Package 'oclust'

August 29, 2019

Type Package

Title Gaussian Model-Based Clustering with Outliers

Version 0.1.0

Author Katharine M. Clark, Paul D. McNicholas

Imports entropy,stats, utils, mclust

Maintainer Paul D. McNicholas <paulmc@mcmaster.ca>

Description Provides a function to detect and trim outliers in Gaussian mixture model-based clustering using methods described in Clark and McNicholas (2019) <arXiv:1907.01136>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-08-29 07:20:02 UTC

R topics documented:

`MixBetaDens` *Mixture of Beta Functions*

Description

MixBetaDens generates the pdf and cdf of a mixture of beta functions, and calculates the area under the graph between two points.

Usage

```
MixBetaDens(n, p, x = seq(0, 15, by = 0.01), a = 0, b = 1,
  n_g = n_g, var = var)
```

Arguments

`n`	The number of observations in the dataset
`p`	The dimension
`x`	A vector of x values to evaluate. Default value is seq(0, 15, by=0.01)
`a`	Lower bound for area evaluation. Default value is 0
`b`	Upper bound for area evaluation. Default value is 1
`n_g`	Vector decribing the number of observations in each cluster
`var`	An array of variances, one slice for each cluster

Details

The domain for this function is not [0,1] as is typical with a beta function. The domain encompasses the shifted log-likelihoods generated in `oclust`.

Value

MixBetaDens returns a list with

`pdf`	The probability density at each x value
`cdf`	The cumulative density at each x value
`area`	The area under the pdf graph between a and b

`oclust` *OCLUST Algorithm*

Description

oclust is a trimming method in model-based clustering. It iterates over possible values for the number of outliers and returns the model parameters for the best model as determined by the minimum KL divergence.

Usage

```
oclust(x, o, G, modelNames = NULL, prior = NULL, mc.cores = 1,
  keepAllRes = F, verb = F)
```

Arguments

`x`	A matrix or dataframe with n rows of observations and p columns
`o`	An upperbound for the number of outliers
`G`	The number of clusters
`modelNames`	The model to fit using the Mclust function. Default is NULL (all models).
`prior`	The prior parameter in the Mclust function. Default is NULL.
`mc.cores`	Number of cores to use if running in parallel. Default is 1
`keepAllRes`	A logical value indicating whether to keep the results from all iterations. Default is F.
`verb`	A logical value indicating whether verbose mode is desired, i.e., whether the value of o should be printed as the algorithm proceeds. Default is F.

Value

oclust returns a list of class oclust with

`data`	The initial data matrix
`numO`	The predicted number of outliers
`G`	The number of clusters
`outs`	The most likely outliers in order of likelihood
`class`	The classification for the optimal solution
`pi.g`	The group proportions for the optimal solution
`mu`	The cluster means for the optimal solution
`sigma`	The cluster variances for the optimal solution
`KL`	The KL divergence for each iteration, with the first value being for o=0
`BIC`	The BIC for each iteration, with the first value being for o=0
`bic=bic`	The BIC for the optimal solution
`all_results`	(Optional) The parameters for each run if keepAllRes=T. For each, index i+1 corresponds to o=i

Author(s)

Katharine M. Clark and Paul D. McNicholas

References

Katharine M. Clark and Paul D. McNicholas (2019), Using subset log-likelihoods to trim outliers in Gaussian mixture models. arXiv preprint arXiv:1907.01136.

Examples

```
data(iris)
iris.o<-oclust(x=iris[,-5],o=10,G=3,modelNames="VVV")
summary(iris.o)
plot(iris.o,what="classification")
plot(iris.o,what="KL")
```

`plot.oclust` *Plots results of the 'oclust' algorithm.*

Description

Plots results of the 'oclust' algorithm.

Usage

```
## S3 method for class 'oclust'
plot(x, what = c("BIC", "classification", "KL"),
  dimens = NULL, xlab = NULL, ylab = NULL, ylim = NULL,
  addEllipses = TRUE, ...)
```

Arguments

`x`	An 'oclust' class object obtained by using `oclust`
`what`	A string specifying the type of graph. The options are: • "BIC" a plot of BICs for each number of outliers • "classification" a plot of the classifications for the optimal solution. For data with p>2, if more than two "dimens" are specified, a pairs plot is produced. If two "dimens" are specified, a coordinate projection plot is produced with the specified "dimens". Ellipses corresponding to covariances of mixture components are also drawn if "addEllipses = TRUE". • "KL" a plot of Kullback-Liebler divergence for each number of outliers
`dimens`	a vector specifying the dimensions of the coordinate projections
`xlab, ylab`	optional argument specifying axis labels for the classsification plot

ylim	optional limits of the y axis of the BIC and KL plots
addEllipses	logical indicating whether to include ellipses corresponding to the covariances of the mixture components
...	other graphical parameters

`print.oclust` *Print oclust*

Description

Prints list of available components for 'oclust' class objects.

Usage

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

Arguments

x	An 'oclust' class object obtained by using `oclust`
...	additional print parameters

`print.summary.oclust` *Prints the summary of key results for 'oclust' class objects.*

Description

Prints the summary of key results for 'oclust' class objects.

Usage

```
## S3 method for class 'summary.oclust'
print(x, digits = getOption("digits"), ...)
```

Arguments

x	An 'oclust' class object obtained by using `oclust`
digits	number of digits to print
...	additional print arguments

`summary.oclust` *Summarizes key results for 'oclust' class objects.*

Description

Summarizes key results for 'oclust' class objects.

Usage

```
## S3 method for class 'oclust'
summary(object, ...)
```

Arguments

`object`	An 'oclust' class object obtained by using `oclust`
`...`	additional summary arguments

Index