Package ‘gpairs’

February 20, 2015

Version 1.2

Date 2014-03-09

Title gpairs: The Generalized Pairs Plot

Author John W. Emerson and Walton A. Green

Maintainer John W. Emerson <john.emerson@yale.edu>

Imports grid, barcode, lattice, vcd, MASS, colorspace

Enhances YaleToolkit

Description Produces a generalized pairs (gpairs) plot.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2014-03-09 18:45:06

R topics documented:

`gpairs` *Generalized Pairs Plots*

Description

Produces a matrix of plots showing pairwise relationships between quantitative and categorical variables in a complex data set.

Usage

```
gpairs(x,
       upper.pars = list(scatter = "points",
                         conditional = "barcode",
                         mosaic = "mosaic"),
       lower.pars = list(scatter = "points",
                         conditional = "boxplot",
                         mosaic = "mosaic"),
       diagonal = "default",
       outer.margins = list(bottom = unit(2, "lines"),
                            left = unit(2, "lines"),
                            top = unit(2, "lines"),
                            right = unit(2, "lines")),
       xylim = NULL,
       outer.labels = NULL, outer.rot = c(0, 90), gap = 0.05,
       buffer = 0.02, reorder = NULL, cluster.pars = NULL,
       stat.pars = NULL, scatter.pars = NULL,
       bwplot.pars = NULL, stripplot.pars = NULL, barcode.pars=NULL,
       mosaic.pars = NULL, axis.pars = NULL, diag.pars = NULL,
       whatis = FALSE)

corrgram(x)
```

Arguments

`x`	a data frame (or matrix the relationships between whose columns are to be examined). Any combination of quantitative and categorical variables is acceptable.
`upper.pars`	see `Details`
`lower.pars`	see `Details`
`diagonal`	by default, the diagonal from the top left to the bottom right is used for displaying the variable names (and, in our version, the marginal distributions of the variables); `diagonal="other"` will place the diagonal running from the top right down to the bottom left.
`outer.margins`	a list of length 4 with units as components named bottom, left, top, and right, giving the outer margins; the default uses two lines of text. A vector of length 4 with units (ordered properly) will work, as will a vector of length 4 with numeric values (interpreted as lines).
`xylim`	optionally specify a single range to be used as `xlim` and `ylim` where appropriate. Note that if this option causes cropping, it will fail to work in barcode panels.
`outer.labels`	the default is `NULL`, for alternating axis labels around the perimeter. If `"all"`, all labels are printed, and if `"none"` no labels are printed.
`outer.rot`	a 2-vector (x, y) rotating the top/bottom outer labels `x` degrees and the left/right outer labels `y` degrees. Only works for categorical labels of boxplot and mosaic panels.
`gap`	the gap between the tiles; defaulting to 0.05 of the width of a tile.

`buffer` the fraction by which to expand the range of quantitative variables to provide plots that will not truncate plotting symbols. Defaults to 0 percent of range currently.

`reorder` currently only support for the string "cluster", which reorders the columns according to the output of `hclust`. Note that factors are coerced to numbers by replacing them with integers, which implicitly assumes what is probably an arbitrary ordering.

`cluster.pars` a list with two elements named `dist.method` and `hclust.method`. These are passed respectively to `dist` and `hclust`. `NULL` is equivalent to `list(dist.method = "euclidean", hcl`

`stat.pars` `NULL` is equivalent to `list(fontsize = 7, signif = 0.05, verbose = FALSE, use.color = TRUE, r` `stat.pars$verbose` can be `TRUE` (providing 5 statistics), `FALSE` (providing 2 statistics), or `NA` (nothing). The string `missing` is used in summaries where there are missing values; `fontsize` and `just` control the size and justification of the text summaries (see `grid.text` and `gpar`. The `use.color=FALSE` option provides an alternative summary of the strength of the correlation (see Green and Hickey (2006)). This is only used with `scatter="stats")` in `upper.pars` and `lower.pars`.

`scatter.pars` `NULL` is equivalent to `list(pch = 1, size = unit(0.25, "char"), col = "black", frame.fill = N`

`bwplot.pars` `NULL`, passed to `bwplot` for producing boxplots.

