Initiation to R software Session IV

Pierre Michel

Master AMSE 1st year, 2019

Graphics

Introduction

In R, graphic are created using **graphical functions** mainly contained in packages `graphics` and `stats`.

The output of a graphical function is an object of type graphic. It is sent to a **graphics device** (default is a graphical windows, can be a file) in which graphics are created.

Two kinds of graphical functions:

- **primary** functions: create a new graph on the active graphics device, with axis, labels. . .
- **secondary** functions: add elements to an existing graph (points, lines, labels. . .).

Graphs are created in function of graphical parameters defined by default, and modifialble using `par()`.

Graphical windows

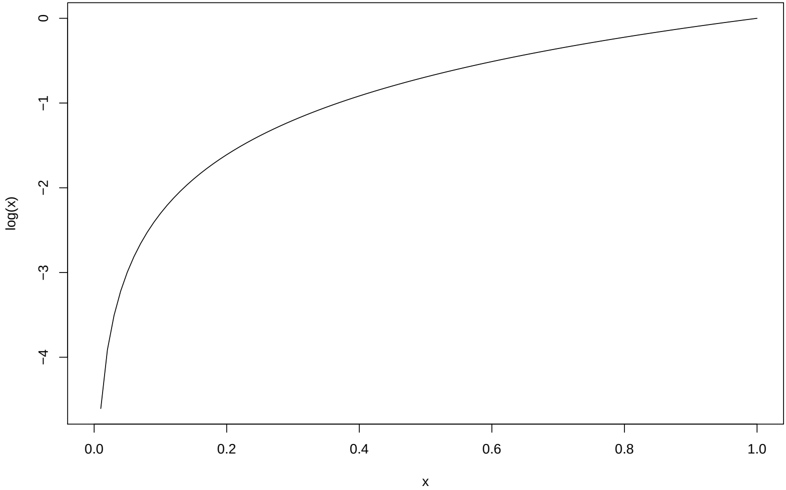
When the user runs a graphical command, if no window (or peripheral device) is opened, R opens a window in which the graphic will be plotted.

Many graphical windows can be opened, defined by their number, but only one (the last one) is active.

- `x11()` opens a graphical window.
- `pdf()`, `postscript()` opens a graphical peripheral device of type file.
- `dev.list()` lists the opened wnidows (or devices).
- `dev.cur()` active windows (or device).
- `dev.set(i)` activate window (or device) i.
- `dev.off(i)` close window (or device) i.
- `split.screen()` splits the active window, `screen()` selects a part, `erase.screen()` erases the last graph.

