# Package ‘fdcov’

December 21, 2017

**Title** Analysis of Covariance Operators

**Version** 1.1.0

**Author** Alessandra Cabassi [aut],
Adam B Kashlak [aut, cre],
Davide Pigoli [ctb]

**Maintainer** Adam B Kashlak <kashlak@ualberta.ca>

**URL** https://github.com/acabassi/fdcov

**BugReports** https://github.com/acabassi/fdcov/issues

**Description** Provides a variety of tools for the analysis of covariance operators including k-sample tests for equality and classification and clustering methods found in the works of Cabassi et al (2017) <doi:10.1214/17-EJS1347>, Kashlak et al (2017) <arXiv:1604.06310>, Pigoli et al (2014) <doi:10.1093/biomet/asu008>, and Panaretos et al (2010) <doi:10.1198/jasa.2010.tm09239>.

**Depends** R (>= 3.2.0)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1.9000

**Imports** stats, matlab, corrplot

**Suggests** fds, knitr

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-12-21 21:17:44 UTC

## R topics documented:

---

## `fdcov-package` *Analysis of Covariance Operators.*

---

### Description

`fdcov` a variety of tools for the analysis of covariance operators including k-sample tests for equality and classification and clustering methods.

### Details

This package contains a collection of tools for performing statistical inference on functional data specifically through an analysis of the covariance structure of the data. It includes two methods for performing a k-sample test for equality of covariance in `ksample.perm` and `ksample.com` and two methods for 2-sample tests for equality assuming Gaussian data in `ksample.gauss` and `ksample.vstab`. For supervised and unsupervised learning, it contains a method to classify functional data with respect to each category's covariance operator in `classif.com`, and it contains a method to cluster functional data, `cluster.com`, again based on the covariance structure of the data.

The current version of this package assumes that all functional data is sampled on the same grid at the same intervals. Future updates are planned to allow for the below methods to interface with the `fda` package and its functional basis representations of the data.

### Author(s)

Alessandra Cabassi <`alessandra.cabassi@mail.polimi.it`>, Adam B Kashlak <`kashlak@ualberta.ca`>

### References

Cabassi, A., Pigoli, D., Secchi, P., Carter, P. A. (2017). Permutation tests for the equality of covariance operators of functional data with applications to evolutionary biology. Electron. J. Statist. 11(2), pp.3815–3840.

Kashlak, Adam B, John AD Aston, and Richard Nickl (2016). "Inference on covariance operators via concentration inequalities: k-sample tests, classification, and clustering via Rademacher complexities", April, 2016 (in review)

Pigoli, Davide, John AD Aston, Ian L Dryden, and Piercesare Secchi. "Distances and inference for covariance operators." Biometrika (2014): 101(2):409–422.

Panaretos, Victor M., David Kraus, and John H. Maddocks. "Second-order comparison of Gaussian random functions and the geometry of DNA minicircles." Journal of the American Statistical Association 105.490 (2010): 670-682.

## See Also

Useful links:

- https://github.com/acabassi/fdcov
- Report bugs at https://github.com/acabassi/fdcov/issues

# `classifier-com` *Functional data classifier via concentration inequalities*

## Description

`classif.com` trains a covariance operator based functional data classifier that makes use of concentration inequalities. `predict.classif.com` uses the previously trained classifier to classify new observations.

## Usage

```
classif.com(datGrp, dat)

## S3 method for class 'classif.com'
predict(object, dat, SOFT = FALSE, LOADING = FALSE,
  ...)
```

## Arguments

| | |
|---|---|
| `datGrp` | A vector of group labels. |
| `dat` | (n X m) data matrix of n samples of m long vectors. |
| `object` | A concentration-of-measure classifier object of class inheriting from `classif.com`. |
| `SOFT` | Boolean flag, which if TRUE, returns soft classification for each observation. |
| `LOADING` | Boolean flag, which if TRUE, prints a loading bar. |
| `...` | additional arguments affecting the predictions produced. |

## Details

These functions are used to train a functional data classifier and to predict the labels for a new set of observations. This method classifies based on the distances between each groups' sample covariance operator. A simplified version of Talagrand's concentration inequality is used to achieve this.

