Package 'mixpack'

December 8, 2015

Encoding UTF-8

Title Tools to Work with Mixture Components

Version 0.3.4

Date 2015-12-06

URL http://mcomas.net/software/mixpack

Description A collection of tools implemented to facilitate the analysis of the components of a finite mixture distributions. The package has some functions to generate random samples coming from a finite mixture. The package provides a C++ implementation for the construction of a hierarchy over the components of a given finite mixture.

Depends R (>= 3.0.2)

Imports Rcpp (>= 0.11.5), mvtnorm, methods, stats

LinkingTo Rcpp, RcppArmadillo

Suggests mclust, Rmixmod, knitr

License GPL

LazyData true

NeedsCompilation yes

RoxygenNote 5.0.1

Author Marc Comas-Cufí [aut, cre],
Josep Antoni Martín-Fernández [aut],
Glòria Mateu-Figueras [aut]

Maintainer Marc Comas-Cufí <mcomas@imae.udg.edu>

Repository CRAN

Date/Publication 2015-12-08 23:37:46

R topics documented:

`clr_coordinates` *clr coordinates of compositional data*

Description

clr coordinates of compositional data

Usage

```
clr_coordinates(X)
```

Arguments

`X`	compositional sample

`clr_mixnorm` *CLR evaluated on gaussian mixture model posterioris of X*

Description

CLR evaluated on gaussian mixture model posterioris of X

Usage

```
clr_mixnorm(X, Pi, Mu, Sigma)
```

Arguments

`X`	dataframe where density function is evaluated
`Pi`	a vector indicating the mixing proportions
`Mu`	a two dimensional array where second component indicates the mean of each gaussian component
`Sigma`	a three dimensional array where third component indicates the variance of each gaussian component

cluster_partition *Create a cluster from a partition*

Description

Given a matrix of tau and a partition decide in which part is classified each observation

Usage

```
cluster_partition(tau, partition)
```

Arguments

tau	matrix of posterioris
partition	list of vectors containing the partition

dmixnorm *Density function of specified gaussian mixture model.*

Description

Density function of specified gaussian mixture model.

Usage

```
dmixnorm(x, Pi, Mu, S, part = 1:length(Pi), closure = TRUE)
```

Arguments

x	vector/matrix where density function is evaluated
Pi	a vector indicating the mixing proportions
Mu	a two dimensional array where second component indicates the mean of each gaussian component
S	a three dimensional array where third component indicates the variance of each gaussian component
part	subcomposition where x shoud be evaluated. Take into an account that if x has dimensions K, K components must be selected by part
closure	are probabilities Pi summing up to 1 (default TRUE)

dmixnorm_solution *Density function of specified gaussian mixture model.*

Description

The parameters are defined from the parameters obtained using other packages (' `Mclust`, `rmixmod`)

Usage

```
dmixnorm_solution(x, solution, ...)
```

Arguments

`x`	vector/matrix where density function is evaluated
`solution`	solution comming from packages `Mclust` or `rmixmod`
`...`	arguments passed to function `dmixnorm`

Examples

```
require(mclust)
mod1 = Mclust(iris[,1:4])
rmixnorm_solution(10, mod1)
```

get_hierarchical_partition *Build a hierchical partition from posterior probabilities*

Description

This function applies the methodology described in [citar article] to build a hierarchy of classes using the weights or probabilities that an element belongs to each class

Usage

```
get_hierarchical_partition(post, omega, lambda, f_omega = NULL,
  f_lambda = NULL)
```

Arguments

`post`	dataframe of probabilities/weights (`tau` must be strictly positive)
`omega`	String giving the function name used to build the hierarchy. Available functions are: entr, prop, dich
`lambda`	String giving the function name used to build the hierarchy. Available functions are: entr, demp, demp.mod, coda, coda.norm, prop

f_omega	function with two parameters (v_tau, a). Parameter v_tau is a vector of probabilities, parameter a is the a selected class. omega(v_tau, a) gives the representativeness of element with probabities v_tau to class a
f_lambda	function with three parameters (v_tau, a, b). Parameter v_tau is a vector of probabilities, parameters a and b are classes to be combined.

