Distribution*

Description

Density, distribution function, quantile function and random generation for the negative occupancy distribution with space and occupancy parameters.

Usage

```
dnegocc(x, space, occupancy, prob = 1, approx = FALSE, log = FALSE)

dnegocc.all(max.x, space, max.occupancy, prob = 1, approx = FALSE, log = FALSE)

pnegocc(
  x,
  space,
  occupancy,
  prob = 1,
  approx = FALSE,
```

```
  log.p = FALSE,
  lower.tail = TRUE
)

qnegocc(
  p,
  space,
  occupancy,
  prob = 1,
  approx = FALSE,
  log.p = FALSE,
  lower.tail = TRUE
)

rnegocc(n, space, occupancy, prob = 1)
```

Arguments

x	vector of quantiles.
space	The space parameter for the negative occupancy distribution (number of bins)
occupancy	The occupancy parameter for the negative occupancy distribution (number of occupied bins)
prob	The probability parameter for the negative occupancy distribution (probability of ball occupying its bin)
approx	A logical value specifying whether to use an approximation for the distribution
log	logical; if TRUE, probabilities p are given as log(p).
max.x	A vector of numeric values to be used as arguments for the mass function
max.occupancy	The maximum occupancy parameter for the negative occupancy distribution (number of occupied bins)
log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ 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

`dnegcount.all` returns the entire PMF.

The computation method uses a recursive algorithm described in the reference.

Value

If all inputs are correctly specified (i.e., parameters are in allowable range and arguments are integers) then the output will be a matrix of probabilities/log-probabilities

References

O'Neill, B. (2021) An examination of the negative-occupancy distribution and the coupon-collector distribution.

Examples

```
x <- rnegocc(10, 2, 2)
p <- pnegocc(x, 2, 2)
qnegocc(0:9/10, 2, 2)
dnegocc.all(5,2,2)
```

---

docc *The Extended Occupancy Distribution*

Description

Density, distribution function, quantile function and random generation for the extended occupancy distribution with size and shape parameters.

Usage

```
docc(x, size, space, prob = 1, approx = FALSE, log = FALSE)

docc.all(max.size, space, prob = 1, approx = FALSE, log = FALSE)

pocc(
  x,
  size,
  space,
  prob = 1,
  approx = FALSE,
  log.p = FALSE,
  lower.tail = TRUE
)

qocc(
  p,
  size,
  space,
  prob = 1,
  approx = FALSE,
  log.p = FALSE,
  lower.tail = TRUE
)

rocc(n, size, space, prob = 1)
```

Arguments

x	vector of quantiles.
size	The size parameter for the occupancy distribution (number of balls)
space	The space parameter for the occupancy distribution (number of bins)
prob	The probability parameter for the occupancy distribution (probability of ball occupying its bin)
approx	A logical value specifying whether to use the normal approximation to the occupancy distribution
log	logical; if TRUE, probabilities p are given as log(p).
max.size	The maximum size parameter for the occupancy distribution (number of balls)
log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ 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

`docc.all` returns the entire PMF.

Value

If all inputs are correctly specified (i.e., parameters are in allowable range and arguments are integers) then the output will be a vector of probabilities/log-probabilities corresponding to the vector argument x

References

O'Neill, B. (2021) Three distributions in the extended occupancy problem.

Examples

```
x <- rocc(10, 2, 2)
p <- pocc(x, 2, 2)
stopifnot(x == qocc(p, 2, 2))
docc.all(2,2)
```

doccgap *The Occupancy-Gap Distribution*

Description

Density, distribution function, quantile function and random generation for the Occupancy-Gap Distribution with size and scale parameters (see note).

Usage

```
doccgap(
  x,
  size,
  space = NULL,
  occupancy = size,
  prob = NULL,
  scale = NULL,
  log = FALSE
)

doccgap.all(
  size,
  space = NULL,
  max.occupancy = size,
  prob = NULL,
  scale = NULL,
  log = FALSE
)

poccgap(
  x,
  size,
  space = NULL,
  occupancy = size,
  prob = NULL,
  scale = NULL,
  log.p = FALSE,
  lower.tail = TRUE
)

qoccgap(
  p,
  size,
  space = NULL,
  occupancy = size,
  prob = NULL,
  scale = NULL,
```

```
  log.p = FALSE,
  lower.tail = TRUE
)

roccgap(n, size, space = NULL, occupancy = size, prob = NULL, scale = NULL)
```

Arguments

x	vector of quantiles.
size	The size parameter for the occupancy-gap distribution (number of balls)
space	The space parameter for the occupancy-gap distribution (number of bins)
occupancy	The occupancy parameter for the occupancy-gap distribution (number of occupied bins)
prob	The probability parameter for the occupancy-gap distribution (probability of ball occupying its bin)
scale	The scale parameter for the occupancy-gap distribution
log	logical; if TRUE, probabilities p are given as log(p).
max.occupancy	The maximum occupancy parameter for the occupancy-gap distribution (number of occupied bins)
log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \leq x]$ 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

`docc.all` returns the entire PMF.

