Package ‘betaNB’

April 8, 2023

Title Nonparametric Bootstrap Confidence Intervals for Standardized Regression Coefficients and Other Effect Sizes

Version 1.2.0.9000

Description
Generates nonparametric bootstrap confidence intervals for standardized regression coefficients and other effect sizes for models fitted by lm().

URL https://github.com/jeksterslab/betaNB,
https://jeksterslab.github.io/betaNB/

BugReports https://github.com/jeksterslab/betaNB/issues

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

Depends R (>= 3.5.0)

Imports stats, methods

Suggests knitr, rmarkdown, testthat

RoxygenNote 7.2.3

NeedsCompilation no

Author Ivan Jacob Agaloos Pesigan [aut, cre, cph]
(<https://orcid.org/0000-0003-4818-8420>)

Maintainer Ivan Jacob Agaloos Pesigan <r.jeksterslab@gmail.com>

R topics documented:

`BetaNB` *Estimate Standardized Regression Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Standardized Regression Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
BetaNB(object)
```

Arguments

`object` Object of class `mc`, that is, the output of the `MC()` function.

Details

The vector of standardized regression coefficients ($\hat{\boldsymbol{\beta}}$) is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1-\alpha)\%$ from the generated sampling distribution of $\hat{\boldsymbol{\beta}}$, where α is the significance level.

Value

Returns an object of class `betanb` which is a list with the following elements:

call Function call.

object The function argument `object`.

thetahatstar Sampling distribution of $\hat{\boldsymbol{\beta}}$.

jackknife Jackknife estimates.

est Vector of estimated $\hat{\boldsymbol{\beta}}$.

fun Function used ("BetaNB").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: .CI(), DeltaRSqNB(), DiffBetaNB(), NB(), PCorNB(), RSqNB(), SCorNB()

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Methods --------------------------------------------------------
print(std)
summary(std)
coef(std)
vcov(std)
confint(std, level = 0.95)
```

---

coef.betanb *Estimated Parameter Method for an Object of Class* betanb

---

Description

Estimated Parameter Method for an Object of Class betanb

Usage

```
## S3 method for class 'betanb'
coef(object, ...)
```

Arguments

object	Object of Class betanb, that is, the output of the BetaNB(), RSqNB(), SCorNB(), DeltaRSqNB(), or PCorNB() functions.
...	additional arguments.

Value

Returns a vector of estimated parameters.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Method ---------------------------------------------------------
coef(std)
```

---

confint.betanb *Confidence Intervals Method for an Object of Class* `betanb`

Description

Confidence Intervals Method for an Object of Class `betanb`

Usage

```
## S3 method for class 'betanb'
confint(object, parm = NULL, level = 0.95, type = "pc", ...)
```

Arguments

`object`	Object of Class `betanb`, that is, the output of the `BetaNB()`, `RSqNB()`, `SCorNB()`, `DeltaRSqNB()`, or `PCorNB()` functions.
`parm`	a specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. If missing, all parameters are considered.
`level`	the confidence level required.
`type`	Charater string. Confidence interval type, that is, `type = "pc"` for percentile; `type = "bc"` for bias corrected; `type = "bca"` for bias corrected and accelerated.
`...`	additional arguments.

Value

Returns a matrix of confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Method ------------------------------------------------------
confint(std, level = 0.95, type = "pc")
confint(std, level = 0.95, type = "bc")
confint(std, level = 0.95, type = "bca")
```

`DeltaRSqNB` *Estimate Improvement in R-Squared and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Improvement in R-Squared and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
DeltaRSqNB(object)
```

Arguments

`object`	Object of class `mc`, that is, the output of the `MC()` function.

Details

The vector of improvement in R-squared (ΔR^2) is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1 - \alpha)\%$ from the generated sampling distribution of ΔR^2, where α is the significance level.

Value

Returns an object of class `betanb` which is a list with the following elements:

call Function call.

object The function argument `object`.

thetahatstar Sampling distribution of ΔR^2.

vcov Sampling variance-covariance matrix of ΔR^2.

est Vector of estimated ΔR^2.

fun Function used ("DeltaRSqMC").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: `.CI()`, `BetaNB()`, `DiffBetaNB()`, `NB()`, `PCorNB()`, `RSqNB()`, `SCorNB()`

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for improvement in R-squared
deltarsq <- DeltaRSqNB(nb)
# Methods -------------------------------------------------------
print(deltarsq)
summary(deltarsq)
coef(deltarsq)
vcov(deltarsq)
confint(deltarsq, level = 0.95)
```