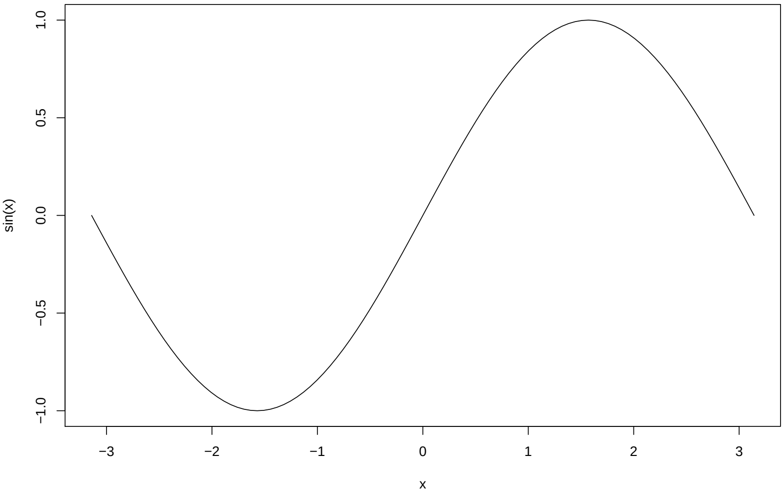
Graphical windows

```r
curve(log)
```



Graphical windows

```
x11(); curve(sin, -pi, pi)
dev.list(); dev.off(3)
```

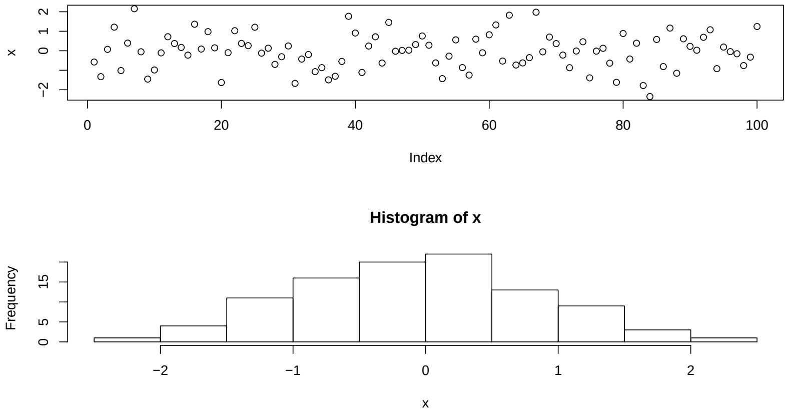


Graphical windows

```
split.screen(c(2,1)); screen(1); x=rnorm(100); plot(x)
```

```
## [1] 1 2
```

```
screen(2); hist(x)
```



Primary graphical functions

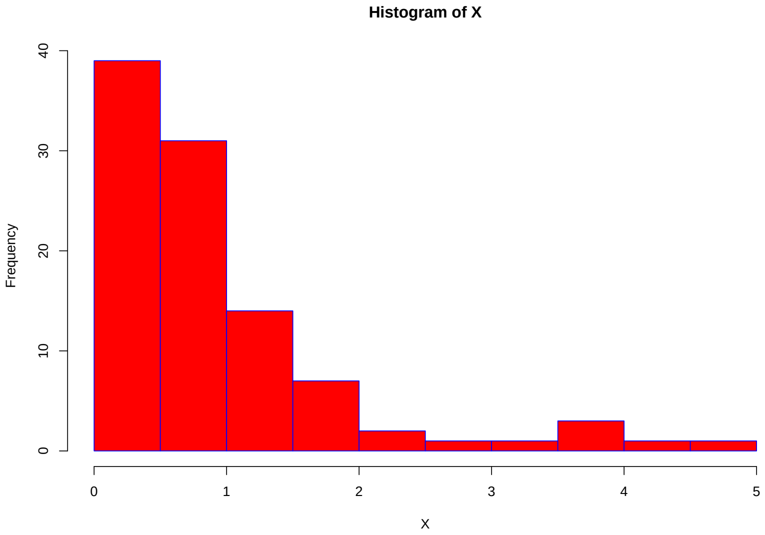
- `plot(x)`: graph of the values of x in ordinates versus $1, ..., n$.
- `plot(x,y)`: scatterplot of points in y in ordinates and x in abscissa.
- `coplot(x~y/z)`: all scatterplots (x, y) for each value of z.
- `boxplot(x)`: boxplot of x.
- `pairs(x)`: plot all possible scatterplots on columns of x (`data.frame` or `matrix`).
- `hist(x)`: histogram of counts of x.
- `barplot(x)`: barplot of x.
- `matplot(x,y)`: scatterplots (column 1 of x, column 1 of y), (column 2 of x, column 2 of y), . . .
- `curve(f)`: curve of f.
- `qqnorm(x)`: quantiles of x in function of expected values of a normal distribution.
- `qqplot(x,y)`: quantiles of y in function quantiles of x.
- `persp(x,y,z)`: perspective plot.

Primary functions: options

param	description
add	If T, overlay the graph to the existing one. F by default.
col	Filling color.
axes	If FALSE, does not plot axes and box. TRUE by default.
log='x', log='y', log='xy'	Consider x-axis, y-axis, or both as logarithmic
type = 'p'	Type of graph to plot: 'p' for points, 'l' for lines...
xlab, ylab, xlim, ylim	Name of x-axis/y-axis (character). Fix axes limits.
main, sub	Title and subtitle of the figure.

