Package 'mixsmsn'

December 22, 2014

Title Fitting finite mixture of scale mixture of skew-normal
distributions .

Date 2011-06-03

Version 1.0-9

Author Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos
<hlachos@ime.unicamp.br> and Celso Cabral
<celsoromulo@gmail.com>

Description Functions to fit finite mixture of scale mixture of
skew-normal (FM-SMSN) distributions.

Depends R (>= 1.9.0), mvtnorm (>= 0.9-9)

Maintainer Marcos Prates <marcosop@est.ufmg.br>

License GPL (>= 2.0)

Repository CRAN

Date/Publication 2013-08-17 07:16:50

NeedsCompilation no

R topics documented:

---

bmi   *Body Mass Index*

Description

The data set has the measure of the Body Mass Index (bmi) for 2107 people.

Usage

```
data(bmi)
```

Format

A data frame with 2107 observations of bmi

Source

Rodrigo M. Basso, Victor H. Lachos, Celso R. B. Cabral, Pulak Ghosh (2009). "Robust mixture modeling based on scale mixtures of skew-normal distributions". *Computational Statistics and Data Analysis* (in press). doi: 10.1016/j.csda.2009.09.031

References

Marcos Oliveira Prates, Celso Romulo Barbosa Cabral, Victor Hugo Lachos (2013)."mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions". Journal of Statistical Software, 54(12), 1-20., URL http://www.jstatsoft.org/v54/i12/.

Examples

```
## Not run:
data(bmi)
y <-bmi$bmi

hist(y,breaks=40)

## Maximum likelihood estimaton (MLE) with generated values
bmi.analysis <- smsn.mix(y, nu = 3, g = 2, get.init = TRUE, criteria = TRUE,
                         group = TRUE, calc.im=TRUE)
mix.hist(y,bmi.analysis)

## Passing initial values to MLE
mu1 <- 20; mu2 <- 35
sigma2.1 <- 9; sigma2.2 <- 9;
lambda1 <- 0; lambda2 <- 0;
pii<- c(0.5,0.5)

mu <- c(mu1,mu2)
sigma2 <- c(sigma2.1,sigma2.2)
```

```
shape <- c(lambda1,lambda2)

bmi.analysis <- smsn.mix(y, nu = 3, mu, sigma2 , shape, pii, get.init = FALSE,
                         criteria = TRUE, group = TRUE, calc.im=FALSE)
mix.hist(y,bmi.analysis)

## Calculate the information matrix (when the calc.im option in smsn.mix is set FALSE)
bmi.im <-  im.smsn(y, bmi.analysis)

## Search for the best number of clusters from g=1 to g=5
bmi.analysis <- smsn.search(y, nu = 3, g.min = 1, g.max=5)
mix.hist(y,bmi.analysis$best.model)

## End(Not run)
```

---

faithful   *Old Faithful Geyser Data*

Description

Waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

Usage

```
data(faithful)
```

Format

A data frame with 272 observations on 2 variables (p=2)

Source

Härdle, W. (1991) "Smoothing Techniques with Implementation in S". *New York: Springer.*

Azzalini, A. and Bowman, A. W. (1990). "A look at some data on the Old Faithful geyser". *Applied Statistics* 39, 357–365.

References

Marcos Oliveira Prates, Celso Romulo Barbosa Cabral, Victor Hugo Lachos (2013)."mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions". Journal of Statistical Software, 54(12), 1-20., URL http://www.jstatsoft.org/v54/i12/.

