Package ‘FRB’

February 19, 2015

Type Package
Title Fast and Robust Bootstrap
Version 1.8
Date 2013-04-15
Author Ella Roelant, Stefan Van Aelst, Gert Willems
Maintainer Stefan Van Aelst <Stefan.VanAelst@ugent.be>
Depends corpcor, rrcov (>= 1.3-01)
Description This package performs robust inference based on applying Fast and Robust Bootstrap on robust estimators. Available methods are multivariate regression, PCA and Hotelling tests.
License GPL-2
Repository CRAN
NeedsCompilation no
Date/Publication 2013-04-16 12:42:04

R topics documented:

  delivery .......................................................... 2
diagplot ............................................................. 3
ForgedBankNotes ..................................................... 5
FRB-internal ......................................................... 6
FRBhotellingMM ..................................................... 6
FRBhotellingS ....................................................... 10
FRBmultiregGS ....................................................... 13
FRBmultiregMM ....................................................... 17
FRBmultiregS ....................................................... 21
FRBpcaMM ........................................................... 25
FRBpcaS ............................................................. 29
GSboot_multireg ..................................................... 33
GSest_multireg ..................................................... 35
hemophilia .......................................................... 37
MMboot_loccov ....................................................... 38
Delivery Time Data, from Montgomery and Peck (1982). The aim is to explain the time required to service a vending machine (Y) by means of the number of products stocked (X1) and the distance walked by the route driver (X2).

Usage

data(delivery)

Format

A data frame with 25 observations on the following 3 variables.

  n.prod  Number of Products
  distance  Distance
  delTime  Delivery time

Source


References

Examples

data(delivery)

Description

Diagnostic plots for objects of class FRBmultireg, FRBpca and FRBhot. It shows robust distances and allows detection of multivariate outliers.

Usage

```r
## S3 method for class 'FRBmultireg'
diagplot(x, Xdist = TRUE, ...)

## S3 method for class 'FRBpca'
diagplot(x, EIF = TRUE, ...)

## S3 method for class 'FRBhot'
diagplot(x, ...)
```

Arguments

- `x`: an R object of class FRBmultireg (typically created by FRBmultiregS, FRBmultiregMM or FRBmultiregGS or by Sest_multireg, MMest_multireg or GSeest_multireg) or an R object of class FRBpca (typically created by FRBpcaS or FRBpcaMM) or an R object of class FRBhot (typically created by FRBhotellingS or FRBhotellingMM)
- `Xdist`: logical: if TRUE, the plot shows the robust distance versus the distance in the space of the explanatory variables; if FALSE, it plots the robust distance versus the index of the observation
- `EIF`: logical: if TRUE, the plot shows the robust distance versus an influence measure for each point; if FALSE, it plots the robust distance versus the index of the observation
- `...`: potentially more arguments to be passed

Details

The diagnostic plots are based on the robust distances of the observations. In a multivariate sample \(X_n = \{x_1, \ldots, x_n\}\), the robust distance \(d_i\) of observation \(i\) is given by \(d_i^2 = (x_i - \hat{\mu})'\hat{\Sigma}^{-1}(x_i - \hat{\mu})\), where \(\hat{\mu}\) and \(\hat{\Sigma}\) are robust estimates of location and covariance. Observations with large robust distance are considered as outlying.

The default diagnostic plot in the multivariate regression setting (i.e. for objects of type FRBmultireg and Xdist=TRUE), shows the residual distances (i.e. the robust distances of the multivariate residuals) based on the estimates in \(x\), versus the distances within the space of the explanatory variables.
The latter are based on robust estimates of location and scatter for the data matrix $X$ (without intercept). Computing these robust estimates may take an appreciable amount of time. The estimator used corresponds to the one which was used in obtaining multireg (with the same breakdown point, for example, and the same control parameters). On the vertical axis a cutoff line is drawn at the square root of the .975 quantile of the chi-squared distribution with degrees of freedom equal to the number of response variables. On the horizontal axis the same quantile is drawn but now with degrees of freedom equal to the number of covariates (not including intercept). Those points to the right of the cutoff can be viewed as high-leverage points. These can be classified into so-called 'bad' or 'good' leverage points depending on whether they are above or below the cutoff. Points above the cutoff but to the left of the vertical cutoff are sometimes called vertical outliers. See also Van Aelst and Willems (2005) for example.

To avoid the additional computation time, one can choose $\text{dist}=\text{FALSE}$, in which case the residual distances are simply plotted versus the index of the observation.

The default plot in the context of PCA (i.e. for objects of type $\text{FRBpca}$ and $\text{EIF}=\text{FALSE}$) is a plot proposed by Pison and Van Aelst (2004). It shows the robust distance versus a measure of the overall empirical influence of the observation on the (classical) principal components. The empirical influences are obtained by using the influence function of the eigenvectors of the empirical or classical shape estimator at the normal model, and by substituting therein the robust estimates for the population parameters. The overall influence value is then defined by averaging the squared influence over all coefficients in the eigenvectors. The vertical line on the plot is an indicative cutoff value, obtained through simulation. This last part takes a few moments of computation time.

Again, to avoid the additional computation time, one can choose $\text{EIF}=\text{FALSE}$, in which case the robust distances are simply plotted versus the index of the observation.

For the result of the robust Hotelling test (i.e. for objects of type $\text{FRBhot}$), the method plots the robust distance versus the index. In case of a two-sample test, the indices are within-sample and a vertical line separates the two groups. In the two-sample case, each group has its own location estimate $\hat{\mu}$ and a common covariance estimate $\hat{\Sigma}$.

Author(s)

Gert Willems and Ella Roelant

References


See Also

FRBmultiregS, FRBmultiregMM, FRBmultiregGS, FRBpcaS, FRBpcaMM, FRBhotellingS, FRBhotellingMM
Examples

# for multivariate regression:
data(schooldata)
MMres <- MMest_multireg(cbind(reading, mathematics, selfesteem)~., data=schooldata)
diagplot(MMres)
# a large 'bad leverage' outlier should be noticeable (observation 59)

# for PCA:
## Not run:
data(ForgedBankNotes)
MMres <- FRBpcaMM(ForgedBankNotes)
diagplot(MMres)
## End(Not run)

# a group of 15 fairly strong outliers can be seen which apparently would have
# a large general influence on a classical PCA analysis

# for Hotelling tests (two-sample)
## Not run:
data(hemophilia)
MMres <- FRBhotellingMM(cbind(AHFactivity, AHFantigen)~gr, data=hemophilia)
diagplot(MMres)
## End(Not run)

# the data seem practically outlier-free

---

ForgedBankNotes Swiss (forged) bank notes data

Description

Six measurements made on 100 forged Swiss bank notes.

Usage

data(ForgedBankNotes)

Format

The data frame contains the following columns:

- **Length** metric length of the bill
- **Left** height of the bill, measured on the left
- **Right** height of the bill, measured on the right
- **Bottom** distance of inner frame to the lower border
- **Top** distance of inner frame to the upper border
- **Diagonal** length of the diagonal
Details

The original data set in Flury and Riedwyl (1988) additionally contained 100 genuine bank notes, but these are not included here.

Source


References


Examples

data(ForgedBankNotes)
pairs(ForgedBankNotes)

FRB-internal

*Internal functions for the package FRB*

Description

These functions are not intended to be used by the user.

FRBhotellingMM

*Robust Hotelling test using the MM-estimator*

Description

Robust one-sample and two-sample Hotelling test using the MM-estimator and the Fast and Robust Bootstrap.

Usage

```r
## S3 method for class 'formula'
FRBhotellingMM(formula, data=NULL, ...)
```

```r
## Default S3 method:
FRBhotellingMM(X, Y=NULL, mu0 = 0, R = 999, conf = 0.95,
               method = c("Hefung", "pool"), control=MMcontrol(...),
               na.action=na.omit, ...)
```
Arguments

- formula: an object of class `formula`; a symbolic description of the model to be fit.
- data: data frame from which variables specified in formula are to be taken.
- X: a matrix or data-frame
- Y: an optional matrix or data-frame in case of a two-sample test
- mu0: an optional vector of data values (or a single number which will be repeated \( p \) times) indicating the true value of the mean (does not apply in case of the two-sample test). Default is the null vector \( \mu_0=\emptyset \).
- R: number of bootstrap samples. Default is \( R=999 \).
- conf: confidence level for the simultaneous confidence intervals. Default is \( \text{conf}=0.95 \).
- method: for the two-sample Hotelling test, indicates the way the common covariance matrix is estimated: "pool"= pooled covariance matrix, "HeFung"= using the multisample method of He and Fung.
- control: a list with control parameters for tuning the MM-estimate and its computing algorithm, see `mmcontrol()`.
- na.action: a function which indicates what should happen when the data contain NAs. Defaults to `na.omit`.
- ... allows for specifying control parameters directly instead of via `control`.

Details

The classical Hotelling test for testing if the mean equals a certain value or if two means are equal is modified into a robust one through substitution of the empirical estimates by the MM-estimates of location and scatter. The MM-estimator, using Tukey’s biweight function, is tuned by default to have a breakdown point of 50% and 95% location efficiency. This could be changed through the `control` argument if desired. The MM-estimates are obtained by first computing the S-estimates with the fast-S algorithm followed by the M-part through reweighted least squares (RWLS) iteration. See `mmcontrol` for some adjustable tuning parameters regarding the algorithm.

The fast and robust bootstrap is used to mimic the distribution of the test statistic under the null hypothesis. For instance, the 5% critical value for the test is given by the 95% quantile of the recalculated statistics.

Robust simultaneous confidence intervals for linear combinations of the mean (or difference in means) are developed similarly to the classical case (Johnson and Wichern, 1988, page 239). The value `CI` is a matrix with the confidence intervals for each element of the mean (or difference in means), with level `conf`. It consists of two rows, the first being the lower bound and the second the upper bound. Note that these intervals are rather conservative in the sense that the simultaneous confidence level holds for all linear combinations and here only \( p \) of these are considered (with \( p \) the dimension of the data).

For the two-sample Hotelling test we assume that the samples have an underlying distribution with the same covariance matrix. This covariance matrix can be estimated in two different ways using the pooled covariance matrix or the two-sample estimator of He and Fung (He and Fung 2000), and argument `method` defaults to the second option. For more details see Roelant et al. (2008).

In the two-sample version, the null hypothesis always states that the two means are equal. For the one-sample version, the default null hypothesis is that the mean equals zero, but the hypothesized value can be changed and specified through argument `mu0`.
Bootstrap samples are discarded if the fast and robust covariance estimate is not positive definite, such that the actual number of recalculations used can be lower than R. This number is returned as ROK.

Value

An object of class FRBhot which extends class htest and contains at least the following components:

- **statistic** the value of the robust test statistic.
- **pvalue** p-value of the robust one or two-sample Hotelling test, determined by the fast and robust bootstrap
- **estimate** the estimated mean vector or vectors depending on whether it was a one-sample test or a two-sample test.
- **alternative** a character string describing the alternative hypothesis.
- **method** a character string indicating what type of Hotelling test was performed.
- **data.name** a character string giving the name(s) of the data.
- **teststat.boot** the bootstrap recalculated values of the robust test statistic.
- **CI** bootstrap simultaneous confidence intervals for each component of the center
- **conf** a copy of the conf argument
- **Sigma** covariance of one-sample or common covariance matrix in the case of two samples
- **w** implicit weights corresponding to the MM-estimates (i.e. final weights in the RWLS procedure)
- **outFlag** outlier flags: 1 if the robust distance of the observation exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of x; 0 otherwise
- **ROK** number of bootstrap samples actually used (i.e. not discarded due to non-positive definite covariance

Author(s)

Ella Roelant, Stefan Van Aelst and Gert Willems

References


See Also

`plot.FRBhot, summary.FRBhot, FRBhotellingS, MMcontrol`

Examples

```r
## One sample robust Hotelling test
data(delivery)
delivery.x <- delivery[,1:2]
FRBhotellingMM(delivery.x, R=199)

## One sample robust Hotelling test
data(ForgedBankNotes)
sample.mean <- apply(ForgedBankNotes, 2, mean)
res = FRBhotellingMM(ForgedBankNotes, mu0=sample.mean, R=199)
res
# Note that the test rejects the hypothesis that the true mean equals the
# sample mean; this is due to outliers in the data (i.e. the robustly estimated
# mean apparently significantly differs from the non-robust sample mean.

# Graphical display of the results:
plot(res)
# It is clear from the (scaled) simultaneous confidence limits that the rejection
# of the hypothesis is due to the differences in variables Bottom and Diagonal

## Two sample robust Hotelling test
data(hemophilia)
grp <- as.factor(hemophilia[,3])
x <- hemophilia[which(grp==levels(grp)[1]),1:2]
y <- hemophilia[which(grp==levels(grp)[2]),1:2]

#using the pooled covariance matrix to estimate the common covariance matrix
## Not run: res <- FRBhotellingMM(x,y,method="pool")

#using the estimator of He and Fung to estimate the common covariance matrix
res <- FRBhotellingMM(x,y,method="HeFung", R=199)

# or using the formula interface
## Not run: res <- FRBhotellingMM(as.matrix(hemophilia[,3])-hemophilia[,3],method="HeFung")

# From the confidence limits it can be seen that the significant difference
```
# is mainly caused by the AHFactivity variable. The graphical display helps too:
plot(res)
# the red line on the histogram indicates the test statistic value in the original
# sample (it is omitted if the statistic exceeds 100)

---

**FRBhotellings**  
*Robust Hotelling test using the S-estimator*

**Description**

Robust one-sample and two-sample Hotelling test using the S-estimator and the Fast and Robust Bootstrap.

