Package 'heteroplasmy'

August 17, 2022

Title Calculation of the standard error of the variance for heteroplasmy data

Version 0.0.2.0

Description The package offers different methods to quantify uncertainty by calculating the standard errros of the variance of given data. The included functions are primarily aimed to heteroplasmy data, which are not assumed to follow a predefined distribution and the sample size is usually low. Included, there is a set of synthetic datasets to use, as well as real heteroplasmy data from mouse specimen, found in (xxx Iain's paper xxx) The code and the methods described in the package are used in the (xxx our report xxx), so please cite that work in case you use the heteroplasmy package.

License `use_mit_license()`, `use_gpl3_license()` or friends to pick a license

Depends R (>= 3.1.0)

URL https://github.com/kostasgian21/heteroplasmy

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1.9001

Imports kimura, stats

Remotes lbozhilova/kimura

R topics documented:

`analyticVar` *Analytic calculation of the standard error of the variance*

Description

This function calculates analytically the standard error of the variance. It's based on the use of the appropriate h-statistic as an estimator, as default. It is offers a corrected version of the method described in Wonnapinij et al.. If instead the Wonnapinij et al. mehod shall be used, make method="Wonnapinij".

Usage

```
analyticVar(data, normal = FALSE, method = "hstatistic")
```

Arguments

`data`	The input data in the form of a vector. NA values are omitted.
`normal`	Parameter that indicates if the normal approximation should be used instead of the general formula from (Wilks, S. S. (1962).Mathematical Statistics). Default is FALSE.
`method`	What method to use for the estimation of the standard error of the variance. Accepted values are "hstatistic" (default) and "Wonnapinij".

Value

The analytically derived standard error of the variance of `data`.

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
analyticVar(data)

mouseData=readHeteroplasmyData("HB")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
analyticVar(mouseData1,method="hstatistic")

# use the package data and load it to variable mouseData
mouseData=mousedataLE
# calculate the standard error of the variance for the LE oocyte sample #3
bootstrapVar(mouseData[,3])
```

bootstrapVar *A boostap method to calculate the standard error of the variance*

Description

This function uses the bootrstap method to calculate the uncertainty of the variance of a given sample based on random resampling. The number of the resamples is a parameter (default is 1000). Given that the resampling methods underestimate the uncertainty and thus provide a biased estimation, we offer the the unbiased method as a default, although the user may change this option through the biased parameter for experimental purposes (they are strongly advised not to do for real problems with small samples).

Usage

```
bootstrapVar(data, nrep = 1000, biased = FALSE)
```

Arguments

data	The input data in the form of a vector. NA values are omitted.
nrep	The number of bootstrap resamples. Default is 1000. The higher the number of the samples, the better the bootstrap outcome.
biased	A logical parameter to indicate if the user wants the biased version. Resampling techniques always underestimate statistics like the variance or the standard error of it for small samples.

Value

The standard error of the variance of data.

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
bootstrapVar(data)

mouseData=readHeteroplasmyData("HB")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
bootstrapVar(mouseData1)

# use the package data and load it to variable mouseData
mouseData=mousedataLE
# calculate the standard error of the variance for the LE oocyte sample #3
bootstrapVar(mouseData[,3])
```

`estimate_parameters_ks` *Max likelihood estimation of Kimura parameters that minimize the KS statistic*

Description

Using maximum likelihood to estimate the parameters of a fitted Kimura distribution to the input sample values that minimizes the KS statistic. Used to showcase thath the use of KS statistic to prove selection needs caution.

Usage

```
estimate_parameters_ks(h)
```

Arguments

`h`	A vector containig heteroplasmy measurements. Every observation should be in `[0,1]`.

Value

The maximum likelihood estimates for a fitted Kimura distribution parameters that minimze the KS statistic of a KS test.

