Package 'MNM'

March 3, 2018

Type Package

Title Multivariate Nonparametric Methods. An Approach Based on Spatial Signs and Ranks

Version 1.0-3

Date 2018-03-02

Author Klaus Nordhausen, Jyrki Mottonen, Hannu Oja

Maintainer Klaus Nordhausen <klaus.nordhausen@tuwien.ac.at>

Depends R (>= 2.9.2), ICSNP, SpatialNP

Imports ellipse, ICS

Suggests Flury, gamlss, mvtnorm

Description Multivariate tests, estimates and methods based on the identity score, spatial sign score and spatial rank score are provided. The methods include one and c-sample problems, shape estimation and testing, linear regression and principal components. The methodology is described in Oja (2010) <doi:10.1007/978-1-4419-0468-3> and Nordhausen and Oja (2011) <doi:10.18637/jss.v043.i05>.

License GPL (>= 2)

LazyLoad yes

Encoding latin1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-03-03 00:54:59 UTC

R topics documented:

`MNM-package` *Multivariate Nonparametric Methods. An Approach Based on Spatial Signs and Ranks.*

Description

Multivariate tests, estimates and methods based on the identity score, spatial sign score and spatial rank score are provided. The methods include one and c-sample problems, shape estimation and testing, linear regression and principal components.

Details

Package:	MNM
Type:	Package
Version:	1.0-3
Date:	2018-02-03
License:	GPL (>= 2)

The methods implemented here are mainly described in Oja (2010) and the package can be used to reproduce most of the examples in the book. The book will be refered to as the MNM book.

Author(s)

Klaus Nordhausen, Jyrki Möttönen and Hannu Oja

Maintainer: Klaus Nordhausen, <klaus.nordhausen@tuwien.ac.at>

References

Oja, H. (2010), Multivariate Nonparametric Methods with R. An Approach Based on Spatial Signs and Ranks, *Springer. <doi:10.1007/978-1-4419-0468-3>.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43****, 1-28. <doi:10.18637/jss.v043.i05>.*

`affine.trans` *Function For Affine Data Transformation*

Description

Function for transformations of the form $Ax + b$ or $A^{1/2}x + b$

Usage

```
affine.trans(X, A = diag(1, dim(X)[2]), b = rep(0, dim(X)[2]),
             A.sqrt = FALSE, na.action = na.fail)
```

Arguments

`X`	a numeric data frame or matrix with p columns.
`A`	full rank p times p matrix.
`b`	numeric vector of length p.
`A.sqrt`	logical. If `TRUE` the symmetric square root of A will be used.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Value

a matrix.

Author(s)

Klaus Nordhausen

Examples

```
data(iris)
IRIS <- iris[,1:4]
colMeans(IRIS)
cov(IRIS)
IRIS.trans <- affine.trans(IRIS, solve(cov(IRIS)), colMeans(IRIS),TRUE)
colMeans(IRIS.trans)
cov(IRIS.trans)
```

---

anova.mvl1lm *Comparisons between Multivariate Linear Models*

Description

Comparisons between nested multivariate linear models fitted by mv.l1lm. The comparison can be based on score type of tests and Wald type of tests.

Usage

```
## S3 method for class 'mvl1lm'
anova(object, object2 = NULL, test = "Score", ...)
```

Arguments

object	an object of class mvl1lm. This gives the full model.
object2	an object of class mvl1lm or NULL. This gives the restricted (nested) model.
test	The test to be used. Options are Score and Wald. The score version is the default.
...	needed for other methods.

Details

If only object is provided the function tests if all parameters equal zero. If object and object2 are provided the function tests the null hypothesis that the the restricted model (object2) is true. For details see chapter 13 of the MNM book. Note that it is the users responsibility to make sure that the two models are nested and fitted on the same data. For the regular L2 regression anova.mlm provides more options.

Value

A list with class 'anovamvl1lm' containing the following components:

`models`	the model call(s) of `object` and `object2`.
`method`	type of the test used.
`statistic`	value of the test statistic.
`parameter`	degrees of freedom.
`p.value`	p-value of the test.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

Examples

```
# creating simple data

X <- cbind(rep(1,100),rmvnorm(100,c(0,0,0)) )
B <- matrix(c(4,1,1,0.5,-3,2,2,2),ncol=4, byrow=TRUE)
Y <- X %*% t(B)+ rmvnorm(100,c(0,0), diag(0.2,2))
DAT <- data.frame(x1=X[,2],x2=X[,3], x3=X[,4])

FullModel <- mv.l1lm(Y ~ x1 + x2 + x3, scores= "s", stand="i", data=DAT)
RestModel <- mv.l1lm(Y ~ x1, scores= "s", stand="i", data=DAT)

anova(FullModel)
anova(FullModel, RestModel)
anova(FullModel, RestModel, test="W")
```

---

beans *Randomized Block Experiment of Plots of Beans*

Description

Results of a randomized block experiment in the Cook Islands involving the effect of six different treatments on plots of beans infested by the serpentine leaf miner insect.

Usage

```
data(beans)
```

Format

A data frame with 24 observations on the following 5 variables.

`Block` a factor with levels `1 2 3 4`.

`Treatment` a factor with levels `1 2 3 4 5 6`.

`y1` a numeric vector. The number of miners per leaf.

`y2` a numeric vector. The weight of beans per plot (in kg).

`y3` a numeric vector. 1/sin(sqrt(p)), where p is the proportion of leaves infested with borer.

Details

The value of variable `y3` in Block 4 for Treatment 2 is an estimate of a missing value.

Source

Data courtesy of Dr. R. Fullerton.

References

Seber, G. A. F. (1998), Multivariate Observations, *London: Arnold.*

Examples

```
data(beans)
plot(beans)
```

`coef.mvl1lm` *Coefficients of an mvl1lm Object*

Description

Extracts the coefficents of an mvl1lm object.

Usage

```
## S3 method for class 'mvl1lm'
coef(object, ...)
```

Arguments

`object`	an object of class `mvl1lm`.
`...`	needed for other methods.

