

# Package ‘deepgmm’

November 20, 2022

**Type** Package

**Title** Deep Gaussian Mixture Models

**Version** 0.2.1

**Date** 2022-11-21

**Author** Cinzia Viroli, Geoffrey J. McLachlan

**Maintainer** Suren Rathnayake <surenr@gmail.com>

**Description** Deep Gaussian mixture models as proposed by Viroli and McLachlan (2019) <[doi:10.1007/s11222-017-9793-z](https://doi.org/10.1007/s11222-017-9793-z)> provide a generalization of classical Gaussian mixtures to multiple layers. Each layer contains a set of latent variables that follow a mixture of Gaussian distributions. To avoid overparameterized solutions, dimension reduction is applied at each layer by way of factor models.

**URL** <https://github.com/suren-rathnayake/deepgmm>

**Imports** mvtnorm, corpcor, mclust

**License** GPL (>= 3)

**NeedsCompilation** no

**Config/testthat/edition** 3

**Repository** CRAN

**Date/Publication** 2022-11-20 21:30:02 UTC

## R topics documented:

deepgmm . . . . .	2
model_selection . . . . .	4
<b>Index</b>	<b>6</b>

---

 deepgmm

*Fits Deep Gaussian Mixture Models Using Stochastic EM algorithm.*


---

### Description

Fits a deep Gaussian mixture model to multivariate data.

### Usage

```
deepgmm(y, layers, k, r,
        it = 250, eps = 0.001, init = "kmeans", init_est = "factanal",
        seed = NULL, scale = TRUE)
```

### Arguments

<code>y</code>	A matrix or a data frame in which the rows correspond to observations and the columns to variables.
<code>layers</code>	The number of layers in the deep Gaussian mixture model. Limited to 1, 2 or 3.
<code>k</code>	A vector of integers of length <code>layers</code> containing the number of groups in the different layers.
<code>r</code>	A vector of integers of length <code>layers</code> containing the dimensions at the different layers. Dimension of the layers must be in decreasing size. See details.
<code>it</code>	Maximum number of EM iterations.
<code>eps</code>	The EM algorithm terminates if the relative increment of the log-likelihood falls below this value.
<code>init</code>	Procedure to obtain an initial partition of the observations. See Details.
<code>init_est</code>	Procedure for computing the initial parameter values for the given initial partition of the data. See Details.
<code>seed</code>	Integer value to be passed to the <code>set.seed</code> function at the beginning of the <code>deepgmm</code> function.
<code>scale</code>	If <code>scale = TRUE</code> , the columns of data, <code>y</code> , will be scaled to zero mean and unit variance.

### Details

The deep Gaussian mixture model is an hierarchical model organized in a multilayered architecture where, at each layer, the variables follow a mixture of Gaussian distributions. This set of nested mixtures of linear models provides a globally nonlinear model that can model the data in a very flexible way. In order to avoid overparameterized solutions, dimension reduction by factor models can be applied at each layer of the architecture, thus resulting in deep mixtures of factor analyzers.

The data `y` must be a matrix or a data frame containing numerical values, with no missing values. The rows must correspond to observations and the columns to variables.

Presently, the maximum number of layers `layers` implemented is 3.

The  $i$ th element of  $k$  contain number of groups in the  $i$ th layer. Thus the length  $k$  must equal to layers.

The parameter vector  $r$  contains the latent variable dimension of each layer. Variables at different layers have progressively decreasing dimension,  $r_1, r_2, \dots, r_h$ , where  $p > r_1 > r_2 > \dots > r_h \geq 1$ .

The EM algorithm used by `dgmm` requires initialization. The initialization is done by first partitioning the dataset, and then estimating the initial values for model parameters based on the partition. There are four options available in `dgmm` for the initial partitioning of the data; random partitioning (`init = "random"`), clustering using the  $k$ -means algorithm of "Hartigan-Wong" (`init = "kmeans"`), agglomerative hierarchical clustering (`init = "hclass"`), and Gaussian mixture model based clustering (`init = "mclust"`).

