Package 'heteroplasmy'

May 31, 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 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://exmple.com>

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1.9001

R topics documented:

---

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

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 will be transformed
                into a dataframe). 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.

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)
```

---

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 sam-
ple based on random resampling. The number of the resamples is a parameter (default is 1000) and
along with the the "vanilla" version, we offer an optimized variation (using the **sigmaOpt** parame-
ter) 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 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 will be transformed into a dataframe). 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.
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.

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,sigmaOpt=TRUE)
```

---

bootstrapVarCor	*The bootstrapVar function with default correction*

Description

This function is simply the bootstrapVar with the correction argument being TRUE. it is provided as a seperate function for usability.

Usage

```
bootstrapVarCor(data, nrep = 1000)
```

Arguments

data
: The input data in the form of a dataframe or matrix (which will be transformed into a dataframe). 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 (see [par](#)).

Value

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

Author(s)

Kostas and Iain, <us@example.com>

References

<https://en.wikipedia.org/>

See Also

[bootstrapVar](#)

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)
bootstrapVarCor(data)

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

---

heteroplsamyShift   *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

```
heteroplsamyShift(h, h0)
```

Arguments

h	The heteroplasmy observation.
h0	The reference heteroplasmy value.

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)
heteroplsamyShift(data)

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

---

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)
## Not run:
#input data of size 1 will fail
data_ex=rnorm(1,0.5,0.1)
jackVar(data)

## End(Not run)
```

---

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. 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 will 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!

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".

Value

A dataframe containing mouse heteroplasmy data.

Examples

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

Index