Author(s)

Kostas and Iain, <us@example.com>

References

Site or paper

See Also

`readHeteroplasmyData`

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
estimate_parameters_ml(data_ex)

mouseData=readHeteroplasmyData("LE")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
estimate_parameters_ks(mouseData1)
```

`estimate_parameters_ml` *Max likelihood estimation for Kimura distribution parameters*

Description

Using maximum likelihood to estimate the parameters of a fitted Kimura distribution to the input sample values.

Usage

```
estimate_parameters_ml(h)
```

Arguments

h	A vector containig heteroplasmy measurements. Every observation should be in `[0,1]`.

Value

The maximum likelihood estimates for a fitted Kimura distribution parameters.

Author(s)

Kostas and Iain, <us@example.com>

References

Site or paper

See Also

`readHeteroplasmyData`

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
estimate_parameters_ml(data_ex)

mouseData=readHeteroplasmyData("LE")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
estimate_parameters_ml(mouseData1)
```

`heteroplasmyShift` *Transformed heteroplasmy shift*

Description

A numerical transformation of the heteroplasmy samples in order to work with the heteroplasmy shifts across diverse samples (e.g., due to time or different tissue samples). This transformation is used for comparing a heteroplasmy observation h to a reference value h_0. It corresponds to the formula:

$$\Delta h = \ln\left(\frac{h(h_0 - 1)}{h_0(h - 1)}\right)$$

Usage

```
heteroplasmyShift(h, h0)
```

Arguments

`h`	The heteroplasmy observation. Can be either a single value or a vector of observations. Every observation should be in `[0,1]`.
`h0`	The reference heteroplasmy value. Should be in `[0,1]`.

Value

The Transformed heteroplasmy shift.

Author(s)

Kostas and Iain, <us@example.com>

References

Site or paper

See Also

`readHeteroplasmyData`

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
heteroplasmyShift(data_ex)

mouseData=readHeteroplasmyData("HB")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
heteroplasmyShift(mouseData1,nrep=10000)
```

hstats *calculate various statistics for a heteroplasmy set h*

Description

can enforce an initial h0 or leave as a free parameter. Can can use population or sample statistics. analyticVar offers a simplified version of this function to compute the standard error of the variance.

Usage

```
hstats(h, h0 = F, usepopn = F)
```

Arguments

h	A vector containig heteroplasmy measurements. Every observation should be in [0,1].
h0	Logical parameter. A particular h0 value Default is to treat h0 as a fit parameter
usepopn	Logical parameter. Use of population or sample statistics (T and F, respectively)

Value

The maximum likelihood for the input data according to the Kimura distribution (using bootstrapping)

Examples

```
 X.1 = rnorm(50,0.5,0.1)
hstats(X.1)
```

invtransfun *The inverse function to transfun*

Description

A transformation function to inveret the effect of transfun (ie, the inverse logit transform).

Usage

```
invtransfun(x)
```

Arguments

x	a real number in [0,1] to be transformed into a real.

Value

The inveresed treansformation of transfun.

Examples

```
invtransfun(0.71)
```

jackVar *A jackknife method to compute the uncertainty of heteroplasmy data*

Description

Similarly to the main bootstrapVar function that implements the bootstra method to measure the standard error of the variance, the jackknife technique is another resampling method that can be used for the same purpose. Unlike bootstrapVar, jackVar (and very jackknife method) is deterministic and doesn not rely on randomness, but instead it uses removals of the sample points, one each time to calculate different sub-samples of size (n-1). Note that the size of the input data should be strictly greater than 1.

Usage

```
jackVar(data)
```

Arguments

data	The input data in the form of a dataframe or matrix (which will be transformed into a dataframe). Its size should be >=2. NA values are omitted.

Value

The analytically derived standard error of the variance of data.

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
jackVar(data)

mouseData=readHeteroplasmyData("HB")
mouseData1 = mouseData[which(!is.na(mouseData[,1])),1]
jackVar(mouseData1)

# use the package data and load it to variable mouseData
mouseData=mousedataLE
# calculate the standard error of the variance for the LE oocyte sample #3
bootstrapVar(mouseData[,3])
## Not run:
#input data of size 1 will fail
data_ex=rnorm(1,0.5,0.1)
jackVar(data)

## End(Not run)
```

joint_neg_log_lik *Joint negative log likelihood function for several heteroplasmy measurements*

Description

joint negative log likelihood function for several families' heteroplasmy measurements theta = [b,h0.1,h0.2,...] (use h values if use.h0s=F, otherwise initial heteroplasmies are enforced via h0s).

Usage

```
joint_neg_log_lik(theta, hlist, use.h0s = F, h0s = -1)
```

Arguments

theta	Kimura parameters p (or h0 here) and b.
hlist	TBDD list of different sets of heteroplasmy measurements
use.h0s	Logical parameter. TBD
h0s	TBDD

Value

The negative log likelihood for the list of inputs.