If the flag SOFT is set to TRUE, then soft classification occurs. In this case, given k different labels, a k-long probability vector is returned for each observation whose entries correspond to the probabilities that the observed function belongs to each specific label.

## Value

`classif.com` returns a functional data classifier object. `predict.classif.com` returns a vector of n labels ( or an array of n probability vectors if SOFT=TRUE )

### Author(s)

Adam B Kashlak <kashlak@ualberta.ca>

### References

Kashlak, Adam B, John AD Aston, and Richard Nickl (2016). "Inference on covariance operators via concentration inequalities: k-sample tests, classification, and clustering via Rademacher complexities", (in review)

### Examples

```
## Not run:
 library(fds);
 # Setup training data
 dat1 = rbind(
   t(aa$y[,1:100]), t(ao$y[,1:100]), t(dcl$y[,1:100]),
   t(iy$y[,1:100]), t(sh$y[,1:100])
 );
 # Setup testing data
 dat2 = rbind(
   t(aa$y[,101:400]), t(ao$y[,101:400]), t(dcl$y[,101:400]),
   t(iy$y[,101:400]), t(sh$y[,101:400])
 );

 datgrp = gl(5,100);
 clCom = classif.com( datgrp, dat1 );
 grp = predict( clCom, dat2, LOADING=TRUE );
 acc = c(
   sum( grp[1:300]==1 ), sum( grp[301:600]==2 ), sum( grp[601:900]==3 ),
   sum( grp[901:1200]==4 ), sum( grp[1201:1500]==5 )
 )/300;
 print(rbind(gl(5,1),signif(acc,3)));

## End(Not run)
```

## cluster.com *Functional data clustering via concentration inequalities*

### Description

cluster.com clusters sets of functional data via their covariance operators making use of an EM style algorithm with concentration inequalities.

### Usage

```
cluster.com(dat, labl = NULL, grpCnt = 2, iter = 30, SOFT = FALSE,
  PRINTLK = TRUE, LOADING = FALSE, IGNORESTOP = FALSE)
```

## Arguments

`dat` (n X m) data matrix of n samples of m long vectors.

`labl` An optional vector of n labels to group curves. (see Details)

`grpCnt` Number of clusters into which to split the data.

`iter` Number of iterations for EM algorithm.

`SOFT` Boolean flag for whether or not category probabilities should be returned.

`PRINTLK` Boolean flag, which if TRUE, prints likelihood values for each iteration.

`LOADING` Boolean flag, which if TRUE, prints a loading bar.

`IGNORESTOP` Boolean flag, which if TRUE, will ignore early stopping conditions and cause the EM algorithm to run for the total amount of desired iterations.

## Details

This function clusters individual curves or sets of curves by considering the distance between their covariance operator and each estimated category covariance operator. The implemented algorithm reworks the concentration inequality based classification method `classif.com` into an EM style algorithm. This method iteratively updates the probability of a given observation belonging to each of the k categories. These probabilities are in turn used to update the category means. This process continues until either the total number of iterations is reached or a computed likelihood begins to decrease signaling the arrival of a local optimum.

If the argument labl is NULL, then every curve is clustered separately. If labl contains factors used to group the curves, then each set of curves is classified as one group. For example, if you have multiple speakers and multiple speech samples from each speaker, you can group the data from each speaker together in order to cluster based on each speakers' covariance operator rather than based on each speech sample individually.

If the flag SOFT is set to TRUE, then soft clustering occurs. In this case, given k different labels, a k-long probability vector is returned for each observation whose entries correspond to the probability that the observed function belongs to a specific label.

## Value

`cluster.com` returns a vector a labels with one entry for each row of data corresponding to one of the k categories ( or an array of probability vectors if SOFT=TRUE ).