Primary functions: options

```
X = rexp(100); hist(X, border="blue", col="red")
```

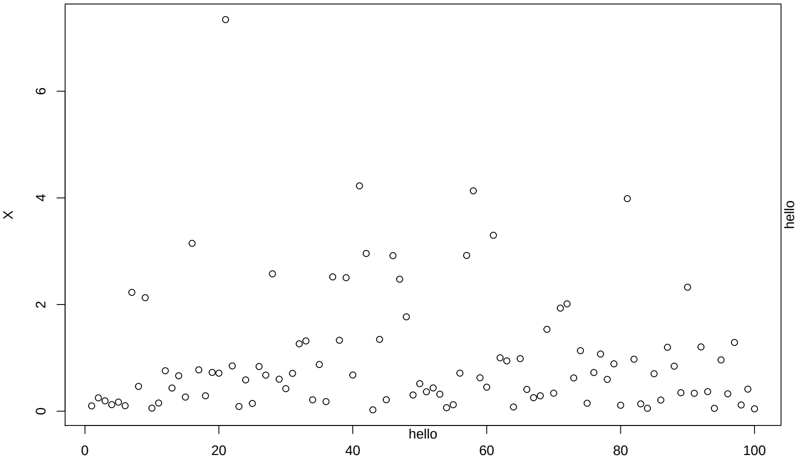


Secondary graphical functions

- `points(x,y)` overlays points to the existing graph.
- `lines(x,y)` same with line segments.
- `text(x,y,labels)` adds specified text by labels to the point (x, y).
- `mtext(text, side)` adds text to a side margin.
- `segments(x0,y0,x1,y1)` draws a segment from (x_0, y_0) to (x_1, y_1).
- `arrows(x0,y0,x1,y1,angle,code)` same with arrows.
- `abline(h)` adds an horizontal line on y-axis.
- `abline(v)` adds an vertical line on x-axis.
- `abline(a,b)` idem with slope b and intercept a.
- `legend(x,y,legend)` adds the legend to point (x, y).
- `title()` adds a title.
- `locator()` locates a point on the graphical window.

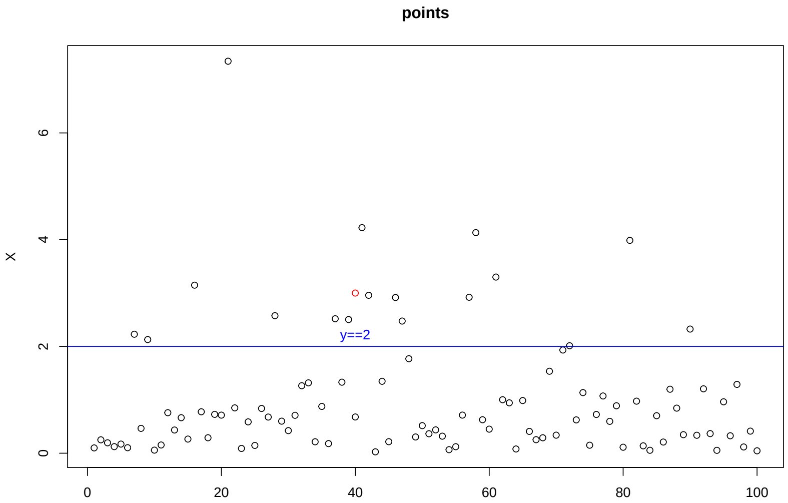
Secondary graphical functions

```
X = rexp(100); plot(X)
mtext("hello",side = c(1,4))
locator()
```



Secondary graphical functions

```r
plot(X); title("points")
points(40,3,col="red"); abline(h=2, col="blue")
text(40,2.2,expression("y==2"),col="blue")
```



The command `par()`

Graphs are created in function of graphical parameters defined by default and midifiable using the command `par()`.

Below are some examples:

- `par(bg="red")` defines the background color.
- `par(xlog=TRUE)` applies a logarithmic scale on x
- `par(mfrow=c(3,2))` splits the graphical window in 3 rows and 2 columns. Default is one graph per window.

There are **73 graphical parameters** that can be set using `par()`.

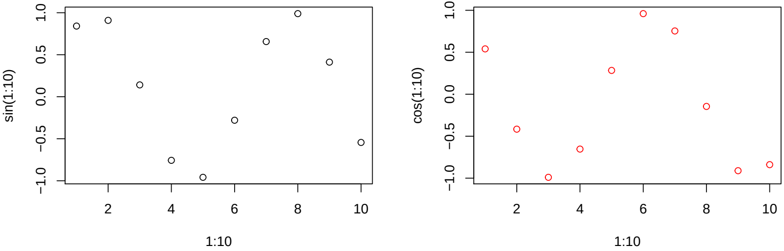
The command `par()`

param	description
adj	Justify the text with respect to the left edge.
bg	Background color: 657 colors available with colors()
bty	Controls how the box is plotted (bty = ‘n’ removes the box)
col	Symbols color.
font	Font.
lty	Type of line plot (1: continuous, 2: dashes, 3: points)
mfrow, mfcol	Vector of size c(nr,nc) splitting the graphical window in nr*nc parts.
pch	Symbol type (integer between 1 and 25)
tcl	Length of the axes graduations as a fraction of the text line height.
ps	Size of text and symbols.

