

Package ‘RFgroove’

March 17, 2016

Type Package

Title Importance Measure and Selection for Groups of Variables with
Random Forests

Version 1.1

Date 2016-03-16

Author Baptiste Gregorutti

Maintainer Baptiste Gregorutti <baptiste.gregorutti@safety-line.fr>

Description Variable selection tools for groups of variables and func-
tional data based on a new grouped variable importance with random forests.

License GPL (>= 2.0)

Depends randomForest, wmtsa, fda

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-03-17 13:20:10

R topics documented:

RFgroove-package	2
fpca	2
hardThresholding	4
plot.fRFE	5
plot.importance	5
print.fRFE	6
projectWavelet	6
selectFunctional	7
selectGroup	9
selectLevel	10
summary.fRFE	12
toyClassif	12
toyReg	13
toyRegFD	13
varImpGroup	14

Index**16**

RFgroove-package	<i>Importance Measure and Selection for Groups of Variables with Random Forests</i>
------------------	---

Description

Variable selection tools for groups of variables and multivariate functional data based on a new grouped variable importance with Random Forests.

Details

Package: RFgroove
Type: Package
Version: 1.1
Date: 2016-03-16
License: GPL (>=2.0)

Author(s)

Baptiste Gregorutti Maintainer: Baptiste Gregorutti <baptiste.gregorutti@safety-line.fr>

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

fpca	<i>Functional Principal Component Analysis</i>
------	--

Description

Compute the functional PCA from a set of curves.

Usage

```
fpca( x, nbasisInit, propVar = 0.9, reconstruct = FALSE,  
      varName = NULL, verbose = FALSE)
```

Arguments

x	The set of curves.
nbasisInit	The number of initial spline coefficients.
propVar	The proportion of explained variance.
reconstruct	Should the data be reconstruct after dimension reduction ?
varName	The name of the current functional variable.
verbose	Should the details be printed.

Details

The Functional PCA is performed in two steps. First we express each discretized curves as a linear combination of 'nbasisInit' spline basis functions. Then a multivariate PCA is computed on the spline coefficients. The final number of principal components is chosen such that the proportion of explained variance is larger than 'propVar'.

Value

A list with two components:

design	The matrix of the principal components ;
smoothData	The reconstructed data if 'reconstruct == TRUE'.

Author(s)

Baptiste Gregorutti

References

Ramsay, J. O., and Silverman, B. W. (2006), Functional Data Analysis, 2nd ed., Springer, New York.

See Also

[hardThresholding](#)

Examples

```
data(toyRegFD)
x <- toyRegFD$FDlist[[1]]
PCs <- fpca(x=x, nbasisInit=32, propVar=.9, reconstruct=TRUE)
plot(x[1,])
lines(PCs$smoothData[1,], col=2)
```

hardThresholding *Wavelets hard-thresholding rule for independents processes*

Description

This function projects n independent processes on a common wavelet basis and shrinks to zero the n coefficients whose ℓ_2 -norm is lower than a threshold.

Usage

```
hardThresholding(xdata, delta, verbose = FALSE, varName = NULL, wavFilter="s8")
```

Arguments

xdata	The matrix of n independent curves of dimension $N = 2^J$, where J is the number of maximum wavelet level.
delta	The desired threshold. If missing, an automatic threshold is computed.
verbose	Should the details be printed.
varName	The name of the current functional variable.
wavFilter	A character string denoting the filter type. Supported types include: EXTREMAL PHASE (daublet): 'haar', 'd2', 'd4', 'd6', 'd8', 'd10', 'd12', 'd14', 'd16', 'd18', 'd20' LEAST ASYMMETRIC (symmlet): 's2', 's4', 's6', 's8', 's10', 's12', 's14', 's16', 's18', 's20' BEST LOCALIZED: 'l2', 'l4', 'l6', 'l14', 'l18', 'l20' COIFLET: 'c6', 'c12', 'c18', 'c24', 'c30' Default: 's8'.

Value

A list with two components

mht.names	The names of the common wavelet basis after thresholding the coefficients.
estimatedDesign	The new design matrix after thresholding.