**Usage**

```r
## S3 method for class 'formula'
FRBhotellings(formula, data=NULL, ...)

## Default S3 method:
FRBhotellings(x, y=NULL, mu=0, R=999, bdp=0.5, conf=0.95,
method=c("HeFung", "pool"), control=Scontrol(...),
na.action=na.omit, ...)
```

**Arguments**

- `formula`  
an object of class `formula`; a symbolic description of the model to be fit.
- `data`  
data frame from which variables specified in formula are to be taken.
- `X`  
a matrix or data-frame
- `Y`  
an optional matrix or data-frame in case of a two-sample test
- `mu`  
an optional vector of data values (or a single number which will be repeated
p times) indicating the true value of the mean (does not apply in case of the
two-sample test). Default is the null vector `mu=0`
- `R`  
number of bootstrap samples. Default is R=999.
- `bdp`  
required breakdown point. Should have 0 < bdp ≤ 0.5, the default is 0.5
- `conf`  
confidence level for the simultaneous confidence intervals. Default is conf=0.95
- `method`  
for the two-sample Hotelling test, indicates the way the common covariance
matrix is estimated: "pool"= pooled covariance matrix, "HeFung"= using the
He and Fung method
- `control`  
a list with control parameters for tuning the computing algorithm, see `Scontrol()`.
- `na.action`  
a function which indicates what should happen when the data contain NAs. De-
defaults to `na.omit`
- `...`  
allows for specifying control parameters directly instead of via `control`
Details

The classical Hotelling test for testing if the mean equals a certain center or if two means are equal is modified into a robust one through substitution of the empirical estimates by the S-estimates of location and scatter. The S-estimator uses Tukey’s biweight function where the constant is chosen to obtain the desired breakdown point as specified by bdp. One-sample S-estimates are computed by a call to the implementation of the fast-S algorithm in the rrcov package of Todorov and Filzmoser (2009). For two-sample S-estimates an adaptation of the fast-S algorithm is used. The tuning parameters of the algorithm can be changed via control.

The fast and robust bootstrap is used to mimic the distribution of the test statistic under the null hypothesis. For instance, the 5% critical value for the test is given by the 95% quantile of the recalculated statistics.

Robust simultaneous confidence intervals for linear combinations of the mean (or difference in means) are developed similarly to the classical case (Johnson and Wichern, 1988, page 239). The value CI is a matrix with the confidence intervals for each element of the mean (or difference in means), with level conf. It consists of two rows, the first being the lower bound and the second the upper bound. Note that these intervals are rather conservative in the sense that the simultaneous confidence level holds for all linear combinations and here only p of these are considered (with p the dimension of the data).

For the two-sample Hotelling test we assume that the samples have an underlying distribution with the same covariance matrix. This covariance matrix can be estimated in two different ways using the pooled covariance matrix or the two-sample estimator of He and Fung (He and Fung 2000), and argument method defaults to the second option. For more details see Roelant et al. (2008).

In the two-sample version, the null hypothesis always states that the two means are equal. For the one-sample version, the default null hypothesis is that the mean equals zero, but the hypothesized value can be changed and specified through argument mu0.

Bootstrap samples are discarded if the fast and robust covariance estimate is not positive definite, such that the actual number of recalculations used can be lower than R. This number is returned as ROK.

Value

An object of class FRBhot which extends class htest and contains at least the following components:

- statistic: the value of the robust test statistic.
- pvalue: p-value of the robust one or two-sample Hotelling test, determined by the fast and robust bootstrap.
- estimate: the estimated mean vector or vectors depending on whether it was a one-sample test or a two-sample test.
- alternative: a character string describing the alternative hypothesis.
- method: a character string indicating what type of Hotelling test was performed.
- data.name: a character string giving the name(s) of the data.
- teststat.boot: the bootstrap recalculated values of the robust test statistic.
- CI: bootstrap simultaneous confidence intervals for each component of the center
- conf: a copy of the conf argument
Sigma

covariance of one-sample or common covariance matrix in the case of two samples

\(w\)

implicit weights corresponding to the S-estimates (i.e. final weights in the RWLS procedure at the end of the fast-S algorithm)

outflag

outlier flags: 1 if the robust distance of the observation exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of \(X\); 0 otherwise

ROK

number of bootstrap samples actually used (i.e. not discarded due to non-positive definite covariance

Author(s)

Ella Roelant, Stefan Van Aelst and Gert Willems

References


See Also

plot.FRBlhot, summary.FRBlhot, FRBlhotellingMM, Scontrol

Examples

```r
## One sample robust Hotelling test
data(delivery)
delivery.x <- delivery[,1:2]
FRBlhotellingS(delivery.x,R=199)

## One sample robust Hotelling test
data(ForgedBankNotes)
samplemean <- apply(ForgedBankNotes, 2, mean)
res = FRBlhotellingS(ForgedBankNotes, mu0=samplemean,R=199)
res
```
# Note that the test rejects the hypothesis that the true mean equals the
# sample mean; this is due to outliers in the data (i.e. the robustly estimated
# mean apparently significantly differs from the non-robust sample mean.

# Graphical display of the results:
plot(res)
# It is clear from the (scaled) simultaneous confidence limits that the rejection
# of the hypothesis is due to the differences in variables Bottom and Diagonal

## Two sample robust Hotelling test
data(hemophilia)
group <- as.factor(hemophilia[,3])
X <- hemophilia[which(grp==levels(grp)[1]),1:2]
Y <- hemophilia[which(grp==levels(grp)[2]),1:2]

# Using the pooled covariance matrix to estimate the common covariance matrix
## Not run: res = FRBHotellingS(X,Y,method="pool")

# Using the estimator of He and Fung to estimate the common covariance matrix
res = FRBHotellingS(X,Y,method="HeFung",R=199)

# or using the formula interface
## Not run: res = FRBHotellingS(as.matrix(hemophilia[,3])-hemophilia[,3],method="HeFung",R=99)

# From the confidence limits it can be seen that the significant difference
# is mainly caused by the AHFactivity variable. The graphical display helps too:
plot(res)
# the red line on the histogram indicates the test statistic value in the original
# sample (it is omitted if the statistic exceeds 100)

---

**FRBmultiregGS**

*GS-Estimates for multivariate regression with bootstrap confidence intervals*

**Description**

Computes GS-estimates for multivariate regression together with standard errors, confidence intervals and p-values based on the Fast and Robust Bootstrap.

**Usage**

```r
## S3 method for class 'formula'
FRBmultiregGS(formula, data=NULL, ...)
```
## Default S3 method:
FRBmultiregGS(X, Y, int = TRUE, R = 999, bdp = 0.5, conf = 0.95,
control=GScontrol(...), na.action=na.omit,...)

### Arguments

- **formula**: an object of class `formula`; a symbolic description of the model to be fit.
- **data**: data frame from which variables specified in formula are to be taken.
- **X**: a matrix or data frame containing the explanatory variables.
- **Y**: a matrix or data frame containing the response variables.
- **int**: logical: if TRUE an intercept term is added to the model (unless it is already present in X)
- **R**: number of bootstrap samples. Default is R=999.
- **bdp**: required breakdown point. Should have 0 < bdp ≤ 0.5, the default is 0.5.
- **conf**: confidence level of the bootstrap confidence intervals. Default is conf=0.95.
- **control**: a list with control parameters for tuning the computing algorithm, see `GScontrol()`.
- **na.action**: a function which indicates what should happen when the data contain NAs. Defaults to `na.omit`.
- **...**: allows for specifying control parameters directly instead of via `control`.

### Details

Generalized S-estimators are defined by minimizing the determinant of a robust estimator of the scatter matrix of the differences of the residuals (Roelant et al. 2009). Hence, this procedure is intercept free and only gives an estimate for the slope matrix. To estimate the intercept, we use the M-type estimator of location of Lopuhaa (1992) on the residuals with the residual scatter matrix estimate of the residuals as a preliminary estimate. This computation is carried out by a call to `GSeest_multireg()`, which uses a fast-S-type algorithm (its tuning parameters can be changed via the control argument). The result of this call is also returned as the value `est`.

The Fast and Robust Bootstrap (Salibian-Barrera and Zamar 2002) is used to calculate so-called basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals (Davison and Hinkley 1997, p.194 and p.204 respectively). Apart from the intervals with the requested confidence level, the function also returns p-values for each coefficient corresponding to the hypothesis that the actual coefficient is zero. The p-values are computed as 1 minus the smallest level for which the confidence intervals would include zero. Both BCa and basic bootstrap p-values in this sense are given. The bootstrap calculation is carried out by a call to `GSboot_multireg()`, the result of which is returned as the value `bootest`. Bootstrap standard errors are returned as well.

Note: Bootstrap samples which contain too few distinct observations with positive weights are discarded (a warning is given if this happens). The number of samples actually used is returned via `rok`.

In the `formula`-interface, a multivariate response is produced via `cbind`. For example `cbind(x4,x5) ~ x1+x2+x3`. All arguments from the default method can also be passed to the `formula` method.

The returned object inherits from class `mlm` such that the standard `coef`, `residuals`, `fitted` and `predict` functions can be used.
Value

An object of class `FRBmultireg` which extends class `mlm` and contains at least the following components:

- **coefficients**: GS-estimates of the regression coefficients
- **residuals**: the residuals, that is, response minus fitted values
- **fitted.values**: the fitted values.
- **Sigma**: GS-estimate of the error covariance matrix
- **scale**: GS-estimate of the size of the multivariate errors
- **weights**: implicit weights corresponding to the GS-estimates (i.e., final weights in the RWLS procedure for the intercept estimate)
- **outFlag**: outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise
- **SE**: bootstrap standard errors corresponding to the regression coefficients
- **cov**: bootstrap covariance matrix corresponding to the regression coefficients (in vectorized form)
- **CI.bca.lower**: a matrix containing the lower bound of the bias corrected and accelerated confidence intervals for the regression coefficients
- **CI.bca.upper**: a matrix containing the upper bound of the bias corrected and accelerated confidence intervals for the regression coefficients
- **CI.basic.lower**: a matrix containing the lower bound of basic bootstrap intervals for the regression coefficients
- **CI.basic.upper**: a matrix containing the upper bound of basic bootstrap intervals for the regression coefficients
- **p.bca**: a matrix containing the p-values based on the BCa confidence intervals for the regression coefficients
- **p.basic**: a matrix containing the p-values based on the basic bootstrap intervals for the regression coefficients
- **est**: GS-estimates as returned by the call to `GSest_multireg()`
- **bootest**: bootstrap results for the GS-estimates as returned by the call to `GSboot_multireg()`
- **conf**: a copy of the conf argument
- **method**: a list with following components: `est` = character string indicating that GS-estimates were used, and `bdp` = a copy of the bdp argument
- **control**: a copy of the control argument
- **X, Y**: either copies of the respective arguments or the corresponding matrices produced from `formula`
- **ROK**: number of bootstrap samples actually used (i.e., not discarded due to too few distinct observations with positive weight)
Author(s)

Ella Roelant, Stefan Van Aelst and Gert Willems

References


See Also

summary.FRBmultireg, plot.FRBmultireg, GSboot_multireg, GSest_multireg, FRBmultiregMM, FRBmultiregS, GScontrol

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

# computes 25% breakdown point GS-estimate and 80% confidence intervals
# based on 99 bootstrap samples:
GSres <- FRBmultiregGS(school.x, school.y, R=99, bdp = 0.25, conf = 0.8, nsamp=50)
# or using the formula interface
## Not run: GSres <- FRBmultiregGS(cbind(reading,mathematics,selfesteem)-, data=schooldata,
## bdp = 0.25, conf = 0.8,R=99)
## End(Not run)

# the print method just displays the coefficient estimates
GSres

# the summary function additionally displays the bootstrap standard errors and p-values
# ("BCA" method by default)
summary(GSres)

summary(GSres, confmethod="basic")

# ask explicitly for the coefficient matrix:
GSres$coefficients
# or equivalently,
coef(GSres)
# For the error covariance matrix:
GSres$sigma

# plot some bootstrap histograms for the coefficient estimates
# (with "BCA" intervals by default)
plot(GSres, expl=c("education", "occupation"), resp=c("selfesteem","reading"))

# plot bootstrap histograms for all coefficient estimates
plot(GSres)
# possibly the plot-function has made a selection of coefficients to plot here,
# since 'all' may have been too many to fit on one page, see help(plot.FRBmultireg);
# this is platform-dependent

---

**FRBmultiregMM**  
**MM-Estimates for Multivariate Regression with Bootstrap Inference**

**Description**

Computes MM-estimates for multivariate regression together with standard errors, confidence intervals and p-values based on the Fast and Robust Bootstrap.

**Usage**

```r
## S3 method for class 'formula'
FRBmultiregMM(formula, data=NULL, ...)
```

## Default S3 method:
FRBmultiregMM(X, Y, int = TRUE, R = 999, conf = 0.95, 
control=MMcontrol(...), na.action=na.omit, ...)

**Arguments**

- `formula`: an object of class `formula`; a symbolic description of the model to be fit.
- `data`: data frame from which variables specified in formula are to be taken.
- `X`: a matrix or data frame containing the explanatory variables.
- `Y`: a matrix or data frame containing the response variables.
- `int`: logical: if TRUE an intercept term is added to the model (unless it is already present in X).
- `R`: number of bootstrap samples. Default is R=999.
- `conf`: level of the bootstrap confidence intervals. Default is conf=0.95.
control a list with control parameters for tuning the MM-estimate and its computing
algorithm, see \texttt{MMcontrol}().

na.action a function which indicates what should happen when the data contain NAs. De-
defaults to \texttt{na.omit}.

