Package ‘FisherEM’

July 22, 2020

Type Package

Title The FisherEM Algorithm to Simultaneously Cluster and Visualize High-Dimensional Data

Version 1.5.2

Date 2020-07-21

Author Charles Bouveyron, Camille Brunet & Nicolas Jouvin.

Maintainer Charles Bouveyron <charles.bouveyron@gmail.com>

Depends MASS, parallel, elasticnet

Description The FisherEM algorithm, proposed by Bouveyron & Brunet (2012) <doi:10.1007/s11222-011-9249-9>, is an efficient method for the clustering of high-dimensional data. FisherEM models and clusters the data in a discriminative and low-dimensional latent subspace. It also provides a low-dimensional representation of the clustered data. A sparse version of Fisher-EM algorithm is also provided.

License GPL-2

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2020-07-22 08:42:07 UTC

R topics documented:

FisherEM-package .................................................. 2
fem ................................................................. 2
fem.ari ............................................................. 5
plot.fem .......................................................... 6
print.fem ......................................................... 7
sfem ............................................................... 7

Index 10
Description

The FisherEM algorithm, proposed by Bouveyron & Brunet (201) <doi:10.1007/s11222-011-9249-9>, is an efficient method for the clustering of high-dimensional data. FisherEM models and clusters the data in a discriminative and low-dimensional latent subspace. It also provides a low-dimensional representation of the clustered data. A sparse version of Fisher-EM algorithm is also provided.

Details

Package: FisherEM
Type: Package
Version: 1.2
Date: 2012-07-09
License: GPL-2
LazyLoad: yes

Author(s)

Charles Bouveyron, Camille Brunet & Nicolas Jouvin.

Maintainer: Charles Bouveyron <charles.bouveyron@gmail.com>

References


**Description**

The Fisher-EM algorithm is a subspace clustering method for high-dimensional data. It is based on the Gaussian Mixture Model and on the idea that the data lives in a common and low dimensional subspace. An EM-like algorithm estimates both the discriminative subspace and the parameters of the mixture model.

**Usage**

```r
fem(Y,K=2:6,model='AkjBk',method='svd',crit='icl',maxit=50,eps=1e-4,init='kmeans',
nstart=5,Tinit=c(),kernel='',disp=FALSE,mc.cores=(detectCores()-1),
subset=NULL)
```

**Arguments**

- **Y**
  - The data matrix. Categorical variables and missing values are not allowed.

- **K**
  - An integer vector specifying the numbers of mixture components (clusters) among which the model selection criterion will choose the most appropriate number of groups. Default is 2:6.

- **model**
  - A vector of discriminative latent mixture (DLM) models to fit. There are 12 different models: "DkBk", "DkB", "DB", "AkjBk", "AkjB", "AkBk", "AkBk", "AjBk", "AjB", "ABk", "AB". The option "all" executes the Fisher-EM algorithm on the 12 DLM models and select the best model according to the maximum value obtained by model selection criterion.

- **method**
  - The method used for the fitting of the projection matrix associated to the discriminative subspace. Three methods are available: 'gs' (Gram-Schmidt, the original proposition), 'svd' (based on SVD, faster) and 'reg' (the Fisher criterion is rewritten as a regression problem). The 'svd' method is the default method since it is the fastest one on most data sets.

- **crit**
  - The model selection criterion to use for selecting the most appropriate model for the data. There are 3 possibilities: "bic", "aic" or "icl". Default is "icl".

- **maxit**
  - The maximum number of iterations before the stop of the Fisher-EM algorithm.

- **eps**
  - The threshold value for the likelihood differences to stop the Fisher-EM algorithm.

- **init**
  - The initialization method for the Fisher-EM algorithm. There are 4 options: "random" for a randomized initialization, "kmeans" for an initialization by the kmeans algorithm, "hclust" for hierarchical clustering initialization or "user" for a specific initialization through the parameter "Tinit". Default is "kmeans". Notice that for "kmeans" and "random", several initializations are asked and the initialization associated with the highest likelihood is kept (see "nstart").