Examples

```
## Not run:
data(faithful)

## Maximum likelihood estimaton (MLE) for the multivariate FM-SMSN distribution
## with generated values
## Normal
Norm.analysis <- smsn.mmix(faithful, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                           group = TRUE, family = "Normal")
mix.contour(faithful,Norm.analysis,x.min=1,x.max=1,y.min=15,y.max=10,
            levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))

## Calculate the information matrix (when the calc.im option in smsn.mmix is set FALSE)
Norm.im <-  imm.smsn(faithful, Norm.analysis)

## Skew-Normal
Snorm.analysis <- smsn.mmix(faithful, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                            group = TRUE, family = "Skew.normal")
mix.contour(faithful,Snorm.analysis,x.min=1,x.max=1,y.min=15,y.max=10,
            levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))

## Calculate the information matrix (when the calc.im option in smsn.mmix is set FALSE)
Snorm.im <-  imm.smsn(faithful, Snorm.analysis)

## Skew-t
St.analysis <- smsn.mmix(faithful, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                         group = TRUE, family = "Skew.t")
mix.contour(faithful,St.analysis,x.min=1,x.max=1,y.min=15,y.max=10,
            levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))

## Calculate the information matrix (when the calc.im option in smsn.mmix is set FALSE)
St.im <-  imm.smsn(faithful, St.analysis)

## Passing initial values to MLE and automaticaly calculate the information matrix
mu1 <- c(5,77)
Sigma1 <- matrix(c(0.18,0.60,0.60,41), 2,2)
shape1 <- c(0.69,0.64)

mu2 <- c(2,52)
Sigma2 <- matrix(c(0.15,1.15,1.15,40), 2,2)
shape2 <- c(4.3,2.7)

pii<-c(0.65,0.35)

mu <- list(mu1,mu2)
Sigma <- list(Sigma1,Sigma2)
shape <- list(shape1,shape2)

Snorm.analysis <- smsn.mmix(faithful, nu=3, mu=mu, Sigma=Sigma, shape=shape, pii=pii,
                            g=2, get.init = FALSE, group = TRUE,
                            family = "Skew.normal", calc.im=TRUE)
```

```
mix.contour(faithful,Snorm.analysis,x.min=1,x.max=1,y.min=15,y.max=10,
            levels = c(0.1, 0.015, 0.005, 0.0009, 0.00015))

## Search for the best number of clusters from g=1 to g=3
faithful.analysis <- smsn.search(faithful, nu = 3, g.min = 1, g.max=3)
mix.contour(faithful,faithful.analysis$best.model,x.min=1,x.max=1,
            y.min=15,y.max=10,levels = c(0.1, 0.015, 0.005, 0.0009,
            0.00015))

## End(Not run)
```

---

im.smsn   *Information matrix*

Description

Calculate the information matrix of returned analysis based on the model `family` choice (univariate case, p=1).

Usage

```
im.smsn(y, model)
```

Arguments

y   the response vector

model   a variable returned by [smsn.mix](#)

Value

Estimate the Information Matrix of the parameters.

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

See Also

[smsn.mix](#)

Examples

```
 ## see \code{\link{bmi}}
```

---

imm.smsn   *Information matrix*

Description

Calculate the information matrix of returned analysis based on the model family choice (multivariate case, p>=2).

Usage

```
imm.smsn(y, model)
```

Arguments

y	the response vector (p>2)
model	a variable returned by [smsn.mmix](#)

Value

Estimate the Information Matrix of the parameters. Note: In the Information Matrix the scale parameters estimates are relative to the entries of square root matrix of Sigma.

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

See Also

[smsn.mmix](#)

Examples

```
  ## see \code{\link{faithful}}
```

---

mix.contour   *Print the selected groups with contours*

Description

Plot the contour of the observations with the group selection.

Usage

```
mix.contour(y, model,
            slice=100, ncontour=10,
            x.min=1, x.max=1,
            y.min=1,y.max=1,
            ...)
```

Arguments

y	the response matrix (dimension nx2)
model	a variable returned by smsn.mmix
slice	number of slices in the sequenceo the contour
ncontour	number of contours to be ploted
x.min	value to be subtracted of the smallest observation in the x-axis
x.max	value to be added of the biggest observation in the x-axis
y.min	value to be subtracted of the smallest observation in the y-axis
y.max	value to be added of the biggest observation in the y-axis
...	further arguments to contour

See Also

smsn.mmix

Examples

```
## see \code{\link{smsn.mmix}}
```

---

mix.dens	*Estimated densities*

Description

Plot the estimated density or log-density (univariate case, p=1).

