Package ‘covafillr’

October 12, 2022

Title  Local Polynomial Regression of State Dependent Covariates in State-Space Models
Version  0.4.4
Date  2019-11-22
Maintainer  Christoffer Moesgaard Albertsen <cmoe@aqua.dtu.dk>
Description  Facilitates local polynomial regression for state dependent covariates in state-space models. The functionality can also be used from 'C++' based model builder tools such as 'Rcpp'/'inline', 'TMB', or 'JAGS'.

BugReports  https://github.com/calbertsen/covafillr/issues
URL  https://github.com/calbertsen/covafillr
Depends  R (>= 3.0.0)
Imports  methods, stats, Rcpp (>= 0.11.0)
LinkingTo  RcppEigen
Suggests  TMB, rjags, inline, ggplot2
License  BSD_2_clause + file LICENSE
LazyData  true
Biarch  true
NeedsCompilation  yes
RoxygenNote  7.0.0
Author  Christoffer Moesgaard Albertsen [aut, cre]
  (<https://orcid.org/0000-0003-0088-4363>)
Repository  CRAN
Date/Publication  2019-11-22 10:40:09 UTC

R topics documented:
covafill-class ................................................................. 2
covatree-class ................................................................. 3
cxxFlags ................................................................. 4
covafill-class

A Reference Class for Local Polynomial Regression with covafill.

Description

A Reference Class for Local Polynomial Regression with covafill.

Fields

- ptr: External pointer to the covafill C++ object

Methods

- getBandwith(): Get the bandwidth.
- getDegree(): Get the polynomial degree.
- getDim(): Get the dimension of the coordinates.
- initialize(coord, obs, h = suggestBandwith(coord, p), p = 3L, ...): Method to initialize the covafill. coord is a matrix of coordinates, obs is a vector of corresponding observations, h is a vector of bandwidths, and p is the polynomial degree.
- predict(coord, se.fit = FALSE): Predict function value and derivatives with local polynomial regression at coord. If se.fit=TRUE a list is returned with estimates and their standard deviations.
- residuals(excludeRadius): Get 'leave-neighborhood-out' residuals, i.e. local polynomial regression predictions excluding points within excludeRadius subtracted from the observation.
- setBandwith(h): Set the bandwidth to h.

Examples

```r
getRefClass('covafill')
fn <- function(x) x ^ 4 - x ^ 2
x <- runif(2000,-3,3)
y <- fn(x) + rnorm(2000,0,0.1)
cf <- covafill(coord = x,obs = y,p = 5L)
cf$getDim()
cf$getDegree()
cf$getBandwith()
x0 <- seq(-1,1,0.1)
y0 <- cf$predict(x0)
par(mfrow=c(3,1))
plot(x0,y0[,1], main = "Function")
```
covatree-class

A Reference Class for Search Tree Approximated Local Polynomial Regression with covatree.

Description

A Reference Class for Search Tree Approximated Local Polynomial Regression with covatree.

Fields

ptr External pointer to the covatree C++ object

Methods

getDim() Get the dimension of the coordinates.

initialize(coord, obs, h = suggestBandwith(coord, p), p = 3L, minLeft = length(obs)/10, ...) Method to initialize the covafill. coord is a matrix of coordinates, obs is a vector of corresponding observations, h is a vector of bandwidths, p is the polynomial degree, and minLeft is the minimum number of observations that will create a sub tree.

predict(coord) Predict function value and first order derivatives with search tree approximated local polynomial regression at coord.

Examples

getRefClass('covatree')
fn <- function(x) x ^ 4 - x ^ 2
x <- runif(2000,-3,3)
y <- fn(x) + rnorm(2000,0,0.1)
ct <- covatree(coord = x,obs = y,p = 5L, minLeft = 50)
ct$getDim()
x0 <- seq(-1,1,0.1)
y0 <- ct$predict(x0)
par(mfrow=c(2,1))
plot(x0,y0[,1], main = "Function")
lines(x0,fn(x0))
plot(x0, y0[,2], main = "First derivative")
lines(x0, 4 * x0 ^ 3 - 2 * x0)
**cxxFlags**  
*CXXFLAGS to compile with covafill*

