

Heatplus

November 11, 2009

R topics documented:

<code>cutplot.dendrogram</code>	1
<code>heatmap_2</code>	2
<code>heatmap_plus</code>	4
<code>picketplot</code>	6
<code>RGBColVec</code>	8

Index	10
--------------	-----------

`cutplot.dendrogram` *Plot Subtrees of a Dendrogram in Different Colors*

Description

Plot a dendrogram, cut the tree at a given height, and draw the resulting subtrees in different colors.

Usage

```
cutplot.dendrogram(x, h, cluscol, leaflab = "none", horiz = FALSE, lwd = 3, ...)
```

Arguments

<code>x</code>	a dendrogram.
<code>h</code>	the height at which the dendrogram is cut.
<code>cluscol</code>	the colors used for the subtrees; defaults to <code>rainbow</code> .
<code>leaflab</code>	indicates how leaf labels are to be drawn< defaults to <code>'perpendicular'</code> .
<code>horiz</code>	logical indicating whether to plot the dendrogram horizontally or vertically.
<code>lwd</code>	the line width used for the color subtrees.
<code>...</code>	arguments to <code>plot.dendrogram</code> .

Details

This routine makes use of the functions `plot.dendrogram` and `plotNode` in package `stats`.

Author(s)

Alexander Ploner <Alexander.Ploner@ki.se>

See Also

[as.dendrogram](#)

Examples

```
data(swiss)
cc = as.dendrogram(hclust(dist(swiss)))
cutplot.dendrogram(cc, h=80)
```

heatmap_2

Display Data as Heatmap

Description

This function displays an expression data matrix as a heatmap. It is based on an old version of `heatmap` in the `stats` package, but offers more flexibility (e.g. skipping dendrograms, skipping row/column labelling, adding a legend).

Usage

```
heatmap_2(x, Rowv, Colv, distfun = dist, hclustfun = hclust, add.expr,
          scale = c("row", "column", "none"), na.rm = TRUE,
          do.dendro = c(TRUE, TRUE), legend = 0, legfrac = 8,
          col = heat.colors(12), trim, ...)
```

Arguments

<code>x</code>	the numerical data matrix to be displayed.
<code>Rowv</code>	either a dendrogram or a vector of reordering indexes for the rows.
<code>Colv</code>	either a dendrogram or a vector of reordering indexes for the columns.
<code>distfun</code>	function to compute the distances between rows and columns. Defaults to <code>dist</code> .
<code>hclustfun</code>	function used to cluster rows and columns. Defaults to <code>hclust</code> .
<code>add.expr</code>	Expression to be evaluated after the call to <code>image</code> . See Details .
<code>scale</code>	indicates whether values should be scaled by either by row, column, or not at all. Defaults to <code>row</code> .
<code>na.rm</code>	logical indicating whether to remove NAs.
<code>do.dendro</code>	logical vector of length two, indicating (in this order) whether to draw the row and column dendrograms.
<code>legend</code>	integer between 1 and 4, indicating on which side of the plot the legend should be drawn, as in <code>mtext</code> .
<code>legfrac</code>	fraction of the plot that is taken up by the legend; larger values correspond to smaller legends.
<code>col</code>	the color scheme for <code>image</code> . The default sucks.
<code>trim</code>	Percentage of values to be trimmed. This helps to keep an informative color scale, see Details .
<code>...</code>	extra arguments to <code>image</code> .

Details

With all parameters at their default, this gives the same result as a very old version of `heatmap` that was the base for the modifications. All parameters of the same name have the same function as in `heatmap`, though `add.expr`, which can be used for adding graphical elements after the call to `image`, will probably not produce useful results. Note also that row- and column labels are optional, i.e. if the corresponding `dimname` of `x` is `NULL`, no labels are displayed.

Note that setting `Rowv` or `Colv` to `NA` completely suppresses re-ordering of rows or columns as well as the corresponding dendrogram. Setting both to `NA` works basically like `image` (though you can still add a legend).

