Package ‘nonsmooth’

July 3, 2024

Title Nonparametric Methods for Smoothing Nonsmooth Data

Version 1.0.0

Imports stats, np, graphics, reshape2

Maintainer John R.J. Thompson <john.thompson@ubc.ca>

Description Nonparametric methods for smoothing regression function data with change-points, utilizing range kernels for iterative and anisotropic smoothing methods. For further details, see the paper by John R.J. Thompson (2024) <doi:10.1080/02664763.2024.2352759>.

License CC BY 4.0

URL https://github.com/jrjthompson/R-package-nonsmooth/

BugReports https://github.com/jrjthompson/R-package-nonsmooth/issues

Depends R (>= 3.5.0)

Repository CRAN

LazyData true

LazyDataCompression xz

NeedsCompilation no

Author John R.J. Thompson [aut, cre] (<https://orcid.org/0000-0002-6303-449X>)

Date/Publication 2024-07-03 16:00:09 UTC

Contents

alc *Iterative anisotropic local constant smoothing*

Description

This function implements the method in Thompson, J.R.J. (2024) for iterative smoothing of change-point data that utilizes oversmoothed estimates of the underlying data generating process to inform re-smoothing. The function calculates a local constant estimator $\tilde{g}(X)$ of $Y = g(X) + \epsilon$, and then utilizes $\tilde{g}(x)$ in the range kernel of another local constant estimator. This process is iterated for a specified number of resmooth iterations.

Usage

```
alc(X,Y,bw.fixed.value=NULL,resmooths=1,...)
```

Arguments

`X`	numeric matrix of p columns of the observations for continuous explanatory variables.
`Y`	numeric vector of the continuous response variable.
`bw.fixed.value`	numeric value for the bandwidth of the range kernel. Setting this value sets the iterative smoothing bandwidths to be the local constant estimator bandwidths for domain kernels and the set value for the range kernel. Default is `NULL`, which isan optimal bandwidth selection procedure during each re-smooth through the `npreg` function.
`resmooths`	integer number of resmooth iterations. Default is 1 resmooth, which is a suggested starting point for iterative smoothing.
`...`	additional specifications for `np` smoothing, such as optimal bandwidth selection procedure, kernel type, regression estimator, and so on. See `npreg` function for more details and defaults.

Value

The code here returns a `npregression` object of the iteratively smoothed estimator. For more details, see the `npreg` function in the `np` package.

References

Thompson, J.R.J. (2024) “Iterative Smoothing for Change-point Regression Function Estimation”, *Journal of Applied Statistics*, 1-25. <doi:10.1080/02664763.2024.2352759>

See Also

`npreg`, `npregbw`

Examples

```
library(np)
options(np.messages=FALSE)

## 1D Simulated change-point data
changepoint.data <- changepoint.sim1D(500)

## Isotropic local constant model
bw.lc <- npregbw(Y~X,data=changepoint.data)
model.lc <- npreg(bw.lc)

## Anisotropic local constant model with one resmooth iteration
model.alc <- alc(changepoint.data$X,changepoint.data$Y)

## Plot isotropic and anistropic smoothers
plot(changepoint.data$X,changepoint.data$Y,xlab = "X",ylab = "Y",
     pch=1,col="grey",las=1)
lines(changepoint.data$X,model.lc$mean,col="blue",lty=1)
lines(changepoint.data$X,model.alc$mean,col="red",lty=1)

## 2D Simulated image change-point data
## This simulation and estimation can take up to 5 minutes
library(reshape2)
changepoint.data <- changepoint.sim2D(data.dim=c(50,50))
image(changepoint.data)

## Melt the 2D image data for model estimation
changepoint.data.melt <- melt(id.var=1:nrow(changepoint.data), changepoint.data)

## Isotropic local constant model
bw.lc <- npregbw(xdat=changepoint.data.melt[,1:2],ydat=changepoint.data.melt[,3])
model.lc <- npreg(bw.lc)

image(1:dim(changepoint.data)[1], 1:dim(changepoint.data)[2],
      matrix(model.lc$mean, nrow=dim(changepoint.data)[1], byrow=FALSE))

## Anisotropic local constant model with one resmooth iteration and
## and fixed range kernel bandwidth
model.alc <- alc(changepoint.data.melt[,1:2],changepoint.data.melt[,3],bw.fixed.value=10)

image(1:dim(changepoint.data)[1], 1:dim(changepoint.data)[2],
      matrix(model.alc$mean, nrow=dim(changepoint.data)[1], byrow=FALSE))

## 2D real fire spread change-point data
  data("fireData")
  changepoint.data <- fireData[,,1,20]

  ## Plot with pixel locations
  image(1:dim(changepoint.data)[1], 1:dim(changepoint.data)[2],
      matrix(changepoint.data, nrow=dim(changepoint.data)[1], byrow=FALSE))

  ## Melt the 2D image data for model estimation
```

```
  changepoint.data.melt <- melt(id.var=1:nrow(changepoint.data), changepoint.data)

  ## Isotropic local constant model
  bw.lc <- npregbw(xdat=changepoint.data.melt[,1:2],ydat=changepoint.data.melt[,3])
  model.lc <- npreg(bw.lc)

  image(1:dim(changepoint.data)[1], 1:dim(changepoint.data)[2],
      matrix(model.lc$mean, nrow=dim(changepoint.data)[1], byrow=FALSE))

  ## Anisotropic local constant model with one resmooth iteration and
  ## and fixed range kernel bandwidth
 model.alc <- alc(changepoint.data.melt[,1:2],changepoint.data.melt[,3],bw.fixed.value=10)

  image(1:dim(changepoint.data)[1], 1:dim(changepoint.data)[2],
      matrix(model.alc$mean, nrow=dim(changepoint.data)[1], byrow=FALSE))

options(np.messages=TRUE)
```

