Package 'CHsharp'

October 16, 2015

Version 0.4

Title Choi and Hall Style Data Sharpening

Author Douglas Woolford and John Braun

Maintainer John Braun <braun@stats.uwo.ca>

Description Functions for use in perturbing data prior to use of nonparametric smoothers and clustering.

LazyLoad true

LazyData true

Depends scatterplot3d, KernSmooth

License GPL-3

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-10-16 00:51:37

R topics documented:

d — *A really neat data set*

Description

This revolutionizes the scientific community's worldview.

Usage

```
data(d)
```

Author(s)

Douglas G. Woolford

lambda — *Penalty Parameter Selector*

Description

Data-driven selector of the penalty parameter, given a bandwidth.

Usage

```
lambda(x, y, h, d, xgrid, A, B, niterations=2)
```

Arguments

x	numeric vector of predictor observations
y	numeric vector of observed responses
h	numeric bandwidth
d	numeric degree of local polynomial regression
xgrid	numeric vector of grid points where regression function is to be evaluated
A	numeric matrix, Smoother matrix
B	numeric matrix, based on penalty
niterations	number of iterations

Value

a numeric vector of smoothing parameters, corresponding to successive iterates

Author(s)

W.J. Braun

MISE *Approximate Mean Integrated Squared Error*

Description

MISE for penalized sharpened regression based on trapezoid integration.

Usage

```
MISE(x, xgrid, sigma2, lambda, h, g, A, B)
```

Arguments

x	numeric explanatory vector
xgrid	numeric vector
sigma2	numeric vector of variance(s)
lambda	numeric penalty constant
h	numeric bandwidth
g	regression function, numeric-valued
A	numeric matrix, smoother
B	numeric matrix, based on penalty

Value

A vector containing the finite sample variance, squared bias, and mean integrated squared error.

Author(s)

W.J. Braun

numericalDerivative *Numerical Derivative of Smooth Function*

Description

Cubic spline interpolation of columns of a matrix for purpose of computing numerical derivatives at a corresponding sequence of gridpoints.

Usage

```
numericalDerivative(x, g, k, delta=.001)
```

Arguments

x	numeric vector
g	numeric-valued function of x
k	number of derivatives to be computed
delta	denominator of Newton quotient approximation

Value

numeric vector of kth derivative of g(x)

Author(s)

W.J. Braun

---

penlocreg *Penalized Local Polynomial Regression*

Description

Data sharpened local polynomial regression subject to a given penalty.

Usage

```
penlocreg(x, y, xgrid, degree = 0, h, lambda, L, ...)
```

Arguments

x	numeric vector of predictor observations
y	numeric vector of observed responses
xgrid	numeric vector of grid points where regression function is evaluated
degree	numeric vector of local polynomial regression degree
h	numeric bandwidth
lambda	numeric penalty constant
L	function related to penalty
...	additional arguments, as required by L

Value

a list containing the original observed predictor values, the sharpened responses, the smoother matrix and the penalty matrix

Examples

```
xx <- faithful$waiting
yy <- faithful$eruptions
h <- dpill(xx,yy)/2; lam <- 20  # tuning parameter selections
yy.pen <- penlocreg(xx, yy, seq(min(xx), max(xx), len=401), lambda=lam, degree=1, h = h, L =
    SecondDerivativePenalty)
plot(xx, yy, xlab="waiting", ylab="eruptions", col="grey")
title("Old Faithful")
points(yy.pen, col=2, cex=.6) # sharpened data points
lines(locpoly(xx, yy, bandwidth=h*2, degree=1), lwd=2) # local linear estimate
lines(locpoly(yy.pen$x, yy.pen$y, bandwidth=h, degree=1), col=2, lwd=2) # sharpened estimate
```

---

pllr *Penalized Local Linear Regression*

Description

Data sharpened local linear regression with roughness penalty with automatically selected bandwidth and tuning parameter.

Usage

```
pllr(x, y)
```

Arguments

x	numeric vector of predictor observations
y	numeric vector of observed responses

Value

a list consisting of the x and y coordinates of the estimated regression function.

