

Package ‘exCon’

January 27, 2015

Type Package

Title Interactive Exploration of Contour Data

Version 0.1-3

Date 2015-01-13

Description exCon is an interactive tool to explore topographic-like data sets. Such data sets take the form of a matrix in which the rows and columns provide location/frequency information, and the matrix elements contain altitude/response information. Such data is found in cartography, 2D spectroscopy and chemometrics. exCon creates an interactive web page showing the contoured data set along with slices from the original matrix parallel to each dimension. The page is written in d3/javascript.

License GPL-3

Imports jsonlite

URL <https://github.com/bryanhanson/exCon>

ByteCompile TRUE

BugReports <https://github.com/bryanhanson/exCon/issues>

Depends R (>= 3.0)

NeedsCompilation no

Author Bryan Hanson [aut, cre],
Kristina Mulry [ctb]

Maintainer Bryan Hanson <hanson@depauw.edu>

Repository CRAN

Date/Publication 2015-01-14 06:28:31

R topics documented:

exCon-package	2
exCon	2
Index	5

 exCon-package

Explore Contour Data Interactively

Description

exCon: Explore Contour Data Interactively

Details

exCon is an interactive tool to explore topographic-like data sets. Such data sets take the form of a matrix in which the rows and columns provide location/frequency information, and the matrix elements contain altitude/response information. Such data is found in cartography, 2D spectroscopy and chemometrics. exCon creates an interactive web page showing the contoured data set along with slices from the original matrix parallel to each dimension. The page is written in d3/javascript.

Author(s)

Bryan A. Hanson & Kristina R. Multry

 exCon

Explore Contour Data Interactively

Description

This function computes contour lines from matrix data and displays them in an interactive web page using the d3 javascript library.

Usage

```
exCon(M = NULL, x = seq(0, 1, length.out = nrow(M)), y = seq(0, 1,
  length.out = ncol(M)), nlevels = 5, levels = pretty(range(M, na.rm =
  TRUE), nlevels), browser = NULL)
```

Arguments

M	A matrix.
x	A vector of numeric values giving the locations of the grid defining the matrix. Must have length <code>nrow(M)</code> .
y	A vector of numeric values giving the locations of the grid defining the matrix. Must have length <code>ncol(M)</code> .
nlevels	Integer. The number of contour levels desired. Ignored if <code>levels</code> is given.
levels	Numeric. A vector of values (altitudes if you will) at which to compute the contours.
browser	Character. Something that will make sense to your OS. Only necessary if you want to override your system specified browser as understood by R. See below for further details.

Value

None; side effect is an interactive web page. The temporary directory containing the files that drive the web page is written to the console in case you wish to use those files. This directory is deleted when you quit R.

Details

The computation of the contour lines is handled by `contourLines`. The result here, however, is transposed so that the output has the same orientation as the original matrix. This is necessary because `contour` transposes its output: "Notice that `contour` interprets the `z` matrix as a table of $f(x[i], y[j])$ values, so that the `x` axis corresponds to row number and the `y` axis to column number, with column 1 at the bottom, i.e. a 90 degree counter-clockwise rotation of the conventional textual layout."

Interpretation

The contour lines are an interpolation of the data in the matrix. The slices are the actual values in the matrix row or column connected point-to-point. Thus a maximum in a slice may not correspond to a peak in the contour plot.

Browser Choice

The browser is called by `browseURL`, which in turn uses `options("browser")`. Exactly how this is handled is OS dependent.

RStudio Viewer

If `browser` is `NULL`, you are using RStudio, and a viewer is specified, this will be called. You can stop this by with `options(viewer = NULL)`.

Browser Choice/Mac

On a Mac, the default browser is called by `/bin/sh/open` which in turn looks at which browser you have set in the system settings. You can override your default with `browser = "/usr/bin/open -a 'Google Chrome'"` for example. Testing shows that on a Mac, Safari and Chrome perform correctly, but in Firefox the mouse cursor is slightly offset from the guides. While it doesn't look quite right, it works correctly (the guides determine which slice is displayed).

Browser Choice/Other Systems

exCon has been tested on a Windows 7 professional instance running in VirtualBox using Firefox and Chrome, and runs correctly (Firefox has the same mouse position issue as mentioned above).

Browser Choice & Performance

You can check the performance of your browser at peacekeeper.futuremark.com The most relevant score for exCon is the rendering category. In limited testing, Chrome does the best.

Performance Limits (YMMV)

On a 4-year old MacBook Pro, with 8 Gb RAM and an Intel Core i7 chip, a 4000 x 4000 matrix with 5 contour levels requires about 30 seconds for R to create the contours. The web page displayed by Chrome 38 appears to be about 85 Mb in size and the guide movements lag the mouse movements quite a bit, but it is still usable. Sometimes the page won't load. The files on disk are about 159 Mb. Firefox 32 will load the 4K x 4K matrix but performance is too sluggish. On the same computer, a 5000 x 5000 matrix with 5 contour levels causes Chrome to crash. Testing on a newer Mac with 16 Gb RAMM shows that the browser may be the limiting factor rather than the RAMM.

Examples

```
require(jsonlite)
exCon(M = volcano)
```

Index

*Topic **package**
exCon-package, [2](#)

*Topic **plot**
exCon, [2](#)

browseURL, [3](#)

contour, [3](#)
contourLines, [3](#)

exCon, [2](#)
exCon-package, [2](#)