

# Package ‘EMMIXmcfa’

October 18, 2017

**Type** Package

**Title** Mixture of Factor Analyzers with Common Factor Loadings

**Version** 2.0.8

**Date** 2017-10-18

**Author** Suren Rathnayake, Geoff McLachlan, Jungsun Baek

**Maintainer** Geoff McLachlan <g.mclachlan@uq.edu.au>

**Description** Fits a mixture of factor analyzers with a common component matrix for the factor loadings before the transformation of the latent factors to be white noise. It is designed specifically for the task of displaying the observed data points in a lower ( $q$ -dimensional) space, where  $q$  is the number of factors adopted in the factor-analytic representation of the observed vector.

**License** GPL ( $\geq 2$ )

**Suggests** EMMIX, mvtnorm, GGally, ggplot2

**NeedsCompilation** no

## R topics documented:

EMMIXmcfa-package	1
ari	3
err	4
factor_scores	5
mcfa	6
plot_factors	9
rmix	10
<b>Index</b>	<b>12</b>

---

EMMIXmcfa-package *Mixture Models with Common Factor Loadings*

---

## Description

Fits a mixture of factor analyzers with a common component matrix for the factor loadings before the transformation of the latent factors to be white noise. It is designed specifically for the task of displaying the observed data points in a lower ( $q$ -dimensional) space, where  $q$  is the number of factors adopted in the factor-analytic representation of the observed vector.

It also provides a greater reduction in the number of parameters in the model. Component distributions can either be from the family of multivariate normals or from the family of multivariate  $t$ -distributions. Maximum likelihood estimators of model parameters are obtained using the Expectation-Maximization algorithm.

## Details

Package: EMMIXmcfapackage  
Type: Package  
Version: 2.0.8  
Date: 2017-10-18  
License: GPL

## Author(s)

Suren Rathnayake, Jangsun Baek, Geoffrey McLachlan

## References

Baek J, McLachlan GJ, and Flack LK (2010). Mixtures of factor analyzers with common factor loadings: applications to the clustering and visualisation of high-dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**, 2089–2097.

Baek J, and McLachlan GJ (2011). Mixtures of common  $t$ -factor analyzers for clustering high-dimensional microarray data. *Bioinformatics* **27**, 1269–1276.

McLachlan GJ, Baek J, and Rathnayake SI (2011). Mixtures of factor analyzers for the analysis of high-dimensional data. In *Mixture Estimation and Applications*, KL Mengersen, CP Robert, and DM Titterton (Eds). Hoboken, New Jersey: Wiley, pp. 171–191.

## See Also

mcfa

## Examples

```
## Not run:
set.seed(1)
Y <- scale(iris[, -5])
mcfa_model <- mcfa(Y, g = 3, q = 3)
mctfa_model <- mctfa(Y, g = 3, q = 3)

## End(Not run)
```

---

`ari`*Computes adjusted Rand Index*

---

**Description**

Computes adjusted Rand Index.

**Usage**

```
ari(cls, hat_cls)
```

**Arguments**

`cls`                Vector containing labels or classes.  
`hat_cls`            Vector of labels same length as `cls`.

**Details**

Measures the agreement between two set of partitions. The upper bound 1 implies perfect agreement. Expected value is zero if the partitions are random.

**Value**

Scaler specifying how closely two partitions agree.

**References**

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

**See Also**

[err](#)

**Examples**

```
## Not run:  
set.seed(1984)  
Y <- scale(iris[, -5])  
model <- mcfa(Y, g = 3, q = 3, nkmeans = 1, nrandom = 0)  
#  
ari(model$clust, iris[, 5])  
#  
err(model$clust, iris[, 5])  
  
## End(Not run)
```

---

`err`*Minimum Number of Mis-Allocations.*

---

### Description

Given two vectors each corresponding to a set of categories, this function finds the minimum number of mis-allocations by rotating the categories.

