Package 'corrplot'

February 14, 2012

Type Package

Title Visualization of a correlation matrix

Version 0.60

Date 2011-12-06

Author Taiyun Wei

Suggests seriation, Cairo

Maintainer Taiyun Wei <weitaiyun@gmail.com>

Description The corrplot package is a graphical display of a
correlation matrix, confidence interval. It also contains some algorithms to do matrix reordering.

License GPL-2 | GPL-3

LazyLoad yes

URL https://github.com/taiyun/corrplot

BugReports https://github.com/taiyun/corrplot/issues

Repository CRAN

Date/Publication 2011-12-05 18:56:23

R topics documented:

corrplot-package *Visualization of a correlation matrix*

Description

The corrplot package is a graphical display of a correlation matrix, confidence interval. It also contains some algorithms to do matrix reordering.

Details

Package:	corrplot
Type:	Package
Version:	0.60
Date:	2011-12-06
License:	GPL-2 | GPL-3
LazyLoad:	yes

Author(s)

Taiyun Wei

Maintainer:Taiyun Wei <weitaiyun@gmail.com>

References

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

D.J. Murdoch, E.D. Chow (1996). *A graphical display of large correlation matrices*. The American Statistician, 50, 178–180.

See Also

The `plotcorr` function in the `ellipse` package and `corrgram` function in the `corrgram` package has some similarities.

colorlegend *Draw color legend.*

Description

Draw color legend.

Usage

```
colorlegend(colbar, labels, at=NULL,  xlim=c(0,1), ylim=c(0,1),
vertical=TRUE, ratio.colbar = 0.4, lim.segment = NULL,
align=c("c","l","r"), addlabels=TRUE, ...)
```

Arguments

colbar	Vector, color of colbar.
labels	Vector, numeric or character to be written.
at	Numeric vector (quantile), the position to put labels. See examples for details.
xlim	See in plot.
ylim	See in plot.
vertical	Logical, whether the colorlegend is vertical or horizon.
ratio.colbar	The width ratio of colorbar to the total colorlegend (including colorbar, segments and labels).
lim.segment	Vector (quantile) of length 2, the elements should be in [-1,1], giving segments coordinates ranges.
align	Character, alignment type of labels, "l" means left, "c" means center and "r" right.
addlabels	Logical, whether add text label or not.
...	Additional arguments, passed to plot.

Author(s)

Taiyun Wei

Examples

```
par(mar=rep(0,4))
plot(0,xlim=c(0,6),ylim=c(-0.5,1.2),type="n")
colorlegend(rainbow(100), 0:9)
colorlegend(heat.colors(100),LETTERS[1:12], xlim=c(1,2))
colorlegend(terrain.colors(100),0:9, ratio.colbar=0.6,
lim.segment=c(0,0.6),xlim=c(2,3), ,align="l")
colorlegend(topo.colors(100), 0:9, lim.segment=c(0,0.6), xlim=c(3,4),align="l", offset=0)
colorlegend(cm.colors(100),1:5, xlim=c(4,5))
colorlegend(sample(rainbow(12)), labels=LETTERS[1:12], at=seq(0.05, 0.95, len=12),
xlim=c(5,6),align="r")
colorlegend(colbar=grey(1:100/100),1:10, col="red",
xlim=c(0,6), ylim=c(-0.5,-0.1), vertical=FALSE, align="l")
colorlegend(sample(rainbow(12)), labels=LETTERS[1:12], at=seq(0.05, 0.95, len=12),
xlim=c(0,6), ylim=c(1.1,1.2), vertical=FALSE)
```

corrMatOrder *Reorder a correlation matrix.*

Description

Draw rectangle(s) around the chart of corrrlation matrix based on the number of each cluster's members.

Usage

```
corrMatOrder(corr, order=c("AOE", "FPC", "hclust", "alphabet"),
hclust.method = c("complete", "ward", "single", "average",
"mcquitty", "median", "centroid"))
```

Arguments

corr	Correlation matrix to reorder.
order	Character, the ordering method for the correlation matrix.