Examples

```
 X.1 = rnorm(50,0.5,0.1)
joint_neg_log_lik(c(0.5,0.91),X.1)
```

kimura_lrt *Likelihood ratio between two heteroplasmy samples*

Description

A function to perform likelihood ratio test exploring difference in bottleneck size between two heteroplasmy samples

Usage

```
kimura_lrt(h1, h2, use.h0s = F, h1.h0set = 0, h2.h0set = 0)
```

Arguments

h1	The first heteroplasmy sample.
h2	The two heteroplasmy sample.
use.h0s	Logical parameter. TBD
h1.h0set	Logical parameter.TBD
h2.h0set	Logical parameter.TBD

Value

The maximum likelihood for the input data according to the Kimura distribution (using bootstrapping)

Examples

```
 X.1 = rnorm(50,0.5,0.1)
 X.2 = rnorm(50,0.5,0.1)
kimura_lrt(X.1,X.2)
```

`kimura_neg_loglik` *Calculate negative log likelihood for heteroplasmy measurements*

Description

calculate negative log likelihood for a given set of heteroplasmy measurements h and parameters theta = logit(p), logit(b) (we write h0 for p) we can do this enforcing a particular h0 value (passed as an argument) or treating h0 as a fit parameter (default) the logit transform is used to ensure h0 and b remain in the `[0,1]` interval regardless of what real-valued argument the numerical optimiser attempts

Usage

```
kimura_neg_loglik(theta, h, h0 = F)
```

Arguments

`theta`	Kimura parameters p (or h0 here) and b.
`h`	A vector containig heteroplasmy measurements. Every observation should be in `[0,1]`.
`h0`	Logical parameter. A particular h0 value Default is to treat h0 as a fit parameter

Value

The negative log likelihood for the input.

Examples

```
 X.1 = rnorm(50,0.5,0.1)
kimura_neg_loglik(c(0.5,0.91),X.1)
```

ks_dist *Kolmogorov-Smirnov distance function*

Description

A function to calculate the Kolmogorov-Smirnov distance. It is used in estimate_parameters_ks to estimate the parameter that minimize the distance in the optim function.

Usage

```
ks_dist(theta, ecdf)
```

Arguments

theta	A vector. with two elements. The two Kimura parameters h0 and b.
ecdf	A vector containing 10000 values from the empirical cumulative distribution function of the input heteroplasmy data vector. To be used in the optim function in estimate_parameters_ks.

Value

The maximum distance from the theoretical Kimura distribution.

Examples

```
h=rnorm(20,0.5,0.1)
ecdf_h <- (stats::ecdf(h))(seq(0, 1, 1e-04))
ks_dist(c(0.5,0.95),ecdf_h)
```

maxlik *compute maximum likelihood parameters and confidence intervals for heteroplasmy data*

Description

compute maximum likelihood parameters and confidence intervals for a given heteroplasmy set. We can do this while imposing a specific h0 as an argument or allowing a search over h0 values.

Usage

```
maxlik(h, conf.level = 0.95, h0 = F)
```

Arguments

h	A vector containig heteroplasmy measurements. Every observation should be in [0,1].
conf.level	The preferred confidence interval calculation, Default value is 0,95 (95%).
h0	Logical parameter. A particular h0 value Default is to treat h0 as a fit parameter

Value

The maximum likelihood for the input data according to the Kimura distribution

Examples

```
 X.1 = rnorm(50,0.5,0.1)
maxlik(X.1,conf.level=0.95)
```

`maxlikboot` *Bootstrap estimates for parameters and confidence intervals for heteroplasmy data*

Description

compute bootstrap estimates for parameters and confidence intervals for a given heteroplasmy set Wwe can do this while imposing a specific h0 as an argument or allowing a search over h0 values

Usage

```
maxlikboot(h, nboot = 1000, conf.level = 0.95, h0 = F)
```

Arguments

`h`	A vector containig heteroplasmy measurements. Every observation should be in `[0,1]`.
`nboot`	The number of bootstrap samples. Default value is 1000
`conf.level`	The preferred confidence interval calculation, Default value is 0,95 (95%).
`h0`	Logical parameter. A particular h0 value Default is to treat h0 as a fit parameter

Value

The maximum likelihood for the input data according to the Kimura distribution (using bootstrapping)

Examples

```
 X.1 = rnorm(50,0.5,0.1)
maxlik(X.1,nboot=10000,0.95)
```

mousedataFreyer *mouse PGCs heteroplasmy data*

Description

A dataset containing heteroplasmy values from mouse PGCs taken from Freyer et al. Each column corresponds to a different specimen. NA values have been added to make the number of rows equal. Please remove them after loading. Heteroplasmy data of the package are in the range `[0,1]`.(Add reference!)

