Package 'xgobi'

February 20, 2015

Version 1.2-15

Date 2012-11-01

Title Interface to the XGobi and XGvis programs for graphical data analysis

Maintainer Martin Maechler <maechler@stat.math.ethz.ch>

Author Martin Maechler, originally packaged for R by Kurt Hornik <Kurt.Hornik@R-project.org>, based on the S code in the XGobi distribution. Windows port based on this and earlier work by Brian Ripley <ripley@stats.ox.ac.uk>.

Description Interface to the XGobi and XGvis programs for graphical data analysis.

SystemRequirements The standalone program xgobi must be installed additionally, see file README, or INSTALL.windows under Windows

NOTE XGobi and XGVis have been superseded by ggobi and ggvis, available from www.ggobi.org. The R package Rggobi can also be obtained there.

Repository CRAN

Date/Publication 2012-11-01 12:14:25

NeedsCompilation no

License_restricts_use no

R topics documented:

laser *Bellcore Experimental Laser Data*

Description

This data came from an investigation of an experimental laser at Bellcore. It was a tunable laser, in the sense that both its wavelength and power output were controllable.

Usage

```
data(laser)
```

Format

A data frame with observations on the variables `If` and `Ir` (the currents applied to the front and rear of the laser), and `power` and `lambda` (the output power and wavelength).

Details

Rotation helped the experimental physicists to characterize the laser, which turned out not to be a very good one, due to its unstable operating region.

This data initially came to the statistics research group when Janette Cooper asked Paul Tukey to help her analyze the data she had collected to describe the laser.

morsecodes *Rothkopf Morse Code Data*

Description

A standard data set for Multidimensional Scaling (MDS) obtained by Rothkopf based on the confusion rates observed by exposing subjects to pairs of morse codes.

Usage

```
data(morsecodes)
```

Format

9 data sets used for analyzing the data in XGvis and XGobi.

`morsecodes.raw`	36 x 36 raw data of confusion rates
`morsecodes.dist`	36 x 36 dissimilarity matrix
`morsecodes.pos`	36 x 10 initial configuration
`morsecodes.colors`	36 point colors
`morsecodes.glyphs`	36 point glyphs
`morsecodes.lines`	33 lines

morsecodes.linecolors	33 line colors
morsecodes.row	36 x 2 matrix of (letter, morsecode)
morsecodes.col	10 column names of the initial configuration

Details

The raw data from the XGvis directory may be read as
`mc.raw <- as.matrix(read.table("....xgobi/data_xgvis/morsecodes.raw"))`
`dimnames(mc.raw) <- NULL; storage.mode(mc.raw) <- "integer"`
`morsecodes.raw <- mc.raw`.

The `*.dist` matrix is produced from the raw data by
`mc.sim <- (mc.raw + t(mc.raw))/2 ;  ds <- diag(mc.sim)`
`morsecodes.dist <- rep(ds,36) + rep(ds,rep(36,36)) - 2*mc.sim`,
i.e., $d_{ij} := s_{ii} + s_{jj} - 2s_{ij}$.

Source

Contained in the 'data_xgvis' subdirectory of the XGobi and XGvis source bundle, available via http://www.research.att.com/areas/stat/xgobi/index.html#download.

References

A. Buja, D. F. Swayne, M. Littman, & N. Dean (1998). *XGvis: Interactive Data Visualization with Multidimensional Scaling*. http://www.research.att.com/areas/stat/xgobi/xgvis98.ps.gz.

Examples

```
data(morsecodes)
ls.str(pat="^morsecode")
morsecodes.row # remember what you learned in...
## real row names
(mc.row <- paste(morsecodes.row[,1], morsecodes.row[,2]))[1:8]

image(1:36, 1:36, morsecodes.raw, main="`morsecodes' raw confusion rates")
text(1:36,1:36, morsecodes.row[,1])

##--> help(xgvis) for running multidimensional scaling (MDS) and XGobi on these
```

PaulKAI *Occurences of "kai" in 10 Epistels of Apostel Paul*

Description

These are the numbers of sentences which have 0, 1, 2, or 3 and more (3+) occurrences of the greek word "kai" (which means "and" and more) in 10 epistles of Apostel Paul, see the reference.

Usage

```
data(PaulKAI)
```

Format

A 10×4 matrix with proper dimnames, see the examples below.

Note

One theological question is about the authorship of "Hebrews" (the last epistle in our matrix), so one might be interested in its "kai pattern" compared to, e.g. "Romans".

