barplot {graphics}
R Documentation

Bar Plots

Description

Creates a bar plot with vertical or horizontal bars.

Usage

```
## Default S3 method:
barplot(height, width = 1, space = NULL,
        names.arg = NULL, legend.text = NULL, beside =
FALSE,
        horiz = FALSE, density = NULL, angle = 45,
        col = NULL, border = par("fg"),
        main = NULL, sub = NULL, xlab = NULL, ylab =
NULL,
        xlim = NULL, ylim = NULL, xpd = TRUE,
        axes = TRUE, axisnames = TRUE,
        cex.axis = par("cex.axis"), cex.names =
par("cex.axis"),
        inside = TRUE, plot = TRUE, axis.lty = 0, offset
= 0, ...)
```

Arguments

`height`

either a vector or matrix of values describing the bars which make up the plot. If `height` is a vector, the plot consists of a sequence of rectangular bars with heights given by the values in the vector. If `height` is a matrix and `beside` is `FALSE` then each bar of the plot corresponds to a column of `height`, with the values in the column giving the heights of stacked “sub-bars” making up the bar. If `height` is a matrix and `beside` is `TRUE`, then the values in each column are juxtaposed rather than stacked.

`width`

optional vector of bar widths. Re-cycled to length the number of bars drawn. Specifying a single value will no visible effect unless `xlim` is specified.

`space`

the amount of space (as a fraction of the average bar width) left before each bar. May be given as a single number or one number per bar. If `height` is a matrix and `beside` is `TRUE`, `space` may be specified by two numbers, where the first is the space between bars in the same group, and the second the space between the groups. If not given explicitly, it defaults to `c(0,1)` if `height` is a matrix and `beside` is `TRUE`, and to 0.2 otherwise.

`names.arg`

a vector of names to be plotted below each bar or group of bars. If this argument is omitted, then the names are taken from the `names` attribute of `height` if this is a vector, or the column names if it is a matrix.

`legend.text`

a vector of text used to construct a legend for the plot, or a logical indicating whether a legend should be included. This is only useful when `height` is a matrix. In that case given legend labels should correspond to the rows of `height`; if `legend.text` is true, the row names of `height` will be used as labels if they are non-null.

`beside`

a logical value. If `FALSE`, the columns of `height` are portrayed as stacked bars, and if `TRUE` the columns are portrayed as juxtaposed bars.

`horiz`

a logical value. If `FALSE`, the bars are drawn vertically with the first bar to the left. If `TRUE`, the bars are drawn horizontally with the first at the bottom.

`density`

a vector giving the density of shading lines, in lines per inch, for the bars or bar components. The default value of `NULL` means that no shading lines are drawn. Non-positive values of `density` also inhibit the drawing of shading lines.

`angle`

the slope of shading lines, given as an angle in degrees (counterclockwise), for the bars or bar components.

`col`

a vector of colors for the bars or bar components. By default, grey is used if `height` is a vector, and a gamma-corrected grey palette if `height` is a matrix.

`border`

the color to be used for the border of the bars.

`main,sub`

overall and sub title for the plot.

`xlab`

a label for the x axis.

`ylab`

a label for the y axis.

`xlim`

limits for the x axis.

`ylim`

limits for the y axis.

`xpd`

logical. Should bars be allowed to go outside region?

`axes`

logical. If `TRUE`, a vertical (or horizontal, if `horiz` is true) axis is drawn.

`axisnames`

logical. If `TRUE`, and if there are `names.arg` (see above), the other axis is drawn (with `lty=0`) and labeled.

`cex.axis`

expansion factor for numeric axis labels.

`cex.names`

expansion factor for axis names (bar labels).

`inside`

logical. If `TRUE`, the lines which divide adjacent (non-stacked!) bars will be drawn. Only applies when `space = 0` (which it partly is when `beside = TRUE`).

`plot`

logical. If `FALSE`, nothing is plotted.

`axis.lty`

the graphics parameter `lty` applied to the axis and tick marks of the categorical (default horizontal) axis. Note that by default the axis is suppressed.

`offset`

a vector indicating how much the bars should be shifted relative to the x axis.

`...`

further graphical parameters (`par`) are passed to `plot.window()`, `title()` and `axis`.

Details

This is a generic function, it currently only has a default method. A formula interface may be added eventually.

Value

A numeric vector (or matrix, when `beside = TRUE`), say `mp`, giving the coordinates of *all* the bar midpoints drawn, useful for adding to the graph.

If `beside` is true, use `colMeans(mp)` for the midpoints of each *group* of bars, see example.

Note

Prior to **R** 1.6.0, `barplot` behaved as if `axis.lty = 1`, unintentionally.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

See Also

plot(..., **type="h"**), **dotchart**, **hist**.

Examples

```
tN <- table(Ni <- rpois(100, lambda=5))
r <- barplot(tN, col='gray')
#- type = "h" plotting *is* 'bar'plot
lines(r, tN, type='h', col='red', lwd=2)

barplot(tN, space = 1.5, axisnames=FALSE,
        sub = "barplot(..., space= 1.5, axisnames =
FALSE)")

barplot(VADeaths, plot = FALSE)
barplot(VADeaths, plot = FALSE, beside = TRUE)

mp <- barplot(VADeaths) # default
tot <- colMeans(VADeaths)
text(mp, tot + 3, format(tot), xpd = TRUE, col = "blue")
barplot(VADeaths, beside = TRUE,
        col = c("lightblue", "mistyrose", "lightcyan",
                "lavender", "cornsilk"),
        legend = rownames(VADeaths), ylim = c(0, 100))
title(main = "Death Rates in Virginia", font.main = 4)

hh <- t(VADeaths)[, 5:1]
mybarcol <- "gray20"
mp <- barplot(hh, beside = TRUE,
        col = c("lightblue", "mistyrose",
                "lightcyan", "lavender"),
        legend = colnames(VADeaths), ylim= c(0,100),
        main = "Death Rates in Virginia", font.main = 4,
        sub = "Faked upper 2*sigma error bars", col.sub
= mybarcol,
        cex.names = 1.5)
segments(mp, hh, mp, hh + 2*sqrt(1000*hh/100), col =
mybarcol, lwd = 1.5)
stopifnot(dim(mp) == dim(hh))# corresponding matrices
mtext(side = 1, at = colMeans(mp), line = -2,
      text = paste("Mean", formatC(colMeans(hh))), col =
"red")
```

```
# Bar shading example
barplot(VADeaths, angle = 15+10*1:5, density = 20, col =
"black",
        legend = rownames(VADeaths))
title(main = list("Death Rates in Virginia", font = 4))

# border :
barplot(VADeaths, border = "dark blue")
```

[Package *graphics* version 2.0.1 Index]