Author(s)

Baptiste Gregorutti

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

See Also

[fpca](#)

Examples

```
data(toyRegFD)
x <- toyRegFD$FDlist[[1]]
newDesignMatrix <- hardThresholding(xdata=x, verbose=TRUE)
```

plot.fRFE *Plot function for object of class fRFE*

Description

Plot the error rates or MSE of an object of class fRFE which is the result of the functions [selectFunctional](#), [selectGroup](#) and [selectLevel](#).

Usage

```
## S3 method for class 'fRFE'
plot(x, ...)
```

Arguments

- x An object of class fRFE.
- ... optional graphical parameters.

Author(s)

Baptiste Gregorutti

plot.importance *Plot function for object of class importance*

Description

Plot the values of the grouped variable importance which is the result of the function [varImpGroup](#).

Usage

```
## S3 method for class 'importance'
plot(x, type, horiz, ...)
```

Arguments

x	An object of class importance.
type	A string denoting the plot type. Supported types include 'dotchart' or 'bar'.
horiz	For barplot only. Should the bars be displayed horizontally (default: TRUE).
...	optional graphical parameters.

Author(s)

Baptiste Gregorutti

print.fRFE	<i>Print function for object of class fRFE</i>
------------	--

Description

A print function of an object of class fRFE which is the result of the functions [selectFunctional](#), [selectGroup](#) and [selectLevel](#).

Usage

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

Arguments

x	An object of class fRFE.
...	optional parameters to be passed to the function 'summary'.

Author(s)

Baptiste Gregorutti

projectWavelet	<i>Projection in a wavelet basis</i>
----------------	--------------------------------------

Description

This function projects a set of curves on a wavelet basis.

Usage

```
projectWavelet(xdata, wavFilter="s8")
```

Arguments

xdata	A set of n curves of dimension N .
wavFilter	A character string denoting the filter type. Supported types include: EXTREMAL PHASE (daublet): 'haar', 'd2', 'd4', 'd6', 'd8', 'd10', 'd12', 'd14', 'd16', 'd18', 'd20' LEAST ASYMMETRIC (symmlet): 's2', 's4', 's6', 's8', 's10', 's12', 's14', 's16', 's18', 's20' BEST LOCALIZED: 'l2', 'l4', 'l6', 'l14', 'l18', 'l20' COIFLET: 'c6', 'c12', 'c18', 'c24', 'c30' Default: 's8'.

Value

A matrix of size $n \times N$ containing the wavelet coefficients.

Author(s)

Baptiste Gregorutti

References

Percival, D. B. and Walden, A. T., Wavelet Methods for Time Series Analysis (2000), Cambridge University Press

See Also

[hardThresholding](#), [fpca](#)

Examples

```
data(toyRegFD)
x <- toyRegFD$FDlist[[1]]

design <- projectWavelet(xdata=x)
print(colnames(design))
```

selectFunctional

Grouped variable selection procedure for functional data

Description

Backward variable selection procedure for multivariate functional data which is a set of p functional covariates and n observations of dimension N .

Usage

```
selectFunctional( FDlist, ydata, normalize=TRUE,
                 dimensionReductionMethod=c("fpca", "wave"),
                 nbasisInit, verbose=TRUE, ...)
```

Arguments

FDlist	A p-dimensional list containing the set of functional variables which are matrices of size $n \times N$.
ydata	The outcome data. Must be a factor for classification.
normalize	Should the functions be normalized ?
dimensionReductionMethod	The dimension reduction method, 'fpca' for Functional Principal Component Analysis or 'wave' for the multiple wavelet thresholding.
nbasisInit	The number of initial spline coefficients.
verbose	Should the details be printed.
...	further arguments passed to or from other methods.

Value

An object of class fRFE which is a list with the following components:

nselected	The number of selected functional variables ;
selection	The selected functional variables ;
selectionIndexes	The indexes of selected functional variables in the input data 'FDlist' ;
error	The prediction error computed in each iteration of the backward procedure ;
typeRF	The type of the forests, classification or regression ;
ranking	The final ranking of the functional variables ;
rankingIndexes	The final ranking indexes of the functional variables.