After the initial partitioning has been chosen, initial values of the parameters in the component analyzers need to be calculated. Curently only one option available. This default option, `init_est = "factanal"` provides initial estimates of the parameters based on factor analysis.

## Value

An object of class "dgmm" containing fitted values. It contains

<code>H</code>	A list in which the $i$ th element is the loading matrix for the $i$ th layer
<code>w</code>	A list containing mixing proportions for each layer. (i.e. the element <code>w[[i]][j]</code> contain the mixing proportion of the $j$ th component in the $i$ layer.)
<code>mu</code>	A list of matrices containing components means in the columns. (i.e. the element <code>mu[[i]][, j]</code> contain the component mean of the $j$ th component in the $i$ layer.)
<code>psi</code>	A list of arrays which contain covariance matrices for the random error components of each component (i.e. the element <code>psi[[i]][j, , ]</code> contain the error covariance matrix for the $j$ th component in the $i$ layer.)
<code>lik</code>	The log-likelihood after each EM iteration
<code>bic</code>	The Bayesian information criterion for the model fit
<code>aic</code>	The Akaike information criterion for the model fit
<code>clc</code>	The Classification likelihood information criterion for the model fit
<code>icl.bic</code>	The integrated classification criterion for the model fit
<code>s</code>	Clustering of the observations
<code>seed</code>	Value of the seed used

## Author(s)

Cinzia Viroli, Geoffrey J. McLachlan

## References

Viroli, C. and McLachlan, G.J. (2019). Deep Gaussian mixture models. *Statistics and Computing* 29, 43-51.

**Examples**

```

layers <- 2
k <- c(3, 4)
r <- c(3, 2)
it <- 50
eps <- 0.001
y <- scale(mtcars)

set.seed(1)
fit <- deepgmm(y = y, layers = layers, k = k, r = r,
               it = it, eps = eps)

fit

summary(fit)

```

---

model\_selection

*Function to compare different models*


---

**Description**

Compares different models and return the best one selected according to criterion (BIC or AIC).

**Usage**

```

model_selection(y, layers, g, seeds = sample(.Machine$integer.max, 10),
               it = 50, eps = 0.001, init = "kmeans",
               init_est = "factanal", criterion = "BIC")

```

**Arguments**

y	A matrix or a data frame in which rows correspond to observations and columns to variables.
layers	The number of layers in the deep Gaussian mixture model. Admitted values are 1, 2 or 3.
g	The number of clusters.
seeds	Integer vector containing seeds to try.
it	Maximum number of EM iterations.
eps	The EM algorithm terminates the relative increment of the log-likelihood falls below this value.
init	Initial partitioning of the observations to determine initial parameter values. See Details.
init_est	To determine how the initial parameter values are computed. See Details.
criterion	Model selection criterion, either "AIC" or "BIC".

**Details**

Compares different models and return the best one selected according to criterion (BIC or AIC). One can use different number of seeds.

**Value**

A list containing an object of class "dgmm" containing fitted values and list of BIC and AIC values.

**References**

Viroli, C. and McLachlan, G.J. (2019). Deep Gaussian mixture models. *Statistics and Computing* 29, 43-51.

**Examples**

```
y <- scale(mtcars)

sel <- model_selection(y, layers = 2, g = 3, seeds = c(1, 2, 12334),
                      it = 250, eps = 0.001, init = "kmeans", criterion = "BIC")

sel

summary(sel)
```

# Index

- \* **cluster**

- deepgmm, [2](#)

- model\_selection, [4](#)

- \* **models**

- deepgmm, [2](#)

- model\_selection, [4](#)

- \* **multivariate**

- deepgmm, [2](#)

- model\_selection, [4](#)

deepgmm, [2](#)

model\_selection, [4](#)