

Package ‘complmrob’

August 17, 2014

Type Package

Title Robust linear regression with compositional data as covariates

Version 0.5.5

Date 2014-08-16

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

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

Description This package provides functionality to perform robust regression on compositional data. To get information on the distribution of the estimates, various bootstrapping methods are implemented for the compositional as well as for standard robust regression models, to provide a direct comparison between them.

License GPL (>= 2)

Imports robustbase, ggplot2, boot, parallel, scales

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-17 09:12:56

R topics documented:

bootcoefs	2
bootStat-methods	3
complmrob	4
confint.bccomplmrob	5
format.perc	7
isomLR	7
isomLRinv	8
plot.bootcoefs	8
plot.complmrob	9

predictdf.complmrob.part	11
print-methods	12
print.summary.complmrob	12
summary-methods	13

Index	14
--------------	-----------

bootcoefs	<i>Bootstrap the regression coefficients for a robust linear regression model</i>
-----------	---

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

The default method is to use fast and robust bootstrap as described in the paper by M. Salibián-Barrera, et al. (see references). 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 instable and take longer to compute.

Value

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

Methods (by class)

- `complmrob`: For robust linear regression models with compositional data
- `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

```
library(robCompositions)
data(expendituresEU)
data <- data.frame(
  y = as.numeric(apply(expendituresEU[, c("Food", "Alcohol", "Restaurants")], 1, sum)),
  expendituresEU[, c("Food", "Alcohol", "Restaurants")])

compModel <- complmrob(y ~ ., data = data)
bc <- bootcoefs(compModel, R = 500) # this can take some time
summary(bc)
```

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)
```

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

```
bootStatFastControl(model)
```

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

Arguments

origData	the original data set.
residData	the original data set with the columns fit, resid and the predictor variables instead of the response variable.
model	The lmrob model
inds	the resampled indices.
coefind	the index of the coefficient to extract.
control	the control object as returned by bootStatFastControl.
formula	the formula to fit the model
intercept	if the model includes an intercept term.
maxTries	the maximum number of tries to increase the maxit control arguments for the S estimator.

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 a 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 will be transformed with the isometric log-ratio transformation ([isomLR](#)) and then the robust linear regression model is applied 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 has to be transformed according to each of these orthonormal basis and a regression model has to be fit to every transformed data set.

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

```
library(robCompositions)
data(expendituresEU)
data <- data.frame(y = as.numeric(apply(expendituresEU , 1, sum)), expendituresEU)

compModel <- complmrob(y ~ ., data = data)
summary(compModel)
```

`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 <code>bootcoefs</code> .
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 <code>boot.ci</code>).
...	currently ignored.

Methods (by class)

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

Examples

```
library(robCompositions)
data(expendituresEU)
data <- data.frame(y = as.numeric(apply(expendituresEU , 1, sum)), expendituresEU)

compModel <- complmrob(y ~ ., data = data)
compBoot <- bootcoefs(compModel, R = 999) # this takes some time
confint(compBoot, level = 0.95, type = "perc")

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

mod <- lmrob(Y ~ ., data = aircraft)
bootEst <- bootcoefs(mod, R = 999) # this can take some time
confint(bootEst, level = 0.95, type = "perc")
```

format.perc	<i>Simple function (just copied from the stats package) to format percentages</i>
-------------	---

Description

Simple function (just copied from the stats package) to format percentages

Usage

```
## S3 method for class 'perc'
format(probs, digits)
```

Arguments

probs	the percentages
digits	the number of digits

isomLR	<i>Isometric log-ratio transformation for compositional data</i>
--------	--

Description

Projects the p -dimensional compositional data on the $(p-1)$ -dimensional simplex isometrically

Usage

```
isomLR(x, comp = 1)
```

Arguments

x	a numeric vector of length p or a numeric matrix with p columns
comp	the component to use as the first compositional part

Value

a numeric matrix with $(p-1)$ 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	<i>Inverse Isometric log-ratio transformation for compositional data</i>
-----------	--

Description

Projects the isometric log-ratio transformed (p-1) dimensional data back to the p-dimensional space

Usage

