Package 'recapr'

July 3, 2020

Type Package

Title Two Event Mark-Recapture Experiment

Version 0.4.3

Date 2020-07-01

Description Tools are provided for estimating, testing, and simulating
abundance in a two-event (Petersen) mark-recapture experiment. Functions are
given to calculate the Petersen, Chapman, and Bailey estimators and associated
variances. However, the principal utility is a set of functions to simulate
random draws from these estimators, and use these to conduct hypothesis
tests and power calculations. Additionally, a set of functions are provided
for generating confidence intervals via bootstrapping. Functions are also
provided to test abundance estimator consistency under complete or partial
stratification, and to calculate stratified or partially stratified estimators.
Functions are also provided to calculate recommended sample sizes.
Referenced methods can be found in Arnason et al. (1996) <ISSN:0706-6457>,
Bailey (1951) <DOI:10.2307/2332575>,
Bailey (1952) <DOI:10.2307/1913>,
Chapman (1951) NAID:20001644490,
Cohen (1988) ISBN:0-12-179060-6,
Darroch (1961) <DOI:10.2307/2332748>, and
Robson and Regier (1964) <ISSN:1548-8659>.

License GPL-2

LazyData TRUE

Imports MASS

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.1.0

NeedsCompilation no

Author Matt Tyers [aut, cre]

Maintainer Matt Tyers <matt.tyers@alaska.gov>

Repository CRAN

Date/Publication 2020-07-03 10:50:02 UTC

R **topics documented:**

`recapr-package`	*Estimating, Testing, and Simulating Abundance in a Mark-Recapture Experiment*

Description

Tools are provided for estimating, testing, and simulating abundance in a two-event (Petersen) mark-recapture experiment. Functions are given to calculate the Petersen, Chapman, and Bailey estimators and associated variances. However, the principal utility is a set of functions to simulate random draws from these estimators, and use these to conduct hypothesis tests and power calculations. Additionally, a set of functions are provided for generating confidence intervals via bootstrapping. Functions are also provided to test abundance estimator consistency under complete or partial stratification, and to calculate stratified or Darroch estimators. Functions are also provided to calculate recommended sample sizes.

Details

Package:	recapr
Type:	Package
Version:	0.4.2
Date:	2020-06-19
License:	GPL-2

Author(s)

Matt Tyers

Maintainer: Matt Tyers <matt.tyers@alaska.gov>

`ciBailey`	*Confidence Intervals for the Bailey Estimator*

Description

Calculates approximate confidence intervals(s) for the Bailey estimator, using bootstrapping, the Normal approximation, or both.

The bootstrap interval is created by resampling the data in the second sampling event, with replacement; that is, drawing bootstrap values of m2 from a binomial distribution with probability parameter m2/n2. This technique has been shown to better approximate the distribution of the abundance estimator. Resulting CI endpoints both have larger values than those calculated from a normal distribution, but this better captures the positive skew of the estimator. Coverage has been investigated by means of simulation under numerous scenarios and has consistently outperformed the normal interval. The user is welcomed to investigate the coverage under relevant scenarios.

Usage

```
ciBailey(n1, n2, m2, conf = 0.95, method = "both", bootreps = 10000)
```

Arguments

n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of marked individuals recaptured in the second sample
conf	The confidence level of the desired intervals. Defaults to 0.95.
method	Which method of confidence interval to return. Allowed values are "norm", "boot", or "both". Defaults to "both".
bootreps	Number of bootstrap replicates to use. Defaults to 10000.

Value

A list with the abundance estimate and confidence interval bounds for the normal-distribution and/or bootstrap confidence intervals.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, vBailey, seBailey, rBailey, pBailey, powBailey

Examples

```
ciBailey(n1=100, n2=100, m2=20)
```

---

ciChapman	*Confidence Intervals for the Chapman Estimator*

Description

Calculates approximate confidence intervals(s) for the Chapman estimator, using bootstrapping, the Normal approximation, or both.

The bootstrap interval is created by resampling the data in the second sampling event, with re-placement; that is, drawing bootstrap values of m2 from a binomial distribution with probability parameter m2/n2. This technique has been shown to better approximate the distribution of the abundance estimator. Resulting CI endpoints both have larger values than those calculated from a normal distribution, but this better captures the positive skew of the estimator. Coverage has been investigated by means of simulation under numerous scenarios and has consistently outperformed the normal interval. The user is welcomed to investigate the coverage under relevant scenarios.

Usage

```
ciChapman(n1, n2, m2, conf = 0.95, method = "both", bootreps = 10000)
```

Arguments

n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of marked individuals recaptured in the second sample
conf	The confidence level of the desired intervals. Defaults to 0.95.
method	Which method of confidence interval to return. Allowed values are "norm", "boot", or "both". Defaults to "both".
bootreps	Number of bootstrap replicates to use. Defaults to 10000.

Value

A list with the abundance estimate and confidence interval bounds for the normal-distribution and/or bootstrap confidence intervals.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

[NChapman](), [vChapman](), [seChapman](), [rChapman](), [pChapman](), [powChapman]()

Examples

```
ciChapman(n1=100, n2=100, m2=20)
```

---

ciPetersen	*Confidence Intervals for the Petersen Estimator*

Description

Calculates approximate confidence intervals(s) for the Petersen estimator, using bootstrapping, the Normal approximation, or both.

The bootstrap interval is created by resampling the data in the second sampling event, with replacement; that is, drawing bootstrap values of m2 from a binomial distribution with probability parameter m2/n2. This technique has been shown to better approximate the distribution of the abundance estimator. Resulting CI endpoints both have larger values than those calculated from a normal distribution, but this better captures the positive skew of the estimator. Coverage has been investigated by means of simulation under numerous scenarios and has consistently outperformed the normal interval. The user is welcomed to investigate the coverage under relevant scenarios.

Usage

```
ciPetersen(
  n1,
  n2,
  m2,
  conf = 0.95,
  method = "both",
  bootreps = 10000,
  useChapvar = FALSE
)
```

Arguments

n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of marked individuals recaptured in the second sample
conf	The confidence level of the desired intervals. Defaults to 0.95.
method	Which method of confidence interval to return. Allowed values are "norm", "boot", or "both". Defaults to "both".

bootreps	Number of bootstrap replicates to use. Defaults to 10000.
useChapvar	Whether to use the Chapman estimator variance instead of the Petersen estimator variance for the normal-distribution interval. Defaults to FALSE.