### Usage

```
err(cls, hat_cls)
```

### Arguments

`cls`                Vector of labels.  
`hat_cls`            Vector of labels same length as `cls`.

### Details

Rotates the categories for all possible permutations, and returns the minimum number of mis-allocations. The number of categories in each set of labels does not need to be the same. It make take several minutes to compute when the number of categories is large.

### Value

Integer specifying the minimum number of mis-allocations.

### Author(s)

Suren Rathnayake

### See Also

[ari](#)

### Examples

```
## Not run:
set.seed(1984)
Y <- scale(iris[, -5])
model <- mcfa(Y, g = 3, q = 3, nkmeans = 1, nrandom = 0)
#
ari(model$clust, iris[, 5])
#
err(model$clust, iris[, 5])

## End(Not run)
```

---

factor\_scores      *Computes Factor Scores.*

---

### Description

This function computes factor scores given a data set and a `EMMIXmcf` model.

### Usage

```
factor_scores(Y, model, tau = NULL, clust= NULL, ...)
```

### Arguments

model	Model of class "mcf", "mctf", "mfa", or "mtf".
Y	Data matrix with variables in columns in the same order as used in model estimation.
tau	Optional. Posterior probabilities of belonging to the components in the mixture model. If not provided, they will be estimated.
clust	Optional. Indicators of belonging to the components. If not provided, will be estimated using tau.
...	Not used.

### Details

Factor scores can be used in visualization of the data in the factor space.

### Value

U	Estimated conditional expected component scores of the unobservable factors given the data and the component membership. Size is $n \times q \times g$ , where $n$ is the number of sample, $q$ is the number of factors and $g$ is the number components.
Fmat	Means of the estimated conditional expected factors scores over estimated posterior distributions. Size $n \times q$ .
UC	Alternative estimate of Fmat where the posterior probabilities for each sample are replaced by component indicator vectors which contain one in the element corresponding to the highest posterior probability while others zero. Size $n \times q$ .

### Author(s)

Geoffrey McLachlan, Suren Rathnayake, Jungsun Baek

### References

McLachlan GJ, Baek J, and Rathnayake SI (2011). Mixtures of factor analyzers for the analysis of high-dimensional data. In *Mixture Estimation and Applications*, KL Mengersen, CP Robert, and DM Titterton (Eds). Hoboken, New Jersey: Wiley, pp. 171–191.

McLachlan GJ, and Peel D (2000). *Finite Mixture Models*. New York: Wiley.

**Examples**

```

# Fit a MCFA model to a subset
set.seed(1)
samp_size <- dim(iris)[1]
sel_subset <- sample(1 : samp_size, 75)
model <- mcfa(iris[sel_subset, -5], g=3, q=2, nkmeans=1, nrandom=0)

# plot the data points in the factor space
plot(model)

# Allocating new samples to the clusters
Y <- iris[-c(sel_subset), -5]
Y <- as.matrix(Y)
clust <- predict(model, Y)

factor_scores <- factor_scores(Y, model)
# Visualizing new data in factor space
plot_factors(factor_scores, type="Fmat", clust=clust)

```

mcfa

*Mixture of Common Factor Analyzers***Description**

Functions for fitting of Mixtures Common Factor Analyzers (MCFA) and Mixtures of Common  $t$ -Factor Analyzers (MCTFA). Maximum Likelihood estimates of the model parameters are obtained using the Expectation–Maximization algorithm.

MCFA adds the following restrictions to,

$$\Sigma_i = A \Omega_i A^T + D \quad (i = 1, \dots, g),$$

and

$$\mu_i = A \xi_i \quad (i = 1, \dots, g)$$

where  $A$  is a  $p \times q$  matrix,  $\xi_i$  is a  $q$ -dimensional vector,  $\Omega_i$  is a  $q \times q$  positive definite symmetric matrix, and  $D$  is a diagonal  $p \times p$  matrix.