`DiffBetaNB` *Estimate Differences of Standardized Slopes and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Differences of Standardized Slopes and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
DiffBetaNB(object)
```

Arguments

`object`	Object of class `mc`, that is, the output of the `MC()` function.

Details

The vector of differences of standardized regression slopes is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1 - \alpha)\%$ from the generated sampling distribution of differences of standardized regression slopes, where α is the significance level.

Value

Returns an object of class `betanb` which is a list with the following elements:

call Function call.

object The function argument `object`.

thetahatstar Sampling distribution of differences of standardized regression slopes.

vcov Sampling variance-covariance matrix of differences of standardized regression slopes.

est Vector of estimated differences of standardized regression slopes.

fun Function used ("DiffBetaMC").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: `.CI()`, `BetaNB()`, `DeltaRSqNB()`, `NB()`, `PCorNB()`, `RSqNB()`, `SCorNB()`

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals
# for differences of standardized regression slopes
diff <- DiffBetaNB(nb)
# Methods --------------------------------------------------------
print(diff)
summary(diff)
coef(diff)
vcov(diff)
confint(diff, level = 0.95)
```

`nas1982` *1982 National Academy of Sciences Doctoral Programs Data*

Description

1982 National Academy of Sciences Doctoral Programs Data

Usage

```
nas1982
```

Format

Ratings of 46 doctoral programs in psychology in the USA with the following variables:

QUALITY Program quality ratings.

NFACUL Number of faculty members in the program.

NGRADS Number of program graduates.

PCTSUPP Percentage of program graduates who received support.

PCTGRT Percent of faculty members holding research grants.

NARTIC Number of published articles attributed to program faculty member.

PCTPUB Percent of faculty with one or more published article.

References

National Research Council. (1982). *An assessment of research-doctorate programs in the United States: Social and behavioral sciences*. doi:10.17226/9781. Reproduced with permission from the National Academy of Sciences, Courtesy of the National Academies Press, Washington, D.C.

`NB` *Generate the Sampling Distribution of Sample Covariances Using Nonparametric Bootstrapping*

Description

Generate the Sampling Distribution of Sample Covariances Using Nonparametric Bootstrapping

Usage

```
NB(object, R = 5000, seed = NULL)
```

Arguments

`object`	Object of class `lm`.
`R`	Positive integer. Number of bootstrap replications.
`seed`	Integer. Seed number for reproducibility.

Value

Returns an object of class `nb` which is a list with the following elements:

call Function call.

args Function arguments.

lm_process Processed `lm` object.

thetahatstar Sampling distribution of sample covariances.

jackknife Jackknife estimates.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Efron, B., & Tibshirani, R. J. (1993) *An introduction to the bootstrap*. Chapman & Hall.

See Also

Other Beta Nonparametric Bootstrap Functions: `.CI()`, `BetaNB()`, `DeltaRSqNB()`, `DiffBetaNB()`, `PCorNB()`, `RSqNB()`, `SCorNB()`

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
NB(object, R = 100)
```

`PCorNB` *Estimate Squared Partial Correlation Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Squared Partial Correlation Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
PCorNB(object)
```

Arguments

`object`	Object of class `mc`, that is, the output of the `MC()` function.

Details

The vector of squared partial correlation coefficients (r^2_p) is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1 - \alpha)\%$ from the generated sampling distribution of r^2_p, where α is the significance level.

Value

Returns an object of class betanb which is a list with the following elements:

call Function call.

object The function argument object.

thetahatstar Sampling distribution of r^2_p.

vcov Sampling variance-covariance matrix of r^2_p.

est Vector of estimated r^2_p.

fun Function used ("PCorMC").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: .CI(), BetaNB(), DeltaRSqNB(), DiffBetaNB(), NB(), RSqNB(), SCorNB()

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
rp <- PCorNB(nb)
# Methods --------------------------------------------------------
print(rp)
summary(rp)
coef(rp)
vcov(rp)
confint(rp, level = 0.95)
```

---

print.betanb *Print Method for an Object of Class* betanb

Description

Print Method for an Object of Class betanb

Usage

```
## S3 method for class 'betanb'
print(x, alpha = c(0.05, 0.01, 0.001), type = "pc", digits = 4, ...)
```

Arguments

x	Object of Class `betanb`, that is, the output of the `BetaNB()`, `RSqNB()`, `SCorNB()`, `DeltaRSqNB()`, or `PCorNB()` functions.
alpha	Significance level.
type	Charater string. Confidence interval type, that is, `type = "pc"` for percentile; `type = "bc"` for bias corrected; `type = "bca"` for bias corrected and accelerated.
digits	Digits to print.
...	additional arguments.

Value

Prints a matrix of estimates, standard errors, number of bootstrap replications, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Method ----------------------------------------------------------
print(std, type = "pc")
print(std, type = "bc")
print(std, type = "bca")
```

---

print.nb *Print Method for an Object of Class* `nb`

Description

Print Method for an Object of Class `nb`

Usage

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

Arguments

x	Object of Class `nb`.
...	additional arguments.