Value

A list with the abundance estimate and confidence interval bounds for the normal-distribution and/or bootstrap confidence intervals.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, vPetersen, sePetersen, rPetersen, pPetersen, powPetersen

Examples

```
ciPetersen(n1=100, n2=100, m2=20)
```

---

cistrat	*Confidence Intervals for the Stratified Estimator*

Description

Calculates approximate confidence intervals(s) for the Stratified estimator, using bootstrapping, the Normal approximation, or both.

The bootstrap interval is created by resampling the data in the second sampling event, with replacement for each stratum; that is, drawing bootstrap values of m2 from a binomial distribution with probability parameter m2/n2.

Usage

```
cistrat(
  n1,
  n2,
  m2,
  conf = 0.95,
  method = "both",
  bootreps = 10000,
  estimator = "Chapman",
  useChapvar = FALSE
)
```

Arguments

n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of marked individuals recaptured in the second sample
conf	The confidence level of the desired intervals. Defaults to 0.95.
method	Which method of confidence interval to return. Allowed values are `"norm"`, `"boot"`, or `"both"`. Defaults to `"both"`.
bootreps	Number of bootstrap replicates to use. Defaults to 10000.
estimator	The type of estimator to use. Allowed values are `"Chapman"`, `"Petersen"`, and `"Bailey"`. Default to `"Chapman"`.
useChapvar	Whether to use the Chapman estimator variance instead of the Petersen estimator variance for the normal-distribution interval, if `"method"` is set to `"Petersen"`. Defaults to FALSE.

Value

A list with the abundance estimate and confidence interval bounds for the normal-distribution and/or bootstrap confidence intervals.

Note

Both the bootstrap and the normal approximation intervals make the naive assumption of independence between strata, which may not be the case. The user therefore cautioned, and is encouraged to investigate the coverage under relevant scenarios.

Author(s)

Matt Tyers

See Also

\linkstrattest, Nstrat, rstrat, vstrat, sestrat, NChapman, NPetersen, NBailey

Examples

```
cistrat(n1=c(100,200), n2=c(100,500), m2=c(10,10))
```

| consistencytest | *Consistency Tests for the Abundance Estimator, Partial Stratification* |

Description

Conducts three chi-squared tests for the consistency of the Petersen-type abundance estimator. These tests explore evidence against the second traditional assumption of the Petersen mark-recapture experiment: that equal capture probabilities exist in either the first or second sampling event, or that complete mixing occurs between events.

Typically, if any of these test p-values is greater than the significance level, use of a Petersen-type estimator is considered justified. If all three tests give p-values below the significance level and no movement occurs between strata (and strata are the same between events), a stratified estimator may be used. If all three tests give p-values below the significance level and some movement between strata occurs, a partially stratified (Darroch-type) estimator must be used, such as NDarroch.

This function assumes stratification in both sampling events, and in different ways (by time, area, etc.) If stratification was the same in both events such that individuals could not move from one strata to another (such as by size or gender), use of strattest is recommended.

Usage

```
consistencytest(
  n1,
  n2,
  m2strata1 = NULL,
  m2strata2 = NULL,
  stratamat = NULL,
  ...
)
```

Arguments

n1	A vector of the total sample sizes in the first event, by strata. For example, setting n1=c(20,30,40) would imply 20 individuals captured and marked in stratum 1, 30 in stratum 2, and 40 in stratum 3.
n2	A vector of the total sample sizes in the second event, by strata.
m2strata1	A vector of the first-event stratum membership of each recaptured individual. Only values 1,2,3,... are allowed. May be used together with m2strata2 instead of stratamat.
m2strata2	A vector of the second-event stratum membership of each recaptured individual. Only values 1,2,3,... are allowed. May be used together with m2strata1 instead of stratamat.

stratamat   A matrix specifying the number of recaptures in each combination of event 1 and event 2 strata, with rows corresponding to event 1 strata and columns corresponding to event 2 strata. May be used instead of `m2strata1` and `m2strata2`.

...   Additional arguments for chisq.test

Value

A list of class `"recapr_consistencytest"` with the following components:

- `test1_tab` The contingency table used for the first test
- `test1_Xsqd` The chi-squared test statistic in the first test
- `test1_df` The associated degrees of freedom in the first test
- `test1_pval` The p-value returned from the first test
- `test2_tab` The contingency table used for the second test
- `test2_Xsqd` The chi-squared test statistic in the second test
- `test2_df` The associated degrees of freedom in the second test
- `test2_pval` The p-value returned from the second test
- `test3_tab` The contingency table used for the third test
- `test3_Xsqd` The chi-squared test statistic in the third test
- `test3_df` The associated degrees of freedom in the third test
- `test3_pval` The p-value returned from the third test

Note

Naming conventions for the second and third tests are taken from SPAS (see reference)

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

References

Stratified Population Analysis System (Arnason, A.N., C.W. Kirby, C.J. Schwarz and J.R. Irvine. 1996. Computer Analysis of Data from Stratified Mark-Recovery Experiments for Estimation of Salmon Escapements and Other Populations, Canadian Technical Report of Fisheries and Aquatic Sciences 2106).

See Also

strattest, NDarroch

Examples

```
consistencytest(n1=c(15,12,6), n2=c(12,9,10,8),
   m2strata1=c(1,1,1,1,1,2,2,2,3,3),
   m2strata2=c(1,1,3,3,4,1,2,4,1,3),
   simulate.p.value=TRUE)

mat <- matrix(c(30,15,1,0,22,15), nrow=2, ncol=3, byrow=TRUE)
consistencytest(n1=c(284,199), n2=c(347,3616,1489), stratamat=mat)
```

---

n2RR   *Mark-Recapture Sample Size, Robson-Regier*

Description

Calculates minimum sample size for one sampling event in a Petersen mark-recapture experiment, given the sample size in the other event and an best guess at true abundance.

Usage

```
n2RR(
  N,
  n1,
  conf = c(0.99, 0.95, 0.9, 0.85, 0.8, 0.75),
  acc = c(0.5, 0.25, 0.2, 0.15, 0.1, 0.05, 0.01)
)
```

Arguments

N	The best guess at true abundance
n1	The size of the first (or second) sampling event
conf	A vector of the desired levels of confidence to investigate. Allowed values are any of `c(0.99,0.95,0.9,0.85,0.8,0.75)`. Defaults to all of `c(0.99,0.95,0.85,0.8,0.75)`.
acc	A vector of the desired levels of relative accuracy to investigate. Allowed values are any of `c(0.5,0.25,0.2,0.15,0.1,0.05,0.01)`. Defaults to all of `c(0.5,0.25,0.2,0.15,0.1,0.05,0.01)`.

