

Package ‘mclust’

February 20, 2010

Version 3.4.3

Date 2010-02-19

Author Chris Fraley and Adrian Raftery

Title Model-Based Clustering / Normal Mixture Modeling

Description Model-based clustering and normal mixture modeling including Bayesian regularization

Depends R (>= 2.2.0), stats, utils

Suggests mix

License file LICENSE

Maintainer Chris Fraley <fraley@stat.washington.edu>

URL <http://www.stat.washington.edu/research/reports/2006/tr504.pdf>

Repository CRAN

Date/Publication 2010-02-20 07:49:59

R topics documented:

adjustedRandIndex	3
bic	4
bicEMtrain	5
cdens	6
cdense	8
chevron	9
classError	10
clPairs	11
coordProj	12
cross	15
cv1EMtrain	15
decomp2sigma	16
defaultPrior	17

Defaults.Mclust	19
dens	21
diabetes	22
em	23
emControl	25
emE	26
estep	28
estepE	30
hc	31
hcE	33
hclass	35
hypvol	36
imputeData	37
imputePairs	38
map	39
mapClass	40
Mclust	41
mclust1Dplot	43
mclust2Dplot	45
mclustBIC	47
mclustDA	50
mclustDAtest	52
mclustDAtrain	53
mclustModel	55
mclustModelNames	57
mclustOptions	58
mclustVariance	60
me	61
meE	63
mstep	65
mstepE	66
mvn	68
mvnX	70
nVarParams	72
partconv	73
partuniq	74
plot.Mclust	75
plot.mclustBIC	76
plot.mclustDA	78
plot.mclustDAtrain	79
priorControl	81
randProj	82
sigma2decomp	84
sim	86
simE	88
summary.mclustBIC	90
summary.mclustDAtest	91
summary.mclustDAtrain	92

<i>adjustedRandIndex</i>	3
summary.mclustModel	93
surfacePlot	94
uncerPlot	96
unmap	97
wreath	98
Index	99

`adjustedRandIndex` *Adjusted Rand Index*

Description

Computes the adjusted Rand index comparing two classifications.

Usage

```
adjustedRandIndex(x, y)
```

Arguments

`x` A numeric or character vector of class labels.
`y` A numeric or character vector of class labels. The length of `y` should be the same as that of `x`.

Value

The adjusted Rand index comparing the two partitions (a scalar). It has the value

References

L. Hubert and P. Arabie (1985) Comparing Partitions, *Journal of the Classification* 2:193-218.

See Also

[classError](#), [mapClass](#), [table](#)

Examples

```
a <- rep(1:3, 3)
a
b <- rep(c("A", "B", "C"), 3)
b
adjustedRandIndex(a, b)

a <- sample(1:3, 9, replace = TRUE)
a
b <- sample(c("A", "B", "C"), 9, replace = TRUE)
b
adjustedRandIndex(a, b)
```

```

a <- rep(1:3, 4)
a
b <- rep(c("A", "B", "C", "D"), 3)
b
adjustedRandIndex(a, b)

irisHCvzv <- hc(modelName = "VZV", data = iris[,-5])
cl3 <- hclass(irisHCvzv, 3)
adjustedRandIndex(cl3, iris[,5])

irisBIC <- mclustBIC(iris[,-5])
adjustedRandIndex(summary(irisBIC, iris[,-5])$classification, iris[,5])
adjustedRandIndex(summary(irisBIC, iris[,-5], G=3)$classification, iris[,5])

```

bic
BIC for Parameterized Gaussian Mixture Models

Description

Computes the BIC (Bayesian Information Criterion) for parameterized mixture models given the loglikelihood, the dimension of the data, and number of mixture components in the model.

Usage

```
bic(modelName, loglik, n, d, G, noise=FALSE, equalPro=FALSE, ...)
```

Arguments

<code>modelName</code>	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
<code>loglik</code>	The loglikelihood for a data set with respect to the Gaussian mixture model specified in the <code>modelName</code> argument.
<code>n</code>	The number of observations in the data used to compute <code>loglik</code> .
<code>d</code>	The dimension of the data used to compute <code>loglik</code> .
<code>G</code>	The number of components in the Gaussian mixture model used to compute <code>loglik</code> .
<code>noise</code>	A logical variable indicating whether or not the model includes an optional Poisson noise component. The default is to assume no noise component.
<code>equalPro</code>	A logical variable indicating whether or not the components in the model are assumed to be present in equal proportion. The default is to assume unequal mixing proportions.
<code>...</code>	Catches unused arguments in an indirect or list call via <code>do.call</code> .

Value

The BIC or Bayesian Information Criterion for the given input arguments.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611:631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`nVarParams`, `mclustBIC`, `do.call`.

Examples

```
n <- nrow(iris)
d <- ncol(iris)-1
G <- 3

emEst <- me(modelName="VVI", data=iris[,-5], unmap(iris[,5]))
names(emEst)

args(bic)
bic(modelName="VVI", loglik=emEst$loglik, n=n, d=d, G=G)
## Not run: do.call("bic", emEst)    ## alternative call
```

bicEMtrain

Select models in discriminant analysis using BIC

Description

Computes the BIC given a dataset and labels for selected models.

Usage

```
bicEMtrain(data, labels, modelNames=NULL)
```

Arguments

<code>data</code>	A numeric vector or matrix of observations.
<code>labels</code>	Labels for each element or row in the data.
<code>modelNames</code>	Vector of model names that should be tested. The default is to select all available model names.

Value

Returns a vector where each element is the BIC for the dataset and labels corresponding to each model.

References

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Author(s)

C. Fraley

See Also

[cvLEMtrain](#)

Examples

```
even <- seq(from=2, to=nrow(chickwts), by=2)
round(bicEMtrain(chickwts[even,1], labels=chickwts[even,2]), 1)
```

cdens

Component Density for Parameterized MVN Mixture Models

Description

Computes component densities for observations in MVN mixture models parameterized by eigenvalue decomposition.

Usage

```
cdens(modelName, data, logarithm = FALSE, parameters, warn = NULL, ...)
```

Arguments

<code>modelName</code>	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
<code>data</code>	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>logarithm</code>	A logical value indicating whether or not the logarithm of the component densities should be returned. The default is to return the component densities, obtained from the log component densities by exponentiation.
<code>parameters</code>	The parameters of the model: mean The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.

`warn` A logical value indicating whether or not a warning should be issued when computations fail. The default is `warn=FALSE`.

`...` Catches unused arguments in indirect or list calls via `do.call`.

Value

A numeric matrix whose $[i, k]$ th entry is the density or log density of observation i in component k . The densities are not scaled by mixing proportions.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Note

When one or more component densities are very large in magnitude, it may be possible to compute the logarithm of the component densities but not the component densities themselves due to overflow.

See Also

`cdense`, `...`, `cdensVVV`, `dens`, `estep`, `mclustModelNames`, `mclustVariance`, `mclustOptions`, `do.call`

Examples

```
z2 <- unmap(hclass(hcVVV(faithful),2)) # initial value for 2 class case

model <- me(modelName="EEE", data=faithful, z=z2)
cdens(modelName="EEE", data=faithful, logarithm = TRUE,
      parameters = model$parameters)[1:5,]

odd <- seq(1, nrow(cross), by = 2)
oddBIC <- mclustBIC(cross[odd,-1])
oddModel <- mclustModel(cross[odd,-1], oddBIC) ## best parameter estimates
names(oddModel)

even <- odd + 1
densities <- cdens(modelName = oddModel$modelName, data = cross[even,-1],
                  parameters = oddModel$parameters)
cbind(class = cross[even,1], densities)[1:5,]
```

cdensE

*Component Density for a Parameterized MVN Mixture Model***Description**

Computes component densities for points in a parameterized MVN mixture model.

Usage

```
cdensE(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensV(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensEII(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensVII(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensEEI(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensVEI(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensEVI(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensVVI(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensEEE(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensEEV(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensVEV(data, logarithm = FALSE, parameters, warn = NULL, ...)
cdensVVV(data, logarithm = FALSE, parameters, warn = NULL, ...)
```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
logarithm	A logical value indicating whether or not the logarithm of the component densities should be returned. The default is to return the component densities, obtained from the log component densities by exponentiation.
parameters	The parameters of the model: <ul style="list-style-type: none"> mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details. pro Mixing proportions for the components of the mixture. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.
warn	A logical value indicating whether or not a warning should be issued when computations fail. The default is <code>warn=FALSE</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A numeric matrix whose $[i, j]$ th entry is the density of observation i in component j . The densities are not scaled by mixing proportions.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631. See <http://www.stat.washington.edu/mclust>.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Note

When one or more component densities are very large in magnitude, then it may be possible to compute the logarithm of the component densities but not the component densities themselves due to overflow.

See Also

[cdens](#), [dens](#), [mclustBIC](#), [mstep](#), [mclustOptions](#), [do.call](#)

Examples

```
z2 <- unmap(hclass(hcVVV(faithful),2)) # initial value for 2 class case

model <- meVVV(data=faithful, z=z2)
cdensVVV(data=faithful, logarithm = TRUE, parameters = model$parameters)

z2 <- unmap(cross[,1])

model <- meEEV(data = cross[,-1], z = z2)

EEVdensities <- cdensEEV( data = cross[,-1], parameters = model$parameters)

cbind(cross[,-1],map(EEVdensities))
```

chevron

Simulated minefield data

Description

A two-dimensional data set of simulated minefield data (1104 observations).

Usage

```
data(chevron)
```

References

- A. Dasgupta and A. E. Raftery (1998). Detecting features in spatial point processes with clutter via model-based clustering. *Journal of the American Statistical Association* 93:294-302.
- C. Fraley and A.E. Raftery (1998). *Computer Journal* 41:578-588.
- G. J. McLachlan and D. Peel (2000). *Finite Mixture Models*, Wiley, pages 110-112.

classError	<i>Classification error.</i>
------------	------------------------------

Description

Error for a given classification relative to a known truth. Location of errors in a given classification relative to a known truth.

Usage

```
classError(classification, truth)
```

Arguments

classification	A numeric or character vector of class labels.
truth	A numeric or character vector of class labels. Must have the same length as classification.

Details

If more than one mapping between classification and truth corresponds to the minimum number of classification errors, only one possible set of misclassified observations is returned.

Value

A list with the following two components:

misclassified	The indexes of the misclassified data points in a minimum error mapping between the given classification and the given truth.
errorRate	The errorRate corresponding to a minimum error mapping between the given classification and the given truth.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mapClass](#), [table](#)

Examples

```
a <- rep(1:3, 3)
a
b <- rep(c("A", "B", "C"), 3)
b
classError(a, b)

a <- sample(1:3, 9, replace = TRUE)
a
b <- sample(c("A", "B", "C"), 9, replace = TRUE)
b
classError(a, b)
```

clPairs

Pairwise Scatter Plots showing Classification

Description

Creates a scatter plot for each pair of variables in given data. Observations in different classes are represented by different symbols.

Usage

```
clPairs(data, classification, symbols, colors, labels=dimnames(data)[[2]],
        CEX=1, ...)
```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
classification	A numeric or character vector representing a classification of observations (rows) of data.
symbols	Either an integer or character vector assigning a plotting symbol to each unique class in <code>classification</code> . Elements in <code>symbols</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
colors	Either an integer or character vector assigning a color to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .

labels	A vector of character strings for labeling the variables. The default is to use the column dimension names of data.
CEX	An argument specifying the size of the plotting symbols. The default value is 1.
...	Additional arguments to be passed to the graphics device.

Side Effects

Scatter plots for each combination of variables in `data` are created on the current graphics device. Observations of different classifications are labeled with different symbols.

References

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`pairs`, `coordProj`, `mclustOptions`

Examples

```
clPairs(iris[,-5], cl=iris[,5], symbols=as.character(1:3))
```

coordProj	<i>Coordinate projections of multidimensional data modeled by an MVN mixture.</i>
-----------	---

Description

Plots coordinate projections given multidimensional data and parameters of an MVN mixture model for the data.

Usage

```
coordProj(data, dims=c(1,2), parameters=NULL, z=NULL,
          classification=NULL, truth=NULL, uncertainty=NULL,
          what = c("classification", "errors", "uncertainty"),
          quantiles = c(0.75, 0.95), symbols=NULL, colors=NULL, scale = FALSE,
          xlim=NULL, ylim=NULL, CEX = 1, PCH = ".", identify = FALSE, ...)
```

Arguments

data	A numeric matrix or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
dimens	A vector of length 2 giving the integer dimensions of the desired coordinate projections. The default is <code>c(1, 2)</code> , in which the first dimension is plotted against the second.
parameters	A named list giving the parameters of an <i>MCLUS</i> T model, used to produce superimposing ellipses on the plot. The relevant components are as follows: mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i> th column is the mean of the <i>k</i> th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
z	A matrix in which the <code>[i, k]</code> th entry gives the probability of observation <i>i</i> belonging to the <i>k</i> th class. Used to compute <code>classification</code> and <code>uncertainty</code> if those arguments aren't available.
classification	A numeric or character vector representing a classification of observations (rows) of data. If present argument <code>z</code> will be ignored.
truth	A numeric or character vector giving a known classification of each data point. If <code>classification</code> or <code>z</code> is also present, this is used for displaying classification errors.
uncertainty	A numeric vector of values in $(0, 1)$ giving the uncertainty of each data point. If present argument <code>z</code> will be ignored.
what	Choose from one of the following three options: <code>"classification"</code> (default), <code>"errors"</code> , <code>"uncertainty"</code> .
quantiles	A vector of length 2 giving quantiles used in plotting uncertainty. The smallest symbols correspond to the smallest quantile (lowest uncertainty), medium-sized (open) symbols to points falling between the given quantiles, and large (filled) symbols to those in the largest quantile (highest uncertainty). The default is $(0.75, 0.95)$.
symbols	Either an integer or character vector assigning a plotting symbol to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
colors	Either an integer or character vector assigning a color to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .
scale	A logical variable indicating whether or not the two chosen dimensions should be plotted on the same scale, and thus preserve the shape of the distribution. Default: <code>scale=FALSE</code>

xlim, ylim	Arguments specifying bounds for the ordinate, abscissa of the plot. This may be useful for when comparing plots.
CEX	An argument specifying the size of the plotting symbols. The default value is 1.
PCH	An argument specifying the symbol to be used when a classification has not been specified for the data. The default value is a small dot ".".
identify	A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
...	Other graphics parameters.

Side Effects

A plot showing a two-dimensional coordinate projection of the data, together with the location of the mixture components, classification, uncertainty, and/or classification errors.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[clPairs](#), [randProj](#), [mclust2Dplot](#), [mclustOptions](#)

Examples

```
est <- meVVV(iris[,-5], unmap(iris[,5]))

## Not run:
par(pty = "s", mfrow = c(1,1))
coordProj(iris[,-5], dimens=c(2,3), parameters = msEst$parameters, z = est$z,
          what = "classification", identify = TRUE)
coordProj(iris[,-5], dimens=c(2,3), parameters = msEst$parameters, z = est$z,
          truth = iris[,5], what = "errors", identify = TRUE)
coordProj(iris[,-5], dimens=c(2,3), parameters = msEst$parameters, z = est$z,
          what = "uncertainty", identify = TRUE)

## End(Not run)
```

 cross

Simulated Cross Data

Description

A 500 by 3 matrix in which the first column is the classification and the remaining columns are two data from a simulation of two crossed elliptical Gaussians.

Usage

```
data(cross)
```

Examples

```
# This dataset was created as follows
## Not run:
n <- 250
set.seed(0)
cross <- rbind(matrix(rnorm(n*2), n, 2) %*% diag(c(1,9)),
               matrix(rnorm(n*2), n, 2) %*% diag(c(1,9))[,2:1])
cross <- cbind(c(rep(1,n), rep(2,n)), x)

## End(Not run)
```

 cv1EMtrain

Select discriminant models using cross validation

Description

Leave-one-out cross validation given a dataset and labels for selected models.