- "original" for original order.
- "AOE" for the angular order of the eigenvectors. It is calculated from the order of the angles, a_i:

$$a_i = tan(e_{i2}/e_{i1}), if e_{i1} > 0$$

$$a_i = tan(e_{i2}/e_{i1}) + \pi, otherwise.$$

  where e_1 and e_2 are the largest two eigenvalues of matrix corr. See Michael Friendly (2002) for details.
- "FPC" for the first principal component order.
- "hclust" for hierarchical clustering order.
- "alphabet" for alphabetical order.

hclust.method	Character, the agglomeration method to be used when order is hclust. This should be one of "ward", "single", "complete", "average", "mcquitty", "median" or "centroid".

Value

Returns a single permutation vector.

Author(s)

Taiyun Wei

See Also

Package seriation offers more methods to reorder matrices, such as ARSA, BBURCG, BBWRCG, MDS, TSP, Chen and so forth.

Examples

```
M <- cor(mtcars)
(order.AOE <- corrMatOrder(M, order="AOE"))
(order.FPC <- corrMatOrder(M, order="FPC"))
(order.hc <- corrMatOrder(M, order="hclust"))
(order.hc2 <- corrMatOrder(M, order="hclust", hclust.method="ward"))
M.AOE <- M[order.AOE,order.AOE ]
M.FPC <- M[order.FPC,order.FPC ]
M.hc  <- M[order.hc, order.hc ]
M.hc2 <- M[order.hc2,order.hc2]



par(ask=TRUE)
corrplot(M)
corrplot(M.AOE)
corrplot(M.FPC)
corrplot(M.hc)

corrplot(M.hc)
corrRect.hclust(corr=M.hc, k=2)

corrplot(M.hc)
corrRect.hclust(corr=M.hc, k=3)

corrplot(M.hc2)
corrRect.hclust(M.hc2, k=2, method="ward")
```

`corrplot` *A visualization of a correlation matrix.*

Description

A graphical display of a correlation matrix, confidence interval. The details are paid great attention to.

It can also visualize a general matrix if using parameter `zoom` to scale matrix.

Usage

```
corrplot(corr,
method = c("circle", "square", "ellipse", "number", "shade", "color", "pie"),
type = c("full", "lower", "upper"), add = FALSE,
col = NULL, bg = "white", title = "",  zoom = 1,
diag = TRUE, outline = FALSE, mar = c(0,0,0,0),
addgrid.col = "gray", addCoef.col = NULL, addCoefasPercent = TRUE,
```

```
order = c("original", "AOE", "FPC", "hclust", "alphabet"),
hclust.method = c("complete", "ward", "single", "average","mcquitty", "median", "centroid"),
addrect = NULL, rect.col = "black", rect.lwd = 2,

addtextlabel = NULL, tl.cex = 1,
tl.col = "red", tl.offset = 0.4, tl.srt = 90,

addcolorlabel = c("right", "bottom","no"), cl.lim = c(-1, 1),
cl.length = NULL, cl.cex = 0.8, cl.ratio = 0.15,
cl.align.text = "c",cl.offset = 0.5,

addshade = c("negative", "positive", "all"),
shade.lwd = 1, shade.col = "white",

p.mat = NULL, sig.level = 0.05,
insig = c("pch","blank", "no"),
pch = 4, pch.col = "black", pch.cex = 3,

plotCI = c("no","square", "circle", "rect"),
lowCI.mat = NULL, uppCI.mat = NULL, ...)
```