References

Morton, A. Q. (1965)
The authorship of Greek prose (with discussion).
Journal of the Royal Statistical Society, Series A, **128**, 169–233.

Posted to S-news by Jim Ramsay, see quadplot.

See Also

quadplot for which this data set was used as illustration.

Examples

```
data(PaulKAI)
rownames(PaulKAI) # the ten epistles researched:
##>  [1] "Rom" "Co1" "Co2" "Gal" "Phi" "Col" "Th1" "Ti1" "Ti2" "Heb"
PaulKAI # the 10 x 4 count table
mosaicplot(PaulKAI)
quadplot(PaulKAI)
```

---

quadplot *Tetrahedral Display for Four-Category Proportions using XGobi*

Description

Four-category proportions are visualized as points inside a tetrahedron, using xgobi.

Usage

```
quadplot(mat4,
         pointlabs  = rownames(mat4),
         vertexlabs = paste(1:4),
         normalize  = median(abs(c(mat4))) > 1)
```

Arguments

mat4	matrix with 4 columns containing the data
pointlabs	character array of labels for rows of data; by default it is the row number as a string.
vertexlabs	character array of length 4 of labels for the vertices; by default it is the column number as a string.
normalize	logical variable indicating whether or not to force each row of data to have unit sum before display, default is false.

Details

The set of all four-category proportions, or, alternatively, probability measures on finite probability fields with 4 atomic events, is the set of nonnegative 4-vectors whose components sum up to 1. The function quadplot uses xgobi to represent such vectors graphically as points inside a tetrahetron with height 1: the four components of the vector are the distances of the point to each of the sides of the tetrahetron. Each vertex of the tetrahedron corresponds to the degenerate probability distribution in which one of the atomic events has probability 1 and the others have probability 0. The labels of these vertices indicate the event which has probability 1.

Author(s)

(port to R) Hans Ehrbar <ehrbar@econ.utah.edu> and Martin Maechler (with explicit permission from Jim Ramsay)

References

quadplot was posted by Jim Ramsay <ramsay@psych.mcgill.ca> to S-news on Fri, 21 May 1993 14:03:15 EDT.

Examples

```
data(PaulKAI)
quadplot(PaulKAI, normalize = TRUE)
```

---

reggeom *Geometry of Regression with Two Regressors*

Description

Using XGobi for visualising the geometry of regression with two explanatory variables.

The function reggeom has exactly the same arguments as xgobi(..), and it simply calls xgobi, but it has different default values for the arguments than the defaults of xgobi itself.

Usage

```
reggeom(matrx = matrix(c(0, 5780, -1156, 3468, 3468, 3468,
    -867, 4335, 0, 0, -612, 4080, 5440, 2652, 3468, 3420, 3468,
    0, 0, 4624, 3468, 3468, 0, 3468, 0, 3468, 4624, 2448, 1020,
    1360, 3264, 3264, 3456, 3456, 0, 0, 0, 4624, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0), nrow = 17, ncol = 3),
    collab = c("U", "V", "W"),
    rowlab = c("o", "x1", "x2", "y", letters[2:8], "k", "m", "p", "q", "r", "s"),
    colors = NULL, glyphs = NULL, erase = NULL,
    lines = matrix(c(1,  6, 8, 1, 11, 7, 1, 1, 5, 6,
                     6, 15, 17, 8, 5, 9, 1, 9, 10,
                    6, 8, 2, 11, 7, 3, 4, 5, 4, 4,
                      15, 17, 5, 5, 9, 7, 9, 10, 3),
                   nrow = 19, ncol = 2),
    linecolors = c("red", "yellow", "yellow", "yellow", "yellow",
        "yellow", "orchid", "green", "green", "red", "skyblue",
        "skyblue", "skyblue", "white", "white", "white", "slateblue",
        "slateblue", "slateblue"),
    resources = c("*showLines: True", "*showAxes: False", "*showPoints: False",
        "*XGobi*PlotWindow.height: 500",
        "*XGobi*PlotWindow.width: 500", "*XGobi*VarPanel.width: 50"),
    title = "Regression Geometry", vgroups = c(1, 1, 1), std = "msd",
    nlinkable = NULL, subset = NULL, display = NULL)
```