Usage

```
cv1EMtrain(data, labels, modelNames=NULL)
```

Arguments

data	A numeric vector or matrix of observations.
labels	Labels for each element or row in the dataset.
modelNames	Vector of model names that should be tested. The default is to select all available model names.

Value

Returns a vector where each element is the the crossvalidated error rate for the dataset and labels corresponding to each model.

References

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Author(s)

C. Fraley

See Also

[bicEMtrain](#)

Examples

```
even <- seq(from=2, to=nrow(chickwts), by=2)
round(cvlEMtrain(chickwts[even,1], labels=chickwts[even,2]), 1)
```

decomp2sigma

Convert mixture component covariances to matrix form.

Description

Converts covariances from a parameterization by eigenvalue decomposition or cholesky factorization to representation as a 3-D array.

Usage

```
decomp2sigma(d, G, scale, shape, orientation, ...)
```

Arguments

<code>d</code>	The dimension of the data.
<code>G</code>	The number of components in the mixture model.
<code>scale</code>	Either a G -vector giving the scale of the covariance (the d th root of its determinant) for each component in the mixture model, or a single numeric value if the scale is the same for each component.
<code>shape</code>	Either a G by d matrix in which the k th column is the shape of the covariance matrix (normalized to have determinant 1) for the k th component, or a d -vector giving a common shape for all components.
<code>orientation</code>	Either a d by d by G array whose $[, , k]$ th entry is the orthonormal matrix whose columns are the eigenvectors of the covariance matrix of the k th component, or a d by d orthonormal matrix if the mixture components have a common orientation. The <code>orientation</code> component of <code>decomp</code> can be omitted in spherical and diagonal models, for which the principal components are parallel to the coordinate axes so that the orientation matrix is the identity.
<code>...</code>	Catches unused arguments from an indirect or list call via <code>do.call</code> .

Value

A 3-D array whose $[, , k]$ th component is the covariance matrix of the k th component in an MVN mixture model.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[sigma2decomp](#)

Examples

```
meEst <- meVEV(iris[,-5], unmap(iris[,5]))
names(meEst)
meEst$parameters$variance

dec <- meEst$parameters$variance
decomp2sigma(d=dec$d, G=dec$G, shape=dec$shape, scale=dec$scale,
             orientation = dec$orientation)
## Not run:
do.call("decomp2sigma", dec) ## alternative call

## End(Not run)
```

defaultPrior	<i>Default conjugate prior for Gaussian mixtures.</i>
--------------	---

Description

Default conjugate prior specification for Gaussian mixtures.

Usage

```
defaultPrior(data, G, modelName, ...)
```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
G	The number of mixture components.

modelName	A character string indicating the model: "E": equal variance (one-dimensional) "V": variable variance (one-dimensional) "EII": spherical, equal volume "VII": spherical, unequal volume "EEI": diagonal, equal volume and shape "VEI": diagonal, varying volume, equal shape "EVI": diagonal, equal volume, varying shape "VVI": diagonal, varying volume and shape "EEE": ellipsoidal, equal volume, shape, and orientation "EEV": ellipsoidal, equal volume and equal shape "VEV": ellipsoidal, equal shape "VVV": ellipsoidal, varying volume, shape, and orientation
...	One or more of the following: <ul style="list-style-type: none"> • dof The degrees of freedom for the prior on the variance. The default is $d + 2$, where d is the dimension of the data. • scale The scale parameter for the prior on the variance. The default is $\text{var}(\text{data}) / G^{(2/d)}$, where d is the dimension of the data. • shrinkage The shrinkage parameter for the prior on the mean. The default value is 0.01. If 0 or NA, no prior is assumed for the mean. • mean The mean parameter for the prior. The default value is <code>colMeans(data)</code>.

Details

`defaultPrior` is a function whose default is to output the default prior specification for EM within *MCLUS*T. `defaultPrior` can be used to specify alternative prior parameters for a conjugate prior.

Value

A list giving the prior degrees of freedom, scale, shrinkage, and mean.

References

- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.
- C. Fraley and A. E. Raftery (2006). MCLUS
- T Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

See Also

[mclustBIC](#), [me](#), [mstep](#), [priorControl](#)

Examples

```

# default prior
irisBIC <- mclustBIC(iris[,-5], prior = priorControl())
summary(irisBIC, iris[,-5])

# equivalent to previous example
irisBIC <- mclustBIC(iris[,-5],
                    prior = priorControl(functionName = "defaultPrior"))
summary(irisBIC, iris[,-5])

# no prior on the mean; default prior on variance
irisBIC <- mclustBIC(iris[,-5], prior = priorControl(shrinkage = 0))
summary(irisBIC, iris[,-5])

# equivalent to previous example
irisBIC <- mclustBIC(iris[,-5], prior =
                    priorControl(functionName="defaultPrior", shrinkage=0))
summary(irisBIC, iris[,-5])

defaultPrior( iris[-5], G = 3, modelName = "VVV")

```

Defaults.Mclust *List of values controlling defaults for some MCLUST functions.*

Description

A named list of values including an enumeration of models used as defaults in MCLUST functions.

Details

A function `mclustOptions` is supplied for assigning values to the `.Mclust` list.

Value

A list with the following components:

`emModelNames` A vector of character strings associated with multivariate models for which EM estimation is available in MCLUST.

The current default is the following list:

```

"EII": spherical, equal volume
"VII": spherical, unequal volume
"EEI": diagonal, equal volume and shape
"VEI": diagonal, varying volume, equal shape
"EVI": diagonal, equal volume, varying shape
"VVI": diagonal, varying volume and shape
"EEE": ellipsoidal, equal volume, shape, and orientation
"EEV": ellipsoidal, equal volume and equal shape
"VEV": ellipsoidal, equal shape
"VVV": ellipsoidal, varying volume, shape, and orientation

```

- `hcModelNames` A vector of character strings associated with multivariate models for which model-based hierarchical clustering is available in MCLUST. The current default is the following list:
- "EII": spherical, equal volume
 - "VII": spherical, unequal volume
 - "EEE": ellipsoidal, equal volume, shape, and orientation
 - "VVV": ellipsoidal, varying volume, shape, and orientation
- `bicPlotSymbols` A vector whose entries correspond to graphics symbols for plotting the BIC values output from `Mclust` and `mclustBIC`. These are displayed in the legend which appears at the lower right of the BIC plots.
- `bicPlotColors` A vector whose entries correspond to colors for plotting the BIC curves from output from `Mclust` and `mclustBIC`. These are displayed in the legend which appears at the lower right of the BIC plots.
- `classPlotSymbols` A vector whose entries are either integers corresponding to graphics symbols or single characters for indicating classifications when plotting data. Classes are assigned symbols in the given order.
- `classPlotColors` A vector whose entries correspond to colors for indicating classifications when plotting data. Classes are assigned colors in the given order.
- `warn` A logical value indicating whether or not to issue certain warnings (usually involving singularity). Default: `warn = TRUE`.

References

- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustOptions](#), [Mclust](#), [mclustBIC](#)

Examples

```
irisBIC <- Mclust(iris[,-5])
summary(irisBIC, iris[-5])

.Mclust
.Mclust <- mclustOptions(emModelNames = c("VII", "VVI", "VVV"))
.Mclust

irisBIC <- Mclust(iris[,-5])
summary(irisBIC, iris[-5])
```

```
.Mclust <- mclustOptions() # restore defaults
.Mclust
```

dens

Density for Parameterized MVN Mixtures

Description

Computes densities of observations in parameterized MVN mixtures.

Usage

```
dens(modelName, data, logarithm = FALSE, parameters, warn=NULL, ...)
```

Arguments

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
logarithm	A logical value indicating whether or not the logarithm of the component densities should be returned. The default is to return the component densities, obtained from the log component densities by exponentiation.
parameters	The parameters of the model: <ul style="list-style-type: none"> pro The vector of mixing proportions for the components of the mixture. mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
warn	A logical value indicating whether or not a warning should be issued when computations fail. The default is <code>warn=FALSE</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A numeric vector whose i th component is the density of the i th observation in `data` in the MVN mixture specified by `parameters`.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`cdens`, `mclustOptions`, `do.call`

Examples

```
faithfulBIC <- mclustBIC(faithful)
faithfulModel <- mclustModel(faithful, faithfulBIC) ## best parameter estimates
names(faithfulModel)

Dens <- dens(modelName = faithfulModel$modelName, data = faithful,
             parameters = faithfulModel$parameters)

Dens

## Not run:
## alternative call
oddDens <- do.call("dens", c(list(data = faithful), faithfulModel))

## End(Not run)
```

diabetes

Diabetes data

Description

Diabetes data from Reaven and Miller. Number of objects: 145; 3 variables. Three classes.

Usage

```
data(diabetes)
```

References

G.M. Reaven and R.G. Miller, *Diabetologica* 16:17-24 (1979).

em	<i>EM algorithm starting with E-step for parameterized Gaussian mixture models.</i>
----	---

Description

Implements the EM algorithm for parameterized Gaussian mixture models, starting with the expectation step.

Usage

```
em(modelName, data, parameters, prior = NULL, control = emControl(),
    warn = NULL, ...)
```

Arguments

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
parameters	A names list giving the parameters of the model. The components are as follows: <ul style="list-style-type: none"> pro Mixing proportions for the components of the mixture. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components. mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i>th column is the mean of the <i>k</i>th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details. Vinv An estimate of the reciprocal hypervolume of the data region. If set to <code>NULL</code> or a negative value, the default is determined by applying function <code>hypvol</code> to the data. Used only when <code>pro</code> includes an additional mixing proportion for a noise component.
prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
control	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .
warn	A logical value indicating whether or not a warning should be issued when computations fail. The default is <code>warn=FALSE</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

<code>modelName</code>	A character string identifying the model (same as the input argument).
<code>z</code>	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>parameters</code>	<p>pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.</p> <p>mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model.</p> <p>variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.</p> <p>Vinv The estimate of the reciprocal hypervolume of the data region used in the computation when the input indicates the addition of a noise component to the model.</p>
<code>loglik</code>	The log likelihood for the data in the mixture model.
<code>Attributes:</code>	<ul style="list-style-type: none"> • "info" Information on the iteration. • "WARNING" An appropriate warning if problems are encountered in the computations.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[emE](#), ..., [emVVV](#), [estep](#), [me](#), [mstep](#), [mclustOptions](#), [do.call](#)

Examples

```
msEst <- mstep(modelName = "EEE", data = iris[,-5],
              z = unmap(iris[,5]))
names(msEst)
```

```

em(modelName = msEst$modelName, data = iris[,-5],
   parameters = msEst$parameters)
## Not run:
do.call("em", c(list(data = iris[,-5]), msEst)) ## alternative call

## End(Not run)

```

emControl

Set control values for use with the EM algorithm.

Description

Supplies a list of values including tolerances for singularity and convergence assessment, for use functions involving EM within *MCLUST*.

Usage

```
emControl(eps, tol, itmax, equalPro)
```

Arguments

eps	A scalar tolerance associated with deciding when to terminate computations due to computational singularity in covariances. Smaller values of <code>eps</code> allow computations to proceed nearer to singularity. The default is the relative machine precision <code>.Machine\$double.eps</code> , which is approximately 2×10^{-16} on IEEE-compliant machines.
tol	A vector of length two giving relative convergence tolerances for the loglikelihood and for parameter convergence in the inner loop for models with iterative M-step ("VEI", "VEE", "VVE", "VEV"), respectively. The default is <code>c(1.e-5, sqrt(.Machine\$double.eps))</code> . If only one number is supplied, it is used as the tolerance for the outer iterations and the tolerance for the inner iterations is as in the default.
itmax	A vector of length two giving integer limits on the number of EM iterations and on the number of iterations in the inner loop for models with iterative M-step ("VEI", "VEE", "VVE", "VEV"), respectively. The default is <code>c(Inf, Inf)</code> allowing termination to be completely governed by <code>tol</code> . If only one number is supplied, it is used as the iteration limit for the outer iteration only.
equalPro	Logical variable indicating whether or not the mixing proportions are equal in the model. Default: <code>equalPro = FALSE</code> .

Details

`emControl` is provided for assigning values and defaults for EM within *MCLUST*.

Value

A named list in which the names are the names of the arguments and the values are the values supplied to the arguments.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[em](#), [estep](#), [me](#), [mstep](#), [mclustBIC](#)

Examples

```
irisBIC<- mclustBIC(iris[,-5], control = emControl(tol = 1.e-6))
summary(irisBIC, iris[,-5])
```

emE

EM algorithm starting with E-step for a parameterized Gaussian mixture model.

Description

Implements the EM algorithm for a parameterized Gaussian mixture model, starting with the expectation step.

Usage

```
emE(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emV(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emEII(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emVII(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emEEI(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emVEI(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emEVI(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emVVI(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emEEE(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emEEV(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emVEV(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
emVVV(data, parameters, prior=NULL, control=emControl(), warn=NULL, ...)
```

Arguments

`data` A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.

`parameters` The parameters of the model:

	pro Mixing proportions for the components of the mixture. There should one more mixing proportion than the number of Gaussian components if the mixture model includes a Poisson noise term.
	mean The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model.
	variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
	Vinv An estimate of the reciprocal hypervolume of the data region. The default is determined by applying function <code>hypvol</code> to the data. Used only when <code>pro</code> includes an additional mixing proportion for a noise component.
<code>prior</code>	The default assumes no prior, but this argument allows specification of a conjugate prior on the means and variances through the function <code>priorControl</code> .
<code>control</code>	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .
<code>warn</code>	A logical value indicating whether or not a warning should be issued whenever a singularity is encountered. The default is set in <code>.Mclust\$warn</code> .
<code>...</code>	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

<code>modelName</code>	A character string identifying the model (same as the input argument).
<code>z</code>	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>parameters</code>	<p>pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.</p> <p>mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model.</p> <p>variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.</p> <p>Vinv The estimate of the reciprocal hypervolume of the data region used in the computation when the input indicates the addition of a noise component to the model.</p>
<code>loglik</code>	The log likelihood for the data in the mixture model.
<code>Attributes:</code>	<ul style="list-style-type: none"> • "info" Information on the iteration. • "WARNING" An appropriate warning if problems are encountered in the computations.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3: An R Package for Normal Mixture Modeling and Model-Based Clustering, Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[me](#), [mstep](#), [mclustOptions](#)

Examples

```
msEst <- mstepEEE(data = iris[,-5], z = unmap(iris[,5]))
names(msEst)

emEEE(data = iris[,-5], parameters = msEst$parameters)
```

estep

E-step for parameterized Gaussian mixture models.

Description

Implements the expectation step of EM algorithm for parameterized Gaussian mixture models.

Usage

```
estep(modelName, data, parameters, warn = NULL, ...)
```

Arguments

- | | |
|-------------------------|--|
| <code>modelName</code> | A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models. |
| <code>data</code> | A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables. |
| <code>parameters</code> | A names list giving the parameters of the model. The components are as follows:
pro Mixing proportions for the components of the mixture. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components. |

	mean	The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model.
	variance	A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
	Vinv	An estimate of the reciprocal hypervolume of the data region. If set to <code>NULL</code> or a negative value, the default is determined by applying function <code>hypvol</code> to the data. Used only when <code>pro</code> includes an additional mixing proportion for a noise component.
<code>warn</code>		A logical value indicating whether or not a warning should be issued when computations fail. The default is <code>warn=FALSE</code> .
<code>...</code>		Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

<code>modelName</code>	A character string identifying the model (same as the input argument).
<code>z</code>	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>parameters</code>	The input parameters.
<code>loglik</code>	The loglikelihood for the data in the mixture model.
<code>Attribute</code>	<ul style="list-style-type: none"> "WARNING": An appropriate warning if problems are encountered in the computations.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[estepE](#), [...](#), [estepVVV](#), [em](#), [mstep](#), [mclustOptions](#) [mclustVariance](#)

Examples

```
msEst <- mstep(modelName = "VVV", data = iris[,-5], z = unmap(iris[,5]))
names(msEst)

estep(modelName = msEst$modelName, data = iris[,-5],
      parameters = msEst$parameters)
```

estepE	<i>E-step in the EM algorithm for a parameterized Gaussian mixture model.</i>
--------	---

Description

Implements the expectation step in the EM algorithm for a parameterized Gaussian mixture model.