Usage

```
mix.dens(y, model, log=FALSE, ylab=NULL, xlab = NULL, main = NULL, ...)
```

Arguments

y	the response vector
model	a variable returned by smsn.mix
log	Logical, plot log-density if TRUE (default = FALSE)
ylab	Title of the ylab, if NULL default is selected
xlab	Title of the xlab, if NULL default is selected
main	Main Title, if NULL default is selected
...	further arguments to plot

See Also

smsn.mix

Examples

```
## see \code{\link{bmi}} and \code{\link{smsn.mix}}
```

---

mix.hist   *Estimated densities*

Description

Plot the histogram along with the estimated density (univariate case, p=1).

Usage

```
mix.hist(y, model, breaks, ...)
```

Arguments

y	the response vector
model	a variable returned by smsn.mix
breaks	the same option in *histogram*
...	further arguments to hist

See Also

smsn.mix

Examples

```
## see \code{\link{bmi}} and \code{\link{smsn.mix}}
```

---

mix.lines   *Plot lines of smsn densities*

Description

Add lines of smsn estimated denisty or log-density in mix.dens plots (univariate case, p=1).

Usage

```
mix.lines(y, model, log=FALSE, ...)
```

Arguments

y	the response vector
model	a variable returned by smsn.mix
log	Logical, plot log-density if TRUE (default = FALSE)
...	further arguments to lines

See Also

smsn.mix

Examples

```
## see \code{\link{bmi}} and \code{\link{smsn.mix}}
```

---

mix.print   *Printing mix object*

Description

Printing a smsn.mix object (univariate case, p=1)

Usage

```
mix.print(model, digits = 3, ...)
```

Arguments

model	an object of class snsm.mix, see smsn.mix for details
digits	rounding for tabular output on the console (default is to round to 3 decimal place)
...	further arguments to print

See Also

smsn.mix

---

rmix   *Random univariate FM-SMSN generator*

Description

Random generator of univariate FM-SMSN distributions.

Usage

```
rmix(n, pii, family, arg)
```

Arguments

n	number of observations
pii	a vector of weights for the mixture (dimension of the number g of clusters). Must sum to one!
family	distribution family to be used in fitting ("t", "Skew.t", "Skew.cn", "Skew.slash", "Skew.normal", "Normal")
arg	a list with each entry containing a vector of size equal to the number of clusters of the necessary parameters from a family

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

See Also

[smsn.mix](#)

Examples

```
## see \code{\link{smsn.mix}}
```

---

rmmix   *Random multivariate FM-SMSN generator*

Description

Random generator of multivariate FM-SMSN distributions.

Usage

```
rmmix(n, pii, family, arg)
```

Arguments

n	number of observations
pii	a vector of weights for the mixture (dimension of the number g of clusters). Must sum to one!
family	distribution family to be used in fitting ("t", "Skew.t", "Skew.cn", "Skew.slash", "Skew.normal", "Normal")
arg	a list of g lists with each list containing the necessary parameters of the selected family

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

See Also

smsn.mmix

Examples

```
  ## see \code{\link{smsn.mmix}}
```

---

smsn.mix	*Fit univariate FM-SMSN distribution*

Description

Return EM algorithm output for FM-SMSN distributions (univaritate case, p=1).