**Description**
Get CXXFLAGS to compile with covafill

**Usage**
cxxFlags()

**Value**
Returns a string with the CXXFLAGS needed to compile C++ code using covafill.

**Author(s)**
Christoffer Moesgaard Albertsen

**See Also**
compile

**Examples**
```r
## Not run:
if(require("TMB")){
  f <- system.file("examples","tmbtest","tmbtest.cpp", package='covafillr')
  TMB::compile(f,CXXFLAGS = cxxFlags())
}
## End(Not run)
```

**kde**  
*Kernel Density Estimation*

**Description**
Wrapper for the covafill reference class to do kernel density estimation.

**Usage**
kde(X, bw = suggestBandwith(X, -1), npred = 100, from = min(X), to = max(X))
Arguments

- **X**: A numeric matrix or vector of data coordinates
- **bw**: Bandwidth used
- **npred**: Number of coordinate wise equally spaced points at which the density is to be estimated. The numbers are repeated if the length is less than the dimension of the coordinates.
- **from**: Coordinate wise lower bound of points at which the density is to be estimated. The numbers are repeated if the length is less than the dimension of the coordinates.
- **to**: Coordinate wise upper bound of points at which the density is to be estimated. The numbers are repeated if the length is less than the dimension of the coordinates.

Value

A list of coordinates and corresponding density estimates

Author(s)

Christoffer Moesgaard Albertsen

Description

Calls rjags::load.module with appropriate arguments to load the covafillr module.

Usage

```r
loadJAGSModule()
```

Value

Nothing

Author(s)

Christoffer Moesgaard Albertsen

See Also

- `load.module`

Examples

```r
if(require("rjags") & covafillr:::.installed_with_jags) {
  loadJAGSModule()
}
```
stat_covafill

Add a covafill smoother to an (x,y) plot

Description

As an extension to the ggplot2 package, the function adds a covafill fit to an (x,y) plot. The fit is predicted to points on the interval range(x).

Usage

stat_covafill(
  mapping = NULL,
  data = NULL,
  geom = "smooth",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  n = 50,
  bandwith = NULL,
  polyDegree = 3L,
  level = 0.95,
  se = TRUE,
  ...
)

Arguments

- mapping: Set of mappings created by 'aes' from the ggplot2 package. The same as ggplot2::stat_smooth.
- data: The data to be displayed in this layer. The same as ggplot2::stat_smooth.
- geom: The same as ggplot2::stat_smooth.
- position: Position adjustments. The same as ggplot2::stat_smooth.
- na.rm: Not used
- show.legend: Should this legend be displayed? The same as ggplot2::stat_smooth.
- inherit.aes: The same as ggplot2::stat_smooth.
- n: Number of points to do prediction on.
- bandwith: Bandwidth used in covafill. Uses suggestBandwith by default.
- polyDegree: Polynomial degree to use in covafill.
- level: Level of confidence interval to use.
- se: Should confidence intervals be displayed?
- ...: Other arguments passed to layer.
suggestBandwith

Value
A ggplot2 layer.

Author(s)
Christoffer Moesgaard Albertsen

See Also
stat_smooth

---

**suggestBandwith**  
*Suggest bandwith for local polynomial regression*

Description
The bandwith is suggested coordinate wise to be

\[
0.9\sqrt{5} \min \left( \frac{sd(x)}{IQR(x)}, \frac{IQR(x)}{1.349} \right) n^{-\frac{1}{p+1}} (p + 1)
\]

where \( p \) is the polynomial degree used and \( n \) is the number of coordinate points.

Usage
suggestBandwith(X, p)

Arguments

- **X**: A numeric matrix or vector of data coordinates
- **p**: Polynomial degree to base the suggestion on

Value
a vector or scalar of suggested bandwiths

Author(s)
Christoffer Moesgaard Albertsen
Index

compile, 4

covafill(covafill-class), 2
covafill-class, 2
covatree(covatree-class), 3
covatree-class, 3
cxxFlags, 4

dkde, 4

load.module, 5
loadJAGSModule, 5

stat_covafill, 6
stat_smooth, 7
suggestBandwith, 7