With this representation, the component distribution of  $Y_j$  is modeled as

$$Y_j = A U_{ij} + e_{ij}$$

with prob.  $\pi_i (i = 1, \dots, g)$  for  $j = 1, \dots, n$ , where the (unobservable) factors  $U_{i1}, \dots, U_{in}$  are distributed independently  $N(\xi_i, \Omega_i)$ , independently of the  $e_{ij}$ , which are distributed independently  $N(0, D)$ , where  $D$  is a diagonal matrix,  $(i = 1, \dots, g)$ .

**Usage**

```

mcfa(Y, g, q, ...)
mctfa(Y, g, q, ...)
## Default S3 method:
mcfa(Y, g, q, itmax = 500, nkmeans = 20, nrandom = 20,
      tol = 1.e-5, init_clust = NULL, init_para = NULL,

```

```

    init_method = 'rand-A', conv_measure = 'diff',
    warn_messages = TRUE, ...)
## Default S3 method:
mctfa(Y, g, q, itmax = 500, nkmeans = 20, nrandom = 20,
      tol = 1.e-5, df_init = rep(30, g), df_update = TRUE,
      init_clust = NULL, init_para = NULL, init_method = 'rand-A',
      conv_measure = 'diff', warn_messages = TRUE, ...)
## S3 method for class 'emmixmcf'
print(x, ...)
## S3 method for class 'emmixmcf'
summary(object, ...)
## S3 method for class 'emmixmcf'
plot(x, ...)
## S3 method for class 'emmixmcf'
predict(object, Y, ...)

```

### Arguments

<code>Y</code>	A matrix or a data frame of which rows correspond to observations and columns to variables.
<code>x</code> , <code>object</code>	An object of class <code>mcf</code> or <code>mctfa</code> .
<code>g</code>	Number of components.
<code>q</code>	Number of factors.
<code>itmax</code>	Maximum number of EM iterations.
<code>nkmeans</code>	The number of times the k-means algorithm to be used in partition the data into <code>g</code> groups. These groupings are then used in initializing the parameters for the EM algorithm.
<code>nrandom</code>	The number of random <code>g</code> -group partitions for the data to be used initializing the EM algorithm.
<code>tol</code>	The EM algorithm terminates if the measure of convergence falls below this value.
<code>init_clust</code>	A vector or matrix consisting of partition of samples to be used in the EM algorithm. For matrix of partitions, columns must corresponds individual partitions of the data. Optional.
<code>init_para</code>	A list containing model parameters to be used as initial parameter estimates for the EM algorithm. Optional.
<code>init_method</code>	To determine how the initial parameter values are computed. See Details.
<code>conv_measure</code>	The default <code>'diff'</code> stops the EM iterations if $ l^{(k+1)} - l^{(k)}  < tol$ where $l^{(k)}$ is the log-likelihood at the $k$ th EM iteration. If <code>'ratio'</code> , then the convergence of the EM steps is measured using the $ l^{(k+1)} - l^{(k)} /l^{(k+1)}$ .
<code>df_init</code>	Initial values of the degree of freedom parameters for <code>mctfa</code> .
<code>df_update</code>	If <code>df_update = TRUE</code> (default), then the degree of freedom parameters values will be updated during the EM iterations. Otherwise, if <code>df_update = FALSE</code> , they will be fixed at the initial values specified in <code>df_init</code> .
<code>warn_messages</code>	If <code>warn_messages = TRUE</code> (default), the output would include error messages for instances, if any, where the model fitting function failed to provide estimates of parameters. Otherwise the messages will not be stored.
<code>...</code>	Not used.

## Details

With the default `init_method = "rand-A"`, initialization of the parameters is done by using the procedure in Baek et al. (2010) where initial values for elements of  $A$  are drawn from the  $N(0, 1)$  distribution. This method is appropriate when the columns of the data are on the same scale. The `init_method = "eigen-A"` takes the first  $q$  eigen vectors of  $Y$  as the loading matrix  $A$ .