The command par()

It is recommended to save the previous parametrization to get it back.

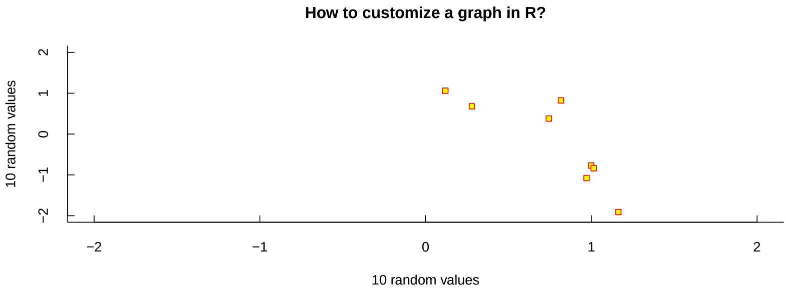
```
op=par() # save the current parametrization in op
par(mfrow=c(1,2)) # modify the parametrization
plot(1:10,sin(1:10)); plot(1:10,cos(1:10),col="red")
```



```
par(op) # get back the old parametrization
```

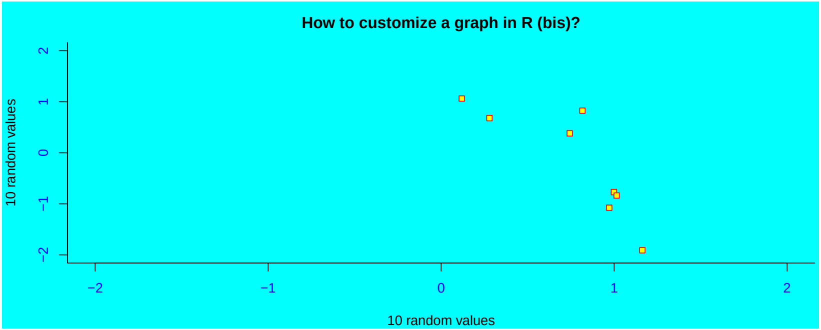
An example

```
x=rnorm(10);y=rnorm(10);
plot(x,y, xlab="10 random values",
     ylab="10 random values",
     xlim=c(-2,2), ylim=c(- 2,2),
     pch=22, col="red",
     bg="yellow", bty= "l", tcl=0.4,
     main="How to customize a graph in R?")
```



An example

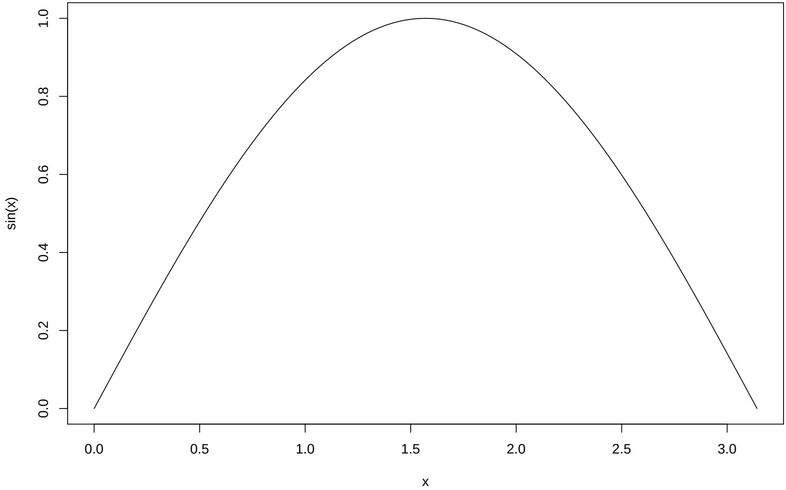
```
op=par()
par(bg="cyan", col.axis="blue", mar=c(4,4,2.5,0.25))
plot(x, y, xlab="10 random values",
     ylab="10 random values",
     xlim=c(-2,2), ylim= c(-2,2), pch=22, col="red",
     bg="yellow", bty="l")
title("How to customize a graph in R (bis)?")
```



```
par(op)
```