Usage

```
estepE(data, parameters, warn = NULL, ...)
estepV(data, parameters, warn = NULL, ...)
estepEII(data, parameters, warn = NULL, ...)
estepVII(data, parameters, warn = NULL, ...)
estepEEI(data, parameters, warn = NULL, ...)
estepVEI(data, parameters, warn = NULL, ...)
estepEVI(data, parameters, warn = NULL, ...)
estepVVI(data, parameters, warn = NULL, ...)
estepEEE(data, parameters, warn = NULL, ...)
estepEEV(data, parameters, warn = NULL, ...)
estepVEV(data, parameters, warn = NULL, ...)
estepVVV(data, parameters, warn = NULL, ...)
```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
parameters	<p>The parameters of the model:</p> <ul style="list-style-type: none"> • An argument describing the variance (depends on the model): <ul style="list-style-type: none"> pro Mixing proportions for the components of the mixture. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components. mu The mean for each component. If there is more than one component, this is a matrix whose columns are the means of the components. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details. Vinv An estimate of the reciprocal hypervolume of the data region. If not supplied or set to a negative value, the default is determined by applying function <code>hypvol</code> to the data. Used only when <code>pro</code> includes an additional mixing proportion for a noise component.
warn	A logical value indicating whether or certain warnings should be issued. The default is set in <code>.Mclust\$warn</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

modelName	Character string identifying the model.
z	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
parameters	The input parameters.
loglik	The loglikelihood for the data in the mixture model.
Attribute	<ul style="list-style-type: none"> "WARNING": An appropriate warning if problems are encountered in the computations.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[estep](#), [em](#), [mstep](#), [do.call](#), [mclustOptions](#), [mclustVariance](#)

Examples

```
msEst <- mstepEII(data = iris[,-5], z = unmap(iris[,5]))
names(msEst)

estepEII(data = iris[,-5], parameters = msEst$parameters)
```

 hc

Model-based Hierarchical Clustering

Description

Agglomerative hierarchical clustering based on maximum likelihood criteria for Gaussian mixture models parameterized by eigenvalue decomposition.

Usage

```
hc(modelName, data, ...)
```

Arguments

<code>modelName</code>	A character string indicating the model. Possible models: "E" : equal variance (one-dimensional) "V" : spherical, variable variance (one-dimensional) "EII": spherical, equal volume "VII": spherical, unequal volume "EEE": ellipsoidal, equal volume, shape, and orientation "VVV": ellipsoidal, varying volume, shape, and orientation
<code>data</code>	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>...</code>	Arguments for the method-specific hc functions. See hcE .

Details

Most models have memory usage of the order of the square of the number groups in the initial partition for fast execution. Some models, such as equal variance or "EEE", do not admit a fast algorithm under the usual agglomerative hierarchical clustering paradigm. These use less memory but are much slower to execute.

Value

A numeric two-column matrix in which the i th row gives the minimum index for observations in each of the two clusters merged at the i th stage of agglomerative hierarchical clustering.

References

- J. D. Banfield and A. E. Raftery (1993). Model-based Gaussian and non-Gaussian Clustering. *Biometrics* 49:803-821.
- C. Fraley (1998). Algorithms for model-based Gaussian hierarchical clustering. *SIAM Journal on Scientific Computing* 20:270-281.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Note

If `modelName = "E"` (univariate with equal variances) or `modelName = "EII"` (multivariate with equal spherical covariances), then the method is equivalent to Ward's method for hierarchical clustering.

See Also

[hcE](#), ..., [hcVVV](#), [hclass](#)

Examples

```

hcTree <- hc(modelName = "VVV", data = iris[,-5])
cl <- hclass(hcTree,c(2,3))

## Not run:
par(pty = "s", mfrow = c(1,1))
clPairs(iris[,-5],cl=cl[, "2"])
clPairs(iris[,-5],cl=cl[, "3"])

par(mfrow = c(1,2))
dimens <- c(1,2)
coordProj(iris[,-5], dimens = dimens, classification=cl[, "2"])
coordProj(iris[,-5], dimens = dimens, classification=cl[, "3"])

## End(Not run)

```

hcE

Model-based Hierarchical Clustering

Description

Agglomerative hierarchical clustering based on maximum likelihood for a Gaussian mixture model parameterized by eigenvalue decomposition.

Usage

```

hcE(data, partition, minclus=1, ...)
hcV(data, partition, minclus = 1, alpha = 1, ...)
hcEII(data, partition, minclus = 1, ...)
hcVII(data, partition, minclus = 1, alpha = 1, ...)
hcEEE(data, partition, minclus = 1, ...)
hcVVV(data, partition, minclus = 1, alpha = 1, beta = 1, ...)

```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
partition	A numeric or character vector representing a partition of observations (rows) of data. If provided, group merges will start with this partition. Otherwise, each observation is assumed to be in a cluster by itself at the start of agglomeration.
minclus	A number indicating the number of clusters at which to stop the agglomeration. The default is to stop when all observations have been merged into a single cluster.
alpha, beta	Additional tuning parameters needed for initialization in some models. For details, see Fraley 1998. The defaults provided are usually adequate.
...	Catch unused arguments from a <code>do.call</code> call.

Details

Most models have memory usage of the order of the square of the number groups in the initial partition for fast execution. Some models, such as equal variance or "EEE", do not admit a fast algorithm under the usual agglomerative hierarchical clustering paradigm. These use less memory but are much slower to execute.

Value

A numeric two-column matrix in which the i th row gives the minimum index for observations in each of the two clusters merged at the i th stage of agglomerative hierarchical clustering.

References

- J. D. Banfield and A. E. Raftery (1993). Model-based Gaussian and non-Gaussian Clustering. *Biometrics* 49:803-821.
- C. Fraley (1998). Algorithms for model-based Gaussian hierarchical clustering. *SIAM Journal on Scientific Computing* 20:270-281.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[hc](#), [hclass](#)

Examples

```
hcTree <- hcEII(data = iris[,-5])
cl <- hclass(hcTree,c(2,3))

## Not run:
par(pty = "s", mfrow = c(1,1))
clPairs(iris[,-5],cl=cl[, "2"])
clPairs(iris[,-5],cl=cl[, "3"])

par(mfrow = c(1,2))
dimens <- c(1,2)
coordProj(iris[,-5], classification=cl[, "2"], dimens=dimens)
coordProj(iris[,-5], classification=cl[, "3"], dimens=dimens)

## End(Not run)
```

`hclass`*Classifications from Hierarchical Agglomeration*

Description

Determines the classifications corresponding to different numbers of groups given merge pairs from hierarchical agglomeration.

Usage

```
hclass(hcPairs, G)
```

Arguments

<code>hcPairs</code>	A numeric two-column matrix in which the <i>i</i> th row gives the minimum index for observations in each of the two clusters merged at the <i>i</i> th stage of agglomerative hierarchical clustering.
<code>G</code>	An integer or vector of integers giving the number of clusters for which the corresponding classifications are wanted.

Value

A matrix with `length(G)` columns, each column corresponding to a classification. Columns are indexed by the character representation of the integers in `G`.

References

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[hc](#), [hcE](#)

Examples

```
hcTree <- hc(modelName="VVV", data = iris[,-5])
cl <- hclass(hcTree, c(2, 3))

## Not run:
par(pty = "s", mfrow = c(1, 1))
clPairs(iris[,-5], cl=cl[, "2"])
clPairs(iris[,-5], cl=cl[, "3"])

## End(Not run)
```

`hypvol`*Approximate Hypervolume for Multivariate Data*

Description

Computes a simple approximation to the hypervolume of a multivariate data set.

Usage

```
hypvol(data, reciprocal=FALSE)
```

Arguments

<code>data</code>	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>reciprocal</code>	A logical variable indicating whether or not the reciprocal hypervolume is desired rather than the hypervolume itself. The default is to return the hypervolume.

Value

Computes the hypervolume by two methods: simple variable bounds and principal components, and returns the minimum value. Used to compute the default hypervolume parameter for the noise component in

References

- A. Dasgupta and A. E. Raftery (1998). Detecting features in spatial point processes with clutter via model-based clustering. *Journal of the American Statistical Association* 93:294-302.
- C. Fraley and A.E. Raftery (1998). *Computer Journal* 41:578-588.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

See Also

[mclustBIC](#)

Examples

```
hypvol(iris[,-5])
```

`imputeData`*Missing Data Imputation via the mix package*

Description

Imputes missing data using the mix package.

Usage

```
imputeData(x, categorical = NULL, seed = NULL)
```

Arguments

<code>x</code>	A numeric vector, matrix, or data frame of observations containing missing values. Categorical variables are allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>categorical</code>	A logical vectors whose <i>i</i> th entry is TRUE if the <i>i</i> th variable or column of <code>x</code> is to be interpreted as categorical and FALSE otherwise. The default is to assume that a variable is to be interpreted as categorical only if it is a factor.
<code>seed</code>	A seed for the function <code>rngseed</code> that is used to initialize the random number generator in <code>mix</code> . By default, a seed is chosen uniformly in the interval $(.Machine\$integer.max/1024, .Machine\$integer.max)$.

Value

A dataset of the same dimensions as `x` with missing values filled in.

References

J. L. Schafer, Analysis of Imcomplete Multivariate Data, Chapman and Hall, 1997.

See Also

[imputePairs](#)

Examples

```
library(mix)

# impute the continuous variables in the stlouis data
stlimp <- imputeData(stlouis[,-(1:3)])

# plot imputed values
imputePairs(stlouis[,-(1:3)], stlimp)
```

 imputePairs

Pairwise Scatter Plots showing Missing Data Imputations

Description

Creates a scatter plot for each pair of variables in given data, allowing display of imputations for missing values in different colors and symbols than nonmissing values.

Usage

```
imputePairs(x, impx, symbols = c(16,1), colors = c("black", "red"), labels,
            panel = points, ..., lower.panel = panel, upper.panel = panel,
            diag.panel = NULL, text.panel = textPanel, label.pos = 0.5 +
            has.diag/3, cex.labels = NULL, font.labels = 1, rowlaptop = TRUE,
            gap = 1)
```

Arguments

<code>x</code>	A numeric vector, matrix, or data frame of observations containing missing values. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>impx</code>	The dataset <code>x</code> with missing values imputed.
<code>symbols</code>	Either an integer or character vector assigning plotting symbols to the nonmissing data and imputed values, respectively. The default is a closed circle for the nonmissing data and an open circle for the imputed values.
<code>colors</code>	Either an integer or character vector assigning colors to the nonmissing data and imputed values, respectively. The default is black for the nonmissing data and red for the imputed values.
<code>labels</code>	As in function <code>pairs</code> .
<code>panel</code>	As in function <code>pairs</code> .
<code>...</code>	As in function <code>pairs</code> .
<code>lower.panel</code>	As in function <code>pairs</code> .
<code>upper.panel</code>	As in function <code>pairs</code> .
<code>diag.panel</code>	As in function <code>pairs</code> .
<code>text.panel</code>	As in function <code>pairs</code> .
<code>label.pos</code>	As in function <code>pairs</code> .
<code>cex.labels</code>	As in function <code>pairs</code> .
<code>font.labels</code>	As in function <code>pairs</code> .
<code>rowlaptop</code>	As in function <code>pairs</code> .
<code>gap</code>	As in function <code>pairs</code> .

Side Effects

A pairs plot displaying the location of missing and nonmissing values.

References

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`pairs`, `imputeData`

Examples

```
library(mix)

# impute the continuous variables in the stlouis data
stlimp <- imputeData( stlouis[,-(1:3)])

# plot imputed values
imputePairs( stlouis[,-(1:3)], stlimp)
```

 map

Classification given Probabilities

Description

Converts a matrix in which each row sums to 1 into the nearest matrix of $(0,1)$ indicator variables.

Usage

```
map(z, warn=TRUE, ...)
```

Arguments

<code>z</code>	A matrix (for example a matrix of conditional probabilities in which each row sums to 1 as produced by the E-step of the EM algorithm).
<code>warn</code>	A logical variable indicating whether or not a warning should be issued when there are some columns of <code>z</code> for which no row attains a maximum.
<code>...</code>	Provided to allow lists with elements other than the arguments can be passed in indirect or list calls with <code>do.call</code> .

Value

A integer vector with one entry for each row of `z`, in which the i -th value is the column index at which the i -th row of `z` attains a maximum.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[unmap](#), [estep](#), [em](#), [me](#)

Examples

```
emEst <- me(modelName = "VVV", data = iris[,-5], z = unmap(iris[,5]))
map(emEst$z)
```

mapClass

Correspondence between classifications.

Description

Best correspondence between classes given two vectors viewed as alternative classifications of the same object.

Usage

```
mapClass(a, b)
```

Arguments

a A numeric or character vector of class labels.
b A numeric or character vector of class labels. Must have the same length as **a**.

Value

A list with two named elements, **aTOb** and **bTOa** which are themselves lists. The **aTOb** list has a component corresponding to each unique element of **a**, which gives the element or elements of **b** that result in the closest class correspondence.

The **bTOa** list has a component corresponding to each unique element of **b**, which gives the element or elements of **a** that result in the closest class correspondence.

See Also

[mapClass](#), [classError](#), [table](#)

Examples

```

a <- rep(1:3, 3)
a
b <- rep(c("A", "B", "C"), 3)
b
mapClass(a, b)
a <- sample(1:3, 9, replace = TRUE)
a
b <- sample(c("A", "B", "C"), 9, replace = TRUE)
b
mapClass(a, b)

```

Mclust

*Model-Based Clustering***Description**

The optimal model according to BIC for EM initialized by hierarchical clustering for parameterized Gaussian mixture models.

Usage

```

Mclust(data, G=NULL, modelNames=NULL, prior=NULL, control=emControl(),
        initialization=NULL, warn=FALSE, ...)

```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
G	An integer vector specifying the numbers of mixture components (clusters) for which the BIC is to be calculated. The default is $G=1:9$.
modelNames	A vector of character strings indicating the models to be fitted in the EM phase of clustering. The help file for <code>mclustModelNames</code> describes the available models. The default is <code>c("E", "V")</code> for univariate data and <code>mclustOptions()\$emModelNames</code> for multivariate data ($n > d$), the spherical and diagonal models <code>c("EII", "VII", "EEI", "EVI", "VEI", "VVI")</code> for multivariate data ($n \leq d$).
prior	The default assumes no prior, but this argument allows specification of a conjugate prior on the means and variances through the function <code>priorControl</code> .
control	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .
initialization	A list containing zero or more of the following components:

- `hcPairs` A matrix of merge pairs for hierarchical clustering such as produced by function `hc`. For multivariate data, the default is to compute a hierarchical clustering tree by applying function `hc` with `modelName = "VVV"` to the data or a subset as indicated by the `subset` argument. The hierarchical clustering results are to start EM. For univariate data, the default is to use quantiles to start EM.
 - `subset` A logical or numeric vector specifying a subset of the data to be used in the initial hierarchical clustering phase.
- `warn` A logical value indicating whether or not certain warnings (usually related to singularity) should be issued. The default is to suppress these warnings.
- `...` Catches unused arguments in indirect or list calls via `do.call`.