---

get_random_hierarchical_partition *Build a hierchical partition randomly from given K*

Description

This function return a hierachical partition contructed randonmly.

Usage

```
get_random_hierarchical_partition(K)
```

Arguments

K	number of initial groups

---

ilr_basis *Orthonormal basis for the Simplex space*

Description

Basis from the simplex space with D components

Usage

```
ilr_basis(D)
```

Arguments

D	numbre of components

ilr_coordinates *Coordinates for an orthonormal basis*

Description

Coordinates respect basis ilr_basis

Usage

```
ilr_coordinates(X)
```

Arguments

X	compositional sample

merge_components *Build a hierchical partition randomly from given K*

Description

This function return a hierachical partition contructed randonmly.

Usage

```
merge_components(post, a, b)
```

Arguments

post	posterior probability matrix
a	first component to merge
b	second component to merge

Value

a matrix of posterior probabilities where components a and b are merged

merge_step *Merging components step*

Description

Merging components step

Usage

```
merge_step(post, omega, lambda, f_omega = NULL, f_lambda = NULL)
```

Arguments

post	Matrix with the posterior probabilities
omega	String giving the function name used to build the hierarchy. Available functions are: entr, prop, dich
lambda	String giving the function name used to build the hierarchy. Available functions are: entr, demp, demp.mod, coda, coda.norm, prop
f_omega	function with two parameters (v_tau, a). Parameter v_tau is a vector of probabilities, parameter a is the a selected class. omega(v_tau, a) gives the representativeness of element with probabities v_tau to class a
f_lambda	function with three parameters (v_tau, a, b). Parameter v_tau is a vector of probabilities, parameters a and b are classes to be combined.

Value

partition returns a matrix with all values for all possible mergings using functions ‘omega‘ and ‘lambda‘

mixpack *mixpack.*

Description

mixpack.

`rmixnorm` *Random sample generated from an specified gaussian mixture model.*

Description

Random sample generated from an specified gaussian mixture model.

Usage

```
rmixnorm(n, Pi, Mu, S, labels = F)
```

Arguments

`n`	Sample size
`Pi`	A vector indicating the mixing proportions
`Mu`	A two dimensional array where second component indicates the mean of each gaussian component
`S`	A three dimensional array where third component indicates the variance of each gaussian component
`labels`	A logical indicating whether or not a label shoud be returned indicating the component from where observation has been generated (default FALSE)

Value

A matrix with `n` row and columns given by the dimension of `Mu` and `S`. If `labels = T` another column is included indicating the component from where the observation was generated.

Examples

```
Pi = c(0.5, 0.3, 0.2)
Mu = array(c(## Mu first component
             5, 5,
             ## Mu second component
             1, 1,
             ## Mu third component
             0, 0), dim = c(2,3))
S = array(c(## Sigma first component
            1, 0,
            0, 1,
            ## Sigma second component
            0.2, 0,
            0, 0.2,
            ## Sigma third component
            0.05, 0,
            0, 0.05), dim = c(2, 2, 3))
X = rmixnorm(100, Pi = Pi, Mu = Mu, S = S, labels = TRUE)
plot(X[,1:2], col=X[,3])
```

`rmixnorm_solution` *Random sample generated from an specified gaussian mixture model.*

Description

The parameters are defined from the parameters obtained using other packages (' `Mclust`, `rmixmod`)

Usage

```
rmixnorm_solution(n, solution, ...)
```

Arguments

`n`	sample size
`solution`	solution comming from packages `Mclust` or `rmixmod`
`...`	arguments passed to function `rmixnorm`

Examples

```
require(mclust)
mod1 = Mclust(iris[,1:4])
rmixnorm_solution(10, mod1)
```

Index