Package 'synfd'

June 4, 2020

Type Package

Title Synthesize Dense or Sparse Functional Data/Snippets

Version 0.1.3

Author Zhenhua Lin

Maintainer Zhenhua Lin <stalz@nus.edu.sg>

Description Provides a flexible and simple tool to synthesize regular or irregular functional data and snippets to facilitate research or analysis of functional data.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Imports L1pack, Rdpack

Suggests MASS, knitr, rmarkdown

RdMacros Rdpack

NeedsCompilation no

Repository CRAN

Date/Publication 2020-06-04 14:10:10 UTC

R topics documented:

`centered.process` *Create a Centered Random Process*

Description

Create a Centered Random Process

Usage

```
centered.process(
  name = c("WIENER", "WHITE.NOISE", "KL", "KARHUNEN.LOEVE", "GAUSSIAN"),
  domain = c(0, 1),
  ...
)
```

Arguments

`name`	name of the process.
`domain`	domain where the process is defined.
`...`	other parameters required to define the process:
	dispersion required by Wiener process
	sig required by white noise process
	eigen.values required by process defined via K-L representation
	eigen.functions required by process defined via K-L representation
	distribution the distribution of PC scores, required by process defined via K-L representation

Value

a function hanlde in the form of `X(tObs,n)` which generates `n` independent trajectories observed at `tObs`

Examples

```
X <- centered.process(name='wiener',dispersion=1)
X <- centered.process(name='white.noise',sig=1)
X(regular.grid(50),25)
```

evaluate.basis *Evaluate Orthonormal Basis Functions*

Description

Evaluate Orthonormal Basis Functions

Usage

```
evaluate.basis(
  K,
  m = 51,
  domain = c(0, 1),
  grid = seq(domain[1], domain[2], length.out = m),
  type = c("FOURIER", "COS", "SIN", "LEGENDRE")
)
```

Arguments

K	positive integer specifying the number of basis functions to be included
m	number of equispaced points on domain.
domain	domain on which basis functions are defined.
grid	vector specifying the time points to evaluate the basis functions; if grid is specified, then m is ignored.
type	string for the type of orthogonal basis.

Value

A m by K matrix, where rows index basis functions while columns index points in the grid.

Examples

```
basis <- evaluate.basis(3, type='fourier')
head(basis)
```

gaussian.process *Create a Centered Gaussian Process*

Description

Create a Centered Gaussian Process

Usage

```
gaussian.process(cov = matern)
```

Arguments

cov	a function handle that defines covariance function; see details.

Details

The parameter cov shall take two arguments arg1 and arg2 as input, both are vectors, and cov(arg1,arg2) returns a matrix R such that R(i,j) is the value of the covariance function at (arg1[i],arg2[j]).

Value

a function hanlde in the form of X(tObs,n) which generates n independent trajectories observed at tObs.

Examples

```
X <- gaussian.process()
X(regular.grid(50),25)
```

---

irreg.fd *Sample Irregular Functional Data*

Description

Sample Irregular Functional Data

Usage

```
irreg.fd(
  mu = 0,
  X = wiener.process(),
  n = 100,
  m = 5,
  sig = NULL,
  snr = 5,
  domain = c(0, 1),
  delta = 1
)
```

Arguments

mu	function, scalar or a vector defining the mean function; default value: 0.
X	centered stochastic process defined by a function of the form X(tObs,n) and returning n*length(tObs) matrix, where each row represents observations from a trajectory. Default value: wiener.process().
n	sample size; default value: 100.
m	a vector of sampling rate or scalar of average sampling rate or a function of the form f(n) generating n positive integers; default value: 5.

sig	standard deviation of measurement errors; if NULL then determined by snr.
snr	signal to noise ratio to determine sig; default value: 5.
domain	the domain; default value: c(0,1).
delta	the proportion of the domain to be observed for each trajectory; default value: 1.

Details

The number of observation for each trajectory is randomly generated by rpois(m)+1. For each trajectory, the reference time Oi is uniformly sampled from the interval [domain[1]+delta*L/2,domain[2]-delta*L/2], where L is the length of domain, and the design points for the trajectory is uniformly sampled from the interval [Oi-delta*L/2,Oi+delta*L/2].

Value

a list with the following members

t list of design points sorted in increasing order for each trajectory.

y list of vectors of observations for each trajectory.

and with attributes sig, snr, domain, delta and

y0 n*m matrix of observations without measurement errors.

References

Lin Z, Wang J, Zhong Q (2020). "Basis Expansions for Functional Snippets." *arxiv*.

Examples

```
# Gaussian trajectories with constant mean function 1
Y <- irreg.fd(mu=1, X=gaussian.process(), n=10, m=5)

# trajectories froma a process defined via K-L representation
Y <- irreg.fd(mu=cos, X=kl.process(eigen.functions='FOURIER',distribution='LAPLACE'),n=10, m=5)

# trajectories with specified individual sampling rate
Y <- irreg.fd(mu=1, X=gaussian.process(cov=matern), n=10, m=rpois(10,3)+2)
```

---

kl.process *Create a Process via Karhunen-Loeve Representation*

Description

Create a Process via Karhunen-Loeve Representation

Usage

```
kl.process(
  domain = c(0, 1),
  eigen.values = 1/(2^(1:25)),
  eigen.functions = c("FOURIER", "COS", "SIN", "LEGENDRE"),
  distribution = c("GAUSSIAN", "LAPLACE", "EXPONENTIAL", "GAMMA"),
  corr = NULL
)
```

Arguments

domain	domain of the process.
eigen.values	vector of eigenvalues in the K-L expansion.
eigen.functions	string that specifies the eigenfunctions in the K-L expansion.
distribution	string specifying the distribution of FPC scores.
corr	correlation matrix optionally specifying correlation among the random coefficients; default: NULL.