Usage

```
smsn.mix(y,
         nu, mu = NULL, sigma2 = NULL, shape = NULL, pii = NULL,
         g = NULL, get.init = TRUE,
         criteria = TRUE, group = FALSE, family = "Skew.normal",
         error = 0.00001, iter.max = 100, calc.im = TRUE, obs.prob = FALSE,
         kmeans.param = NULL)
```

Arguments

y	the response vector
nu	the parameter of the scale variable (vector or scalar) of the SMSN family (kurtosis parameter). It is necessary to all distributions. For the "Skew.cn" must be a vector of length 2 and values in (0,1)
mu	the vector of initial values (dimension g) for the location parameters
sigma2	the vector of initial values (dimension g) for the scale parameters

shape	the vector of initial values (dimension g) for the skewness parameters
pii	the vector of initial values (dimension g) for the weights for each cluster. Must sum one!
g	the number of cluster to be considered in fitting
get.init	if TRUE, the initial values are generated via k-means
criteria	if TRUE, AIC, DIC, EDC and ICL will be calculated
group	if TRUE, the vector with the classification of the response is returned
family	distribution family to be used in fitting ("t", "Skew.t", "Skew.cn", "Skew.slash", "Skew.normal", "Normal")
error	the covergence maximum error
iter.max	the maximum number of iterations of the EM algorithm. Default = 100
calc.im	if TRUE, the information matrix is calculated and the standard errors are reported
obs.prob	if TRUE, the posterior probability of each observation belonging to one of the g groups is reported
kmeans.param	a list with alternative parameters for the kmeans function when generating initial values, list(iter.max = 10, n.start = 1, algorithm = "Hartigan-Wong")

Value

Estimated values of the location, scale, skewness and kurtosis parameter.

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

References

Rodrigo M. Basso, Victor H. Lachos, Celso R. B. Cabral, Pulak Ghosh (2010). "Robust mixture modeling based on scale mixtures of skew-normal distributions". Computational Statistics and Data Analysis, 54, 2926-2941. doi: 10.1016/j.csda.2009.09.031

Marcos Oliveira Prates, Celso Romulo Barbosa Cabral, Victor Hugo Lachos (2013)."mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions". Journal of Statistical Software, 54(12), 1-20., URL http://www.jstatsoft.org/v54/i12/.

See Also

[mix.hist](), [im.smsn]() and [smsn.search]()

Examples

```
mu1 <- 5; mu2 <- 20; mu3 <- 35
sigma2.1 <- 9; sigma2.2 <- 16; sigma2.3 <- 9
lambda1 <- 5; lambda2 <- -3; lambda3 <- -6
nu = 5
```

```
mu <- c(mu1,mu2,mu3)
sigma2 <- c(sigma2.1,sigma2.2,sigma2.3)
shape <- c(lambda1,lambda2,lambda3)
pii <- c(0.5,0.2,0.3)

arg1 = c(mu1, sigma2.1, lambda1, nu)
arg2 = c(mu2, sigma2.2, lambda2, nu)
arg3 = c(mu3, sigma2.3, lambda3, nu)
y <- rmix(n=1000, p=pii, family="Skew.t", arg=list(arg1,arg2,arg3))

## Not run:
par(mfrow=c(1,2))
## Normal fit
Norm.analysis <- smsn.mix(y, nu = 3, g = 3, get.init = TRUE, criteria = TRUE,
                          group = TRUE, family = "Normal", calc.im=FALSE)
mix.hist(y,Norm.analysis)
mix.print(Norm.analysis)
mix.dens(y,Norm.analysis)

## Skew Normal fit
Snorm.analysis <- smsn.mix(y, nu = 3, g = 3, get.init = TRUE, criteria = TRUE,
                           group = TRUE, family = "Skew.normal", calc.im=FALSE)
mix.hist(y,Snorm.analysis)
mix.print(Snorm.analysis)
mix.dens(y,Snorm.analysis)

## t fit
t.analysis <- smsn.mix(y, nu = 3, g = 3, get.init = TRUE, criteria = TRUE,
                       group = TRUE, family = "t", calc.im=FALSE)
mix.hist(y,t.analysis)
mix.print(t.analysis)
mix.dens(y,t.analysis)

## Skew t fit
St.analysis <- smsn.mix(y, nu = 3, g = 3, get.init = TRUE, criteria = TRUE,
                        group = TRUE, family = "Skew.t", calc.im=FALSE)
mix.hist(y,St.analysis)
mix.print(St.analysis)
mix.dens(y,St.analysis)

## Skew Contaminated Normal fit
Scn.analysis <- smsn.mix(y, nu = c(0.3,0.3), g = 3, get.init = TRUE, criteria = TRUE,
                         group = TRUE, family = "Skew.cn", calc.im=FALSE)
mix.hist(y,Scn.analysis)
mix.print(Scn.analysis)
mix.dens(y,Scn.analysis)

par(mfrow=c(1,1))
mix.dens(y,Norm.analysis)
mix.lines(y,Snorm.analysis,col="green")
mix.lines(y,t.analysis,col="red")
mix.lines(y,St.analysis,col="blue")
```

```
mix.lines(y,Scn.analysis,col="grey")

## End(Not run)
```