... allows for specifying control parameters directly instead of via control

Details

Multivariate MM-estimates combine high breakdown point and high Gaussian efficiency. They are
defined by first computing an S-estimate of regression, then fixing the scale component of the error
covariance estimate, and finally re-estimating the regression coefficients and the shape part of the
error covariance by a more efficient M-estimate (see Tatsuoka and Tyler (2000) for MM-estimates
in the special case of location/scatter estimation, and Van Aelst and Willems (2005) for S-estimates
of multivariate regression).

Tukey’s biweight is used for the loss functions. By default, the first loss function (in the S-estimate)
is tuned in order to obtain 50% breakdown point. The default tuning of the second loss function
(M-estimate) ensures 95% efficiency at the normal model for the coefficient estimates. The desired
efficiency can be changed through argument control.

The computation is carried out by a call to \texttt{Mest_multireg()}, which first performs the fast-S
algorithm (see \texttt{Sest_multireg}) and does the M-part by reweighted least squares (RWLS) iteration.
See \texttt{MMcontrol} for some adjustable tuning parameters regarding the algorithm. The result of this
call is also returned as the value \texttt{est}.

The Fast and Robust Bootstrap (Salibian-Barrera and Zamar 2002) is used to calculate so-called
basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals
(Davison and Hinkley 1997, p.194 and p.204 respectively). Apart from the intervals with the re-
quested confidence level, the function also returns p-values for each coefficient corresponding to
the hypothesis that the actual coefficient is zero. The p-values are computed as 1 minus the smallest
level for which the confidence intervals would include zero. Both BCa and basic bootstrap p-values
in this sense are given. The bootstrap calculation is carried out by a call to \texttt{MMboot_multireg}(), the
result of which is returned as the value \texttt{bootest}. Bootstrap standard errors are returned as well.

Note: Bootstrap samples which contain too few distinct observations with positive weights are
discarded (a warning is given if this happens). The number of samples actually used is returned via
\texttt{rok}.

In the \texttt{formula}-interface, a multivariate response is produced via \texttt{cbind}. For example
\texttt{cbind(x4,x5) \sim x1+x2+x3.}

All arguments from the default method can also be passed to the \texttt{formula} method except for \texttt{int}
(passing \texttt{int} explicitly will produce an error; the inclusion of an intercept term is determined by
\texttt{formula}).

The returned object inherits from class \texttt{mlm} such that the standard \texttt{coef, residuals, fitted} and
\texttt{predict} functions can be used.

Value

An object of class \texttt{FRBmultireg} which extends class \texttt{mlm} and contains at least the following com-
ponents:

\texttt{coefficients} MM-estimates of the regression coefficients
residuals  the residuals, that is response minus fitted values
fitted.values  the fitted values.
Sigma  MM-estimate of the error covariance matrix
scale  MM-estimate of the size of the multivariate errors
weights  implicit weights corresponding to the MM-estimates (i.e. final weights in the RWLS procedure)
outFlag  outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise
SE  bootstrap standard errors corresponding the regression coefficients
cov  bootstrap covariance matrix corresponding to the regression coefficients (in vectorized form)
CI.bca.lower  a matrix containing the lower bounds of the bias corrected and accelerated confidence intervals for the regression coefficients.
CI.bca.upper  a matrix containing the upper bounds of the bias corrected and accelerated confidence intervals for the regression coefficients.
CI.basic.lower  a matrix containing the lower bounds of basic bootstrap intervals for the regression coefficients.
CI.basic.upper  a matrix containing the upper bounds of basic bootstrap intervals for the regression coefficients.
p.bca  a matrix containing the p-values based on the BCa confidence intervals for the regression coefficients.
p.basic  a matrix containing the p-values based on the basic bootstrap intervals for the regression coefficients.
est  MM-estimates as returned by the call to \texttt{MMest_multireg()}
bootest  bootstrap results for the MM-estimates as returned by the call to \texttt{MMboot_multireg()}
conf  a copy of the conf argument
method  a list with following components: est = character string indicating that MM-estimates were used, bdp = a copy of bdp from the control argument, and eff = a copy of eff from the control argument
control  a copy of the control argument
X, Y  either copies of the respective arguments or the corresponding matrices produced from \texttt{formula}
ROK  number of bootstrap samples actually used (i.e. not discarded due to too few distinct observations with positive weight)

\textbf{Author(s)}

Gert Willems, stefan Van Aelst and Ella Roelant
References


See Also

`summary.FRBmultireg.plot.FRBmultireg.MMboot_multireg, MMest_multireg, FRBmultiregS, FRBmultiregGS, MMcontrol`

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

# computes MM-estimate and 95% confidence intervals
# based on 999 bootstrap samples:
MMres <- FRBmultiregMM(school.x, school.y, R=999, conf = 0.95)
# or, equivalently using the formula interface
## Not run: MMres <- FRBmultiregMM(cbind(reading,mathematics,selfesteem)-, data=schooldata, R=999, conf = 0.95)
## End(Not run)

# the print method displays the coefficient estimates
MMres

# the summary function additionally displays the bootstrap standard errors and p-values
# ("BCA" method by default)
summary(MMres)
summary(MMres, confmethod="basic")

# ask explicitly for the coefficient matrix:
MMres$coefficients
# or equivalently,
coef(MMres)
# For the error covariance matrix:
MMres$Sigma
### Description

Computes S-estimates for multivariate regression together with standard errors, confidence intervals and p-values based on the Fast and Robust Bootstrap.

### Usage

```r
## S3 method for class 'formula'
FRBmultiregS(formula, data=NULL, ...)

## Default S3 method:
FRBmultiregS(X, Y, int = TRUE, R = 999, bdp = 0.5, conf = 0.95, control=Scontrol(...), na.action=na.omit, ...)
```

### Arguments

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>formula</td>
<td>an object of class <code>formula</code>; a symbolic description of the model to be fit.</td>
</tr>
<tr>
<td>data</td>
<td>data frame from which variables specified in formula are to be taken.</td>
</tr>
<tr>
<td>X</td>
<td>a matrix or data frame containing the explanatory variables.</td>
</tr>
<tr>
<td>Y</td>
<td>a matrix or data frame containing the response variables.</td>
</tr>
<tr>
<td>int</td>
<td>logical: if TRUE an intercept term is added to the model (unless it is already present in X)</td>
</tr>
<tr>
<td>R</td>
<td>number of bootstrap samples. Default is R=999.</td>
</tr>
<tr>
<td>bdp</td>
<td>required breakdown point for the S-estimates. Should have (0 &lt; bdp \leq 0.5), the default is 0.5</td>
</tr>
</tbody>
</table>
conf level of the bootstrap confidence intervals. Default is \texttt{conf}=\texttt{0.95}

control a list with control parameters for tuning the computing algorithm, see \texttt{Scontrol()}.

na.action a function which indicates what should happen when the data contain NAs. Defaults to \texttt{na.omit}.

... allows for specifying control parameters directly instead of via \texttt{control}

**Details**

Multivariate S-estimates were introduced by Davies (1987) and can be highly robust while enjoying a reasonable Gaussian efficiency. Their use in the multivariate regression setting was discussed in Van Aelst and Willems (2005). The loss function used here is Tukey’s biweight. It is tuned in order to achieve the required breakdown point \texttt{bdp} (any value between 0 and 0.5).

The computation is carried out by a call to \texttt{Sest_multireg()}, which performs the fast-S algorithm (Salibian-Barrera and Yohai 2006), see \texttt{Scontrol} for its tuning parameters. The result of this call is also returned as the value \texttt{est}.

The Fast and Robust Bootstrap (Salibian-Barrera and Zamar 2002) is used to calculate so-called basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals (Davison and Hinkley 1997, p.194 and p.204 respectively). Apart from the intervals with the requested confidence level, the function also returns p-values for each coefficient corresponding to the hypothesis that the actual coefficient is zero. The p-values are computed as 1 minus the smallest level for which the confidence intervals would include zero. Both BCa and basic bootstrap p-values in this sense are given. The bootstrap calculation is carried out by a call to \texttt{Sboot_multireg()}, the result of which is returned as the value \texttt{bootest}. Bootstrap standard errors are returned as well.

Note: Bootstrap samples which contain too few distinct observations with positive weights are discarded (a warning is given if this happens). The number of samples actually used is returned via \texttt{ROK}.

In the \texttt{formula}-interface, a multivariate response is produced via \texttt{cbind}. For example \texttt{cbind(x4,x5) ~ x1+x2+x3}. All arguments from the default method can also be passed to the \texttt{formula} method except for \texttt{int} (passing \texttt{int} explicitly will produce an error; the inclusion of an intercept term is determined by \texttt{formula}).

The returned object inherits from class \texttt{mlm} such that the standard \texttt{coef}, \texttt{residuals}, \texttt{fitted} and \texttt{predict} functions can be used.

**Value**

An object of class \texttt{FRBmultiregS} which extends class \texttt{mlm} and contains at least the following components:

- \texttt{coefficients} MM-estimates of the regression coefficients
- \texttt{residuals} the residuals, that is response minus fitted values
- \texttt{fitted.values} the fitted values.
- \texttt{Sigma} S-estimate of the error covariance matrix
- \texttt{scale} MM-estimate of the size of the multivariate errors
- \texttt{weights} implicit weights corresponding to the S-estimates (i.e. final weights in the RWLS procedure at the end of the fast-S algorithm)
outFlag outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise

SE bootstrap standard errors corresponding to the regression coefficients.

cov bootstrap covariance matrix corresponding to the regression coefficients (in vectorized form)

CI.bca.lower a matrix containing the lower bounds of the bias corrected and accelerated confidence intervals for the regression coefficients.

CI.bca.upper a matrix containing the upper bounds of the bias corrected and accelerated confidence intervals for the regression coefficients.

CI.basic.lower a matrix containing the lower bounds of basic bootstrap intervals for the regression coefficients.

CI.basic.upper a matrix containing the upper bounds of basic bootstrap intervals for the regression coefficients.

p.bca a matrix containing the p-values based on the BCa confidence intervals for the regression coefficients.

p.basic a matrix containing the p-values based on the basic bootstrap intervals for the regression coefficients.

est S-estimates as returned by the call to Sest_multireg()

bootest bootstrap results for the S-estimates as returned by the call to Sboot_multireg()

conf a copy of the conf argument

method a list with following components: est = character string indicating that S-estimates were used, and bdp = a copy of the bdp argument

control a copy of the control argument

X, Y either copies of the respective arguments or the corresponding matrices produced from formula

ROK number of bootstrap samples actually used (i.e. not discarded due to too few distinct observations with positive weight)

Author(s)
Gert Willems, Stefan Van Aelst and Ella Roelant

References


See Also

summary.FRBMultireg, plot.FRBMultireg, Sboot_multireg, Sest_multireg, FRBMultiregMM, FRBMultiregGS, Scontrol

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

#computes 25% breakdown point S-estimate and 99% confidence intervals
#based on 999 bootstrap samples:
Sres <- FRBMultiregS(school.x, school.y, R=999, bdp = 0.25, conf = 0.99)
#or, equivalently using the formula interface
## Not run: Sres <- FRBMultiregS(cbind(reading,mathematics,selfesteem)-., data=schooldata, R=999, bdp = 0.25, conf = 0.99)
## End(Not run)

#the print method displays the coefficient estimates
Sres

#the summary function additionally displays the bootstrap standard errors and p-values
#("BCA" method by default)
summary(Sres)

summary(Sres, confmethod="basic")

#ask explicitly for the coefficient matrix:
Sres$coefficients
# or equivalently,
coef(Sres)
#For the error covariance matrix:
Sres$Sigma

#plot some bootstrap histograms for the coefficient estimates
#(with "BCA" intervals by default)
plot(Sres, expl=c("education", "occupation"), resp=c("selfesteem","reading"))

#plot bootstrap histograms for all coefficient estimates
plot(Sres)
#probably the plot-function has made a selection of coefficients to plot here,
#since 'all' was too many to fit on one page, see help(plot.FRBMultireg);
#this is platform-dependent

---

**FRBpcaMM**  
*PCA based on Multivariate MM-estimators with Fast and Robust Bootstrap*

## Description

Performs principal components analysis based on the robust MM-estimate of the shape matrix. Additionally uses the Fast and Robust Bootstrap method to compute inference measures such as standard errors and confidence intervals.

## Usage

```r
## S3 method for class 'formula'
FRBpcaMM(formula, data=NULL, ...)
```

## Default S3 method:
```r
FRBpcaMM(Y, R = 999, conf = 0.95, control=MMcontrol(...),
na.action=na.omit, ...)
```

## Arguments

- `formula`  
an object of class `formula`; a symbolic description of the model to be fit.
- `data`  
data frame from which variables specified in formula are to be taken.
- `Y`  
matrix or data frame.
- `R`  
number of bootstrap samples. Default is R=999.
- `conf`  
level of the bootstrap confidence intervals. Default is conf=0.95.
- `control`  
a list with control parameters for tuning the MM-estimate and its computing algorithm, see `MMcontrol()`.
- `na.action`  
a function which indicates what should happen when the data contain NAs. Defaults to `na.omit`.
- `...`  
allows for specifying control parameters directly instead of via `control`. 
Details

Multivariate MM-estimates are defined by first computing an S-estimate of location and covariance, then fixing its scale component and re-estimating the location and the shape by a more efficient M-estimate, see Tatsuoka and Tyler (2000). Tukey’s biweight is used for the loss functions. By default, the first loss function (in the S-estimate) is tuned in order to obtain 50% breakdown point. The default tuning of the second loss function (M-estimate) ensures 95% efficiency for the shape matrix estimate at the normal model. The desired efficiency can be changed through argument control. (However, control parameter shapeEff will always be considered as TRUE by this function, whichever value is specified.) The MM-estimates are computed by a call to the implementation in the rrcov package of Todorov and Filzmoser (2009). The result of this call is also returned as the value est.