```
isomLRinv(x, perc = TRUE)
```

Arguments

x	a numeric vector of length p-1 or a numeric matrix with p-1 columns
perc	should the result be a matrix with percentual shares (default TRUE).

Value

a numeric matrix with p columns with the transformed values. The values in the matrix are not on the original scale, but the percentual shares are equal

plot.bootcoefs	<i>Plot the distribution of the bootstrap estimates</i>
----------------	---

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 = ggplot2::theme_bw(), confStyle = list(color = "#56B4E9", alpha =
0.4), estLineStyle = list(color = "black", width = ggplot2::rel(1), alpha =
1, linetype = "dashed"), densityStyle = list(color = "black", width =
ggplot2::rel(0.5), alpha = 1, linetype = "solid"), ...)
```


Arguments

x	the object returned by a call to the <code>bootcoefs</code> 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 <code>boot.ci</code> for details.
kernel	the kernel used for density estimation, see <code>density</code> for details.
adjust	see <code>density</code> 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

```
library(robCompositions)
data(expendituresEU)
data <- data.frame(
  y = as.numeric(apply(expendituresEU[, c("Food", "Alcohol", "Restaurants")], 1, sum)),
  expendituresEU[, c("Food", "Alcohol", "Restaurants")])

compModel <- complmrob(y ~ ., data = data)
bc <- bootcoefs(compModel, R = 500) # this can take some time
plot(bc) # for the model diagnostic plots
```

plot.complmrob	<i>Diagnostic plots for the robust regression model with compositional covariats</i>
----------------	--

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 = ggplot2::theme_bw(), pointStyle = list(color = "black", size =
ggplot2::rel(1), alpha = 1, shape = 19), lineStyle = list(color = "grey20",
width = ggplot2::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 may change in the future, as it is sub-optimal.

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

```
library(robCompositions)
data(expendituresEU)
data <- data.frame(
  y = as.numeric(apply(expendituresEU[, c("Food", "Alcohol", "Restaurants")], 1, sum)),
  expendituresEU[, c("Food", "Alcohol", "Restaurants")])

compModel <- complmrob(y ~ ., data = data)
plot(compModel) # for the response plot
plot(compModel, type = "model") # for the model diagnostic plots
```

```
predictdf.complmrob.part
```

Predict values for a complmrob.part object

Description

This function is used by `ggplot2` to predict the values for a `complmrob` model and should usually not be needed by the user.

Usage

```
predictdf.complmrob.part(model, xseq, se, level)
```

Arguments

<code>model</code>	the <code>complmrob.part</code> model the prediction should be done for
<code>xseq</code>	the sequence of x values to predict for
<code>se</code>	should the confidence interval be returned as well
<code>level</code>	the level of the confidence interval (if any)

Details

The sole reason that this function is visible is because the `ggplot2` function `predictdf` is not exported and thus this function could not be used for `complmrob.part` objects if it was not exported.

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

<code>x</code>	the object to be printed.
<code>conf.level</code>	the confidence level for the confidence interval.
<code>conf.type</code>	the type of the printed confidence interval.
<code>...</code>	ignored.

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	<i>Get summary information</i>
-----------------	--------------------------------

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

Index

`boot.ci`, [6](#), [9](#), [13](#)
`bootcoefs`, [2](#), [4](#), [6](#), [9](#), [12](#), [13](#)
`bootStat-methods`, [3](#)
`bootStatCases` (`bootStat-methods`), [3](#)
`bootStatFast` (`bootStat-methods`), [3](#)
`bootStatFastControl` (`bootStat-methods`),
[3](#)
`bootStatResiduals` (`bootStat-methods`), [3](#)

`complmrob`, [4](#), [10](#), [12](#), [13](#)
`confint`, [9](#)
`confint.bccomplmrob`, [5](#)
`confint.bclmrob` (`confint.bccomplmrob`), [5](#)

`density`, [9](#)

`format.perc`, [7](#)

`ggplot2`, [11](#)

`isomLR`, [5](#), [7](#)
`isomLRinv`, [8](#)

`lmrob`, [4](#), [10](#)

`plot`, [3](#)
`plot.bootcoefs`, [8](#)
`plot.complmrob`, [9](#)
`plot.lmrob`, [10](#)
`predictdf.complmrob.part`, [11](#)
`print`, [3](#)
`print-methods`, [12](#)
`print.bootcoefs` (`print-methods`), [12](#)
`print.complmrob` (`print-methods`), [12](#)
`print.summary.complmrob`, [12](#)

`summary`, [3](#), [12](#)
`summary-methods`, [13](#)
`summary.bccomplmrob` (`summary-methods`),
[13](#)
`summary.bclmrob` (`summary-methods`), [13](#)
`summary.complmrob` (`summary-methods`), [13](#)