Value

A list of minimum sample sizes. Each list element corresponds to a unique level of confidence, and is defined as a data frame with each row corresponding to a unique value of relative accuracy. Two minimum sample sizes are given: one calculated from the sample size provided for the other event, and the other calculated under n1=n2, the most efficient scenario.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

It is possible that the sample size - accuracy relationship will be better illustrated using plotn2sim.

Author(s)

Matt Tyers

References

Robson, D. S., and H. A. Regier. 1964. Sample size in Petersen mark-recapture experiments. Transactions of the American FisheriesSociety 93:215-226.

See Also

plotn2sim, plotn1n2simmatrix

Examples

```
n2RR(N=1000, n1=100)
```

NBailey	*Bailey Estimator*

Description

Calculates the value of the Bailey estimator for abundance in a mark-recapture experiment, with given values of sample sizes and number of recaptures. The Bailey estimator assumes a binomial probability model in the second sampling event (i.e. sampling with replacement), rather than the hypergeometric model assumed by the Petersen and Chapman estimators.

Usage

```
NBailey(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The value of the Bailey estimator, calculated as n1*(n2+1)/(m2+1)

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

References

Bailey, N.T.J. (1951). On estimating the size of mobile populations from capture-recapture data. *Biometrika* **38**, 293-306.

Bailey, N.T.J. (1952). Improvements in the interpretation of recapture data. *J. Animal Ecol.* **21**, 120-7.

See Also

NPetersen, NChapman, vBailey, seBailey, rBailey, pBailey, powBailey, ciBailey

Examples

```
NBailey(n1=100, n2=100, m2=20)
```

NChapman   *Chapman Estimator*

Description

Calculates the value of the Chapman estimator for abundance in a mark-recapture experiment, with given values of sample sizes and number of recaptures. The Chapman estimator (Chapman modification of the Petersen estimator) typically outperforms the Petersen estimator, even though the Peterson estimator is the MLE.

Usage

```
NChapman(n1, n2, m2)
```

Arguments

n1   Number of individuals captured and marked in the first sample. This may be a single number or vector of values.

n2   Number of individuals captured in the second sample. This may be a single number or vector of values.

m2   Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The value of the Chapman estimator, calculated as (n1+1)*(n2+1)/(m2+1) - 1

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

References

Chapman, D.G. (1951). Some properties of the hypergeometric distribution with applications to zoological censuses. *Univ. Calif. Public. Stat.* **1**, 131-60.

See Also

NPetersen, NBailey, vChapman, seChapman, rChapman, pChapman, powChapman, ciChapman

Examples

```
NChapman(n1=100, n2=100, m2=20)
```

NDarroch	*Spatially or Temporally Stratified Abundance Est (Darroch)*

Description

Computes abundance estimates and associated variance in the event of spatial or temporal stratification, or in any stratification in which individuals can move between strata. Marking (event 1) and recapture (event 2) strata do not need to be the same.

Inputs are vectors of total event 1 and 2 sample sizes, and either vectors of event 1 and 2 strata corresponding to each recaptured individual, or a matrix of total number of recaptures for each combination of event 1 and event 2 strata.

Implementation is currently using Darroch's method, and will only accept non-singular input matrices.

Usage

```
NDarroch(
  n1counts,
  n2counts,
  m2strata1 = NULL,
  m2strata2 = NULL,
  stratamat = NULL
)
```

Arguments

n1counts	A vector of the total sample sizes in the first event, by strata. For example, setting n1counts=c(20,30,40) would imply 20 individuals captured and marked in stratum 1, 30 in stratum 2, and 40 in stratum 3.
n2counts	A vector of the total sample sizes in the second event, by strata.
m2strata1	A vector of the first-event stratum membership of each recaptured individual. Only values 1,2,3,... are allowed. May be used together with m2strata2 instead of stratamat.
m2strata2	A vector of the second-event stratum membership of each recaptured individual. Only values 1,2,3,... are allowed. May be used together with m2strata1 instead of stratamat.
stratamat	A matrix specifying the number of recaptures in each combination of event 1 and event 2 strata, with rows corresponding to event 1 strata and columns corresponding to event 2 strata. May be used instead of m2strata1 and m2strata2.

Value

A numeric list, giving the strata matrix if originally given in vector form, abundance estimates and standard errors by event 1 and event 2 strata, and the total abundance estimate and standard error.

Author(s)

Matt Tyers

References

Darroch, J.N. (1961). The two-sample capture-recapture census when tagging and sampling are stratified. *Biometrika* **48**, 241-60.

See Also

consistencytest

Examples

```
mat <- matrix(c(59,30,1,45,280,38,0,42,25), nrow=3, ncol=3, byrow=TRUE)

NDarroch(n1counts=c(484,1468,399)), n2counts=c(847,6616,2489), stratamat=mat)
```

NPetersen   *Petersen Estimator*

Description

Calculates the value of the Petersen estimator for abundance in a mark-recapture experiment, with given values of sample sizes and number of recaptures. The Petersen estimator is the MLE, but is typically outperformed by the Chapman estimator.

Usage

```
NPetersen(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The value of the Petersen estimator, calculated as n1*n2/m2

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, NBailey, vPetersen, sePetersen, rPetersen, pPetersen, powPetersen, ciPetersen

Examples

```
NPetersen(n1=100, n2=100, m2=20)
```

Nstrat	*Stratified Abundance Estimator*

Description

Calculates the value of the stratified estimator for abundance in a mark-recapture experiment, from vectors of sample sizes and number of recaptures, with each element corresponding to each sampling stratum.

Usage

```
Nstrat(n1, n2, m2, estimator = "Chapman")
```

Arguments

n1	Vector of individuals captured and marked in the first sample, from each stratum
n2	Vector of individuals captured and marked in the second sample, from each stratum
m2	Vector of marked individuals recaptured in the second sample, from each stratum
estimator	The type of estimator to use. Allowed values are "Chapman", "Petersen", and "Bailey". Default to "Chapman".

Value

The value of the stratified estimator

Note

It is possible that even the stratified estimate may be biased if capture probabilities differ greatly between strata. However, the bias in the stratified estimator will be much less than an estimator calculated without stratification.

Author(s)

Matt Tyers

See Also

strattest, rstrat, vstrat, sestrat, cistrat, NChapman, NPetersen, NBailey

Examples

```
Nstrat(n1=c(100,200), n2=c(100,500), m2=c(10,10))
```

---

pBailey   *Hypothesis Testing Using the Bailey Estimator*

Description

Approximates a p-value for a hypothesis test of the Bailey estimator by means of many simulated draws from the null distribution, conditioned on sample sizes.