Author(s)

Baptiste Gregorutti

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

See Also

[selectGroup](#), [varImpGroup](#)

Examples

```

data(toyRegFD)
varSel <- selectFunctional( toyRegFD$FDlist, toyRegFD$Y, normalize=FALSE,
                           dimensionReductionMethod="fpca", nbasisInit=16,
                           verbose=FALSE, ntree=10)

summary(varSel)
plot(varSel)

```

selectGroup	<i>Grouped variable selection procedure</i>
-------------	---

Description

A grouped backward variable selection procedure.

Usage

```

selectGroup(design, ydata, varNames, nvarGroup,
            typeRF = ifelse(is.factor(ydata), "classif", "reg"),
            verbose = TRUE, ntree = 500, ...)

```

Arguments

design	The design matrix.
ydata	The outcome data. Must be a factor for classification.
varNames	The vector of the group names.
nvarGroup	The vector of the number of variables in each group.
typeRF	The type of forest we want to construct, 'classif' for classification or 'reg' for regression.
verbose	Should the details be printed.
ntree	The number of trees in the forests (default: 500).
...	optional parameters to be passed to the 'varImpGroup' function.

Value

An object of class fRFE which is a list with the following components:

nselected	The number of selected groups ;
selection	The selected groups ;
selectionIndexes	The indexes of selected groups in the input matrix 'design' ;
error	The prediction error computed in each iteration of the backward procedure ;
typeRF	The type of the forests, classification or regression ;
ranking	The final ranking of the groups ;
rankingIndexes	The final ranking indexes of the groups.

Author(s)

Baptiste Gregorutti

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

See Also

[selectLevel](#), [selectFunctional](#), [varImpGroup](#)

Examples

```
cat("\n\nClassification\n")
data(toyClassif)
attach(toyClassif)

cat("Case 1\n")
nvarGroup <- c(2,1,6); names(nvarGroup) <- paste("G", 1:length(nvarGroup), sep="")
summary(varSel <- selectGroup(design=X, ydata=Y, nvarGroup=nvarGroup,
                             verbose=TRUE, ntree=500, normalize=FALSE))
plot(varSel)

cat("Case 2\n")
nvarGroup <- rep(1,9); names(nvarGroup) <- paste("G", 1:length(nvarGroup), sep="")
summary(varSel <- selectGroup(design=X, ydata=Y, nvarGroup=nvarGroup,
                             verbose=TRUE, ntree=500, normalize=FALSE))
plot(varSel)

detach(toyClassif)
```

selectLevel

Wavelet levels selection procedure

Description

A grouped backward variable selection procedure for selecting the most significant wavelet levels of a functional variable. The groups are the wavelet coefficients belonging to the same frequency level.

Usage

```
selectLevel(design, ydata, typeRF = ifelse(is.factor(ydata), "classif", "reg"),
            verbose = TRUE, ntree = 500, ...)
```

Arguments

design	The design matrix of a functional variable.
ydata	The outcome data. Must be a factor for classification.
typeRF	The type of forest we want to construct, 'classif' for classification or 'reg' for regression.
verbose	Should the details be printed.
ntree	The number of trees in the forests (default: 500).
...	optional parameters to be passed to the 'varImpGroup' function.

Value

An object of class fRFE which is a list with the following components:

nselected	The number of selected wavelet levels.
selection	The selected wavelet levels.
selectionIndexes	The indexes of selected wavelet levels in the input matrix 'design'.
error	The prediction error computed in each iteration of the backward procedure.
typeRF	The type of the forests, classification or regression.
ranking	The final ranking of the wavelet levels.
rankingIndexes	The final ranking indexes of the wavelet levels.

Author(s)

Baptiste Gregorutti

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

See Also

[selectGroup](#), [selectFunctional](#), [varImpGroup](#)

Examples

```
data(toyRegFD)
x <- toyRegFD$FDlist[[1]]
y <- toyRegFD$Y

design <- projectWavelet(xdata=x)
summary(levSel <- selectLevel(design, y, ntree=100, verbose=TRUE))
plot(levSel)
```

`summary.fRFE`*Summary function for object of class fRFE*

Description

A summary function of an object of class fRFE which is the result of the functions [selectFunctional](#), [selectGroup](#) and [selectLevel](#).