Setting `trim` to a number between 0 and 1 uses equidistant classes between the (`trim`)- and (1-`trim`)-quantile, and lumps the values below and above this range into separate open-ended classes. If the data comes from a heavy-tailed distribution, this can save the display from putting too many values into too few classes.

Value

Same as `heatmap` with `keep.dendro=FALSE`: an invisible list giving the reordered indices of the row- and column-elements as `elements$rowInd` and `colInd`.

Author(s)

Original by Andy Liaw, with revisions by Robert Gentleman and Martin Maechler.

Alexander Ploner for this version.

See Also

[heatmap](#), [hclust](#), [heatmap_plus](#)

Examples

```
# create data
mm = matrix(rnorm(1000, m=1), 100, 10)
mm = cbind(mm, matrix(rnorm(2000), 100, 20))
mm = cbind(mm, matrix(rnorm(1500, m=-1), 100, 15))
mm2 = matrix(rnorm(450), 30, 15)
mm2 = cbind(mm2, matrix(rnorm(900, m=1.5), 30, 30))
mm=rbind(mm, mm2)
colnames(mm) = paste("Sample", 1:45)
rownames(mm) = paste("Gene", 1:130)

# similar to base heatmap
heatmap_2(mm)

# remove column dendrogram
heatmap_2(mm, do.dendro=c(TRUE, FALSE))

# add a legend under the plot
heatmap_2(mm, legend=1)
# make it smaller
heatmap_2(mm, legend=1, legfrac=10)
# ... on the left side
heatmap_2(mm, legend=2, legfrac=10)

# remove the column labels by removing the column names
```

```

colnames(mm)=NULL
heatmap_2(mm, legend=1, legfrac=10)

# truncate the data drastically
heatmap_2(mm, legend=1, legfrac=10, trim=0.1)

```

heatmap_plus	<i>Display an Annotated Heatmap</i>
--------------	-------------------------------------

Description

This function displays an expression data matrix as a heatmap with a column dendrogram. A given clustering will be shown in color. Additionally, a number of binary and interval scaled covariates can be added to characterize these clusters.

Usage

```

heatmap_plus(x, addvar, covariate = NULL, picket.control = list(), h, clus,
             cluscol, cluslabel = NULL, Rowv, Colv, reorder = c(TRUE, TRUE),
             distfun = dist, hclustfun = hclust, scale = c("row", "c",
             "none"), na.rm = TRUE, do.dendro = TRUE, col = heat.col,
             trim, equalize = FALSE, ...)

```

Arguments

<code>x</code>	the numerical data matrix to be displayed.
<code>addvar</code>	data frame with (mostly binary) covariates.
<code>covariate</code>	integer indicating the one column in <code>addvar</code> that is interval scaled.
<code>picket.control</code>	list of option for drawing the covariates, passed to <code>picketplot</code> .
<code>h</code>	height at which to cut the dendrogram, as in <code>cutree</code> ; overrides <code>clus</code> .
<code>clus</code>	an explicit vector of cluster memberships for the columns of <code>x</code> , if no dendrogram is used; ignored if <code>do.dendro=TRUE</code> and <code>h</code> is specified.
<code>cluscol</code>	a vector of colors used to indicate clusters.
<code>cluslabel</code>	labels to designate cluster names.
<code>Rowv</code>	either a dendrogram or a vector of reordering indexes for the rows.
<code>Colv</code>	either a dendrogram or a vector of reordering indexes for the columns.
<code>reorder</code>	logical vector of length two, indicating whether the rows and columns (in this order) should be reordered using <code>order.dendrogram</code> .
<code>distfun</code>	function to compute the distances between rows and columns. Defaults to <code>dist</code> .
<code>hclustfun</code>	function used to cluster rows and columns. Defaults to <code>hclust</code> .
<code>scale</code>	indicates whether values should be scaled by either by row, column, or not at all. Defaults to <code>row</code> .
<code>na.rm</code>	logical indicating whther to remove NAs.
<code>do.dendro</code>	logical indicating whether to draw the column dendrogram.
<code>col</code>	the color scheme for image. The default sucks.