Value

A list giving the optimal (according to BIC) parameters, conditional probabilities z , and loglikelihood, together with the associated classification and its uncertainty. The details of the output components are as follows:

- `modelName` A character string denoting the model at which the optimal BIC occurs.
- `n` The number of observations in the data.
- `d` The dimension of the data.
- `G` The optimal number of mixture components.
- `BIC` All BIC values.
- `bic` Optimal BIC value.
- `loglik` The loglikelihood corresponding to the optimal BIC.
- `z` A matrix whose $[i,k]$ th entry is the probability that observation i in the test data belongs to the k th class.
- `classification`
`map(z)`: The classification corresponding to z .
- `uncertainty` The uncertainty associated with the classification.
- `Attributes:` The input parameters other than the data.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611:631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[priorControl](#), [emControl](#), [mclustBIC](#), [mclustModelNames](#), [mclustOptions](#)

Examples

```
irisMclust <- Mclust(iris[, -5])
## Not run:
plot(irisMclust)

## End(Not run)
```

mclust1Dplot

Plot one-dimensional data modeled by an MVN mixture.

Description

Plot one-dimensional data given parameters of an MVN mixture model for the data.

Usage

```
mclust1Dplot(data, parameters=NULL, z=NULL,
             classification=NULL, truth=NULL, uncertainty=NULL,
             what = c("classification", "density", "errors", "uncertainty"),
             symbols=NULL, ngrid=length(data), xlab = NULL, xlim=NULL, CEX=1,
             identify=FALSE, ...)
```

Arguments

<code>data</code>	A numeric vector of observations. Categorical variables are not allowed.
<code>parameters</code>	A named list giving the parameters of an <i>MCLUST</i> model, used to produce superimposing ellipses on the plot. The relevant components are as follows: <ul style="list-style-type: none"> pro Mixing proportions for the components of the mixture. There should one more mixing proportion than the number of Gaussian components if the mixture model includes a Poisson noise term. mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
<code>z</code>	A matrix in which the $[i, k]$ th entry gives the probability of observation i belonging to the k th class. Used to compute <code>classification</code> and <code>uncertainty</code> if those arguments aren't available.
<code>classification</code>	A numeric or character vector representing a classification of observations (rows) of <code>data</code> . If present argument <code>z</code> will be ignored.

truth	A numeric or character vector giving a known classification of each data point. If <code>classification</code> or <code>z</code> is also present, this is used for displaying classification errors.
uncertainty	A numeric vector of values in $(0,1)$ giving the uncertainty of each data point. If present argument <code>z</code> will be ignored.
what	Choose from one of the following three options: "classification" (default), "density", "errors", "uncertainty".
symbols	Either an integer or character vector assigning a plotting symbol to each unique class <code>classification</code> . Elements in <code>symbols</code> correspond to classes in <code>classification</code> in order of appearance in the observations (the order used by the function <code>unique</code>). The default is to use a single plotting symbol <code>l</code> . Classes are delineated by showing them in separate lines above the whole of the data.
ngrid	Number of grid points to use for density computation over the interval spanned by the data. The default is the length of the data set.
xlab	An argument specifying a label for the horizontal axis.
xlim	An argument specifying bounds of the plot. This may be useful for when comparing plots.
CEX	An argument specifying the size of the plotting symbols. The default value is 1.
identify	A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
...	Other graphics parameters.

Side Effects

A plot showing location of the mixture components, classification, uncertainty, density and/or classification errors. Points in the different classes are shown in separated levels above the whole of the data.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclust2Dplot](#), [clPairs](#), [coordProj](#)

Examples

```
n <- 250 ## create artificial data
set.seed(1)
y <- c(rnorm(n,-5), rnorm(n,0), rnorm(n,5))
yclass <- c(rep(1,n), rep(2,n), rep(3,n))
```

```

yModel <- mclustModel(y, mclustBIC(y))

mclust1Dplot(y, parameters = yModel$parameters, z = yModel$z,
             what = "classification", identify = TRUE)

mclust1Dplot(y, parameters = yModel$parameters, z = yModel$z,
             truth = yclass, what = "errors", identify = TRUE)

mclust1Dplot(y, parameters = yModel$parameters, z = yModel$z,
             what = "density", identify = TRUE)

mclust1Dplot(y, z = yModel$z, parameters = yModel$parameters,
             what = "uncertainty", identify = TRUE)

```

mclust2Dplot	<i>Plot two-dimensional data modelled by an MVN mixture.</i>
--------------	--

Description

Plot two-dimensional data given parameters of an MVN mixture model for the data.

Usage

```

mclust2Dplot(data, parameters=NULL, z=NULL,
             classification=NULL, truth=NULL, uncertainty=NULL,
             what = c("classification", "uncertainty", "errors"),
             quantiles = c(0.75, 0.95), symbols=NULL, colors=NULL,
             scale=FALSE, xlim=NULL, ylim=NULL, CEX = 1, PCH = ".",
             identify = FALSE, swapAxes = FALSE, ...)

```

Arguments

data	A numeric matrix or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables. In this case the data are two dimensional, so there are two columns.
parameters	<p>A named list giving the parameters of an <i>MCLUST</i> model, used to produce superimposing ellipses on the plot. The relevant components are as follows:</p> <p>pro Mixing proportions for the components of the mixture. There should one more mixing proportion than the number of Gaussian components if the mixture model includes a Poisson noise term.</p> <p>mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i>th column is the mean of the <i>k</i>th component of the mixture model.</p>

	variance	A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
<code>z</code>		A matrix in which the $[i, k]$ th entry gives the probability of observation i belonging to the k th class. Used to compute <code>classification</code> and <code>uncertainty</code> if those arguments aren't available.
<code>classification</code>		A numeric or character vector representing a classification of observations (rows) of data. If present argument <code>z</code> will be ignored.
<code>truth</code>		A numeric or character vector giving a known classification of each data point. If <code>classification</code> or <code>z</code> is also present, this is used for displaying classification errors.
<code>uncertainty</code>		A numeric vector of values in $(0,1)$ giving the uncertainty of each data point. If present argument <code>z</code> will be ignored.
<code>what</code>		Choose from one of the following three options: "classification" (default), "errors", "uncertainty".
<code>quantiles</code>		A vector of length 2 giving quantiles used in plotting uncertainty. The smallest symbols correspond to the smallest quantile (lowest uncertainty), medium-sized (open) symbols to points falling between the given quantiles, and large (filled) symbols to those in the largest quantile (highest uncertainty). The default is $(0.75, 0.95)$.
<code>symbols</code>		Either an integer or character vector assigning a plotting symbol to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
<code>colors</code>		Either an integer or character vector assigning a color to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .
<code>scale</code>		A logical variable indicating whether or not the two chosen dimensions should be plotted on the same scale, and thus preserve the shape of the distribution. Default: <code>scale=FALSE</code>
<code>xlim, ylim</code>		An argument specifying bounds for the ordinate, abscissa of the plot. This may be useful for when comparing plots.
<code>CEX</code>		An argument specifying the size of the plotting symbols. The default value is 1.
<code>PCH</code>		An argument specifying the symbol to be used when a classification has not been specified for the data. The default value is a small dot ".".
<code>identify</code>		A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
<code>swapAxes</code>		A logical variable indicating whether or not the axes should be swapped for the plot.
<code>...</code>		Other graphics parameters.

Side Effects

A plot showing the data, together with the location of the mixture components, classification, uncertainty, and/or classification errors.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[surfacePlot](#), [clPairs](#), [coordProj](#), [mclustOptions](#)

Examples

```
faithfulModel <- mclustModel(faithful,mclustBIC(faithful))

mclust2Dplot(faithful, parameters=faithfulModel$parameters,
             z=faithfulModel$z, what = "classification", identify = TRUE)

mclust2Dplot(faithful, parameters=faithfulModel$parameters,
             z=faithfulModel$z, what = "uncertainty", identify = TRUE)
```

mclustBIC

BIC for Model-Based Clustering

Description

BIC for EM initialized by model-based hierarchical clustering for parameterized Gaussian mixture models.

Usage

```
mclustBIC(data, G=NULL, modelNames=NULL, prior=NULL, control=emControl(),
          initialization=list(hcPairs=NULL, subset=NULL, noise=NULL),
          Vinv=NULL, warn=FALSE, x=NULL, ...)
```

Arguments

`data` A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.

G	An integer vector specifying the numbers of mixture components (clusters) for which the BIC is to be calculated. The default is $G=1:9$, unless the argument <code>x</code> is specified, in which case the default is taken from the values associated with <code>x</code> .
modelName	A vector of character strings indicating the models to be fitted in the EM phase of clustering. The help file for <code>mclustModelNames</code> describes the available models. The default is <code>c("E", "V")</code> for univariate data and <code>mclustOptions()\$emModelNames</code> for multivariate data ($n > d$), the spherical and diagonal models <code>c("EII", "VII", "EEI", "EVI", "VEI", "VVI")</code> for multivariate data ($n \leq d$), unless the argument <code>x</code> is specified, in which case the default is taken from the values associated with <code>x</code> .
prior	The default assumes no prior, but this argument allows specification of a conjugate prior on the means and variances through the function <code>priorControl</code> .
control initialization	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> . A list containing zero or more of the following components: <ul style="list-style-type: none"> • <code>hcPairs</code> A matrix of merge pairs for hierarchical clustering such as produced by function <code>hc</code>. For multivariate data, the default is to compute a hierarchical clustering tree by applying function <code>hc</code> with <code>modelName = "VVV"</code> to the data or a subset as indicated by the <code>subset</code> argument. The hierarchical clustering results are to start EM. For univariate data, the default is to use quantiles to start EM. • <code>subset</code> A logical or numeric vector specifying a subset of the data to be used in the initial hierarchical clustering phase. • <code>noise</code> A logical or numeric vector indicating an initial guess as to which observations are noise in the data. If numeric the entries should correspond to row indexes of the data. If supplied, a noise term will be added to the model in the estimation.
Vinv	An estimate of the reciprocal hypervolume of the data region. The default is determined by applying function <code>hypvol</code> to the data. Used only if an initial guess as to which observations are noise is supplied.
warn	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when estimation fails. The default is to suppress these warnings.
x	An object of class <code>"mclustBIC"</code> . If supplied, <code>mclustBIC</code> will use the settings in <code>x</code> to produce another object of class <code>"mclustBIC"</code> , but with <code>G</code> and <code>modelName</code> as specified in the arguments. Models that have already been computed in <code>x</code> are not recomputed. All arguments to <code>mclustBIC</code> except <code>data</code> , <code>G</code> and <code>modelName</code> are ignored and their values are set as specified in the attributes of <code>x</code> . Defaults for <code>G</code> and <code>modelName</code> are taken from <code>x</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

Bayesian Information Criterion for the specified mixture models numbers of clusters. Auxiliary information returned as attributes.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[priorControl](#), [emControl](#), [mclustModel](#), [summary.mclustBIC](#), [hc](#), [me](#), [mclustModelNames](#), [mclustOptions](#)

Examples

```
irisBIC <- mclustBIC(iris[,-5])
irisBIC
plot(irisBIC)

subset <- sample(1:nrow(iris), 100)
irisBIC <- mclustBIC(iris[,-5], initialization=list(subset =subset))
irisBIC
plot(irisBIC)

irisBIC1 <- mclustBIC(iris[,-5], G=seq(from=1,to=9,by=2),
                    modelNames=c("EII", "EEI", "EEE"))
irisBIC1
plot(irisBIC1)
irisBIC2 <- mclustBIC(iris[,-5], G=seq(from=2,to=8,by=2),
                    modelNames=c("VII", "VVI", "VVV"), x= irisBIC1)
irisBIC2
plot(irisBIC2)

nNoise <- 450
set.seed(0)
poissonNoise <- apply(apply( iris[,-5], 2, range), 2, function(x, n)
                      runif(n, min = x[1]-.1, max = x[2]+.1), n = nNoise)
set.seed(0)
noiseInit <- sample(c(TRUE,FALSE),size=nrow(iris)+nNoise,replace=TRUE,
                  prob=c(3,1))
irisNdata <- rbind(iris[,-5], poissonNoise)
irisNbic <- mclustBIC(data = irisNdata,
                    initialization = list(noise = noiseInit))

irisNbic
plot(irisNbic)
```

mclustDA

MclustDA discriminant analysis.

Description

MclustDA training and testing.

Usage

```
mclustDA(train, test, pro=NULL, G=NULL, modelNames=NULL, prior=NULL,
         control=emControl(), initialization=NULL,
         warn=FALSE, verbose=FALSE, ...)
```

Arguments

train	A list with two named components: data giving the data and labels giving the class labels for the observations in the data.
test	A list with two named components: data giving the data and labels giving the class labels for the observations in the data. The labels are used only to compute the error rate in the <code>print</code> method and can be set to <code>NULL</code> if unknown. The default is to test the training data.
pro	Optional prior probabilities for each class in the training data.
G	An integer vector specifying the numbers of mixture components (clusters) for which the BIC is to be calculated. The default is <code>G=1:9</code> .
modelNames	A vector of character strings indicating the models to be fitted in the EM phase of clustering. The help file for <code>mclustModelNames</code> describes the available models. The default is <code>c("E", "V")</code> for univariate data and <code>mclustOptions()\$emModelNames</code> for multivariate data.
prior	The default assumes no prior, but this argument allows specification of a conjugate prior on the means and variances through the function <code>priorControl</code> .
control	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .
initialization	A list containing zero or more of the following components: <ul style="list-style-type: none"> • <code>hcPairs</code> A matrix of merge pairs for hierarchical clustering such as produced by function <code>hc</code>. The default is to compute a hierarchical clustering tree by applying function <code>hc</code> with <code>modelName = "E"</code> to univariate data and <code>modelName = "VVV"</code> to multivariate data or a subset as indicated by the <code>subset</code> argument. The hierarchical clustering results are used as starting values for EM. • <code>subset</code> A logical or numeric vector specifying a subset of the data to be used in the initial hierarchical clustering phase.
warn	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when estimation fails. The default is to suppress these warnings.

`verbose` A logical variable telling whether or not to print an indication that the function is in the training phase, which may take some time to complete.

`...` Catches unused arguments in indirect or list calls via `do.call`.

Details

`mclustDA` combines functions `mclustDAtrain` and `mclustDAtest` and their summaries. This is suitable when all test data are available in advance, so that the training model is only used once.

Value

A list with the following components:

`test` A list with the following components:

- classification** The classification of the test data for this instance of `mclustDA`.
- uncertainty** The uncertainty of the classification (0 least certain, 1 most certain).
- labels** The test labels (if any) from the input.

`training` A list with the following components:

- classification** The classification of the training data for this instance of `mclustDA`.
- z** A matrix whose $[i,k]$ th entry is the probability that observation i in the training data belongs to the k th class.
- labels** The training labels from the input.

`summary` A data frame summarizing the `mclustDA` results including the mixture models and numbers of components for the training classes.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[plot.mclustDA](#), [mclustDAtrain](#), [mclustDAtest](#), [classError](#)

Examples

```
n <- 250 ## create artificial data
set.seed(1)
triModal <- c(rnorm(n,-5), rnorm(n,0), rnorm(n,5))
triClass <- c(rep(1,n), rep(2,n), rep(3,n))

odd <- seq(from = 1, to = length(triModal), by = 2)
even <- odd + 1
```

```

triMclustDA <- mclustDA(train=list(data=triModal[odd],labels=triClass[odd]),
                      test= list(data=triModal[even],labels=triClass[even]),
                      verbose = TRUE)

names(triMclustDA)
## Not run:
  plot(triMclustDA, trainData = triModal[odd], testData = triModal[even])

## End(Not run)

odd <- seq(from = 1, to = nrow(cross), by = 2)
even <- odd + 1
crossMclustDA <- mclustDA( train=list(data=cross[odd,-1],
                                   labels=cross[odd,1]),
                        test= list(data=cross[even,-1],labels=cross[even,1]),
                        verbose = TRUE)

## Not run:
  plot(crossMclustDA, trainData = cross[odd,-1], testData = cross[even,-1])

## End(Not run)

odd <- seq(from = 1, to = nrow(iris), by = 2)
even <- odd + 1
irisMclustDA <- mclustDA(train=list(data=iris[odd,-5],labels=iris[odd,5]),
                        test= list(data=iris[even,-5],labels=iris[even,5]),
                        verbose = TRUE)

## Not run:
  plot(irisMclustDA, trainData = iris[odd,-5], testData = iris[even,-5])

## End(Not run)

```

mclustDAtest

MclustDA Testing

Description

Testing phase for MclustDA discriminant analysis.