Details

Note that for `rank` scores the intercept, even when specified in the model, is not considered a coefficient.

Author(s)

Klaus Nordhausen

`fitted.mvl1lm` *Fitted Values of an mvl1lm Object*

Description

Extracts the fitted values of an mvl1lm object.

Usage

```
## S3 method for class 'mvl1lm'
fitted(object, ...)
```

Arguments

`object`	an object of class `mvl1lm`.
`...`	needed for other methods.

Author(s)

Klaus Nordhausen

`mv.1sample.est` *Multivariate One Sample Location Estimates*

Description

Estimates the multivariate location for different score functions and their asymptotic covariance matrices in the one sample case.

Usage

```
mv.1sample.est(X, score = "identity", stand = "outer", maxiter = 100,
               eps = 1e-06, na.action = na.fail, ... )
```

Arguments

`X`	a numeric data frame or matrix.
`score`	the score to be used. Possible choices are `identity`, `sign` and `rank`.
`stand`	the standardization method used. Possible choices are `outer` and `inner`.
`maxiter`	maximum number of iterations. Used only for `score = "sign"` and `score = "rank"`.
`eps`	convergence tolerance. Used only for `score = "sign"` and `score = "rank"`.
`...`	arguments that can be passed on to functions used for the estimation of location.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Details

For identity scores the location estimate is the regular mean vector. For the spatial sign score it is the spatial median in the outer standardization case and the Hettmansperger-Randles estimate in the inner standardization case. The rank estimate is the spatial Hodges-Lehmann estimator, either regular (`stand = "outer"`) or affine equivariant (`stand = "inner"`).

Computation with outer standardization is faster than with inner standardization and especially the rank version might be slow and memory consuming.

For further details see chapters 3, 5, 6, 7 and 8 of the MNM book.

Value

A list with class 'mvloc' containing the following components:

`location`	the location estimate as a vector.
`vcov`	the asymptotic covariance matrix of the location estimate.
`est.name`	name of the location estimate.
`dname`	name of the data set.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), *Multivariate Nonparametric Methods with R*, Springer.

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

`spatial.sign`, `spatial.signrank`, `spatial.median`, `HR.Mest`

Examples

```
set.seed(1)
X <- rmvt(100, diag(c(1, 2, 0.5)), 3)

est.Hot.X <- mv.1sample.est(X)
est.SS.o.X <- mv.1sample.est(X,"s")
est.SS.i.X <- mv.1sample.est(X,"s","i")
est.SR.o.X <- mv.1sample.est(X,"r")
est.SR.i.X <- mv.1sample.est(X,"r","i")

est.SR.o.X
summary(est.SR.o.X)

# plotting

plot(est.Hot.X, est.SS.i.X, est.SR.i.X, X)
```

```
# or
plot(est.Hot.X, est.SS.i.X, est.SR.i.X)
```

`mv.1sample.test` *Multivariate Location Tests*

Description

Tests for multivariate location using different score functions.

Usage

```
mv.1sample.test(X, mu = 0, score = "identity", stand = "outer",
                method = "approximation", n.simu = 1000,
                na.action = na.fail)
```

Arguments

`X`	a numeric data frame or matrix.
`mu`	the null hypothesis value. Default is the zero vector.
`score`	the score to be used. Possible choices are `identity`, `sign` and `rank`.
`stand`	the standardization method used. Possible choices are `outer` and `inner`.
`method`	method for the computation of the p-value for the spatial sign and spatial signed-rank tests. Possible choices are `approximation` and `signchange`.
`n.simu`	number of simulated sign changes if `method=signchange`.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Details

The tests provided here are the Hotelling's T^2 test, the spatial sign test and the signed-rank test and their affine invariant versions in the one sample location case.

Note that for the identity score the provided test is not the traditional Hotelling's T^2 test because here the covariance matrix is computed wrt to the null value and not wrt to the sample mean. Use the function `HotellingsT2` for the traditional version of Hotelling's T^2 test. Details about the tests can be found in the chapters 5-8 of the MNM book.

Value

A list with class 'htest' containing the following components:

`statistic`	the value of the test statistic.
`parameter`	the degrees of freedom for the test statistic or the number of replications in the simulation.
`p.value`	the p-value for the test.

null.value	the specified hypothesized value of the location.
alternative	a character string with the value 'two.sided'.
method	a character string indicating what type of test was performed.
data.name	a character string giving the name of the data set.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

HotellingsT2, sr.loc.test

Examples

```
library(mvtnorm)
X <- rmvt(100, diag(c(1, 2, 0.5)), 3)

mv.1sample.test(X,mu=c(0,0,0.5))
mv.1sample.test(X,score="s", stand="i")
mv.1sample.test(X,score="s", stand="i", method="s")
mv.1sample.test(X,score="r", stand="o")
mv.1sample.test(X,score="r", stand="i")
```

---

mv.2sample.est *Multivariate Two Sample Shift Estimates*

Description

Estimates the multivariate shift for different score functions and their asymptotic covariance matrices in the two sample case.

Usage

```
mv.2sample.est(X, g, score = "identity", stand = "outer",
               maxiter = 100, eps = 1e-06, na.action = na.fail,
               ...)
```

Arguments

X	a numeric data frame or matrix.
g	a factor with two levels
score	the score to be used. Possible choices are identity, sign and rank.
stand	the standardization method used. Possible choices are outer and inner.
maxiter	maximum number of iterations. Used only for score = "sign" and score = "rank".
eps	convergence tolerance. Used only for score = "sign" and score = "rank".
na.action	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
...	arguments that can be passed on to functions used for the estimation of location.

Details

This implements the location estimates and their asymptotic covariance matrices as described in chapter 11 of the MNM book. Note that the shift is the parameter for the defference between 'values of level 1 - values of level 2' where the levels are as defined in the factor g.

For the general c sample location case the function mv.l1lm might be used.

