# Package 'RmixmodCombi'

February 19, 2015

**Type** Package

**Title** Combining Mixture Components for Clustering

**Version** 1.0

**Date** 2013-03-04

**Author** J.-P. Baudry and G. Celeux

**Maintainer** J.-P. Baudry <RmixmodCombi@gmail.com>

**Description**

The Rmixmod package provides model-based clustering by fitting a mixture model (e.g. Gaussian components for quantitative continuous data) to the data and identifying each cluster with one of its components. The number of components can be determined from the data, typically using the BIC criterion. In practice, however, individual clusters can be poorly fitted by Gaussian distributions, and in that case model-based clustering tends to represent one non-Gaussian cluster by a mixture of two or more Gaussian components. If the number of mixture components is interpreted as the number of clusters, this can lead to overestimation of the number of clusters. This is because BIC selects the number of mixture components needed to provide a good approximation to the density. This package, RmixmodCombi, according to \{}emph{Combining Mixture Components for Clustering} by J.P. Baudry, A.E. Raftery, G. Celeux, K. Lo, R. Gottardo, combines the components of the EM/BIC solution (provided by Rmixmod) hierarchically according to an entropy criterion. This yields a clustering for each number of clusters less than or equal to K. These clusterings can be compared on substantive grounds, and we also provide a way of selecting the number of clusters via a piecewise linear regression fit to the (possibly rescaled) entropy plot.

**License** GPL-3

**Depends** R(>= 3.0.2), Rmixmod(>= 2.0.1), Rcpp(>= 0.8.0), methods, graphics

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-07-08 19:17:05

## R topics documented:

---

`RmixmodCombi-package` *Combining Mixture Components for Clustering*

---

## Description

The Rmixmod package provides model-based clustering by fitting a mixture model (e.g. Gaussian components for quantitative continuous data) to the data and identifying each cluster with one of its components. The number of components can be determined from the data, typically using the BIC criterion. In practice, however, individual clusters can be poorly fitted by Gaussian distributions, and in that case model-based clustering tends to represent one non-Gaussian cluster by a mixture of two or more Gaussian components. This package, RmixmodCombi, following the article in the references, combines the components of the EM/BIC solution (provided by the Rmixmod package) hierarchically according to an entropy criterion. This yields a clustering for each number of clusters less than or equal to K. These clusterings can be compared on substantive grounds, and we also provide a way of selecting the number of clusters via a piecewise linear regression fit to the (possibly rescaled) entropy plot.

## Details

| | |
|---|---|
| Package: | RmixmodCombi |
| Type: | Package |
| Version: | 1.0 |
| Date: | 2013-09-20 |
| Depends: | Rmixmod, Rcpp |

See the Rmixmod package documentation for more details about fitting mixture models. See the cited article for more details about the combining methodology.

## Author(s)

J.-P. Baudry and G. Celeux Maintainer: J.-P. Baudry <RmixmodCombi@gmail.com>

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
##### Example of quantitative data #####

set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res # is of class MixmodCombi

res@mixmodOutput # is the initial EM/BIC solution (provided by mixmodCluster or by the user as a
# [\code{\linkS4class{MixmodCluster}}] object) from which the hierarchy is computed

res@hierarchy[[3]] # is the 3-cluster solution obtained by hierarchically combining the initial
# EM/BIC solution

## Not run:
plot(res) # This is a simulated example where the clusters should obviously not be identified to
# Gaussian components...

hist(res, nbCluster = 4)

## End(Not run)

##### Example of qualtitative data #####

set.seed(1)

data(car)
res <- mixmodCombi(car[1:300,], nbCluster = 1:10) # Only the 300 first observations for a
# quick example

res # is of class MixmodCombi

res@mixmodOutput # is the initial EM/BIC solution (provided by mixmodCluster or by the user as a
# [\code{\linkS4class{MixmodCluster}}] object) from which the hierarchy is computed

res@hierarchy[[res@ICLNbCluster]] # is the solution obtained by hierarchically combining the initial
# EM/BIC solution for the number of clusters selected with ICL

## Not run:
plot(res)

barplot(res)