Arguments

matrx	the default dataset is a matrix with three columns. The rows represent the dependent and the two independent variables, as well as fitted values and residuals in the regression on one or both regressors, and other auxiliary variables. Since the matrix has three columns, each variable is represented as a vector in 3-dimensional space.
collab	column labels for matrx, by default "U", "V", and "W", not very meaningful since the columns represent oblique directions in n-dimensional space.
rowlab	character vector of labels for the variables; by default, "x1" and "x2" for the independent and "y" for the dependent variable, "o" for the origin, and other letters for the auxiliary variables.
colors	as in xgobi all points are of the same color.
glyphs	as in xgobi all points are drawn with the same glyph.
erase	as in xgobi no points will be erased.
lines	the default lines argument displays some of the data in matrx as straight lines. The user may want to substitute different lines in order to emphasize or deemphasize certain relationships, as in the example given below.
linecolors	The default line colors are: **purple** for the dependent variable, **yellow** for the two independent variables, **green** for fitted values and residuals in the full regression,

red for fitted values and residuals in the regression on the first independent variable only, and

light blue ,

dark blue , and

white for auxiliary lines.

`resources` by default, points and axes are not shown; only lines are.

`title` by default, `"Regression Geometry"`

`vgroups` by default, all three variables are in the same group.

`std` by default, the view is centered on the mean of the data.

`nlinkable, subset, display`
the same as in `xgobi`.

Details

If called without arguments, `reggeom` loads a dataset which represents the geometry of regression with two explanatory variables. The idea is to place the dataset into the rotation view in order to get an intuition of the geometry involved. `reggeom` should only then be called with arguments if specific built-in defaults must be overriden.

The explanatory variables are x1=(5,0,0) and x2=(-1,4,0), and the target (dependent) variable is y=(3,3,4). However all coordinates are multiplied by 1156, with the effect that all the points passed as arguments to `xgobi` have integer coordinates.

Value

As in the call of `xgobi`, the UNIX `status` upon completion, i.e. `0` if ok.

Side Effects

As in `xgobi`.

author

Hans Ehrbar `<ehrbar@econ.utah.edu>`

References

`reggeom` can be considered a 3-dimensional visualization of the figures in Davidson, R. and MacKinnon, J. G. (1993) *Estimation and Inference in Economics*, Oxford University Press, p. 22.

The chapter "Additional Regressors" in Hans Ehrbar's on-line econometrics class notes `http://www.econ.utah.edu/ehrbar/ecmet.pdf` uses `reggeom` for teaching and has several exercise questions about it.

See Also

`xgobi`

Examples

```
reggeom()

## The arguments given in this example are modifications of the default,
## some lines dropped, some added, some line colors changed,
## in order to emphasize the geometry of backfitting.
reggeom(
      lines= cbind(c(1,6,8,1,11,7,1,1,6,6,15,17,8,5,9, 5,6,14,15,16,14,15,5),
                   c(6,8,2,11,7,3,4,5,4,15,17,5,5,9,7,11,14,15,16,17,4,4,4)),
      linecolors=c("red", rep("yellow",5), "orchid", "green",
                   "slateblue", rep("skyblue",3), rep("white",3), "skyblue",
                   rep("red",4), rep("slateblue", 2), "green"),
      title="Regression Geometry - Backfitting")
```

---

xgobi *XGobi: Dynamic Graphics for Data Analysis*

Description

Dynamic graphics, including brushing, rotation, grand tour, projection pursuit, slicing. Most effectively used when called more than once on same data, which then allows linked plots. Brushing with several glyphs and colors is supported. (On monochrome displays, only glyphs can be used.)

Usage

```
xgobi(matrx,
      collab = colnames(matrx),
      rowlab = rownames(matrx),
      colors = NULL, glyphs = NULL, erase = NULL,
      lines = NULL, linecolors = NULL, resources = NULL,
      title = deparse(substitute(matrx)),
      vgroups = NULL, std = "mmx",
      nlinkable = NULL, subset = NULL, display = NULL,
      multi = TRUE,
      keep = FALSE, fprefix = "xgobi-")

xgobi.colors.default
```

Arguments

matrx	numeric n * p matrix or data.frame.
collab	character vector of p column labels (defaulting to those of matrx); if no default exists, xgobi constructs its own ("Var1",....).
rowlab	character vector of n row labels (defaulting to those of matrx); if no default exists, xgobi constructs its own (numbers 1:n).
colors	Optional character vector, used to supply initial point colors to be used; the default is that all points are the same color. Details, see below.

`glyphs` Optional integer vector, used to supply glyphs to be used on startup; the default is that all points are drawn with the same glyph.

Glyphs have been coming as six different types (plus, X, open and filled rectangle, open and filled circle) in five different sizes, plus "point", giving 31 available glyphs.

`erase` Optional integer vector of length equal to the number of rows in the data and composed of 1s and 0s. A 1 in position i specifies that point i should be erased. The default is a vector of 0s.

