boxplot {graphics}
R Documentation

Box Plots

Description

Produce box-and-whisker plot(s) of the given (grouped) values.

Usage

```
boxplot(x, ...)

## S3 method for class 'formula':
boxplot(formula, data = NULL, ..., subset, na.action =
NULL)

## Default S3 method:
boxplot(x, ..., range = 1.5, width = NULL, varwidth =
FALSE,
        notch = FALSE, outline = TRUE, names, plot =
TRUE,
        border = par("fg"), col = NULL, log = "",
        pars = list(boxwex = 0.8, staplewex = 0.5,
outwex = 0.5),
        horizontal = FALSE, add = FALSE, at = NULL)
```

Arguments

formula
a formula, such as y ~ grp, where y is a numeric vector of data values
to be split into groups according to the grouping variable grp (usually a
factor).

data
a data.frame (or list) from which the variables in formula should be
taken.

subset
an optional vector specifying a subset of observations to be used for

plotting.

`na.action`

a function which indicates what should happen when the data contain NAs. The default is to ignore missing values in either the response or the group.

`x`

for specifying data from which the boxplots are to be produced. Either a numeric vector, or a single list containing such vectors. Additional unnamed arguments specify further data as separate vectors (each corresponding to a component boxplot). NAs are allowed in the data.

`...`

For the `formula` method, arguments to the default method and graphical parameters.

For the default method, unnamed arguments are additional data vectors (unless `x` is a list when they are ignored), and named arguments are graphical parameters in addition to the ones given by argument `pars`.

`range`

this determines how far the plot whiskers extend out from the box. If `range` is positive, the whiskers extend to the most extreme data point which is no more than `range` times the interquartile range from the box. A value of zero causes the whiskers to extend to the data extremes.

`width`

a vector giving the relative widths of the boxes making up the plot.

`varwidth`

if `varwidth` is TRUE, the boxes are drawn with widths proportional to the square-roots of the number of observations in the groups.

`notch`

if `notch` is TRUE, a notch is drawn in each side of the boxes. If the notches of two plots do not overlap this is 'strong evidence' that the two medians differ (Chambers *et al.*, 1983, p. 62). See `boxplot.stats` for the calculations used.

`outline`

if `outline` is not true, the outliers are not drawn (as points whereas S+ uses lines).

`names`

group labels which will be printed under each boxplot.

`boxwex`

a scale factor to be applied to all boxes. When there are only a few groups, the appearance of the plot can be improved by making the boxes narrower.

`staplewex`

staple line width expansion, proportional to box width.

`outwex`

outlier line width expansion, proportional to box width.

`plot`

if TRUE (the default) then a boxplot is produced. If not, the summaries which the boxplots are based on are returned.

`border`

an optional vector of colors for the outlines of the boxplots. The values in `border` are recycled if the length of `border` is less than the number of plots.

`col`

if `col` is non-null it is assumed to contain colors to be used to colour the bodies of the box plots.

`log`

character indicating if x or y or both coordinates should be plotted in log scale.

`pars`

a list of (potentially many) more graphical parameters, e.g., `boxwex` or `outpch`; these are passed to `bxp` (if `plot` is true); for details, see there.

`horizontal`

logical indicating if the boxplots should be horizontal; default FALSE means vertical boxes.

`add`

logical, if true *add* boxplot to current plot.

`at`

numeric vector giving the locations where the boxplots should be drawn, particularly when add = TRUE; defaults to 1:n where n is the number of boxes.

Details

The generic function `boxplot` currently has a default method
(`boxplot.default`) and a formula interface (`boxplot.formula`).

Missing values are ignored when forming boxplots.

Value

List with the following components:
`stats`
a matrix, each column contains the extreme of the lower whisker, the
lower hinge, the median, the upper hinge and the extreme of the upper
whisker for one group/plot.
`n`
a vector with the number of observations in each group.
`conf`
a matrix where each column contains the lower and upper extremes of
the notch.
`out`
the values of any data points which lie beyond the extremes of the
whiskers.
`group`
a vector of the same length as `out` whose elements indicate which
group the outlier belongs to
`names`
a vector of names for the groups

References

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

Chambers, J. M., Cleveland, W. S., Kleiner, B. and Tukey, P. A.
(1983) *Graphical Methods for Data Analysis*. Wadsworth & Brooks/

Cole.

See also `boxplot.stats`.

`boxplot.stats` **which does the computation,** `bxp` **for the plotting and more examples; and** `stripchart` **for an alternative (with small data sets).**

Examples

```
## boxplot on a formula:
boxplot(count ~ spray, data = InsectSprays, col =
"lightgray")
# *add* notches (somewhat funny here):
boxplot(count ~ spray, data = InsectSprays,
        notch = TRUE, add = TRUE, col = "blue")

boxplot(decrease ~ treatment, data = OrchardSprays,
        log = "y", col="bisque")

rb <- boxplot(decrease ~ treatment, data =
OrchardSprays, col="bisque")
title("Comparing boxplot()s and non-robust mean +/- SD")

mn.t <- tapply(OrchardSprays$decrease,
OrchardSprays$treatment, mean)
sd.t <- tapply(OrchardSprays$decrease,
OrchardSprays$treatment, sd)
xi <- 0.3 + seq(rb$n)
points(xi, mn.t, col = "orange", pch = 18)
arrows(xi, mn.t - sd.t, xi, mn.t + sd.t,
        code = 3, col = "pink", angle = 75, length = .1)

## boxplot on a matrix:
mat <- cbind(Uni05 = (1:100)/21, Norm = rnorm(100),
             T5 = rt(100, df = 5), Gam2 = rgamma(100,
shape = 2))
```

```
boxplot(data.frame(mat), main =
"boxplot(data.frame(mat), main = ...)")
par(las=1)# all axis labels horizontal
boxplot(data.frame(mat), main = "boxplot(*, horizontal =
TRUE)",
        horizontal = TRUE)

## Using 'at = ' and adding boxplots -- example idea by
Roger Bivand :

boxplot(len ~ dose, data = ToothGrowth,
        boxwex = 0.25, at = 1:3 - 0.2,
        subset= supp == "VC", col="yellow",
        main="Guinea Pigs' Tooth Growth",
        xlab="Vitamin C dose mg",
        ylab="tooth length", ylim=c(0,35))
boxplot(len ~ dose, data = ToothGrowth, add = TRUE,
        boxwex = 0.25, at = 1:3 + 0.2,
        subset= supp == "OJ", col="orange")
legend(2, 9, c("Ascorbic acid", "Orange juice"),
        fill = c("yellow", "orange"))

## more examples in  help(bxp)
```

[Package *graphics* version 2.0.1 Index]