Arguments

corr	The correlation matrix to visualize, must be square if order is not "original". For general matrix, please use zoom to convert.
method	Character, the visualization method of correlation matrix to be used. Currently, it supports seven methods, named "circle" (default), "square", "ellipse", "number", "pie", "shade" and "color". See examples for details. The areas of circles or squares show the absolute value of corresponding correlation coefficients. Method "pie" and "shade" came from Michael Friendly's job (with some adjustment about the shade added on), and "ellipse" came from D.J. Murdoch and E.D. Chow's job, see in section References.
type	Character, "full" (default), "upper" or "lower", display full matrix, lower triangular or upper triangular matrix.
add	Logical, if TRUE, the graph is added to an existing plot, otherwise a new plot is created.
col	Vector, the color of glyphs. It is distributed uniformly in [-1, 1] (after zoom). If NULL, col will be colorRampPalette(col2)(200), see example about col2.
bg	The background color.
title	Character, title of the graph.
zoom	Numeric, zoom for matrix corr: corr <- corr * zoom. If the input matrix is not a correlation matrix in [-1, 1], then we can specify zoom to make the input matrix in [-1,1] and use corrplot() to visualize it. Warning: please make sure the visualization you take after data zoom is meaningful!
diag	Logical, whether display the correlation coefficients on the principal diagonal.

`outline` Logical, whether plot outline of circles, square and ellipse.

`mar` See `par`.

`addgrid.col` The color of grid, if `NULL`, don't add grid.

`addCoef.col` Color of coefficients added on the graph. If `NULL` (default), add no coefficients.

`addCoefasPercent`
Logic, whether translate coefficients into percentage style for spacesaving.

`order` Character, the ordering method of the correlation matrix.

- `"original"` for original order (default).
- `"AOE"` for the angular order of the eigenvectors.
- `"FPC"` for the first principal component order.
- `"hclust"` for the hierarchical clustering order.
- `"alphabet"` for alphabetical order.

See function `corrMatOrder` for details.

`hclust.method` Character, the agglomeration method to be used when `order` is `hclust`. This should be one of `"ward"`, `"single"`, `"complete"`, `"average"`, `"mcquitty"`, `"median"` or `"centroid"`.

`addrect` Integer, the number of rectangles draws on the graph according to the hierarchical cluster, only valid when `order` is `hclust`. If `NULL` (default), then add no rectangles.

`rect.col` Color for rectangle border(s), only valid when `addrect` is equal or greater than 1.

`rect.lwd` Numeric, line width for borders for rectangle border(s), only valid when `addrect` is equal or greater than 1.

`addtextlabel` Character, `"lt"`, `"ld"`, `"td"`, `"d"` or `"no"`, giving position of text labels; `"lt"` means left and top, `"ld"` means left and diagonal, `"td"` means top and diagonal(near), `"d"` means diagonal, `"no"` means don't add textlabel.

`tl.cex` Numeric, for the size of text label (variable names).

`tl.col` The color of text label.

`tl.offset` Numeric, for text label, see `text`.

`tl.srt` Numeric, for text label string rotation in degrees, see `text`.

`addcolorlabel` Character, `"right"` (default), `"bottom"` or `"no"`, where to draw colorlabel. If `"no"`, don't draw colorlabel.

`cl.lim` The limits `(x1, x2)` (after zoom) in the colorlabel.

`cl.length` Integer, the number of number-text in colorlabel, passed to `colorlegend`. If `NULL`, `cl.length` is `length(col) + 1` when `length(col) <=20`; `cl.length` is 11 when `length(col) > 20`

`cl.cex` Numeric, cex of number-label in colorlabel, passed to `colorlegend`.

`cl.ratio` Numeric, to justify the width of colorlabel, 0.1~0.2 is suggested.

`cl.align.text` Character, `"l"`, `"c"` (default) or `"r"`, for number-label in colorlabel, `"l"` means left, `"c"` means center, and `"r"` means right.