---

smsn.mmix   *Fit multivariate FM-SMSN distributions.*

Description

Return EM algorithm output for multivariate FM-SMSN distributions.

Usage

```
smsn.mmix(y, nu=1,
          mu = NULL, Sigma = NULL, shape = NULL, pii = NULL,
          g = NULL, get.init = TRUE, criteria = TRUE,
          group = FALSE, family = "Skew.normal",
          error = 0.0001, iter.max = 100, uni.Gama = FALSE,
          calc.im=FALSE, obs.prob = FALSE, kmeans.param = NULL)
```

Arguments

y	the response matrix (dimension nxp)
nu	the parameter of the scale variable (vector or scalar) of the SMSN family (kurtosis parameter). It is necessary to all distributions. For the "Skew.cn" must be a vector of length 2 and values in (0,1)
mu	a list of g arguments of vectors of initial values (dimension p) for the location parameters
Sigma	a list of g arguments of matrices of initial values (dimension pxp) for the scale parameters
shape	a list of g arguments of vectors of initial values (dimension p)for the skewness parameters
pii	the vector of initial values (dimension g) for the weights for each cluster. Must sum one!
g	the number of cluster to be considered in fitting
get.init	if TRUE, the initial values are generated via k-means
criteria	if TRUE, log-likelihood (logLik), AIC, DIC, EDC and ICL will be calculated
group	if TRUE, the vector with the classification of the response is returned
family	distribution family to be used in fitting ("t", "Skew.t", "Skew.cn", "Skew.slash", "Skew.normal", "Normal")
error	the covergence maximum error
iter.max	the maximum number of iterations of the EM algorithm. Default = 100
uni.Gama	if TRUE, the Gamma parameters are restricted to be the same for all clusters

calc.im	if TRUE, the information matrix is calculated and the starndard erros are reported
obs.prob	if TRUE, the posterior probability of each observation belonging to one of the g groups is reported
kmeans.param	a list with alternative parameters for the kmeans function when generating initial values, list(iter.max = 10, n.start = 1, algorithm = "Hartigan-Wong")

Value

Estimated values of the location, scale, skewness and kurtosis parameter. Note: The scale parameters estimated are relative to the entries of the squae root matrix of Sigma.

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

References

Cabral, C. R. B., Lachos, V. H. and Prates, M. O. (2012). "Multivariate Mixture Modeling Using Skew-Normal Independent Distributions". *Computational Statistics & Data Analysis*, 56, 126-142, doi:10.1016/j.csda.2011.06.026.

Marcos Oliveira Prates, Celso Romulo Barbosa Cabral, Victor Hugo Lachos (2013)."mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions". Journal of Statistical Software, 54(12), 1-20., URL http://www.jstatsoft.org/v54/i12/.