Value

A list with class 'mvloc' containing the following components:

location	the location estimate as a vector.
vcov	the asymptotic covariance matrix of the location estimate.
est.name	name of the location estimate.
dname	name of data set for which the location was computed.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

spatial.sign, spatial.signrank

Examples

```
X1<- rmvnorm(50,c(0,0,0))
X2<- rmvnorm(70,c(1,1,2))
X<-rbind(X1,X2)
g<-factor(rep(1:2,c(50,70)))

  est.Hot.X <- mv.2sample.est(X, g)
  est.SS.o.X <- mv.2sample.est(X, g, "s")
  est.SS.i.X <- mv.2sample.est(X, g, "s", "i")
  est.SR.o.X <- mv.2sample.est(X, g, "r")
  est.SR.i.X <- mv.2sample.est(X, g, "r", "i")

  est.SS.o.X

  summary(est.SS.o.X)

  # plotting

  plotMvloc(est.Hot.X, est.SS.i.X, est.SR.i.X)
```

`mv.2way.est` *Treatment Effect Estimates in the Randomized Complete Block Case*

Description

The treatment effect estimates for different score functions and their asymptotic covariance matrices in the randomized complete block case.

Usage

```
mv.2way.est(x, block, treatment, score = c("identity", "sign", "rank"),
             stand = c("outer", "inner"),
             eps=1.0e-10, n.iter=1000, na.action = na.fail)
```

Arguments

`x`	a numeric data frame or matrix.
`block`	a factor with at least two levels.
`treatment`	a factor with at least two levels.
`score`	the score to be used. Possible choices are `identity`, `sign` and `rank`.
`stand`	the standardization method used. Possible choices are `outer` and `inner`.
`eps`	convergence criterion.
`n.iter`	maximum number of iterations.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Details

This implements the treatment effect estimates described in chapter 12 of the MNM book.

Value

A list of length c(c-1)/2 with class 'mvcloc' where c is the number of treatments. Each component of the list is a list with class 'mvloc' containing the following components:

`location`	the adjusted treatment effect estimate when comparing the treatment pair given in `dname`.
`vcov`	the asymptotic covariance matrix of the adjusted treatment effect estimate.
`est.name`	name of the adjusted treatment effect estimate.
`dname`	the treatment pair for which the adjusted treatment effect estimate was computed.

Author(s)

Jyrki Möttönen <jyrki.mottonen@helsinki.fi>

References

Oja, H. (2010), Multivariate Nonparametric Methods with R*, Springer.*

See Also

`mv.2way.test`, `mv.1sample.est`, `mv.2sample.est`

Examples

```
data(beans)
est<-mv.2way.est(beans[,3:5],beans$Block,beans$Treatment,score="r",stand="i")
summary(est)
```

---

mv.2way.test *Randomized Complete Block Design.*

Description

Multivariate tests for testing the null hypothesis that there is no treatment effect in a randomized complete block design using different scores.

Usage

```
mv.2way.test(x, block, treatment, score = c("identity", "sign",
             "rank"), stand = c("outer", "inner"),
             method = c("approximation", "permutation"),
             n.simu = 1000, eps=1.0e-10, n.iter=10000,
             na.action = na.fail)
```

Arguments

x	a numeric data frame or matrix of response variables.
block	a factor with at least two levels.
treatment	a factor with at least two levels.
score	the score to be used. Possible choices are identity, sign and rank.
stand	the standardization method used. Possible choices are outer and inner.
method	method for the computation of the p-value for the spatial sign and spatial rank tests. Possible choices are approximation and permutation.
n.simu	number of simulated permutations if method="permutation".
eps	convergence criterion.
n.iter	maximum number of iterations.
na.action	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Details

This implements the tests described in chapter 12 of the MNM book.

Value

A list with class 'htest' containing the following components:

statistic	the value of the test statistic.
parameter	the degrees of freedom for the test statistic or the number of replications in the simulation.
p.value	the p-value for the test.
null.value	the specified null hypothesis value of the location.
alternative	a character string with the value 'two.sided'.
method	a character string indicating what type of test was performed.
data.name	a character string giving the name of the data set and of the grouping vector.

Author(s)

Jyrki Möttönen <jyrki.mottonen@helsinki.fi>

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

See Also

mv.1sample.test, mv.Csample.test , mv.2way.est

Examples

```
blocks <- gl(10, 5)
treatments <- factor(rep(1:5, 10))
X <- rmvnorm(n = 50, mean = c(1,2,3), sigma = diag(3))
mv.2way.test(X, blocks, treatments, score="r", stand="i", method="a")
```

`mv.Csample.test` *C Sample Test of Location*

Description

Several samples location tests using different scores.

Usage

```
mv.Csample.test(X, g, score = "identity", stand = "outer",
                method = "approximation", n.simu = 1000,
                na.action = na.fail, ...)
```

Arguments

`X`	a numeric data frame or matrix of response values.
`g`	a factor with at least two levels.
`score`	the score to be used. Possible choices are `identity`, `sign` and `rank`.
`stand`	the standardization method used. Possible choices are `outer` and `inner`.
`method`	method for the computation of the p-value for the spatial sign and spatial signed-rank tests. Possible choices are `approximation` and `permutation`.
`n.simu`	number of simulated sign changes if `method="permutation"`.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
`...`	arguments that can be passed on to functions used for the estimation of the spatial signs and spatial ranks.

Details

This implements the location tests based on identity, sign or rank scores as described in chapter 11 of the MNM book.

Value

A list with class 'htest' containing the following components:

`statistic`	the value of the test statistic.
`parameter`	the degrees of freedom for the test statistic or the number of replications in the simulation.
`p.value`	the p-value for the test.
`null.value`	the specified hypothesized value of the location.
`alternative`	a character string with the value 'two.sided'.
`method`	a character string indicating what type of test was performed.
`data.name`	a character string giving the name of the data set and of the grouping vector.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

`spatial.sign`, `spatial.rank`, `HotellingsT2`

Examples

```
X <- rmvt(150,diag(1,3))
g1 <- gl(3,50)
mv.Csample.test(X, g1)
mv.Csample.test(X, g1, score = "s")
mv.Csample.test(X, g1, score = "r")

Y <- rbind(rmvnorm(40,c(0,0,0)), rmvnorm(60,c(0,0,0.4)))
g2 <- factor(rep(1:2, c(40, 60)))
mv.Csample.test(Y, g2, score = "r")
mv.Csample.test(Y, g2, score = "r", method="p")
```

`mv.ind.test` *Independence Test*

Description

Tests for independence of two vectors using different scores.