`cl.offset` Numeric, for number-label in colorlabel, see `text`.

addshade	Character for shade style, "negative", "positive" or "all", only valid when method is "shade". If "all", all correlation coefficients' glyph will be shaded; if "positive", only the positive will be shaded; if "negative", only the negative will be shaded. Note: the angle of shade line is different, 45 degrees for positive and 135 degrees for negative.
shade.lwd	Numeric, the line width of shade.
shade.col	The color of shade line.
p.mat	Matrix of p-value, if NULL, arguments sig.level, insig, pch, pch.col, pch.cex is invalid.
sig.level	Significant level, if the p-value in p-mat is bigger than sig.level, then the corresponding correlation coefficient is regarded as insignificant.
insig	Character, specialized insignificant correlation coefficients, "pch" (default), "blank" or "no". If "blank", wipe away the corresponding glyphs; if "pch", add characters (see pch for details) on corresponding glyphs; if "no", don't take any measures.
pch	Add character on the glyphs of insignificant correlation coefficients(only valid when insig is "pch"). See par.
pch.col	The color of pch (only valid when insig is "pch").
pch.cex	The cex of pch (only valid when insig is "pch").
plotCI	Character, method of ploting confidence interval. If "no", don't plot confidence interval. If "rect", plot rectangles whose upper side means upper bound and lower side means lower bound, respectively, and meanwhile correlation coefficients are also added on the rectangles. If "circle", first plot a circle with the bigger absolute bound, and then plot the smaller. Warning: if the two bounds are the same sign, the smaller circle will be wiped away, thus forming a ring. Method "square" is similar to "circle".
lowCI.mat	Matrix of the lower bound of confidence interval.
uppCI.mat	Matrix of the upper bound of confidence interval.
...	Additional arguments passing to function text for drawing text lable.

Details

corrplot function offers flexible ways to visualize correlation matrix, lower and upper bound of confidence interval matrix.

Value

(Invisibly) returns a single permutation vector.

Note

Cairo and cairoDevice packages is strongly recommended to produce high-quality PNG, JPEG, TIFF bitmap files, especially for that method circle, ellipse .

Author(s)

Taiyun Wei

References

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

D.J. Murdoch, E.D. Chow (1996). *A graphical display of large correlation matrices*. The American Statistician, 50, 178–180.

See Also

Function `plotcorr` in the `ellipse` package and `corrgram` in the `corrgram` package have some similarities.

Package `seriation` offered more methods to reorder matrices, such as ARSA, BBURCG, BBWRCG, MDS, TSP, Chen and so forth.