Usage

```
mclustDAtest(data, models)
```

Arguments

data	A numeric vector, matrix, or data frame of observations to be classified.
models	A list of MCLUST-style models including parameters, usually the result of applying <code>mclustDAtrain</code> to some training data.

Details

Apply `summary` to the output to obtain the classification of the test data.

Value

A matrix in which the $[i, j]$ th entry is the density for that test observation i in the model for class j .

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`summary.mclustDAtest`, `classError`, `mclustDAtrain`

Examples

```
odd <- seq(1, nrow(cross), by = 2)
train <- mclustDAtrain(cross[odd,-1], labels = cross[odd,1]) ## training step
summary(train)

even <- odd + 1
test <- mclustDAtest(cross[even,-1], train) ## compute model densities
clEven <- summary(test)$class ## classify training set
classError(clEven, cross[even,1])
```

mclustDAtrain

MclustDA Training

Description

Training phase for MclustDA discriminant analysis.

Usage

```
mclustDAtrain(data, labels, G=NULL, modelNames=NULL, prior=NULL,
              control=emControl(), initialization=NULL, warn=FALSE,
              verbose=TRUE, ...)
```

Arguments

<code>data</code>	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>labels</code>	A numeric or character vector assigning a class label to each observation.
<code>G</code>	An integer vector specifying the numbers of mixture components (clusters) for which the BIC is to be calculated. The default is <code>G=1:9</code> .
<code>modelName</code> s	A vector of character strings indicating the models to be fitted in the EM phase of clustering. The help file for <code>mclustModelNames</code> describes the available models. The default is <code>c("E", "V")</code> for univariate data and <code>mclustOptions()\$emModelNames</code> for multivariate data.
<code>prior</code>	The default assumes no prior, but this argument allows specification of a conjugate prior on the means and variances through the function <code>priorControl</code> .
<code>control</code> <code>initialization</code>	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> . A list containing zero or more of the following components: <ul style="list-style-type: none"> • <code>hcPairs</code> A matrix of merge pairs for hierarchical clustering such as produced by function <code>hc</code>. The default is to compute a hierarchical clustering tree by applying function <code>hc</code> with <code>modelName = "E"</code> to univariate data and <code>modelName = "VVV"</code> to multivariate data or a subset as indicated by the <code>subset</code> argument. The hierarchical clustering results are used as starting values for EM. • <code>subset</code> A logical or numeric vector specifying a subset of the data to be used in the initial hierarchical clustering phase.
<code>warn</code>	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when estimation fails. The default is to suppress these warnings.
<code>verbose</code>	A logical value indicating whether or not to print the models and numbers of components for each class. Default: <code>verbose=TRUE</code> .
<code>...</code>	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Details

Except for `labels` and `verbose`, the arguments are the same as those for `mclustBIC`.

Value

A list in which each element gives the parameters and other summary information for the model best fitting each class according to BIC. Attributes are the input parameters other than `data`, `labels` and `verbose`.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[summary.mclustDAtrain](#), [mclustDAtest](#), [mclustBIC](#)

Examples

```
odd <- seq(1, nrow(cross), by = 2)
train <- mclustDAtrain(cross[odd,-1], labels = cross[odd,1]) ## training step
summary(train)

even <- odd + 1
test <- mclustDAtest(cross[even,-1], train) ## compute model densities
clEven <- summary(test)$class ## classify training set
classError(clEven, cross[even,1])
```

mclustModel

Best model based on BIC.

Description

Determines the best model from clustering via `mclustBIC` for a given set of model parameterizations and numbers of components.

Usage

```
mclustModel(data, BICvalues, G, modelNames, ...)
```

Arguments

<code>data</code>	The matrix or vector of observations used to generate ‘object’.
<code>BICvalues</code>	An "mclustBIC" object, which is the result of applying <code>mclustBIC</code> to <code>data</code> .
<code>G</code>	A vector of integers giving the numbers of mixture components (clusters) from which the best model according to BIC will be selected (<code>as.character(G)</code> must be a subset of the row names of <code>BICvalues</code>). The default is to select the best model for all numbers of mixture components used to obtain <code>BICvalues</code> .
<code>modelNames</code>	A vector of integers giving the model parameterizations from which the best model according to BIC will be selected (<code>as.character(model)</code> must be a subset of the column names of <code>BICvalues</code>). The default is to select the best model for parameterizations used to obtain <code>BICvalues</code> .
<code>...</code>	Not used. For generic/method consistency.

Value

A list giving the optimal (according to BIC) parameters, conditional probabilities z , and loglikelihood, together with the associated classification and its uncertainty.

The details of the output components are as follows:

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
n	The number of observations in the data.
d	The dimension of the data.
G	The number of components in the Gaussian mixture model corresponding to the optimal BIC.
bic	The optimal BIC value.
loglik	The loglikelihood corresponding to the optimal BIC.
parameters	A list with the following components: <ul style="list-style-type: none"> pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If missing, equal proportions are assumed. mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details. Vinv The estimate of the reciprocal hypervolume of the data region used in the computation when the input indicates the addition of a noise component to the model.
z	A matrix whose $[i,k]$ th entry is the probability that observation i in the test data belongs to the k th class.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustBIC](#)

Examples

```
irisBIC <- mclustBIC(iris[, -5])
mclustModel(iris[, -5], irisBIC)
mclustModel(iris[, -5], irisBIC, G = 1:6, modelNames = c("VII", "VVI", "VVV"))
```

mclustModelNames *MCLUST Model Names*

Description

Model names used in the *MCLUST* package.

Value

A list including the following components:

univariateMixture

A vector with the following components:

"E": equal variance (one-dimensional)

"V": variable variance (one-dimensional)

multivariateMixture

A vector with the following components:

"EII": spherical, equal volume

"VII": spherical, unequal volume

"EEI": diagonal, equal volume and shape

"VEI": diagonal, varying volume, equal shape

"EVI": diagonal, equal volume, varying shape

"VVI": diagonal, varying volume and shape

"EEE": ellipsoidal, equal volume, shape, and orientation

"EEV": ellipsoidal, equal volume and equal shape

"VEV": ellipsoidal, equal shape

"VVV": ellipsoidal, varying volume, shape, and orientation

singleComponent

A vector with the following components:

"X": one-dimensional

"XII": spherical

"XXI": diagonal

"XXX": ellipsoidal

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). *MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering*, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[Mclust](#) [mclustBIC](#)

Examples

```
mclustModelNames
```

```
mclustOptions          Set default values for use with MCLUST.
```

Description

Supplies a list of values an enumeration of models for use with *MCLUST*.

Usage

```
mclustOptions(emModelNames=NULL, hcModelNames=NULL,
              bicPlotSymbols=NULL, bicPlotColors=NULL,
              classPlotSymbols=NULL, classPlotColors=NULL, warn=TRUE)
```

Arguments

`emModelNames` A vector of 3-character strings that are associated with multivariate models for which EM estimation is available in *MCLUST*.

The current default is all of the multivariate mixture models supported in *MCLUST*.

The help file for `mclustModelNames` describes the available models.

`hcModelNames` A vector of character strings associated with multivariate models for which model-based hierarchical clustering is available in *MCLUST*.

The current default is the following list:

"EII": spherical, equal volume

"VII": spherical, unequal volume

"EEE": ellipsoidal, equal volume, shape, and orientation

"VVV": ellipsoidal, varying volume, shape, and orientation

`bicPlotSymbols`

A vector whose entries are either integers corresponding to graphics symbols or single characters for plotting BIC curves. The default is

```
c(EII=17, VII=2, EEI=16, EVI=10, VEI=13, VVI=1,
  EEE=15, EEV=12, VEV=7, VVV=0, E=17, V=2).
```

`bicPlotColors`

A vector whose entries are either integers corresponding to colors to BIC curves.

```
c(EII="gray", VII="black",
  EEI="orange", EVI="brown", VEI="red", VVI="magenta",
  EEE="forestgreen", EEV="green", VEV="cyan", VVV="blue",
  E="gray", V="black").
```

`classPlotSymbols`

A vector whose entries are either integers corresponding to graphics symbols or single characters for plotting for classifications. Classes are assigned symbols in the given order. The default is `c(17, 0, 10, 4, 11, 18, 6, 7, 3, 16, 2, 12, 8, 15, 1, 9, 14, 13, 5)`.

`classPlotColors` A vector whose entries are either integers corresponding to graphics symbols or single characters for plotting for classifications. Classes are assigned symbols in the given order. The default is
`"blue", "red", "green", "cyan", "magenta",
"forestgreen", "purple", "orange", "gray", "brown", "black")`

`warn` A logical value allowing some types of warnings to be turned on or off globally. Most of these warnings have to do with situations in which singularities are encountered. The default is `warn = TRUE`.

Details

`mclustOptions` is provided for assigning values to the `.Mclust` list, which is used to supply default values to various functions in MCLUST.

Calls to `mclustOptions` do not in themselves affect the outcome of computations.

Value

A named list in which the names are the names of the arguments and the values are the values supplied to the arguments.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`.Mclust`, `emControl`

Examples

```
irisBIC <- mclustBIC(iris[, -5])
summary(irisBIC, iris[, -5])

.Mclust
.Mclust <- mclustOptions(emModelNames = c("VII", "VVI", "VVV"))
.Mclust

irisBIC <- mclustBIC(iris[, -5])
summary(irisBIC, iris[, -5])

.Mclust <- mclustOptions() # restore default values
.Mclust
```

mclustVariance *Template for variance specification for parameterized Gaussian mixture models.*

Description

Specification of variance parameters for the various types of Gaussian mixture models.

Details

- The `variance` component in the parameters list from the output to e.g. `me` or `mstep` or input to e.g. `estep` may contain one or more of the following arguments, depending on the model:

modelName A character string indicating the model.

d The dimension of the data.

G The number of components in the mixture model.

sigmasq for the one-dimensional models ("E", "V") and spherical models ("EII", "VII"). This is either a vector whose k th component is the variance for the k th component in the mixture model ("V" and "VII"), or a scalar giving the common variance for all components in the mixture model ("E" and "EII").

Sigma For the equal variance models "EII", "EEI", and "EEE". A d by d matrix giving the common covariance for all components of the mixture model.

cholSigma For the equal variance model "EEE". A d by d upper triangular matrix giving the Cholesky factor of the common covariance for all components of the mixture model.

sigma For all multidimensional mixture models. A d by d by G matrix array whose $[, , k]$ th entry is the covariance matrix for the k th component of the mixture model.

cholsigma For the unconstrained covariance mixture model "VVV". A d by d by G matrix array whose $[, , k]$ th entry is the upper triangular Cholesky factor of the covariance matrix for the k th component of the mixture model.

scale For diagonal models "EEI", "EVI", "VEI", "VVI" and constant-shape models "EEV" and "VEV". Either a G -vector giving the scale of the covariance (the d th root of its determinant) for each component in the mixture model, or a single numeric value if the scale is the same for each component.

shape For diagonal models "EEI", "EVI", "VEI", "VVI" and constant-shape models "EEV" and "VEV". Either a G by d matrix in which the k th column is the shape of the covariance matrix (normalized to have determinant 1) for the k th component, or a d -vector giving a common shape for all components.

orientation For the constant-shape models "EEV" and "VEV". Either a d by d by G array whose $[, , k]$ th entry is the orthonormal matrix whose columns are the eigenvectors of the covariance matrix of the k th component, or a d by d orthonormal matrix if the mixture components have a common orientation. The `orientation` component is not needed in spherical and diagonal models, since the principal components are parallel to the coordinate axes so that the orientation matrix is the identity.

In all cases, the value `-1` is used as a placeholder for unknown nonzero entries.

References

- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611:631.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

me	<i>EM algorithm starting with M-step for parameterized MVN mixture models.</i>
----	--

Description

Implements the EM algorithm for MVN mixture models parameterized by eigenvalue decomposition, starting with the maximization step.

Usage

```
me(modelName, data, z, prior = NULL, control = emControl(),
    Vinv = NULL, warn = NULL, ...)
```

Arguments

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
z	A matrix whose $[i, k]$ th entry is an initial estimate of the conditional probability of the i th observation belonging to the k th component of the mixture.
prior	Specification of a conjugate prior on the means and variances. See the help file for <code>priorControl</code> for further information. The default assumes no prior.
control	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .
Vinv	If the model is to include a noise term, <code>Vinv</code> is an estimate of the reciprocal hypervolume of the data region. If set to a negative value or 0, the model will include a noise term with the reciprocal hypervolume estimated by the function <code>hypvol</code> . The default is not to assume a noise term in the model through the setting <code>Vinv=NULL</code> .
warn	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when the estimation fails. The default is set in <code>.Mclust\$warn</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

<code>modelName</code>	A character string identifying the model (same as the input argument).
<code>z</code>	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>parameters</code>	<p>pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.</p> <p>mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model.</p> <p>variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.</p> <p>Vinv The estimate of the reciprocal hypervolume of the data region used in the computation when the input indicates the addition of a noise component to the model.</p>
<code>loglik</code>	The log likelihood for the data in the mixture model.
<code>Attributes:</code>	<ul style="list-style-type: none"> • "info" Information on the iteration. • "WARNING" An appropriate warning if problems are encountered in the computations.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[meE](#), ..., [meVVV](#), [em](#), [mstep](#), [estep](#), [priorControl](#), [mclustModelNames](#), [mclustVariance](#), [mclustOptions](#)

Examples

```
me(modelName = "VVV", data = iris[,-5], z = unmap(iris[,5]))
```

meE	<i>EM algorithm starting with M-step for a parameterized Gaussian mixture model.</i>
-----	--

Description

Implements the EM algorithm for a parameterized Gaussian mixture model, starting with the maximization step.

Usage

```
meE(data, z, prior=NULL, control=emControl(),
     Vinv=NULL, warn=NULL, ...)
meV(data, z, prior=NULL, control=emControl(),
     Vinv=NULL, warn=NULL, ...)
meEII(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meVII(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meEEI(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meVEI(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meEVI(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meVVI(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meEEE(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meEEV(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meVEV(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
meVVV(data, z, prior=NULL, control=emControl(),
       Vinv=NULL, warn=NULL, ...)
```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
z	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
control	A list of control parameters for EM. The defaults are set by the call <code>emControl()</code> .

<code>Vinv</code>	An estimate of the reciprocal hypervolume of the data region, when the model is to include a noise term. Set to a negative value or zero if a noise term is desired, but an estimate is unavailable — in that case function <code>hypvol</code> will be used to obtain the estimate. The default is not to assume a noise term in the model through the setting <code>Vinv=NULL</code> .
<code>warn</code>	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when the estimation fails. The default is set in <code>.Mclust\$warn</code> .
<code>...</code>	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

<code>modelName</code>	A character string identifying the model (same as the input argument).
<code>z</code>	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>parameters</code>	<p>pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.</p> <p>mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model.</p> <p>variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.</p> <p>Vinv The estimate of the reciprocal hypervolume of the data region used in the computation when the input indicates the addition of a noise component to the model.</p>
<code>loglik</code>	The log likelihood for the data in the mixture model.
<code>Attributes:</code>	<ul style="list-style-type: none"> • "info" Information on the iteration. • "WARNING" An appropriate warning if problems are encountered in the computations.

References

- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002a). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.

See Also

[em](#), [me](#), [estep](#), [mclustOptions](#)

Examples

```
meVVV(data = iris[,-5], z = unmap(iris[,5]))
```

mstep

M-step for parameterized Gaussian mixture models.

Description

Maximization step in the EM algorithm for parameterized Gaussian mixture models.

Usage

```
mstep(modelName, data, z, prior = NULL, warn = NULL, ...)
```

Arguments

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
z	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture. In analyses involving noise, this should not include the conditional probabilities for the noise component.
prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
warn	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when the estimation fails. The default is set in <code>.Mclust\$warn</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

modelName	A character string identifying the model (same as the input argument).
parameters	pro A vector whose k th component is the mixing proportion for the k th component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.

mean The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model.

variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for `mclustVariance` for details.

