Package ‘EMMIXmfa’

May 7, 2015

Type Package
Title Mixture models with component-wise factor analyzers.
Version 1.2.6
Date 2015-05-04
Author Suren Rathnayake, Geoff McLachlan, Jungsun Baek
Maintainer Geoff McLachlan <g.mclachlan@uq.edu.au>
Description We provide functions to fit finite mixture of multivariate normal or t distributions to data where components adopt factor analytic structures. Maximum likelihood estimators of model parameters are obtained via Expectation-Maximization.
License GPL (>= 2)

R topics documented:

EMMIXmfa-package .......................................................... 1
ari ................................................................. 3
err ................................................................. 3
getscores .............................................................. 4
mcfa ................................................................. 6
mfa ................................................................. 8
plotscores ............................................................ 10

Index 12

EMMIXmfa-package Mixture models with component-wise factor analyzers.

Description

Fits finite mixture models that adopt component-wise factor analyzers to multivariate data. Component distributions can either be from the family multivariate normal or multivariate t-distributions. Maximum likelihood estimators of model parameters are obtained using Expectation-Maximization algorithm.
Details

Package: EMMIXmfa
Type: Package
Version: 1.2.5
Date: 2015-05-04
License: GPL

Author(s)

Suren Rathnayake, Jangsun Baek, Geoffrey McLachlan

References


See Also

mcfa, mfa

Examples

```r
set.seed(1)
Y <- scale(iris[, -5])
mcfa_model <- mcfa(Y, q=3, q=3)
mctfa_model <- mctfa(Y, q=3, q=3)
mfa_model <- mfa(Y, q=3, q=3)
mctfa_model <- mctfa(Y, q=3, q=3)
```
ari

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 in cls.

Details

Measures the agreement between two set of partitions.

Value

ARI Scaler specifying how closely two set of partitions agree. The upper bound 1 implies perfect agreement.

References


See Also

err

Examples

cls1 <- floor(runif(100, 1, 3))
cls2 <- floor(runif(100, 1, 3))
ari(cls1, cls2)

err

Description

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

Usage

err(cls, hat.cls)
**Argument**

- `cls` vector of labels.
- `hat.cls` vector of labels same length as in `cls`.

**Details**

Rotates the categories for all possible permutations, and returns the minimum number of misallocations. The number of categories in each set of labels does not need to be the same. It may take several minutes to compute for large number of categories.

**Value**

- `min.err` Integer specifying the minimum number of misallocations.

**Author(s)**

Suren Rathnayake

**See Also**

- `ari`

**Examples**

```r
c1 <- floor(runif(100, 1, 3))
c2 <- floor(runif(100, 1, 3))
err(c1, c2)
```

---

**Description**

This function computes factor scores given a data set and an EMMIXmfa model.

**Usage**

```r
getscores(Y, model, tau = NULL, clust= NULL, ...)
```

**Arguments**

- `model` Model of class "mcfa", "mctfa", "mfa", or "mtfa".
- `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 $n$ by $q$ by $g$, 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 $n$ by $q$.

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 $n$ by $q$.

Author(s)

Geoffrey McLachlan, Suren Rathnayake, Jungsun Baek

References


Examples

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

# Visualizing new data
Y <- iris[-c(indSample), -5]
Y <- as.matrix(Y)
clust <- predict(model, Y)
err(clust, iris[-c(indSample), 5])
facScores <- getscores(Y, model)
plotscores(facScores, type="Fmat", clust=clust)
```
**mcfa**

### Mixture of Common Factor Analyzers

**Description**

Fits a mixture of common factor analyzers model to a data set.

**Usage**

\[
\text{mcfa}(Y, g, q, \ldots)
\]

\[
\text{mctfa}(Y, g, q, \ldots)
\]

---

**Arguments**

- **Y** data.frame or a matrix of data matrix (samples in rows)
- **x, object** An object of class "mcfa", an object returned from `mcfa` function.
- **g** number of components
- **q** number of factors
- **itmax** maximum number of EM iterations
- **nkmeans** number of k-means starts
- **nrandom** number of random starts
- **tol** EM steps terminates if the measure of convergence falls below this value
- **initClust** Option to provide initial clustering of the samples
- **initPara** A list containing model parameters to be used as initial parameter estimates for the EM algorithm. Optional.
- **initMethod** determine how the initial parameter values are computed. See Details.
- **convMeas** If 'ratio', then the convergence of the EM steps are measured using the |(L^(k+1) - L^(k))/L^(k+1)|. The default 'diff' stops the EM iterations if |L^(k+1) - L^(k)| < tol
- **df.init** Initial values of the degree of freedom parameters for `mtfa`
df.update  If \( \text{df.update} = \text{TRUE} \) (default), then the DOF values will be updated during the EM iterations. Otherwise, if \( \text{df.update} = \text{FALSE} \), they will be fixed at the initial values specified in \( \text{df.init} \).

terrMsg logical. If \text{TRUE}, returns error messages.

Details

The default is 'eigenA' which initialize the columns of the loading vector \( A \) as the first \( q \) eigen vectors of the \( Y^T Y \). Use \text{initMethod} = "randA" for the procedure used by Baek et al. (2010).

Value

- \text{pivec} component probabilities
- \text{A} factor loading matrix
- \text{xi} matrix of factor mean vectors
- \text{omega} array of factor covariance matrices
- \text{D} matrix of error covariance matrix
- \text{U} array of estimated \( u_{ij} \) vectors for all components
- \text{UC} matrix of estimated \( u_{ij} \) vectors according to cluster labels
- \text{Fmat} matrix of estimated factor score vectors
- \text{clust} cluster label determined by MCFA
- \text{tau} posterior probabilities
- \text{logL} log-likelihood
- \text{BIC} Bayesian Information Criteria
- \text{ERRMSG} Error Message

References


Examples

