Package 'hexDensity'

July 22, 2025

Type Package

Title Fast Kernel Density Estimation with Hexagonal Grid

Version 1.4.9

Maintainer Quoc Hoang Nguyen <nguyen.q@wehi.edu.au>

Description Kernel density estimation with hexagonal grid for bivariate data.
Hexagonal grid has many beneficial properties like equidistant neighbours
and less edge bias, making it better for spatial analyses than the more
commonly used rectangular grid.
Carr, D. B. et al. (1987) <doi:10.2307/2289444>.
Diggle, P. J. (2010) <doi:10.1201/9781420072884>.
Hill, B. (2017) <https://blog.bruce-hill.com/meandering-triangles>.
Jones, M. C. (1993) <doi:10.1007/BF00147776>.

URL https://github.com/ChenLaboratory/hexDensity

License GPL

Encoding UTF-8

NeedsCompilation yes

Imports hexbin, spatstat.geom, grid, grDevices, methods

Suggests ggplot2

RoxygenNote 7.3.2

Author Quoc Hoang Nguyen [aut, cre] (ORCID:
<https://orcid.org/0009-0007-2828-7567>),
Yunshun Chen [aut] (ORCID: <https://orcid.org/0000-0003-4911-5653>),
Dan Carr [ctb],
binning routine ported by Nicholas Lewin-Koh [ctb],
Martin Maechler [ctb]

Repository CRAN

Date/Publication 2025-06-07 16:50:02 UTC

Contents

`grid.hexagontile` *Draw hexagon tiles with grid package*

Description

Adapted from grid.hexagons by hexbin with speedup specific for hexagonal tiling (avoid plotting the most abundance hexagons by setting its color as background).

Usage

```
grid.hexagontile(
  hexDensity,
  use.count = TRUE,
  cell.at = NULL,
  trans = NULL,
  colorcut = seq(0, 1, length = 1024),
  colramp = colorRampPalette(col.viridis),
  def.unit = "native"
)
```

Arguments

`hexDensity`	hexbin object returned by hexDensity.
`use.count`	logical specifying if counts from hexbin object should be used.
`cell.at`	numeric vector to be plotted instead of counts, must be same length as the number of cells.
`trans`	a transformation function (or NULL) for the counts, e.g., sqrt.
`colorcut`	An integer for the number of equi-spaced colorcut in [0,1] to assign colors to values. Alternatively, a vector of custom colorcut spacing between [0, 1].
`colramp`	Color function that accept an integer n and return n colors.
`def.unit`	Default unit to be used.

Value

No return value

SIDE EFFECTS

Adds hexagons to plot

Author(s)

Dan Carr <dcarr@voxel.galaxy.gmu.edu>; ported and extended by Nicholas Lewin-Koh nikko@hailmail.net. Modified by Quoc Hoang Nguyen <nguyen.q@wehi.edu.au> for hexDensity.

`hexbinFull` *Hexagonal binning with whole grid output.*

Description

Adapted from hexbin to output hexagons with 0 count, and also to allow points with different weights. Default to use regular hexagon. See hexbin for extra detail on the output.

Usage

```
hexbinFull(
  x,
  y = NULL,
  xbins = 128,
  shape = NULL,
  xbnds = range(x),
  ybnds = range(y),
  xlab = NULL,
  ylab = NULL,
  IDs = FALSE,
  weight = NULL,
  fractional = FALSE
)
```

Arguments

x, y	Coords of the points or a single plotting structure to be used in binning. See xy.coords.
xbins	Number of bins in a row.
shape	shape = yheight/xwidth of the plotting regions
xbnds, ybnds	Horizontal and vertical limits of the binning region in x or y units respectively, must encompass range(x) or range(y) respectively; Vector of length 2
xlab, ylab	Optional character strings used as labels for x and y. If NULL, sensible defaults are used.
IDs	Logical indicating if the hexagonal cell ID for each point should be returned, see hexbin.
weight	Numeric weight vector to be assigned to points.
fractional	Logical. Whether to use fractional binning scheme.