`stripplot.pars` `NULL` is equivalent to `list(pch = 1, size = unit(0.5, 'char'), col = 'black', jitter = FALSE)`

`barcode.pars` `NULL` is equivalent to `list(nint = 0, ptsize = unit(0.25, "char"), ptpch = 1, bcspace = NULL,`

`mosaic.pars` `NULL`. Currently `shade`, `gp_labels`, `gp`, and `gp_args` are passed through to `strucplot` for producing mosaic tiles.

`axis.pars` `NULL` is equivalent to `list(n.ticks = 5, fontsize = 9)`.

`diag.pars` `NULL` is equivalent to `list(fontsize = 9, show.hist = TRUE, hist.color = 'black')`.

`whatis` default is `FALSE`; `TRUE` returns `whatis(x)`.

Details

In some cases, the graphics device can not be resized after production of the plot because of the way rotation of barcodes is performed.

`upper.pars` and `lower.pars` are lists possibly containing named elements `'scatter'`, `'conditional'` and `'mosaic'`. Each element of the list is a string implementing the following options: `scatter` = exactly one of (`'points', 'lm', 'ci', 'symlm', 'loess', 'corrgram', 'stats', 'qqplot'`); `'conditional'` = exactly one of (`'boxplot', 'stripplot', 'barcode'`); `mosaic='mosaic'` (only option currently implemented).

`corrgram()` is just a wrapper to `gpairs()` producing a 'corrgram' in the style of Michael Friendly.

Value

If `whatis=TRUE`, the value is a data frame containing variable names, types, numbers of missing values, numbers of distinct values, precisions, maxima and minima.

Author(s)

John W. Emerson, Walton Green; thanks to Michael Friendly for augmenting the functionality with arguments to `strucplot`.

References

Emerson, John W. (1998) "Mosaic Displays in S-PLUS: A General Implementation and a Case Study." *Statistical Computing and Graphics Newsletter* Vol. 9,No. 1, 1998.

Basford, K. E. and J. W. Tukey (1999) *Graphical Analysis of Multiresponse Data: Illustrated with a Plant Breeding Trial.*

Friendly, M. (2000). *Visualizing Categorical Data.* SAS Press.

Friendly, M., 2002, "Corrgrams: Exploratory displays for correlation matrices." *American Statistician* 56(4), 316–324.

Green, W. A. (2006) "Loosening the CLAMP: An exploratory graphical approach to the Climate Leaf Analysis Multivariate Program." *Palaeontologia Electronica* 9(2):9A.

See Also

`pairs`, `splom`, `mosaicplot`, `strucplot`, `bwplot`, `barcode`, `stripplot`.

Examples

```
allexamples <- FALSE

y <- data.frame(A=c(rep("red", 100), rep("blue", 100)),
                B=c(rnorm(100),round(rnorm(100,5,1),1)), C=runif(200),
                D=c(rep("big", 150), rep("small", 50)),
                E=rnorm(200))
gpairs(y)

data(iris)
gpairs(iris)
if (allexamples) {
  gpairs(iris, upper.pars = list(scatter = 'stats'),
         scatter.pars = list(pch = substr(as.character(iris$Species), 1, 1),
                             col = as.numeric(iris$Species)),
         stat.pars = list(verbose = FALSE))
  gpairs(iris, lower.pars = list(scatter = 'corrgram'),
         upper.pars = list(conditional = 'boxplot', scatter = 'loess'),
         scatter.pars = list(pch = 20))
}

data(Leaves)
gpairs(Leaves[1:10], lower.pars = list(scatter = 'loess'))
if (allexamples) {
  gpairs(Leaves[1:10], upper.pars = list(scatter = 'stats'),
         lower.pars = list(scatter = 'corrgram'),
         stat.pars = list(verbose = FALSE), gap = 0)
  corrgram(Leaves[,-33])
```

```
}

runexample <- FALSE
if (runexample) {
  data(NewHavenResidential)
  gpairs(NewHavenResidential)
}
```

Leaves *Morphological descriptions of leaf floras*

Description

Measurements of the percentages of leaves in 31 morphological (or architectural) categories found in 245 leaf floras from 4 studies.

Usage

```
data(Leaves)
```

Format

A data frame with 245 observations on the following 33 variables.