## Value

Object of class `c("emmixmefa", "mefa")` or `c("emmixmefa", "mctfa")` containing the fitted model parameters is returned. Details of the components are as follows:

<code>g</code>	Number of mixture components.
<code>q</code>	Number of factors.
<code>pivec</code>	Mixing proportions of the components.
<code>A</code>	Loading matrix. Size $p \times q$ .
<code>xi</code>	Matrix containing factor means for components in columns. Size $q \times g$ .
<code>omega</code>	Array containing factor covariance matrices for components. Size $q \times q \times g$ .
<code>D</code>	Error covariance matrix. Size $p \times p$ .
<code>U</code>	Estimated conditional expected component scores of the unobservable factors given the data and the component membership. Size is Size $n \times q \times g$ .
<code>Fmat</code>	Means of the estimated conditional expected factors scores over estimated posterior distributions. Size $n \times q$ .
<code>UC</code>	Alternative estimate of <code>Fmat</code> where the posterior probabilities for each sample are replaced by component indicator vectors which contain one in the element corresponding to the highest posterior probability while others zero. Size $n \times q$ .
<code>clust</code>	Cluster labels.
<code>tau</code>	Posterior probabilities.
<code>logL</code>	Log-likelihood of the model.
<code>BIC</code>	Bayesian Information Criteria.
<code>warn_msg</code>	Description of error messages, if any.

## Author(s)

Suren Rathnayake, Jangsun Baek, Geoffrey McLachlan

## References

- Baek J, McLachlan GJ, and Flack LK (2010). Mixtures of factor analyzers with common factor loadings: applications to the clustering and visualisation of high-dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**, 2089–2097.
- Baek J, and McLachlan GJ (2011). Mixtures of common  $t$ -factor analyzers for clustering high-dimensional microarray data. *Bioinformatics* **27**, 1269–1276.
- McLachlan GJ, Baek J, and Rathnayake SI (2011). Mixtures of factor analyzers for the analysis of high-dimensional data. In *Mixture Estimation and Applications*, KL Mengersen, CP Robert, and DM Titterton (Eds). Hoboken, New Jersey: Wiley, pp. 171–191.



**Examples**

```
mcfa_fit <- mcfa(iris[, -5], g = 3, q = 3,
               itmax = 250, nkmeans = 5, nrandom = 5, tol = 1.e-5)

plot(mcfa_fit)
```

---

plot\_factors                      *Plot Function for Factor Scores.*

---

**Description**

Plot function for factor scores given factor score matrix of fitted model.

**Usage**

```
plot_factors(scores, type = "Fmat",
             clust=if (exists('clust', where = scores)) scores$clust else NULL,
             limx = NULL, limy = NULL)
```

**Arguments**

scores	A list containing factor scores specified by Fmat, UC or U, or a model of class mcfa, mctfa, mfa, or mtfa.
type	What type of factor scores are to be plotted. See Details.
clust	Indicators of belonging to components. If available, they will be portrayed in plots. If not provided, looks for clust in scores, and sets to NULL if still not available.
limx	Numeric vector. Values in limx will only be used in setting the x-axis range for 1-D and 2-D plots.
limy	Numeric vector. Values in limy will only be used in setting the y-axis range for 1-D and 2-D plots.

**Details**

The `type` should either be "U", "UC" or the default "Fmat". If `type = "U"`, then the estimated conditional expected component scores of the unobservable factors given the data and the component membership are plotted. If `type = "Fmat"`, then the means of the estimated conditional expected factors scores over estimated posterior distributions are plotted. If `type = "UC"`, then an alternative estimate of "Fmat", where the posterior probabilities are replaced by component indicator vector, is plotted.

**Author(s)**

Geoffrey McLachlan, Suren Rathnayake, Jungsun Baek

**References**

McLachlan GJ, and Peel D (2000). *Finite Mixture Models*. New York: Wiley.