Usage

```
## S3 method for class 'fRFE'  
summary(object, ...)
```

Arguments

<code>object</code>	An object of class fRFE.
<code>...</code>	further arguments passed to or from other methods.

Author(s)

Baptiste Gregorutti

`toyClassif`*A toy data of classification*

Description

A toy data of classification drawn from a mixture Gaussian distribution. Only the three first variables are discriminant.

Usage

```
data(toyClassif)
```

Format

A list of two components:

- X: the design matrix of size $n \times p$ drawn from a mixture Gaussian distribution ;
- Y: the class labels.

toyReg	<i>A toy data of regression</i>
--------	---------------------------------

Description

A toy data of regression drawn from a multivariate Gaussian distribution. Only the three first variables are discriminant.

Usage

```
data(toyReg)
```

Format

A list of two components:

- X: the design matrix of size $n \times p$ drawn from a multivariate Gaussian distribution ;
- Y: the outcome drawn from the Gaussian distribution.

toyRegFD	<i>A functional dataset for regression</i>
----------	--

Description

A set of 20 functional variables for regression. Only the 8 first variables are predictive.

Usage

```
data(toyRegFD)
```

Format

A list of two components:

- FDlist: a list containing the p functional variables, i.e. p matrices of size $n \times N$ with n the number of N -dimensional curves ;
- Y: the outcome drawn from the Gaussian distribution.

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2014). Grouped variable importance with random forests and application to multivariate functional data analysis. arXiv:1411.4170

varImpGroup *A grouped variable importance with Random Forests*

Description

A permutation variable importance for groups of variables with Random Forests.

Usage

```
varImpGroup(object, xdata, ngroups = length(nvarGroup), nvarGroup,  
            idxGroup, groupsNames = names(nvarGroup),  
            normalize = (length(unique(nvarGroup)) != 1))
```

Arguments

object	A randomForest object.
xdata	The input data.
ngroups	The number of groups.
nvarGroup	The vector of the number of variables in each group.
idxGroup	A list of size ‘ngroups’ containing the indexes of each group starting from 0.
groupsNames	The group names.
normalize	Should the normalized grouped importance measure be computed.

Value

An object of class ‘importance’ which is a vector of the importance for each group.

Author(s)

Baptiste Gregorutti

References

Gregorutti, B., Michel, B. and Saint Pierre, P. (2015). Grouped variable importance with random forests and application to multiple functional data analysis, *Computational Statistics and Data Analysis* 90, 15-35.

See Also

[selectGroup](#), [selectLevel](#), [selectFunctional](#), [plot.importance](#)

Examples

```
data(toyClassif)
attach(toyClassif)

rf <- randomForest(x=X,y=Y,keep.forest=TRUE, keep.inbag=TRUE, ntree=500)
ngroups <- 3
nvarGroup <- c(4,3,6)
idxGroup <- list(c(0,1,2,5), c(2,4,5), c(0,1,5,6,7,8))
grImp <- varImpGroup(rf, X, ngroups, nvarGroup, idxGroup, NULL, normalize=FALSE )
cat("Group importance\n", grImp, "\n")

detach(toyClassif)
```

Index

`fpca`, [2](#), [5](#), [7](#)

`hardThresholding`, [3](#), [4](#), [7](#)

`plot.fRFE`, [5](#)

`plot.importance`, [5](#), [14](#)

`print.fRFE`, [6](#)

`projectWavelet`, [6](#)

`RFgroove` (`RFgroove`-package), [2](#)

`RFgroove`-package, [2](#)

`selectFunctional`, [5](#), [6](#), [7](#), [10–12](#), [14](#)

`selectGroup`, [5](#), [6](#), [8](#), [9](#), [11](#), [12](#), [14](#)

`selectLevel`, [5](#), [6](#), [10](#), [10](#), [12](#), [14](#)

`summary.fRFE`, [12](#)

`toyClassif`, [12](#)

`toyReg`, [13](#)

`toyRegFD`, [13](#)

`varImpGroup`, [5](#), [8](#), [10](#), [11](#), [14](#)