`lines` Optional integer matrix, n by 2, which specifies by row number pairs of points to be connected by line segments. The default connecting line matrix connects each point to the one that follows it in the data; that is, (1 2), (2 3), (3 4), ..., (n-1, n).

`linecolors` Optional integer vector, of length n where n is the number of lines specified by the 'lines' argument. It is used to supply line colors to be used on startup; the default is for all the lines to be drawn in the standard foreground color.

`resources` Optional character vector created by clicking on the "Save Resources" button in XGobi (if this XGobi was initiated during an R session).

`title` Optional character string which defines the `-title` argument used by X. Defaults to the name (expression) of the current `matrx` argument. See documentation for xgobi, or for X.

`vgroups` Optional integer vector, used to assign columns to groups for transformation and axis scaling. This vector must contain one integer for each variable. Columns to be grouped together should share the same integer. Default is the vector `1:(ncol(matrx))`.

`std` Optional string; which standardization of view to use. Default is `"mmx"`, minimum-maximum scaling, in which the view is centered at the midpoint of the data, and all the data fits inside the plotting window. Alternatives are `"msd"`, in which the plot is centered at the mean of the data, or `"mmd"` in which the plot is centered at the median. In those two cases, the view is standardized using the largest distance.

`nlinkable` Optional integer scalar, the number of rows to be used in linking of brushing and identification; the default is for all rows to be used. This feature can be used to link ordinary scatterplots with plots that have some decorations requiring additional points, such as clustering trees.

`subset` Optional integer scalar, the number of rows to be included in the initial display. That is, all data will be read in, but an initial random sample will be drawn for display. Use the Subset panel on the Tools Menu to select a new subset during the session.

`display` Optional character string, identifying the monitor on which to display the xgobi window. The default is `"machine:0.0"` where `machine` is the name of the user's workstation. See documentation for xgobi or for X.

`multi` logical, indicating if the `xgobi` process should be run multi-tasking with R. If true, control returns to the R command prompt after 3 seconds.

`keep` logical, indicating if the temporary files should be kept (e.g. for calling the `xgobi` program outside R)

`fprefix` character string for the **f**ile name **prefix** to be used for temporary files.

Details

`xgobi.colors.default` is the vector of the ten default brush colors from which to choose by the `colors` argument.

Note that this sef of default brush colors can be modified by a (site or user) specific '`app-defaults`' file, or directly by `xgobi(*, resources = ..)`, redefining (`*brushColor`n (with n from `0:9`).

A warning is issued if `colors` contains strings not in the `brushColor` resources.

Value

The UNIX `status` upon completion, i.e. `0` if ok.

Side Effects

The R function `xgobi` executes a call to the C program of the same name, an interactive statistical graphics program which runs under the X Window System, and returns control of the R command line to the user.

XGobi can be used to create vectors of brushing information and rotation coefficients; see the documentation for XGobi for details.

CONTACT

(xgobi main program): D. F. Swayne <dfs@research.att.com>

Author(s)

of R port: Kurt Hornik and Martin Maechler <maechler@stat.math.ethz.ch>

References

http://www.research.att.com/areas/stat/xgobi/,
http://www.public.iastate.edu/~dicook/

See Also

`xgvis` which uses `xgobi` for interactive MDS.

Examples

```
data(laser)
xgobi(laser)

Xdir <- file.path(dirname(tempfile()), "xgobi")

dir.create(Xdir)
xgobi(laser, colors = xgobi.colors.default[c(1,3,5,7,9,10)[as.factor(laser$ Ir)]],
      glyphs = c(23,8)[1+(laser$lambda > 1576)],
      keep = TRUE, fprefix="xgobi/L-")
file.info(list.files(Xdir, full=TRUE))[, c(1,3,4)] # >> Files "L-laser..."
## remove manually when finally unused:
```

```
unlink(Xdir, recursive = TRUE)

##>>> see also the morsecodes example in  help(xgvis) <<<
##                 ------------------     -----------
```

---

xgvis *XGvis: Interactive Multidimensional Scaling Using XGobi for Display*

Description

R interface to XGvis, an interactive multidimensional scaling (MDS) program that consists of a control panel to manipulate the parameters of the MDS stress function and an xgobi window for data display. It can be used either for visualization of dissimilarity data, for dimension reduction, or for graph layout. Graph layout is usually done in 2D, but xgvis allows layouts in arbitrary dimensions, 3D being the default. It permits missing values, which can be used to implement multidimensional unfolding.