Examples

```
data(mtcars)
M <- cor(mtcars)
##  different color series
col1 <- colorRampPalette(c("#7F0000","red","#FF7F00","yellow","white",
"cyan", "#007FFF", "blue","#00007F"))
col2 <- colorRampPalette(c("#67001F", "#B2182B", "#D6604D", "#F4A582", "#FDDBC7",
"#FFFFFF", "#D1E5F0", "#92C5DE", "#4393C3", "#2166AC", "#053061"))
col3 <- colorRampPalette(c("red", "white", "blue"))
col4 <- colorRampPalette(c("#7F0000","red","#FF7F00","yellow","#7FFF7F",
"cyan", "#007FFF", "blue","#00007F"))
wb <- c("white","black")


par(ask = TRUE)


## different color scale and methods to display corr-matrix
corrplot(M, method="number", col="black", addcolorlabel="no")
corrplot(M, method="number")
corrplot(M)
corrplot(M, order ="AOE")
corrplot(M, order ="AOE", addCoef.col="grey")

corrplot(M, order="AOE", col=col1(20), cl.length=21,addCoef.col="grey")
corrplot(M, order="AOE", col=col1(10),addCoef.col="grey")

corrplot(M, order="AOE", col=col2(200))
corrplot(M, order="AOE", col=col2(200),addCoef.col="grey")
corrplot(M, order="AOE", col=col2(20), cl.length=21,addCoef.col="grey")
corrplot(M, order="AOE", col=col2(10),addCoef.col="grey")
```

```
corrplot(M, order="AOE", col=col3(100))
corrplot(M, order="AOE", col=col3(10))


corrplot(M, method="color", col=col1(20), cl.length=21,order = "AOE", addCoef.col="grey")
corrplot(M, method="square", col=col2(200),order = "AOE")
corrplot(M, method="ellipse", col=col1(200),order = "AOE")
corrplot(M, method="shade", col=col3(20),order = "AOE")
corrplot(M, method="pie", order = "AOE")

## col=wb
corrplot(M, col = wb, order="AOE", outline=TRUE, addcolorlabel="no")
## like Chinese wiqi, suit for either on screen or white-black print.
corrplot(M, col = wb, bg="gold2",  order="AOE", addcolorlabel="no")


## mixed methods: It's more efficient if using function "corrplot.mixed"
## circle + ellipse
corrplot(M,order="AOE",type="upper",addtextlabel="d")
corrplot(M,add=TRUE, type="lower", method="ell",order="AOE",
diag=FALSE,addtextlabel="no", addcolorlabel="no")

## circle + square
corrplot(M,order="AOE",type="upper",addtextlabel="d")
corrplot(M,add=TRUE, type="lower", method="square",order="AOE",
diag=FALSE,addtextlabel="no", addcolorlabel="no")

## circle + colorful number
corrplot(M,order="AOE",type="upper",addtextlabel="d")
corrplot(M,add=TRUE, type="lower", method="number",order="AOE",
diag=FALSE,addtextlabel="no", addcolorlabel="no")

## circle + black number
corrplot(M,order="AOE",type="upper",addtextlabel="tp")
corrplot(M,add=TRUE, type="lower", method="number",order="AOE", col="black",
diag=FALSE,addtextlabel="no", addcolorlabel="no")


## order is hclust and draw rectangles
corrplot(M, order="hclust")
corrplot(M, order="hclust", addrect = 2)
corrplot(M, order="hclust", addrect = 3, rect.col = "red")
corrplot(M, order="hclust", addrect = 4, rect.col = "blue")
corrplot(M, order="hclust", hclust.method="ward", addrect = 4)



## visualize a  matrix in [0, 1]
corrplot(abs(M),order="AOE", cl.lim=c(0,1))
corrplot(abs(M),order="AOE", col=col1(20), cl.lim=c(0,1))
corrplot(abs(M),order="AOE", col=col3(200), cl.lim=c(0,1))
```

```
## visualize a  matrix in [0, 6]
corrplot(6*abs(M),order="AOE", zoom=1/6, col=col3(200), cl.lim=c(0,1))

## visualize a  matrix in [-6, 6]
corrplot(6*M, order="AOE", zoom=1/6,col=col3(200))
corrplot(round(6*M,1), order="AOE", zoom=1/6,col=col3(200),
addCoef.col="grey", addCoefasPercent=FALSE)

## text-labels and plot type
corrplot(M, order="AOE", tl.srt=45)
corrplot(M, order="AOE", tl.srt=60)