Author(s)

W.J. Braun

SecondDerivativePenalty *A Roughness Penalty Based on the Squared Second Derivative*

Description

A roughness penalty function based on squared second derivatives evaluated numerically. This is a possible template function for other types of penalties.

Usage

```
SecondDerivativePenalty(xgrid, a)
```

Arguments

xgrid	vector of length m, must be increasing
a	a function of one numeric variable

Value

a vector of second derivatives evaluated at the points of xgrid

Author(s)

W.J. Braun

sharp1d *Data Sharpening for Density Estimation*

Description

Application of Choi and Hall's (1999) data sharpening method for univariate data, for use prior to density estimation.

Usage

```
sharp1d(x, h, v = 1)
```

Arguments

x	the x coordinates of the data
h	the bandwidth for sharpening in the direction of the x axis
v	a positive integer representing the number of iterations to perform

Value

Returns a vector containing the sharpened points x.sharp.

Author(s)

Douglas G. Woolford, W. John Braun

References

Choi, E. and Hall, P. (1999) Data sharpening as a prelude to density estimation. Biometrika 86, 941-947.

Examples

```
# Example 1:
y <- c(rnorm(50,-1,1),rnorm(50,2,2), rnorm(100,0,.5))
data.sharp1 <- sharp1d(y,5,1)
data.sharp2 <- sharp1d(y,5,2)
# original data:
plot(density(y, bw=5))
# sharpened data after 1 iterations:
lines(density(data.sharp1, bw=5), col=2)
# sharpened data after 2 iterations:
lines(density(data.sharp2, bw=5), col=4)

x <- rt(100, df=3)
h <- dpik(x)

# Example 2:
curve(dt(x, df=3), from=-4, to=4)
lines(bkde(x, bandwidth=h), col=2, lty=2)

x.sharp <- sharp1d(x, h, 1)
lines(bkde(x.sharp, bandwidth=h), col=3, lty=3)

x.sharp2 <- sharp1d(x, h, 2)
lines(bkde(x.sharp2, bandwidth=h), col=4, lty=4)

x.sharp3 <- sharp1d(x, h, 3)
lines(bkde(x.sharp3, bandwidth=h), col=5, lty=5)
```

---

sharp2d *Identify Cluster Centres for 2-dimensional Data via Data Sharpening*

Description

Identifies the centres of clusters for 2-dimensional data using a converged form of Choi and Hall's (1999) data sharpening method.

Usage

```
sharp2d(x, y, hspace = 1, htime = 1, v = 1)
```

Arguments

x	the x coordinates of the data
y	the y coordinates of the data
hspace	the bandwidth for sharpening in the direction of the x axis
htime	the bandwidth for sharpening in the y direction
v	a positive integer representing the number of iterations to perform

Details

Identifies the centres of clusters based on a converged form of Choi and Hall's data sharpening method. This function was originally built for identifying clusters in space-time where space is the x-y plane and time is the z-axis.

Value

Returns a (number of data points x 2) data frame containing the sharpened points x.sharp and y.sharp, respectively.

Author(s)

Douglas G. Woolford, W. John Braun

References

Woolford, D. G. and Braun, W. J. (2004) Exploring lightning and fire ignition data as point processes. 2004 Proceeding of the American Statistical Association, Statistics and the Environment Section [CD-ROM], Alexandria, VA: American Statistical Association.

Choi, E. and Hall, P. (1999) Data sharpening as a prelude to density estimation. Biometrika 86, 941-947.

Examples

```
x <- 1:200
y <- c(rnorm(50,-1,1),rnorm(50,2,2), rnorm(100,0,.5))
data.sharp5 <- sharp2d(x,y,5,10,5)
data.sharp10 <- sharp2d(x,y,5,10,10)
# original data:
plot(x,y)
# sharpened data after 5 iterations:
points(data.sharp5$x.sharp, data.sharp5$y.sharp, col=2,pch=19)
# sharpened data after 10 iterations:
points(data.sharp10$x.sharp, data.sharp10$y.sharp, col=4, pch=19)
```

`sharp3d` *Identify Cluster Centres for 3-dimensional Data via Data Sharpening*

Description

Identifies the centres of clusters for 3-dimensional data using a convergent form of Choi and Hall's (1999) data sharpening method.