This function computes probabilities or log-probabilities from the mass function of the occupancy-gap distribution. The computation method uses a recursive algorithm from the following paper:

Value

If all inputs are correctly specified (i.e., parameters are in allowable range) then the output will be a matrix of probabilities/log-probabilities

References

O'Neill, B. (forthcoming) An examination of the occupancy-gap distribution.

Note

The distribution is parameterised by a `scale` parameter, but in applied problems in the context of the extended occupancy problem this parameter is a function of `space` and `prob` parameters. The function allows either parameterisation (i.e., the user can either specify the `scale` paramater or both the `space` and `prob` parameters).

Examples

```
x <- roccgap(10, 20, 2, 2, .5)
p <- poccgap(x, 20, 2, 2, .5)
stopifnot(x == qoccgap(p, 20, 2, 2, .5))
doccgap.all(20, 2, 2, .5)
```

`logStirling` *Logarithms of the Stirling numbers of the second kind*

Description

`log.Stirling` returns a matrix of the logarithms of the Stirling numbers of the second kind.

Usage

```
logStirling(n, k, ncp = 0)
```

Arguments

`n`	A vector of non-negative integer values
`k`	A vector of non-negative integer values
`ncp`	Non-centrality parameter (non-negative numeric value)

Details

This function computes a matrix of the logarithms of the Stirling numbers of the second kind. The function allows the user to give a non-centrality parameter for the non-central Stirling numbers.

Value

If all inputs are correctly specified then the output will be a matrix containing the logarithms of the Stirling numbers of the second kind

---

`occupancy` *occupancy: A package for computing with occupancy distributions.*

Description

The occupancy package provides standard mass functions, distribution functions, quantile functions and random generation for various distributions. These distributions occur in cases where balls are randomly allocated to bins.

`sample.ballbin` *Generates simulations from the extended balls-in-bins process*

Description

`sample.ballbin` generates simulated data from the extended balls-in-bins process.

Usage

```
sample.ballbin(n, size, space, prob, alloc.prob = NULL)

## S3 method for class 'ballbin'
print(x, ...)

## S3 method for class 'ballbin'
plot(
  x,
  ...,
  ball.size = NULL,
  ball.color = NULL,
  ball.colour = ball.color,
  max.plots = 30
)

## S3 method for class 'ballbin'
summary(object, ...)

## S3 method for class 'summary.ballbin'
print(x, ...)

## S3 method for class 'summary.ballbin'
plot(x, ..., bar.color = NULL, bar.colour = bar.color)
```

Arguments

`n`	The number of simulations of the process
`size`	The size parameter for the occupancy distribution (number of balls)
`space`	The space pararmeter for the occupancy distribution (number of bins)
`prob`	The probability parameter for the occupancy distribution (probability of ball occupying its bin)
`alloc.prob`	(Optional) A probability vector for the allocation probabilities for the bins
`x, object`	ballbin objects (for generics)
`...`	unused
`ball.size, ball.color, ball.colour, max.plots`	Set the size, color, and number of plots
`bar.color, bar.colour`	plotting arguments

Details

This function generates a simulated set of data from the extended balls-in-bins process. The outcome is an object of class `ballbin` containing the simulations from the process. The output object contains the initial bin-allocation and resulting samples from the process. Calling `summary` on the simulation object creates a new object of class `summary.ballbin` containing summary statistics for each sample, including the bin-counts, effective sample-size, occupancy number, max-count number, and hitting times for each possible occupancy number. Each of these objects has a custom printing and plot methods to give user-friendly output.

Value

If all inputs are correctly specified (i.e., parameters are in allowable range) then the output will be a list of class `ballbin` containing `n` random samples from the process. If you call `summary` on this object the output will be another list of class `summary.ballbin` containing summary statistics for each random sample from the process.

Methods (by generic)

- `print`: prints the sample
- `plot`: plots the sample
- `summary`: summarizes the sample
- `print`: prints the summary
- `plot`: plots the summary

Examples

```
d <- sample.ballbin(12, 10, 4, .4)
print(d)
plot(d)
summary(d)
plot(summary(d))
```

Index