R documentation

of ‘colorbar.Rd’

February 13, 2013

colorbar *Add Colorbars to Plots*

Description

This function adds a colorbar to a plot.

Usage

```
colorbar(x, y = NULL, col = palette(), labels = TRUE,
         tick = !any(is.na(c(minlabel, maxlabel))) ||
         !is.logical(labels),
         horiz = "auto", minlabel = NA, maxlabel = NA,
         nticks = 2, tickat = NULL, margin = rep(-0.03, 4),
         longside = -0.6, shortside = -0.05,
         axisloc = c("in", "out"), reverse = "auto",
         xjust = 0.5, yjust = 0.5, labeljust = TRUE,
         noaxissize = FALSE, totalsize = FALSE, ...)
```

Arguments

x — The x coordinate(s) or other location specifier used to position the colorbar. See ‘Details’.

y — The y coordinate(s) used to position the colorbar.

col — The colors used in the colorbar

labels — A vector of labels to draw. See *tickat* for help on setting the location of the labels (each marked with a tick). Can also be `FALSE` for no labels or `TRUE` for automatic labels and locations when *minlabel* and *maxlabel* are specified. Mathematical annotation is supported, see `plotmath`.

tick — A `logical` flag. If `TRUE`, draw ticks (and labels). If `FALSE`, just the colorbar is drawn with no extras.

horiz — A `logical` flag or `"auto"` indicating if the colorbar will be drawn horizontally (`TRUE`) or vertically (`FALSE`). If `"auto"`, the orientation will be determined automatically: When *x* is `"top"` or `"bottom"` (`"left"` or `"right"`), horizontal (vertical) orientation is assumed. When two (x, y) points are given, the colorbar runs along the axis with the longer distance (in inches) between the points. Otherwise, the ratio of the width and height (in inches) of the plotting region determines the orientation: the colorbar runs along the longer axis (vertically if the axes have equal length).

`minlabel` A `numeric` value giving the label corresponding to the first color. If `NA`, manual *labels* are used if available.

`maxlabel` A `numeric` value giving the label corresponding to the last color.

`nticks` The desired number of ticks and labels. Used when a linear number range is indicated by giving values for *minlabel* and *maxlabel* or when only ticks and no labels are used. In the first case, the number will not necessarily match the actual number of ticks and labels that are drawn.

`tickat` A vector giving the tick locations on the annotation axis. Values between `i - 0.5` and `i + 0.5` cover the i:th color. When *minlabel* and *maxlabel* are specified, automatic tick locations are used instead, guided by *nticks*.

`margin` A `numeric` vector of length 4 giving the margins on the bottom, left, top, right sides (in this order). Given either as a proportion of the relevant dimension of the plotting region (negative value, absolute value taken) or as inches (positive value). If the colorbar is placed in a corner, two of the margin values are used. When *x* is `"standalone"`, all four values are used. Otherwise, only one margin is relevant.

`longside` The size of the "long" side of the colorbar. Given either as a proportion of the relevant dimension of the plotting region (negative value, absolute value taken) or as inches (positive value). Long side means the side along which the colors change. Ignored if two (x, y) coordinates are supplied.

`shortside` The size of the "short" side of the colorbar. See *longside*. Ignored if two (x, y) coordinates are supplied. See argument *totalsize*.

`axisloc` Location of annotation axis relative to colorbar. Either `"in"` (closer to the center of the plotting region) or `"out"` (closer to the border). When a horizontal / vertical colorbar is drawn in the middle / center, `"out"` means the bottom / right side of the colorbar.

`reverse` A `logical` flag. If `TRUE`, the drawing order of the colors is reversed (right to left or up to down).

`xjust` A `numeric` value indicating the horizontal justification of the colorbar when its position is given with a single (x, y) coordinate. A value of 0 means left justified, 0.5 is centered and 1 right justified.

`yjust` A `numeric` value indicating the vertical justification. 0 means bottom justified, 0.5 is centered and 1 top justified.

`labeljust` A `logical` flag. If `TRUE`, some adjustments are made so that the labels are justified next to the colorbar.

`noaxissize` A `logical` flag. If `TRUE`, the size of the axis is not taken into account when positioning the colorbar. When two (x, y) coordinates are given, `TRUE` means that the colorbar itself fills the whole space, and `FALSE` means that the annotation axis and its labels (if drawn) consume some of the space.