- **nstart**
  - The number of restart if the initialization is "kmeans" or "random". In such a case, the initialization associated with the highest likelihood is kept.

- **Tinit**
  - A n x K matrix which contains posterior probabilities for initializing the algorithm (each line corresponds to an individual).

- **kernel**
  - It enables to deal with the n < p problem. By default, no kernel (" ") is used. But the user has the choice between 3 options for the kernel: "linear", "sigmoid" or "rbf".
disp  If true, some messages are printed during the clustering. Default is false.
mccores  The number of CPUs to use to fit in parallel the different models (only for non-Windows platforms). Default is the number of available cores minus 1.
subset  A positive integer defining the size of the subsample, default is NULL. In case of large data sets, it might be useful to fit a FisherEM model on a subsample of the data, and then use this model to predict cluster assignments for the whole data set. Notice that in, such a case, likelihood values and model selection criteria are computed for the subsample and not the whole data set.

Value

A list is returned:

K  The number of groups.
cls  the group membership of each individual estimated by the Fisher-EM algorithm.
P  the posterior probabilities of each individual for each group.
U  The loading matrix which determines the orientation of the discriminative subspace.
mean  The estimated mean in the subspace.
my  The estimated mean in the observation space.
prop  The estimated mixture proportion.
D  The covariance matrices in the subspace.
aic  The value of the Akaike information criterion.
bic  The value of the Bayesian information criterion.
icl  The value of the integrated completed likelihood criterion.
loglik  The log-likelihood values computed at each iteration of the FEM algorithm.
ll  the log-likelihood value obtained at the last iteration of the FEM algorithm.
method  The method used.
call  The call of the function.
plot  Some information to pass to the plot.fem function.
crit  The model selection criterion used.

Author(s)

Charles Bouveyron, Camille Brunet & Nicolas Jouvin.

References


fem.ari

Adjusted Rand index

Description

The function computes the adjusted Rand index (ARI) which allows to compare two clustering partitions.

Usage

fem.ari(x, y)

Arguments

x  
A 'fem' object containing the first partition to compare.

y  
The second partition to compare (as vector).

Value

ari  
The value of the ARI.

See Also

fem, sfem, plot.fem, summary.fem
Examples

data(iris)
res = fem(iris[, -5], K=3, model='DkBk', method='reg')
res
plot(res)
fem.ari(res, as.numeric(iris[, 5]))

Description

This function plots different information about 'fem' objects such as model selection, log-likelihood evolution and visualization of the clustered data into the discriminative subspace fitted by the Fisher-EM algorithm.

Usage

## S3 method for class 'fem'
plot(x, frame=0, crit=c(), ...)

Arguments

x The fem object.
frame 0: all plots; 1: selection of the number of groups; 2: log-likelihood; projection of the data into the discriminative subspace.
crit The model selection criterion to display. Default is the criterion used in the 'fem' function ('icl' by default).
... Additional options to pass to the plot function.

See Also

fem, sfem, fem.ari, summary.fem

Examples

data(iris)
res = fem(iris[, -5], K=3, model='DkBk', method='reg')
res
plot(res)
fem.ari(res, as.numeric(iris[, 5]))
print.fem

The print function for 'fem' objects.

Description

This function summarizes 'fem' objects. It in particular indicates which DLM model has been chosen and displays the loading matrix 'U' if the original dimension is smaller than 10.

Usage

```r
## S3 method for class 'fem'
print(x,...)
```

Arguments

- `x`: The fem object.
- `...`: Additional options to pass to the summary function.

See Also

fem, sfem, fem.ari, plot.fem

Examples

```r
data(iris)
res = fem(iris[, -5], K=3, model='DkBk', method='reg')
res
plot(res)
fem.ari(res, as.numeric(iris[, 5]))
```

sfem

The sparse Fisher-EM algorithm

Description

The sparse Fisher-EM algorithm is a sparse version of the Fisher-EM algorithm. The sparsity is introduced within the F step which estimates the discriminative subspace. The sparsity on U is obtained by adding a l1 penalty to the optimization problem of the F step.