<code>trim</code>	Percentage of values to be trimmed. This helps to keep an informative color scale, see Details .
<code>equalize</code>	logical indicating whether to use the ranks of the data for setting the color scheme; alternative to <code>trim</code> , see Details .
<code>...</code>	extra arguments to <code>image</code> .

Details

This is a heavily modified version of `heatmap_2`, which is a heavily modified version of an old version of `heatmap` in package `stats`, so some of the arguments are described in more detail there. The main distinguishing feature of this routine is the possibility to color a cluster solution, and to add a covariate display.

Covariates are assumed to be binary, coded as 0 and 1 (or `FALSE` and `TRUE` respectively). One of the covariates can be interval scaled, the column index of this variable is supplied via argument `covariate`. The details of the added display are handled by the function `picketplot`.

Setting `trim` to a number between 0 and 1 uses equidistant classes between the (`trim`)- and (1-`trim`)-quantile, and lumps the values below and above this range into separate open-ended classes. If the data comes from a heavy-tailed distribution, this can save the display from putting too many values into too few classes. Alternatively, you can set `equal=TRUE`, which uses an equidistant color scheme for the ranks of the values.

Value

A list with components

<code>rowInd</code>	indices of the rows of the display in terms of the rows of <code>x</code> .
<code>colInd</code>	ditto for the columns of the display.
<code>clus</code>	the cluster indices of the columns of the display.

Author(s)

Original by Andy Liaw, with revisions by Robert Gentleman and Martin Maechler.

Alexander Ploner for the modifications documented [here](#).

See Also

[heatmap_2](#), [heatmap](#), [picketplot](#), [cutplot.dendrogram](#), [RGBColVec](#)

Examples

```
# create data
mm = matrix(rnorm(1000, m=1), 100, 10)
mm = cbind(mm, matrix(rnorm(2000), 100, 20))
mm = cbind(mm, matrix(rnorm(1500, m=-1), 100, 15))
mm2 = matrix(rnorm(450), 30, 15)
mm2 = cbind(mm2, matrix(rnorm(900, m=1.5), 30, 30))
mm=rbind(mm, mm2)
colnames(mm) = paste("Sample", 1:45)
rownames(mm) = paste("Gene", 1:130)
addvar = data.frame(Var1=rep(c(0,1,0), c(10,20,15)),
                    Var2=rep(c(1,0,0), c(10,20,15)),
                    Var3=rep(c(1,0), c(15,30)),
                    Var4=rep(seq(0,1,length=4), c(10,5,15,15))+rnorm(45, sd=0.5))
```

```

addvar[3,3] = addvar[17,2] = addvar[34,1] =NA
colnames(addvar) = c("Variable X", "Variable Y", "ZZ", "Interval")

# the lame default, without clustering
# Labels do not look too hot that way
heatmap_plus(mm)

# without labels, but with cluster
dimnames(mm)=NULL
heatmap_plus(mm, h=40)

# add some covariates, with nice names
heatmap_plus(mm, addvar=addvar, cov=4)

# covariates and clustering
heatmap_plus(mm, addvar=addvar, cov=4, h=20, col=RGBColVec(64), equal=TRUE)

# Clustering without the dendrogram
cc = cutree(hclust(dist(t(mm))), k=5)
heatmap_plus(mm, addvar=addvar, cov=4, clus=cc, do.dendro=FALSE)

```

picketplot

Barplots for Several Binary Variables

Description

Display one or more binary variables by using black bars for presence/validity of a condition, empty space for absence/invalidity, and an extra color for missing values. Additionally, an index plot for one interval scaled variable can be added, possibly with a smoothing function.

This routine is primarily intended for augmenting heatmaps. It might be useful in other contexts, but misses most frills for using it comfortably.

Usage

```

picketplot(x, covariate = NULL, grp = NULL, grpcol, grplabel = NULL,
           add = FALSE, control = list())

```

Arguments

<code>x</code>	a matrix or data frame containing the data.
<code>covariate</code>	the index of the column in <code>x</code> that contains the interval scaled variable, if any.
<code>grp</code>	cluster indices for the rows of <code>x</code> , used for assigning background color.
<code>grpcol</code>	colors corresponding to the clusters.
<code>grplabel</code>	cluster names.
<code>add</code>	logical indicating whether to start a new plot, or whether to add the plot to the existing one.
<code>control</code>	a list of parameters controlling the appearance of the plot, see Details.

