Package ‘covsep’

January 28, 2016

Title Tests for Determining if the Covariance Structure of 2-Dimensional Data is Separable

Version 1.0.0

Author Shahin Tavakoli [aut, cre],
Davide Pigoli [ctb],
John Aston [ctb]

Maintainer Shahin Tavakoli <s.tavakoli@statslab.cam.ac.uk>

Description Functions for testing if the covariance structure of 2-dimensional data (e.g. samples of surfaces X_i = X_i(s,t)) is separable, i.e. if covariance(X) = C_1 x C_2. A complete descriptions of the implemented tests can be found in the paper arXiv:1505.02023.

Depends R (>= 3.2.3)

Imports mvtnorm (>= 1.0.4)

License GPL-2

LazyData true

URL http://arxiv.org/abs/1505.02023

NeedsCompilation no

Repository CRAN

Date/Publication 2016-01-28 20:45:37

R topics documented:

C1 *A covariance matrix*

Description

Marginal covariance matrix C1 used for simulations in the paper http://arxiv.org/abs/1505.02023

Usage

```
C1
```

Format

```
num [1:32, 1:32] 0.1036 0.0726 0.0654 0.063 0.0891 ...
```

Details

This is a 32x32 real-valued covariance matrix.

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, http://arxiv.org/abs/1505.02023

Examples

```
data(C1)
str(C1)
```

C2 — *A covariance matrix*

Description

Marginal covariance matrix C2 used for simulations in paper http://arxiv.org/abs/1505.02023

Usage

```
C2
```

Format

```
num [1:7, 1:7] 1.598 1.604 1.384 1.167 0.909 ...
```

Details

This is a 7x7 real-valued covariance matrix.

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, http://arxiv.org/abs/1505.02023

Examples

```
data(C2)
str(C2)
```

clt_test — *Test for separability of covariance operators for Gaussian process.*

Description

This function performs the asymptotic test for the separability of the covariance operator for a random surface generated from a Gaussian process (described in the paper http://arxiv.org/abs/1505.02023).

Usage

```
clt_test(Data, L1, L2)
```

Arguments

- `Data` — a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.
- `L1` — an integer or vector of integers in $1 : p$ indicating the eigenfunctions in the first direction to be used for the test.
- `L2` — an integer or vector of integers in $1 : q$ indicating the eigenfunctions in the second direction to be used for the test.

Value

The p-value of the test for each pair `(l1,l2) = (L1[k], L2[k])`, for `k = 1:length(L1)`.

Details

If L1 and L2 are vectors, they need to be of the same length.

The function tests for separability using the projection of the covariance operator in the separable eigenfunctions `u_i tensor v_j : i = 1, ..., l1; j = 1, ..., l2`, for each pair `(l1,l2) = (L1[k], L2[k])`, for `k = 1:length(L1)`.

The test works by using asymptotics, and is only valid if the data is assumed to be Gaussian.

The surface data needs to be measured or resampled on a common regular grid or on common basis functions.

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, http://arxiv.org/abs/1505.02023

See Also

`empirical_bootstrap_test`, `gaussian_bootstrap_test`

Examples

```
data(SurfacesData)
clt_test(SurfacesData, L1=c(1,2), L2=c(1,4))
```

---

covsep — *covsep: tests for determining if the covariance structure of 2-dimensional data is separable*

Description

Functions for testing if the covariance structure of 2-dimensional data (e.g. samples of surfaces X_i = X_i(s,t)) is separable, i.e. if cov(X) = C_1 x C_2. A complete descriptions of the implemented tests can be found in the paper arXiv:1505.02023.

Main functions

The main functions are

- clt_test,
- gaussian_bootstrap_test,
- empirical_bootstrap_test,
- HS_gaussian_bootstrap_test,
- HS_empirical_bootstrap_test

`difference_fullcov` *compute the difference between the full sample covariance and its separable approximation*

Description

compute the difference between the full sample covariance and its separable approximation

Usage

```
difference_fullcov(Data)
```

Arguments

`Data`	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.

Value

A `d1 x d2 x d1 x d2` array, where `d1 = nrow(Data)` and `d2 = ncol(Data)`.

Details

This is an internal function.

empirical_bootstrap_test

Projection-based empirical bootstrap test for separability of covariance structure

Description

This function performs the test for the separability of covariance structure of a random surface based on the empirical bootstrap procedure described in the paper http://arxiv.org/abs/1505.02023.

Usage

```
empirical_bootstrap_test(Data, L1, L2, studentize, B = 100, verbose = FALSE)
```

Arguments

Data	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.
L1	an integer or vector of integers in $1:p$ indicating the eigenfunctions in the first direction to be used for the test.
L2	an integer or vector of integers in $1:q$ indicating the eigenfunctions in the second direction to be used for the test.
studentize	parameter to specify which type of studentization is performed. Possible options are 'no', 'diag' or 'full' (see details section).
B	number of bootstrap replicates to be used.
verbose	logical parameter for printing progress

Value

The p-value of the test for each pair `(l1,l2) = (L1[k], L2[k])`, for `k = 1:length(L1)`.