PCA is performed by computing the eigenvalues (eigval) and eigenvectors (eigvec) of the MM-estimate of shape, which is a rescaled version of the MM-estimate of covariance (rescaled to have determinant equal to 1). With pvar the function also provides the estimates for the percentage of variance explained by the first k principal components, which are simply the cumulative proportions of the eigenvalues sum. Here, k ranges from 1 to p − 1 (with p the number of variables in y). The eigenvectors are always given in the order of descending eigenvalues.

The Fast and Robust Bootstrap (Salibian-Barrera and Zamar 2002) is used to calculate standard errors, and also so-called basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals (Davison and Hinkley 1997, p.194 and p.204 respectively) corresponding to the estimates eigval, eigvec and pvar. The bootstrap is also used to estimate the average angles between true and estimated eigenvectors, returned as avgangle. See Salibian-Barrera, Van Aelst and Willems (2006). The fast and robust bootstrap computations for the MM-estimates are performed by mmboot_loccov() and its raw result can be found in bootest. The actual bootstrap recalculations for the PCA-related quantities can be found in eigval.boot, eigvec.boot and pvar.boot, where each column represents a bootstrap sample. For eigvec.boot, the eigenvectors are stacked on top of each other and the same goes for eigvec.CI.bca and eigvec.CI.basic which hold the confidence limits.

The two columns in the confidence limits always respectively represent the lower and upper limits. For the percentage of variance the function also provides one-sided confidence intervals ([-inf, upper]), which can be used to test the hypothesis that the true percentage at least equals a certain value.

Bootstrap samples are discarded if the fast and robust shape estimate is not positive definite, such that the actual number of recalculations used can be lower than R. This actual number equals R - failsamples. However, if more than 0.75R of the bootstrap shape estimates is non-positive definite, all bootstrap samples will be used anyway, and the negative eigenvalues are simply set to zero (which may impact the confidence limits and standard errors for the smallest eigenvalues in eigval and pvar).

Value

An object of class FRBpca, which contains the following components:

- shape (p x p) MM-estimate of the shape matrix of y
- eigval (p x 1) eigenvalues of MM shape
- eigvec (p x p) eigenvectors of MM-shape
pvar  (p-1 x 1) percentages of variance for MM eigenvalues

eigval.boot (p x R) eigenvalues of MM shape

eigvec.boot (p*p x R) eigenvectors of MM-shape (vectorized)

eigvec.boot (p x p) bootstrap standard error for MM eigenvectors

eigval.SE (p x 1) bootstrap standard error for MM eigenvalues

angles (p x R) angles between bootstrap eigenvectors and original MM eigenvectors (in radians; in [0 pi/2])

avgangle (p x 1) average angles between bootstrap eigenvectors and original MM eigenvectors (in radians; in [0 pi/2])

eigval.CI.bca (p x 2) BCa intervals for MM eigenvalues

eigvec.CI.bca (p*p x 2) BCa intervals for MM eigenvectors (vectorized)

pvar.CI.bca (p-1 x 2) BCa intervals for percentage of variance for MM-eigenvalues

pvar.CIone.bca (p-1 x 1) one-sided BCa intervals for percentage of variance for MM-eigenvalues ([-infty upper])

eigval.CI.basic (p x 2) basic bootstrap intervals for MM eigenvalues

eigvec.CI.basic (p*p x 2) basic bootstrap intervals for MM eigenvectors (vectorized)

pvar.CI.basic (p-1 x 2) basic bootstrap intervals for percentage of variance for MM-eigenvalues

pvar.CIone.basic (p-1 x 1) one-sided basic bootstrap intervals for percentage of variance for MM-eigenvalues ([-infty upper])

est list containing the MM-estimates of location and scatter

bootest (list) result of \texttt{mmboot_loccov}()

failedsamples number of bootstrap samples with non-positive definiteness of shape

conf a copy of the conf argument

method a character string giving the robust PCA method that was used

w implicit weights corresponding to the MM-estimates (i.e. final weights in the RWLS procedure)

outFlag outlier flags: 1 if the robust distance of the observation exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of \texttt{Y}; 0 otherwise

Y copy of the data argument as a matrix

Author(s)

Gert Willems, Stefan Van Aelst and Ella Roelant
References


See Also

plot.FRBlogca, summary.FRBlogca, print.FRBlogca, FRBlogcaS, MMboot_loccov, MMcontrol

Examples

data(ForgedBankNotes)

MMpcares <- FRBlogcaMM(ForgedBankNotes, R=999, conf=0.95)
# or using the formula interface
## Not run: MMpcares <- FRBlogcaMM(~., data=ForgedBankNotes, R=999, conf=0.95)

# the simple print method shows the standard deviations with confidence limits:
MMpcares

# the summary functions shows a lot more (see help(summary.FRBlogca)):
summary(MMPcares)

# ask for the eigenvalues:
MMpcares$eigval

# or, in more pretty format, with confidence limits:
summary(MMPcares)$eigvals

# note that the standard deviations of the print-output can also be asked for by:
sqrt( summary(MMPcares)$eigvals )

# the eigenvectors and their standard errors:
MMpcares$eigvec  # or prettier: summary(MMPcares)$eigvecs
# take a look at the bootstrap distribution of the first eigenvalue
hist(MMpcares$eigval.boot[,1])

# that bootstrap distribution is used to compute confidence limits as depicted
# by the screeplot function:
plotFRBvars(MMpcares, cumul=0)

# all plots for the FRB-PCA result:
plot(MMpcares)

FRBpcas

PCA based on Multivariate S-estimators with Fast and Robust Bootstrap

Description
Performs principal components analysis based on the robust S-estimate of the shape matrix. Additionally uses the Fast and Robust Bootstrap method to compute inference measures such as standard errors and confidence intervals.

Usage

## S3 method for class 'formula'
FRBpcas(formula, data=NULL, ...)

## Default S3 method:
FRBpcas(Y, R = 999, bdp = 0.5, conf = 0.95, control=Scontrol(...),
na.action=na.omit, ...)

Arguments

- **formula**: an object of class formula; a symbolic description of the model to be fit.
- **data**: data frame from which variables specified in formula are to be taken.
- **Y**: matrix or data frame.
- **R**: number of bootstrap samples. Default is R=999.
- **bdp**: required breakdown point for the S-estimates. Should have 0 < bdp ≤ 0.5, the default is 0.5.
- **conf**: level of the bootstrap confidence intervals. Default is conf=0.95.
- **control**: a list with control parameters for tuning the computing algorithm, see Scontrol().
- **na.action**: a function which indicates what should happen when the data contain NAs. Defaults to na.omit.
- **...**: allows for specifying control parameters directly instead of via control.
Details

Multivariate S-estimates were introduced by Davies (1987) and can be highly robust while enjoying a reasonable Gaussian efficiency. The loss function used here is Tukey’s biweight. It will be tuned in order to achieve the required breakdown point bdp (any value between 0 and 0.5). The MM-estimates are computed by a call to the implementation of the fast-S algorithm (Salibian-Barrera and Yohai 2006) in the rrcov package of Todorov and Filzmoser (2009). Scontrol provides some adjustable tuning parameters regarding the algorithm. The result of this call is also returned as the value est.

PCA is performed by computing the eigenvalues (eigval) and eigenvectors (eigvec) of the S-estimate of shape, which is a rescaled version of the S-estimate of covariance (rescaled to have determinant equal to 1). With pvar the function also provides the estimates for the percentage of variance explained by the first $k$ principal components, which are simply the cumulative proportions of the eigenvalues sum. Here, $k$ ranges from 1 to $p - 1$ (with $p$ the number of variables in $Y$). The eigenvectors are always given in the order of descending eigenvalues.

The Fast and Robust Bootstrap (Salibian-Barrera and Zamar 2002) is used to calculate standard errors, and also so-called basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals (Davison and Hinkley 1997, p.194 and p.204 respectively) corresponding to the estimates eigval, eigvec and pvar. The bootstrap is also used to estimate the average angles between true and estimated eigenvectors, returned as avgangle. See Salibian-Barrera, Van Aelst and Willems (2006). The fast and robust bootstrap computations for the S-estimates are performed by sboot_loccov() and its raw result can be found in bootest. The actual bootstrap values of the PCA-related quantities can be found in eigval.boot, eigvec.boot and pvar.boot, where each column represents a bootstrap sample. For eigvec.boot, the eigenvectors are stacked on top of each other and the same goes for eigvec.CI.bca and eigvec.CI.basic which hold the confidence limits.

The two columns in the confidence limits always respectively represent the lower and upper limits. For the percentage of variance the function also provides one-sided confidence intervals ([$-\infty$, upper]), which can be used to test the hypothesis that the true percentage at least equals a certain value.

Bootstrap samples are discarded if the fast and robust covariance estimate is not positive definite, such that the actual number of recalculations used can be lower than R. This actual number equals R - failedsamples. However, if more than 0.75R of the bootstrap shape estimates is non-positive definite, the failed bootstrap samples are recovered by applying the make_positive_definite function (from package corpcor). If this also fails, the corresponding bootstrap sample is discarded after all, but such situation should be rare. This recovery may have an impact on the confidence limits and standard errors of especially the smallest eigenvalues in eigval and pvar.

Value

An object of class FRBpca, which contains the following components:

- **shape** (p x p) S-estimate of the shape matrix of Y
- **eigval** (p x 1) eigenvalues of S shape
- **eigvec** (p x p) eigenvectors of S-shape
- **pvar** (p-1 x 1) percentages of variance for S eigenvalues
- **eigval.boot** (p x R) eigenvalues of S shape
eigvec.boot  (p*p x R) eigenvectors of S-shape (vectorized)
pvar.boot  (p-1 x R) percentages of variance for S eigenvalues
eigval.SE  (p x 1) bootstrap standard error for S eigenvalues
eigvec.SE  (p x p) bootstrap standard error for S eigenvectors
pvar.SE  (p-1 x 1) bootstrap standard error for percentage of variance for S eigenvalues
angles  (p x R) angles between bootstrap eigenvectors and original S eigenvectors (in radians; in [0 pi/2])
avgangle  (p x 1) average angles between bootstrap eigenvectors and original S eigenvectors (in radians; in [0 pi/2])
eigval.CI.bca  (p x 2) BCa intervals for S eigenvalues
eigvec.CI.bca  (p*p x 2) BCa intervals for S eigenvectors (vectorized)
pvar.CI.bca  (p-1 x 2) BCa intervals for percentage of variance for S-eigenvalues
pvar.CIone.bca  (p-1 x 1) one-sided BCa intervals for percentage of variance for S-eigenvalues ([-infty upper])
eigval.CI.basic  (p x 2) basic bootstrap intervals for S eigenvalues
eigvec.CI.basic  (p*p x 2) basic bootstrap intervals for S eigenvectors (vectorized)
pvar.CI.basic  (p-1 x 2) basic bootstrap intervals for percentage of variance for S-eigenvalues
pvar.CIone.basic  (p-1 x 1) one-sided basic bootstrap intervals for percentage of variance for S-eigenvalues ([-infty upper])
est  list containing the S-estimates of location and scatter
bootest  (list) result of Sboot_loccov()
failedsamples  number of bootstrap samples with non-positive definiteness of shape
conf  a copy of the conf argument
method  a character string giving the robust PCA method that was used
w  implicit weights corresponding to the S-estimates (i.e. final weights in the RWLS procedure at the end of the fast-S algorithm)
outFlag  outlier flags: 1 if the robust distance of the observation exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of Y; 0 otherwise
Y  copy of the data argument as a matrix

Author(s)

Gert Willems, Stefan Van Aelst and Ella Roelant
References


See Also

- `plot.FRBPca`, `summary.FRBPca`, `print.FRBPca`, `FRBPcaMM`, `Sboot_loccov`, `Scontrol`  

Examples

```r
data(ForgedBankNotes)

Spcares <- FRBPcaS(ForgedBankNotes, R=999, bdp=0.25, conf=0.95)
# or using the formula interface
## Not run: Spcares <- FRBPcaMM(~., data=ForgedBankNotes, R=999, conf=0.95)

# the simple print method shows the standard deviations with confidence limits:
Spcares

# the summary functions shows a lot more (see help(summary.FRBPca)):
summary(Spcares)

# ask for the eigenvalues:
Spcares$eigval

# or, in more pretty format, with confidence limits:
summary(Spcares)$eigvals

# note that the standard deviations of the print-output can also be asked for by:
sqrt(summary(Spcares)$eigvals )

# the eigenvectors and their standard errors:
Spcares$eigvec   # or prettier: summary(MMpcares)$eigvecs
```
Spcares$eigvec.SE

# take a look at the bootstrap distribution of the first eigenvalue
hist(Spcares$eigval.boot[1,])
# that bootstrap distribution is used to compute confidence limits as depicted
# by the screeplot function:
plotFRBvars(Spcares, cumul=0)

# all plots for the FRB-PCA result:
plot(Spcares)

GSboot_multireg  Fast and Robust Bootstrap for GS-Estimates

Description
Calculates bootstrapped GS-estimates and bootstrap confidence intervals using the Fast and Robust Bootstrap method.

Usage
GSboot_multireg(X, Y, R = 999, conf=0.95, ests = GSest_multireg(X, Y))

Arguments
X  a matrix or data frame containing the explanatory variables (possibly including intercept).
Y  a matrix or data frame containing the response variables.
R  number of bootstrap samples. Default is R=999.
conf confidence level of the bootstrap confidence intervals. Default is conf=0.95.
est  GS-estimates as returned by GSest_multireg().