Usage

```
pBailey(
  estN = NULL,
  nullN,
  n1,
  n2,
  m2 = NULL,
  nsim = 1e+05,
  alternative = "less"
)
```

Arguments

estN	The estimated abundance. Either this or the number of recaptures (`m2`) must be specified.
nullN	The abundance given by the null hypothesis
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of recaptures. Either this or the estimated abundance (`estN`) must be specified.
nsim	Number of simulated values to draw. Defaults to 100000.
alternative	Direction of the alternative hypothesis. Allowed values are `"less"`, `"greater"`, or `"2-sided"`. Defaults to `"less"`.

Value

An approximate p-value for the specified hypothesis test. If `m2` is specified rather than `estN`, output will be returned as a list with two elements: the estimated abundance and p-value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, vBailey, seBailey, rBailey, powBailey, ciBailey

Examples

```
output <- pBailey(nullN=500, n1=100, n2=100, m2=28)
output

plotdiscdensity(rBailey(length=100000, N=500, n1=100, n2=100))
abline(v=output$estN, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)
```

```
output <- pBailey(nullN=500, n1=100, n2=100, m2=28, alternative="2-sided")
output

plotdiscdensity(rBailey(length=100000, N=500, n1=100, n2=100))
twosided <- 500 + c(-1,1)*abs(500-output$estN)
abline(v=twosided, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)
```

---

pChapman   *Hypothesis Testing Using the Chapman Estimator*

Description

Approximates a p-value for a hypothesis test of the Chapman estimator by means of many simulated
draws from the null distribution, conditioned on sample sizes.

Usage

```
pChapman(
  estN = NULL,
  nullN,
  n1,
  n2,
  m2 = NULL,
  nsim = 1e+05,
  alternative = "less"
)
```

Arguments

estN	The estimated abundance. Either this or the number of recaptures (m2) must be specified.
nullN	The abundance given by the null hypothesis
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of recaptures. Either this or the estimated abundance (estN) must be specified.
nsim	Number of simulated values to draw. Defaults to 100000.
alternative	Direction of the alternative hypothesis. Allowed values are "less", "greater", or "2-sided". Defaults to "less".

Value

An approximate p-value for the specified hypothesis test. If m2 is specified rather than estN, output
will be returned as a list with two elements: the estimated abundance and p-value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, vChapman, seChapman, rChapman, powChapman, ciChapman

Examples

```
output <- pChapman(nullN=500, n1=100, n2=100, m2=28)
output

plotdiscdensity(rChapman(length=100000, N=500, n1=100, n2=100))
abline(v=output$estN, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)


output <- pChapman(nullN=500, n1=100, n2=100, m2=28, alternative="2-sided")
output

plotdiscdensity(rChapman(length=100000, N=500, n1=100, n2=100))
twosided <- 500 + c(-1,1)*abs(500-output$estN)
abline(v=twosided, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)
```

| plotdiscdensity | *Plotting the Density of a Vector of Discrete Values* |

Description

Plots the empirical density of a vector of discrete values, approximating the probability mass function (pmf). This can be considered a more appropriate alternative to plot(density(x)) in the case of a vector with a discrete (non-continuous) support, such as that calculated by an abundance estimator.

Usage

```
plotdiscdensity(x, xlab = "value", ylab = "density", ...)
```

Arguments

x	The vector of values to plot
xlab	The X-axis label for plotting
ylab	The Y-axis label for plotting
...	Additional plotting arguments

Author(s)

Matt Tyers

Examples

```
draws <- rChapman(length=100000, N=500, n1=100, n2=100)
plotdiscdensity(draws)  #plots the density of a vector of discrete values
```

---

plotn1n2simmatrix   *Mark-Recapture Sample Size Via Sim, Both Samples*

Description

Produces a plot of the simulated relative accuracy of a mark-recapture abundance estimator for various sample sizes in both events. This may be a useful exploration, but it is possible that plotn2sim may be more informative.

Usage

```
plotn1n2simmatrix(
  N,
  conf = 0.95,
  nrange = NULL,
  nstep = NULL,
  estimator = "Chapman",
  nsim = 10000,
  ...
)
```

Arguments

N	The best guess at true abundance
conf	The desired level of confidence to investigate. Defaults to 0.95.
nrange	A two-element vector describing the range of sample sizes to investigate. If the default (NULL) is accepted, an appropriate value will be chosen.
nstep	The step size between sample sizes to investigate. If the default (NULL) is accepted, an appropriate value will be chosen.
estimator	The abundance estimator to use. Allowed values are "Chapman", "Petersen", and "Bailey". Defaults to "Chapman".
nsim	The number of replicates. Defaults to 10000.
...	Additional plotting parameters

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events

- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events

- Marking in the first event does not affect probability of recapture in the second event

- Individuals do not lose marks between events

- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

n2RR, plotn2sim

Examples

```
plotn1n2simmatrix(N=10000, nsim=2000)
```

plotn2sim   *Mark-Recapture Sample Size Via Simulation*

Description

Produces a plot of the simulated relative accuracy of a mark-recapture abundance estimator for various sample sizes. This may be a better representation of the sample size - accuracy relationship than that provided by n2RR.

Usage

```
plotn2sim(
  N,
  n1,
  conf = c(0.99, 0.95, 0.85, 0.8, 0.75),
  n2range = NULL,
  n2step = NULL,
  estimator = "Chapman",
  nsim = 10000,
  accrange = 1,
  ...
)
```

Arguments

N	The best guess at true abundance
n1	The size of the first (or second) sampling event
conf	A vector of the desired levels of confidence to investigate. Allowed values are any of `c(0.99,0.95,0.85,0.8,0.75)`. Defaults to all of `c(0.99,0.95,0.85,0.8,0.75)`.
n2range	A two-element vector describing the range of sample sizes to investigate. If the default (`NULL`) is accepted, an appropriate value will be chosen.
n2step	The step size between sample sizes to investigate. If the default (`NULL`) is accepted, an appropriate value will be chosen.
estimator	The abundance estimator to use. Allowed values are `"Chapman"`, `"Petersen"`, and `"Bailey"`. Defaults to `"Chapman"`.
nsim	The number of replicates. Defaults to 10000.
accrange	The maximum level of relative accuracy for plotting. Defaults to 1.
...	Additional plotting parameters

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events

- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

n2RR, plotn1n2simmatrix

Examples

```
plotn2sim(N=1000, n1=100)
```

---

powBailey   *Power for Hypothesis Testing Using the Bailey Estimator*

Description

Approximates the power of a hypothesis test of the Bailey estimator by means of many simulated draws from a specified alternative distribution, conditioned on sample sizes.