See Also

[mix.contour](), [rmmix]() and [smsn.search]()

Examples

```
mu1 <- c(0,0)
Sigma1 <- matrix(c(3,1,1,3), 2,2)
shape1 <-c(4,4)
nu1 <- 4

mu2 <- c(5,5)
Sigma2 <- matrix(c(2,1,1,2), 2,2)
shape2 <-c(2,2)
nu2 <- 4

pii<-c(0.6,0.4)

arg1 = list(mu=mu1, Sigma=Sigma1, shape=shape1, nu=nu1)
arg2 = list(mu=mu2, Sigma=Sigma2, shape=shape2, nu=nu2)
y <- rmmix(n= 500, p = pii, "Skew.t", list(arg1,arg2))

## Not run:
```

```
## Normal fit giving intial values
mu <- list(mu1,mu2)
Sigma <- list(Sigma1,Sigma2)
shape <- list(shape1,shape2)
pii <- c(0.6,0.4)

Norm.analysis <- smsn.mmix(y, nu=3, mu=mu, Sigma=Sigma, shape=shape, pii = pii,
                           criteria = TRUE, g=2, get.init = FALSE, group = TRUE,
                           family = "Normal")
mix.contour(y,Norm.analysis)

## Normal fit
Norm.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                           group = TRUE, family = "Normal")
mix.contour(y,Norm.analysis)

## Normal fit with a unique Gamma
Norm.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                           group = TRUE, family = "Normal", uni.Gama = TRUE)
mix.contour(y,Norm.analysis)


## Skew Normal fit
Snorm.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                            group = TRUE, family = "Skew.normal")
mix.contour(y,Snorm.analysis)

## t fit
t.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                        group = TRUE, family = "t")
mix.contour(y,t.analysis)

## Skew t fit
St.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                         group = TRUE, family = "Skew.t")
mix.contour(y,St.analysis)

## Skew Contaminated Normal fit
Scn.analysis <- smsn.mmix(y, nu=c(0.1,0.1), g=2, get.init = TRUE, criteria = TRUE,
                          group = TRUE, family = "Skew.cn",error=0.01)
mix.contour(y,Scn.analysis)

## Skew Contaminated Normal fit
Sslash.analysis <- smsn.mmix(y, nu=3, g=2, get.init = TRUE, criteria = TRUE,
                             group = TRUE, family = "Skew.slash", error=0.1)
mix.contour(y,Sslash.analysis)


## End(Not run)
```

---

smsn.search   *Find the best number of cluster for a determined data set.*

Description

Search for the best fitting for number of cluster from g.min to g.max for a selected `family` and `criteria` for both univariate and multivariate distributions.

Usage

```
smsn.search(y, nu,
            g.min = 1, g.max = 3,
            family = "Skew.normal", criteria = "bic",
            error = 0.0001, iter.max = 100,
            calc.im = FALSE, uni.Gama = FALSE, kmeans.param = NULL)
```

Arguments

y	the response vector(matrix)
nu	the parameter of the scale variable (vector or scalar) of the SMSN family (kurtosis parameter). It is necessary to all distributions. For the "Skew.cn" must be a vector of length 2 and values in (0,1)
g.min	the minimum number of cluster to be modeled
g.max	the maximum number of cluster to be modeled
family	distribution famility to be used in fitting ("t", "Skew.t", "Skew.nc", "Skew.slash", "Skew.normal", "Normal")
criteria	the selection criteria method to be used ("aic", "bic", "edc", "icl")
error	the covergence maximum error
iter.max	the maximum number of iterations of the EM algorithm
calc.im	if TRUE, the infomation matrix is calculated and the starndard erros are reported
uni.Gama	if TRUE, the Gamma parameters are restricted to be the same for all clusters (Only valid in the multivariate case, p>1)
kmeans.param	a list with alternative parameters for the kmeans function when generating initial values, list(iter.max = 10, n.start = 1, algorithm = "Hartigan-Wong")

Value

Estimated values of the location, scale, skewness and kurtosis parameter from the optimum number of clusters.

Author(s)

Marcos Prates <marcosop@est.ufmg.br>, Victor Lachos <hlachos@ime.unicamp.br> and Celso Cabral <celsoromulo@gmail.com>

See Also

[smsn.mix](#) and [smsn.mmix](#)

Examples

```
## see \code{\link{bmi}} and \code{\link{faithful}}
```

Index