Package 'chyper'

May 8, 2026

Type Package

Title Functions for Conditional Hypergeometric Distributions

Version 0.3.1

Author William Nickols

Maintainer William Nickols <willnickols@college.harvard.edu>

Description An implementation of the probability mass function, cumulative density function, quantile function, random number generator, maximum likelihood estimator, and p-value generator from a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-08-13 09:20:05 UTC

Contents

dchyper — *Probability mass function for conditional hypergeometric distributions*

Description

Calculates the PMF of a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
dchyper(k, s, n, m, verbose = T)
```

Arguments

k	an integer or vector of integers representing the overlap size
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population
m	a vector of integers representing the sample sizes
verbose	T/F should intermediate messages be printed?

Value

The probability of sampling k of the same items in all samples

Examples

```
dchyper(c(3,5), 10, c(12,13,14), c(7,8,9))
```

mleM — *Maximum likelihood estimator for sample size in conditional hypergeometric distributions*

Description

Calculates the MLE of a sample size in a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
mleM(population, k, s, n, m, verbose = T)
```

Arguments

population	the index of the unknown sample size
k	the observed overlaps
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population
m	a vector of integers representing the sample sizes where the value of the unknown sample size should be any integer as a placeholder
verbose	T/F should intermediate messages be printed?

Value

The maximum likelihood estimator of the unknown sample size

Examples

```
mleM(1, c(0,0,1,1,0,2,0), 8, c(12,13,14), c(0,8,9))
```

---

mleN *Maximum likelihood estimator for a unique population size in conditional hypergeometric distributions*

Description

Calculates the MLE of a unique population size in a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
mleN(population, k, s, n, m, verbose = T)
```

Arguments

population	the index of the unique population to estimate
k	the observed overlaps
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population where the value of the unknown population size should be any integer as a placeholder
m	a vector of integers representing the sample sizes
verbose	T/F should intermediate messages be printed?

Value

The maximum likelihood estimator of the unknown unique population size

Examples

```
mleN(1, c(0,0,1,1,0,2,0), 8, c(0,13,14), c(7,8,9))
```

`mleS` *Maximum likelihood estimator for overlap size in conditional hypergeometric distributions*

Description

Calculates the MLE of the overlap size in a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
mleS(k, n, m, verbose = T)
```

Arguments

`k`	the observed overlaps
`n`	a vector of integers representing the sizes of each non-intersecting population
`m`	a vector of integers representing the sample sizes
`verbose`	T/F should intermediate messages be printed?

Value

The maximum likelihood estimator of the intersecting population size

Examples

```
mleS(c(0,0,1,1,0,2,0), c(12,13,14), c(7,8,9))
```

pchyper — *Cumulative density function for conditional hypergeometric distributions*

Description

Calculates the CDF of a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
pchyper(k, s, n, m, verbose = T)
```

Arguments

k	an integer or vector of integers representing the overlap size
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population
m	a vector of integers representing the sample sizes
verbose	T/F should intermediate messages be printed?

Value

The probability of sampling k or less of the same items in all samples

Examples

```
pchyper(c(3,5), 10, c(12,13,14), c(7,8,9))
```

pvalchyper — *P-values from a conditional hypergeometric distribution*

Description

Calculates p-values from a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
pvalchyper(k, s, n, m, tail = "upper", verbose = T)
```

Arguments

k	an integer or vector of integers representing the overlap size
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population
m	a vector of integers representing the sample sizes
tail	whether the p-value should be from the upper or lower tail (options: "upper", "lower")
verbose	T/F should intermediate messages be printed?

Value

The probability of getting the k or more (or less if tail="lower") overlaps by chance from the conditional hypergeometric distribution specified by the parameters

Examples

```
pvalchyper(c(1,2), 8, c(12,13,14), c(7,8,9), "upper")
```

---

qchyper *Quantile function for conditional hypergeometric distributions*

Description

Calculates the quantile function of a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
qchyper(p, s, n, m, verbose = T)
```

Arguments

p	the desired quantile or quantiles
s	an integer representing the size of the intersecting population
n	a vector of integers representing the sizes of each non-intersecting population
m	a vector of integers representing the sample sizes
verbose	T/F should intermediate messages be printed?

Value

The minimum integer (or integers for a vector input) such that the input probability is less than or equal to the probability of sampling that many of the same items in all samples.

Examples

```
qchyper(c(0,0.9,1), 10, c(12,13,14), c(7,8,9))
```

`rchyper` *Random number generator for conditional hypergeometric distributions*

Description

Generates random numbers from a conditional hypergeometric distribution: the distribution of how many items are in the overlap of all samples when samples of arbitrary size are each taken without replacement from populations of arbitrary size.

Usage

```
rchyper(size, s, n, m, verbose = T)
```

Arguments

`size`	the number of random numbers to generate
`s`	an integer representing the size of the intersecting population
`n`	a vector of integers representing the sizes of each non-intersecting population
`m`	a vector of integers representing the sample sizes
`verbose`	T/F should intermediate messages be printed?

Value

A vector of random numbers generated from the PMF of the conditional hypergeometric distribution specified by the parameters

Examples

```
rchyper(100, 10, c(12,13,14), c(7,8,9))
```

Index