Value

Prints the first six bootstrap covariance matrices.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
print(nb)
```

RSqNB — *Estimate Multiple Correlation Coefficients (R-Squared and Adjusted R-Squared) and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Multiple Correlation Coefficients (R-Squared and Adjusted R-Squared) and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
RSqNB(object)
```

Arguments

`object`	Object of class `mc`, that is, the output of the `MC()` function.

Details

R-squared (R^2) and adjusted R-squared ($\bar{R}^2$) is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1 - \alpha)\%$ from the generated sampling distribution of R^2 and $\bar{R}^2$, where α is the significance level.

Value

Returns an object of class `betanb` which is a list with the following elements:

call Function call.

object The function argument `object`.

thetahatstar Sampling distribution of R^2 and $\bar{R}^2$.

vcov Sampling variance-covariance matrix of R^2 and $\bar{R}^2$.

est Vector of estimated R^2 and $\bar{R}^2$.

fun Function used ("RSqMC").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: `.CI()`, `BetaNB()`, `DeltaRSqNB()`, `DiffBetaNB()`, `NB()`, `PCorNB()`, `SCorNB()`

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
rsq <- RSqNB(nb)
# Methods -------------------------------------------------------
print(rsq)
summary(rsq)
coef(rsq)
vcov(rsq)
confint(rsq, level = 0.95)
```

`SCorNB` *Estimate Semipartial Correlation Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping*

Description

Estimate Semipartial Correlation Coefficients and Generate the Corresponding Sampling Distribution Using Nonparametric Bootstrapping

Usage

```
SCorNB(object)
```

Arguments

`object`	Object of class `mc`, that is, the output of the `MC()` function.

Details

The vector of semipartial correlation coefficients (r_s) is estimated from bootstrap samples. Confidence intervals are generated by obtaining percentiles corresponding to $100(1 - \alpha)\%$ from the generated sampling distribution of r_s, where α is the significance level.

Value

Returns an object of class `betanb` which is a list with the following elements:

call Function call.

object The function argument `object`.

thetahatstar Sampling distribution of r_s.

vcov Sampling variance-covariance matrix of r_s.

est Vector of estimated r_s.

fun Function used ("SCorMC").

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Beta Nonparametric Bootstrap Functions: `.CI()`, `BetaNB()`, `DeltaRSqNB()`, `DiffBetaNB()`, `NB()`, `PCorNB()`, `RSqNB()`

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate the sampling distribution of sample covariances
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
rs <- SCorNB(nb)
# Methods --------------------------------------------------------
print(rs)
summary(rs)
coef(rs)
vcov(rs)
confint(rs, level = 0.95)
```

`summary.betanb` *Summary Method for an Object of Class* `betanb`

Description

Summary Method for an Object of Class `betanb`

Usage

```
## S3 method for class 'betanb'
summary(object, alpha = c(0.05, 0.01, 0.001), type = "pc", digits = 4, ...)
```

Arguments

`object`	Object of Class `betanb`, that is, the output of the `BetaNB()`, `RSqNB()`, `SCorNB()`, `DeltaRSqNB()`, or `PCorNB()` functions.
`alpha`	Significance level.
`type`	Charater string. Confidence interval type, that is, `type = "pc"` for percentile; `type = "bc"` for bias corrected; `type = "bca"` for bias corrected and accelerated.
`digits`	Digits to print.
`...`	additional arguments.

Value

Returns a matrix of estimates, standard errors, number of bootstrap replications, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Method ----------------------------------------------------------
summary(std, type = "pc")
summary(std, type = "bc")
summary(std, type = "bca")
```

vcov.betanb *Sampling Variance-Covariance Matrix Method for an Object of Class* betanb

Description

Sampling Variance-Covariance Matrix Method for an Object of Class betanb

Usage

```
## S3 method for class 'betanb'
vcov(object, ...)
```

Arguments

object	Object of Class betanb, that is, the output of the BetaNB(), RSqNB(), SCorNB(), DeltaRSqNB(), or PCorNB() functions.
...	additional arguments.

Value

Returns the variance-covariance matrix of estimates.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# Fit the regression model
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
# Generate bootstrap covariance matrices
# (use a large R, for example, R = 5000 for actual research)
nb <- NB(object, R = 50)
# Generate confidence intervals for standardized regression slopes
std <- BetaNB(nb)
# Method -----------------------------------------------------------
vcov(std)
```

Index