

Package ‘funFEM’

May 8, 2026

Type Package

Title Clustering in the Discriminative Functional Subspace

Version 1.2

Date 2021-10-28

Author Charles Bouveyron

Depends R (>= 2.10), MASS, fda, elasticnet

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

Description The funFEM algorithm (Bouveyron et al., 2014) allows to cluster functional data by modeling the curves within a common and discriminative functional subspace.

License GPL-2

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2021-10-27 14:10:11 UTC

Contents

funFEM-package	2
funFEM	3
velib	6
velov	7
Index	8

funFEM-package	<i>Model-based clustering in the discriminative functional subspaces with the funFEM algorithm</i>
----------------	--

Description

The package provides the funFEM algorithm (Bouveyron et al., 2014) which allows to cluster functional data by modeling the curves within a common and discriminative functional subspace.

Details

Package:	funFEM
Type:	Package
Version:	1.0
Date:	2014-09-06
License:	GPL-2

Author(s)

Charles Bouveyron

Maintainer: <charles.bouveyron@parisdescartes.fr>

References

C. Bouveyron, E. Côme and J. Jacques, The discriminative functional mixture model for the analysis of bike sharing systems, Preprint HAL n.01024186, University Paris Descartes, 2014.

Examples

```
# Clustering the well-known "Canadian temperature" data (Ramsay & Silverman)
basis <- create.bspline.basis(c(0, 365), nbasis=21, norder=4)
fdobj <- smooth.basis(day.5, CanadianWeather$dailyAv[, "Temperature.C"], basis,
  fdnames=list("Day", "Station", "Deg C"))$fd
res = funFEM(fdobj, K=4)

# Visualization of the partition and the group means
par(mfrow=c(1,2))
plot(fdobj, col=res$cls, lwd=2, lty=1)
fdmeans = fdobj; fdmeans$coefs = t(res$prms$my)
plot(fdmeans, col=1:max(res$cls), lwd=2)
```

funFEM

*The funFEM algorithm for the clustering of functional data.***Description**

The funFEM algorithm allows to cluster time series or, more generally, functional data. It is based on a discriminative functional mixture model which allows the clustering of the data in a unique and discriminative functional subspace. This model presents the advantage to be parsimonious and can therefore handle long time series.

Usage

```
funFEM(fd, K=2:6, model = "AkjBk", crit = "bic", init = "kmeans", Tinit = c(), maxit = 50,
      eps = 1e-06, disp = FALSE, lambda = 0, graph = FALSE)
```

Arguments

fd	a functional data object produced by the fda package.
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", "DBk", "DB", "AkjBk", "AkjB", "AkBk", "AkB", "AjBk", "AjB", "ABk", "AB". The option "all" executes the funFEM algorithm on the 12 models and select the best model according to the maximum value obtained by model selection criterion.
crit	the criterion to be used for model selection ('bic', 'aic' or 'icl'). 'bic' is the default.
init	the initialization type ('random', 'kmeans' or 'hclust'). 'kmeans' is the default.
Tinit	a $n \times K$ matrix which contains posterior probabilities for initializing the algorithm (each line corresponds to an individual).
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.
disp	if true, some messages are printed during the clustering. Default is false.
lambda	the l0 penalty (between 0 and 1) for the sparse version. See (Bouveyron et al., 2014) for details. Default is 0.
graph	if true, it plots the evolution of the log-likelihood. Default is false.

Value

A list is returned:

model	the model name.
-------	-----------------

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.
prms	the model parameters.
U	the orientation of the functional subspace according to the basis functions.
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.
nbprm	the number of free parameters in the model.
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

References

C. Bouveyron, E. Côme and J. Jacques, The discriminative functional mixture model for the analysis of bike sharing systems, Preprint HAL n.01024186, University Paris Descartes, 2014.