---

changepoint.sim1D *Simulated change-point data for one-dimension*

Description

This function simulates one-dimension change-point data for three data types, and one smooth data type for testing change-point regression estimators.

Usage

```
changepoint.sim1D(n,sigma=1,data.type = "continuousWithJump")
```

Arguments

n	Integer value for sample size.
sigma	Numeric value of standard deviation.
data.type	Character value for different data types. The options for change-point data are: constant functions seperated by two jumps ("uniformJump"), linear functions seperated by two change-points in the first derivative ("gradualJump"), and nonlinear data with a jump ("continuousWithJump"). A smooth continuous function with no change-points ("continuous") is the same functions as "continuousWithJump" but without the change-point. See Thompson, J.R.J. (2024) for more details on these data types.

Value

This function produces a data.frame, consisting of the simulated data and the data generating process.

X	Numeric vector of explanatory data
Y	Numeric vector of response data
oracle	Numeric vector of the data generating process for Y

References

Thompson, J.R.J. (2024) “Iterative Smoothing for Change-point Regression Function Estimation”, *Journal of Applied Statistics*, 1-25. <doi:10.1080/02664763.2024.2352759>

Examples

```
## 1D continuous data of nonlinear functions with a jump change-point
changepoint.data <- changepoint.sim1D(100)
plot(changepoint.data$X,changepoint.data$Y,xlab = "X",ylab = "Y",pch=1,col="grey",las=1)
lines(changepoint.data$X,changepoint.data$oracle,col="red",lty=1)

## 1D continuous data of constant functions with two jump change-points
changepoint.data <- changepoint.sim1D(100,data.type="uniformJump")
plot(changepoint.data$X,changepoint.data$Y,xlab = "X",ylab = "Y",pch=1,col="grey",las=1)
lines(changepoint.data$X,changepoint.data$oracle,col="red",lty=1)

## 1D continuous data of linear functions with two derivative change-points
changepoint.data <- changepoint.sim1D(100,data.type="gradualJump")
plot(changepoint.data$X,changepoint.data$Y,xlab = "X",ylab = "Y",pch=1,col="grey",las=1)
lines(changepoint.data$X,changepoint.data$oracle,col="red",lty=1)

## 1D continuous data of a nonlinear continuous function
changepoint.data <- changepoint.sim1D(100,data.type="continuous")
plot(changepoint.data$X,changepoint.data$Y,xlab = "X",ylab = "Y",pch=1,col="grey",las=1)
lines(changepoint.data$X,changepoint.data$oracle,col="red",lty=1)
```

`changepoint.sim2D` *Simulated change-point data for two-dimensions*

Description

This function simulates circular change-point data with Gaussian noise.

Usage

```
changepoint.sim2D(data.dim = c(100,100),sigma = 20,radius=NULL,cbase=80,ctop=130)
```

Arguments

`data.dim`	Vector of two integers for the size of the two-dimensional dataset. The dimensions are suggested to be the same. However, for uneven dimensions, the first value must be larger. The default is an image of 100 by 100 pixels.
`sigma`	Numeric value of standard deviation.
`radius`	Numeric value of the radius of the inner disk before the change-point. The radius cannot be larger than one-half of either dimension in `data.dim`. Defaults to one-quarter of the first dimension of `data.dim`.
`cbase`	Numeric value for the disk that radiates out from the approximate center of the dataset.