## Author(s)

Adam B Kashlak <`kashlak@ualberta.ca`>

## References

Kashlak, Adam B, John A D Aston, and Richard Nickl (2016). "Inference on covariance operators via concentration inequalities: k-sample tests, classification, and clustering via Rademacher complexities", in review

### Examples

```
## Not run:
 # Load phoneme data
 library(fds);
 # Setup data to be clustered
 dat  = rbind( t(aa$y[,1:20]),t(iy$y[,1:20]),t(sh$y[,1:20]) );
 # Cluster data into three groups
 clst = cluster.com(dat,grpCnt=3);
 matrix(clst,3,20,byrow=TRUE);

 # cluster groups of curves
 dat  = rbind( t(aa$y[,1:40]),t(iy$y[,1:40]),t(sh$y[,1:40]) );
 lab  = gl(30,4);
 # Cluster data into three groups
 clst = cluster.com(dat,labl=lab,grpCnt=3);
 matrix(clst,3,10,byrow=TRUE);

## End(Not run)
```

## `ksample.com` *k-sample test for equality of covariance operators*

### Description

`ksample.com` performs a k-sample test for equality of covariance operators using concentration inequalities.

### Usage

```
ksample.com(dat, grp, p = 1, alpha = 0.05, scl1 = 1, scl2 = 1)
```

### Arguments

| | |
|---|---|
| `dat` | (n X m) data matrix of n samples of m long vectors. |
| `grp` | n long vector of group labels. |
| `p` | p-Schatten norm in [1,Inf], Default is 1. (see Details) |
| `alpha` | the desired size of the test, Default is 0.05. |
| `scl1` | scales the deviation part of the concentration inequality. (see Details) |
| `scl2` | scales the Rademacher part of the concentration inequality. (see Details) |

## Details

This function tests for the equality of k covariance operators given k sets of functional data. It makes use of Talagrand's concentration inequality in the Banach space setting. The argument p specifies the p-Schatten norm used in the test. As detailed in Kashlak et al (2016), the most power is achieved using the trace class norm (p=1), which is the default value.

This test is inherently conservative as it constructed by concatenating many concentration inequalities together. Consequently, the method may be tuned by adjusting the arguments scl1 and scl2 to achieve the desired empirical size for the users specific data set. Otherwise, it can be used as a quick first pass before a more powerful but more computational test, such as specifically `ksample.perm`, is run. More information on tuning this method can be found in the reference.

## Value

Boolean value for whether or not the test believes the alternative hypothesis is true. ( i.e. Does there exist at least two categories of the k whose covariance operators are not equal? )

## Author(s)

Adam B Kashlak <`kashlak@ualberta.ca`>

## References

Kashlak, Adam B, John AD Aston, and Richard Nickl (2016). "Inference on covariance operators via concentration inequalities: k-sample tests, classification, and clustering via Rademacher complexities", (in review)

## Examples

```
# Load in phoneme data
library(fds)
# Setup data arrays
dat1 = rbind( t(aa$y)[1:20,], t(sh$y)[1:20,] );
dat2 = rbind( t(aa$y)[1:20,], t(ao$y)[1:20,] );
dat3 = rbind( dat1, t(ao$y)[1:20,] );
# Setup group labels
grp1 = gl(2,20);
grp2 = gl(2,20);
grp3 = gl(3,20);
# Compare two disimilar phonemes (should return TRUE)
ksample.com(dat1,grp1);
# Compare two similar phonemes (should return FALSE)
ksample.com(dat2,grp2);
# Compare three phonemes (should return TRUE)
ksample.com(dat3,grp3);
```

## ksample.gauss *k-sample test for equality of covariance operators*

### Description

ksample.gauss performs a k-sample test for equality of covariance operators under the assumption that the data arises from a Gaussian process.

### Usage

```
ksample.gauss(dat1, dat2, K = 5)
```

### Arguments

| | |
|---|---|
| dat1 | the first set of data with one entry per row |
| dat2 | the second set of data with one entry per row |
| K | the number of basis vectors to use, Default is 5. |

### Details

ksample.vstab applies a similar method that has been modified to stabilize the variance. See the reference paper for more details on the mathematics of these methods.

