Package ‘complmrob’

April 14, 2025

Type Package

Title Robust Linear Regression with Compositional Data as Covariates

Version 0.7.1

Date 2025-04-14

URL https://github.com/dakep/complmrob

Description Robust regression methods for compositional data.
The distribution of the estimates can be approximated with various bootstrap methods. These bootstrap methods are available for the compositional as well as for standard robust regression estimates. This allows for direct comparison between them.

License GPL (>= 2)

Imports robustbase, ggplot2, boot, parallel, scales

RoxygenNote 7.3.2

Encoding UTF-8

NeedsCompilation no

Author David Kepplinger [aut, cre]

Maintainer David Kepplinger <david.kepplinger@gmail.com>

Repository CRAN

Date/Publication 2025-04-14 16:30:06 UTC

Contents

bootcoefs *Bootstrap the regression coefficients for a robust linear regression model*

Description

This function provides an easy interface and useful output to bootstrapping the regression coefficients of robust linear regression models

Usage

```
bootcoefs(
  object,
  R = 999,
  method = c("frb", "residuals", "cases"),
  ncpus = NULL,
  cl = NULL,
  ...
)

## S3 method for class 'complmrob'
bootcoefs(
  object,
  R = 999,
  method = c("frb", "residuals", "cases"),
  ncpus = NULL,
  cl = NULL,
  ...
)

## S3 method for class 'lmrob'
bootcoefs(
  object,
  R = 999,
  method = c("frb", "residuals", "cases"),
  ncpus = NULL,
  cl = NULL,
  ...
)
```

Arguments

`object` the model to bootstrap the coefficients from

`R` the number of bootstrap replicates.

`method` one of `"frb"` for fast and robust bootstrap, `"residuals"` to resample the residuals or `"cases"` to resample the cases.

ncpus	the number of CPUs to utilize for bootstrapping.
cl	a snow or parallel cluster to use for bootstrapping.
...	currently ignored.

Details

If ‘object‘ is created by ‘complmrob‘ the default method is to use fast and robust bootstrap (FRB) as described in the paper by M. Salibian-Barrera, et al (2008). The same default is used if ‘object‘ is an MM-estimate created by ‘lmrob(..., method = ’SM’)‘. The other options are to bootstrap the residuals or to bootstrap cases (observations), but the sampling distribution of the estimates from these methods can be numerically unstable and take longer to compute. If the ‘object‘ is a robust estimate created by ‘lmrob‘, but not an MM-estimate, the default is to bootstrap the residuals.

Value

A list of type `bootcoefs` for which `print`, `summary` and `plot` methods are available

Methods (by class)

- `bootcoefs(complmrob)`: For robust linear regression models with compositional data
- `bootcoefs(lmrob)`: For standard robust linear regression models

References

M. Salibian-Barrera, S. Aelst, and G. Willems. Fast and robust bootstrap. Statistical Methods and Applications, 17(1):41-71, 2008.

Examples

```
data <- data.frame(lifeExp = state.x77[, "Life Exp"], USArrests[ , -3])
mUSArr <- complmrob(lifeExp ~ ., data = data)
bc <- bootcoefs(mUSArr, R = 200) # the number of bootstrap replicates should
                                 # normally be higher!
summary(bc)
plot(bc) # for the model diagnostic plots
```

bootStat-methods *Bootstrap statistics functions*

Description

Functions to calculate the coefficient(s) of the robust linear regression model from a bootstrapped sample

Usage

```
bootStatResiduals(
  residData,
  inds,
  coefind,
  intercept = TRUE,
  maxTries = 4L,
  control
)

bootStatCases(origData, inds, coefind, formula, maxTries = 4L, control)

bootStatFastControl(model)

bootStatFast(origData, inds, control, coefind)
```

Arguments

`residData` the original data set with the columns fit, resid and the predictor variables instead of the response variable.

`inds` the resampled indices.