Usage

```
powBailey(
  nullN,
  trueN,
  n1,
  n2,
  alpha = 0.05,
  nsim = 10000,
  alternative = "less"
)
```

Arguments

nullN	The abundance given by the null hypothesis
trueN	The assumed abundance for the power calculation
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
alpha	The alpha level for the test
nsim	Number of simulated values to draw. Defaults to 10000.
alternative	Direction of the alternative hypothesis. Allowed values are "less", "greater", or "2-sided". Defaults to "less".

Value

The approximate power of the specified hypothesis test, for the specified alternative value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, vBailey, seBailey, rBailey, pBailey, ciBailey

Examples

```
powBailey(nullN=500, trueN=400, n1=100, n2=100, nsim=1000)

Ntotry <- seq(from=250, to=450, by=25)
pows <- sapply(Ntotry, function(x)
  powBailey(nullN=500, trueN=x, n1=100, n2=100, nsim=1000))
plot(Ntotry, pows)
```

---

powChapman   *Power for Hypothesis Testing Using the Chapman Estimator*

Description

Approximates the power of a hypothesis test of the Chapman estimator by means of many simulated draws from a specified alternative distribution, conditioned on sample sizes.

Usage

```
powChapman(
  nullN,
  trueN,
  n1,
  n2,
  alpha = 0.05,
  nsim = 10000,
  alternative = "less"
)
```

Arguments

nullN	The abundance given by the null hypothesis
trueN	The assumed abundance for the power calculation
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
alpha	The alpha level for the test
nsim	Number of simulated values to draw. Defaults to 10000.
alternative	Direction of the alternative hypothesis. Allowed values are "less", "greater", or "2-sided". Defaults to "less".

Value

The approximate power of the specified hypothesis test, for the specified alternative value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, vChapman, seChapman, rChapman, pChapman, ciChapman

Examples

```
powChapman(nullN=500, trueN=400, n1=100, n2=100, nsim=1000)

Ntotry <- seq(from=250, to=450, by=25)
pows <- sapply(Ntotry, function(x)
  powChapman(nullN=500, trueN=x, n1=100, n2=100, nsim=1000))
plot(Ntotry, pows)
```

---

powconsistencytest   *Power of Consistency Tests, Partial Stratification*

Description

Conducts power calculations of the chi-squared tests for the consistency of the Petersen-type abundance estimator, in a partial stratification setting, such as by time or geographic area. In the case of partial stratification, individuals may move from one stratum to another between the first and second sampling events, and strata do not need to be the same between events.

Usage

```
powconsistencytest(n1, n2, pmat, alpha = 0.05, sim = TRUE, nsim = 10000)
```

Arguments

n1	Vector of anticipated n1 counts (sample size in the first event), each element corresponding to one stratum.
n2	Vector of anticipated n2 counts (sample size in the second event), each element corresponding to one stratum.
pmat	Matrix of assumed movement probabilities between strata, with rows corresponding to first-event strata and columns corresponding to second-event strata, and an additional column corresponding to the probability of NOT being recaptured in the second event. Values will be standardized by row, that is, by first-event strata. See note on usage below.
alpha	Significance level for testing. Defaults to 0.05
sim	Whether to conduct power calculation by simulation as well as Cohen's method. Defaults to TRUE.
nsim	Number of simulations if sim is TRUE. Defaults to 10000.

Value

An object of class "recapr_consistencypow" with the following components:

- pwr1_c Power of the first test, according to Cohen's method
- pwr1_sim Power of the first test, from simulation
- ntest1 The sample size used for the first test

- `p0test1` The null-hypothesis probabilities for the first test
- `p1test1` The alt-hypothesis probabilities for the first test
- `pwr2_c` Power of the second test, according to Cohen's method
- `pwr2_sim` Power of the second test, from simulation
- `ntest2` The sample size used for the second test
- `p0test2` The null-hypothesis probabilities for the second test
- `p1test2` The alt-hypothesis probabilities for the second test
- `pwr3_c` Power of the third test, according to Cohen's method
- `pwr3_sim` Power of the third test, from simulation
- `ntest3` The sample size used for the third test
- `p0test3` The null-hypothesis probabilities for the third test
- `p1test3` The alt-hypothesis probabilities for the third test
- `alpha` The significance level used

Note

The movement probability matrix specified in `pmat` is considered conditional on each row, that is, first-event strata, with columns corresponding to second-event strata and the final column specifying the probability of not being recaptured in the second event. Values do not need to sum to one for each row, but will be standardized by the function to sum to one.

A `pmat` with a first row equal to (`0.05,0.1,0.15,0.7`) would imply a 5 percent chance that individuals captured in the first-event strata 1 will be recaptured in second-event strata 1, and a 70 percent chance that individuals captured in the first-event strata 1 will not be recaptured in event 2.

Because of the row-wise scaling, specifying a row equal to (`0.05,0.1,0.15,0.7`) would be equivalent to that row having values (`1,2,3,14`).

Author(s)

Matt Tyers

References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). Hillsdale,NJ: Lawrence Erlbaum.

Code adapted from the 'pwr' package: Stephane Champely (2015). pwr: Basic Functions for Power Analysis. R package version 1.1-3. https://CRAN.R-project.org/package=pwr

See Also

consistencytest, NDarroch

Examples

```
mat <- matrix(c(4,3,2,1,10,3,4,3,2,10,2,3,4,3,10,1,2,3,4,10),
    nrow=4, ncol=5, byrow=TRUE)
powconsistencytest(n1=c(50,50,50,50), n2=c(50,50,50,50), pmat=mat)

mat <- matrix(c(4,3,2,1,10,4,3,2,1,10,4,3,2,1,10,4,3,2,1,10),
    nrow=4, ncol=5, byrow=TRUE)
powconsistencytest(n1=c(50,50,50,50), n2=c(50,50,50,50), pmat=mat)

mat <- matrix(c(1,1,1,1,10,2,2,2,2,10,3,3,3,3,10,4,4,4,4,10),
    nrow=4, ncol=5, byrow=TRUE)
powconsistencytest(n1=c(50,50,50,50), n2=c(50,50,50,50), pmat=mat)

mat <- matrix(c(1,1,1,1,10,1,1,1,1,10,1,1,1,1,10,1,1,1,1,10),
    nrow=4, ncol=5, byrow=TRUE)
powconsistencytest(n1=c(50,50,50,50), n2=c(20,30,40,50), pmat=mat)

mat <- matrix(c(1,1,1,1,5,1,1,1,1,8,1,1,1,1,10,1,1,1,1,15),
    nrow=4, ncol=5, byrow=TRUE)
powconsistencytest(n1=c(50,50,50,50), n2=c(50,50,50,50), pmat=mat)
```

---

powPetersen	*Power for Hypothesis Testing Using the Petersen Estimator*

Description

Approximates the power of a hypothesis test of the Petersen estimator by means of many simulated draws from a specified alternative distribution, conditioned on sample sizes.