Usage

```
mousedataFreyer
```

Format

A data frame with 111 rows and 18 columns (TO FIX)

Source

http://www.example.info/

mousedataHB *HB oocyte heteroplasmy data*

Description

A dataset containing heteroplasmy values for the HB oocyte mouse lines. Each column corresponds to a different specimen. NA values have been added to make the number of rows equal. Please remove them after loading. Heteroplasmy data of the package are in the range `[0,1]`.

Usage

```
mousedataHB
```

Format

A data frame with 25 rows and 56 columns (TO FIX):

Source

http://www.example.info/

mousedataLE *LE oocyte heteroplasmy data*

Description

A dataset containing heteroplasmy values for the HB oocyte mouse lines. Each column corresponds to a different specimen. NA values have been added to make the number of rows equal. Please remove them after loading. Heteroplasmy data of the package are in the range `[0,1]`.

Usage

```
mousedataLE
```

Format

A data frame with 20 rows and 43 columns (TO FIX)

Source

http://www.example.info/

plotStdErrVar *An example plotting function*

Description

This function is used as a toy example on how to represent the data statistics regarding the variance of the sample. The mean variance and its standard error are depicted (XX maybe plot 2*SEM? XX). Note that this is just an illustration to show that the analytic and the resampling approaches almost match each other.

Usage

```
plotStdErrVar(
  data,
  functions = c("normalApr", "analytic", "bootstrap", "jackknife"),
  ...
)
```

Arguments

`data`	The input data in the form of a vector. NA values are omitted.
`functions`	Choose the subset of the functions you wish use for the calculation and subsequent plot of the standard error of the variance. You can use one or a combination of "normalApr", "analytic","bootstrap","correctedBoot", and "jackknife". For now, it outputs all of the aforementioned methods!

Warning

This is a plotting function just for demonstration purposes.

Examples

```
# size of the sample
n=50
#generate a random sample of size n from a normal distribution
data_ex=rnorm(n,0.5,0.1)
plotStdErrVar(data_ex)
```

---

readHeteroplasmyData *A Function to read mouse heteroplasmy data (not finished!)*

Description

This function allows you to read mouse heteroplasmy data from external files. Use with caution (for now).

Usage

```
readHeteroplasmyData(nameD = "HB")
```

Arguments

nameD	Either "HB" or "LE" or "Freyer".

Value

A dataframe containing mouse heteroplasmy data.

Examples

```
readHeteroplasmyData(nameD="LE")
```

---

test_kimura_par *Generalised MC KS test for genetic drift*

Description

This function is a generalisation of the test_kimura function from the lbozhilova/kimura package. It corresponds to a Monte Carlo Kolmogorov-Smirnov test to detect genetic drift by examining deviation from a Kimura distribution. (add references)

Usage

```
test_kimura_par(h, p, b, num_MC = 1000, round = TRUE)
```

Arguments

h	A vector containig heteroplasmy measurements. Every observation should be in [0,1].
p	The p parameter of the Kimura distribution. Should be in [0,1].
b	The b parameter of of the Kimura distribution. Should be in [0,1].
num_MC	number of Monte Carlo runs
round	a logical argument. True if heteroplasmy fractions are rounded to two significant digits.

Value

object of class htest

Examples

```
data_ex=rnorm(n,0.5,0.1)
  fit = estimate_parameters_ml(data_ex)
  p=fit[1]
  b=fit[2]
test_kimura_par(data_ex,p,b)
```

---

transfun *A Function to cast real numbers to* [0,1]

Description

A transformation function to cast any real number onto the interval [0,1]. Equivalent to the inverse logit transform.

Usage

```
transfun(x)
```

Arguments

x	a real number to be transformed.

Value

The transformed cast of the input value to the interval [0,1].

Examples

```
transfun(5.1)
```

Index