Package ‘bpkde’

January 27, 2015

Version 1.0-7

Date 2014-09-05

Title Back-Projected Kernel Density Estimation

Author Kjell Konis <`kjell.konis@me.com`>,
Victor M. Panaretos <`victor.panaretos@epfl.ch`>.

Maintainer Kjell Konis <`kjell.konis@me.com`>

Suggests mvtnorm, KernSmooth

Description Nonparametric multivariate kernel density \{ }
estimation using a back-projected kernel.

License GPL-2

Repository CRAN

Repository/R-Forge/Project bpkde

Repository/R-Forge/Revision 8

Repository/R-Forge/DateTimeStamp 2014-09-05 18:17:41

Date/Publication 2014-09-09 19:40:58

NeedsCompilation yes

R topics documented:

`bpk` *Back Projected Kernel*

Description

Evaluate a `d` dimensional back projected kernel.

Usage

```
bpk(X, bandwidths, a = 1.0, kernel = dnorm, ...)
```

Arguments

`X`	a numeric matrix of dimension `N` by `d` who's rows contain the points where the kernel will be evaluated.
`bandwidths`	a list with elements `alphas`, the set of projection directions, and `lambdas`, their associated bandwidth estimates.
`a`	a single numeric value containing a common scaling parameter.
`kernel`	a function for evaluating the univariate kernel.
`...`	additional arguments are ignored.

Value

a numeric vector where element i contains the value of the kernel evaluated at row i of `X`.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

`bpkde` *Back-Projected Kernel Density Estimation*

Description

Compute a `d`-dimensional kernel density estimate using a back-projected kernel.

Usage

```
bpkde(data, alphas, kernel = dnorm, bw = bw.SJ, score.fun = M1, r = 7, padding = 4)
```

Arguments

`data`	a matrix or data frame. The data is coerced to a numeric matrix using the `data.matrix` function.
`alphas`	a numeric matrix of dimension `d` by `K` whose columns contain the directions (as unit vectors) used for the back projection. If missing, 90 equally space directions spanning the interval $[-\pi/2, \pi/2)$ are used when `d == 2` and 450 randomly chosen directions are used when `d == 3`. This argument must be provided for `d >= 4`.
`kernel`	a function for evaluating the univariate kernel.
`bw`	the function used to compute the univariate bandwidth estimates.
`score.fun`	the function used to compute the least squares cross-validation score for the kernel; see `M0` and `M1`.
`r`	the computations are performed using linear binning and the discrete Fourier transform. The number of the grid points used is 2^r.
`padding`	a postive numeric value specifying the amount of zero-padding in units of bandwidth.

Value

a list with class `c("bpkde", "mvkde")` containing the following elements.

`axes`	a numeric matrix whose columns contain the grid points used along each axis to bin the data.
`z`	a numeric array containing the discrete kernel density estimate.
`params`	a list containing the optimal common scaling parameter `omega.hat`, the input set of directions `alphas`, and the computed univariate bandwidths `lambdas`.

References

Panaretos, Victor M. and Konis, Kjell (2012). Nonparametric Construction of Multivariate Kernels. *Journal of the American Statistical Association* 107(499):1085-1095.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

Examples

```
data(Trimodal2)
f.hat <- bpkde(Trimodal2)
```

`discretize.kernel` *Discretize Kernel*

Description

Computes a discrete approximation of the kernel.

Usage

```
discretize.kernel(grid, kern.fun, ..., grid.fun = NULL, scale = TRUE)
```

Arguments

`grid`	a list created by the `mvlinbin` function.
`kern.fun`	a function for evaluating the kernel. The first argument must be a matrix whose rows contain the points where the density is to be evaluated. See, for example, the `dmvnorm` function in the `mvtnorm` package.
`...`	additional arguments are passed to `kern.fun`.
`grid.fun`	a function that returns the grid points where the kernel will be evaulated.
`scale`	a logical value. If `TRUE` then the kernel is scaled so that it integrates to 1 on the provided grid.

Value

a list with class `kernel` contining the following elements.