Usage

```
powPetersen(
  nullN,
  trueN,
  n1,
  n2,
  alpha = 0.05,
  nsim = 10000,
  alternative = "less"
)
```

Arguments

nullN	The abundance given by the null hypothesis
trueN	The assumed abundance for the power calculation
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample

alpha	The alpha level for the test
nsim	Number of simulated values to draw. Defaults to 10000.
alternative	Direction of the alternative hypothesis. Allowed values are `"less"`, `"greater"`, or `"2-sided"`. Defaults to `"less"`.

Value

The approximate power of the specified hypothesis test, for the specified alternative value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events

- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events

- Marking in the first event does not affect probability of recapture in the second event

- Individuals do not lose marks between events

- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, vPetersen, sePetersen, rPetersen, pPetersen, ciPetersen

Examples

```
powPetersen(nullN=500, trueN=400, n1=100, n2=100, nsim=1000)

Ntotry <- seq(from=250, to=450, by=25)
pows <- sapply(Ntotry, function(x)
  powPetersen(nullN=500, trueN=x, n1=100, n2=100, nsim=1000))
plot(Ntotry, pows)
```

powstrattest   *Power of Consistency Tests, Complete Stratification*

Description

Conducts power calculations of the chi-squared tests for the consistency of the Petersen-type abundance estimator, in a complete stratification setting.

Usage

```
powstrattest(N, n1, n2, alpha = 0.05, sim = TRUE, nsim = 1e+05)
```

Arguments

N   Vector of total abundance, with each element corresponding to one stratum.

n1   Vector of anticipated n1 counts (sample size in the first event), each element corresponding to one stratum.

n2   Vector of anticipated n2 counts (sample size in the second event), each element corresponding to one stratum.

alpha   Significance level for testing. Defaults to `0.05`

sim   Whether to conduct power calculation by simulation as well as Cohen's method. Defaults to `TRUE`.

nsim   Number of simulations if `sim` is `TRUE`. Defaults to `100000`.

Value

A list of three elements, each with class `"recapr_stratpow"` with the following components:

- `prob` A vector of capture probabilities corresponding to the alternative hypothesis investigated
- `prob_null` A vector of capture probabilities corresponding to the null hypothesis (all probabilities equal)
- `n` The sample size used for the test
- `alpha` The significance level used for testing
- `power` The test power, calculated by Cohen's method
- `power_sim` The test power, calculated via simulation

Author(s)

Matt Tyers

References

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). Hillsdale,NJ: Lawrence Erlbaum.

Code adapted from the 'pwr' package: Stephane Champely (2015). pwr: Basic Functions for Power Analysis. R package version 1.1-3. https://CRAN.R-project.org/package=pwr

See Also

strattest, Nstrat

Examples

```
powstrattest(N=c(10000,20000), n1=c(1000,2000), n2=c(200,200))
```

---

pPetersen	*Hypothesis Testing Using the Petersen Estimator*

Description

Approximates a p-value for a hypothesis test of the Petersen estimator by means of many simulated draws from the null distribution, conditioned on sample sizes.

Usage

```
pPetersen(
  estN = NULL,
  nullN,
  n1,
  n2,
  m2 = NULL,
  nsim = 1e+05,
  alternative = "less"
)
```

Arguments

estN	The estimated abundance. Either this or the number of recaptures (m2) must be specified.
nullN	The abundance given by the null hypothesis
n1	Number of individuals captured and marked in the first sample
n2	Number of individuals captured in the second sample
m2	Number of recaptures. Either this or the estimated abundance (estN) must be specified.
nsim	Number of simulated values to draw. Defaults to 100000.
alternative	Direction of the alternative hypothesis. Allowed values are "less", "greater", or "2-sided". Defaults to "less".

Value

An approximate p-value for the specified hypothesis test. If m2 is specified rather than estN, output will be returned as a list with two elements: the estimated abundance and p-value.

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, vPetersen, sePetersen, rPetersen, powPetersen, ciPetersen

Examples

```
output <- pPetersen(nullN=500, n1=100, n2=100, m2=28)
output

plotdiscdensity(rPetersen(length=100000, N=500, n1=100, n2=100))
abline(v=output$estN, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)


output <- pPetersen(nullN=500, n1=100, n2=100, m2=28, alternative="2-sided")
output

plotdiscdensity(rPetersen(length=100000, N=500, n1=100, n2=100))
twosided <- 500 + c(-1,1)*abs(500-output$estN)
abline(v=twosided, lwd=2, col=2)
abline(v=500, lwd=2, lty=2)
```

---

print.recapr_consistencypow
                                 Print method for consistency test power

---

Description

Print method for consistency test power

Usage

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

Arguments

x	Output from powconsistencytest()
...	additional print arguments

Author(s)

Matt Tyers

---

print.recapr_consistencytest

Print method for consistency test

Description

Print method for consistency test

Usage

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

Arguments

x	Output from consistencytest()
...	additional print arguments

Author(s)

Matt Tyers

---

print.recapr_stratpow   *Print method for stratification test power*

Description

Print method for stratification test power

Usage

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

Arguments

x	Output from `powstrattest()`
...	additional print arguments

Author(s)

Matt Tyers

---

print.recapr_strattest

Print method for stratification test

Description

Print method for stratification test

Usage

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

Arguments

x	Output from `strattest()`
...	additional print arguments

Author(s)

Matt Tyers

---

rBailey	*Random Draws from the Bailey Estimator*

Description

Returns a vector of random draws from the Bailey estimator in a mark-recapture experiment, given values of the true abundance and the sample size in both events. The function first simulates a vector of recaptures (m2) from a binomial distribution, and then uses these to compute a vector of draws from the estimator.

If capture probabilities (p1 and/or p2) are specified instead of sample size(s), the sample size(s) will first be drawn from a binomial distribution, then the number of recaptures. If both sample size and capture probability are specified for a given sampling event, only the sample size will be used.