Details
Called by FRBmultiregGS and typically not to be used on its own. If no original GS-estimates are provided the function calls GSest_multireg with its default settings.

The fast and robust bootstrap was first introduced by Salibian-Barrera and Zamar (2002) for univariate regression MM-estimators and developed for GS-estimates by Roelant et al. (2009).

The value centered gives a matrix with R columns and \( p\times q + q\times q \) rows (\( p \) is the number of explanatory variables and \( q \) is the number of response variables), containing the recalculated GS-estimates. Each column represents a different bootstrap sample. The first \( p \times q \) rows are the recalculated coefficient estimates and the next \( q \times q \) rows are the covariance estimates (the estimates are vectorized, i.e. columns stacked on top of each other). These bootstrap estimates are centered by the original estimates, which are also returned through vecest in vectorized form.

The output list further contains bootstrap standard errors, as well as so-called basic bootstrap confidence intervals and bias corrected and accelerated confidence intervals (Davison and Hinkley, 1997, 1997).
p.194 and p.204 respectively). Also in the output are p-values defined as 1 minus the smallest confidence level for which the confidence intervals would include the (hypothesised) value of zero. Both BCa and basic bootstrap p-values are given. These are only useful for the regression coefficient estimates (not really for the covariance estimates).

Bootstrap samples which contain too few distinct observations with positive weights are discarded (a warning is given if this happens). The number of samples actually used is returned via ROK.

Value

A list containing the following components:

- **centered**: a matrix of all fast and robust bootstrap recalculations where the recalculations are centered by the original estimates (see Details)
- **vecest**: a vector containing the original estimates stacked on top of each other
- **SE**: bootstrap standard errors for the estimates in vecest
- **cov**: bootstrap covariance matrix for the estimates in vecest
- **CI.bca**: a matrix containing bias corrected and accelerated confidence intervals, corresponding to the estimates in vecest (first column are lower limits, second column are upper limits)
- **CI.basic**: a matrix containing basic bootstrap intervals, corresponding to the estimates in vecest (first column are lower limits, second column are upper limits)
- **p.bca**: a vector containing p-values based on the bias corrected and accelerated confidence intervals (corresponding to the estimates in vecest)
- **p.basic**: a vector containing p-values based on the basic bootstrap intervals (corresponding to the estimates in vecest)
- **ROK**: number of bootstrap samples actually used (i.e. not discarded due to too few distinct observations with positive weight)

Author(s)

Ella Roelant, Stefan Van Aelst and Gert Willems

References

**GSest_multireg**

*GS Estimates for Multivariate Regression*

**Description**
Computes GS-Estimates of multivariate regression based on Tukey’s biweight function.

**Usage**

```r
## S3 method for class 'formula'
GSest_multireg(formula, data=NULL, ...)

## Default S3 method:
GSest_multireg(X, Y, int = TRUE, bdp = 0.5, control=GScontrol(...),
na.action=na.omit, ...)```

**Arguments**

- `formula`: an object of class `formula`; a symbolic description of the model to be fit.
- `data`: data frame from which variables specified in formula are to be taken.
- `X`: a matrix or data frame containing the explanatory variables.
- `Y`: a matrix or data frame containing the response variables.
- `int`: logical: if TRUE an intercept term is added to the model (unless it is already present in X).
- `bdp`: required breakdown point. Should have $0 < bdp \leq 0.5$, the default is 0.5.
- `control`: a list with control parameters for tuning the computing algorithm, see `GScontrol()`.
- `na.action`: a function which indicates what should happen when the data contain NAs. Defaults to `na.omit`.
- `...`: allows for specifying control parameters directly instead of via `control`.

**Examples**

```r
data(schooldata)
school.x1 <- data.matrix(schooldata[,1:2])
school.y <- data.matrix(schooldata[,6:8])

# computes 10 bootstrap recalculation starting from the GS-estimator
# obtained from GSest_multireg
## Not run:
bootres <- GSboot_multireg(school.x1,school.y,R=5)
## End(Not run)
```
Details

Generalized S-estimators are defined by minimizing the determinant of a robust estimator of the scatter matrix of the differences of the residuals. Hence, this procedure is intercept free and only gives an estimate for the slope matrix. To estimate the intercept, we use the M-type estimator of location of Lopuhaa (1992) on the residuals with the residual scatter matrix estimate of the residuals as a preliminary estimate. We use a fast algorithm similar to the one proposed by Salibian-Barrera and Yohai (2006) for the regression case. See GScontrol for the adjustable tuning parameters of this algorithm.

The returned object inherits from class mlm such that the standard coef, residuals, fitted and predict functions can be used.

Value

An object of class FRBmultireg which extends class mlm and contains at least the following components:

- coefficients: GS-estimates of the regression coefficients
- residuals: the residuals, that is response minus fitted values
- fitted.values: the fitted values.
- Sigma: GS-estimate of the error covariance matrix
- Gamma: GS-estimate of the error shape matrix
- scale: GS-estimate of the size of the multivariate errors
- weights: implicit weights corresponding to the GS-estimates (i.e. final weights in the RWLS procedure for the intercept estimate)
- outFlag: outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise
- b,c: tuning parameters used in Tukey biweight loss function, as determined by bdp
- method: a list with following components: est = character string indicating that GS-estimates were used, and bdp = a copy of the bdp argument
- control: a copy of the control argument

Author(s)

Ella Roelant, Gert Willems and Stefan Van Aelst

References


**See Also**

diagplot.FRBmultireg, FRBmultiregGS, GSboot_multireg, Sest_multireg, GScontrol

**Examples**

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])
GSeest <- GSeest_multireg(school.x,school.y,nsamp=50)
# or using the formula interface
## Not run: GSeests <- GSeest_multireg(cbind(reading,mathematics,selfesteem)~., data=schooldata)

---

**hemophilia**

*Hemophilia Data*

**Description**

The hemophilia data set contains two measured variables on 75 women, belonging to two groups: \( n_1 = 30 \) of them are non-carriers (normal group) and \( n_2 = 45 \) are known hemophilia A carriers (obligatory carriers).

**Usage**

data(hemophilia)

**Format**

A data frame with 75 observations on the following 3 variables.

- **AHFactivity** AHF activity
- **AHFantigen** AHF antigen
- **gr** group - normal or obligatory carrier

**Details**

Originally analyzed in the context of discriminant analysis by Habemma and Hermans (1974). The objective is to find a procedure for detecting potential hemophilia A carriers on the basis of two measured variables: \( X_1 = \log_{10}(\text{AHV activity}) \) and \( X_2 = \log_{10}(\text{AHV-like antigen}) \). The first group of \( n_1 = 30 \) women consists of known non-carriers (normal group) and the second group of \( n_2 = 45 \) women is selected from known hemophilia A carriers (obligatory carriers). This data set was also analyzed by Johnson and Wichern (1998) as well as, in the context of robust Linear Discriminant Analysis by Hawkins and McLachlan (1997) and Hubert and Van Driessen (2004).
Source

References

Examples
data(hemophilia)

---

**MMboot_loccov**

*Fast and Robust Bootstrap for MM-estimates of Location and Covariance*

**Description**
Calculates bootstrapped MM-estimates of multivariate location and scatter using the Fast and Robust Bootstrap method.

**Usage**

```r
MMboot_loccov(Y, R = 999, ests = MMest_loccov(Y))
```

**Arguments**
- `Y` matrix or data frame.
- `R` number of bootstrap samples. Default is `R=999`.
- `ests` original MM-estimates as returned by `MMest_loccov`.

**Details**
This function is called by `FRBpcaMM` and `FRBhotellingMM`, it is typically not to be used on its own. It requires the MM-estimates of multivariate location and scatter/shape (the result of `MMest_loccov` applied on `Y`), supplied through the argument `ests`. If `ests` is not provided, `MMest_loccov` calls the implementation of the multivariate MM-estimates in package `rrcov` of Todorov and Filzmoser (2009) with default arguments.
For multivariate data the fast and robust bootstrap was developed by Salibian-Barrera, Van Aelst and Willems (2006).

The value centered gives a matrix with \( R \) columns and \( 2 \times (p + p \times p) \) rows (\( p \) is the number of variables in \( Y \)), containing the recalculated estimates of the MM-location, MM-shape, S-covariance and S-location. Each column represents a different bootstrap sample. The first \( p \) rows are the MM-location estimates, the next \( p \times p \) rows are the MM-shape estimates (vectorized). Then the next \( p \times p \) rows are the S-covariance estimates (vectorized) and the final \( p \) rows are the S-location estimates. The estimates are centered by the original estimates, which are also returned through \texttt{mmest} in vectorized form.

**Value**

A list containing:

- \texttt{centered} recalculated MM- and S-estimates of location and scatter (centered by original estimates), see Details
- \texttt{mmest} original MM- and S-estimates of location and scatter, see Details

**Author(s)**

Gert Willems, Ella Roelant and Stefan Van Aelst

**References**


**See Also**

\texttt{frbpcamm}, \texttt{frbhotellingMM}, \texttt{sboot_loccov}

**Examples**

```r
Y <- matrix(rnorm(50*5), ncol=5)
MMests <- mmest_loccov(Y)
bootresult <- MMboot_loccov(Y, R = 1000, ests = MMests)
```
**MMboot_multireg**  
*Fast and Robust Bootstrap for MM-Estimates of Multivariate Regression*

**Description**
Calculates bootstrapped MM-estimates of multivariate regression and corresponding bootstrap confidence intervals using the Fast and Robust Bootstrap method.

**Usage**

```r
MMboot_multireg(x, y, R = 999, conf = 0.95, ests = MMest_multireg(x, y))
```

**Arguments**
- `x`: a matrix or data frame containing the explanatory variables (possibly including intercept).
- `y`: a matrix or data frame containing the response variables.
- `R`: number of bootstrap samples. Default is `R=999`.
- `conf`: level of the bootstrap confidence intervals. Default is `conf=0.95`.
- `ests`: MM-estimates as returned by `MMest_multireg()`.

**Details**
Called by `FRBmultiregMM` and typically not to be used on its own. It requires the result of `MMest_multireg` applied on `x` and `y`, supplied through the argument `ests`. If `ests` is not provided, `MMest_multireg` will be called with default arguments.

The fast and robust bootstrap was first developed by Salibian-Barrera and Zamar (2002) for univariate regression MM-estimators and extended to multivariate regression by Van Aelst and Willems (2005).

The value centered gives a matrix with `R` columns and `2 * (p * q + q * q)` rows (`p` is the number of explanatory variables and `q` the number of response variables), containing the recalculated MM-estimates and initial S-estimates. Each column represents a different bootstrap sample.

The first `p * q` rows are the MM-coefficient estimates, the next `q * q` rows represent the MM-estimate of the error shape matrix (having determinant 1). Then the next `q * q` rows are the S-estimate of error covariance and the final `p * q` rows are the S-estimates of the regression coefficients (all estimates are vectorized, i.e. columns stacked on top of each other). These estimates are centered by the original estimates, which are also returned through `vecest` in vectorized form.

The output list further contains bootstrap standard errors, as well as so-called basic bootstrap confidence intervals and bias corrected and accelerated confidence intervals (Davison and Hinkley, 1997, p.194 and p.204 respectively). Also in the output are p-values defined as 1 minus the smallest confidence level for which the confidence intervals would include the (hypothesised) value of zero. Both BCa and basic bootstrap p-values are given. These are only useful for the regression coefficient estimates (not really for the covariance estimates).
Bootstrap samples which contain less than \( p \) distinct observations with positive weights are discarded (a warning is given if this happens). The number of samples actually used is returned via \( \text{rok} \).

**Value**

A list containing the following components:

- `centered`: a matrix of all fast/robust bootstrap recalculations where the recalculations are centered by original estimates (see Details)
- `vecest`: a vector containing the original estimates (see Details)
- `SE`: bootstrap standard errors for the estimates in `vecest`
- `cov`: bootstrap covariance matrix for the estimates in `vecest`
- `CI.bca`: a matrix containing 95% bias corrected and accelerated confidence intervals corresponding to the estimates in `vecest` (first column are lower limits, second column are upper limits)
- `CI.basic`: a matrix containing 95% basic bootstrap intervals corresponding to the estimates in `vecest` (first column are lower limits, second column are upper limits)
- `p.bca`: a vector containing p-values based on the bias corrected and accelerated confidence intervals (corresponding to the estimates in `vecest`)
- `p.basic`: a vector containing p-values based on the basic bootstrap intervals (corresponding to the estimates in `vecest`)
- `rok`: number of bootstrap samples actually used (i.e. not discarded due to too few distinct observations with positive weight)

**Author(s)**

Gert Willems, Ella Roelant and Stefan Van Aelst

**References**


**See Also**

`FRBmultiregMM`, `MMest_multireg`, `Sboot_multireg`
**Examples**

```r
data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

# computes 1000 bootstrap recalculations starting from the MM-estimator
# obtained from MMest_multireg()
bootres <- MMboot_multireg(school.x, school.y, R = 1000)
```

---

**MMboot_twosample**

*Fast and Robust Bootstrap for Two-Sample MM-estimates of Location and Covariance*

**Description**

Calculates bootstrapped two sample MM-estimates using the Fast and Robust Bootstrap method.

**Usage**

