

Package ‘ash’

February 3, 2010

Version 1.0-10

VersionPlus 1.0-10-2

Title David Scott’s ASH routines

Author S original by David W. Scott R port by Albrecht Gebhardt
<albrecht.gebhardt@uni-klu.ac.at>

Maintainer Albrecht Gebhardt <albrecht.gebhardt@uni-klu.ac.at>

MaintainerPlus Stephen Kaluzny <skaluzny@tibco.com>

Description David Scotts ASH routines

License S original available at statlib

Dialect R, S-PLUS

R topics documented:

ash-internal	1
ash1	2
ash2	3
bin1	3
bin2	4

ash-internal	<i>Internal ash functions</i>
--------------	-------------------------------

Description

Internal ash functions

Details

These functions are not intended to be called by the user.

ash1 *univariate ASH*

Description

Computes univariate averaged shifted histogram (polynomial kernel)

Usage

```
ash1(bins, m, kopt)
```

Arguments

bins (input list) $\$nc$ =integer vector of bin counts and $\$ab$ =bin interval

m (input) optional integer smoothing parameter; default=5.

kopt (input) vector of length 2 specifying the kernel, which is proportional to $(1 - abs(i/m)^k opt(1))i^k opt(2)$; (2,2)=biweight (default); (0,0)=uniform; (1,0)=triangle; (2,1)=Epanechnikov; (2,3)=triweight.

Value

returns structure suitable for input to `plot`

x vector of bin center locations

y vector of ash estimates

ier error flag: 0 = normal exit; 1 = estimate nonzero outside interval ab

See Also

`bin1`

Examples

```
x <- rnorm(100)                      # data
f <- ash1(bin1(x,nbin=50),5) # compute ash estimate
plot( f , type="l" )                # line plot of estimate
```

ash2	<i>bivariate ASH</i>
------	----------------------

Description

Compute bivariate ASH estimate (product polynomial kernel)

Usage

```
ash2(bins, m, kopt)
```

Arguments

<code>bins</code>	(input list) bin count matrix <code>nc</code> and interval matrix <code>ab</code> from <code>bin2</code>
<code>m</code>	(input integer vector of length 2) x and y direction smoothing parameters. Default is 5 by 5.
<code>kopt</code>	see <code>ash1</code>

Value

Matrix of ASH estimates returned. Components `x,y,z` can be given to the contour function directly. Other input variables returned in list for record keeping.

See Also

`bin2`

Examples

```
# Continuing example from help(bin2)
m <- c(5,5)
f <- ash2(bins,m)
image(f$x,f$y,f$z)
contour(f$x,f$y,f$z,add=TRUE)
```

<code>bin1</code>	<i>univariate binning</i>
-------------------	---------------------------

Description

Function to compute array of bin counts for a data vector

Usage

```
bin1(x, ab, nbin=50)
```

Arguments

x (input) data vector

ab (input vector of length 2): half-open interval for bins $[a, b)$. If no value is specified, the range of x is stretched by 5% at each end and used the interval.

nbin (input integer): number of bins desired. Default 50.

Value

`bin1` returns a list including the vector of integer bin counts and the `ab` vector and the number of points outside the `ab` interval.

See Also

`ash1`

Examples

```
x <- rnorm(100)      # data vector
ab <- c(-5,5)       # bin interval
bins <- bin1(x,ab,10) # bin x into 10 bins over ab
```

`bin2`

2D binning

Description

Bin bivariate data x

Usage

```
bin2(x, ab, nbin)
```

Arguments

x (input matrix with 2 columns) data sample

ab (input 2 x 2 matrix) rows 1 and 2 contain x and y axis bin intervals, respectively. If not specified, the ranges are stretched by 5% at each end for each dimension.

nbin (input vector of length 2) number of bins along x and y axes. Default is 20 by 20.

Value

`bin2` returns a list including the bivariate bin matrix and the number of points outside the `ab` rectangle.

See Also

`ash2`

Examples

```
x <- matrix( rnorm(200), 100 , 2)      # bivariate normal n=100
ab <- matrix( c(-5,-5,5,5), 2, 2)     # interval [-5,5) x [-5,5)
nbin <- c( 20, 20)                   # 400 bins
bins <- bin2(x, ab, nbin)             # bin counts,ab,nskip
```

Index

*Topic **nonparametric**

ash-internal, 1

ash1, 2

ash2, 3

bin1, 3

bin2, 4

ash-internal, 1

ash1, **2**, 3, 4

ash2, **3**, 5

bin1, 2, **3**

bin2, 3, 4

center (*ash-internal*), 1

nicerange (*ash-internal*), 1