`coefind` the index of the coefficient to extract.

`intercept` if the model includes an intercept term.

`maxTries` the maximum number of tries to increase the maxit control arguments for the S estimator.

`control` either the control object as returned by `bootStatFastControl` (for ‘bootStatFast‘) or the control object used to fit the model(s) with ‘lmrob‘.

`origData` the original data set.

`formula` the formula to fit the model

`model` The lmrob model

Details

Different approaches for bootstrapping have been implemented. The default "fast and robust bootstrap" (FRB) proposed by M. Salibian-Barrera, et al. (2002), implemented with `bootStatFast` is the fastest and most resistant to outliers, while the other two `bootStatResiduals` and `bootStatCases` are standard bootstrap methods, where the residuals resp. the cases are resampled and the model is fit to this data.

References

M. Salibian-Barrera, S. Aelst, and G. Willems. Fast and robust bootstrap. Statistical Methods and Applications, 17(1):41-71, 2008.

See Also

`bootcoefs`

complmrob *MM-type estimators for linear regression on compositional data*

Description

Uses the lmrob method for robust linear regression models to fit linear regression models to compositional data.

Usage

```
complmrob(formula, data)
```

Arguments

formula	The formula for the regression model
data	The data.frame to use

Details

The variables on the right-hand-side of the formula are transformed with the isometric log-ratio transformation (isomLR) and a robust linear regression model is fit to those transformed variables. The orthonormal basis can be constructed in p different ways, where p is the number of variables on the RHS of the formula.

To get an interpretable estimate of the regression coefficient for each part of the composition, the data is transformed separately for each part. To estimate the coefficient for the *k*-th part, the *k*-th part is used as the orthonormal basis in the transformation and a regression model is fit to this data.

Value

A list of type complmrob with fields

coefficients the estimated coefficients

models the single regression models (one for each orthonormal basis)

npred the number of predictor variables

predictors the names of the predictor variables

coefind the index of the relevant coefficient in the single regression models

call how the function was called

intercept if an intercept is included

References

K. Hron, P. Filzmoser & K. Thompson (2012): Linear regression with compositional explanatory variables, Journal of Applied Statistics, DOI:10.1080/02664763.2011.644268

Examples

```
crimes <- data.frame(lifeExp = state.x77[, "Life Exp"],
                     USArrests[ , c("Murder", "Assault", "Rape")])
mUSArr <- complmrob(lifeExp ~ ., data = crimes)
summary(mUSArr)
```

confint.bccomplmrob *Calculate confidence intervals*

Description

Calculate confidence intervals for bootstrapped robust linear regression estimates with or without compositional data

Usage

```
## S3 method for class 'bccomplmrob'
confint(
  object,
  parm,
  level = 0.95,
  type = c("bca", "perc", "norm", "basic", "stud"),
  ...
)

## S3 method for class 'bclmrob'
confint(
  object,
  parm,
  level = 0.95,
  type = c("bca", "perc", "norm", "basic", "stud"),
  ...
)
```

Arguments

`object`	an object returned from `bootcoefs`.
`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`	the type of interval required (see the type argument of `boot::boot.ci()`).
`...`	currently ignored.

Methods (by class)

- `confint(bccomplmrob)`: for bootstrapped estimates of robust linear regression models for compositional data
- `confint(bclmrob)`: for bootstrapped estimates of robust linear regression models

Examples

```
data <- data.frame(lifeExp = state.x77[, "Life Exp"], USArrests[ , -3])
mUSArr <- complmrob(lifeExp ~ ., data = data)
bc <- bootcoefs(mUSArr, R = 200) # the number of bootstrap replicates should
                                 # normally be higher!
confint(bc, level = 0.95, type = "perc")

### For normal robust linear regression models ###
require(robustbase)
data(aircraft)

mod <- lmrob(Y ~ ., data = aircraft)
bootEst <- bootcoefs(mod, R = 200)
confint(bootEst, level = 0.95, type = "perc")
```