corrplot(M, order="AOE", addtextlabel="d",addcolorlabel="no")
corrplot(M, order="AOE", diag=FALSE, addtextlabel="d")
corrplot(M, order="AOE", type="upper")
corrplot(M, order="AOE", type="upper", diag=FALSE)
corrplot(M, order="AOE", type="lower", addcolorlabel="b")
corrplot(M, order="AOE", type="lower", addcolorlabel="b", diag=FALSE)



#### color-legend
corrplot(M, order="AOE", cl.ratio=0.2, cl.align="l")
corrplot(M, order="AOE", cl.ratio=0.2, cl.align="c")
corrplot(M, order="AOE", cl.ratio=0.2, cl.align="r")
corrplot(M, order="AOE", addcolorlabel="bottom")
corrplot(M, order="AOE", addcolorlabel="bottom", addtextlabel="d")
corrplot(M, order="AOE", addcolorlabel="no")



##the input matrix is not square
corrplot(M[1:8,])
corrplot(M[,1:8])



cor.mtest <- function(mat, conf.level = 0.95){
mat <- as.matrix(mat)
n <- ncol(mat)
p.mat <- lowCI.mat <- uppCI.mat <- matrix(NA, n, n)
diag(p.mat) <- 0
diag(lowCI.mat) <- diag(uppCI.mat) <- 1
for(i in 1:(n-1)){
for(j in (i+1):n){
tmp <- cor.test(mat[,i], mat[,j], conf.level = conf.level)
p.mat[i,j] <- p.mat[j,i] <- tmp$p.value
lowCI.mat[i,j] <- lowCI.mat[j,i] <- tmp$conf.int[1]
uppCI.mat[i,j] <- uppCI.mat[j,i] <- tmp$conf.int[2]
}
}
return(list(p.mat, lowCI.mat, uppCI.mat))
}
```

```
res1 <- cor.mtest(mtcars,0.95)
res2 <- cor.mtest(mtcars,0.99)


## specialized the insignificant value according to the significant level
corrplot(M, p.mat = res1[[1]], sig.level=0.2)
corrplot(M, p.mat = res1[[1]], sig.level=0.05)
corrplot(M, p.mat = res1[[1]], sig.level=0.01)
corrplot(M, p.mat = res1[[1]], insig = "blank")
corrplot(M, p.mat = res1[[1]], order="hclust", insig = "blank", addrect=3)
corrplot(M, p.mat = res1[[1]], order="hclust", insig = "pch", addrect=3)



## plot confidence interval(0.95), "square" method
corrplot(M,low=res1[[2]], upp=res1[[3]],
plotC="circle", addg="grey20",addc="no")
corrplot(M, p.mat = res1[[1]],low=res1[[2]], upp=res1[[3]],
plotC="circle", addg="grey20",addc="no")
corrplot(M, low=res1[[2]], upp=res1[[3]],
col=c("white","black"),bg="gold2",order="AOE",
plotCI="circle",addc="no",pch.col="red")
corrplot(M, p.mat = res1[[1]], low=res1[[2]], upp=res1[[3]],
col=c("white","black"),bg="gold2",order="AOE",
plotCI="circle",addc="no",pch.col="red")

## plot confidence interval(0.95), "square" method
corrplot(M, low=res1[[2]], upp=res1[[3]],
col=c("white","black"),bg="gold2", order="AOE",
plotCI="square",addg=NULL,addc="no")
corrplot(M, p.mat = res1[[1]],low=res1[[2]], upp=res1[[3]],
col=c("white","black"),bg="gold2",order="AOE",pch.col="red",
plotC="square", addg=NULL,addc="no")

## plot confidence interval(0.95, 0.95, 0.99), "rect" method
corrplot(M, low=res1[[2]], upp=res1[[3]], order="hclust",
rect.col="navy", plotC="rect",addc="no")
corrplot(M, p.mat = res1[[1]], low=res1[[2]], upp=res1[[3]], order="hclust",
pch.col="red", sig.level = 0.05, addrect=3, rect.col="navy",
plotC="rect",addc="no")
corrplot(M, p.mat = res2[[1]], low=res2[[2]], upp=res2[[3]], order="hclust",
pch.col="red", sig.level = 0.01, addrect=3, rect.col="navy",
plotC="rect",addc="no")


## an animation of changing confidence interval in different significance level
par(ask=FALSE)
for(i in seq(0.1, 0.001, -0.001)){
tmp <- cor.mtest(mtcars,1-i)
corrplot(M, p.mat = tmp[[1]], low=tmp[[2]], upp=tmp[[3]], order="hclust",
pch.col="red", sig.level = i, plotC="rect", addc="no",
mar=c(0,0,1,0),
title=substitute(alpha == x,list(x=format(i,digits=3,nsmall=3))))
```

```
Sys.sleep(0.01)
}
```