Details

This function performs the test of separability of the covariance structure for a random surface (introduced in the paper http://arxiv.org/abs/1505.02023), when generated from a Gaussian process. The sample surfaces need to be measured on a common regular grid. The test consider a subspace formed by the tensor product of eigenfunctions of the separable covariances. It is possible to specify the number of eigenfunctions to be considered in each direction.

If L1 and L2 are vectors, they need to be of the same length.

The function tests for separability using the projection of the covariance operator in the separable eigenfunctions `u_i x v_j : i = 1, ..., l1; j = 1, ..., l2`, for each pair (l1,l2) = (L1[k], L2[k]), for k = 1:length(L1).

`studentize` can take the values

'no' no studentization is performed

'diag' each projection coordinate is renormalized by an estimate of its standard deviation

'full' the projection coordinates are renormalized by an estimate of their joint covariance

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, `http://arxiv.org/abs/1505.02023`

See Also

`gaussian_bootstrap_test`, `clt_test`

Examples

```
data(SurfacesData)
## Not run: empirical_bootstrap_test(SurfacesData,L1=1,L2=1, B=100, studentize='full')
```

`gaussian_bootstrap_test` *Projection-based Gaussian (parametric) bootstrap test for separability of covariance structure*

Description

This function performs the test for the separability of covariance structure of a random surface generated from a Gaussian process, based on the parametric bootstrap procedure described in the paper `http://arxiv.org/abs/1505.02023`

Usage

```
gaussian_bootstrap_test(Data, L1, L2, studentize = "full", B = 100,
  verbose = FALSE)
```

Arguments

`Data`	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.
`L1`	an integer or vector of integers in $1:p$ indicating the eigenfunctions in the first direction to be used for the test.
`L2`	an integer or vector of integers in $1:q$ indicating the eigenfunctions in the second direction to be used for the test.
`studentize`	parameter to specify which type of studentization is performed. Possible options are 'no', 'diag' or 'full' (see details section).
`B`	number of bootstrap replicates to be used.
`verbose`	logical parameter for printing progress

Value

The p-value of the test for each pair `(l1,l2) = (L1[k], L2[k])`, for `k = 1:length(L1)`.

Details

This function performs the test of separability of the covariance structure for a random surface (introduced in the paper http://arxiv.org/abs/1505.02023), when generated from a Gaussian process. The sample surfaces need to be measured on a common regular grid. The test consider a subspace formed by the tensor product of eigenfunctions of the separable covariances. It is possible to specify the number of eigenfunctions to be considered in each direction.

If L1 and L2 are vectors, they need to be of the same length.

The function tests for separability using the projection of the covariance operator in the separable eigenfunctions `u_i x v_j : i = 1, ..., l1; j = 1, ..., l2`, for each pair (l1,l2) = (L1[k], L2[k]), for k = 1:length(L1).

`studentize` can take the values

'no' no studentization is performed

'diag' each projection coordinate is renormalized by an estimate of its standard deviation

'full' the projection coordinates are renormalized by an estimate of their joint covariance

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, http://arxiv.org/abs/1505.02023

See Also

empirical_bootstrap_test, clt_test

Examples

```
data(SurfacesData)
## Not run: gaussian_bootstrap_test(SurfacesData,L1=1,L2=1,B=100,
studentize='full')
## End(Not run) #' pvalue of the test
```

---

generate_surface_data *Generate surface data*

Description

Generate samples of surface data

Usage

```
generate_surface_data(N, C1, C2, gamma, distribution = "gaussian")
```

Arguments

`N`	sample size
`C1`	row covariance
`C2`	column covariance
`gamma`	parameter to specify how much the covariance is separable.
`distribution`	distribution of the data

Value

A `N x dim(C1)[1] x dim(C2)[1]` array containing the generated data

Details

`gamma` can take values between 0 and 1; `gamma=0` corresponds to a separable covariance, `gamma=1` corresponds to a non-separable covariance (described in the paper http://arxiv.org/abs/1505.02023). Values of `gamma` between 0 and 1 corresponds to an interpolation between these two covariances

`distribution` can take the values 'gaussian' or 'student'

References

John A. D. Aston, Davide Pigoli, Shahin Tavakoli, "Tests for separability in nonparametric covariance operators of random surfaces", 2015, under revision, http://arxiv.org/abs/1505.02023

Examples

```
Data = generate_surface_data(30, C1, C2, gamma=0)
```

HS_empirical_bootstrap_test

Empirical bootstrap test for separability of covariance structure using Hilbert–Schmidt distance

---

Description

Empirical bootstrap test for separability of covariance structure using Hilbert–Schmidt distance

Usage

```
HS_empirical_bootstrap_test(Data, B = 100, verbose = FALSE)
```

Arguments

`Data`	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.
`B`	number of bootstrap replicates to be used.
`verbose`	logical parameter for printing progress

Value

The p-value of the test.