Attributes: "info" For those models with iterative M-steps ("VEI" and "VEV"), information on the iteration.

"WARNING" An appropriate warning if problems are encountered in the computations.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Note

This function computes the M-step only for MVN mixtures, so in analyses involving noise, the conditional probabilities input should exclude those for the noise component.

In contrast to `me` for the EM algorithm, computations in `mstep` are carried out unless failure due to overflow would occur. To impose stricter tolerances on a single `mstep`, use `me` with the `itmax` component of the `control` argument set to 1.

See Also

[mstepE](#), ..., [mstepVVV](#), [emControl](#), [me](#), [estep](#), [mclustOptions](#).

Examples

```
mstep(modelName = "VII", data = iris[,-5], z = unmap(iris[,5]))
```

mstepE

M-step for a parameterized Gaussian mixture model.

Description

Maximization step in the EM algorithm for a parameterized Gaussian mixture model.

Usage

```

mstepE( data, z, prior=NULL, warn=NULL, ...)
mstepV( data, z, prior=NULL, warn=NULL, ...)
mstepEII( data, z, prior=NULL, warn=NULL, ...)
mstepVII( data, z, prior=NULL, warn=NULL, ...)
mstepEEI( data, z, prior=NULL, warn=NULL, ...)
mstepVEI( data, z, prior=NULL, warn=NULL, control=NULL, ...)
mstepEVI( data, z, prior=NULL, warn=NULL, ...)
mstepVVI( data, z, prior=NULL, warn=NULL, ...)
mstepEEE( data, z, prior=NULL, warn=NULL, ...)
mstepEEV( data, z, prior=NULL, warn=NULL, ...)
mstepVEV( data, z, prior=NULL, warn=NULL, control=NULL, ...)
mstepVVV( data, z, prior=NULL, warn=NULL, ...)

```

Arguments

data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
z	A matrix whose $[i, k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture. In analyses involving noise, this should not include the conditional probabilities for the noise component.
prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
warn	A logical value indicating whether or not certain warnings (usually related to singularity) should be issued when the estimation fails. The default is set in <code>.Mclust\$warn</code> .
control	Values controlling termination for models "VEI" and "VEV" that have an iterative M-step. This should be a list with components named <i>itmax</i> and <i>tol</i> . These components can be of length 1 or 2; in the latter case, <code>mstep</code> will use the second value, under the assumption that the first applies to an outer iteration (as in the function <code>me</code>). The default uses the default values from the function <code>emControl</code> , which sets no limit on the number of iterations, and a relative tolerance of <code>sqrt(.Machine\$double.eps)</code> on successive iterates.
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

modelName	A character string identifying the model (same as the input argument).
parameters	pro A vector whose k th component is the mixing proportion for the k th component of the mixture model. If the model includes a Poisson term for noise, there should be one more mixing proportion than the number of Gaussian components.

mean The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model.

variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for `mclustVariance` for details.

Attributes: "info" For those models with iterative M-steps ("VEI" and "VEV"), information on the iteration.

"WARNING" An appropriate warning if problems are encountered in the computations.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Note

This function computes the M-step only for MVN mixtures, so in analyses involving noise, the conditional probabilities input should exclude those for the noise component.

In contrast to `me` for the EM algorithm, computations in `mstep` are carried out unless failure due to overflow would occur. To impose stricter tolerances on a single `mstep`, use `me` with the `itmax` component of the `control` argument set to 1.

See Also

`mstep`, `me`, `estep`, `priorControl` `emControl`

Examples

```
mstepVII(data = iris[,-5], z = unmap(iris[,5]))
```

mvn

Univariate or Multivariate Normal Fit

Description

Computes the mean, covariance, and loglikelihood from fitting a single Gaussian to given data (univariate or multivariate normal).

Usage

```
mvn(modelName, data, prior = NULL, warn = NULL, ...)
```

Arguments

modelName	A character string representing a model name. This can be either "Spherical", "Diagonal", or "Ellipsoidal" or else "X" for one-dimensional data, "XII" for a spherical Gaussian, "XXI" for a diagonal Gaussian "XXX" for a general ellipsoidal Gaussian
data	A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
warn	A logical value indicating whether or not a warning should be issued whenever a singularity is encountered. The default is set in <code>.Mclust\$warn</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

A list including the following components:

modelName	A character string identifying the model (same as the input argument).
parameters	mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i> th column is the mean of the <i>k</i> th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
loglik	The log likelihood for the data in the mixture model.
Attributes:	<ul style="list-style-type: none"> "WARNING" An appropriate warning if problems are encountered in the computations.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mvnX](#), [mvnXII](#), [mvnXXI](#), [mvnXXX](#), [mclustModelNames](#)

Examples

```

n <- 1000

set.seed(0)
x <- rnorm(n, mean = -1, sd = 2)
mvn(modelName = "X", x)

mu <- c(-1, 0, 1)

set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% (2*diag(3)),
            MARGIN = 2, STATS = mu, FUN = "+")
mvn(modelName = "XII", x)
mvn(modelName = "Spherical", x)

set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% diag(1:3),
            MARGIN = 2, STATS = mu, FUN = "+")
mvn(modelName = "XXI", x)
mvn(modelName = "Diagonal", x)

Sigma <- matrix(c(9,-4,1,-4,9,4,1,4,9), 3, 3)
set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% chol(Sigma),
            MARGIN = 2, STATS = mu, FUN = "+")
mvn(modelName = "XXX", x)
mvn(modelName = "Ellipsoidal", x)

```

mvnX

*Univariate or Multivariate Normal Fit***Description**

Computes the mean, covariance, and loglikelihood from fitting a single Gaussian (univariate or multivariate normal).

Usage

```

mvnX(data, prior = NULL, warn = NULL, ...)
mvnXII(data, prior = NULL, warn = NULL, ...)
mvnXXI(data, prior = NULL, warn = NULL, ...)
mvnXXX(data, prior = NULL, warn = NULL, ...)

```

Arguments

`data` A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.

prior	Specification of a conjugate prior on the means and variances. The default assumes no prior.
warn	A logical value indicating whether or not a warning should be issued whenever a singularity is encountered. The default is set in <code>.Mclust\$warn</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Details

- `mvnXII` computes the best fitting Gaussian with the covariance restricted to be a multiple of the identity.
- `mvnXXI` computes the best fitting Gaussian with the covariance restricted to be diagonal.
- `mvnXXX` computes the best fitting Gaussian with ellipsoidal (unrestricted) covariance.

Value

A list including the following components:

modelName	A character string identifying the model (same as the input argument).
parameters	mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i> th column is the mean of the <i>k</i> th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
loglik	The log likelihood for the data in the mixture model.
Attributes:	• "WARNING" An appropriate warning if problems are encountered in the computations.

References

- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3: An R Package for Normal Mixture Modeling and Model-Based Clustering, Technical Report, Department of Statistics, University of Washington.

See Also

[mvn](#), [mstepE](#)

Examples

```
n <- 1000

set.seed(0)
x <- rnorm(n, mean = -1, sd = 2)
mvnX(x)
```

```

mu <- c(-1, 0, 1)

set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% (2*diag(3)),
            MARGIN = 2, STATS = mu, FUN = "+")
mvnXII(x)

set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% diag(1:3),
            MARGIN = 2, STATS = mu, FUN = "+")
mvnXXI(x)

Sigma <- matrix(c(9,-4,1,-4,9,4,1,4,9), 3, 3)
set.seed(0)
x <- sweep(matrix(rnorm(n*3), n, 3) %*% chol(Sigma),
            MARGIN = 2, STATS = mu, FUN = "+")
mvnXXX(x)

```

nVarParams

Number of Variance Parameters in Gaussian Mixture Models

Description

Gives the number of variance parameters for parameterizations of the Gaussian mixture model that are used in MCLUST.

Usage

```
nVarParams(modelName, d, G)
```

Arguments

modelName	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
d	The dimension of the data. Not used for models in which neither the shape nor the orientation varies.
G	The number of components in the Gaussian mixture model used to compute <code>loglik</code> .

Details

To get the total number of parameters in model, add $G*d$ for the means and $G-1$ for the mixing proportions if they are unequal.

Value

The number of variance parameters in the corresponding Gaussian mixture model.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611:631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[bic](#)

Examples

```
sapply(.Mclust$emModelNames, nVarParams, d=2, G=1)
```

partconv

Numeric Encoding of a Partitioning

Description

Converts a vector interpreted as a classification or partitioning into a numeric vector.

Usage

```
partconv(x, consec=TRUE)
```

Arguments

x	A vector interpreted as a classification or partitioning.
consec	Logical value indicating whether or not to consecutive class numbers should be used .

Value

Numeric encoding of x. When `consec = TRUE`, the distinct values in x are numbered by the order in which they appear. When `consec = FALSE`, each distinct value in x is numbered by the index corresponding to its first appearance in x.

See Also

[partuniq](#)

Examples

```
partconv(iris[,5])

set.seed(0)
cl <- sample(LETTERS[1:9], 25, replace=TRUE)
partconv(cl, consec=FALSE)
partconv(cl, consec=TRUE)
```

partuniq

Classifies Data According to Unique Observations

Description

Gives a one-to-one mapping from unique observations to rows of a data matrix.

Usage

```
partuniq(x)
```

Arguments

`x` Matrix of observations.

Value

A vector of length `nrow(x)` with integer entries. An observation `k` is assigned an integer `i` whenever observation `i` is the first row of `x` that is identical to observation `k` (note that `i <= k`).

See Also

[partconv](#)

Examples

```
set.seed(0)

mat <- data.frame(lets = sample(LETTERS[1:2], 9, TRUE), nums = sample(1:2, 9, TRUE))
mat

ans <- partuniq(mat)
ans

partconv(ans, consec=TRUE)
```

plot.Mclust

*Plot Model-Based Clustering Results***Description**

Plot model-based clustering results: BIC, classification, uncertainty and (for one- and two-dimensional data) density.

Usage

```
## S3 method for class 'Mclust':
plot(x, data = NULL, what = c("BIC", "classification",
    "uncertainty", "density"), dims = c(1,2), xlab = NULL, ylim = NULL,
    legendArgs = list(x = "bottomright", ncol = 2, cex = 1),
    identify = TRUE, ...)
```

Arguments

x	Output from Mclust.
data	The data used to produce x.
what	Choose one or more of: "BIC", "classification", "uncertainty". If the data dimension is less than 3, "density" can also be chosen.
dims	A vector of length 2 giving the integer dimensions of the desired coordinate projections for multivariate data. The default is c(1,2), in which the first dimension is plotted against the second.
xlab	Optional label for the horizontal axis of the BIC plot.
ylim	Optional limits for the vertical axis of the BIC plot.
legendArgs	Arguments to pass to the legend function. Set to NULL for no legend.
identify	A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
...	Other graphics parameters.

Details

For more flexibility in plotting, use `mclust1Dplot`, `mclust2Dplot`, `surfacePlot`, `coordProj`, or `randProj`.

Value

Model-based clustering plots: BIC values used for choosing the number of clusters. For data in more than two dimensions, a pairs plot of the showing the classification, a coordinate projections of the data showing location of the mixture components, classification, and uncertainty. For one- and two- dimensional data, plots showing location of the mixture components, classification, uncertainty, and density.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[Mclust](#), [mclust1Dplot](#), [mclust2Dplot](#), [surfacePlot](#), [coordProj](#), [randProj](#)

Examples

```
## Not run:
plot(Mclust(precip), precip)

plot(Mclust(faithful), faithful)

plot(Mclust(iris[, -5]), iris[, -5])

## End(Not run)
```

plot.mclustBIC *BIC Plot*

Description

Plots the BIC from mclust modeling via function mclustBIC.

Usage

```
## S3 method for class 'mclustBIC':
plot(x, G = NULL, modelNames = NULL, symbols = NULL,
      colors = NULL, xlab = NULL, ylim = NULL,
      legendArgs = list(x="bottomright", ncol=2, cex=1),
      CEX = 1, ...)
```

Arguments

x	Output from mclustBIC.
G	One or more numbers of components corresponding to models fit in x. The default is to plot the BIC for all of the numbers of components fit.
modelNames	One or more model names corresponding to models fit in x. The default is to plot the BIC for all of the models fit.

symbols	Either an integer or character vector assigning a plotting symbol to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
colors	Either an integer or character vector assigning a color to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .
xlab	Optional label for the horizontal axis of the BIC plot.
ylim	Optional limits for the vertical axis of the BIC plot.
legendArgs	Arguments to pass to the <code>legend</code> function. Set to <code>NULL</code> for no legend.
CEX	A scalar controlling the size of the splot symbols.
...	Other graphics parameters.

Value

A plot of the BIC values for the models specified in the `modelName`s argument.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustBIC](#)

Examples

```
## Not run:
plot(mclustBIC(precip), legendArgs = list(x = "bottomleft"))

plot(mclustBIC(faithful))

plot(mclustBIC(iris[, -5]))

## End(Not run)
```

plot.mclustDA *Plotting method for MclustDA discriminant analysis.*

Description

Plots training and test data, known training data classification, mclustDA test data classification, and/or training errors.

Usage

```
## S3 method for class 'mclustDA':  
plot(x, trainData, testData, ...)
```

Arguments

x	The object produced by applying mclustDA with trainingData and classification labels to testData.
trainData	The numeric vector, matrix, or data frame of training observations used to obtain x.
testData	A numeric vector, matrix, or data frame of training observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
...	Further arguments to the lower level plotting functions.

Value

Plots of the following: training and test data, known training data classification, mclustDA test data classification, and (if test labels were supplied to mclustDA when x was created) test errors.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustDA](#)

Examples

```

n <- 250 ## create artificial data
set.seed(1)
triModal <- c(rnorm(n,-5), rnorm(n,0), rnorm(n,5))
triClass <- c(rep(1,n), rep(2,n), rep(3,n))

odd <- seq(from = 1, to = length(triModal), by = 2)
even <- odd + 1
triMclustDA <- mclustDA(train=list(data=triModal[odd],labels=triClass[odd]),
                       test= list(data=triModal[even],labels=triClass[even]),
                       verbose = TRUE)

names(triMclustDA)
## Not run:
plot(triMclustDA, trainData = triModal[odd], testData = triModal[even])

## End(Not run)

odd <- seq(from = 1, to = nrow(cross), by = 2)
even <- odd + 1
crossMclustDA <- mclustDA( train=list(data=cross[odd,-1],
                                     labels=cross[odd,1]),
                          test= list(data=cross[even,-1],labels=cross[even,1]),
                          verbose = TRUE)

## Not run:
plot(crossMclustDA, trainData = cross[odd,-1], testData = cross[even,-1])

## End(Not run)

odd <- seq(from = 1, to = nrow(iris), by = 2)
even <- odd + 1
irisMclustDA <- mclustDA(train=list(data=iris[odd,-5],labels=iris[odd,5]),
                        test= list(data=iris[even,-5],labels=iris[even,5]),
                        verbose = TRUE)

## Not run:
plot(irisMclustDA, trainData = iris[odd,-5], testData = iris[even,-5])

## End(Not run)

```

plot.mclustDAtrain *Plot mclustDA training models.*

Description

Plots representation of the models produced by mclustDAtrain. For multidimensional data, the plot is a coordinate projection and the ellipses shown correspond to the covariance matrices.

Usage

```
## S3 method for class 'mclustDAtrain':
plot(x, data, dims=c(1,2), symbols=NULL, colors=NULL,
      scale = FALSE, xlim=NULL, ylim=NULL, CEX = 1, ...)
```

Arguments

x	An object produced by a call to <code>mclustDAtrain</code> .
data	A numeric matrix or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
dims	A vector of length 2 giving the integer dimensions of the desired coordinate projections. The default is <code>c(1,2)</code> , in which the first dimension is plotted against the second.
symbols	Either an integer or character vector assigning a plotting symbol to each unique class in classification. Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
colors	Either an integer or character vector assigning a color to each unique class in classification. Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .
scale	A logical variable indicating whether or not the two chosen dimensions should be plotted on the same scale, and thus preserve the shape of the distribution. Default: <code>scale=FALSE</code>
xlim, ylim	Arguments specifying bounds for the ordinate, abscissa of the plot. This may be useful for when comparing plots.
CEX	An argument specifying the size of the plotting symbols. The default value is 1.
...	Other graphics parameters.