## End(Not run)
```

---

Baudry_etal_2010_JCGS_examples

*Simulated Example Datasets From Baudry et al. (2010)*

---

## Description

Simulated Datasets used in Baudry et al. (2010) to illustrate the proposed mixture components combining method for clustering.

Please see the cited article for a detailed presentation of these datasets. The data frame with name exN.M is presented in Section N.M in the paper.

Test1D (not in the article) has been simulated from a Gaussian mixture distribution in R.

ex4.1 and ex4.2 have been simulated from a Gaussian mixture distribution in R^2.

ex4.3 has been simulated from a mixture of a uniform distribution on a square and a spherical Gaussian distribution in R^2.

ex4.4.1 has been simulated from a Gaussian mixture model in R^2

ex4.4.2 has been simulated from a mixture of two uniform distributions in R^3.

## Usage

```
data(Baudry_etal_2010_JCGS_examples)
```

## Format

ex4.1 is a data frame with 600 observations of 2 real variables.

ex4.2 is a data frame with 600 observations of 2 real variables.

ex4.3 is a data frame with 200 observations of 2 real variables.

ex4.4.1 is a data frame with 800 observations of 2 real variables.

ex4.4.2 is a data frame with 300 observations of 3 real variables.

Test1D is a data frame with 200 observations of 1 real variable.

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)

output <- mixmodCombi(ex4.4.2, nbCluster = 1:10,
models = mixmodGaussianModel(listModels = "Gaussian_pk_Lk_Ck"))

output # is of class MixmodCombi

## Not run:
plot(output) # plots the hierarchy of combined solutions, then some "entropy plots"
# which may help to select the number of classes
```

```
## End(Not run)
```

---

combMat *Combining Matrix*

---

## Description

Create a combining matrix

## Usage

```
combMat(K,l1,l2)
```

## Arguments

| | |
|---|---|
| K | The original number of classes: the matrix will define a combining from K to (K-1) classes. |
| l1 | Label of one of the two classes to be combined. |
| l2 | Label of the other class to be combined. |

## Value

If z is a vector (length *K*) whose *k*th entry is the probability that an observation belongs to the *k*th class in a *K*-class classification, then `combiM %*% z` is the vector (length *K-1*) whose *k*th entry is the probability that the observation belongs to the *k*th class in the *K-1*-class classification obtained by combining classes l1 and l2 in the initial classification.

## Author(s)

J.-P. Baudry

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## See Also

mixmodCombi

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res@hierarchy[[5]]@proba # each line of this matrix is the vector of the posterior probabilities of
# each class for an observation in the 5-cluster solution

t(combMat(5, 3, 4) %*% t(res@hierarchy[[5]]@proba) ) # each line of this matrix is the vector of
# the posterior probabilities of each class for an observation in the 4-cluster solution obtained by
# combining clusters 3 and 4 in the 5-cluster solution
```

---

GvHD *GvHD Dataset*

---

## Description

GvHD (Graft-versus-Host Disease) data of Brinkman et al. (2007). Two samples of this flow cytometry data, one from a patient with the GvHD, and the other from a control patient. The GvHD positive and control samples consist of 9083 and 6809 observations, respectively. Both samples include four biomarker variables, namely, CD4, CD8b, CD3, and CD8. The objective of the analysis is to identify CD3+ CD4+ CD8b+ cell sub-populations present in the GvHD positive sample.

A treatment of this data by combining mixtures is proposed in Baudry et al. (2010).

## Usage

```
data(GvHD)
```

## Format

GvHD.pos (positive patient) is a data frame with 9083 observations of the following 4 variables, which are biomarker measurements.

**CD4**

**CD8b**

**CD3**

**CD8**

GvHD.control (control patient) is a data frame with 6809 observations of the following 4 variables, which are biomarker measurements.

**CD4**

**CD8b**

**CD3**

**CD8**

## References

R. R. Brinkman, M. Gasparetto, S.-J. J. Lee, A. J. Ribickas, J. Perkins, W. Janssen, R. Smiley and C. Smith (2007). High-content flow cytometry and temporal data analysis for defining a cellular signature of Graft-versus-Host Disease. *Biology of Blood and Marrow Transplantation, 13: 691-700.*

K. Lo, R. R. Brinkman, R. Gottardo (2008). Automated gating of flow cytometry data via robust model-based clustering. *Cytometry A, 73: 321-332.*

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(GvHD)

dat <- GvHD.pos[1:500,] # only a few lines for a quick example

output <- mixmodCombi(dat, nbCluster = 1:10,
models = mixmodGaussianModel(listModels = "Gaussian_pk_Lk_Ck"))

output # is of class MixmodCombi

## Not run:
plot(output) # plots the hierarchy of combined solutions, then some "entropy plots" which may help
# to select the number of classes

## End(Not run)
```

---

`mixmodCombi` *Combining Mixture Components for Clustering*

---

## Description

Provides a hierarchy of combined clusterings from the EM/BIC mixture solution provided by Rmixmod to one class, following the methodology proposed in the article cited in the references.