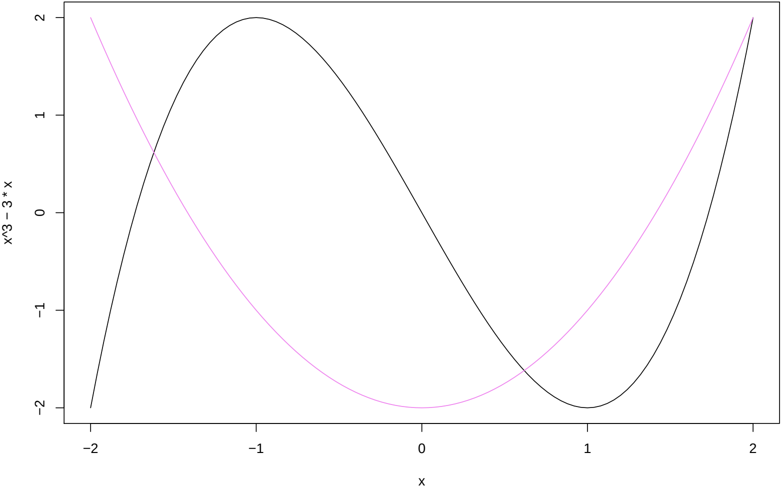
Other examples: curve()

```
curve(sin,0, pi)
```



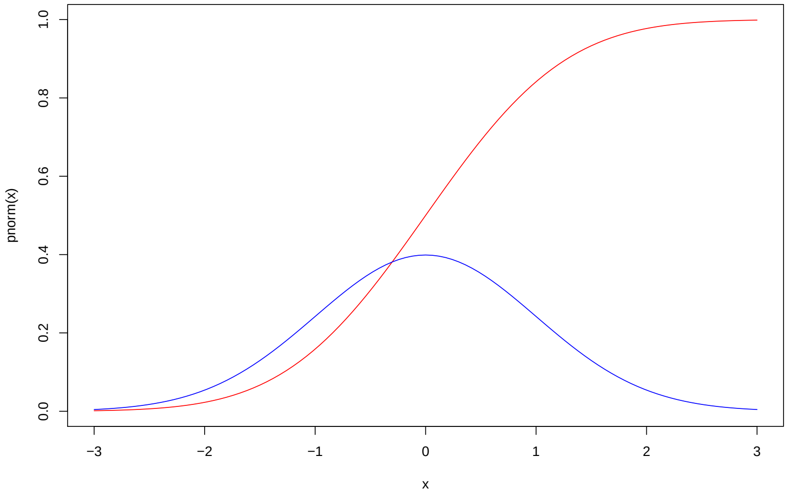
Other examples: curve()

```
curve(x^3-3*x, -2, 2)
curve(x^2-2, add = TRUE, col = "violet")
```



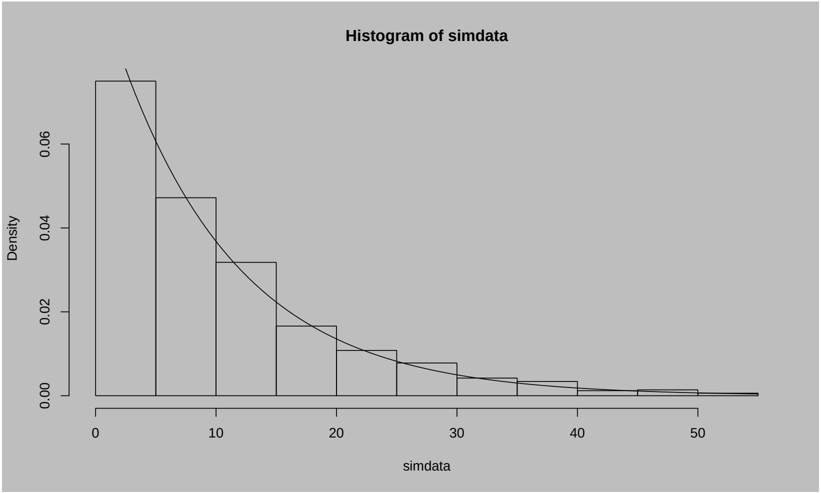
Other examples: curve()

```
curve(pnorm(x),-3,3, col="red")
curve(dnorm(x),-3,3, col="blue", add = TRUE)
```