Side Effects

A plot showing a two-dimensional coordinate projection of the data, together with the location of the mixture components, classification, uncertainty, and/or classification errors.

References

- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[coordProj](#), [mclust1Dplot](#), [mclust2Dplot](#), [mclustOptions](#)

Examples

```
odd <- seq(from = 1, to = nrow(iris), by = 2)

irisTrain <- mclustDAttrain(data = iris[odd,-5], labels = iris[odd,5])
## Not run:
plot(irisTrain, iris[odd,-5])

## End(Not run)
```

priorControl *Conjugate Prior for Gaussian Mixtures.*

Description

Specify a conjugate prior for Gaussian mixtures.

Usage

```
priorControl(functionName = "defaultPrior", ...)
```

Arguments

`functionName` The name of the function specifying the conjugate prior. The default function is `defaultPrior`, which can be used as a template for alternative specification.

`...` Optional named arguments to the function specified in `functionName` together with their values.

Details

`priorControl` is used to specify a conjugate prior for EM within *MCLUST*.

Value

A list with the function name as the first component. The remaining components (if any) consist of a list of arguments to the function with assigned values.

References

- C. Fraley and A. E. Raftery (2005). Bayesian regularization for normal mixture estimation and model-based clustering. Technical Report, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification* 24:155-181.
- C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.
- C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

See Also

`mclustBIC`, `me`, `mstep`, `defaultPrior`

Examples

```
# default prior
irisBIC <- mclustBIC(iris[,-5], prior = priorControl())
summary(irisBIC, iris[,-5])

# no prior on the mean; default prior on variance
irisBIC <- mclustBIC(iris[,-5], prior = priorControl(shrinkage = 0))
summary(irisBIC, iris[,-5])
```

randProj

Random projections of multidimensional data modeled by an MVN mixture.

Description

Plots random projections given multidimensional data and parameters of an MVN mixture model for the data.

Usage

```
randProj(data, seeds=0, parameters=NULL, z=NULL,
         classification=NULL, truth=NULL, uncertainty=NULL,
         what = c("classification", "errors", "uncertainty"),
         quantiles = c(0.75, 0.95), symbols=NULL, colors=NULL, scale = FALSE,
         xlim=NULL, ylim=NULL, CEX = 1, PCH = ".", identify = FALSE, ...)
```

Arguments

<code>data</code>	A numeric matrix or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.
<code>seeds</code>	A vector of integer seeds for random number generation. Elements should be in the range 0 : 1000. Each seed should produce a different projection.
<code>parameters</code>	A named list giving the parameters of an <i>MCLUST</i> model, used to produce superimposing ellipses on the plot. The relevant components are as follows: <ul style="list-style-type: none"> mean The mean for each component. If there is more than one component, this is a matrix whose <i>k</i>th column is the mean of the <i>k</i>th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.

<code>z</code>	A matrix in which the $[i, k]$ th entry gives the probability of observation i belonging to the k th class. Used to compute <code>classification</code> and <code>uncertainty</code> if those arguments aren't available.
<code>classification</code>	A numeric or character vector representing a classification of observations (rows) of data. If present argument <code>z</code> will be ignored.
<code>truth</code>	A numeric or character vector giving a known classification of each data point. If <code>classification</code> or <code>z</code> is also present, this is used for displaying classification errors.
<code>uncertainty</code>	A numeric vector of values in $(0, 1)$ giving the uncertainty of each data point. If present argument <code>z</code> will be ignored.
<code>what</code>	Choose from one of the following three options: "classification" (default), "errors", "uncertainty".
<code>quantiles</code>	A vector of length 2 giving quantiles used in plotting uncertainty. The smallest symbols correspond to the smallest quantile (lowest uncertainty), medium-sized (open) symbols to points falling between the given quantiles, and large (filled) symbols to those in the largest quantile (highest uncertainty). The default is $(0.75, 0.95)$.
<code>symbols</code>	Either an integer or character vector assigning a plotting symbol to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotSymbols</code> .
<code>colors</code>	Either an integer or character vector assigning a color to each unique class in <code>classification</code> . Elements in <code>colors</code> correspond to classes in order of appearance in the sequence of observations (the order used by the function <code>unique</code>). The default is given is <code>.Mclust\$classPlotColors</code> .
<code>scale</code>	A logical variable indicating whether or not the two chosen dimensions should be plotted on the same scale, and thus preserve the shape of the distribution. Default: <code>scale=FALSE</code>
<code>xlim, ylim</code>	Arguments specifying bounds for the ordinate, abscissa of the plot. This may be useful for when comparing plots.
<code>CEX</code>	An argument specifying the size of the plotting symbols. The default value is 1.
<code>PCH</code>	An argument specifying the symbol to be used when a classification has not been specified for the data. The default value is a small dot ".".
<code>identify</code>	A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
<code>...</code>	Other graphics parameters.

Side Effects

A plot showing a random two-dimensional projection of the data, together with the location of the mixture components, classification, uncertainty, and/or classification errors.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3: An R Package for Normal Mixture Modeling and Model-Based Clustering, Technical Report, Department of Statistics, University of Washington.

See Also

`clPairs`, `coordProj`, `mclust2Dplot`, `mclustOptions`

Examples

```
est <- meVVV(iris[,-5], unmap(iris[,5]))

## Not run:
par(pty = "s", mfrow = c(1,1))
randProj(iris[,-5], seeds=1:3, parameters = est$parameters, z = est$z,
         what = "classification", identify = TRUE)
randProj(iris[,-5], seeds=1:3, parameters = est$parameters, z = est$z,
         truth = iris[,5], what = "errors", identify = TRUE)
randProj(iris[,-5], seeds=1:3, parameters = est$parameters, z = est$z,
         what = "uncertainty", identify = TRUE)

## End(Not run)
```

sigma2decomp

Convert mixture component covariances to decomposition form.

Description

Converts a set of covariance matrices from representation as a 3-D array to a parameterization by eigenvalue decomposition.

Usage

```
sigma2decomp(sigma, G=NULL, tol=NULL, ...)
```

Arguments

sigma	Either a 3-D array whose $[,k]$ th component is the covariance matrix for the k th component in an MVN mixture model, or a single covariance matrix in the case that all components have the same covariance.
G	The number of components in the mixture. When <code>sigma</code> is a 3-D array, the number of components can be inferred from its dimensions.
tol	Tolerance for determining whether or not the covariances have equal volume, shape, and or orientation. The default is the square root of the relative machine precision, <code>sqrt(.Machine\$double.eps)</code> , which is about $1.e-8$.
...	Catches unused arguments from an indirect or list call via <code>do.call</code> .

Value

The covariance matrices for the mixture components in decomposition form, including the following components:

<code>modelName</code>	A character string indicating the inferred model. The help file for <code>mclustModelNames</code> describes the available models.
<code>d</code>	The dimension of the data.
<code>G</code>	The number of components in the mixture model.
<code>scale</code>	Either a G -vector giving the scale of the covariance (the d th root of its determinant) for each component in the mixture model, or a single numeric value if the scale is the same for each component.
<code>shape</code>	Either a G by d matrix in which the k th column is the shape of the covariance matrix (normalized to have determinant 1) for the k th component, or a d -vector giving a common shape for all components.
<code>orientation</code>	Either a d by d by G array whose $[, , k]$ th entry is the orthonormal matrix whose columns are the eigenvectors of the covariance matrix of the k th component, or a d by d orthonormal matrix if the mixture components have a common orientation. The <code>orientation</code> component of <code>decomp</code> can be omitted in spherical and diagonal models, for which the principal components are parallel to the coordinate axes so that the orientation matrix is the identity.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[decomp2sigma](#)

Examples

```
meEst <- meEEE(iris[,-5], unmap(iris[,5]))
names(meEst$parameters$variance)
meEst$parameters$variance$Sigma

sigma2decomp(meEst$parameters$variance$Sigma, G = length(unique(iris[,5])))
```

sim

*Simulate from Parameterized MVN Mixture Models***Description**

Simulate data from parameterized MVN mixture models.

Usage

```
sim(modelName, parameters, n, seed = NULL, ...)
```

Arguments

<code>modelName</code>	A character string indicating the model. The help file for <code>mclustModelNames</code> describes the available models.
<code>parameters</code>	A list with the following components: <ul style="list-style-type: none"> pro A vector whose kth component is the mixing proportion for the kth component of the mixture model. If missing, equal proportions are assumed. mean The mean for each component. If there is more than one component, this is a matrix whose kth column is the mean of the kth component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
<code>n</code>	An integer specifying the number of data points to be simulated.
<code>seed</code>	An optional integer argument to <code>set.seed</code> for reproducible random class assignment. By default the current seed will be used. Reproducibility can also be achieved by calling <code>set.seed</code> before calling <code>sim</code> .
<code>...</code>	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Details

This function can be used with an indirect or list call using `do.call`, allowing the output of e.g. `mstep`, `em`, `me`, `Mclust` to be passed directly without the need to specify individual parameters as arguments.

Value

A matrix in which first column is the classification and the remaining columns are the n observations simulated from the specified MVN mixture model.

Attributes: • "modelName" A character string indicating the variance model used for the simulation.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[simE](#), ..., [simVVV](#), [Mclust](#), [mstep](#), [do.call](#)

Examples

```
irisBIC <- mclustBIC(iris[,-5])
irisModel <- mclustModel(iris[,-5], irisBIC)
names(irisModel)
irisSim <- sim(modelName = irisModel$modelName,
              parameters = irisModel$parameters,
              n = nrow(iris))

## Not run:
do.call("sim", irisModel) # alternative call

## End(Not run)

par(pty = "s", mfrow = c(1,2))

dimnames(irisSim) <- list(NULL, c("dummy", (dimnames(iris)[[2]])[-5]))

dimens <- c(1,2)
lim1 <- apply(iris[,dimens],2,range)
lim2 <- apply(irisSim[,dimens+1],2,range)
lims <- apply(rbind(lim1,lim2),2,range)
xlim <- lims[,1]
ylim <- lims[,2]

coordProj(iris[,-5], parameters=irisModel$parameters,
          classification=map(irisModel$z),
          dimens=dimens, xlim=xlim, ylim=ylim)

coordProj(iris[,-5], parameters=irisModel$parameters,
          classification=map(irisModel$z), truth = irisSim[,-1],
          dimens=dimens, xlim=xlim, ylim=ylim)

irisModel3 <- mclustModel(iris[,-5], irisBIC, G=3)
irisSim3 <- sim(modelName = irisModel3$modelName,
              parameters = irisModel3$parameters, n = 500, seed = 1)

## Not run:
irisModel3$n <- NULL
irisSim3 <- do.call("sim",c(list(n=500,seed=1),irisModel3)) # alternative call
```

```
## End(Not run)
clPairs(irisSim3[,-1], cl = irisSim3[,1])
```

simE

Simulate from a Parameterized MVN Mixture Model

Description

Simulate data from a parameterized MVN mixture model.

Usage

```
simE(parameters, n, seed = NULL, ...)
simV(parameters, n, seed = NULL, ...)
simEII(parameters, n, seed = NULL, ...)
simVII(parameters, n, seed = NULL, ...)
simEEI(parameters, n, seed = NULL, ...)
simVEI(parameters, n, seed = NULL, ...)
simEVI(parameters, n, seed = NULL, ...)
simVVI(parameters, n, seed = NULL, ...)
simEEE(parameters, n, seed = NULL, ...)
simEEV(parameters, n, seed = NULL, ...)
simVEV(parameters, n, seed = NULL, ...)
simVVV(parameters, n, seed = NULL, ...)
```

Arguments

parameters	A list with the following components: pro A vector whose k th component is the mixing proportion for the k th component of the mixture model. If missing, equal proportions are assumed. mean The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model. variance A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
n	An integer specifying the number of data points to be simulated.
seed	An optional integer argument to <code>set.seed</code> for reproducible random class assignment. By default the current seed will be used. Reproducibility can also be achieved by calling <code>set.seed</code> before calling <code>sim</code> .
...	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Details

This function can be used with an indirect or list call using `do.call`, allowing the output of e.g. `mstep`, `em.me`, `Mclust`, to be passed directly without the need to specify individual parameters as arguments.

Value

A matrix in which first column is the classification and the remaining columns are the n observations simulated from the specified MVN mixture model.

Attributes: • "modelName" A character string indicating the variance model used for the simulation.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[sim](#), [Mclust](#), [mstepE](#), [do.call](#)

Examples

```
d <- 2
G <- 2
scale <- 1
shape <- c(1, 9)

O1 <- diag(2)
O2 <- diag(2)[,c(2,1)]
O <- array(cbind(O1,O2), c(2, 2, 2))
O

variance <- list(d= d, G = G, scale = scale, shape = shape, orientation = 0)
mu <- matrix(0, d, G) ## center at the origin
simdat <- simEEV( n = 200,
                 parameters = list(pro=c(1,1),mean=mu,variance=variance),
                 seed = NULL)

c1 <- simdat[,1]

## Not run:
sigma <- array(apply(O, 3, function(x,y) crossprod(x*y),
                    y = sqrt(scale*shape)), c(2,2,2))
paramList <- list(mu = mu, sigma = sigma)
coordProj( simdat, paramList = paramList, classification = c1)

## End(Not run)
```

summary.mclustBIC *Summary Function for model-based clustering.*

Description

Optimal model characteristics and classification for model-based clustering via mclustBIC.

Usage

```
## S3 method for class 'mclustBIC':
summary(object, data, G, modelNames, ...)
```

Arguments

object	An "mclustBIC" object, which is the result of applying mclustBIC to data.
data	The matrix or vector of observations used to generate 'object'.
G	A vector of integers giving the numbers of mixture components (clusters) from which the best model according to BIC will be selected (as.character(G) must be a subset of the row names of object). The default is to select the best model for all numbers of mixture components used to obtain object.
modelNames	A vector of integers giving the model parameterizations from which the best model according to BIC will be selected (as.character(model) must be a subset of the column names of object). The default is to select the best model for parameterizations used to obtain object.
...	Not used. For generic/method consistency.

Value

A list giving the optimal (according to BIC) parameters, conditional probabilities z , and loglikelihood, together with the associated classification and its uncertainty.

The details of the output components are as follows:

modelName	A character string denoting the model corresponding to the optimal BIC.
n	The number of observations in the data.
d	The dimension of the data.
G	The number of mixture components in the model corresponding to the optimal BIC.
bic	The optimal BIC value.
loglik	The loglikelihood corresponding to the optimal BIC.
z	A matrix whose $[i,k]$ th entry is the probability that observation i in the data belongs to the k th class.
classification	map(z): The classification corresponding to z .

uncertainty The uncertainty associated with the classification.

Attributes:

- "bestBICvalues" Some of the best bic values for the analysis.
- "prior" The prior as specified in the input.
- "control" The control parameters for EM as specified in the input.
- "initialization" The parameters used to initial EM for computing the maximum likelihood values used to obtain the BIC.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustBIC](#) [mclustModel](#)

Examples

```
irisBIC <- mclustBIC(iris[,-5])
summary(irisBIC, iris[,-5])
summary(irisBIC, iris[,-5], G = 1:6, modelNames = c("VII", "VVI", "VVV"))
```

```
summary.mclustDAtest
```

Classification and posterior probability from mclustDAtest.

Description

Extract classifications and the corresponding posterior probabilities from `mclustDAtest`.

Usage

```
## S3 method for class 'mclustDAtest':
summary(object, pro=NULL, ...)
```

Arguments

`object` The output of `mclustDAtest`.