These two methods use the Karhunen-Loeve expansion (eigen expansion for functional data) to represent the data in terms of K eigen-functions. Then a test statistic with asymptotic chi-squared distribution is computed in order to test for the equality of the covariance operators based on the two samples. If K is set to be 0, then the methods determine the number of eigen-functions to retain.

### Value

p-value testing whether or not the two samples have differing covariance operators.

### Author(s)

Adam B Kashlak <kashlak@ualberta.ca>

### References

Panaretos, Victor M., David Kraus, and John H. Maddocks. "Second-order comparison of Gaussian random functions and the geometry of DNA minicircles." Journal of the American Statistical Association 105.490 (2010): 670-682.

## Examples

```
# Load in phoneme data
library(fds)
# Set up test data
dat1 = t(aa$y)[1:20,];
dat2 = t(sh$y)[1:20,];
dat3 = t(aa$y)[21:40,];
# Compare two disimilar phonemes
# Resulting in a small p-value
ksample.gauss(dat1,dat2,K=5);
ksample.vstab(dat1,dat2,K=5);
# Compare two sets of the same phonemes
# Resulting in a large p-value
ksample.gauss(dat1,dat3,K=5);
ksample.vstab(dat1,dat3,K=5);
```

---

# ksample.perm *Multiple-sample permutation test for the equality of covariance operators of functional data*

---

## Description

The method performs a test for the equality of the covariance operators of multiple data samples. It can also perform all of the pairwise comparisons between the groups and compute a p-value for each of them. This feature is useful when the global null hypothesis is rejected, so one may want to find out which samples have different covariances.

## Usage

```
ksample.perm(dat, grp, iter = 1000, perm = "sync", dist = "sq",
  adj = TRUE, comb = "tipp", part = FALSE, cent = FALSE, load = FALSE)
```

## Arguments

| | |
|---|---|
| dat | n X p data matrix of n samples of p long vectors. |
| grp | n long vector of group labels. |
| iter | Number of permutations. Defaults to 1000. |
| perm | Type of permutation, can be ’sync’ (if all the data samples are of the same size) or ’pool’. Defaults to ’sync’ |
| dist | Distance between covariance operators. Can be ’sq’ (square-root distance), ’tr’ (trace distance),’pr’ (Procrustes distance), ’hs’(Hilbert-Schmidt distance) or ’op’ (operator distance). Defaults to ’sq’. |
| adj | p-value adjustment. Defaults to TRUE. |
| comb | Can be ’tipp’ (for Tippett), ’maxT’, ’dire’ (direct), ’fish’ (Fisher) or ’lipt’ (Liptak). Defaults to ’tipp’. |

`part` If FALSE, the function computes only the global p-value; otherwise it computes also all the p-values corresponding to the pairwise comparisons. Defaults to FALSE.

`cent` If FALSE, the mean functions of the groups are supposed to be different, therefore data are centred before performing the test. Defaults to FALSE.

`load` Boolean flag, which if TRUE, prints a loading bar.

## Value

If `part` is set to FALSE, the output is the p-value associated to the global test. If `part` is TRUE, the function returns also all the p-values of the pairwise comparisons.

## Author(s)

Alessandra Cabassi <`alessandra.cabassi@mail.polimi.it`>

## References

Cabassi, A., Pigoli, D., Secchi, P., Carter, P. A. (2017). Permutation tests for the equality of covariance operators of functional data with applications to evolutionary biology. Electron. J. Statist. 11(2), pp.3815–3840.

Pigoli, Davide, John A. D. Aston, Ian L. Dryden, and Piercesare Secchi (2014). "Distances and inference for covariance operators." Biometrika: 101(2):409–422.