`totalsize` A `logical` flag. If `TRUE`, *shortside* must be large enough to accommodate the possible annotation axis and labels. If `FALSE` (the default), the axis consumes space additional to *shortside*.

`...` Arguments passed to `axis`

Details

The values of *x* and *y* are interpreted using `xy.coords`. This allows giving one coordinate point, two points (the corners of the colorbar) and a number of other possibilities. For example, *x* can be a call to `locator`. Alternatively, *x* can be one of the preset locations `"bottomright"`, `"bottom"`, `"bottomleft"`, `"left"`, `"topleft"`, `"top"`, `"topright"`, `"right"`, `"center"`. When *x* is given the special value `"standalone"`, a standalone colorbar occupies the whole device except *margin*.

Additional arguments can be given to `axis`. For example, *las* controls the orientation of axis labels and *cex.axis* adjusts text size.

Value

An `invisible` named `numeric` vector containing the approximate coordinates defining the bounding box of the colorbar and its axis, with components (in this order)

`xleft`	x coordinate, left side
`ybottom`	y coordinate, bottom side
`xright`	x coordinate, right side
`ytop`	y coordinate, top side

Author(s)

Mikko Korpela

See Also

`par` for how to specify colors

Examples

```
### Example 1
dev.new()
pal <- cm.colors(128)
agricol <- round(swiss$Agriculture / 100 * 127 + 1)
plot(swiss$Education, swiss$Fertility, bg = pal[agricol], pch=21, cex=2,
     xlab = "Education", ylab = "Fertility")
coord <- colorbar("right", col = pal, minlabel = 0, maxlabel = 100,
                  nticks=6, las=1, axisloc="out", cex.axis=0.75)
text(coord[1] - 2, mean(coord[c(2, 4)]), "Agriculture",
     srt = 90, adj=c(0.5,0))
if (!dev.interactive()) {
    dev.off()
}

### Example 2 (4 rows, 2 columns)
dev.new(width = 7, height = 10, units = "in")
par(mfrow = c(4, 2))
pal8 <- rainbow(8)
pal64 <- terrain.colors(64)

## [1, 1]
## Sine. Draw colorbars under peaks, over valleys.
x <- seq(0, 6*pi, length.out=101)
plot(x, sin(x), type="l", xaxt="n", main="Sine")
axis(1, at = pi/2 * c(1, 5, 9),
     labels = c(expression(pi/2), expression(5*pi/2), expression(9*pi/2)))
colorbar(3*pi/2, 1, xjust = 0.5, yjust = 1, col = sample(colors(), 8),
         horiz = FALSE, minlabel=1, maxlabel=8, axisloc="out",
         mgp = c(3, 0.5, 0), shortside = -0.025)
colorbar(pi/2, -1, xjust = 0.5, yjust = 0, col = sample(colors(), 8),
         horiz = FALSE, minlabel=1, maxlabel=8, nticks=4, axisloc="in",
         mgp = c(3, 0.5, 0), shortside = -0.025)
colorbar(9*pi/2, -1, xjust = 0.5, yjust = 0, col = sample(colors(), 8),
         horiz = FALSE, minlabel=0, maxlabel=1000, nticks=4, axisloc="out",
         mgp = c(3, 0.5, 0), shortside = -0.025)
colorbar(11*pi/2, 1, xjust = 0.5, yjust = 1, col = sample(colors(), 8),
         horiz = FALSE, minlabel=1, maxlabel=8, axisloc="in",
         mgp = c(3, 0.5, 0), shortside = -0.025)

## [1, 2]
x <- 1:10
plot(x, 10^x, log="y", main="Log y axis")
colorbar("topleft", horiz=TRUE, col=pal64)
colorbar("bottomright", horiz=TRUE, col=pal64)
colorbar("topright", horiz=FALSE, col=pal64)
colorbar("bottomleft", horiz=FALSE, col=pal64)

## [2, 1]
plot(x, log10(x), log="x", main="Log x axis")
pin <- par("pin")
shortside <- 0.05 * max(pin)
longside <- 0.6 * min(pin)
oldpar <- list(mgp = par("mgp"), las = par("las"))