Usage

```
mv.ind.test(X, Y, score = "identity", method = "approximation",
            n.simu = 1000, na.action = na.fail)
```

Arguments

`X`	a numeric data frame or matrix. Must have the same number of rows as `Y`.
`Y`	a numeric data frame or matrix. Must have the same number of rows as `X`.
`score`	the score to be used. Possible are `identity`, `sign`, `symm` and `rank`.
`method`	method for for computation of the p-value for the spatial sign and spatial signed-rank tests. Possible are `approximation` and `permutation`.
`n.simu`	number of permutations if `method="permutation"`.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.

Details

This implements the independence tests as described in chapter 10 of the MNM book. Note that only inner test versions are implemented and that for the symmetrized sign score only the approximative method for the computation of the p-value is available.

Value

A list with class 'htest' containing the following components:

`statistic`	the value of the test statistic.
`parameter`	the degrees of freedom for the test statistic or the number of replications in the simulation.
`p.value`	the p-value for the test.
`null.value`	the specified null hypothesis value.
`alternative`	a character string with the value 'two.sided'.
`method`	a character string indicating what type of test was performed.
`data.name`	a character string giving the name of the two data matrices.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

Examples

```
X <- rmvt(150,diag(1,3),df=3)
Y <- rmvt(150, matrix(c(1,0.5,0.5,1),nrow=2),df=3)

mv.ind.test(X, Y)
mv.ind.test(X, Y, method = "p")

mv.ind.test(X, Y, score = "si")
mv.ind.test(X, Y, score = "si", method = "p")

mv.ind.test(X, Y, score = "r")
mv.ind.test(X, Y, score = "r", method = "p")

mv.ind.test(X, Y, score = "sy")
```

---

mv.l1lm — *Linear Regression Based on Identity, Spatial Sign or Spatial Rank Scores*

Description

This function fits a multivariate linear regression model based on identity, spatial sign or spatial rank scores. Both inner and outer standardization are possible.

Usage

```
mv.l1lm(formula, scores = "identity", stand = "outer",
        maxiter = 1000, eps = 1e-06, eps.S = 1e-06,
        x = TRUE, y = TRUE, data, subset, na.action)
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The left part of the formula (the response) must be a n x p matrix with at least two columns.
scores	score to be used. Can be either "identity", "sign" or "rank".
stand	can be "outer" or "inner".
maxiter	maximum number of iterations. Used only for score = "sign" and score = "rank".
eps	convergence tolerance. Used only for score = "sign" or score = "rank".
eps.S	lower limit for the residual norms. Used only for score = "sign" or score = "rank" in the iteration procedure to avoid to divide by a zero norm.

x	logical. Indicating whether the design matrix 'x' returned from the model matrix should be stored. Default is TRUE. Might be needed for example in the anova function.
y	logical. Indicating whether the response matrix 'y' should be stored. Default is TRUE.
data	an optional data frame, list or environment (or object coercible by 'as.data.frame' to a data frame) containing the variables in the model. If not found in 'data', the variables are taken from 'environment(formula)', typically the environment from which 'mv.l1lm' is called.
subset	an optional vector specifying a subset of observations to be used in the fitting process.
na.action	a function which indicates what should happen when the data contain 'NA's.

Details

The theory behind this function is described in detail in Chapter 13 of the MNM book.

For regular multivariate L2-regression the function `lm` might be more efficient and offers more methods. Note however that the results given by `lm` and `mv.l1lm` may differ slightly due to different divisors of the covariance matrix.

The algorithms for the sign and rank scores are still in an early phase and therefore any feedback is very welcome. For example if p+1 residuals are 0, then the algorithms may not return correct values. Note also that the computations for rank scores might be slow.

Rank regression does not provide an estimate for the intercept parameter is not considered a parameter, a Hodges-Lehmann estimator of the residuals is then an estimate when an interecept term is in the formula. For the one sample case however the function cannot be used for rank scores. We recommend that the regression function should not be used for the one or two sample case. There are distinct functions designed for that purpose. Note furthermore that in the two sample case the covariance matrix returned from the regression function differs slightly from the one returned by the function `mv.2sample.est` since there matrix A is computed in a different way.

In general it is recommended to use the `data` argument and specify there the data frame that contains the variables and matrices. For having a matrix Y in a data frame for example the following methods work:

- `MyData <- data.frame(I(Y),...)`
  or
- `MyData <- data.frame(...)`
  `MyData$Y <- Y`

Otherwise also the function `cbind` can be used on the left side of the formula to combine numeric vectors on the fly.

Value

`mv.l1ml` returns an object of 'class' `mvl1lm`.

The functions `summary` is the best choice to view the results. The generic accessor functions `coefficients`, `fitted`, `residuals` and `vcov` extract various useful features of the value returned by `mv.l1ml`.

An object of class `mv.l1ml` is a list wich contains different information depending on the scores and standardization used. To see its content use the function `str`.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

`lm`, `mv.1sample.est`, `mv.1sample.test`, `mv.2sample.est`, `mv.Csample.test`

Examples

```
# creating simple data

X <- cbind(rep(1,100),rmvnorm(100,c(0,0,0)) )
B <- matrix(c(4,1,1,0.5,-3,2,2,2),ncol=4, byrow=TRUE)
Y <- X %*% t(B)+ rmvnorm(100,c(0,0), diag(0.2,2))
DAT <- data.frame(x1=X[,2],x2=X[,3], x3=X[,4], Y=I(Y))

# true B
t(B)

# example using identity scores
test1 <- mv.l1lm(Y ~ x1 + x2 + x3, data=DAT)

print(test1)
summary(test1)
coef(test1)
vcov(test1)
head(fitted(test1))
head(residuals(test1))

