Package: features (via r-universe)

November 2, 2024

Type Package

Title Feature Extraction for Discretely-Sampled Functional Data

Description Discretely-sampled function is first smoothed. Features of the smoothed function are then extracted. Some of the key features include mean value, first and second derivatives, critical points (i.e. local maxima and minima), curvature of cunction at critical points, wiggliness of the function, noise in data, and outliers in data.

Depends lokern

Version 2015.12-1

Date 2015-12-01

Author Ravi Varadhan, Johns Hopkins University, and MKG Subramaniam, AT&T Reserach Labs.

Maintainer Ravi Varadhan <rvaradhan@jhmi.edu>

URL http://www.jhsph.edu/agingandhealth/People/Faculty_personal_pages/Varadhan.html

License GPL (>= 2)

LazyLoad yes

Date/Publication 2015-12-02 00:01:49

NeedsCompilation no

Repository https://rvaradhan.r-universe.dev

RemoteUrl https://github.com/cran/features

RemoteRef HEAD

RemoteSha 3e4e990c1d1127ed42edc5c2051a411fa67f90a7

Contents

`features` *Estimate features of a discretely-sampled functional data.*

Description

Discretely-sampled function is first smoothed. Features of the smoothed function are then extracted. Some of the key features include mean value, first and second derivatives, critical points (i.e. local maxima and minima), curvature of function at critical points, wiggliness of the function, noise in data, and outliers in data.

Usage

```
features(x, y, smoother=c("glkerns", "smooth.spline"), fits.return=TRUE,
 control = list(), ...)
```

Arguments

`x`	a vector of independent variable (e.g. time) at which the function is sampled.
`y`	a vector of response values to be smoothed
`smoother`	a character string specifying the name of the smoothing algorithm; currently only 2 smoothers are implemented: smoothing spline and a kernel smoother with a global plug-in bandwidth
`fits.return`	a logical variable specifying whether or not to return the smoothed function and its first 2 derivatives. Default is TRUE. It must be TRUE for plotting the results.
`control`	A list of control parameters. See *Details* for more information.
`...`	Additional arguments to be passed to the smoothers. An important optional parameter is the `bandwidth` that controls the smoothness of the fit. See the help for 'glkerns' for mode details.

Details

Argument `control` is a list specifing any changes to default values of control parameters. Note that the names of these must be specified completely. Partial matching will not work.

Default values of `control` are: ctrl <- list(npts=100, c.outlier=3, decim.out=2)

`npts`: an integer. Number of points to use in estimating smoothed function and for computing features

`c.outlier`: a constant denoting number of standard deviations away from smooth fit for determining whether a point is an outlier. Default is 3.

`decim.out`: number of decimals to display in the features output.

Value

A list with a number of extracted features of the underlying smooth function:

f	: a numeric vector containing 10 basic features of the function: mean, min, max, std.dev, noise, signa-to-noise ratio, minimum and maximum of first derivative, wiggliness, and number of critical points
cpts	: Locations of the critical points (e.g. points where the first derivative is zero)
curvature	: Value of second derivative at the critical points
outliers	: Locations of outlying data points
fits	an attribute of the object returned by `features` when `fits.return=TRUE`. A list with 4 vectors: abscissae, smoothed function, first derivative, and second derivative

See Also

`fget`, `plot.features`, `plot`, `glkerns`, `smooth.spline`

Examples

```
# Estimating the smooth and the derivatives of a noisy and discretely sampled function.
n <- 200
x <- sort(runif(n))
y <- exp(-0.2 * sin(2*pi*x)) + rnorm(n, sd=0.05)

ans <- features(x, y)

fget(ans)
```

`fget` *Utility function to extract features from an object of* `class "features"`.

Description

This is a utility function that operates only on objects of `class "features"` that are returned by the function `features()`. Some of the key extracted features include mean value, first and second derivatives, critical points (i.e. local maxima and minima), curvature of function at critical points, wiggliness of the function, noise in data, and outliers in data.

Usage

```
fget(x)
```

Arguments

x	An object of `class "features"`

Value

A list with a number of extracted features of the underlying smooth function:

`f`	: a numeric vector containing 10 basic features of the function (in order): mean, min, max, std.dev, noise, signal-to-noise ratio, minimum and maximum of first derivative, wiggliness, and number of critical points
`cpts`	: Locations of the critical points (e.g. points where the first derivative is zero)
`curvature`	: Value of second derivative at the critical points
`outliers`	: Locations of outlying data points

See Also

`features`, `plot.features`, `plot`

Examples

```
# Estimating the smooth and the derivatives of a noisy and discretely sampled function.
n <- 200
x <- sort(runif(n))
y <- exp(-0.2 * sin(2*pi*x)) + rnorm(n, sd=0.05)

ans <- features(x, y)
fget(ans)
```

---

plot *Plotting the smoothed fucntion and its derivatives.*

Description

Plots of the smoothed function and its first 2 derivatives.

Usage

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

Arguments

`x`	an object of `class(features)` that is returned by the `features()` function.
`...`	Additional arguments to be passed to the `plot()` function

Value

A plot is shown in a 2 x 2 layout, where the top 2 frames are the same and they depict the raw data and the smoothed function. The bottom left panel shows the smoothed first derivative, and the bottom right panel depicts the smoothed second derivative.

Examples

```
# Estimating the smooth and the derivatives of a noisy and discretely sampled function.
n <- 200
x <- sort(runif(n))
y <- exp(-0.2 * sin(2*pi*x)) + rnorm(n, sd=0.05)

ans <- features(x, y)
plot(ans)

ans.sm <- features(x, y, smoother="smooth.spline")
plot(ans.sm)
```

Index