## Examples

```
## Not run:
## Phoneme data

library(fdcov)
library(fds)

# Create data set
data(aa); data(ao); data(dcl);data(iy);data(sh)
dat = cbind(aa$y[,1:20],ao$y[,1:20],dcl$y[,1:20],iy$y[,1:20],sh$y[,1:20])
dat = t(dat)
grp = c(rep(1,20),rep(2,20),rep(3,20),rep(4,20),rep(5,20))

# Test the equality of the covariance operators
p = ksample.perm(dat, grp, iter=100, part = TRUE)
p$global # global p-value
p$partial # partial p-values

## End(Not run)
```

## ksample.vstab *k-sample test for equality of covariance operators*

### Description

`ksample.gauss` performs a k-sample test for equality of covariance operators under the assumption that the data arises from a Gaussian process.

### Usage

```
ksample.vstab(dat1, dat2, K = 5)
```

### Arguments

| | |
|---|---|
| `dat1` | the first set of data with one entry per row |
| `dat2` | the second set of data with one entry per row |
| `K` | the number of basis vectors to use, Default is 5. |

### Details

`ksample.vstab` applies a similar method that has been modified to stabilize the variance. See the reference paper for more details on the mathematics of these methods.

These two methods use the Karhunen-Loeve expansion (eigen expansion for functional data) to represent the data in terms of K eigen-functions. Then a test statistic with asymptotic chi-squared distribution is computed in order to test for the equality of the covariance operators based on the two samples. If K is set to be 0, then the methods determine the number of eigen-functions to retain.

### Value

p-value testing whether or not the two samples have differing covariance operators.

### Author(s)

Adam B Kashlak <`kashlak@ualberta.ca`>

### References

Panaretos, Victor M., David Kraus, and John H. Maddocks. "Second-order comparison of Gaussian random functions and the geometry of DNA minicircles." Journal of the American Statistical Association 105.490 (2010): 670-682.

## Examples

```
# Load in phoneme data
library(fds)
# Set up test data
dat1 = t(aa$y)[1:20,];
dat2 = t(sh$y)[1:20,];
dat3 = t(aa$y)[21:40,];
# Compare two disimilar phonemes
# Resulting in a small p-value
ksample.gauss(dat1,dat2,K=5);
ksample.vstab(dat1,dat2,K=5);
# Compare two sets of the same phonemes
# Resulting in a large p-value
ksample.gauss(dat1,dat3,K=5);
ksample.vstab(dat1,dat3,K=5);
```

# perm.plot *Plot partial p-values*

## Description

`perm.plot` plots all of the partial comparison p-values in a matrix.

## Usage

```
perm.plot(p, k, lab = NULL, save = FALSE, name = "pvalues.eps")
```

## Arguments

| | |
|---|---|
| `p` | Output of function perm.test, if part = TRUE. |
| `k` | Number of groups, must be greater than 2. |
| `lab` | Group labels. Defaults to 1, 2, ..., k. |
| `save` | Boolean variable that indicates if the plot must be saved as an .eps. Defaults to FALSE. |
| `name` | If save is TRUE, this is the filename of the plot. Defaults to `pvalues.eps`. |

## Value

`perm.plot` plots the partial p-values in a matrix.

## Author(s)

Alessandra Cabassi `<alessandra.cabassi@mail.polimi.it>`

## References

Pigoli, Davide, John A. D. Aston, Ian L. Dryden, and Piercesare Secchi (2014). "Distances and inference for covariance operators." Biometrika: 101(2):409–422.

## Examples

```
## Not run:
## Phoneme data

library(fdcov)
library(fds)

# Create data set
data(aa); data(ao); data(dcl);data(iy);data(sh)
dat=cbind(aa$y[,1:20],ao$y[,1:20],dcl$y[,1:20],iy$y[,1:20],sh$y[,1:20])
dat=t(dat)
grp=c(rep(1,20),rep(2,20),rep(3,20),rep(4,20),rep(5,20))

# Test the equality of the covariance operators
p=ksample.perm(dat,grp,iter=100,only.glob=FALSE)

# Plot partial p-values
perm.plot(p,5, lab=c('aa','ao','dcl','iy','sh'))

## End(Not run)
```

# Index