`ctop`	Numeric value after the circular change-point, seperating the disk and the outer region.

Value

This function produces a matrix of integer values of the same dimensions as `data.dim`.

References

Thompson, J.R.J. (2024) "Iterative Smoothing for Change-point Regression Function Estimation", *Journal of Applied Statistics*, 1-25. <doi:10.1080/02664763.2024.2352759>

Examples

```
## Simulate 2D data and plot it
library(reshape2)
changepoint.data <- changepoint.sim2D()
image(1:nrow(changepoint.data), 1:ncol(changepoint.data),
      matrix(changepoint.data, nrow=nrow(changepoint.data), byrow=FALSE))
```

`fireData` *Wax paper fire smouldering experiment conducted in a fume hood*

Description

The fire spread data consists of 45 segmented RGB images from a fire smouldering experiment of wax paper. The data were measured using a digital camera at a birds-eye view above the experiment. The data is segmented 1 frame per second.

Usage

```
data("fireData")
```

Format

The movie of the fire spread is a data frame with four dimensions. The first and second dimensions of the data frame are the pixel coordinates of one image. The third dimension is the RGB channel, with the red channel (1), blue channel (2), and green channel (3). The fourth dimension is time, starting at ignition (time point 1), and then each RGB image is separated by one second for a total of 45 seconds.

Source

John R.J. Thompson

References

Thompson, J.R.J., Wang, X.J., & Braun, W.J. (2020) "A mouse model for studying fire spread rates using experimental micro-fires", *Journal of Environmental Statistics*, 9(1), 1-19. <[https://www.jenvstat.org/v09/i06]https://w

Wang, X.J., Thompson, J.R.J., Braun, W.J., & Woolford, D.G. (2019) "Fitting a stochastic fire spread model to data." *Advances in Statistical Climatology, Meteorology and Oceanography*, 5(1), 57-66. <[https://ascmo.copernicus.org/articles/5/57/2019/]https://ascmo.copernicus.org/articles/5/57/2019/>

Examples

```
## Example - viewing a fire spread experiment that contains change-points

data("fireData")

## Plot the red channel at 10 seconds as a 2d image

image(1:dim(fireData)[1], 1:dim(fireData)[2],
      matrix(fireData[,,1,10], nrow=dim(fireData)[1], byrow=FALSE))
```

---

nonsmooth — *Nonparametric methods for smoothing nonsmooth data*

Description

This package provides nonparametric methods for smoothing nonsmooth data. Change-point data is the intended application, with a focus on those with jumps in the regression function. Descriptions of the implementation of these methods can be found in Thompson, J.R.J. (2024).

Details

This package contains two additional functions for simulated one-D and two-D change-point data. This package also contains a real fire spread dataset from a micro-fire experiment. This data can be viewed as time dependent two-dimensional change-point data. The boundaries between fuel, burning and burn-out regions are seperated by two change-point curves. More information on experimentation and data can befound in Thompson, Wang, and Braun (2020) and Wang, Thompson, and Braun (2019).

Author(s)

John R.J. Thompson <john.thompson@ubc.ca>

Maintainer: John R.J. Thompson <john.thompson@ubc.ca>

I would like to acknowledge funding support from the University of British Columbia Aspire Fund (UBC:www.ok.ubc.ca/).

References

Thompson, J.R.J. (2024) “Iterative Smoothing for Change-point Regression Function Estimation”, *Journal of Applied Statistics*, 1-25. <doi:10.1080/02664763.2024.2352759>

Thompson, J.R.J., Wang, X.J., & Braun, W.J. (2020) “A mouse model for studying fire spread rates using experimental micro-fires”, *Journal of Environmental Statistics*, 9(1), 1-19. <[https://www.jenvstat.org/v09/i06]https://w

Wang, X.J., Thompson, J.R.J., Braun, W.J., & Woolford, D.G. (2019) “Fitting a stochastic fire spread model to data.” *Advances in Statistical Climatology, Meteorology and Oceanography*, 5(1), 57-66. <[https://ascmo.copernicus.org/articles/5/57/2019/]https://ascmo.copernicus.org/articles/5/57/2019/>

Index