```r
require(MASS)
mu1 <- c(0,0,0)
sig1 <- rbind(c(4,1.8,0.8), c(-1.8, 2, 0.9), c(-1, 0.9,2))
mu2 <- c(2,2,6)
sig2 <- rbind(c(4,1.8,0.8), c(1.8, 2, 0.5), c(0.8, 0.5, 2))
Y1 <- mvrnorm(100, mu1, sig1)
Y2 <- mvrnorm(100, mu2, sig2)
Y <-cbind(Y1,Y2)
mcfa.fit<-mcfa(Y, g=2, q=2, itmax=250, nkmeans=5, nrandom=5, tol=1.e-3)
plot(mcfa.fit)
```
Mixtures of Factor Analyzers and Mixtures of t-Factor Analyzers.

Description

Functions for fitting Mixtures of Factor Analyzers (MFA) models and Mixtures of t-Factor Analyzers (MtFA) model to data. Maximum Likelihood estimates of the model parameters are obtained using the Alternating Expectation Conditional Maximization (AECM) algorithm.

Usage

mfa(Y, g, q, ...)  
mtfa(Y, g, q, ...)  

## Default S3 method:  
mfa(Y, g, q, itmax=50, nkmeans=20, nrandom=20, 
tol=1.e-5, sigmaType = 'common', Dtype='common', initClust=NULL, 
initPara=NULL, convMeas='diff', errorMsg=TRUE, ...)  

## Default S3 method:  
mtfa(Y, g, q, itmax=50, nkmeans=20, nrandom=20, 
tol=1.e-5, df.init=rep(30, g), df.update=FALSE, sigmaType = 'common', 
Dtype='common', initClust=NULL, initPara=NULL, convMeas='diff', 
errorMsg=TRUE, ...)

Arguments

Y  
A matrix or a data frame of which rows correspond to observations and columns to variables.

g  
Number of components.

q  
Number of factors.

itmax  
Maximum number of EM iterations.

nkmeans  
The k-means algorithm is used nkmeans-times to partition the data into g groups. Each of the partition is used in estimating initial values for the EM algorithm.

nrandom  
The number of random g-group partitions to be used in obtaining initial parameter estimates for the EM algorithm.

tol  
Threshold value used to stop EM iterations based on criteria specified in convMeas.

sigmaType  
This allows to specify whether the covariance matrices (for mfa) or the scale matrices (for mtfa) are constraint to be the same (default) for each component or not. Default is sigmaType = "common", otherwise use sigmaType = "unique".

Dtype  
To specify whether the diagonal error covariance matrix is common to all the components or not. If sigmaType = "unique", then Dtype could either be "common" (the default) to each component, or "unique". If the sigmaType = "common", then Dtype must also be set to "common".

initClust  
A vector or matrix consisting of partition of samples to be used in the EM algorithm. For a matrix of partitions, rows corresponds individual partitions.

initPara  
A list containing model parameters to be used as initial parameter estimates for the EM algorithm. Optional.
Method to identify convergence of parameters during EM iterations. The default "diff" stops the EM iterations if the difference between the current and previous log-likelihood values falls below \( tol \) value. If \( \text{convMeas} = \text{"ratio"} \) is used, then the relative difference in log-likelihood values are considered.

**df.init** Initial values of the degree of freedom parameters for \( \text{mfa} \).

**df.update** If \( \text{df.update} = \text{TRUE} \) (default), then the DOF values will be updated during the EM iterations. Otherwise, if \( \text{df.update} = \text{FALSE} \), they will be fixed at the initial values specified in \( \text{df.init} \).

**errorMsg** This aims to provide a brief explanation for the cases where the function failed to provide estimates of parameters. Default is \( \text{errorMsg} = \text{TRUE} \), otherwise the messages will not be stored.

**Details**
Cluster a given data set using Mixtures of Factor Analyzers or approach or using Mixtures of \( t \)-Factor Analyzers.

**Value**
Object of class \( \text{c("emmixmfa", "mfa")} \) or \( \text{c("emmixmfa", "mtfa")} \) containing the fitted model parameters is returned. Details of the components are as fellows:

- \( g \) Number of mixture components.
- \( q \) Number of factors.
- \( pivec \) Vector containing mixing proportions.
- \( \mu \) Matrix containing estimates of component means for each mixture components. Size \( p \) by \( g \).
- \( B \) Array containing component dependent loading matrices. Size \( p \) by \( q \) by \( g \).
- \( D \) Estimates of error covariance matrices. If \( \text{Dtype} = \text{"common"} \) was used then \( D \) is a \( p \) by \( p \) matrix common to all components, if \( \text{Dtype} = \text{"unique"} \), then \( D \) is a \( p \) by \( p \) by \( g \) array.
- \( v \) degrees of freedom for each component.
- \( \logL \) Log-likelihood.
- \( \text{BIC} \) Bayesian Information Criterion.
- \( \tau \) Matrix of posterior probabilities for the data used based on the fitted values. Matrix of size \( n \) by \( g \).
- \( \text{clust} \) Vector of integers 1 to \( g \) indicating cluster allocations of the observations.
- \( \text{U} \) Estimated factor scores. Matrix of size \( n \) by \( q \) by \( g \).
- \( \text{UC} \) Estimated factor scores of the observations corresponding to clusters they were assigned. Matrix of size \( n \) by \( q \).
- \( \text{Fmat} \) Estimated conditional means of the factors over the estimated posterior distribution. Matrix of size \( n \) by \( q \).
- \( \text{ERRMSG} \) Error messages.
- \( \text{Dtype} \) Whether common or unique error covariance is used, as specified in model fitting.
- \( \text{df.update} \) Whether DOF (\( v \)) was fixed or estimated, as specified in model fitting.
Author(s)

Suren Rathnayake, Geoffrey McLachlan

References


See Also

mcfa

Examples