## Usage

```
mixmodCombi(data = NULL, nbCluster = NULL, mixmodOutput = NULL,
criterion = c("BIC", "ICL"), ...)
```

## Arguments

| | |
|---|---|
| data | matrix or data frame containing quantitative or qualitative data. Rows correspond to observations and columns correspond to variables. |
| nbCluster | numeric listing the numbers of clusters to consider. |
| mixmodOutput | [MixmodCluster] object, as returned by the mixmodCluster function, containing the optimal mixture (according to BIC) associated to the data in data. Please see the Rmixmod documentation for the details of the components. Default value is NULL, in which case mixmodCluster is called. |
| criterion | as for the mixmodCluster function, list of characters defining the criterion to select the best model. The best model is the one with the lowest criterion value. Possible values: "BIC", "ICL", "NEC", c("BIC", "ICL", "NEC"). Unlike the mixmodCluster function, the default value is c("BIC", "ICL") and should only be modified with care (the plot and print functions may then wrongly refer to the "BIC" and "ICL" solutions). |
| ... | any optional argument that should be passed to the mixmodCluster function, for example the list of models to consider... Please see the mixmodCluster function documentation. |

## Details

mixmodCluster provides a mixture fitted to the data by maximum likelihood through the EM algorithm, for the model and number of components selected according to BIC. The corresponding components are hierarchically combined according to an entropy criterion, following the methodology described in the article cited in the references section. The combined clusterings with numbers of classes between the one selected by BIC and one are returned as a [MixmodCombi] object.

## Value

MixmodCombi object:

| | |
|---|---|
| mixmodOutput | [MixmodCluster] object. EM/BIC solution from which the hierarchy is computed. Either provided by the user or computed by a call to the mixmodCluster function |
| hierarchy | a list of MixmodCombiSol objects, each of which is the solution for the corresponding number of clusters obtained by hierarchically combining the EM/BIC solution according to the method proposed in the article in the references. Each one contains: the number of cluters, the partition of the data, the posterior probabilities of each class for each observation, the entropy value for the solution and a "combining matrix" combiM which enables to get the K-cluster solution from the (K+1)-cluster solution (please see the combMat function documentation about the combining matrices and how to use them). |
| ICLNbCluster | number of clusters selected by ICL, according to the mixmodOutput solution (if the criterion option has not been changed). |

## Note

Be careful: the hierarchy is computed from the solution in `mixmodOutput@bestResult`. This is notably the solution selected with the first criterion specified in the `criterion` option. By default, this is the BIC solution, as suggested in the paper. The `criterion` should then be changed only with care (the plot and print function may then wrongly refer to the "BIC" and "ICL" solutions).

## Author(s)

J.-P. Baudry and G. Celeux

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
##### Example of quantitative data #####

set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res # is of class MixmodCombi

res@mixmodOutput # is the initial EM/BIC solution (provided by mixmodCluster or by the user as a
# [\code{\linkS4class{MixmodCluster}}] object) from which the hierarchy is computed

res@hierarchy[[3]] # is the 3-cluster solution obtained by hierarchically combining the initial
# EM/BIC solution

## Not run:
plot(res)

hist(res, nbCluster = 4)

## End(Not run)

##### Example of qualtitative data #####

set.seed(1)

data(car)
res <- mixmodCombi(car[1:300,], nbCluster = 1:10) # Only the 300 first observations for a
# quick example

res # is of class MixmodCombi

res@mixmodOutput # is the initial EM/BIC solution (provided by mixmodCluster or by the user as a
```

```
# [\code{\linkS4class{MixmodCluster}}] object) from which the hierarchy is computed

res@hierarchy[[res@ICLNbCluster]] # is the solution obtained by hierarchically combining the initial
# EM/BIC solution for the number of clusters selected with ICL

## Not run: plot(res)

barplot(res)