Usage

```
rBailey(length, N, n1 = NULL, n2 = NULL, p1 = NULL, p2 = NULL)
```

Arguments

length	The length of the random vector to return.
N	The value of the true abundance. This may be a single number or vector of values equal to length.
n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values equal to length.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values equal to length.
p1	Alternately, probability of capture in the first sample. This may be a single number or vector of values equal to length.
p2	Alternately, probability of capture in the second sample. This may be a single number or vector of values equal to length.

Value

A vector of random draws from the Bailey estimator

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, vBailey, seBailey, pBailey, powBailey, ciBailey

Examples

```
draws <- rBailey(length=100000, N=500, n1=100, n2=100)
plotdiscdensity(draws)  #plots the density of a vector of discrete values
```

---

rChapman   *Random Draws from the Chapman Estimator*

Description

Returns a vector of random draws from the Chapman estimator in a mark-recapture experiment, given values of the true abundance and the sample size in both events. The function first simulates a vector of recaptures (m2) from a hypergeometric distribution, and then uses these to compute a vector of draws from the estimator.

If capture probabilities (p1 and/or p2) are specified instead of sample size(s), the sample size(s) will first be drawn from a binomial distribution, then the number of recaptures. If both sample size and capture probability are specified for a given sampling event, only the sample size will be used.

Usage

```
rChapman(length, N, n1 = NULL, n2 = NULL, p1 = NULL, p2 = NULL)
```

Arguments

length	The length of the random vector to return.
N	The value of the true abundance. This may be a single number or vector of values equal to length.
n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values equal to length.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values equal to length.
p1	Alternately, probability of capture in the first sample. This may be a single number or vector of values equal to length.
p2	Alternately, probability of capture in the second sample. This may be a single number or vector of values equal to length.

Value

A vector of random draws from the Chapman estimator

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, vChapman, seChapman, pChapman, powChapman, ciChapman

Examples

```
draws <- rChapman(length=100000, N=500, n1=100, n2=100)
plotdiscdensity(draws)  #plots the density of a vector of discrete values
```

---

rPetersen   *Random Draws from the Petersen Estimator*

Description

Returns a vector of random draws from the Petersen estimator in a mark-recapture experiment, given values of the true abundance and the sample size in both events. The function first simulates a vector of recaptures (m2) from a hypergeometric distribution, and then uses these to compute a vector of draws from the estimator.

If capture probabilities (p1 and/or p2) are specified instead of sample size(s), the sample size(s) will first be drawn from a binomial distribution, then the number of recaptures. If both sample size and capture probability are specified for a given sampling event, only the sample size will be used.

Usage

```
rPetersen(length, N, n1 = NULL, n2 = NULL, p1 = NULL, p2 = NULL)
```

Arguments

length	The length of the random vector to return.
N	The value of the true abundance. This may be a single number or vector of values equal to length.
n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values equal to length.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values equal to length.
p1	Alternately, probability of capture in the first sample. This may be a single number or vector of values equal to length.
p2	Alternately, probability of capture in the second sample. This may be a single number or vector of values equal to length.

Value

A vector of random draws from the Petersen estimator

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, vPetersen, sePetersen, pPetersen, powPetersen, ciPetersen

Examples

```
draws <- rPetersen(length=100000, N=500, n1=100, n2=100)
plotdiscdensity(draws)  #plots the density of a vector of discrete values
```

---

rstrat   *Random Draws from the Stratified Estimator*

Description

Returns a vector of random draws from the stratified estimator in a mark-recapture experiment, given values of the true abundance and the sample size in both events. The function first simulates a vector of recaptures (m2) for each stratum, and then uses these to compute a vector of draws from the estimator.

It may prove useful to investigate the behavior of the stratified estimator under relevant scenarios.

If capture probabilities (p1 and/or p2) are specified instead of sample size(s), the sample size(s) will first be drawn from a binomial distribution, then the number of recaptures. If both sample size and capture probability are specified for a given sampling event, only the sample size will be used.

Usage

```
rstrat(
  length,
  N,
  n1 = NULL,
  n2 = NULL,
  p1 = NULL,
  p2 = NULL,
  estimator = "Chapman"
)
```

Arguments

length	The length of the random vector to return.
N	A vector of values of the true abundance for each stratum.
n1	A vector of the number of individuals captured and marked in the first sample, for each stratum.
n2	A vector of the number of individuals captured in the second sample, for each stratum.
p1	Alternately, a vector of probabilities of capture for the first event, for each stratum.
p2	Alternately, a vector of probabilities of capture for the second event, for each stratum.
estimator	The type of estimator to use. Allowed values are "Chapman", "Petersen", and "Bailey". Default to "Chapman".

Value

A vector of random draws from the stratified estimator

Author(s)

Matt Tyers

See Also

strattest, Nstrat, vstrat, cistrat,NChapman, NPetersen, NBailey

Examples

```
draws <- rstrat(length=100000, N=c(5000,10000), n1=c(500,200), n2=c(500,200))
plotdiscdensity(draws)  #plots the density of a vector of discrete values
mean(draws)
```

---

seBailey   *Standard Error of the Bailey Estimator*

Description

Calculates the standard error of the Bailey estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
seBailey(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Bailey estimator, calculated as sqrt((n1^2)*(n2+1)*(n2-m2)/(m2+1)/(m2+1)/(m2+2))

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, vBailey, rBailey, pBailey, powBailey, ciBailey

Examples

```
seBailey(n1=100, n2=100, m2=20)
```

---

seChapman   *Standard Error of the Chapman Estimator*

Description

Calculates the standard error of the Chapman estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
seChapman(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Chapman estimator, calculated as sqrt((n1+1)*(n2+1)*(n1-m2)*(n2-m2)/((m2+2)*(m2+1)^2))

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, vChapman, rChapman, pChapman, powChapman, ciChapman

Examples

```
seChapman(n1=100, n2=100, m2=20)
```

---

sePetersen	*Standard Error of the Petersen Estimator*

Description

Calculates the standard error of the Petersen estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
sePetersen(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Petersen estimator, calculated as sqrt((n1^2)*n2*(n2-m2)/(m2^3))

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, vPetersen, rPetersen, pPetersen, powPetersen, ciPetersen

Examples

```
sePetersen(n1=100, n2=100, m2=20)
```

---

sestrat   *Standard Error of Stratified Abundance Estimator*

Description

Calculates the standard error of the stratified estimator for abundance in a mark-recapture experiment, from vectors of sample sizes and number of recaptures, with each element corresponding to each sampling stratum.