---

corrplot.mixed *Using mixed methods to visualize a correlation matrix.*

Description

Using mixed methods to visualize a correlation matrix.

Usage

```
corrplot.mixed(corr, lower = "number", upper = "circle",
addtextlabel = c("d","lt","no"), diag = c("n","l","u"),
bg = "white", addgrid.col = "gray",...)
```

Arguments

corr	Matrix, the correlation matrix to visualize.
lower	Character, the visualization method for the lower triangular correlation matrix.
upper	Character, the visualization method for the upper triangular correlation matrix.
addtextlabel	Character, "lt", "d" or "no", giving position of text labels, "lt" means left and top, "d" means diagonal. If "no", add no textlabel.
diag	Character, for specifying the glyph on the principal diagonal. It is one of "n" (default, draw nothing), "l" (draw the glyphs of lower triangular) or "u" (draw the glyphs of upper triangular).
bg	The background color.
addgrid.col	The color of grid, if NULL, don't add grid.
...	Additional arguments for corrplot's wrappers

Author(s)

Taiyun Wei

Examples

```
M <- cor(mtcars)
ord <- corrMatOrder(M, order="AOE")
M2 <- M[ord,ord]
par(ask=TRUE)
corrplot.mixed(M2)
corrplot.mixed(M2, lower="ellipse", upper="circle")
corrplot.mixed(M2, lower="square", upper="circle")
corrplot.mixed(M2, lower="shade", upper="circle")
corrplot.mixed(M2, addtextlabel="lt")
```

```
corrplot.mixed(M2, addtextlabel="lt", diag="u")
corrplot.mixed(M2, addtextlabel="lt", diag="l")
corrplot.mixed(M2, addtextlabel="no")
```

---

corrRect *Draw rectangle(s) on the correlation matrix graph.*

Description

Draw rectangle(s) around the chart of corrrlation matrix.

corrRect needs the number(parameter clus) of each cluster's members, while corrRect.hclust can get the members in each cluster based on hierarchical clustering (hclust).

Usage

```
corrRect(clus,  col = "black", lwd = 2)

corrRect.hclust(corr, k=2, col = "black", lwd = 2,
method = c("complete", "ward", "single", "average","mcquitty", "median", "centroid"))
```

Arguments

clus	Vector, the number of each cluster's members.
corr	Correlation matrix for function corrRect.hclust. It use 1-corr as dist in hierarchical clustering (hclust).
k	Integer, the number of rectangles drawn on the graph according to the hierarchical cluster, for function corrRect.hclust.
col	Color of rectangles.
lwd	Line width of rectangles.
method	Character, the agglomeration method to be used for hierarchical clustering (hclust). This should be (an unambiguous abbreviation of) one of "ward", "single", "complete", "average", "mcquitty", "median" or "centroid".

Author(s)

Taiyun Wei

Examples

```
data(mtcars)
M <- cor(mtcars)
corrplot(M, method="circle", order = "FPC")
corrRect(c(5,6))

(order.hc <- corrMatOrder(M, order="hclust"))
```

```
(order.hc2 <- corrMatOrder(M, order="hclust", hclust.method="ward"))
M.hc  <- M[order.hc, order.hc ]
M.hc2 <- M[order.hc2,order.hc2]

par(ask=TRUE)
##same as: corrplot(M, order="hclust", addrect=2)
corrplot(M.hc)
corrRect.hclust(corr=M.hc, k=2)

##same as: corrplot(M, order="hclust", addrect=3)
corrplot(M.hc)
corrRect.hclust(corr=M.hc, k=3)

##same as: corrplot(M, order="hclust", hclust.method="ward", addrect=2)
corrplot(M.hc2)
corrRect.hclust(M.hc2, k=2, method="ward")

##same as: corrplot(M, order="hclust", hclust.method="ward", addrect=3)
corrplot(M.hc2)
corrRect.hclust(M.hc2, k=3, method="ward")

##same as: corrplot(M, order="hclust", hclust.method="ward", addrect=4)
corrplot(M.hc2)
corrRect.hclust(M.hc2, k=4, method="ward")
```

Index