Package ‘fctbases’

October 13, 2022

Type Package
Title Functional Bases
Version 1.1.1
Date 2022-05-17
Author Niels Olsen
Maintainer Niels Olsen <nalo@dtu.dk>
Description Easy-to-use, very fast implementation of various functional bases. Easily used together with other packages.
A functional basis is a collection of basis functions \[ \phi_1, ..., \phi_n \] that can represent a smooth function, i.e. \( f(t) = \sum c_k \phi_k(t) \).
First- and second-order derivatives are also included. These are the mathematically correct ones, no approximations applied.
As of version 1.1, this package includes B-splines, Fourier bases and polynomials.
URL https://github.com/naolsen/fctbases
License GPL-3
Imports Rcpp (>= 0.12.19)
Suggests knitr, rmarkdown, microbenchmark
LinkingTo Rcpp, RcppArmadillo
NeedsCompilation yes
Repository CRAN
Date/Publication 2022-05-17 17:10:02 UTC

R topics documented:

  fctbases-package ........................................ 2
  Functional basis function ............................. 2
  make.bspline.basis .................................... 3
  make.fourier.basis ..................................... 4
  make.pol.basis ......................................... 5
  make.std.bspline.basis ................................ 5
  object.info ............................................. 6
fctbases-package

fctbases: Functional bases

Description

fctbases is a fast and easy implementation of functional bases in R. Simply initialize the desired basis, which returns function of class fctbasis.

Details

Internally, functions are stored as C++ objects, which are masked by the package. The package maintains the bookkeeping of fctbasis objects. Parameters are validated at initialization which also reduces some of the overhead. fctbases objects cannot be saved across sessions and must be re-initialized.

Derivatives are provided. These are the mathematically correct ones and are as fast as the non-derivatives.

See Also

Functional basis function

Functional basis function

Description

A fctbases object is a function of class fctbasis which takes three arguments \((t, x, \text{deriv})\)

Arguments

- **t**
  - time points
- **x**
  - vector or matrix of coefficients (optional)
- **deriv**
  - Should the derivative be used and which order? Defaults to \text{FALSE}

Details

If \text{deriv} is zero or \text{FALSE}, the function itself is evaluated. If \text{deriv} is one or \text{TRUE}, the first derivative is evaluated. If \text{deriv} is two, the second derivative is evaluated.

The dimension of \(x\) must match the number of basis functions.
make.bspline.basis

Description
Make B-spline basis

Usage
make.bspline.basis(knots, order = 4)

Arguments
knots Knots of the basis, including endpoints
order Spline order. Defaults to 4.

Value
Function of class "fctbasis"

See Also
Functional basis function, make.std.bspline.basis

make.bspline.basis

Examples
## Create basis (here a b spline)
bf <- make.bspline.basis(knots = 0:12/12)

## Use a functional basis

bf(0.2)
tt <- seq(0,1, length = 50)
bf(tt) ## evaluates bf in tt
bf(tt, deriv = TRUE) ## evaluates derivative of bf in tt

## Apply bf to some coefficients
set.seed(661)
x <- runif(15)
bf(tt, x) ## Evaluate bf in tt with coefficients x.

bf(0.2, deriv = 2) ## Second derivative.
bf(0.2, x, deriv = 2) ## Second derivative with coefficients x.
## Examples

### B-spline with equidistant knots with 13 basis function
```r
bf <- make.bspline.basis(knots = 0:10, order = 4)
```

### B-spline of order 2 (i.e. a linear approximation) with some uneven knots
```r
bf <- make.bspline.basis(knots = c(-1.3, 0, 0.5, 0.7, 1.1), order = 2)
```

### Description

Make fourier basis

### Usage

```r
make.fourier.basis(range, order, use.trig.id = FALSE)
```

### Arguments

- `range`  
  Left and right end points.
- `order`  
  Order of harmonics
- `use.trig.id`  
  Use trigonometrical identities with this function?

### Details

The number of basis elements (degrees of freedom) is 2 * order + 1.  
The basis functions are ordered [1, sin(t), cos(t), sin(2t), cos(2t), ...]  
Using trigonometrical identities is faster, but introduces (negligible) round-off errors.

### Value

Function of class "fctbasis"

### See Also

Functional basis function

### Examples

```r
## A fourier basis with period 1 and 11 basis functions.
bf <- make.fourier.basis(c(0,1), order = 5)
```
**Description**

Make polynomial basis

**Usage**

```r
make.pol.basis(order)
```

**Arguments**

- `order` Order of polynomial (= degree + 1)

**Details**

The polynomial basis is ordered \([1, t, t^2, t^3, ..., t^n]\)

**Value**

Function of class "fctbasis"

**See Also**

- Functional basis function

**Examples**

```r
## A four-degree polynomial
mypol <- make.pol.basis(order = 5)
```

---

**make.std.bspline.basis**

*Standard* B-spline basis

**Description**

This initializes a bspline of order 4 with uniformly placed knots. df = intervals + 3.

**Usage**

```r
make.std.bspline.basis(range = c(0, 1), intervals)
```
Arguments

- range: End points of spline
- intervals: Number of intervals

Details

`make.std.bspline.basis` uses a different implementation than `make.bspline.basis`, but is not faster in all uses.

Value

function

See Also

- `Functional basis function`, `make.bspline.basis`

Examples

```r
cat("## 16 equidistant knots between 0 and 2 (both included)"
bf <- make.std.bspline.basis(range = c(0,2), intervals = 15)
```

Description

This function returns details about a functional basis.

Usage

`object.info(fctbasis)`

Arguments

- `fctbasis`: object of class `fctbasis`

Value

A named list including no. of basis, type of basis, and possibly additional information.
Index

fctbases (fctbases-package), 2
fctbases-package, 2
Functional basis function, 2, 2, 3–6

make.bspline.basis, 3, 6
make.fourier.basis, 4
make.pol.basis, 5
make.std.bspline.basis, 3, 5

object.info, 6