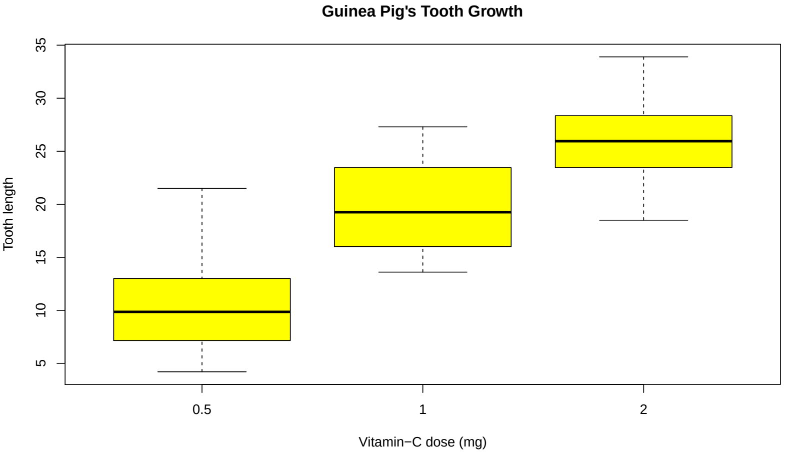
Other examples: `hist()`

```r
par(bg="gray")
simdata = rexp(1000, rate = 0.1)
hist(simdata, prob=T)
curve(dexp(x, rate=0.1), add=TRUE)
```



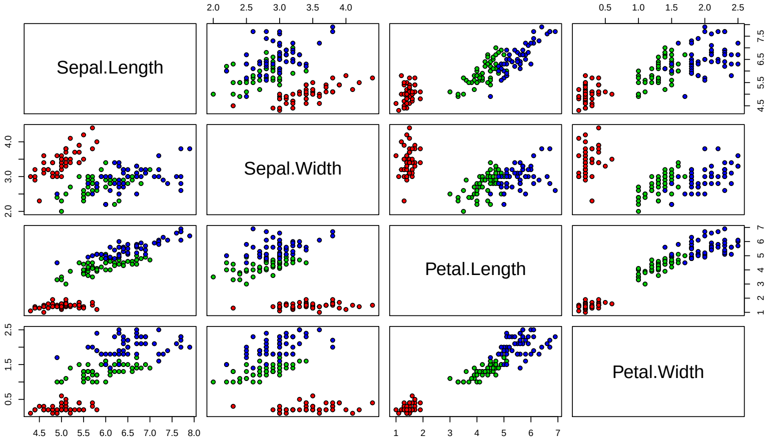
Other examples: boxplot()

```
boxplot(len ~ dose, data = ToothGrowth,
        col="yellow",xlab="Vitamin-C dose (mg)",
        ylab="Tooth length")
title("Guinea Pig's Tooth Growth")
```



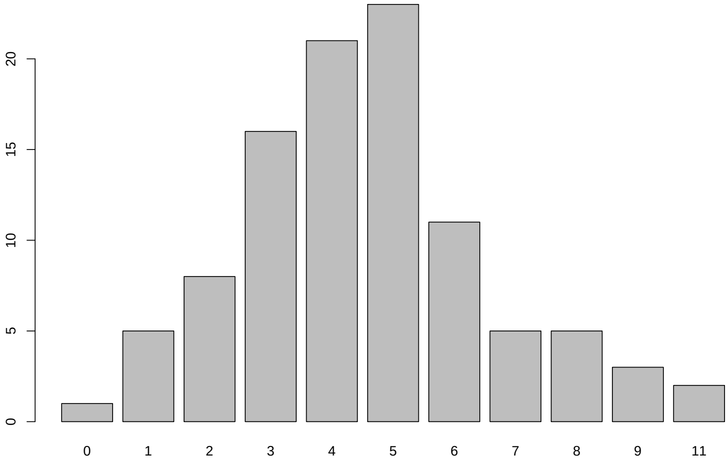
Other examples: pairs()

```r
pairs(iris[1:4],  pch = 21,
      bg = c("red", "green3", "blue")[unclass(iris[,5])])
```



Other examples: barplot()

```
t = table(rpois(100,lambda=5))
r = barplot(t, col='gray')
```


0
5
10
15
20
0
1
2
3
4
5
6
7
8
9
11