`pro` Optional prior probabilities for each class in the training data.

`...` Not used. For generic/method consistency.

Value

A list with the following two components:

classification	The classification from <code>mclustDAtest</code> .
z	Matrix of posterior probabilities in which the $[i, j]$ th entry is the probability of observation i belonging to class j .

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`classError`, `mclustDAtest`

Examples

```
odd <- seq(1, nrow(cross), by = 2)
train <- mclustDAtrain(cross[odd,-1], labels = cross[odd,1]) ## training step
summary(train)

even <- odd + 1
test <- mclustDAtest(cross[even,-1], train) ## compute model densities
testSummary <- summary(test)
names(testSummary)
classError(testSummary$classification, cross[even,1])
```

summary.mclustDAtrain

Models and classifications from mclustDAtrain

Description

Extracts the models selected in `mclustDAtrain` and the corresponding classifications.

Usage

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

Arguments

object The output of `mclustDAtrain`.
 ... Not used. For generic/method consistency.

Value

A list identifying the model selected by `mclustDAtrain` for each class of training data and the corresponding classification.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.
 C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustDAtrain](#)

Examples

```
odd <- seq(1, nrow(cross), by = 2)
train <- mclustDAtrain(cross[odd,-1], labels = cross[odd,1])
summary(train)
```

summary.mclustModel

Summary Function for MCLUST Models

Description

Classification and uncertainty for a mixture models as output by `mclustModel`.

Usage

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

Arguments

object An "mclustModel" object.
 ... Not used. For generic/method consistency.

Value

A data frame giving the classification and uncertainty corresponding to the model.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclustModel](#)

Examples

```
irisBIC <- mclustBIC(iris[,-5])
irisModel <- mclustModel(iris[,-5], irisBIC)
summary(irisModel)
```

surfacePlot

Density or uncertainty surface for two dimensional mixtures.

Description

Plots a density or uncertainty surface given data in more than two dimensions and parameters of an MVN mixture model for the data.

Usage

```
surfacePlot(data, parameters,
            type = c("contour", "image", "persp"),
            what = c("density", "uncertainty"),
            transformation = c("none", "log", "sqrt"),
            grid = 50, nlevels = 20, scale = FALSE,
            xlim=NULL, ylim=NULL,
            identify = FALSE, verbose = FALSE, swapAxes = FALSE, ...)
```

Arguments

data A numeric vector, matrix, or data frame of observations. Categorical variables are not allowed. If a matrix or data frame, rows correspond to observations and columns correspond to variables.

parameters A named list giving the parameters of an *MCLUST* model, used to produce superimposing ellipses on the plot. The relevant components are as follows:

	mean	The mean for each component. If there is more than one component, this is a matrix whose k th column is the mean of the k th component of the mixture model.
	variance	A list of variance parameters for the model. The components of this list depend on the model specification. See the help file for <code>mclustVariance</code> for details.
type		Choose from one of the following three options: "contour" (default), "image", "persp" indicating the plot type.
what		Choose from one of the following options: "density" (default), "uncertainty" indicating what to plot.
transformation		Choose from one of the following three options: "none" (default), "log", "sqrt" indicating a transformation to be applied before plotting.
grid		The number of grid points (evenly spaced on each axis). The mixture density and uncertainty is computed at <code>grid</code> x <code>grid</code> points to produce the surface plot. Default: 50.
nlevels		The number of levels to use for a contour plot. Default: 20.
scale		A logical variable indicating whether or not the two dimensions should be plotted on the same scale, and thus preserve the shape of the distribution. The default is not to scale.
xlim, ylim		An argument specifying bounds for the ordinate, abscissa of the plot. This may be useful for when comparing plots.
identify		A logical variable indicating whether or not to add a title to the plot identifying the dimensions used.
verbose		A logical variable telling whether or not to print an indication that the function is in the process of computing values at the grid points, which typically takes some time to complete.
swapAxes		A logical variable indicating whether or not the axes should be swapped for the plot.
...		Other graphics parameters.

Value

An invisible list with components `x`, `y`, and `z` in which `x` and `y` are the values used to define the grid and `z` is the transformed density or uncertainty at the grid points.

Side Effects

A plots showing (a transformation of) the density or uncertainty for the given mixture model and data.

Details

For an image plot, a color scheme may need to be selected on the display device in order to view the plot.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[mclust2Dplot](#)

Examples

```
faithfulModel <- mclustModel(faithful,mclustBIC(faithful))
surfacePlot(faithful, parameters = faithfulModel$parameters,
            type = "contour", what = "density", transformation = "none",
            drawlabels = FALSE)
```

uncerPlot

Uncertainty Plot for Model-Based Clustering

Description

Displays the uncertainty in converting a conditional probability from EM to a classification in model-based clustering.

Usage

```
uncerPlot(z, truth, ...)
```

Arguments

<code>z</code>	A matrix whose $[i,k]$ th entry is the conditional probability of the i th observation belonging to the k th component of the mixture.
<code>truth</code>	A numeric or character vector giving the true classification of the data.
<code>...</code>	Provided to allow lists with elements other than the arguments can be passed in indirect or list calls with <code>do.call</code> .

Details

When `truth` is provided and the number of classes is compatible with `z`, the function `compareClass` is used to find best correspondence between classes in `truth` and `z`.

Value

A plot of the uncertainty profile of the data, with uncertainties in increasing order of magnitude. If `truth` is supplied and the number of classes is the same as the number of columns of `z`, the uncertainty of the misclassified data is marked by vertical lines on the plot.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

`mclustBIC`, `em`, `me`, `mapClass`

Examples

```
irisBIC <- mclustBIC(iris[, -5])
irisModel3 <- mclustModel(iris[, -5], irisBIC, G = 3)

uncerPlot(z = irisModel3$z)

uncerPlot(z = irisModel3$z, truth = iris[, 5])
```

unmap

Indicator Variables given Classification

Description

Converts a classification into a matrix of indicator variables.

Usage

```
unmap(classification, groups=NULL, noise=NULL, ...)
```

Arguments

<code>classification</code>	A numeric or character vector. Typically the distinct entries of this vector would represent a classification of observations in a data set.
<code>groups</code>	A numeric or character vector indicating the groups from which <code>classification</code> is drawn. If not supplied, the default is to assumed to be the unique entries of <code>classification</code> .
<code>noise</code>	A single numeric or character value used to indicate the value of <code>groups</code> corresponding to noise.
<code>...</code>	Catches unused arguments in indirect or list calls via <code>do.call</code> .

Value

An n by m matrix of $(0,1)$ indicator variables, where n is the length of `classification` and m is the number of unique values or symbols in `classification`. Columns are labeled by the unique values in `classification`, and the $[i, j]$ th entry is 1 if `classification[i]` is the j th unique value or symbol in sorted order `classification`. If a noise value of symbol is designated, the corresponding indicator variables are relocated to the last column of the matrix.

References

C. Fraley and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97:611-631.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

See Also

[map](#), [estep](#), [me](#)

Examples

```
z <- unmap(iris[,5])
z[1:5, ]

emEst <- me(modelName = "VVV", data = iris[,-5], z = z)
emEst$z[1:5, ]

map(emEst$z)
```

wreath

Data Simulated from a 14-Component Mixture

Description

A dataset consisting of 1000 observations drawn from a 14-component normal mixture in which the covariances of the components have the same size and shape but differ in orientation.

Usage

```
data(wreath)
```

References

C. Fraley, A. E. Raftery and R. Wehrens (2005). Incremental model-based clustering for large datasets with small clusters. *Journal of Computational and Graphical Statistics* 14:1:18.

C. Fraley and A. E. Raftery (2006). MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report no. 504, Department of Statistics, University of Washington.

Index

*Topic **cluster**

- adjustedRandIndex, 1
- bic, 3
- bicEMtrain, 4
- cdens, 5
- cdenseE, 6
- classError, 9
- clPairs, 10
- coordProj, 11
- cv1EMtrain, 14
- decomp2sigma, 15
- defaultPrior, 16
- Defaults.Mclust, 18
- dens, 20
- em, 21
- emControl, 23
- emE, 25
- estep, 27
- estepE, 28
- hc, 30
- hcE, 32
- hclass, 33
- hypvol, 34
- imputeData, 35
- imputePairs, 36
- map, 38
- mapClass, 39
- Mclust, 40
- mclust1Dplot, 42
- mclust2Dplot, 44
- mclustBIC, 46
- mclustDA, 48
- mclustDAtest, 51
- mclustDAtrain, 52
- mclustModel, 54
- mclustModelNames, 55
- mclustOptions, 57
- mclustVariance, 59
- me, 60

- meE, 62
- mstep, 64
- mstepE, 65
- mvn, 67
- mvnX, 69
- nVarParams, 71
- partconv, 72
- partuniq, 73
- plot.Mclust, 74
- plot.mclustBIC, 75
- plot.mclustDA, 77
- plot.mclustDAtrain, 78
- priorControl, 80
- randProj, 81
- sigma2decomp, 83
- sim, 85
- simE, 87
- summary.mclustBIC, 89
- summary.mclustDAtest, 90
- summary.mclustDAtrain, 91
- summary.mclustModel, 92
- surfacePlot, 93
- uncerPlot, 95
- unmap, 96

*Topic **datasets**

- chevron, 8
- cross, 13
- diabetes, 21
- wreath, 97
- .Mclust, 58
- .Mclust (*Defaults.Mclust*), 18
- adjustedRandIndex, 1
- bic, 3, 72
- bicEMtrain, 4, 14
- cdens, 5, 8, 21
- cdenseE, 6, 6
- cdenseEEE (*cdenseE*), 6

- cdensEEI (*cdensE*), 6
- cdensEEV (*cdensE*), 6
- cdensEII (*cdensE*), 6
- cdensEVI (*cdensE*), 6
- cdensV (*cdensE*), 6
- cdensVEI (*cdensE*), 6
- cdensVEV (*cdensE*), 6
- cdensVII (*cdensE*), 6
- cdensVVI (*cdensE*), 6
- cdensVVV, 6
- cdensVVV (*cdensE*), 6
- chevron, 8
- classError, 2, 9, 39, 50, 52, 91
- clPairs, 10, 13, 43, 46, 83
- coordProj, 11, 11, 43, 46, 75, 79, 83
- cross, 13
- cvlEMtrain, 4, 14

- decomp2sigma, 15, 84
- defaultPrior, 16, 81
- Defaults.Mclust, 18
- dens, 6, 8, 20
- diabetes, 21
- do.call, 3, 6, 8, 21, 23, 30, 86, 88

- em, 21, 24, 28, 30, 38, 61, 64, 96
- EMclust (*mclustBIC*), 46
- emControl, 23, 41, 48, 58, 65, 67
- emE, 23, 25
- emEEE (*emE*), 25
- emEEI (*emE*), 25
- emEEV (*emE*), 25
- emEII (*emE*), 25
- emEVI (*emE*), 25
- emV (*emE*), 25
- emVEI (*emE*), 25
- emVEV (*emE*), 25
- emVII (*emE*), 25
- emVVI (*emE*), 25
- emVVV, 23
- emVVV (*emE*), 25
- estep, 6, 23, 24, 27, 30, 38, 61, 64, 65, 67, 97
- estepE, 28, 28
- estepEEE (*estepE*), 28
- estepEEI (*estepE*), 28
- estepEEV (*estepE*), 28
- estepEII (*estepE*), 28
- estepEVI (*estepE*), 28
- estepV (*estepE*), 28

- estepVEI (*estepE*), 28
- estepVEV (*estepE*), 28
- estepVII (*estepE*), 28
- estepVVI (*estepE*), 28
- estepVVV, 28
- estepVVV (*estepE*), 28

- hc, 30, 33, 34, 48
- hcE, 30, 31, 32, 34
- hcEEE (*hcE*), 32
- hcEII (*hcE*), 32
- hclass, 31, 33, 33
- hcV (*hcE*), 32
- hcVII (*hcE*), 32
- hcVVV, 31
- hcVVV (*hcE*), 32
- hypvol, 34

- imputeData, 35, 37
- imputePairs, 36, 36

- map, 38, 97
- mapClass, 2, 9, 39, 39, 96
- Mclust, 19, 40, 56, 75, 86, 88
- mclust1Dplot, 42, 75, 79
- mclust2Dplot, 13, 43, 44, 75, 79, 83, 95
- mclustBIC, 3, 8, 17, 19, 24, 35, 41, 46, 53, 55, 56, 76, 81, 90, 96
- mclustDA, 48, 77
- mclustDAtest, 50, 51, 53, 91
- mclustDATrain, 50, 52, 52, 92
- mclustModel, 48, 54, 90, 93
- mclustModelNames, 6, 41, 48, 55, 61, 68
- mclustOptions, 6, 8, 11, 13, 19, 21, 23, 26, 28, 30, 41, 46, 48, 57, 61, 64, 65, 79, 83
- mclustVariance, 6, 28, 30, 59, 61
- me, 17, 23, 24, 26, 38, 48, 60, 64, 65, 67, 81, 96, 97
- meE, 61, 62
- meEEE (*meE*), 62
- meEEI (*meE*), 62
- meEEV (*meE*), 62
- meEII (*meE*), 62
- meEVI (*meE*), 62
- meV (*meE*), 62
- meVEI (*meE*), 62
- meVEV (*meE*), 62
- meVII (*meE*), 62

- meVVI (*meE*), 62
- meVVV, 61
- meVVV (*meE*), 62
- mstep, 8, 17, 23, 24, 26, 28, 30, 61, 64, 67, 81, 86
- mstepE, 65, 65, 70, 88
- mstepEEE (*mstepE*), 65
- mstepEEI (*mstepE*), 65
- mstepEEV (*mstepE*), 65
- mstepEII (*mstepE*), 65
- mstepEVI (*mstepE*), 65
- mstepV (*mstepE*), 65
- mstepVEI (*mstepE*), 65
- mstepVEV (*mstepE*), 65
- mstepVII (*mstepE*), 65
- mstepVVI (*mstepE*), 65
- mstepVVV, 65
- mstepVVV (*mstepE*), 65
- mvn, 67, 70
- mvnX, 68, 69
- mvnXII, 68
- mvnXII (*mvnX*), 69
- mvnXXI, 68
- mvnXXI (*mvnX*), 69
- mvnXXX, 68
- mvnXXX (*mvnX*), 69

- nVarParams, 3, 71

- pairs, 11, 37
- partconv, 72, 73
- partuniq, 72, 73
- plot.Mclust, 74
- plot.mclustBIC, 75
- plot.mclustDA, 50, 77
- plot.mclustDAtrain, 78
- print.Mclust (*Mclust*), 40
- print.mclustBIC (*mclustBIC*), 46
- print.mclustDA (*mclustDA*), 48
- print.mclustDAtrain (*mclustDAtrain*), 52
- print.summary.mclustBIC (*summary.mclustBIC*), 89
- printSummaryMclustBIC (*summary.mclustBIC*), 89
- printSummaryMclustBICn (*summary.mclustBIC*), 89
- priorControl, 17, 41, 48, 61, 67, 80

- randProj, 13, 75, 81

- sigma2decomp, 16, 83
- sim, 85, 88
- simE, 86, 87
- simEEE (*simE*), 87
- simEEI (*simE*), 87
- simEEV (*simE*), 87
- simEII (*simE*), 87
- simEVI (*simE*), 87
- simV (*simE*), 87
- simVEI (*simE*), 87
- simVEV (*simE*), 87
- simVII (*simE*), 87
- simVVI (*simE*), 87
- simVVV, 86
- simVVV (*simE*), 87
- summary.mclustBIC, 48, 89
- summary.mclustDAtest, 52, 90
- summary.mclustDAtrain, 53, 91
- summary.mclustModel, 92
- summaryMclustBIC (*summary.mclustBIC*), 89
- summaryMclustBICn (*summary.mclustBIC*), 89
- surfacePlot, 46, 75, 93

- table, 2, 9, 39

- uncerPlot, 95
- unmap, 38, 96

- wreath, 97