`axes`	a numeric matrix whose columns contain the grid points along each axis where the kernel was evaluated.
`z`	a numeric array containing the discrete representation of the kernel.
`dft`	a numeric array containing the discrete Fourier transform of the kernel `z`.
`kern.fun`	the name of `kern.fun`.
`params`	a list containing the pearameters passed to `kern.fun` through `...`.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

`kernels` *Univariate Kernels*

Description

Evaluate univariate kernels.

Usage

```
biweight(x)
epanechnikov(x)
rectangular(x)
triangular(x)
```

Arguments

`x`	a numeic vector.

Value

a numeric vector.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

`M0` *M0 Least-Squares Cross-Validation*

Description

Compute the M0 least-squares cross-validation score.

Usage

```
M0(grid, kern.fun, ...)
```

Arguments

`grid`	an object of class `linbin`.
`kern.fun`	the density function of the kernel. The first argument must be a matrix whose rows contain the points where the density function will be evaluated. See, for example, the `dmvnorm` function in the `mvtnorm` package.
`...`	additional arguments are passed to `kern.fun`.

Value

a single numeric value: the M0 least-squares cross-validation score.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

References

Silverman, B. W. (1986) *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall.

`M1` *M1 Least-Squares Cross-Validation*

Description

Compute the M1 least-squares cross-validation score.

Usage

```
M1(grid, kern.fun, ...)
```

Arguments

`grid`	an object of class `linbin`.
`kern.fun`	the density function of the kernel. The first argument must be a matrix whose rows contain the points where the density function will be evaluated. See, for example, the `dmvnorm` function in the `mvtnorm` package.
`...`	additional arguments are passed to `kern.fun`.

Details

The computation is done using the Fourier transform of the kernel and the data as described in Silverman (1986).

Value

a single numeric value: the M1 least-squares cross-validation score.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

References

Silverman, B. W. (1986) *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall.

`mvlinbin` *Multivariate Linear Binning*

Description

Compute a binned approximation of the data on a regularly spaced grid using the multivariate linear binning rule described in Wand (1994).

Usage

```
mvlinbin(X, r = 7, padding)
```

Arguments

`X`	a numeric matrix.
`r`	a positive integer value. The number of grid points M in each dimension is given by $M = 2^r$.
`padding`	a numeric vector of positive values with length equal to the number of columns of `X` specifying the amount of zero-padding added to each coordinate direction. No padding is added when this argument is missing.

Value

a list with class `mvlinbin` containing the following elements.

`axes`	a numeric matrix whose columns contain the grid points used along each axis to bin the data.
`xi`	a numeric array containing the binned approximation of the data.
`X`	a numeric matrix containing the input data.
`deltas`	a numeric vector containing the grid spacing.
`M`	an integer value giving the number of grid points used in each coordinate direction.
`n`	an integer value containing the number of data points binned.
`d`	an integer value giving the dimensionality of the data.

Author(s)

Kjell Konis <`kjell.konis@me.com`>

References

Wand, M. P. (1994). Fast Computation of Multivariate Kernel Estimators. *Journal of Computational and Graphical Statistics*, 3, 433-445.

`WandJones` *Example Data Sets from Wand & Jones*

Description

The 12 normal mixture densities considered in Wand and Jones (1993).

Usage

```
data(Uncorrelated.Normal)
data(Correlated.Normal)
data(Skewed)
data(Kurtotic)
data(Bimodal1)
data(Bimodal2)
data(Bimodal3)
data(Bimodal4)
data(Trimodal1)
data(Trimodal2)
data(Trimodal3)
data(Quadrimodal)
```

Details

These 12 data sets consist of 500 points drawn from the Gaussian mixture distributions described in Wand and Jones (1993). The code is provided in the `scripts` subdirectory of this package.

Source

Wand, M. P. and Jones, M. C. (1993). Comparison of Smoothing Parameterizations in Bivariate Kernel Density Estimation. *Journal of the American Statistical Association*, Vol. 88, No. 422, pp. 520-528.

Index