

Package ‘r2d2’

February 20, 2015

Version 1.0-0

Date 2014-03-31

Title Bivariate (Two-Dimensional) Confidence Region and Frequency Distribution

Author Arni Magnusson [aut], Julian Burgos [aut, cre], Gregory R. Warnes [ctb]

Maintainer Julian Burgos <julian@hafro.is>

Imports KernSmooth, MASS, sp

Suggests cluster, lattice

LazyData yes

Description This package provides generic functions to analyze the distribution of two continuous variables: 'conf2d' to calculate a smooth empirical confidence region, and 'freq2d' to calculate a frequency distribution.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2014-03-31 19:10:21

R topics documented:

r2d2-package	2
conf2d	2
freq2d	5
saithe	7
Ushape	7

Index	9
--------------	----------

r2d2-package

Bivariate (Two-Dimensional) Confidence Region and Frequency Distribution

Description

This package provides generic functions to analyze the distribution of two continuous variables.

Details

Bivariate calculations:

`conf2d` empirical confidence region, a smooth polygon
`freq2d` frequency distribution, a table

Examples:

`saithe` MCMC results in two columns
`Ushape` U-shaped cloud in two columns

Author(s)

Arni Magnusson and Julian Burgos, based on earlier functions by Gregory R. Warnes.

References

Bivand, R. S., Pebesma, E., and Gomez-Rubio, V. (2013) *Applied Spatial Data Analysis with R*. Second edition. New York: Springer.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. New York: Springer.

Wand, M. P. and Jones, M. C. (1995) *Kernel Smoothing*. London: Chapman and Hall.

See Also

Combines existing tools from the **KernSmooth**, **MASS**, and **sp** packages.

conf2d*Bivariate (Two-Dimensional) Confidence Region*

Description

Calculate an empirical confidence region for two variables, and optionally overlay the smooth polygon on a scatterplot.

Usage

```

conf2d(x, ...)

## S3 method for class 'formula'
conf2d(formula, data, subset, ...)

## Default S3 method:
conf2d(x, y, level=0.95, n=200, method="wand", shape=1, smooth=50,
       plot=TRUE, add=FALSE, xlab=NULL, ylab=NULL, col.points="gray",
       col="black", lwd=2, ...)

conf2d_int(x, y, surf, level, n) # internal function

```

Arguments

x	a vector of x values, or a data frame whose first two columns contain the x and y values.
y	a vector of y values.
formula	a formula , such as $y \sim x$.
data	a <code>data.frame</code> , <code>matrix</code> , or <code>list</code> from which the variables in <code>formula</code> should be taken.
subset	an optional vector specifying a subset of observations to be used.
level	the proportion of points that should be inside the region.
n	the number of regions to evaluate, before choosing the region that matches level best.
method	kernel smoothing function to use: "wand" or "mass".
shape	a bandwidth scaling factor, affecting the polygon shape.
smooth	the number of bins (scalar or vector of length 2), affecting the polygon smoothness.
plot	whether to plot a scatterplot and overlay the region as a polygon.
add	whether to add a polygon to an existing plot.
xlab	a label for the x axis.
ylab	a label for the y axis.
col.points	color of points.
col	color of polygon.
lwd	line width of polygon.
...	further arguments passed to <code>plot</code> and <code>polygon</code> .
surf	a list whose first three elements are x coordinates, y coordinates, and a surface matrix.

Details

This function constructs a large number (n) of smooth polygons, and then chooses the polygon that comes closest to containing a given proportion (`level`) of the total points.

The default `method="wand"` calls the `bkde2D` kernel smoother from the **KernSmooth** package, while `method="mass"` calls `kde2d` from the **MASS** package.

The `conf2d` function calls `bkde2D` or `kde2d` to compute a smooth surface from x and y . If users already have a smoothed surface to work from, the internal `conf2d_int` can be used directly to find the empirical confidence region that matches `level` best.

Value

List containing five elements:

<code>x</code>	<code>x</code> coordinates defining the region.
<code>y</code>	<code>y</code> coordinates defining the region.
<code>inside</code>	logical vector indicating which of the original data coordinates are inside the region.
<code>area</code>	area inside the region.
<code>prop</code>	actual proportion of points inside the region.

Note

The area of a bivariate region is analogous to the range of a univariate interval. This allows a quantitative comparison of different confidence regions.

Ellipses are a more restrictive approach to calculate an empirical bivariate confidence region. Smooth polygons make fewer assumptions about how x and y covary.

The `conf2d` and `freq2d` functions are closely related. The advantage of `conf2d` is that it returns a region as a smooth polygon. The advantage of `freq2d` is that it returns a set that is guaranteed to contain the correct proportion of points, even for spatially complex datasets.

Author(s)

Arni Magnusson and Julian Burgos, based on an earlier function by Gregory R. Warnes.

See Also

[quantile](#) is the corresponding univariate equivalent.

The **distfree.cr** package uses a different smoothing algorithm to calculate bivariate empirical confidence regions.

[ci2d](#) in the **gplots** package is a predecessor of `conf2d`.

