Package 'MEPDF'

October 12, 2022

Type Package

Title Creation of Empirical Density Functions Based on Multivariate Data

Version 3.0

Author Martin Wiegand, Saralees Nadarajah

Date 2018-09-26

Maintainer Martin Wiegand <Martin.Wiegand@manchester.ac.uk>

Depends R (>= 3.0.1)

Description Based on the input data an n-dimensional cube with sub cells of user specified side length is created.
The number of sample points which fall in each sub cube is counted, and with the cell volume and overall sample size an empirical probability can be computed. A number of cubes of higher resolution can be superimposed. The basic method stems from J.L. Bentley in ``Multidimensional Divide and Conquer''.
J. L. Bentley (1980) <doi:10.1145/358841.358850>.
Furthermore a simple kernel density estimation method is made available, as well as an expansion of Bentleys method, which offers a kernel approach for the grid method.

License GPL-2

LazyData TRUE

NeedsCompilation no

Imports plyr,mvtnorm,pracma,stats,gtools

Repository CRAN

Date/Publication 2018-09-26 14:10:03 UTC

R topics documented:

---

cube *cube*

Description

Single grid size empirical density function. To be used to construct the epdf function.

Usage

```
cube(data,mx,mn,grid.sizes)
```

Arguments

`data`	N-dimensional data set.
`mx`	Upper cropping point of the data.
`mn`	Lower cropping point of the data.
`grid.sizes`	Vector of grid sizes for the grid.

Examples

```
library("pracma")
library("plyr")

data<-cbind(rnorm(1000),rnorm(1000))

pdf<-cube(data,mx=c(1,1),mn=c(-1,-1),grid.sizes = c(0.01,0.01))
```

---

ekde *ekde*

Description

Kernel function based on the normal distribution.

Usage

```
ekde(x,data,H,rule,kernel)
```

Arguments

x	Evaluation point.
data	Input data.
H	Positive-definite, symmetric matrix as bandwidth.
rule	In absence of a bandwidth matrix a rule-of-thumb can be chosen, either the "silverman" or "scott" rule.
kernel	The kernel function of choice.

Examples

```
library("pracma")
library("plyr")

data<-cbind(rnorm(1000),rnorm(1000))
pdf<-ekde(x = 0,data = data, rule = "silverman",kernel = normkernel)
```

---

epakernel *epakernel*

Description

Kernel function based on the normal distribution.

Usage

```
epakernel(x,H)
```

Arguments

x	Evaluation point.
H	Positive-definite, symmetric matrix as bandwidth.

Examples

```
epakernel(c(1,1),H = diag(2))
```

epdf *epdf*

Description

Assembles main grid and superimposes grids of different resolution.

Usage

```
epdf(data,max.corner,min.corner,main.gridsize,rescubes)
```

Arguments

data	N-dimensional data set.
max.corner	Upper cropping point of the data.
min.corner	Lower cropping point of the data.
main.gridsize	Vector of grid sizes for the main grid.
rescubes	List of upper and lower cropping points, as well as grid sizes for cubes that are to be superimposed

Examples

```
library("pracma")
library("plyr")

a<-list(c(-1,-1),c(1,1),c(0.01,0.01))
b<-list(c(-2,-2),c(2,2),c(0.02,0.02))
cubes<-list(a,b)

min.corner = c(-4,-4)
max.corner= c(4,4)
main.gridsize = c(0.05,0.05)

# Data & Density
data<-cbind(rnorm(1000),rnorm(1000))

pdf<-epdf(data,max.corner,min.corner,main.gridsize,rescubes = cubes)
```

`normkernel` *normkernel*

Description

Kernel function based on the Epanechnikov kernel.

Usage

```
normkernel(x,H)
```

Arguments

`x`	Evaluation point.
`H`	Positive-definite, symmetric matrix as bandwidth.

Examples

```
normkernel(c(1,1),H = diag(2))
```

`pseudokernel` *pseudokernel*

Description

Single grid size empirical density function. Offers a modified kernel density approach via the optional argument "rings"

Usage

```
pseudokernel(data,mn,mx,grid.sizes,rings)
```

Arguments

`data`	N-dimensional data set.
`mx`	Upper cropping point of the data.
`mn`	Lower cropping point of the data.
`grid.sizes`	Vector of grid sizes for the grid.
`rings`	Number of additional cell grid smoothing layers.

Examples

```
data<-mvtnorm::rmvnorm(n = 100,mean = c(0,0),sigma = diag(2))

pdf<-pseudokernel(data = data,mn = c(-2,-2),mx = c(2,2),grid.sizes = c(0.1,0.1),rings = 1)
```

Index