# example using outer sign scores
test2 <- mv.l1lm(Y ~ x1 + x2 + x3, scores= "s", data=DAT)

print(test2)
summary(test2)
coef(test2)
vcov(test2)
head(fitted(test2))
head(residuals(test2))

# example using inner sign scores
test3 <- mv.l1lm(Y ~ x1 + x2 + x3, scores= "s", stand="i",
data=DAT)
```

```
print(test3)
summary(test3)
coef(test3)
vcov(test3)
head(fitted(test3))
head(residuals(test3))

# example using outer rank scores
test4 <- mv.l1lm(Y ~ x1 + x2 + x3, scores= "r", stand="o",
data=DAT)

print(test4)
summary(test4)
coef(test4)
vcov(test4)
head(fitted(test4))
head(residuals(test4))

# example using inner rank scores
test5 <- mv.l1lm(Y ~ x1 + x2 + x3, scores= "r", stand="i",
data=DAT)

print(test5)
summary(test5)
coef(test5)
vcov(test5)
head(fitted(test5))
head(residuals(test5))

# prediction

newData <- data.frame(x1=c(1,-2),x2=c(0.5,0.7), x3=c(-1,-1))
newData
predict(test1,newData)
predict(test2,newData)
predict(test3,newData)
predict(test4,newData)
predict(test5,newData)
```

`mv.shape.est` *Shape Matrices*

Description

Shape matrix estimates using different score functions.

Usage

```
mv.shape.est(X, score = "identity", estimate = "outer",
             location = NULL, na.action = na.fail, ...)
```

Arguments

X	a numeric data frame or matrix.
score	score to be used. Can be either "identity", "sign", "symmsign" or "rank".
estimate	can be "outer" or "inner".
location	If NULL the location vector is estimated. Alternatively a numeric p vector of location.
na.action	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
...	further arguments passed to or from other methods.

Details

This functions returns different shape matrices depending on the score function chosen. For details see chapter 9 of the MNM book.

Value

a matrix

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

cov, tyler.shape, duembgen.shape, HR.Mest, spatial.shape

Examples

```
data(iris)
IRIS <- iris[,1:4]
mv.shape.est(IRIS, "sign")
mv.shape.est(IRIS, "symmsign", "o")
mv.shape.est(IRIS, "rank")
```

`mv.shape.test` *Test for Sphericity*

Description

Test for sphericity based on different score functions.

Usage

```
mv.shape.test(X, score = "identity", location = "est",
              na.action = na.fail, ...)
```

Arguments

`X`	a numeric data frame or matrix.
`score`	the score to be used. Possible are `identity`, `sign`, and `symmsign`.
`location`	specifies if the location should be estimated or taken to be the origin. Possible choices are `est` and `origin`.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
`...`	arguments passed on to other functions.

Details

Note that here inner standardization is not logical. The rank score test is not implemented. Otherwise the tests are as described in chapter 9 of the MNM book.

To test for other "shapes" than sperical, transform the data accordingly and then test for spericity.

Value

A list with class 'htest' containing the following components:

`statistic`	the value of the test statistic.
`parameter`	the degrees of freedom for the test statistic or the number of replications in the simulation.
`p.value`	the p-value for the test.
`method`	a character string indicating what type of test was performed.
`data.name`	a character string giving the name of the data used.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Nordhausen, K. and Oja, H. (2011), Multivariate L1 Methods: The Package MNM, Journal of Statistical Software, ***43***, *1-28.*

See Also

`sr.sphere.test`

Examples

```
X <- rmvt(150,diag(1,3))
mv.shape.test(X)
mv.shape.test(X,"sym")
```

`mvPCA` *Principal Component Analysis*

Description

Principal component analysis based on different score functions

Usage

```
mvPCA(X, score = "identity", estimate = "outer",
      na.action = na.fail, ...)
```

Arguments

`X`	a numeric data frame or matrix with p columns.
`score`	score to be used. Can be either `"identity"`, `"sign"`, `"symmsign"` or `"rank"`.
`estimate`	can be `"outer"` or `"inner"`.
`na.action`	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
`...`	further arguments passed to or from other methods.

Details

PCA as descriped in chapter 9 of the MNM book. Note that here ALL scatter matrices used are standardized to have trace(p). This function differs from most other PCA functions in R in that it does not center the data. The 'mvPCA' class has a print, summary, plot and predict method.

Value

A list with class 'mvloc' containing the following components:

`EigenV`	the standardized eigenvalues.
`loadings`	matrix with the corresponding loadings.
`scores`	matrix with the principal components.
`dname`	name of X.
`method`	Which shape matrix was used for the computation.
`n.obs`	number of observations used.
`p`	number of variables.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

See Also

`princomp`, `prcomp`

Examples

```
data(iris)
IRIS <- iris[,1:4]
iris.pca <- mvPCA(IRIS, "sign", "i")
iris.pca
summary(iris.pca)
pairs(iris.pca$scores, col=iris[,5])
```

`pairs2` *Plotting two numeric matrices*

Description

The function plots each variable contained in the matrix 'x' against the all variables contained in matrix 'y'. The function is not very sophisticated and only used to consider the residuals in a multivariate regression.

Usage

```
pairs2(x, y, mars = c(4, 4, 0.1, 0.1), ...)
```

Arguments

x	a numeric matrix. Same number or rows as y.
y	a numeric matrix. Same number or rows as x.
mars	A numerical vector of the form 'c(bottom, left, top, right)' which gives the number of lines of margin to be specified on the four sides of the plot. The default is c(4, 4, 0.1, 0.1).
...	Arguments to be passed to methods, such as graphical parameters (see par). Should not contain xlab and ylab.

Author(s)

Klaus Nordhausen

Examples

```
X <- rmvnorm(50, c(0,0,1))
Y <- rmvnorm(50, c(20,1), matrix(rep(0.5,4),ncol=2))
colnames(X) <- LETTERS[1:3]
colnames(Y) <- letters[1:2]
pairs2(X,Y)
```

---

plot.mvl1lm *Residual Plot for an mvl1lm Object*

Description

Scatterplots of fitted vs. residual values of the response variable for an mvl1lm object.

Usage

