Package 'ClusterBootstrap'

April 24, 2017

Title Analyze Clustered Data with Generalized Linear Models using the Cluster Bootstrap

Date 2017-04-24

Version 0.9.1.1-10

Description The ClusterBootstrap package provides functionality for the analysis of clustered data using the cluster bootstrap.

Depends R (>= 3.0), stats, utils, graphics, parallel

License GPL-3 | file LICENSE

URL https://github.com/mathijsdeen/ClusterBootstrap

BugReport https://github.com/mathijsdeen/ClusterBootstrap/issues

LazyData true

RoxygenNote 6.0.1

Author Mathijs Deen [aut, cre],
Mark de Rooij [aut]

Maintainer Mathijs Deen <m.l.deen@fsw.leidenuniv.nl>

R topics documented:

`clusbootglm` *Fit generalized linear models with the cluster bootstrap*

Description

Fit a generalized linear model with the cluster bootstrap for analysis of clustered data.

Usage

```
clusbootglm(model, data, clusterid, family = gaussian, B = 5000,
  confint.level = 0.95, no_cores = 1)
```

Arguments

`model`	generalized linear model to be fitted with the cluster bootstrap.
`data`	dataframe that contains the data.
`clusterid`	variable in data that identifies the clusters.
`family`	error distribution to be used in the model, e.g. `gaussian` or `binomial`.
`B`	number of bootstrap samples.
`confint.level`	level of confidence interval.
`no_cores`	number of CPU cores to be used.

Details

Some useful methods for the obtained `clusbootglm` class object are `summary.clusbootglm`, `coef.clusbootglm`, `plot.clusbootglm`, `clusbootmatrix` and `clusbootsample`.

Value

`clusbootglm` produces an object of class `"clusbootglm"`, containing the following relevant components:

`coefficients`	A matrix of `B` rows, containing the parameter estimates for all bootstrap samples.
`bootstrap.matrix`	n*B matrix, of which each column represents a bootstrap sample; each value in a column represents a unit of `subjectid`.
`lm.coefs`	Parameter estimates from a single (generalized) linear model.
`boot.coefs`	Mean values of the paramater estimates, derived from the bootstrap coefficients.
`boot.sds`	Standard deviations of cluster bootstrap parameter estimates.
`ci.level`	User defined confidence interval level.
`percentile.interval`	Confidence interval based on percentiles, given the user defined confidence interval level.
`parametric.interval`	Confidence interval based on `lm.coefs` and column standard deviations of `coefficients`, given the user defined confidence interval level.
`BCa.interval`	Confidence interval based on percentiles with bias correction and acceleration, given the user defined confidence interval level.

`samples.with.NA.coef`
: Cluster bootstrap sample numbers with at least one coefficient being `NA`.

`failed.bootstrap.samples`
: For each of the coefficients, the number of failed bootstrap samples are given.

Author(s)

Mathijs Deen, Mark de Rooij

Examples

```
## Not run:
data(opposites)
clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
## End(Not run)
```

---

clusbootmatrix *Extract matrix with bootstrap samples*

Description

Obtain a matrix containing the `clusterid` values for the bootstrap samples in a `clusbootglm` object.

Usage

```
clusbootmatrix(object, whichsample = "all")
```

Arguments

`object`	object of class `clusbootglm`, created with the `clusbootglm` function.
`whichsample`	input which bootstrap sample(s) should be returned. Choose "all" for the complete bootstrap matrix, "failed" for bootstrap samples that returned NAs, or a vector of values for specific bootstrap samples.

Author(s)

Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
clusbootmatrix(cbglm.1, whichsample=c(1:5))
## End(Not run)
```

clusbootsample *Return data for specified bootstrap sample*

Description

Returns the full data frame for a specified bootstrap sample in a clusbootglm object.

Usage

```
clusbootsample(object, samplenr)
```

Arguments

object	object of class clusbootglm, created with the clusbootglm function.
samplenr	sample number for which the data frame should be returned.

Author(s)

Mark de Rooij, Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
clusbootsample(cbglm.1, samplenr=1)
## End(Not run)
```

coef.clusbootglm *Obtain coefficients from cluster bootstrap object*

Description

Returns the coefficients of an object of class clusbootglm.

Usage

```
## S3 method for class 'clusbootglm'
coef(object, type = "bootstrap", ...)
```

Arguments

object	object of class clusbootglm.
type	type of coefficient (bootstrap or GLM).
...	other arguments.

Author(s)

Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
coef(cbglm.1, type="bootstrap")
## End(Not run)
```

`confint.clusbootglm` *Confidence intervals for cluster bootstrap model parameters*

Description

Computes confidence intervals for one or more parameters in a fitted GLM with the cluster bootstrap.

Usage

```
## S3 method for class 'clusbootglm'
confint(object, parm = "all", level = 0.95,
  type = "percentile", ...)
```

Arguments

`object`	object of class `clusbootglm`.
`parm`	a specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. Defaults to all parameters.
`level`	the required confidence level
`type`	type of confidence level. Options are `percentile`, `parametric` and `BCa`.
`...`	other arguments.

Author(s)

Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
confint(cbglm.1,parm=c("Time","COG"), level=.90, type="BCa")
## End(Not run)
```

opposites *Opposites naming data*

Description

The opposites dataframe consists of 144 observations within 36 individuals that completed an inventory that assesses their performance on a timed cognitive task called "opposites naming".

The dataset does not contain the empirical data within 35 individuals from the experiment by Willett (1988), but a simulation based on the multilevel model from Singer & Willett (2003) within 36 individuals.

Usage

```
opposites
```

Format

the following variables are available:

- Subject: subject indicator
- Time: a time variable, ranging 0-3
- COG: cognitive skill, measured once (at time=0)
- SCORE: score on opposites naming task

References

- Willett, J.B. (1988). Questions and answers in the measurement of change. In: E. Rothkopf (Ed.), *Review of research in education (1988-89)* (pp. 345-422). Washington, DC: American Educational Research Association.
- Singer, J.D., & Willett, J.B. (2003). *Applied longitudinal data analysis. Modeling change and event occurence.* NY: Oxford University Press, Inc.

plot.clusbootglm *Plot estimates and confidence intervals of cluster bootstrap GLM*

Description

Plots the estimates and their confidence intervals for an object of class clusbootglm.

Usage

```
## S3 method for class 'clusbootglm'
plot(x, interval.type = "percentile",
  show.intercept = FALSE, ...)
```

Arguments

x	object of class clusbootglm.
interval.type	which confidence interval should be used. Choose par for parametric, per for percentile, or BCa for BCa interval.
show.intercept	plot estimate and confidence interval of the intercept.
...	other arguments.

Author(s)

Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
plot(cbglm.1,interval.type="BCa")
## End(Not run)
```

---

summary.clusbootglm *Summarize output of cluster bootstrap GLM*

Description

Returns the summary of an object of class clusbootglm.

Usage

```
## S3 method for class 'clusbootglm'
summary(object, interval.type = "BCa", ...)
```

Arguments

object	object of class clusbootglm.
interval.type	which confidence interval should be used. Options are parametric, percentile and BCa intervals.
...	other arguments.

Author(s)

Mathijs Deen

Examples

```
## Not run:
data(opposites)
cbglm.1 <- clusbootglm(SCORE~Time*COG,data=opposites,clusterid=opposites$Subject)
summary(cbglm.1, interval.type="percentile")
## End(Not run)
```

Index