Value

a function hanlde in the form of `X(tObs,n)` which generates `n` independent trajectories observed at `tObs`.

Examples

```
X <- kl.process()
X(regular.grid(50),25)
```

---

matern *The Matern covariance function*

Description

The Matern covariance function

Usage

```
matern(x, y = NULL, nu = 1, rho = 1, sig = 1)
```

Arguments

x	a vector
y	If NULL, then y=x.
nu	positive number specifying smoothness of the function.
rho	scale parameter
sig	magnitude parameter

Details

The precise definition of Matern covariance function can be found in see https://en.wikipedia.org/wiki/Matern_covariance_fu

Value

matrix of dimension length(x)*length(y), where the (i,j) entry of the matrix is the Matern covariance evaluated at the point (X(i),Y(j)).

---

plot.dense.fd *Plot Regular Functional Data*

Description

Plot Regular Functional Data

Usage

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

Arguments

x	the data object generated from reg.fd.
...	other parameters passed to matplot.

Value

a plot of the dataset x.

---

plot.sparse.fd *Plot Irregular Functional Data*

Description

Plot Irregular Functional Data

Usage

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

Arguments

x	the object generated by irreg.fd.
...	other parameters passed to plot and lines.

Value

a plot of the dataset x.

---

reg.fd *Sample Regular Functional Data*

Description

Sample Regular Functional Data

Usage

```
reg.fd(
  mu = 0,
  X = wiener.process(),
  n = 100,
  m = 51,
  sig = NULL,
  snr = 5,
  domain = c(0, 1),
  grid = seq(domain[1], domain[2], length.out = m)
)
```

Arguments

mu	function, scalar or a vector defining the mean function; default value: 0.
X	centered stochastic process defined by a function of the form X(tObs,n) and returning n*length(tObs) matrix, where each row represents observations from a trajectory. Default value: wiener.process().
n	sample size; default value: 100.
m	sampling rate; ignored if grid is specified; default value: 51.
sig	standard deviation of measurement errors; if NULL then determined by snr.
snr	signal to noise ratio to determine sig; default value: 5.
domain	the domain; default value: c(0,1).
grid	vector of design points; default value: NULL.

Value

a list with the following members

t a vector of design points sorted in increasing order.

y n*m matrix; each row represents observations from a trajectory.

and with attributes sig, snr, domain, grid and

y0 n*m matrix of observations without measurement errors.

Examples

```
Y <- reg.fd()
Y <- reg.fd(mu=1, X=gaussian.process(), n=10, m=20)
Y <- reg.fd(mu=cos, X=kl.process(),n=100, m=20)
Y <- reg.fd(mu=cos, X=kl.process(distribution='EXPONENTIAL'),n=100, m=20)
```

`regular.grid` *Generate a Regular Grid*

Description

Generate a Regular Grid

Usage

```
regular.grid(M = 100, domain = c(0, 1), h = 1/(2 * M))
```

Arguments

`M`	the number of points
`domain`	the interval
`h`	margin at boundaries

Value

equally spaced `M` points in `domain`.

`rfd` *Sample Functional Data*

Description

Sample Functional Data

Usage

```
rfd(
  mu = 0,
  X = wiener.process(),
  n = 100,
  m = 5,
  sig = NULL,
  snr = 5,
  domain = c(0, 1),
  delta = 1,
  grid = seq(domain[1], domain[2], length.out = m),
  type = "irregular"
)
```

Arguments

mu	function, scalar or a vector defining the mean function; default value: 0.
X	centered stochastic process defined by a function of the form X(tObs,n) and returning n*length(tObs) matrix, where each row represents observations from a trajectory. Default value: wiener.process().
n	sample size; default value: 100.
m	average sampling rate; default value: 5.
sig	standard deviation of measurement errors; if NULL then determined by snr.
snr	signal to noise ratio to determine sig; default value: 5.
domain	the domain; default value: c(0,1).
delta	the proportion of the domain to be observed for each trajectory; only required when type="irregular"; default value: 1.
grid	vector of design points; only required when type="regular". default value: NULL.
type	the data type, either "regular" or "irregular"; default value: "irregular".

Details

This is a unified interface for reg.fd and irreg.fd; see the manual for these functions for details.

Value

a list with the following members

t design points sorted in increasing order for each trajectory; if type="irregular" then a list; otherwise a vector.

y observations for each trajectory; if type="irregular" then a list; otherwise, a matrix.

and with attributes sig, snr, domain, delta, grid and

y0 the measurement-error-free counterpart of y.

References

Lin Z, Wang J, Zhong Q (2020). "Basis Expansions for Functional Snippets." *arxiv*.

Examples

```
# irregularly observed Gaussian trajectories with constant mean function 1
Y <- rfd(mu=1, X=gaussian.process(), n=10, m=5)

# regularly observed trajectories with a K-L representation
Y <- rfd(X=kl.process(eigen.functions='FOURIER',distribution='LAPLACE'), type='regular')
```

white.noise *Create a White Noise Process*

Description

Create a White Noise Process

Usage

```
white.noise(sig = 1)
```

Arguments

sig	sigma parameter of the white noise process

Value

a function hanlde in the form of `X(tObs,n)` which generates `n` independent trajectories observed at `tObs`.

Examples

```
X <- white.noise()
X(regular.grid(50),25)
```

wiener.process *Create a Wiener Process*

Description

Create a Wiener Process

Usage

```
wiener.process(dispersion = 1)
```

Arguments

dispersion	the dispersion parameter of the Wiener process.

Value

a function hanlde in the form of `X(tObs,n)` which generates `n` independent trajectories observed at `tObs`.

Examples

```
X <- wiener.process()
X(regular.grid(50),25)
```

Index