```
## S3 method for class 'mvl1lm'
plot(x, captation = "Residuals vs fitted", ...)
```

Arguments

x	an object of class mvl1lm.
captation	captation of the figure.
...	optional plotting arguments.

Author(s)

Klaus Nordhausen

plot.mvloc *Plotting Method for mvloc Objects*

Description

Visually presents and compares different multivariate location estimates and their confidence ellipsoids.

Usage

```
## S3 method for class 'mvloc'
plot(x, est2 = NULL, est3 = NULL, X = NULL, ...)
```

Arguments

x	an object of class mvl1lm.
est2	An optional additional location estimate. A list with the components location, vcov and est.name, for example an object of class 'mvloc'.
est3	An optional additional location estimate. A list with the components location, vcov and est.name, for example an object of class 'mvloc'.
X	a numeric data frame or matrix. Optional data points on which the estimates could have been based.
...	optional plotting arguments. For details see plotMvloc .

Details

The figure can be used to compare different multivariate location estimates. The location of the legend is currently problematic and it is recommended that the user should provide the coordinates for the legend. The function calls plotMvloc.

Author(s)

Klaus Nordhausen

See Also

plotMvloc

Examples

```
X <- rmvt(50, diag(c(1, 2)), 3)

est1 <- mv.1sample.est(X)
est2 <- mv.1sample.est(X, "sign")
est3 <- mv.1sample.est(X, "rank", "inner")

plot(est1)
```

```
plot(est1, est2, est3, X, alim="b", lty.ell=1:3, pch.ell=14:16)
plot(est1, est2, est3, X, alim="e")
```

---

plotMvloc *Function to Plot Multivariate Location Estimates and Their Confidence Ellipsoids.*

Description

Visually presents and compares different multivariate location estimates and their confidence ellipsoids.

Usage

```
plotMvloc(est1, est2 = NULL, est3 = NULL, X = NULL, alim = NULL,
          color.ell = 2:4, color.points = grey(0.5),
          lty.ell = rep(1, 3), pch.ell = rep(16, 3),
          lwd.ell = rep(1, 3), cex.ell = rep(1, 3),
          pch.points = 1, level = 0.95, npoints = 100,
          x.legend, y.legend, cex.legend = 1, pty = "s", gap = 1,
          oma.bottom, labels, cex.labels = 2, main, ...)
```

Arguments

est1	The location of interest. A list with the components location, vcov and est.name, for example an object of class 'mvloc'.
est2	An optional additional location estimate. A list with the components location, vcov and est.name, for example an object of class 'mvloc'.
est3	An optional additional location estimate. A list with the components location, vcov and est.name, for example an object of class 'mvloc'.
X	a numeric data frame or matrix. Optional data points on which the estimates could have been based.
alim	can be NULL, both or ellipses. Specifies whether the plotting regions are are based on the confidence ellipsoids only or also the range of the data points. If NULL it chooses both if X is provided and otherwise ellipses.
color.ell	vector of length 3 that gives the colors for the corresponding estimates 'est1', 'est2' and 'est3'.
color.points	the color of the data points.
lty.ell	line types of the confidence ellipsoids.
pch.ell	plotting symbols for the location estimates, the centers of the confidence ellipsoids.
lwd.ell	line width values of the confidence ellipsoids.
cex.ell	cex values for the location estimates, the centers of the confidence ellipsoids.
pch.points	plotting symbol for the data points X.

level	the level of the confidence ellipsoids.
npoints	the number of points used to approximate each ellipsoid.
x.legend	vertical position of the legend. By default tries to find for 2 to 4 dimensional data a good location. If NULL no legend is drawn.
y.legend	horicontal position of the legend. By default tries to find for 2 to 4 dimensional data a good location. If NULL no legend is drawn.
cex.legend	cex for the legend.
pty	pty value for the individual plots of the scatter matrix. Default is "s".
gap	distance between subplots, in margin lines.
oma.bottom	oma value of the bottom.
labels	optional labels for the diagonals.
cex.labels	cex for the labels. Default is 2.
main	optional title of the plot.
...	further arguments passed to or from other methods.

Details

The figure can be used to compare different multivariate location estimates. The location of the legend is currently problematic and it is recommended that the user should provide the coordinates for the legend.

Value

A scatterplot matrix.

Author(s)

Klaus Nordhausen

See Also

ellipse, plotShape

Examples

```
X <- rmvt(50, diag(c(1, 2)), 3)

est1 <- mv.1sample.est(X)
est2 <- mv.1sample.est(X, "sign")
est3 <- mv.1sample.est(X, "rank", "inner")

plotMvloc(est1)
plotMvloc(est1, est2, est3, X, alim="b", lty.ell=1:3, pch.ell=14:16)
plotMvloc(est1, est2, est3, X, alim="e")
```

plotShape *Pairwise Scatterplot Matrix of Shape Matrices*

Description

Function for visual comparisons for up to three shape matrices.

Usage

