

Package ‘heteroplasmy’

May 28, 2021

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

Version 0.0.0.9000

Description The package offers different methods to quantify uncertainty by calculating the standard errors 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. Includes, 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://example.com>

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1.9001

RemoteType github

RemoteHost api.github.com

RemoteRepo heteroplasmy

RemoteUsername kostasgian21

RemoteRef HEAD

RemoteSha be7d3c649185ec0db267d126a1b153be8d30dcb0

GithubRepo heteroplasmy

GithubUsername kostasgian21

GithubRef HEAD

GithubSHA1 be7d3c649185ec0db267d126a1b153be8d30dcb0

NeedsCompilation no

Author StochasticBio Group [aut, cre] (YOUR-ORCID-ID)

Maintainer StochasticBio Group <first.last@example.com>

R topics documented:

analyticVar	2
bootstrapVar	3
jackVar	4
plotStdErrVar	5
readHeteroplasmyData	5
Index	7

analyticVar	<i>Analytic calculation of the standard error of the variance</i>
-------------	---

Description

This function calculates analytically the standard error of the variance.

Usage

```
analyticVar(data, normal = FALSE)
```

Arguments

data	The input data in the form of a dataframe or matrix (which be transformed into a dataframe).
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.

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)
```

bootstrapVar

*A bootstrap method to calculate the standard error of the variance***Description**

This function uses the bootstrap 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) and along with the "vanilla" version, we offer an optimized variation (using the sigmaOpt parameter) which has been seen to improve the precision of the calculation (see our report/paper). Given that the resampling methods underestimate the uncertainty and thus provide a biased estimation, we offer 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,
  corrected = FALSE,
  sigmaOpt = FALSE
)
```

Arguments

data	The input data in the form of a dataframe or matrix (which be transformed into a dataframe).
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.
corrected	Simle correction with a factor of 2.61 that was experimentally found. It is included also in the case of sigmaOpt=TRUE
sigmaOpt	The outcome of the bootstrap resampling with a fitted sigmoid function $g(x)$ with four parameters. Derived through simulations on both real heteroplasmy data and various synthetic ones. Try the plotStdErrVar function in this package to observe it.

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,sigmaOpt=TRUE)
```

jackVar

A jackknife method to compute the uncertainty of heteroplasmy data

Description

Similarly to the main bootstrapVar function that implements the bootstrap 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't 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 be transformed into a dataframe). Its size should be ≥ 2
------	--

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)
```

plotStdErrVar	<i>An example plotting function</i>
---------------	-------------------------------------

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. 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", "correctedBoot", "jackknife"),
  ...
)
```

Arguments

data	The input data in the form of a dataframe or matrix (which be transformed into a dataframe).
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!

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	<i>A Function to read mouse heteroplasmy data</i>
----------------------	---

Description

This function allows you to read mouse heteroplasmy data from external files.

Usage

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

Arguments

nameD Either "HB" or "LE".

Examples

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

Index

- * **boot-**
 - strap,uncertainty,heteroplasmy,resampling**
 - bootstrapVar, [3](#)
- * **data**
 - readHeteroplasmyData, [5](#)
- * **eror**
 - plotStdErrVar, [5](#)
- * **error**
 - analyticVar, [2](#)
- * **heteroplasmy**
 - analyticVar, [2](#)
 - readHeteroplasmyData, [5](#)
- * **jack-**
 - knife,uncertainty,heteroplasmy,resampling**
 - jackVar, [4](#)
- * **of**
 - analyticVar, [2](#)
- * **plot**
 - plotStdErrVar, [5](#)
- * **standard**
 - analyticVar, [2](#)
 - plotStdErrVar, [5](#)
- * **the**
 - analyticVar, [2](#)
- * **variance,**
 - analyticVar, [2](#)

analyticVar, [2](#)

bootstrapVar, [3](#)

jackVar, [4](#)

plotStdErrVar, [5](#)

readHeteroplasmyData, [5](#)