McLachlan GJ, Baek J, and Rathnayake SI (2011). Mixtures of factor analyzers for the analysis of high-dimensional data. In *Mixture Estimation and Applications*, KL Mengersen, CP Robert, and DM Titterton (Eds). Hoboken, New Jersey: Wiley, pp. 171–191.

## Examples

```
# Visualizing data used in model estimation
set.seed(1)
inds <- dim(iris)[1]
indSample <- sample(1 : inds, 75)
model <- mcfa (iris[indSample, -5], g = 3, q = 2, nkmeans = 1, nrandom = 0)
err (model$clust, iris[indSample, 5])

#same as plot_factors(model, tyep = "Fmat", clust = model$clust)
plot (model)

#can provide alternative groupings of samples via plot_factors
plot_factors (model, clust = iris[indSample, 5])

#same as plot_factors(model, tyep = "UC")
plot (model, type = "UC")

Y <- iris[-c(indSample), -5]
Y <- as.matrix(Y)
clust <- predict(model, Y)
err(clust, iris[-c(indSample), 5])

fac_scores <- factor_scores(Y, model)
plot_factors (fac_scores, type="Fmat", clust = clust)
plot_factors (fac_scores, type="Fmat", clust = iris[-c(indSample), 5])
```

---

 rmix

*Random Deviates from EMMIXmcfa Models*


---

## Description

Random number generator for EMMIXmcfa models.

## Usage

```
rmix(n, model, ...)
```

## Arguments

model	Model of class mcfa, mctfa, mfa, or mtfa.
n	Number of sample to generate.
...	Not used.

## Details

This function uses the `rdemmix2` function in the **EMMIX** package to generate samples from the mixture components.

Algorithm works by first drawing a component based on the mixture proportion in the model, and then drawing a sample from the component distribution.

**Value**

dat                    Matrix with samples drawn in rows.

**Author(s)**

Geoffrey McLachlan, Suren Rathnayake

**References**

[https://people.smp.uq.edu.au/GeoffMcLachlan/mix\\_soft/EMMIX\\_R/](https://people.smp.uq.edu.au/GeoffMcLachlan/mix_soft/EMMIX_R/)

McLachlan GJ, and Peel D (2000). *Finite Mixture Models*. New York: Wiley.

McLachlan GJ, Baek J, and Rathnayake SI (2011). Mixtures of factor analyzers for the analysis of high-dimensional data. In *Mixture Estimation and Applications*, KL Mengersen, CP Robert, and DM Titterton (Eds). Hoboken, New Jersey: Wiley, pp. 171–191.

**Examples**

```
## Not run:
set.seed(1)
model <- mcfa(iris[, -5], g=3, q=2, nkmeans=1, nrandom=1)
dat <- rmix(n = 10, model = model)

## End(Not run)
```

# Index

**\*Topic clustering**  
EMMIXmcfa-package, 1

**\*Topic cluster**  
ari, 3  
err, 4  
factor\_scores, 5  
mcfa, 6  
plot\_factors, 9  
rmix, 10

**\*Topic models**  
factor\_scores, 5  
mcfa, 6  
plot\_factors, 9  
rmix, 10

**\*Topic model**  
EMMIXmcfa-package, 1

**\*Topic multivariate**  
EMMIXmcfa-package, 1  
factor\_scores, 5  
mcfa, 6  
plot\_factors, 9  
rmix, 10

**\*Topic package**  
EMMIXmcfa-package, 1

ari, 3, 4

EMMIXmcfa (*EMMIXmcfa-package*), 1  
emmixmcfa (*EMMIXmcfa-package*), 1  
EMMIXmcfa-package, 1  
emmixmcfa-package  
(*EMMIXmcfa-package*), 1  
err, 3, 4

factor\_scores, 5

mcfa, 6  
mctfa (*mcfa*), 6

plot.emmixmcfa (*mcfa*), 6  
plot\_factors, 9  
predict.emmixmcfa (*mcfa*), 6  
print.emmixmcfa (*mcfa*), 6

rmix, 10

summary.emmixmcfa (*mcfa*), 6