```
plotShape(est1, est2 = NULL, est3 = NULL, X = NULL, alim = NULL,
          color.ell = 2:4, color.points = grey(0.5),
          lty.ell = rep(1, 3), pch.ell = rep(16, 3),
          lwd.ell = rep(1, 3), cex.ell = rep(1, 3),
          pch.points = 1, level = 0.5, npoints = 100,
          x.legend, y.legend, cex.legend = 1, pty = "s", gap = 1,
          oma.bottom, labels, cex.labels = 2, main, ...)
```

Arguments

est1	The shape matrix of interest. A list with the components location, scatter and est.name.
est2	An optional additional shape matrix. A list with the components location, scatter and est.name.
est3	An optional additional shape matrix. A list with the components location, scatter and est.name.
X	a numeric data frame or matrix. Optional data points on which the estimates could have been based.
alim	can be NULL, both or ellipses. Specifies when the plotting regions are computed if only the size of the ellipses are to be considered or also the range of the data points. If NULL it chooses both if X is provided and otherwise ellipses.
color.ell	vector of length 3 that gives the colors for the corresponding estimates 'est1', 'est2' and 'est3'.
color.points	the color of the data points.
lty.ell	line types of the confidence ellipsoids.
pch.ell	plotting symbols for the location estimates, the centers of the confidence ellipsoids.
lwd.ell	line width values of the confidence ellipsoids.
cex.ell	cex values for the location estimates, the centers of the confidence ellipsoids.
pch.points	plotting symbol for the data points X.
level	The proportion of the data points that should be inside the ellipses. If there is no data the value for t in the function ellipse.
npoints	the number of points used to approximate each ellipsoid.

x.legend	vertical position of the legend. By default tries to find for 2 to 4 dimensional data a good location. If NULL no legend is drawn.
y.legend	horicontal position of the legend. By default tries to find for 2 to 4 dimensional data a good location. If NULL no legend is drawn.
cex.legend	cex for the legend.
pty	pty value for the individual plots of the scatter matrix. Default is "s".
gap	distance between subplots, in margin lines.
oma.bottom	oma value of the bottom.
labels	optional labels for the diagonals.
cex.labels	cex for the labels. Default is 2.
main	optional title of the plot.
...	further arguments passed to or from other methods.

Details

All scatter matrices are standardized to have determinant 1. If X is given, the Mahalanobis distances based on the location and shape estimates are computed, and t in the function ellipse is the level quantile of the Mahalanobis distances. If no X is provided t equals level.

The location of the legend is currently problematic and it is recommended that the user should provide the coordinates for the legend.

Value

A scatter plot matrix.

Author(s)

Klaus Nordhausen

See Also

ellipse, plotMvloc

Examples

```
X <- rmvt(100, diag(3), df=3)

EST1 <- list(location=colMeans(X), scatter=cov(X), est.name="COV")
HR <- HR.Mest(X)
EST2 <- list(location=HR$center, scatter=HR$scatter, est.name="Tyler")
plotShape(EST1,EST2, X=X)
```

predict.mvl1lm *Predicted Values Based on a Model Fitted by mv.l1lm*

Description

Predicted response values based on a model fitted by `mv.l1lm`.

Usage

```
## S3 method for class 'mvl1lm'
predict(object, newdata, na.action = na.pass, ...)
```

Arguments

`object`	an object of class `mvl1lm`.
`newdata`	An optional data frame with the values of the explaining variables. If omitted, the fitted values are used.
`na.action`	function determining what should be done with missing values in 'newdata'.
`...`	needed for other methods.

Author(s)

Klaus Nordhausen

predict.mvPCA *Prediction Method for a Principal Component Object of Type mvPCA*

Description

Prediction method for class mvPCA.

Usage

```
## S3 method for class 'mvPCA'
predict(object, newdata, ...)
```

Arguments

`object`	an object of class `mvloc`.
`newdata`	New data with the same variables. If missing just the scores of object are returned.
`...`	needed for other methods.

Value

a matrix with the predicted principal components.

Author(s)

Klaus Nordhausen

---

print.anovamvl1lm *Printing an Object of Class anovamvl1lm*

Description

Printing an object of class 'anovamvl1lm'.

Usage

```
## S3 method for class 'anovamvl1lm'
print(x, ...)
```

Arguments

x	an object of class anovamvl1lm.
...	needed for other methods.

Author(s)

Klaus Nordhausen

---

print.mvcloc *Printing an 'mvcloc' Object*

Description

Printing an mvcloc object.

Usage

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

Arguments

x	an object of class mvcloc.
...	arguments that can be passed further on.

Author(s)

Jyrki Möttönen <jyrki.mottonen@helsinki.fi>

print.mvl1lm *Printing an mvl1lm Object*

Description

Printing of an mvl1lm object.

Usage

```
## S3 method for class 'mvl1lm'
print(x, digits = 3, ...)
```

Arguments

x	an object of class mvl1lm.
digits	minimal number of _significant_ digits.
...	needed for other methods.

Author(s)

Klaus Nordhausen

print.mvloc *Printing an 'mvloc' Object*

Description

Printing an mvloc object.

Usage

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

Arguments

x	an object of class mvloc.
...	arguments that can be passed further on.

Author(s)

Klaus Nordhausen

print.mvPCA *Printing Method for a Principal Component Object of Type mvPCA*

Description

Prints an object of class mvPCA.

Usage

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

Arguments

x	object of type 'mvPCA'
...	needed for other printing methods.

Author(s)

Klaus Nordhausen

residuals.mvl1lm *Residuals of an mvl1lm Object*

Description

Extracts the residuals of an mvl1lm object.

Usage

```
## S3 method for class 'mvl1lm'
residuals(object, ...)
```

Arguments

object	an object of class mvl1lm.
...	needed for other methods.

Author(s)

Klaus Nordhausen

rmvpowerexp *Random Samples From a Power Exponential Distributions*

Description

Function to obtain random samples from a multivariate power exponential distribution.

Usage

```
rmvpowerexp(n, Location = rep(0, nrow(Scatter)),
            Scatter = diag(length(Location)), Beta = 1)
```

Arguments

n	number of random samples.
Location	Location vector of the distribution.
Scatter	Scatter matrix of the distribution.
Beta	shape parameter of the distribution.

Details

The power exponential distribution is an elliptical distribution which can have light or heavy tails. `Beta = 1` yields a multivariate normal distribution, `Beta = 0.5` the multivariate Laplace distribution and with increasing `Beta` converges to a multivariate uniform distribution.

Value

a matrix.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

See Also

rmvnorm, rmvt

Examples

```
X1 <- rmvpowerexp(100,c(0,0,0),Beta = 0.5)
pairs(X1)
X2 <- rmvpowerexp(100,c(0,0,0),Beta = 1)
pairs(X2)
X3 <- rmvpowerexp(100,c(0,0,0),Beta = 10)
pairs(X3)
```

runifsphere *Random Samples From the Unit Sphere*

Description

Function to sample uniformly distributed observations on the unit sphere.

Usage

```
runifsphere(n, p)
```

Arguments

n	number of random samples.
p	dimension of the unit sphere.

Value

a matrix.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Examples