Usage

```
sestrat(n1, n2, m2, estimator = "Chapman")
```

Arguments

n1	Vector of individuals captured and marked in the first sample, from each stratum
n2	Vector of individuals captured and marked in the second sample, from each stratum
m2	Vector of marked individuals recaptured in the second sample, from each stratum
estimator	The type of estimator to use. Allowed values are "Chapman", "Petersen", and "Bailey". Default to "Chapman".

Value

The standard error of the stratified estimator

Note

It is possible that even the stratified estimate of abundance may be biased if capture probabilities differ greatly between strata. However, the bias in the stratified estimator will be much less than an estimator calculated without stratification.

This function makes the naive assumption of independence between strata. Caution is therefore recommended.

Author(s)

Matt Tyers

See Also

strattest, Nstrat, rstrat, vstrat, cistrat, NChapman, NPetersen, NBailey

Examples

```
sestrat(n1=c(100,200), n2=c(100,500), m2=c(10,10))
```

---

strattest	*Consistency Tests for the Abundance Estimator, Complete Stratification*

Description

Conducts two chi-squared tests for the consistency of the Petersen-type abundance estimator. These tests provide explore evidence against equal capture probabilities in either the first or second sampling event. Also conducts a third chi-squared test of unequal capture probabilities between sampling events for each stratum, in the case of small sample sizes (fewer than 100 in either sampling event and fewer than 30 recaptures), which may be used to suggest unequal capture probabilities in either the first or second event.

Typically, if either of the first two test p-values is greater than the significance level, use of a Petersen-type estimator is considered justified.

If tests give evidence of unequal capture probabilities between strata, a stratified estimator should be used, such as Nstrat.

This function assumes stratification in both sampling events, such that individuals cannot move from one strata to another (such as by size or gender). If movement between strata may occur (such as in the case of stratification by time or area), use of consistencytest is recommended.

Usage

```
strattest(n1, n2, m2, ...)
```

Arguments

n1	Vector of n1 counts (sample size in the first event), each element corresponding to one stratum.
n2	Vector of n2 counts (sample size in the second event), each element corresponding to one stratum.
m2	Vector of m2 counts (number of recaptures in the second event), each element corresponding to one stratum.
...	Additional arguments for chisq.test

Value

A list of class `"recapr_strattest"` with the following components:

- `event1_table` The contingency table used for the first test
- `event1_Xsqd` The chi-squared test statistic in the first test
- `event1_df` The associated degrees of freedom in the first test
- `event1_pval` The p-value returned from the first test
- `event2_table` The contingency table used for the second test
- `event2_Xsqd` The chi-squared test statistic in the second test
- `event2_df` The associated degrees of freedom in the second test
- `event2_pval` The p-value returned from the second test
- `event1v2_table` The contingency table used for the third test
- `event1v2_Xsqd` The chi-squared test statistic in the third test
- `event1v2_df` The associated degrees of freedom in the third test
- `event1v2_pval` The p-value returned from the second third

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

powstrattest, Nstrat, consistencytest

Examples

```
strattest(n1=c(100,100), n2=c(50,200), m2=c(20,15))
```

---

vBailey   *Estimated Variance of the Bailey Estimator*

Description

Calculates the estimated variance of the Bailey estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
vBailey(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Bailey estimator, calculated as (n1^2)*(n2+1)*(n2-m2)/(m2+1)/(m2+1)/(m2+2)

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

* The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
* All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
* Marking in the first event does not affect probability of recapture in the second event
* Individuals do not lose marks between events
* All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NBailey, seBailey, rBailey, pBailey, powBailey, ciBailey

Examples

```
vBailey(n1=100, n2=100, m2=20)
```

---

vChapman   *Estimated Variance of the Chapman Estimator*

Description

Calculates the estimated variance of the Chapman estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
vChapman(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Chapman estimator, calculated as (n1+1)*(n2+1)*(n1-m2)*(n2-m2)/((m2+2)*(m2+1)^2)

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NChapman, seChapman, rChapman, pChapman, powChapman, ciChapman

Examples

```
vChapman(n1=100, n2=100, m2=20)
```

---

vPetersen   *Estimated Variance of the Petersen Estimator*

Description

Calculates the estimated variance of the Petersen estimator in a mark-recapture experiment, with given values of sample sizes and number of recaptures.

Usage

```
vPetersen(n1, n2, m2)
```

Arguments

n1	Number of individuals captured and marked in the first sample. This may be a single number or vector of values.
n2	Number of individuals captured in the second sample. This may be a single number or vector of values.
m2	Number of marked individuals recaptured in the second sample. This may be a single number or vector of values.

Value

The estimate variance of the Petersen estimator, calculated as (n1^2)*n2*(n2-m2)/(m2^3)

Note

Any Petersen-type estimator (such as this) depends on a set of assumptions:

- The population is closed; that is, that there are no births, deaths, immigration, or emigration between sampling events
- All individuals have the same probability of capture in one of the two events, or complete mixing occurs between events
- Marking in the first event does not affect probability of recapture in the second event
- Individuals do not lose marks between events
- All marks will be reported in the second event

Author(s)

Matt Tyers

See Also

NPetersen, sePetersen, rPetersen, pPetersen, powPetersen, ciPetersen

Examples

```
vPetersen(n1=100, n2=100, m2=20)
```

---

vstrat   *Estimated Variance of Stratified Abundance Estimator*

Description

Calculates the estimated variance of the stratified estimator for abundance in a mark-recapture experiment, from vectors of sample sizes and number of recaptures, with each element corresponding to each sampling stratum.

Usage

```
vstrat(n1, n2, m2, estimator = "Chapman")
```

Arguments

n1	Vector of individuals captured and marked in the first sample, from each stratum
n2	Vector of individuals captured and marked in the second sample, from each stratum
m2	Vector of marked individuals recaptured in the second sample, from each stratum
estimator	The type of estimator to use. Allowed values are "Chapman", "Petersen", and "Bailey". Default to "Chapman".

Value

The estimated variance of the stratified estimator

Note

It is possible that even the stratified estimate of abundance may be biased if capture probabilities differ greatly between strata. However, the bias in the stratified estimator will be much less than an estimator calculated without stratification.

This function makes the naive assumption of independence between strata. Caution is therefore recommended.

Author(s)

Matt Tyers

See Also

strattest, Nstrat, rstrat, sestrat, cistrat, NChapman, NPetersen, NBailey

Examples

```
vstrat(n1=c(100,200), n2=c(100,500), m2=c(10,10))
```

Index