Details

This function performs the test of separability of the covariance structure for a random surface (introduced in the paper http://arxiv.org/abs/1505.02023), when generated from a Gaussian process. The sample surfaces need to be measured on a common regular grid. The test considers the Hilbert–Schmidt distance between the sample covariance and its separable approximation.

Examples

```
data(SurfacesData)
## Not run: HS_empirical_bootstrap_test(SurfacesData, B = 100)
```

HS_gaussian_bootstrap_test

Gaussian (parametric) bootstrap test for separability of covariance structure using Hilbert–Schmidt distance

---

Description

Gaussian (parametric) bootstrap test for separability of covariance structure using Hilbert–Schmidt distance

Usage

```
HS_gaussian_bootstrap_test(Data, B = 100, verbose = FALSE)
```

Arguments

`Data`	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.
`B`	number of bootstrap replicates to be used.
`verbose`	logical parameter for printing progress

Value

The p-value of the test.

Details

This function performs the test of separability of the covariance structure for a random surface (introduced in the paper http://arxiv.org/abs/1505.02023), when generated from a Gaussian process. The sample surfaces need to be measured on a common regular grid. The test considers the Hilbert–Schmidt distance between the sample covariance and its separable approximation.

Examples

```
data(SurfacesData)
## Not run: HS_gaussian_bootstrap_test(SurfacesData, B = 100)
```

`marginal_covariances` *estimates marginal covariances (e.g. row and column covariances) of bi-dimensional sample*

Description

estimates marginal covariances (e.g. row and column covariances) of bi-dimensional sample

Usage

```
marginal_covariances(Data)
```

Arguments

`Data`	a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.

Value

A list containing the row covariance (`C1`) and column covariance (`C2`)

Examples

```
Data  <-  rmtnorm(30, C1, C2)
marginal.cov  <- marginal_covariances(Data)
```

`projected_differences` *Compute the projection of the rescaled difference between the sample covariance and its separable approximation onto the separable eigenfunctions*

Description

Compute the projection of the rescaled difference between the sample covariance and its separable approximation onto the separable eigenfunctions

Usage

```
projected_differences(Data, l1 = 1, l2 = 1,
  with.asymptotic.variances = TRUE)
```

Arguments

Data
: a (non-empty) `N x d1 x d2` array of data values. The first direction indices the N observations, each consisting of a `d1 x d2` discretization of the surface, so that `Data[i,,]` corresponds to the i-th observed surface.

l1
: number of eigenfunctions to be used in the first (row) dimension for the projection

l2
: number of eigenfunctions to be used in the second (column) dimension for the projection

with.asymptotic.variances
: logical variable; if TRUE, the function outputs the estimate asymptotic variances of the projected differences

Value

A list with

T.N The projected differences

sigma.left The row covariances of `T.N`

sigma.right The column covariances of `T.N`

Details

The function computes the projection of the rescaled difference between the sample covariance and its separable approximation onto the separable eigenfunctions `u_i x v_j : i = 1, ..., l1; j = 1, ..., l2`.

Examples

```
Data  <-  rmtnorm(30, C1, C2)
ans <- projected_differences(Data, l1=1, l2=2)
```

---

renormalize_mtnorm *renormalize a matrix normal random matrix to have iid entries*

Description

renormalize a matrix normal random matrix to have iid entries

Usage

```
renormalize_mtnorm(X, C1, C2, type = "full")
```

Arguments

X
: a matrix normal random matrix with mean zero

C1
: row covariance

C2
: column covariance

type
: the type of renormalization to do. Possible options are 'no', 'diag' or 'full' (see details section).

Value

A matrix with renormalized entries

Details

`type` can take the values

'diag' each entry of `X` is renormalized by its marginal standard deviation

'full' `X` is renormalized by its root inverse covariance

Examples

```
Data  <-  rmtnorm(30, C1, C2)
ans  <-  renormalize_mtnorm(Data[1,,], C1, C2)
```

`rmtnorm` *Generate a sample from a Matrix Gaussian distribution*

Description

Generate a sample from a Matrix Gaussian distribution

Usage

```
rmtnorm(N, C1, C2, M = matrix(0, nrow(C1), nrow(C2)))
```

Arguments

`N`	sample size
`C1`	row covariance
`C2`	column covariance
`M`	mean matrix

Value

A `N x dim(C1)[1] x dim(C2)[1]` array containing the generated data

Examples

```
Data = rmtnorm(30, C1, C2)
```

SurfacesData *A data set of surfaces*

Description

Dataset of 50 surfaces simulated from a Gaussian process with a separable covariance structure. `SurfacesData[i,,]` corresponds to the i-th surface observed on a 32x7 uniform grid.

Usage

```
SurfacesData
```

Format

```
 num [1:50, 1:32, 1:7] -0.229 0.64 0.373 0.346 -0.402 ...
```

Details

This is a `50 x 32 x 7` array.

Examples

```
data(SurfacesData)
image(SurfacesData[1,,]) # color image of the first surface in the dataset
```

Index