Usage

```
sharp3d(x, y, z, hspace = 1, htime = 1, v = 1)
```

Arguments

`x`	the x coordinates of the data
`y`	the y coordinates of the data
`z`	the z coordinates of the data
`hspace`	the bandwidth for sharpening in the direction of the x-y plane
`htime`	the bandwidth for sharpening in the z direction
`v`	a positive integer representing the number of iterations to perform

Details

Identifies the centres of clusters based on a convergent form of Choi and Hall's data sharpening method. This function was originally built for identifying clusters in space-time where space is the x-y plane and time is the z-axis.

Value

Returns a (number of data points x 3) data frame containing the sharpened points x.sharp, y.sharp and z.sharp, respectively.

Author(s)

Douglas G. Woolford, W. John Braun

References

Woolford, D. G. and Braun, W. J. (2004) Exploring lightning and fire ignition data as point processes. 2004 Proceeding of the American Statistical Association, Statistics and the Environment Section [CD-ROM], Alexandria, VA: American Statistical Association.

Choi, E. and Hall, P. (1999) Data sharpening as a prelude to density estimation. Biometrika 86, 941-947.

See Also

sharp3dB

Examples

```
x <- 1:200
y <- c(rnorm(50,-1,1),rnorm(50,2,2), rnorm(100,0,.5))
z <- c(sample(1:50,50), sample(26:75,50), sample(51:150,100))
data.sharp5 <- sharp3d(x,y,z,5,10,5)
data.sharp10 <- sharp3d(x,y,z,5,10,10)
# original data:
dataPlot <- scatterplot3d(x,y,z)
# sharpened data after 5 iterations:
dataPlot$points3d(data.sharp5$x.sharp, data.sharp5$y.sharp,
data.sharp5$z.sharp, col=2,pch=19)
# sharpened data after 10 iterations:
dataPlot$points3d(data.sharp10$x.sharp, data.sharp10$y.sharp,
data.sharp10$z.sharp, col=4, pch=19)
```

---

sharp3dB *Identify Cluster Centres for 3-dimensional Data via Data Sharpening*

Description

Identifies the centres of clusters for 3-dimensional data using a convergent form of Choi and Hall's (1999) data sharpening method. For use when the data is such that the z coordinates are in increasing order.

Usage

```
sharp3dB(x, y, z, hspace = 1, htime = 1, v = 1)
```

Arguments

x	the x coordinates of the data
y	the y coordinates of the data
z	the z coordinates of the data, in increasing order
hspace	the bandwidth for sharpening in the direction of the x-y plane
htime	the bandwidth for sharpening in the z direction
v	a positive integer representing the number of iterations to perform

Details

Identifies the centres of clusters based on a convergent form of Choi and Hall's data sharpening method. This function was originally built for identifying clusters in space-time where space is the x-y plane and time is the z-axis. Provided the z-data is in increasing order, this function is significantly faster than sharp3d().

Value

Returns a (number of data points x 3) data frame containing the sharpened points x.sharp, y.sharp and z.sharp, respectively.

Author(s)

Douglas G. Woolford, W. John Braun

References

Woolford, D. G. and Braun, W. J. (2004) Exploring lightning and fire ignition data as point processes. 2004 Proceeding of the American Statistical Association, Statistics and the Environment Section [CD-ROM], Alexandria, VA: American Statistical Association.

Choi, E. and Hall, P. (1999) Data sharpening as a prelude to density estimation. Biometrika 86, 941-947.

See Also

sharp3d

---

sharpen *Penalized Data Sharpening Operator for Local Polynomial Regression*

Description

Data perturbation operator which moves responses a minimal amount subject to a given penalty.

Usage

```
sharpen(x, y, lambda, B)
```

Arguments

x	numeric vector of predictor observations
y	numeric vector of observed responses
lambda	numeric penalty constant
B	numeric matrix, based on penalty

Value

a numeric vector containing the sharpened responses

Author(s)

W.J. Braun

Index