Usage

```
xgvis(dmat = NULL,
      edges= NULL,
      pos  = NULL,
      rowlab = colnames(dmat),
      colors = NULL, glyphs = NULL,
      erase = NULL, lines = NULL, linecolors = NULL,
      resources = NULL, display = NULL,
      multi = TRUE,
      keep = FALSE, fprefix = "xgvis-")
```

Arguments

dmat	numeric $n \times n$ distance matrix.
edges	$n \times 2$ or $n \times 3$ matrix of specifications for the pattern of line segments which connect pairs of points. Must contain at least two numbers per line. The first two numbers represent the row numbers of the two points that should be connected. (This is exactly like the structure of a the lines argument of xgobi.) In addition, if a third number is present, it is taken to be an edge weight. If edges is specified and dmat not, then the distance matrix is computed from edges, with each edge representing a distance of one.
pos	Starting positions: an n * p matrix. If pos is specified and dmat not, the distance matrix is computed from pos.
rowlab	character vector of n row labels (defaulting to those of dmat); if no default exists, xgobi constructs its own (numbers 1:n).
colors	optional character vector supplying initial point colors to be used; see xgobi.
glyphs	integer vector, used to supply glyphs to be used on startup, see xgobi.

erase	Optional integer vector of length equal to the number of rows in the data and composed of 1s and 0s. A 1 in position i specifies that point i should be erased. The default is a vector of 0s.
lines	Optional integer matrix, n by 2, which specifies by row number pairs of points to be connected by line segments. If lines are specified, then the edges is used to create the distance matrix but lines is used to draw the edges.
linecolors	Optional integer vector, of length n where n is the number of lines specified by the lines argument. It is used to supply line colors to be used on startup; the default is for all the lines to be drawn in the standard foreground color.
resources	Optional character vector created by clicking on the "Save Resources" button in XGobi.
display	Optional character string, identifying the monitor on which to display the xgvis window. The default is "machine:0.0" where machine is the name of the user's workstation. See documentation for X.
multi	logical, indicating if the xgobi process should be run multi-tasking with R. If true, control returns to the R command prompt after 3 seconds.
keep	logical, indicating if the temporary files should be kept (e.g. for calling the xgobi program outside R)
fprefix	character string for the **f**ile name **prefix** to be used for temporary files.

Value

The UNIX status upon completion, i.e. 0 if ok.

Side Effects

The xgvis R function executes a call to the C program of the same name, and returns control of the R command line to the user.

CONTACT

D. F. Swayne <dfs@research.att.com>

Author(s)

of R port: Kurt Hornik and Martin Maechler <maechler@stat.math.ethz.ch>

References

http://www.research.att.com/areas/stat/xgobi/,
http://www.public.iastate.edu/~dicook/

See Also

xgobi.

Examples

```
data(morsecodes) ## from the XGobi/XGvis data, see  ?morsecodes
mc.row <- paste(morsecodes.row[,1],morsecodes.row[,2])

xgvis(dmat = morsecodes.dist,
      pos = morsecodes.pos,
      rowlab = mc.row,
      colors = morsecodes.colors,
      glyphs = morsecodes.glyphs,
      lines = morsecodes.lines,
      linecolors = morsecodes.linecolors)

##>   2) Show lines by hitting "l" with the mouse over the plot.
##>   3) Examine morsecode labels by hitting "i" and mousing around on the plot.
##>   3b) Press "r" (on the plot) to switch 3D rotation in xgobi.
##>   4) Run MDS in 3D by clicking "Run MDS" (in xgvis).
##>   5) Speed up the optimization by increasing the "Stepsize" with the slider.
##>      The "Stress function" value may go as low as 0.1925 (MM).
##>   6) When the optimization calms down, click "Run MDS" to toggle MDS off.
##>   7) Rotate the MDS configuration in 3D {by "r" with mouse over plot}.
##>   8) Increase the rotation speed with the slider in the top left and
##>      control the rotation direction by dragging the mouse on the plot.
##>   9) You can check out the initial configuration by

## In order to have no color warning :
Mcolors <- unique(morsecodes.colors)
(Mcolors <- paste("*brushColor", 0:(length(Mcolors)-1),": ", Mcolors, sep=""))

xgobi(morsecodes.pos, collab = morsecodes.col, rowlab = mc.row,
      colors = morsecodes.colors,
      glyphs = morsecodes.glyphs,
      lines  = morsecodes.lines,
      linecolors = morsecodes.linecolors,
      resources= c("*showLines: True", Mcolors))

##>      This XGobi window will be linked with
##>      the XGvis window for glyph-color brushing and labeling.
```

Index