```r
MMboot_twosample(X, groups, R = 999, ests = MMest_twosample(X, groups))
```

**Arguments**

- `X`: matrix of data frame.
- `groups`: vector of 1’s and 2’s, indicating group numbers.
- `R`: number of bootstrap samples. Default is `R = 999`.
- `ests`: original MM-estimates as returned by `MMest_twosample()`.

**Details**

This function is called by `FRBhotellingMM`, it is typically not to be used on its own. It requires the result of `MMest_twosample` applied on `X`, supplied through the argument `ests`. If `ests` is not provided, `MMest_twosample` will be called with default arguments.

The fast and robust bootstrap was first developed by Salibian-Barrera and Zamar (2002) for univariate regression MM-estimators and extended to the two sample setting by Roelant et al. (2008).

The value centered gives a matrix with `R` columns and `2 * (2 * p + p * p)` rows (where `p` is the number of variables in `X`), containing the recalculated estimates of the MM-locations, MM-shape, S-covariance and S-locations. Each column represents a different bootstrap sample. The first `p` rows are the MM-location estimates of the first sample, the next `p` rows are the MM-location estimates of the second sample, the next `p * p` rows are the common MM-shape estimates (vectorized). Then the next `p * p` rows are the common S-covariance estimates (vectorized), the next `p` are the S-location estimates of the first sample, the final `p` rows are the S-location estimates of the second sample. The estimates are centered by the original estimates, which are also returned through `MMest` in vectorized form.
Value

A list containing:

- `centered` recalculated two sample MM- and S-estimates of location and scatter (centered by original estimates), see Details
- `Mnest` original two sample MM- and S-estimates of location and scatter, see Details

Author(s)

Ella Roelant, Gert Willems and Stefan Van Aelst

References


See Also

See Also `FRBhotellingMM`, `Sboot_twosample`

Examples

```r
Y1 <- matrix(rnorm(50*5), ncol=5)
Y2 <- matrix(rnorm(50*5), ncol=5)
Ybig <- rbind(Y1, Y2)
grp <- c(rep(1,50), rep(2,50))
Mest <- mmest_twosample(Ybig, grp)
bootresult <- MMboot_twosample(Ybig, grp, R=500, ests=Mest)
```
\textbf{mmest_multireg} \quad \textit{MM-Estimates for Multivariate Regression}

\section*{Description}
Computes MM-Estimates of multivariate regression, using initial S-estimates

\section*{Usage}

\texttt{## S3 method for class 'formula'}
\texttt{mmest_multireg(formula, data=NULL, ...)}

\texttt{## Default S3 method:}
\texttt{mmest_multireg(X, Y, int = TRUE, control=mmcontrol(...), na.action=na.omit, ...)}

\section*{Arguments}

\texttt{formula} \quad an object of class \texttt{formula}; a symbolic description of the model to be fit.

\texttt{data} \quad data frame from which variables specified in formula are to be taken.

\texttt{X} \quad a matrix or data frame containing the explanatory variables (possibly including intercept).

\texttt{Y} \quad a matrix or data frame containing the response variables.

\texttt{int} \quad logical: if \texttt{TRUE} an intercept term is added to the model (unless it is already present in \texttt{X})

\texttt{control} \quad a list with control parameters for tuning the MM-estimate and its computing algorithm, see \texttt{mmcontrol}().

\texttt{na.action} \quad a function which indicates what should happen when the data contain NAs. Defaults to \texttt{na.omit}.

\texttt{...} \quad allows for specifying control parameters directly instead of via \texttt{control}

\section*{Details}
This function is called by \texttt{FRBmultiregMM}.

The MM-estimates are defined by first computing S-estimates of regression, then fixing the scale component of the error covariance estimate, and finally re-estimating the regression coefficients and the shape part of the error covariance by more efficient M-estimates (see Tatsuoka and Tyler (2000) for MM-estimates in the special case of location/scatter estimation, and Van Aelst and Willems (2005) for S-estimates of multivariate regression). Tukey's biweight is used for the loss functions. By default, the first loss function (in the S-estimates) is tuned in order to obtain 50\% breakdown point. The default tuning of the second loss function (M-estimates) ensures 95\% efficiency at the normal model for the coefficient estimates. The desired efficiency can be changed via argument \texttt{control}. 
The computation of the S-estimates is performed by a call to `sest_multireg`, which uses the fast-S algorithm. See `mmcontrol()` to see or change the tuning parameters for this algorithm. The M-estimate part is computed through iteratively reweighted least squares (RWLS).

Apart from the MM-estimate of the regression coefficients, the function returns both the MM-estimate of the error covariance Sigma and the corresponding shape estimate Gamma (which has determinant equal to 1). Additionally, the initial S-estimates are returned as well (their Gaussian efficiency is usually lower than the MM-estimates but they may have a lower bias).

The returned object inherits from class `mlm` such that the standard `coef`, `residuals`, `fitted` and `predict` functions can be used.

**Value**

An object of class `FRBmultireg` which extends class `mlm` and contains at least the following components:

- **coefficients**: MM-estimates of the regression coefficients
- **residuals**: the residuals, that is response minus fitted values
- **fitted.values**: the fitted values.
- **Sigma**: MM-estimate of the error covariance matrix
- **Gamma**: MM-estimate of the error shape matrix
- **scale**: S-estimate of the size of the multivariate errors
- **weights**: implicit weights corresponding to the MM-estimates (i.e. final weights in the RWLS procedure)
- **outFlag**: outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise
- **c0,b,c1**: tuning parameters of the loss functions (depend on control parameters bdp and eff)
- **method**: a list with following components: `est` = character string indicating that GS-estimates were used, `bdp` = a copy of the bdp argument, `eff` a copy of the eff argument
- **control**: a copy of the control argument
- **Sbeta**: S-estimate of the regression coefficient matrix
- **SSigma**: S-estimate of the error covariance matrix
- **SGamma**: S-estimate of the error shape matrix

**Author(s)**

Gert Willems, Stefan Van Aelst and Ella Roelant
References


See Also

FRBmultiregMM, MMboot_multireg, Sest_multireg, MMcontrol

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

# compute 95% efficient MM-estimates
MMres <- MMest_multireg(school.x,school.y)
# or using the formula interface
MMres <- MMest_multireg(cbind(reading,mathematics,selfesteem)-., data=schooldata)

# the MM-estimate of the regression coefficient matrix:
MMres$coefficients
# or alternatively
coef(MMres)

n <- nrow(schooldata)
par(mfrow=c(2,1))
# the estimates can be considered as weighted least squares estimates with the
# following implicit weights
plot(1:n, MMres$weights)
# Sres$outflag tells which points are outliers based on whether or not their
# robust distance exceeds the .975 chi-square cut-off:
plot(1:n, MMres$outFlag)
# (see also the diagnostic plot in plotDiag())
Usage

```r
## S3 method for class 'FRBhot'
plot(x, ...)
```

Arguments

- `x`: an R object of class `FRBhot`, typically created by `FRBhotellingS` or `FRBhotellingMM`.
- `...`: potentially more arguments.

Details

The generic `plot` function presents two graphs. The first (top panel) is a histogram representing the test statistics in the bootstrap samples, which estimate the null distribution. A red line indicates the test statistic in the original sample (but is not shown when this value exceeds 100).

The second (bottom panel) displays the simultaneous confidence intervals based on the same bootstrap result. The intervals are scaled such that they all have the same length. Furthermore, in case of the one-sample test the intervals are shown relative to the hypothesized value $\mu_0$. Such visualization is meant to easily recognize the extent to which each variable is responsible for the overall deviation from the hypothesized value.

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst

References


See Also

`FRBhotellingS`, `FRBhotellingMM`

Examples

```r
## One sample robust Hotelling test
data(ForgedBankNotes)
samplemean <- apply(ForgedBankNotes, 2, mean)
res = FRBhotellingS(ForgedBankNotes, mu0=samplemean, R=99)
plot(res)

# Note that the test rejects the hypothesis that the true mean equals the
# sample mean; this is due to outliers in the data (i.e. the robustly estimated
# center apparently significantly differs from the non-robust sample mean.

# It is clear from the scaled simultaneous confidence limits that the rejection
# of the hypothesis is due to the differences in variables Bottom and Diagonal
```
# For comparison, the hypothesis would be accepted if only the first three variables were considered:
res = FRBHotellingS(ForgedBankNotes[,1:3], muX=samplemean[1:3],R=99)
plot(res)

## Two sample robust Hotelling test
data(hemophilia)
res <- FRBHotellingMM(cbind(AHFactActivity,AHFAntigen) ~ gr, data=hemophilia,R=99)
plot(res)

# From the confidence limits it can be seen that the significant difference is mainly caused by the AHFactActivity variable.
# the red line on the histogram indicates the test statistic value in the original sample (it is omitted if the statistic exceeds 100)

---

**plot.FRBmultireg**  
*Plot Method for Objects of class 'FRBmultireg'*

**Description**

Plot function for objects of class FRBmultireg. It produces histograms for the bootstrap estimates for all (or a selection) of the regression coefficients, based on Fast and Robust Bootstrap and with visualization of bootstrap confidence limits.

**Usage**

```r
## S3 method for class 'FRBmultireg'
plot(x, expl, resp, confmethod = c("BCA","basic"), onepage = TRUE, ...)
```

**Arguments**

- `x`: an R object of class FRBmultireg, typically created by `FRBmultiregS`, `FRBmultiregMM` or `FRBmultiregGS`
- `expl`: optional; vector specifying the explanatory variables to be shown (either by index or by variable name)
- `resp`: optional; vector specifying the response variables to be shown (either by index or by variable name)
- `confmethod`: which kind of bootstrap confidence intervals to be displayed: 'BCA'= bias corrected and accelerated method, 'basic'= basic bootstrap method
- `onepage`: logical: if TRUE, all requested histograms are plotted on one page; if FALSE, separate pages are used for each response variable
- `...`: potentially more arguments to be passed
**Details**

With \(p\) and \(q\) the number of explanatory resp. response variables specified, the function by default (i.e. if `onelpage=TRUE`) plots a \(p\) by \(q\) matrix of histograms, showing the bootstrap recalculations of the corresponding entry in the regression coefficient matrix as provided in \(x\). The original estimates for the coefficients are indicated by dotted lines, while the solid lines are the bootstrap confidence limits. In case the interval does not contain zero, the plot title is printed in red and a star is added, indicating significance.

However, if \(p\) and/or \(q\) are large, the histograms may not fit on the page and an attempt to do it may result in an error. Therefore, the function first tries whether it fits (the outcome is platform-dependent), and if not it reduces \(p\) and/or \(q\) until all plots do fit on the page. Hence, only a selection may be shown and the user is given a warning in that case.

If `onelpage=FALSE`, separate pages are used for each response variable and the user is prompted for page change. In case the number \((p)\) of explanatory variables is very large, the function again may show only a selection.

**Author(s)**

Gert Willems and Ella Roelant

**References**


**See Also**

`FRBmultiregS, FRBmultiregMM, FRBmultiregGS, summary.FRBoinitreg`

**Examples**

data(schoolda)
school.x <- data.matrix(schoolda[,1:5])
school.y <- data.matrix(schoolda[,6:8])

Sres <- FRBmultiregS(school.x, school.y, R=999, bdp = 0.25, conf = 0.99)

plot(Sres)

# the plot command above selected a subset, since otherwise an error may occur;
# as may happen when you explicitly ask for all coefficients to be plotted on one page:
## Not run:
plot(Sres, expl=1:6, resp=1:3)
## End(Not run)

# use separate pages for each response in case of many covariates:
plot(Sres, onelpage=FALSE)
#perhaps specify some specific variables of interest:
plot(Sres, expl=c("education", "occupation"), resp=c("selfesteem","reading"))
#or (the same):
plot(Sres, expl=2:3, resp=c(3,1))

plot.FRBpca

Plot Method for Objects of class 'FRBpca'

Description

Plot functions for FRBpca objects: plots PC variances, PC angles and PC loadings, with bootstrap inference

Usage

## S3 method for class 'FRBpca'
plot(x, which = 1:3, pcs.loadings = 1:min(5, length(x$eigval)),
     confmethod = c("BCA","basic"), ...)

plotFRBvars(x, cumul = 2, confmethod = c("BCA","basic"),
            npcs = min(10, length(x$eigval)))
plotFRBangles(x, pcs = 1:min(12,length(x$eigval)))
plotFRBloadings(x, confmethod = c("BCA","basic"),
                pcs = 1:min(5, length(x$eigval)), nvars=min(10, length(x$eigval)))

Arguments

x an R object of class FRBpca, typically created by FRBpcaS or FRBpcaMM
which integer number(s) between 1 and 3 to specify which plot is desired (1 = variances; 2 = angles; 3 = loadings)
pcs.loadings integer number(s) indicating for which of the PCs the loadings should be shown (in case the which argument contains 2)
cumul integer between 0 and 2: 0 = screeplot, i.e. the variances of the PCs are shown; 1 = the cumulative variances (percentage) of the PCs are shown; 2 = (default) both plots are shown on the same page
confmethod which kind of bootstrap confidence intervals to be displayed: 'BCA'= bias corrected and accelerated method, 'basic'= basic bootstrap method
npcs number of PCs to be included in screeplot/cumulative variances plot
pcs PCs to consider in plot; defaults to first 12 (maximally) for plotFRBangles; defaults to first 5 for plotFRBloading (each PC is on a separate page here)
nvars number of variables for which loadings should be shown in each PC; the loadings are shown in decreasing order in each PC
... potentially more arguments
plot.FRBpca

Details

The generic plot function calls plotFRBvars, plotFRBangles and plotFRBloadings, according to which of these are respectively specified in argument which, and displays the plots on separate pages (the user is prompted for each new page). The PCs for which the loadings should be plotted can be specified through the pcs.loadings argument. The other arguments are set to their default values by plot.

The solid curves displayed by plotFRBvars indicate the actual estimates of the variances (or percentages), while the dashed curves represent the confidence limits as computed by FRBpcaS or FRBpcaMM.

plotFRBangles plots, for each PC, histograms of the angles between the bootstrapped PC and the original PC estimate. The angles are in radians, between 0 and pi/2. These limits are indicated by the red vertical lines. Angles close to zero correspond to bootstrapped PCs closely aligned with the original PC, while an angle close to pi/2 means the bootstrapped PC is roughly perpendicular to the original estimate (hence a large number of angles close to pi/2 implies high variability). If the number of PCs specified in pcs is very large (usually larger than the default settings), the histograms may not fit on one page and a selection will be made (the user will be given a warning in that case).

In plotFRBloadings, the red dots represent the loadings, which are between -1 and 1. The square brackets indicate the confidence limits as computed by FRBpcaS or FRBpcaMM. Only the loadings of the first nvars variables are shown, where the variables were ordered according to the absolute value of the loading (i.e. only the nvars most important variables for that particular PC are shown).

Author(s)

Gert Willems and Ella Roelant

References


See Also

FRBpcaS, FRBpcaMM, summary.FRBpca

Examples

data(ForgedBankNotes)

MMpcares <- FRBpcaMM(ForgedBankNotes, R=999, conf=0.95)
plot(MMpcares)

# a closer look at the screeplot, specifying basic bootstrap intervals
plotFRBvars(MMpcares, cumul=0, confmethod="basic")

# plots the bootstrap angles for the first PC only
plotFRBangles(MMpcares, pcs=1)
# plots the loadings, with basic bootstrap intervals, for *all* the PCs
plotFRBloadings(MMpcares, confmethod="basic", pcs=1:ncol(ForgedBankNotes))

print.FRBmultireg  

Print Method for Objects of Class 'FRBmultireg'

Description

This is the print method for objects of class FRBmultireg.

Usage

## S3 method for class 'FRBmultireg'
print(x, digits = 3, ...)

Arguments

x  
an R object of class FRBmultireg, typically created by FRBmultiregS, FRBmultiregMM  
or FRBmultiregGS or by GSest_multireg, MMest_multireg or Sest_multireg

digits  
number of digits for printing (defaulting to 3)

...  
potentially more arguments

Details

The print method for FRBmultireg displays the estimated coefficients of the multivariate linear regression model.

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst

References


See Also

summary.FRBMultireg, FRBmultiregS, FRBmultiregMM, FRBmultiregGS, GSest_multireg, MMest_multireg, Sest_multireg

Examples

data(schooldata)
MMres <- FRBmultiregMM(cbind(reading, mathematics, selfesteem)~., data=schooldata)

MMres # -> print.FRBmultireg() method
print.FRBpca

Description

This is the print method for objects of class FRBpca.

Usage