```
X <- runifsphere(100,2)
plot(X, pty = "s")
```

screeplot.mvPCA *Plotting Method for a Principal Component Object of Type mvPCA*

Description

Creates a screeplot for an object of class mvPCA. Works analogously to a normal screeplot for a classical principal component analysis. Here however the y-axis gives the proportion of the variation explained by the components.

Usage

```
## S3 method for class 'mvPCA'
plot(x, main = deparse(substitute(x)), ...)
## S3 method for class 'mvPCA'
screeplot(x, npcs = min(10, length(x$EigenV)),
          type = c("barplot", "lines"),
          main = deparse(substitute(x)), ...)
```

Arguments

x	an object to type mvPCA.
npcs	the number of components to be plotted.
type	the type of plot.
main	title of the plot.
...	other graphical parameters passed to or from other methods.

Value

A screeplot.

Author(s)

Klaus Nordhausen

See Also

mvPCA

Examples

```
data(IRIS)
IRIS <- iris[,1:4]
iris.pca <- mvPCA(IRIS, "sign", "i")
plot(iris.pca, type="lines")
```

---

spatial.sign2 *Spatial Signs*

Description

The function computes the spatial signs for a data set. This function differs from the function spatial.sign in the way how observations with small norms are treated. For details see below.

Usage

```
spatial.sign2(X, center = TRUE, shape = TRUE, eps.S = 1e-05,
              na.action = na.fail, ...)
```

Arguments

X	a numeric data frame or matrix.
center	either a logical value or a numeric vector of length equal to the number of columns of 'X'. See below for more information.
shape	either a logical value or a square numeric matrix with number of columns equal to the number of columns of 'X'. See below for more information.
eps.S	treshold value which defines which obersvations are considered to have a small norm.
na.action	a function which indicates what should happen when the data contain 'NA's. Default is to fail.
...	arguments that can be passed on to functions used for the estimation of location and shape.

Details

The spatial signs U of X with location μ and shape V are given by transforming the data points $z_i = (x_i - \mu)V^{-\frac{1}{2}}$ and then computing

$$u_i = \frac{z_i}{\|z_i\|}.$$

If a numeric value is given as 'center' and/or 'shape' these are used as μ and/or V in the above formula. If 'center' and/or 'shape' are 'TRUE' the values for μ and/or V are estimated, if 'FALSE' the origin is used as the value of μ and/or the identity matrix as the value of V.

When the norm $\|z_i\|$ is 0 then the spatial sign is set usually to 0 as for example in the function spatial.sign. Here however if the spatial designs are defined as

$$u_i = \frac{z_i}{\|z_i\|} I(\|z_i\| > eps.S) + \frac{z_i}{eps.S} I(\|z_i\| \leq eps.S).$$

Value

a matrix with the spatial signs of the data as rows or the univariate signs as a px1 matrix. The centering vector and scaling matrix used are returned as attributes 'center' and 'shape'.

Author(s)

Klaus Nordhausen

See Also

spatial.sign, HR.Mest

Examples

```
# comparing spatial.sign and spatial.sign2
data(pulmonary)
head(spatial.sign2(pulmonary, c(-0.1099999,-0.12,-4.3),FALSE))
head(spatial.sign(pulmonary, c(-0.1099999,-0.12,-4.3),FALSE))
```

summary.mvcloc *Summarizing an 'mvcloc' Object*

Description

Summarizing an 'mvcloc' object.

Usage

```
## S3 method for class 'mvcloc'
summary(object,..., digits = 4)
```

Arguments

object	an object of class mvcloc.
...	needed for other summary methods.
digits	number of digits for rounding.

Author(s)

Jyrki Möttönen <jyrki.mottonen@helsinki.fi>

summary.mvl1lm *Summary for an mvl1lm Object*

Description

Gives a detailed output for an object of class mvl1lm. Note that the output will differ for different score functions used.

Usage

```
## S3 method for class 'mvl1lm'
summary(object, ..., digits = 3)
```

Arguments

object	an object of class mvl1lm.
...	needed for other methods.
digits	minimal number of _significant_ digits.

Author(s)

Klaus Nordhausen

summary.mvloc *Summarizing an 'mvloc' Object*

Description

Summarizing an 'mvloc' object.

Usage

```
## S3 method for class 'mvloc'
summary(object,..., digits = 4)
```

Arguments

object	an object of class mvloc.
...	needed for other summary methods.
digits	number of digits for rounding.

Author(s)

Klaus Nordhausen

summary.mvPCA *Summary for an object of class mvPCA.*

Description

Summary method for an object of class mvPCA.

Usage

```
## S3 method for class 'mvPCA'
summary(object, loadings = FALSE, cutoff = 0.1, ...)

## S3 method for class 'summary.mvPCA'
print(x, digits = 3, loadings = x$print.loadings,
               cutoff = x$cutoff, ...)
```

Arguments

object	an object to type mvPCA.
loadings	logical. Should the loadings be returned.
cutoff	numeric. Loadings below this cutoff in absolute value are shown as blank in the output.
x	an object of class "summary.mvPCA".
digits	the number of significant digits to be used in listing of loadings.
...	arguments to be passed to or from other methods.

Value

'object' with additional components 'cutoff' and 'print.loadings'.

Author(s)

Klaus Nordhausen

See Also

mvPCA

Examples

```
data(iris)
IRIS <- iris[,1:4]
iris.pca <- mvPCA(IRIS, "sign", "i")
summary(iris.pca, loadings = TRUE)
```

---

vcov.mvl1lm *Variance-Covariance Matrix of an mvl1lm Object*

Description

Extracts the variance-covariance matrix of an mvl1lm Object.

Usage

```
## S3 method for class 'mvl1lm'
vcov(object, ...)
```

Arguments

object	an object of class mvl1lm.
...	needed for other methods.

Details

For details see Chapter 13 of the MNM book.

Author(s)

Klaus Nordhausen

References

Oja, H. (2010), Multivariate Nonparametric Methods with R, *Springer.*

Index