Details

The following named list elements can be set to change the appearance of the plot:

boxw the relative width of a marking box.

boxh the relative height of a marking box.

hbuff the horizontal separation around marking boxes; equals half the horizontal distance between two marking boxes.

vbuff ditto for vertical separation.

span passed on to `loess` used for the smoothing curve.

nacol color for missing values of binary variables.

degree if 0, no smoothing line is drawn; otherwise passed on to `loess` used for the smoothing curve.

cex.label the character size for `grplabel`.

Note

The plot looks like a more or less derelict picket fence, and 'picketplot' sounds somewhat like the 'pocketplot' used in geostatistics.

Author(s)

Alexander Ploner <Alexander.Ploner@ki.se>

See Also

[heatmap_plus](#)

Examples

```
# without covariate
mm = cbind(sample(0:1, 42, rep=TRUE), sample(0:1, 42, rep=TRUE))
mm[sample(42, 5), 1] = NA
picketplot(mm)
```

```
# with clustering
cl = rep(1:3, c(10,22,10))
cn = c("Cluster I","Cluster II","Cluster III")
cc = c("lightblue","lightgreen","lightpink") # windows palette
picketplot(mm, grp=cl, grplabel=cn, grpcol=cc)
```

```
# add a covariate; setting the colnames makes the variable labels
mm = cbind(mm, rnorm(42) + cl/2)
colnames(mm) = c("State A","State B", "X")
picketplot(mm, covariate=3,grp=cl, grplabel=cn, grpcol=cc)
```

```
# using extra controls
picketplot(mm, covariate=3,grp=cl, grplabel=cn, grpcol=cc, control=list(nacol="white", de
```

`RGBColVec`*Alternative color schemes*

Description

`RGBColVec` returns a vector of colors that is equally spaced from red through black to green, suitable for heatmaps.

`RainbowPastel` returns a vector of colors like `rainbow`, but more pastelly.

Usage

```
RGBColVec(nrgcols = 12)
```

```
RainbowPastel(n, blanche = 200, ...)
```

Arguments

<code>nrgcols, n</code>	desired number of colors
<code>blanche</code>	the amount of whiteness added; value between 0 and 255
<code>...</code>	extra arguments to <code>rainbow</code>

Value

A character vector of length `nrgcols` or `n` giving the RGB codes for the colors.

Author(s)

`RGBColVec` is based on function `rgcolors.func` in package `sma` by Sandrine Dudoit and Jane Fridlyand.

`RGBColVec` as documented and `RainbowPastel` by Alexander Ploner

See Also

[heat.colors](#)

Examples

```
# A Color Wheel
pie(rep(1,12), col=RGBColVec(12))

# A color wheel in the original rainbow
pie(rep(1,6), col=rainbow(6))

# Pastel
pie(rep(1,6), col=RainbowPastel(6))

# Less whiteness
pie(rep(1,6), col=RainbowPastel(6, blanche=127))

# More steps require less whiteness
pie(rep(1,12), col=RainbowPastel(12, blanche=60))
```



```
# Test your screen & eyes: any differences?  
pie(rep(1,12), col=RainbowPastel(12, blanche=80))
```

Index

*Topic **aplot**

cutplot.dendrogram, 1
picketplot, 6

*Topic **cluster**

cutplot.dendrogram, 1

*Topic **color**

RGBColVec, 8

*Topic **hplot**

cutplot.dendrogram, 1
heatmap_2, 2
heatmap_plus, 4
picketplot, 6

as.dendrogram, 1

cutplot.dendrogram, 1, 5

hclust, 3

heat.colors, 8

heatmap, 3, 5

heatmap_2, 2, 5

heatmap_plus, 3, 4, 7

picketplot, 5, 6

RainbowPastel (*RGBColVec*), 8

RGBColVec, 5, 8