Value

an S4 object of class hexbin.

Author(s)

Dan Carr <dcarr@voxel.galaxy.gmu.edu>; ported and extended by Nicholas Lewin-Koh <nikko@hailmail.net>. Modified by Quoc Hoang Nguyen <nguyen.q@wehi.edu.au> for hexDensity.

References

Carr, D. B. et al. (1987) Scatterplot Matrix Techniques for Large N. JASA 83, 398, 424–436.

Examples

```
set.seed(133)
x=rnorm(20000)
y=rnorm(20000)
d=hexbinFull(x,y,xbins=50)
plotHexDensity(d)
d=hexbinFull(x,y,xbins=50,fractional=TRUE)
plotHexDensity(d)
```

---

hexContour *Generate contour for a hexagonal grid.*

Description

Algorithm is a modification of the meandering triangles as described in https://blog.bruce-hill.com/meandering-triangles to work with hexagons. See isolines for details about the output.

Usage

```
hexContour(hexDensity, levels)
```

Arguments

`hexDensity`	hexDensity object to be contoured.
`levels`	Numeric vector for which contour lines should be generated

Details

This function is made to follow the same behaviour as isolines. A dedicated plotting function is in the work. Meanwhile, see example of how to plot the output with ggplot2's geom_path.

Value

A list of x, y, and ID, for the contour line at each levels. ID indicates the different line segments making up the contour.

Examples

```
set.seed(133)
x=rnorm(200)
y=rnorm(200)
d = hexDensity(x=x,y=y,bandwidth=0.4)
cutoff=quantile(d@count,0.9)
lines = hexContour(d,cutoff)

library(ggplot2)
library(hexbin)
#plot against density
ggplot()+
  geom_point(
    aes(x=hcell2xy(d)$x,
        y=hcell2xy(d)$y,
        col=d@count)
 ) +
  scale_color_viridis_c()+
  geom_path(
    aes(
     x = lines[[1]]$x, y = lines[[1]]$y, group = lines[[1]]$id
    )
  )

#plot against data points
ggplot() +
  geom_point(
    aes(x=x,y=y)) +
  geom_path(
    aes(
      x = lines[[1]]$x, y = lines[[1]]$y, group = lines[[1]]$id
    )
  )
```

---

hexDensity *Kernel Density Estimation with Hexagonal grid.*

Description

Kernel Density Estimation with Hexagonal grid.

Usage

```
hexDensity(
  x,
  y = NULL,
  xbins = 128,
  bandwidth = NULL,
  edge = TRUE,
```

```
  diggle = FALSE,
  weight = NULL,
  ...
)
```

Arguments

x, y	Coords of the points or a single plotting structure to be used in binning. See xy.coords.
xbins	Number of bins in a row.
bandwidth	Bandwidth for the smoothing kernel. Either a scalar, a vector of length 2, or a 2x2 variance-covariance matrix for the bandwidths in the x and y directions.
edge	Logical value for whether to apply edge correction. Default is TRUE.
diggle	Logical value for apply edge correction with the more accurate Jones-Diggle method (need 'edge' to be TRUE).
weight	numeric weight vector to be assigned to points.
...	arguments for hexbinFull

Details

Default bandwidth is the normal scale bandwidth selector n^(-1/3)*var where n is sample size and var is the variance-covariance matrix.

Value

an S4 object of class hexbin.

References

Diggle, P. J. (2010) Nonparametric methods. Chapter 18, pp. 299–316 in A.E. Gelfand, P.J. Diggle, M. Fuentes and P. Guttorp (eds.) Handbook of Spatial Statistics, CRC Press, Boca Raton, FL.

Jones, M. C. (1993) Simple boundary corrections for kernel density estimation. Statistics and Computing 3, 135–146.