`isomLR` *(Inverse) Isometric log-ratio transformation for compositional data*

Description

Projects the D-dimensional compositional data on the (D-1)-dimensional simplex isometrically back and forth by transforming the values according to

$$z_i = \sqrt{\frac{D-i}{D-i+1}} \log \frac{x_i}{\left(\prod_{j=i+1}^{D} x_j\right)^{1/(D-i)}}$$

Usage

```
isomLR(x, comp = 1)

isomLRinv(z, perc = TRUE)
```

Arguments

`x` a numeric vector of length `D` or a numeric matrix with `D` columns

`comp` the component to use as the first compositional part

`z` a numeric vector of length `D-1` or a numeric matrix with `D-1` columns.

`perc` should the result be a matrix with percentage shares (default `TRUE`).

Value

isomLR: a numeric matrix with (D-1) columns with the transformed values. The name of the first column is the name of the first part (the other names are according to the order of the columns in the given matrix x)

isomLRinv: a numeric matrix with D columns with the transformed values. The values in the matrix are not on the original scale, but the percentage shares are equal.

Functions

- isomLRinv(): Inverse transformation

Examples

```
X <- as.matrix(USArrests[ , -3])
# Get the ilr with relative information of the 1st column to the other cols
ilrZ1 <- isomLR(X)
# Get the ilr with relative information of the 2nd column to the other cols
ilrZ2 <- isomLR(X, 2)
isomLRinv(ilrZ1)
```

plot.bootcoefs *Plot the distribution of the bootstrap estimates*

Description

Plot the distribution of the bootstrap estimates and the confidence intervals for the estimates

Usage

```
## S3 method for class 'bootcoefs'
plot(
  x,
  y = NULL,
  conf.level = 0.95,
  conf.type = "perc",
  kernel = "gaussian",
  adjust = 1,
  which = "all",
  theme = theme_bw(),
  confStyle = list(color = "#56B4E9", alpha = 0.4),
 estLineStyle = list(color = "black", width = rel(1), alpha = 1, linetype = "dashed"),
 densityStyle = list(color = "black", width = rel(0.5), alpha = 1, linetype = "solid"),
  ...
)
```

Arguments

`x` the object returned by a call to the `bootcoefs` function.

`y` ignored.

`conf.level` the level of the confidence interval that is plotted as shaded region under the density estimate.

`conf.type` the confidence interval type, see `boot.ci` for details.

`kernel` the kernel used for density estimation, see `density` for details.

`adjust` see `density` for details.

`which` which parameters to plot

`theme` the ggplot2 theme to use for the plot.

`confStyle` a list with style parameters for the confidence region below the density estimate (possible entries are `color`, and `alpha`)

`estLineStyle` a list with style parameters for the line at the original parameter estimate (possible entries are `color`, `width`, `alpha`, and `linetype`)

`densityStyle` a list with style parameters for the line of the density estimate (possible entries are `color`, `width`, `alpha`, and `linetype`)

`...` ignored

See Also

`confint` to get the numerical values for the confidence intervals

Examples

```
data <- data.frame(lifeExp = state.x77[, ”Life Exp”], USArrests[ , -3])
mUSArr <- complmrob(lifeExp ~ ., data = data)
bc <- bootcoefs(mUSArr, R = 200) # this can take some time
plot(bc) # for the model diagnostic plots
```

plot.complmrob *Diagnostic plots for the robust regression model with compositional covariates*

Description

Plot the response or the model diagnostic plots for robust linear regression model with compositional data

Usage

```
## S3 method for class 'complmrob'
plot(
  x,
  y = NULL,
  type = c("response", "model"),
  se = TRUE,
  conf.level = 0.95,
  scale = c("ilr", "percent"),
  theme = theme_bw(),
  pointStyle = list(color = "black", size = rel(1), alpha = 1, shape = 19),
  lineStyle = list(color = "grey20", width = rel(1), linetype = "solid"),
  seBandStyle = list(color = "gray80", alpha = 0.5),
  stack = c("horizontal", "vertical"),
  ...
)
```

Arguments

`x` the object returned by `complmrob`.

