Arguments for High-level functions

main="..."	to title the plot heading
sub="..."	to subtitle the plot below
xlab="...", ylab="..."	to label the axes
axes="FALSE"	suppress axis lines, tick marks and tick labels
xlim=c(min,max), ylim=c(min, max)	minimum and maximum values for each axis

type="plbohn"	p=points, l=lines, b=both, o=overstruck,
	h=high density, n=none

For example, a labelled display is created by

```
plot(z[,3],z[,2],main="Plot of Maximum versus Minimum Soil Temperature",
        sub="Figure 1.  An example of a figure",xlab="Min",ylab="Max")
```

Graphical parameters for controlling primitive characteristics of the figure include

cex= character expansion. 0.5 means half regular size, 1.5 means one and a half times regular size

las= label orientation. 0 is parallel to axis, 1 is horizontal

pch= plotting character. Use *, . or a numeric character to identify symbols

lty= line type. 1 is solid, 2,3.. give other types

lwd= line width. >1 thicker, <1 thinner

col= colour. default is 1, other integers give predefined colours

Graphical parameters for controlling the layout include

pty= plot shape m is maximum for device, s is square

mar=c(m1,m2,m3,m4) controls the size of margins. Default is 5,4,4,2 lines of text

oma=c(om1,om2,om3,om4) size of outer margins for arrays of plots. Default is 0,0,0,0

mfrow=c(r,p) an array of p figures in r rows

mfcol=c(c,p) an array of p figures in c columns

For displays with multiple plots, the individual margins should be as small as possible.

Low level functions include

points(x,y) draw a set of plotting characters at co-ordinates (x,y)

lines(x,y) draw a set of lines connecting the co-ordinates (x,y)

polygon(x,y) fill the polygon outlined by the co-ordinates (x,y)

segments(x1,y1,x2,y2) draw a set of segments from (x1,y1) to (x2,y2)

arrows(x1,y1,x2,y2) draw a set of segments with arrows from (x1,y1) to (x2,y2)

title(main,sub.xlab,ylab,axes=F) adds titles

axes(main,sub.xlab,ylab,axes=T) adds titles

axis(side,at,labels=T,ticks=T,pos) adds tick marks and/or labels to side axis at user defined co-ordinate at

text(x,y,"...") add text at (x,y)

mtext(side,line,"...") prints text in margin on side (1,2,3 or 4) parallel to the side, line lines of text away from the plot

legend(x,y,c("...",...,"..."),symb) draws a box with lines, points, marks and/or shading given by symb identified by the given text strings)

abline(a,b) draw y= a + bx

locator(n) accept n mouse identified points (position cursor at desired position and hit button 1, buttons 2 or 3 terminate the input)

identify(x,y,labels) find (x,y) co-ordinates of mouse identified points

The functions text, title, mtext, legend and axis are useful for adding explanatory information to figures. Note that \n creates a new line in text and text strings are controlled by the arguments

adj= justification 0 left .5 centre 1 right

srt= rotation in degrees

crt=character rotation in degrees

High level functions can be coerced into acting like low level functions by using the add=TRUE argument. This argument means that the result is added to the existing figure rather than creating a new figure.

Examples

1. The identify function can be used to identify points from the plot for subsequent action. For example, to remove points identified on a scatter plot from a data set, type

```
bad <- identify(x,y,plot=FALSE)
```

Then use the left button to click on the points. When finished, click on the right button and then type

```
xgood <- x[-bad]

ygood <- y[-bad]
```

2. The legend function can be very conveniently used in conjunction with locator as

```
text(locator(1),"Regression line")

legend(locator(1),legend=c("IBM","AT&T","GM"),lty=c(1:3))
```

3. To plot a sequence of numeric or character values in vec, type

```
plot(x,y,type="n")

text(x,y,label=vec)
```

4. To add text to selected points

```
text(x[select],y[select],label=vec)
```

5. To label an axis, type

```
plot(x,y,axes=F)

axis(2)

axis(1,seq(-3:3),c("a","b","c","d","e","f","g"))
```

6. To set axis ranges so two plots have the same y-axis, type

```
plot(x1,y1,ylim=range(y1,y2))

plot(x2,y2,ylim=range(y1,y2))
```

To obtain square plots with the same axes, type

```
par(pty="s")

plot(x,y,xlim=range(x,y),ylim=range(x,y))
```

7. To add a vertical {horizontal} line through x = 4 {y = 4}, type

```
abline(v=4)     {abline(h=4)}
```

" Drawing graphs, like motor-car driving and love-making, is one of those activities which almost every statistician thinks he can do well without instruction.
The results are of course usually abominable. "

Quoted by H.Wainer.