Usage

```r
sfem(Y, K=2:6, obj=NULL, model='AkjBk', method='reg', crit='icl', maxit=50, eps=1e-6, init='kmeans', nstart=5, Tinit=c(), kernel='', disp=FALSE, l1=0.1, l2=0, nbit=2)
```
Arguments

Y
  The data matrix. Categorical variables and missing values are not allowed.

K
  An integer vector specifying the numbers of mixture components (clusters) among
  which the model selection criterion will choose the most appropriate number of
  groups. Default is 2:6.

obj
  An object of class 'fem' previously learned with the 'fem' function which will
  be used as initialization of the sparse FisherEM algorithm.

model
  A vector of discriminative latent mixture (DLM) models to fit. There are 12
  different models: "DkBk", "DkB", "DBk", "DB", "AkjBk", "AkjB", "AkBk", "AkBk",
  "AjBk", "AjB", "ABk", "AB". The option "all" executes the Fisher-EM algorithm on the 12 DLM models and select the best model according to
  the maximum value obtained by model selection criterion.

method
  The method use for the fitting of the projection matrix associated to the discrimi-
  native subspace. Three methods are available: 'svd', 'reg' and 'gs'. The 'reg'
  method is the default.

crit
  The model selection criterion to use for selecting the most appropriate model for
  the data. There are 3 possibilities: "bic", "aic" or "icl". Default is "icl".

maxit
  The maximum number of iterations before the stop of the Fisher-EM algorithm.

eps
  The threshold value for the likelihood differences to stop the Fisher-EM algo-
  rithm.

init
  The initialization method for the Fisher-EM algorithm. There are 4 options:
  "random" for a randomized initialization, "kmeans" for an initialization by the
  kmeans algorithm, "hclust" for hierarchical clustering initialization or "user"
  for a specific initialization through the parameter "Tinit". Default is "kmeans".
  Notice that for "kmeans" and "random", several initializations are asked and the
  initialization associated with the highest likelihood is kept (see "nstart").

nstart
  The number of restart if the initialization is "kmeans" or "random". In such a
  case, the initialization associated with the highest likelihood is kept.

Tinit
  A n x K matrix which contains posterior probabilities for initializing the algo-
  rithm (each line corresponds to an individual).

kernel
  It enables to deal with the n < p problem. By default, no kernel (" ") is used. But
  the user has the choice between 3 options for the kernel: "linear", "sigmoid" or
  "rbf".

disp
  If true, some messages are printed during the clustering. Default is false.

l1
  The l1 penalty value (lasso) which has to be in [0,1]. A small value (close to 0)
  leads to a very sparse loading matrix whereas a value equals to 1 corresponds to
  no sparsity. Default is 0.1.

l2
  The l2 penalty value (elasticnet). Defaults is 0 (no regularization).

nbit
  The number of iterations for the lasso procedure. Defaults is 2.

Value

A list is returned:
K The number of groups.
cls the group membership of each individual estimated by the Fisher-EM algorithm.
P the posterior probabilities of each individual for each group.
U The loading matrix which determines the orientation of the discriminative subspace.
mean The estimated mean in the subspace.
my The estimated mean in the observation space.
prop The estimated mixture proportion.
D The covariance matrices in the subspace.
aic The value of the Akaike information criterion.
bic The value of the Bayesian information criterion.
icl The value of the integrated completed likelihood criterion.
loglik The log-likelihood values computed at each iteration of the FEM algorithm.
l1 The log-likelihood value obtained at the last iteration of the FEM algorithm.
method The method used.
call The call of the function.
plot Some information to pass to the plot.fem function.
crit The model selection criterion used.
l1 The l1 value.
l2 The l2 value.

Author(s)
Charles Bouveyron and Camille Brunet

References

See Also
fem, plot.fem, fem.ari, summary.fem

Examples
data(iris)
res = sfem(iris[, -5], K = 3, model = 'DkBk', l1 = seq(.01, .3, .05))
res
plot(res)
fem.ari(res, as.numeric(iris[, 5]))
Index

fem, 2
fem.ari, 5
FisherEM (FisherEM-package), 2
FisherEM-package, 2
plot.fem, 6
print.fem, 7
sfem, 7