`y` ignored.

`type` one of `"response"` to plot the response or `"model"` to get the standard `lmrob` model diagnostic plots. Partial matching is performed, so any unique abbreviation of the two possible values is excepted (e.g., `"r"` for the response plot).

`se` should the confidence interval be shown in the response plot.

`conf.level` if the confidence interval is shown in the response plot, this parameter sets the level of the confidence interval.

`scale` should the x-axis in the response plot be in percentage or in the ILR-transformed scale?

`theme` the ggplot2 theme to use for the response plot.

`pointStyle` a list with style parameters for the points in the response plot (possible entries are `color`, `size`, `alpha`, and `shape`). If `color` and/or `shape` is a vector of length equal to the number of observations in the model, the points will be colored/shaped according to this vector.

`lineStyle` list with style parameters for the smoothing lines in the response plot (possible entries are `color`, `width`, and `linetype`)

`seBandStyle` a list with style parameters (`color` and `alpha`) for the confidence band (if `se` is `TRUE`)

`stack` how the facets are laid out in the response plot. `"horizontal"` for side by side and `"vertical"` for on top of each other.

`...` further arguments to the model diagnostic plot method (see `plot.lmrob` for details).

Details

The response plot shows the value on the first component of the orthonormal basis versus the response and the fitted values. For the fitted values, the other components are set to the median of the values in that direction. This usually causes aberrant predictions when plotting on the *percent* scale.

For the model diagnostic plots see the details in the help file for `plot.lmrob`. The model diagnostic plots are the same for all sub-models fit to the data transformed with the different orthonormal basis.

Examples

```
data <- data.frame(lifeExp = state.x77[, "Life Exp"], USArrests[ , -3])
mUSArr <- complmrob(lifeExp ~ ., data = data)
plot(mUSArr)
plot(mUSArr, type = "model") # for the model diagnostic plots
```

`print-methods` *Print the object*

Description

Print information about the models returned by `complmrob` or `bootcoefs`. For a detailed description see the help on `summary`.

Usage

```
## S3 method for class 'complmrob'
print(x, conf.level = 0.95, ...)

## S3 method for class 'bootcoefs'
print(x, conf.level = 0.95, conf.type = "perc", ...)
```

Arguments

`x`	the object to be printed.
`conf.level`	the confidence level for the confidence interval.
`...`	ignored.
`conf.type`	the type of the printed confidence interval.

See Also

`summary-methods`

print.summary.complmrob *Print the summary information*

Description

Print the summary information

Usage

```
## S3 method for class 'summary.complmrob'
print(
  x,
  digits = max(3, getOption("digits") - 3),
  signif.stars = getOption("show.signif.stars"),
  ...
)
```

Arguments

`x`	the summary object.
`digits`	the number of digits for the reported figures
`signif.stars`	should stars be displayed to show the significance of certain figures
`...`	further arguments currently not used

summary-methods *Get summary information*

Description

List the estimates, standard errors, p-values and confidence intervals for the coefficients of robust linear regression models with compositional data as returned by `complmrob` or `bootcoefs`

Usage

```
## S3 method for class 'complmrob'
summary(object, conf.level = 0.95, ...)

## S3 method for class 'bccomplmrob'
summary(object, conf.level = 0.95, conf.type = "perc", ...)

## S3 method for class 'bclmrob'
summary(object, conf.level = 0.95, conf.type = "perc", ...)
```

Arguments

`object` the object for which the summary information should be returned.

`conf.level` the level of the returned confidence intervals.

`...` ignored.

`conf.type` the type of the returned confidence interval (see `boot.ci` for the meaning of this parameter).

Index