`Lobd` a numeric vector giving percentage Lobed leaves

`Entr` a numeric vector giving percentage Entire leaves

`TReg` a numeric vector giving percentage leaves with Regular Teeth

`TCls` a numeric vector giving percentage leaves with Close Teeth

`TRnd` a numeric vector giving percentage leaves with Round Teeth

`TAcu` a numeric vector giving percentage leaves Acute Teeth

`TCmp` a numeric vector giving percentage leaves with Compound Teeth

`ZNan` a numeric vector giving percentage Nanophyll leaves

`ZLe1` a numeric vector giving percentage Leptophyll1 leaves

`ZLe2` a numeric vector giving percentage Leptophyll2 leaves

`ZMi1` a numeric vector giving percentage Microphyll1 leaves

`ZMi2` a numeric vector giving percentage Microphyll2 leaves

`ZMi3` a numeric vector giving percentage Microphyll3 leaves

`ZMe1` a numeric vector giving percentage Megaphyll1 leaves

`ZMe2` a numeric vector giving percentage Megaphyll2 leaves

`ZMe3` a numeric vector giving percentage Megaphyll3 leaves

`AEmg` a numeric vector giving percentage leaves with Emarginate Apexes

`ARnd` a numeric vector giving percentage leaves with Round Apexes

AAcu a numeric vector giving percentage leaves with Acute Apexes

AAtn a numeric vector giving percentage leaves with Attenuate Apexes

BCor a numeric vector giving percentage leaves with Cordate Bases

BRnd a numeric vector giving percentage leaves with Round Bases

BAcu a numeric vector giving percentage leaves with Acute Bases

Rlt1 a numeric vector giving percentage leaves with aspect ratio less than 1:1 (i.e. wider than long)

Rb12 a numeric vector giving percentage leaves with aspect ratio between 1:1 and 1:2

Rb23 a numeric vector giving percentage leaves with aspect ratio between 1:2 and 1:3

Rb34 a numeric vector giving percentage leaves with aspect ratio between 1:3 and 1:4

Rgt4 a numeric vector giving percentage leaves with aspect ratio between greater than 1:4

SObo a numeric vector giving percentage Obovate leaves

SElp a numeric vector giving percentage Elliptical leaves

SOvt a numeric vector giving percentage Ovate leaves

MAT a numeric vector giving mean annual temperature in degrees Centigrade

Study a factor with levels Wolfe173 Jacobs Gregory Kowalski

Details

Data consists of a data frame with 245 rows and 33 columns (variables). The rows represent floras (collections of plants from a defined locality); the first 31 variables are percentages of leaves in each flora in each of 31 morphological categories; the 32nd variable is mean annual temperature of the area from which the floras was collected in degrees C, and the 32nd is a factor indicating which of 4 published studies the floras come from. See cited publications for more details.

Source

Green, W. A. (2006) Loosening the CLAMP: An exploratory graphical approach to the Climate Leaf Analysis Multivariate Program *Palaeontologia Electronica* 9(2):9A.

http://www.palaeo-electronica.org/2006_2/clamp/index.html

http://www.open.ac.uk/earth-research/spicer/CLAMP/Physg3ar.xls (the climatic variables)

http://www.open.ac.uk/earth-research/spicer/CLAMP/MET3AR.xls (the morphological leaf scores)

References

Gregory-Wodzicki, K. M. (2000) Relationships between leaf morphology and climate, Bolivia: implications for estimating paleoclimate from fossil floras. *Paleobiology* 26(4):668–688.

Jacobs, B. F. (1999) Estimation of rainfall variables from leaf characters in tropical Africa. *Palaeogeography, Palaeoclimatology, Palaeoecology* 145:231–250.

Jacobs, B. F. (2002) Estimation of low-latitude paleoclimates using fossil angiosperm leaves: examples from the Miocene Tugen Hills, Kenya. *Paleobiology* 28(3):399–421.

Kowalski, E. A. (2002) Mean annual temperature estimation base on leaf morphology: a test from tropical South America. *Palaeogeography, Palaeoclimatology, Palaeoecology* 188:141–165.

Wolfe, J.A., (1993), A method of obtaining climatic parameters from leaf assemblages. *U.S. Geological Survey Bulletin* 2040, 73 pp.

Examples

```
data(Leaves)
## maybe str(Leaves) ; plot(Leaves) ...
```

Index