[freq2d](#) calculates a discrete frequency distribution for two continuous variables.

[r2d2-package](#) gives an overview of the package.

Examples

```

conf2d(Ushape)$prop
conf2d(saithe, pch=16, cex=1.2, col.points=rgb(0,0,0,0.1), lwd=3)

# First surface, then region
plot(saithe, col="gray")
surf <- MASS::kde2d(saithe$Bio, saithe$HR, h=0.25, n=100)
region <- conf2d_int(saithe$Bio, saithe$HR, surf, level=0.95, n=200)
polygon(region, lwd=2)

```

freq2d

Bivariate (Two-Dimensional) Frequency Distribution

Description

Calculate a frequency distribution for two continuous variables.

Usage

```

freq2d(x, ...)

## S3 method for class 'formula'
freq2d(formula, data, subset, ...)

## Default S3 method:
freq2d(x, y, n=20, pad=0, layout=1, print=TRUE, dnn=NULL, ...)

```

Arguments

x	a vector of x values, or a data frame whose first two columns contain the x and y values.
y	a vector of y values.
formula	a formula , such as $y\sim x$.
data	a data.frame, matrix, or list from which the variables in formula should be taken.
subset	an optional vector specifying a subset of observations to be used.
n	the desired number of bins for the output, a scalar or a vector of length 2.
pad	number of rows and columns to add to each margin, containing only zeros.
layout	one of three layouts for the output: 1, 2, or 3.
print	whether to display the resulting matrix on the screen using dots for zeros.
dnn	the names to be given to the dimensions in the result.
...	named arguments to be passed to the default method.

Details

The exact number of bins is determined by the [pretty](#) function, based on the value of `n`.

Padding the margins with zeros can be helpful for subsequent analysis, such as smoothing.

The `print` logical flag only has an effect when `layout=1`.

Value

The `layout` argument specifies one of the following formats for the binned frequency output:

1. `matrix` that is easy to read, aligned like a scatterplot.
2. `list` with three elements (`x`, `y`, `matrix`) that can be passed to various plotting functions.
3. `data.frame` with three columns (`x`, `y`, `frequency`) that can be analyzed further.

Author(s)

Arni Magnusson.

See Also

[cut](#), [table](#), and [print.table](#) are the basic underlying functions.

[hist2d](#) in the **gplots** package is a related function with graphical capabilities.

[conf2d](#) calculates a bivariate empirical confidence region, a smooth polygon.

[r2d2-package](#) gives an overview of the package.

Examples

```
freq2d(Ushape)
freq2d(quakes$long, quakes$lat, dnn="")
freq2d(lat~long, quakes, n=c(10,20), pad=1)

# Suppress display
freq2d(saithe)
range(freq2d(saithe, print=FALSE))

# Layout, plot
freq2d(saithe, layout=2)
freq2d(saithe, layout=3)
contour(freq2d(saithe, layout=2))
lattice::contourplot(Freq~Bio+HR, freq2d(saithe, layout=3))
```

saithe

MCMC Results from Saithe Assessment

Description

Markov chain Monte Carlo results from the analysis of the saithe (*Pollachius virens*) fishery in Icelandic waters.

Usage

saithe

Format

Data frame containing 1000 rows and 2 columns:

Bio population biomass in 2013, relative to the expected long-term biomass under optimal harvest rate.
HR harvest rate in 2013, relative to the optimal harvest rate.

References

Magnusson, A. (2013) Icelandic saithe. In: *Report of the North Western Working Group (NWWG)*. ICES CM 2013/ACOM:07, pp. 231–252.

Magnusson, A., Punt, A. E., and Hilborn, R. (2013) Measuring uncertainty in fisheries stock assessment: the delta method, bootstrap, and MCMC. *Fish and Fisheries* **14**, 325–342.

Examples

```
conf2d(saithe, level=0.9)
freq2d(saithe)
```

Ushape

U-Shaped Cloud

Description

Bivariate scatter shaped like an open circle, for testing spatial algorithms.

Usage

Ushape

Format

Matrix containing 1000 rows and 2 columns:

x x coordinates.
y y coordinates.

Examples

```
freq2d(Ushape)  
conf2d(Ushape)
```


Index

*Topic **datasets**

sai the, 7

Ushape, 7

*Topic **distribution**

conf2d, 2

freq2d, 5

r2d2-package, 2

*Topic **dplot**

conf2d, 2

freq2d, 5

r2d2-package, 2

*Topic **manip**

freq2d, 5

r2d2-package, 2

*Topic **multivariate**

conf2d, 2

freq2d, 5

r2d2-package, 2

*Topic **smooth**

conf2d, 2

r2d2-package, 2

bkde2D, 4

ci2d, 4

conf2d, 2, 2, 6

conf2d_int (conf2d), 2

cut, 6

formula, 3, 5

freq2d, 2, 4, 5

hist2d, 6

kde2d, 4

pretty, 6

print.table, 6

quantile, 4

r2d2 (r2d2-package), 2

r2d2-package, 2

sai the, 2, 7

table, 6

Ushape, 2, 7