```r
model <- mfa(iris[, -5], g=3, q=2, itmax=200, nkmeans=1, nrandom=5)
plot(model)
summary(model)
```

Description

Plots factors scores.

Usage

```r
plotscores(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". These stands for, "U" the estimated conditional expected component scores of the unobservable factors given the data and the component membership (size is \( n \) by \( q \) by \( g \), where \( n \) is the number of sample, \( q \) number of factors and \( g \) number of components), "Fmat" is the means of the estimated conditional expected factors scores over estimated posterior distributions (size \( n \) by \( q \)), and "UC" is an alternative estimate of "Fmat" where the posterior probabilities are replaced by component indicator vector which contain one in the element corresponding to the highest posterior probability, while others zero.

Author(s)

Geoffrey McLachlan, Suren Rathnayake, Jungsun Baek

References


Examples

```r
# Visualizing data used in model estimation
set.seed(1)
indAll <- dim(iris)[1]
indSample <- sample(1:indAll, 75)
model <- mcfa(iris[indSample, -5], g=3, q=2, nkmeans=1, nrandom=0)
err(model$clust, iris[indSample, 5])
#same as plotscores(model, tyep="Fmat", clust=model$clust)
plot(model)
#can provide alternative groupings of samples via plotscores
plotscores(model, clust = iris[indSample, 5])
#same as plotscores(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])

facScores <- getscores(Y, model)
plotscores(facScores, type="Fmat", clust=clust)
plotscores(facScores, type="Fmat", clust=iris[-c(indSample), 5])
```
Index

*Topic clustering
  EMMIXmfa-package, 1

*Topic cluster
  ari, 3
  err, 3
  getscores, 4
  mcfa, 6
  mfa, 8
  plotscores, 10

*Topic models
  getscores, 4
  mcfa, 6
  mfa, 8
  plotscores, 10

*Topic model
  EMMIXmfa-package, 1

*Topic multivariate
  EMMIXmfa-package, 1
  getscores, 4
  mcfa, 6
  mfa, 8
  plotscores, 10

*Topic package
  EMMIXmfa-package, 1

ari, 3, 4

EMMIXmfa (EMMIXmfa-package), 1
emmixmfa (EMMIXmfa-package), 1
EMMIXmfa-package, 1
emmixmfa-package
  (EMMIXmfa-package), 1
err, 3, 3

getscores, 4

mcfa, 6, 10
mctfa (mcfa), 6
mfa, 8
mtfa (mfa), 8

plot.emmixmfa (mcfa), 6
plot.mfa (mfa), 8
plot.mtfa (mfa), 8
plotscores, 10

predict.emmixmfa (mcfa), 6
print.emmixmfa (mcfa), 6
print.mfa (mfa), 8
print.mtfa (mfa), 8
summary.emmixmfa (mcfa), 6
summary.mfa (mfa), 8
summary.mtfa (mfa), 8