```
## S3 method for class 'FRBpca'
print(x, digits=3, ...)
```

Arguments

- `x`: an R object of class FRBpca, typically created by `FRBpcaS` or `FRBpcaMM`
- `digits`: number of digits for printing (defaulting to 3)
- `...`: potentially more arguments

Details

The print method for 'FRBpca' displays the estimated standard deviations of the principal components, with corresponding bootstrap confidence intervals (BCa method).

Author(s)

Gert Willems and Ella Roelant

References


See Also

`FRBpcaS, FRBpcaMM, summary.FRBPca`

Examples

```R
data(ForgedBankNotes)

mpcares <- FRBpcaMM(ForgedBankNotes, R=999, conf=0.95)
mpcares # -> print.FRBpca() method
```
Sboot_loccov

Fast and Robust Bootstrap for S-estimates of location/covariance

Description

Calculates bootstrapped S-estimates using the Fast and Robust Bootstrap method.

Usage

Sboot_loccov(y, R = 999, ests = Sest_loccov(y))

Arguments

Y     matrix or data frame.
R     number of bootstrap samples. Default is R=999.
estss original S-estimates as returned by Sest_loccov().

Details

This function is called by FRBpcaS and FRBhotellings, it is typically not to be used on its own. It requires the S-estimates of multivariate location and scatter/shape (the result of Sest_loccov applied on Y), supplied through the argument ests. If ests is not provided, Sest_loccov calls the implementation of the multivariate S-estimates in package rrcov of Todorov and Filzmoser (2009) with default arguments.

For multivariate data the fast and robust bootstrap was developed by Salibian-Barrera, Van Aelst and Willems (2006).

The value centered gives a matrix with R columns and \( p + p \times p \) rows (\( p \) is the number of variables in \( Y \)), containing the recalculated estimates of the S-location and -covariance. Each column represents a different bootstrap sample. The first \( p \) rows are the location estimates and the next \( p \times p \) rows are the covariance estimates (vectorized). The estimates are centered by the original estimates, which are also returned through Sest.

Value

A list containing:

centered recalculated estimates of location and covariance (centered by original estimates)
Sest original estimates of location and covariance

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst
References


See Also

FRBpcaS, FRBhotellingS, Mmboot_loccov

Examples

```r
Y <- matrix(rnorm(50*5), ncol=5)
Sests <- Sest_loccov(Y, bdp = 0.25)
bootresult <- Sboot_loccov(Y, R = 1000, ests = Sests)
```

**Sboot_multireg**

*Fast and Robust Bootstrap for S-Estimates of Multivariate Regression*

Description

Calculates bootstrapped S-estimates of multivariate regression and corresponding bootstrap confidence intervals using the Fast and Robust Bootstrap method.

Usage

Sboot_multireg(X, Y, R = 999, conf=0.95, ests = Sest_multireg(X, Y))

Arguments

- **X**
  a matrix or data frame containing the explanatory variables (possibly including intercept).
- **Y**
  a matrix or data frame containing the response variables.
- **R**
  number of bootstrap samples. Default is R=999.
- **conf**
  level of the bootstrap confidence intervals. Default is conf=0.95.
- **ests**
  S-estimates as returned by Sest_multireg().
**Details**

Called by `FRBmultireg$` and typically not to be used on its own. It requires the result of `Sest_multireg` applied on X and Y, supplied through the argument ests. If ests is not provided, `Sest_multireg` will be called with default arguments.

The fast and robust bootstrap was first developed by Salibian-Barrera and Zamar (2002) for univariate regression MM-estimators and extended to multivariate regression by Van Aelst and Willems (2005).

The value centered gives a matrix with R columns and \( p \times q + q \times q \) rows (\( p \) is the number of explanatory variables and \( q \) the number of response variables), containing the recalculated S-estimates of the regression coefficients and the error covariance matrix. Each column represents a different bootstrap sample. The first \( p \times q \) rows are the coefficient estimates, the next \( q \times q \) rows represent the covariance estimate (the estimates are vectorized, i.e. columns stacked on top of each other). The estimates are centered by the original estimates, which are also returned through vecest in vectorized form.

The output list further contains bootstrap standard errors, as well as so-called basic bootstrap confidence intervals and bias corrected and accelerated (BCa) confidence intervals (Davison and Hinkley, 1997, p.194 and p.204 respectively). Also in the output are p-values defined as 1 minus the smallest confidence level for which the confidence intervals would include the (hypothesised) value of zero. Both BCa and basic bootstrap p-values are given. These are only useful for the regression coefficient estimates (not really for the covariance estimates).

Bootstrap samples which contain less than \( p \) distinct observations with positive weights are discarded (a warning is given if this happens). The number of samples actually used is returned via ROK.

**Value**

A list containing the following components:

- `centered`: a matrix of all fast/robust bootstrap recalculations where the recalculations are centered by original estimates (see Details)
- `vecest`: a vector containing the original estimates (see Details)
- `SE`: bootstrap standard errors for the estimates in vecest
- `cov`: bootstrap covariance matrix for the estimates in vecest
- `CI.bca`: a matrix containing bias corrected and accelerated confidence intervals corresponding to the estimates in vecest (first column are lower limits, second column are upper limits)
- `CI.basic`: a matrix containing basic bootstrap intervals corresponding to the estimates in vecest (first column are lower limits, second column are upper limits)
- `p.bca`: a vector containing p-values based on the bias corrected and accelerated confidence intervals (corresponding to the estimates in vecest)
- `p.basic`: a vector containing p-values based on the basic bootstrap intervals (corresponding to the estimates in vecest)
- `ROK`: number of bootstrap samples actually used (i.e. not discarded due to too few distinct observations with positive weight)
Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst

References


See Also

`frbmultireg`, `Sest_multireg`, `MMboot_multireg`

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

#computes 1000 bootstrap recalculations starting from the S-estimator
#obtained from Sest_multireg()
bootres <- sboot_multireg(school.x,school.y,R=1000)

Sboot_twosample

*Fast and Robust Bootstrap for Two-Sample S-estimates of Location and Covariance*

Description

Calculates bootstrapped two-sample S-estimates using the Fast and Robust Bootstrap method.

Usage

`sboot_twosample(X, groups, R = 999, ests = Sest_twosample(X, groups))`
Arguments

- **X**: matrix or data frame.
- **groups**: vector of 1’s and 2’s, indicating group numbers.
- **R**: number of bootstrap samples. Default is R=999.
- **ests**: original two-sample S-estimates as returned by `Sest_twosample()`.

Details

This function is called by `FRBhotellingS`, it is typically not to be used on its own. It requires the result of `Sest_twosample` applied on `X`, supplied through the argument `ests`. If `ests` is not provided, `Sest_twosample` will be called with default arguments.

The fast and robust bootstrap was first developed by Salibian-Barrera and Zamar (2002) for univariate regression MM-estimators and extended to the two sample setting by Roelant et al. (2008).

The value centered gives a matrix with R columns and 2 * p + p * p rows (p is the number of variables in `X`), containing the recalculated estimates of the S-location for the first and second center and common S-covariance. Each column represents a different bootstrap sample. The first p rows are the location estimates of the first center, the next p rows are the location estimates of the second center and the last p * p rows are the common covariance estimates (vectorized). The estimates are centered by the original estimates, which are also returned through `Sest`.

Value

A list containing:

- **centered**: recalculated estimates of location of first and second center and covariance (centered by original estimates)
- **Sest**: original estimates of first and second center and common covariance

Author(s)

Ella Roelant, Gert Willems and Stefan Van Aelst

References

See Also

FRBhotellingS

Examples

```r
Y1 <- matrix(rnorm(50*5), ncol=5)
Y2 <- matrix(rnorm(50*5), ncol=5)
Ybig <- rbind(Y1,Y2)
grp <- c(rep(1,50),rep(2,50))
Sests <- Sest_twosample(Ybig, grp, bdp=0.25)
bootresult <- Sboot_twosample(Ybig,grp,R=1000,ests=Sests)
```

### schooldata

**School Data**

**Description**

School Data, from Charnes et al. (1981). The aim is to explain scores on 3 different tests from 70 school sites by means of 5 explanatory variables.

**Usage**

```r
data(schooldata)
```

**Format**

A data frame with 70 observations on the following 8 variables.

- education: education level of mother as measured in terms of percentage of high school graduates among female parents
- occupation: highest occupation of a family member according to a pre-arranged rating scale
- visit: parental visits index representing the number of visits to the school site
- counseling: parent counseling index calculated from data on time spent with child on school-related topics such as reading together, etc.
- teacher: number of teachers at a given site
- reading: total reading score as measured by the Metropolitan Achievement Test
- mathematics: total mathematics score as measured by the Metropolitan Achievement Test
- selfesteem: Coopersmith Self-Esteem Inventory, intended as a measure of self-esteem

**Source**

Charnes et al. (1981)
References


Examples

data(schooldata)

---

### Scontrol

*Tuning parameters for multivariate S, MM and GS estimates*

#### Description

Tuning parameters for multivariate S, MM and GS estimates as used in FRB functions for multivariate regression, PCA and Hotelling tests. Mainly regarding the fast-(G)S algorithm.

#### Usage

```r
Scontrol(nsamp = 500, k = 3, bestr = 5, convTol = 1e-10, maxIt = 50)

MMcontrol(bdp = 0.5, eff = 0.95, shapeEff = FALSE, convTol.MM = 1e-07, maxIt.MM = 50, fastScontrols = Scontrol(...), ...)

GScontrol(nsamp = 100, k = 3, bestr = 5, convTol = 1e-10, maxIt = 50)
```

#### Arguments

- `nsamp`: number of random subsamples to be used in the fast-(G)S algorithm
- `k`: number of initial concentration steps performed on each subsample candidate
- `bestr`: number of best candidates to keep for full iteration (i.e. concentration steps until convergence)
- `convTol`: relative convergence tolerance for estimates used in (G)S-concentration iteration
- `maxIt`: maximal number of steps in (G)S-concentration iteration
- `bdp`: breakdown point of the MM-estimates; usually equals 0.5
- `eff`: Gaussian efficiency of the MM-estimates; usually set at 0.95
- `shapeEff`: logical; if TRUE, `eff` is with regard to shape-efficiency, otherwise location-efficiency
- `convTol.MM`: relative convergence tolerance for estimates used in MM-iteration
- `maxIt.MM`: maximal number of steps in MM-iteration
- `fastScontrols`: the tuning parameters of the initial S-estimate
- `...`: allows for any individual parameter from `Scontrol` to be set directly
Details
The default number of random samples is lower for GS-estimates than for S-estimates, because computations regarding the former are more demanding.

Value
A list with the tuning parameters as set by the arguments.

Author(s)
Gert Willems and Ella Roelant

References

See Also
`gsest_multireg, sest_multireg, mmest_multireg, sest_twosample, mmest_twosample, frbpcas, ...`

Examples
```r
## Show the default settings:
str(Scontrol())
str(MMcontrol())
str(GScontrol())
```

---

**Sest_multireg**  
*S-Estimates for Multivariate Regression*

Description
Computes S-Estimates of multivariate regression based on Tukey’s biweight function using the fast-S algorithm.

Usage
```r
## S3 method for class 'formula'
Sest_multireg(formula, data=NULL, ...)