## End(Not run)
```

---

MixmodCombi-class *[MixmodCombi] class*

---

## Description

Class of an output from the `mixmodCombi` function

## Details

**mixmodOutput** [MixmodCluster] object. EM/BIC solution from which the hierarchy is computed. Either provided by the user or computed by a call to the `mixmodCluster` function

**hierarchy** list of `MixmodCombiSol` objects, each of which is the solution for the corresponding number of clusters obtained by hierarchically combining the EM/BIC solution according to the method proposed in the article in the references. Each one contains: the number of clusters, the partition of the data, the posterior probabilities of each class for each observation, the entropy value for the solution and a "combining matrix" `combiM` which enables to get the K-cluster solution from the (K+1)-cluster solution (please see the `combMat` function documentation about the combining matrices and how to use them).

**ICLNbCluster** number of clusters selected by ICL, according to the mixmodOutput solution (if the `criterion` option has not been changed when calling the `mixmodCombi` function).

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)

res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res # is of class MixmodCombi

getSlots("MixmodCombi")
```

mixmodMap *Computes a hard clustering corresponding to a soft clustering*

## Description

Computes the hard clustering through the Maximum A Posteriori rule from the matrix of a posteriori probabilities (soft clustering).

## Usage

```
mixmodMap(tau, n = nrow(tau), K = ncol(tau))
```

## Arguments

| | |
|---|---|
| tau | matrix with posterior probabilities of each class for each observation (classes in columns, observations in rows). |
| n | number of observations. |
| K | number of classes. |

## Value

a matrix of same dimensions as *tau*. For each observation (row), only zeros except for the column corresponding to the cluster to which the observation is assigned, which value is one.

## Author(s)

J.-P. Baudry and G. Celeux

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res@hierarchy[[3]]@proba[1:10,] # Is the matrix of posterior probabilities of each of the combined
# classes in the 3-class solution, for the 10 first observations

mixmodMap(res@hierarchy[[3]]@proba[1:10,]) # Is the matrix of corresponding class assignments for
# the 10 first observations (available as a labels vector: res@hierarchy[[3]]@partition[1:10])
```

mixmodMap_M2V *Matrix of Class Assignments to Vector of Labels Conversion*

## Description

Converts a matrix of class assignments to a vector of labels.

## Usage

```
mixmodMap_M2V(M, n = nrow(M), K = ncol(M))
```

## Arguments

| | |
|---|---|
| M | matrix of class assignments. Rows correspond to observations and columns correspond to classes. Each row contains only zeros except for the column corresponding to the class to which the observation is assigned, which value is one. |
| n | number of observations. |
| K | number of classes. |

## Value

a vector of labels of length n.

## Author(s)

J.-P. Baudry and G. Celeux

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res@hierarchy[[3]]@proba[1:10,] # Is the matrix of posterior probabilities of each of the combined
# classes in the 3-class solution, for the 10 first observations

mixmodMap(res@hierarchy[[3]]@proba[1:10,]) # Is the matrix of corresponding class assignments for
# the 10 first observations

mixmodMap_M2V(mixmodMap(res@hierarchy[[3]]@proba[1:10,])) # Is the labels vector of the classes
# assigned to the 10 first observations
```

mixmodMap_V2M *Vector of Labels to Matrix of Class Assignments Conversion*

## Description

Converts a vector of labels to a matrix of class assignments.

## Usage

```
mixmodMap_V2M(z, n = length(z), K = max(z))
```

## Arguments

| | |
|---|---|
| z | vector of labels corresponding to the class assigned to each observation. |
| n | number of observations. |
| K | number of classes. |

## Value

matrix of class assignments. Rows correspond to observations and columns correspond to classes. Each row contains only zeros except for the column corresponding to the class to which the observation is assigned, which value is one.

## Author(s)

J.-P. Baudry and G. Celeux

## References

J.-P. Baudry, A. E. Raftery, G. Celeux, K. Lo and R. Gottardo (2010). Combining mixture components for clustering. *Journal of Computational and Graphical Statistics, 19(2):332-353.*

## Examples

```
set.seed(1)

data(Baudry_etal_2010_JCGS_examples)
res <- mixmodCombi(ex4.1, nbCluster = 1:8)

res@hierarchy[[3]]@partition[1:10] # Is the labels vector of the classes assigned to the 10 first
# observations

mixmodMap_V2M(res@hierarchy[[3]]@partition[1:10]) #  Is the corresponding matrix of class
# assignments for the 10 first observations
```

# Index