Examples

```
set.seed(133)
d = hexDensity(x=rnorm(200),y=rnorm(200))
```

meanderingTriangles *Meandering triangles for hexagonal grid in C++*

Description

Meandering triangles for hexagonal grid in C++

Usage

```
meanderingTriangles(x.coords.left, x.coords.right, y.coords, z, levels)
```

Arguments

x.coords.left	Vector for x coords of left-aligned rows (row 1,3,5,...)
x.coords.right	Vector for x coords of right-aligned rows (row 2,4,6,...)
y.coords	Vector for y coords of all rows.
z	Matrix for elevation values for the grid point
levels	Vector of z value cutoffs for contouring.

Details

This function is not meant to be used as is, unless you are very familiar with how hexContour works.

Value

list of x, y, and ID, for the contour line at each levels.

References

Hill, B. (2017) Meandering triangles. Naming Things. https://blog.bruce-hill.com/meandering-triangles

plotHexDensity *Plotting method for hexagonal Kernel Density Estimation*

Description

Adapted the plotting function from hexbin. X and Y axes now have the same scale with option for different aspect ratio. Ribbon legend for continuous data.

Usage

```
plotHexDensity(
  hexDensity,
  main = NULL,
  xlab = NULL,
  ylab = NULL,
  xaxt = TRUE,
  yaxt = TRUE,
  lcex = 1,
  colramp = colorRampPalette(col.viridis),
  colorcut = 1024,
  legend = TRUE,
  legendWidth = 0.05,
  legendDistance = 0.15,
  aspectRatio = diff(hexDensity@xbnds)/diff(hexDensity@ybnds),
  margin = 0.18,
  newpage = TRUE
)
```

Arguments

hexDensity	hexbin object returned by hexDensity
main	Main title
xlab, ylab	x-axis and y-axis label
xaxt, yaxt	Logical. Whether to plot x,y axes
lcex	Expansion factor for all letters.
colramp	Color function that accept an integer n and return n colors.
colorcut	An integer for the number of equi-spaced colorcut in [0,1] to assign colors to values. Alternatively, a vector of custom colorcut spacing between [0, 1].
legend	Legend is currently non-functional and should be ignored.
legendWidth	Expansion factor for legend width.
legendDistance	Expansion factor for the space between the plot and the legend.s
aspectRatio	width to height ratio of the plot. Default is the (inverse of) shape value of hexDensity.
margin	Minimum guaranteed margin for the plot. Different aspect ratio between the screen and the plot means that margin can be larger on certain sides.
newpage	logical for whether to plot on a new page.

Value

No return value

SIDE EFFECTS

Create kernel density estimate plot with hexagons

Author(s)

Dan Carr <dcarr@voxel.galaxy.gmu.edu>; ported and extended by Nicholas Lewin-Koh nikko@hailmail.net. Modified by Quoc Hoang Nguyen <nguyen.q@wehi.edu.au> for hexDensity.

Examples

```
set.seed(133)
d = hexDensity(x=rnorm(200),y=rnorm(200),bandwidth=0.15)
plotHexDensity(d)
```

---

xy2hcell *Find the hexagon cells from xy coordinates given a hexbin object.*

Description

Find the hexagon cells IDs from xy coordinates given a hexbin object. Useful if you want to get the KDE value at a certain coordinate.

Usage

```
xy2hcell(
  hexbin = NULL,
  x,
  y = NULL,
  xbins = NULL,
  xbnds = NULL,
  ybnds = NULL,
  shape = NULL
)
```

Arguments

hexbin	hexbin object to be referenced to.
x, y	coordinates or vectors of coordinates of the points.
xbins	number of bins partitioning the range of xbnds
xbnds, ybnds	horizontal and vertical limit of the binning region. Must be numeric vector of length 2.
shape	shape = yheight/xwidth of the plotting regions.

Details

If a hexbin object is not provided, parameters of the binning region (xbins, xbnds, ybnds, shape) can be used instead. For finding the xy coordinates of the hexagons for a hexbin object, see hcell2xy.

Value

a vector the same length as x with the hexagonal cell ID for each point

Examples

```
library(hexbin)
set.seed(133)
d=hexDensity(x=rnorm(20000),y=rnorm(20000),xbins=50)
#Get KDE value at the coordinate x=0,y=0
loc = xy2hcell(d,x=0,y=0)
d@count[loc]
```

Index