## Default S3 method:
Sest_multireg(X, Y, int = TRUE, bdp = 0.5, control=Scontrol(...), na.action=na.omit, ...)```
Arguments

formula  an object of class `formula`; a symbolic description of the model to be fit.
data     data frame from which variables specified in formula are to be taken.
X         a matrix or data frame containing the explanatory variables (possibly including intercept).
Y         a matrix or data frame containing the response variables.
int       logical: if \texttt{TRUE} an intercept term is added to the model (unless it is already present in \texttt{X}).
bdp       required breakdown point. Should have \(0 < \text{bdp} \leq 0.5\), the default is 0.5.
control   a list with control parameters for tuning the computing algorithm, see \texttt{Scontrol}().
na.action a function which indicates what should happen when the data contain NAs. Defaults to \texttt{na.omit}.

...          allows for specifying control parameters directly instead of via \texttt{control}.

Details

This function is called by \texttt{FRBmultiregS}.

S-estimates for multivariate regression were discussed in Van Aelst and Willems (2005). The algorithm used here is a multivariate version of the fast-S algorithm introduced by Salibian-Barrera and Yohai (2006). See \texttt{Scontrol} for the adjustable tuning parameters of this algorithm.

Apart from the regression coefficients, the function returns both the error covariance matrix estimate \(\Sigma\) and the corresponding shape estimate \(\gamma\) (which has determinant equal to 1). The scale is determined by \(\text{det}(\Sigma)^{1/2}/q\), with \(q\) the number of response variables.

The returned object inherits from class \texttt{mlm} such that the standard \texttt{coef}, \texttt{residuals}, \texttt{fitted} and \texttt{predict} functions can be used.

Value

An object of class \texttt{FRBmultireg} which extends class \texttt{mlm} and contains at least the following components:

- \texttt{coefficients}  S-estimates of the regression coefficients
- \texttt{residuals}     the residuals, that is response minus fitted values
- \texttt{fitted.values} the fitted values.
- \texttt{Gamma}        S-estimate of the error shape matrix
- \texttt{Sigma}        S-estimate of the error covariance matrix
- \texttt{scale}        S-estimate of the size of the multivariate errors
- \texttt{weights}      implicit weights corresponding to the S-estimates (i.e. final weights in the RWLS procedure at the end of the fast-S algorithm)
- \texttt{outFlag}      outlier flags: 1 if the robust distance of the residual exceeds the .975 quantile of (the square root of) the chi-square distribution with degrees of freedom equal to the dimension of the responses; 0 otherwise
tuning parameters used in Tukey biweight loss function, as determined by bdp
a list with following components: est = character string indicating that GS-
estimates were used and bdp = a copy of the bdp argument

control a copy of the control argument

Author(s)

Gert Willems, Stefan Van Aelst and Ella Roelant

References


See Also

FRBmultiregS.Sboot_multireg.MMest_multireg.Scontrol

Examples

data(schooldata)
school.x <- data.matrix(schooldata[,1:5])
school.y <- data.matrix(schooldata[,6:8])

# compute 25% breakdown S-estimates
Sres <- sest_multireg(school.x,school.y, bdp=0.25)
# or using the formula interface
Sres <- sest_multireg(cbind(reading,mathematics,selfesteem)-., data=schooldata, bdp=0.25)

# the regression coefficients:
Sres$coefficients
# or alternatively
coeff(Sres)

n <- nrow(schooldata)
par(mfrow=c(2,1))
# the estimates can be considered as weighted least squares estimates with the
# following implicit weights
plot(1:n, Sres$weights)
# Sres$outflag tells which points are outliers based on whether or not their
# robust distance exceeds the .975 chi-square cut-off:
plot(1:n, Sres$outflag)
# (see also the diagnostic plot in plotDiag())
Summary Method for Objects of Class ‘FRBhot’

Description

Summary method for objects of class FRBhot, and print method of the summary object.

Usage

## S3 method for class 'FRBhot'
summary(object, digits = 5, ...)

## S3 method for class 'summary.FRBhot'
print(x, ...)

Arguments

- **object**: an R object of class FRBhot, typically created by FRBhotellingS or FRBhotellingMM
- **digits**: number of digits for printing (default is 5)
- **x**: an R object of class summary.FRBhot, resulting from summary(FRBhotellingS(), ...) or summary(FRBhotellingMM(), ...)
- **...**: potentially more arguments to be passed to methods

Details

The print method here displays the value of the test statistic and the corresponding bootstrap p-value. It also presents the simultaneous confidence intervals for the components of the location vector (or difference between the two location vectors), and the robust estimates for the location vector(s) and covariance matrix.

Value

summary.FRBhot simply returns its two arguments in a list.

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst

References


See Also

FRBhotellingS, FRBhotellingMM, plot.FRBhot
Examples

data(ForgedBankNotes)
samplemean <- apply(ForgedBankNotes, 2, mean)
res = FRBhotellingS(ForgedBankNotes, mu0=samplemean)

summary(res) # -> print.summary.FRBhot() method

summary.FRBMultireg  Summary Method for Objects of Class ‘FRBMultireg’

Description

Summary method for objects of class FRBMultireg, and print method of the summary object.

Usage

## S3 method for class 'FRBMultireg'
summary(object, confmethod = c("BCA", "basic", "both"), digits = 3, 
print.CI=FALSE, sep="", ...) 

## S3 method for class 'summary.FRBMultireg'
print(x, ...) 

Arguments

object an R object of class FRBMultireg, typically created by FRBMultiregS, FRBMultiregMM or FRBMultiregGS
confmethod which kind of bootstrap confidence intervals to be displayed: 'BCA'= bias corrected and accelerated method, 'basic'= basic bootstrap method, 'both'=both kinds of confidence intervals
digits number of digits for printing (default is 3)
print.CI logical: Should Confidence intervals be printed?
sep Symbol to separate columns in output. Default is ""
x an R object of class summary.FRBMultireg, resulting for example from summary(FRBMultiregS(),...) 
... potentially more arguments to be passed to methods

Details

The print method displays in a “familiar way” the components of the summary object, which are listed in the Value section.
Value

summary returns an object of class summary.FRBmultireg, which contains the following components:

responses the names of the response variables in the fitted model
covariates the names of the covariates (predictors) in the fitted model
Betawstd a data frame containing the coefficient estimates and their bootstrap standard errors
Sigma estimate for the error covariance matrix
table.bca a list with for each response variable a matrix containing the estimates, standard errors, lower and upper limits of the BCa confidence intervals, p-values and a significance code (only present when confmethod="BCa" or confmethod="both")
table.basic a list with for each response variable a matrix containing the estimates, standard errors, lower and upper limits of the basic bootstrap confidence intervals, p-values and a significance code (only present when confmethod="basic" or confmethod="both")
method multivariate regression method that was used
conf confidence level that was used
digits number of digits for printing

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst

References


See Also

FRBmultiregS, FRBmultiregMM, FRBmultiregGS, print.FRBmultireg, plot.FRBmultireg

Examples

data(schooldata)

MMres <- FRBmultiregMM(cbind(reading,mathematics,selfesteem) ~ ., data=schooldata, R=199, conf = 0.99, nsamp=200)
summary(MMres) # -> print.summary.FRBmultireg() method

GSres <- FRBmultiregGS(cbind(reading,mathematics,selfesteem) ~ ., data=schooldata, bdp = 0.25, R=199, nsamp=50)
summary(GSres, confmethod="both") # -> print.summary.FRBmultireg() method
Summary Method for Objects of Class ‘FRBpca’

Description

Summary method for objects of class FRBpca, and print method of the summary object.

Usage

```r
## S3 method for class 'FRBpca'
summary(object, confmethod = c("BCA", "basic", "both"), digits = 3, ...)
## S3 method for class 'summary.FRBPca'
print(x, ...)
```

Arguments

- `object` an R object of class FRBpca, typically created by `FRBpcaS` or `FRBpcaMM`
- `confmethod` which kind of bootstrap confidence intervals to be displayed: 'BCA'= bias corrected and accelerated method, 'basic'= basic bootstrap method, 'both'= both kinds of confidence intervals
- `digits` number of digits for printing (default is 3)
- `x` an R object of class `summary.FRBPca`, resulting from `summary(FRBpcaS(),...)` or `summary(FRBpcaMM(),...)`
- `...` potentially more arguments to be passed to methods

Details

The print method displays mostly the components of the summary object as listed in the Value section.

Value

`summary` returns an object of class `summary.FRBPca`, which contains the following components:

- `eigvals` eigenvalues of the shape estimate (variances of the principal components) with confidence limits
- `eigvecs` eigenvectors of the shape estimate (loadings of the principal components)
- `avgangle` bootstrap estimates of average angles between true and estimated eigenvectors
- `pvars` cumulative percentage of variance explained by first principal components with confidence limits
- `method` PCA method that was used
- `digits` number of digits for printing

Author(s)

Gert Willems, Ella Roelant and Stefan Van Aelst
Summary

FRBpca

References


See Also

FRBpcaS, FRBpcaMM, print.FRBPca, plot.FRBPca

Examples

data(ForgedBankNotes)

MPpcares <- FRBpcaMM(ForgedBankNotes, R=999, conf=0.95)
summary(MMpcares) # -> print.summary.FRBPca() method
Index

*Topic **datasets**
- delivery, 2
- ForgedBankNotes, 5
- hemophilia, 37
- schooldata, 59

*Topic **htest**
- FRBHotellingMM, 6
- FRBHotellingS, 10

*Topic **multivariate**
- FRBHotellingMM, 6
- FRBHotellingS, 10
- FRBmultiregGS, 13
- FRBmultiregMM, 17
- FRBmultiregS, 21
- FRBpcaMM, 25
- FRBpcaS, 29

*Topic **robust**
- FRBHotellingMM, 6
- FRBHotellingS, 10
- FRBmultiregGS, 13
- FRBmultiregMM, 17
- FRBmultiregS, 21
- FRBpcaMM, 25
- FRBpcaS, 29

**coef.** 14, 18, 22, 36, 45, 62

**delivery.** 2
**diagplot.** 3
**diagplot.FRBmultireg.** 37

**fitted.** 14, 18, 22, 36, 45, 62
**ForgedBankNotes.** 5
**formula.** 7, 10, 14, 17, 21, 25, 29, 35, 44, 62
**FRB-internal.** 6
**FRBHotellingMM.** 3, 4, 6, 12, 38, 39, 42, 43, 47, 64
**FRBHotellingS.** 3, 4, 9, 10, 47, 54, 55, 58, 59, 64

FRBmultiregGS, 3, 4, 13, 20, 24, 33, 35, 37, 48, 49, 52, 65, 66
FRBmultiregMM, 3, 4, 16, 17, 24, 40, 41, 44, 46, 48, 49, 52, 65, 66
FRBmultiregS, 3, 4, 16, 20, 21, 48, 49, 52, 56, 57, 62, 63, 65, 66
FRBpcaMM, 3, 4, 25, 38, 39, 50, 51, 53, 67, 68
FRBpcaS, 3, 4, 28, 29, 50, 51, 53–55, 61, 67, 68
GSboot_multireg, 14–16, 33, 37
GScontrol, 14, 16, 35–37
GScontrol (Scontrol), 60
GSeinsfs_multireg (FRB-internal), 6
GSeest_multireg, 3, 14–16, 33, 35, 35, 52, 61
hemophilia, 37
MMboot_loccov, 26–28, 38, 55
MMboot_multireg, 18–20, 40, 46, 57
MMboot_twosample, 42
MMcontrol, 7, 9, 18, 20, 25, 28, 44–46
MMcontrol (Scontrol), 60
MMeinsfs_multireg (FRB-internal), 6
MMeinsfs_pca (FRB-internal), 6
MMest_loccov, 38
MMest_loccov (FRB-internal), 6
MMest_multireg, 3, 18–20, 40, 41, 44, 52, 61, 63
MMest_twosample, 42, 61
MMest_twosample (FRB-internal), 6
na.omit, 7, 10, 14, 18, 22, 25, 29, 35, 44, 62
plot.FRBhot, 9, 12, 46, 64
plot.FRBmultireg, 16, 20, 24, 48, 66
plot.FRBpca, 28, 32, 50, 68
plotFRBangles (plot.FRBpca), 50
plotFRBloadings (plot.FRBpca), 50
plotFRBvars (plot.FRBpca), 50
predict, 14, 18, 22, 36, 45, 62
predict.FRBmultireg (FRB-internal), 6
print.FRBmultireg, 52, 66
print.FRBpca, 28, 32, 53, 68
print.summary.FRBhot (summary.FRBhot), 64
print.summary.FRBmultireg (summary.FRBmultireg), 65
print.summary.FRBpca (summary.FRBpca), 67

residuals, 14, 18, 22, 36, 45, 62

sboot_loccov, 30–32, 39, 54
sboot_multireg, 22–24, 41, 55, 63
sboot_twosample, 43, 57
schooldata, 59
Scontrol, 10, 12, 22, 24, 29, 30, 32, 60, 62, 63
Seinfs_multireg (FRB-internal), 6
Seinfs_pca (FRB-internal), 6
Sest_loccov, 54
Sest_loccov (FRB-internal), 6
Sest_multireg, 3, 18, 22–24, 37, 45, 46, 52, 55–57, 61, 63
Sest_twosample, 58, 61
Sest_twosample (FRB-internal), 6
summary.FRBhot, 9, 12, 64
summary.FRBmultireg, 16, 20, 24, 49, 52, 65
summary.FRBpca, 28, 32, 51, 53, 67

vcov.FRBmultireg (FRB-internal), 6