par(mgp = c(3, 0.5, 0), las = 1)
colorbar("topleft", horiz=FALSE, col=pal64, reverse=TRUE,
         shortside = shortside, longside = longside,
         minlabel = 0, maxlabel = 1, nticks = 5, axisloc = "out")
colorbar("bottomright", horiz=FALSE, col=pal64, reverse=TRUE,
         shortside = shortside, longside = longside,
         minlabel = 0, maxlabel = 1, nticks = 5, axisloc = "out")
colorbar("top", horiz=TRUE, col=pal64, reverse=TRUE,
         shortside = shortside, longside = longside,
         minlabel = 0, maxlabel = 1, nticks = 10, axisloc = "out")
colorbar("bottom", horiz=TRUE, col=pal64, reverse=TRUE,
         shortside = shortside, longside = longside,
         minlabel = 0, maxlabel = 1, nticks = 10, axisloc = "out")
par(oldpar)

## [2, 2]
## 3 cycloid cycles; vertical, horizontal, vertical colorbar.
## Coordinates are given as a pair of (x, y) points.
## The default ’reverse="auto"’ means that the order of points in ’x’ or
## ’y’ determines the drawing order of the colors. The default
## ’horiz="auto"’ compares the width and height of the colorbar and
## automatically selects horizontal or vertical orientation.
t <- seq(0, 6*pi, length.out=101)
x <- t - sin(t)
y <- 1 - cos(t)
plot(x, y, type="l", main="Cycloid with radius 1", asp = 1, xaxt = "n")
axis(1, at = pi * c(0, 2, 4, 6),
     labels = c("0", expression(2*pi), expression(4*pi), expression(6*pi)))
barshort <- 0.8
```

```
barlong <- 5
colorbar(x = c(pi-barshort/2, pi+barshort/2), y = c(1.8-barlong, 1.8),
         col = pal8)
colorbar(x = c(3*pi-barlong/2, 3*pi+barlong/2),
         y = c(2.2, 2.2+barshort), col = pal8)
## Reverse order of 'y' points
colorbar(x = c(5*pi-barshort/2, 5*pi+barshort/2), y = c(1.8, 1.8-barlong),
         col = pal8)


## [3, 1]
## Stair steps. Two horizontal colorbars:
## * Each lean on one of the steps
## * Top colorbar is bottom aligned, extends to left border of plot
## * Bottom bar is top aligned, extends to right border
## * Height of each colorbar is height of the corresponding step
x <- 1:10
y <- sort(x + rnorm(10, sd=0.3))
plot(x, y, type="s", main="Stair steps")
diffy <- diff(y)
leftx <- which.max(diffy[5:9]) + 5
rightx <- which.max(diffy[1:4]) + 1
usr <- par("usr")
leftwidth <- -(leftx - usr[1]) / (usr[2] - usr[1])
rightwidth <- -(usr[2] - rightx) / (usr[2] - usr[1])
leftheight <- -diffy[leftx - 1] / (usr[4] - usr[3])
rightheight <- -diffy[rightx - 1] / (usr[4] - usr[3])
lefty <- y[leftx - 1]
righty <- y[rightx - 1]
colorbar(leftx, lefty, longside = leftwidth, shortside = leftheight,
         xjust = 1, yjust = 0, horiz = TRUE, col = pal8)
colorbar(rightx, righty, longside = rightwidth, shortside = rightheight,
         xjust = 0, yjust = 0, horiz = TRUE, col = pal8)

## [3, 2]
## "standalone" horizontal colorbar
colorbar("standalone", horiz = TRUE, axisloc = "out", col = pal8,
         labels = c("Hello", "Hello\nthere"), tickat=c(3, 6))

## [4, 1]
## cex.axis as an argument
plot(1:5)
colorbar("right", horiz=FALSE, col=pal64, minlabel = 0, maxlabel = 1,
         nticks = 3, axisloc = "out", las = 1, cex.axis = 2)

## [4, 2]
## cex.axis in par(), affecting all axes
oldpar <- list(cex.axis = par("cex.axis"))
par(cex.axis = 2)
plot(1:5)
colorbar("right", horiz=FALSE, col=pal64, minlabel = 0, maxlabel = 1,
         nticks = 3, axisloc = "out", las = 1)
par(oldpar)

if (!dev.interactive()) {
    dev.off()
}
```

Index