Examples

```
# Clustering the well-known "Canadian temperature" data (Ramsay & Silverman)
basis <- create.bspline.basis(c(0, 365), nbasis=21, norder=4)
fdobj <- smooth.basis(day.5, CanadianWeather$dailyAv[, "Temperature.C"], basis,
  fdnames=list("Day", "Station", "Deg C"))$fd
res = funFEM(fdobj, K=4)

# Visualization of the partition and the group means
par(mfrow=c(1,2))
plot(fdobj); lines(fdobj, col=res$cls, lwd=2, lty=1)
fdmeans = fdobj; fdmeans$coefs = t(res$prms$my)
plot(fdmeans); lines(fdmeans, col=1:max(res$cls), lwd=2)

# Visualization in the discriminative subspace (projected scores)
par(mfrow=c(1,1))
plot(t(fdobj$coefs) %*% res$U, col=res$cls, pch=19, main="Discriminative space")

#####
# Analysis of the Velib data set

# Load the velib data and smoothing
```

```

data(velib)
basis<- create.fourier.basis(c(0, 181), nbasis=25)
fdobj <- smooth.basis(1:181,t(velib$data),basis)$fd

# Clustrering with FunFEM
res = funFEM(fdobj,K=6,model='AkjBk',init='kmeans',lambda=0,disp=TRUE)

# Visualization of group means
fdmeans = fdobj; fdmeans$coefs = t(res$prms$my)
plot(fdmeans); lines(fdmeans,col=1:res$K,lwd=2,lty=1)
axis(1,at=seq(5,181,6),labels=velib$dates[seq(5,181,6)],las=2)

# # Choice of K (may be long!)
# res = funFEM(fdobj,K=2:20,model='AkjBk',init='kmeans',lambda=0,disp=TRUE)
# plot(2:20,res$plot$bic,type='b',xlab='K',main='BIC')

# Computation of the closest stations from the group means
par(mfrow=c(3,2))
for (i in 1:res$K) {
  matplot(t(velib$data[which.max(res$P[,i]),]),type='l',lty=i,col=i,xaxt='n',
          lwd=2,ylim=c(0,1))
  axis(1,at=seq(5,181,6),labels=velib$dates[seq(5,181,6)],las=2)
  title(main=paste('Cluster',i,' - ',velib$names[which.max(res$P[,i])]))
}

# Visualization in the discriminative subspace (projected scores)
par(mfrow=c(1,1))
plot(t(fdobj$coefs) %%% res$U,col=res$c1s,pch=19,main="Discriminative space")
text(t(fdobj$coefs) %%% res$U)

# # Spatial visualization of the clustering (with library ggmap)
# library(ggmap)
# Mymap = get_map(location = 'Paris', zoom = 12, matype = 'terrain')
# ggmap(Mymap) + geom_point(data=velib$position,aes(longitude,latitude),
#                            colour = I(res$c1), size = I(3))

# FunFEM clustering with sparsity
res2 = funFEM(fdobj,K=res$K,model='AkjBk',init='user',Tinit=res$P,
              lambda=0.01,disp=TRUE)

# Visualization of group means and the selected functional bases
split.screen(c(2,1))
fdmeans = fdobj; fdmeans$coefs = t(res2$prms$my)
screen(1); plot(fdmeans,col=1:res2$K,xaxt='n',lwd=2)
axis(1,at=seq(5,181,6),labels=velib$dates[seq(5,181,6)],las=2)
basis$dropind = which(rowSums(abs(res2$U))==0)
screen(2); plot(basis,col=1,lty=1,xaxt='n',xlab='Disc. basis functions')
axis(1,at=seq(5,181,6),labels=velib$dates[seq(5,181,6)],las=2)
close.screen(all=TRUE)

```

velib

The Vélib data set

Description

This data set contains data from the bike sharing system of Paris, called Vélib. The data are loading profiles of the bike stations over one week. The data were collected every hour during the period Sunday 1st Sept. - Sunday 7th Sept., 2014.

Usage

```
data(velib)
```

Format

The format is:

- data: the loading profiles (nb of available bikes / nb of bike docks) of the 1189 stations at 181 time points.
- position: the longitude and latitude of the 1189 bike stations.
- dates: the download dates.
- bonus: indicates if the station is on a hill (bonus = 1).
- names: the names of the stations.

Source

The real time data are available at <https://developer.jcdecaux.com/> (with an api key).

References

The data were first used in C. Bouveyron, E. Côme and J. Jacques, The discriminative functional mixture model for the analysis of bike sharing systems, Preprint HAL n.01024186, University Paris Descartes, 2014.

Examples

```
data(velib)
matplot(t(velib$data[1:5,]), type='l', lty=1, col=2:5, xaxt='n', lwd=2, ylim=c(0,1))
axis(1, at=seq(5,181,6), labels=velib$dates[seq(5,181,6)], las=2)
```

velov

The Vélov data set

Description

This data set contains data from the bike sharing system of Lyon, called Vélo`v. The data are loading profiles of the bike stations over one week. The data were collected every hour during the period Sunday 9th March - Sunday 16th March, 2014.

Usage

```
data(velov)
```

Format

The format is:

- data: the loading profiles (nb of available bikes / nb of bike docks) of the 345 stations at 181 times.
- position: the longitude and latitude of the 345 bike stations.
- dates: the download dates.
- bonus: indicates if the station is on a hill (bonus = 1).
- names: the names of the stations.

Source

The real time data are available at <https://developer.jcdecaux.com/> (with an api key).

References

The data were first used in C. Bouveyron, E. Côme and J. Jacques, The discriminative functional mixture model for the analysis of bike sharing systems, Preprint HAL n.01024186, University Paris Descartes, 2014.

Examples

```
data(velov)
matplot(t(velov$data[1:5,]), type='l', lty=1, col=2:5, xaxt='n', lwd=2, ylim=c(0,1))
axis(1, at=seq(5, 181, 6), labels=velov$dates[seq(5, 181, 6)], las=2)
```

Index

- * **clustering**

- funFEM, [3](#)

- * **datasets**

- velib, [6](#)

- velov, [7](#)

- * **functional data**

- funFEM, [3](#)

- * **package**

- funFEM-package, [2](#)

funFEM, [3](#)

funFEM-package, [2](#)

velib, [6](#)

velov, [7](#)