Advanced Scientific Computing with R

4. Plots

Michael Hahsler

Southern Methodist University

February 16, 2014

These slides are largely based on "An Introduction to R"
http://CRAN.R-Project.org/

Table of Contents

Introduction

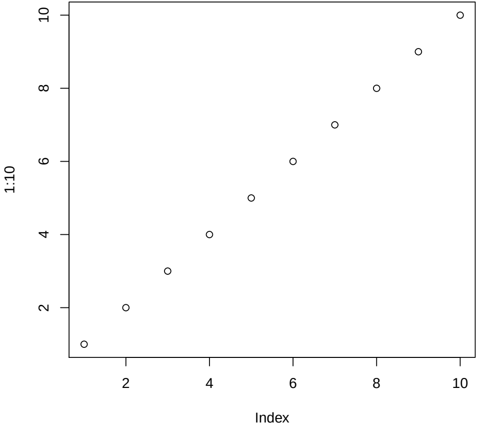
- Ploting is an integral part of R.
- R plots on devices (e.g., `X11()`, `quarz()`, `windows()`, `pdf()`, `png()`)
- Plotting commands are divided into three basic groups:
  - **High-level plotting functions** create a new plot on the graphics device, possibly with axes, labels, titles and so on.
  - **Low-level plotting functions** add more information to an existing plot, such as extra points, lines and labels.
  - **Interactive graphics functions** allow you interactively add information to, or extract information from, an existing plot, using a pointing device such as a mouse.

We will only discuss 'base' graphics. An advanced graphics sub-system called 'grid' also exists.

Table of Contents

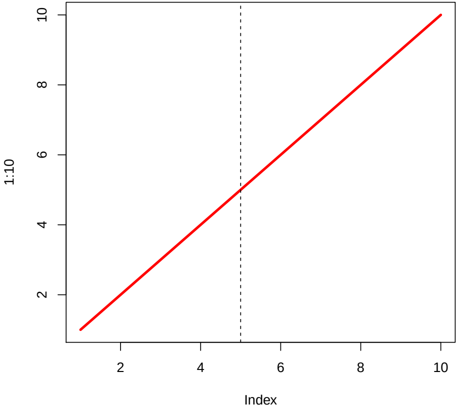
plot

```
R> plot(1:10)
```



plot

```
R> plot(1:10, type="l", col="red", lwd=3)
R> abline(v=5, lty=2)
```



Getting help for plot

`>? plot`

Shows that plot is a so called generic function. Generic functions have implementations dor different data types which get "dispatched" at call-time.

`>? plot.default`

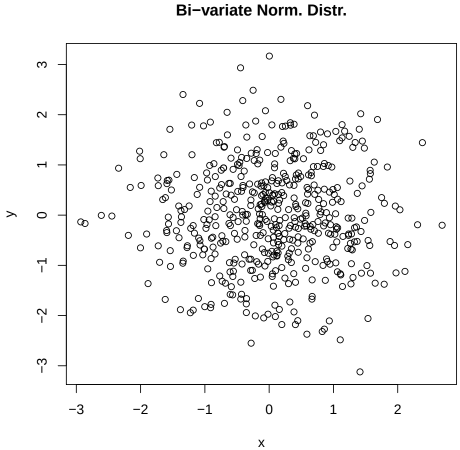
This is the default function for plot.

`>? par`

Graphical parameters which typically can be passed on as `...` to plot.

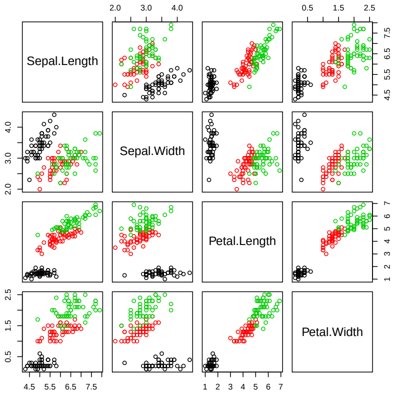
Scatterplot

```
R> plot(x=rnorm(500), y=rnorm(500), xlab="x", ylab="y",
+     main="Bi-variate Norm. Distr.")
```



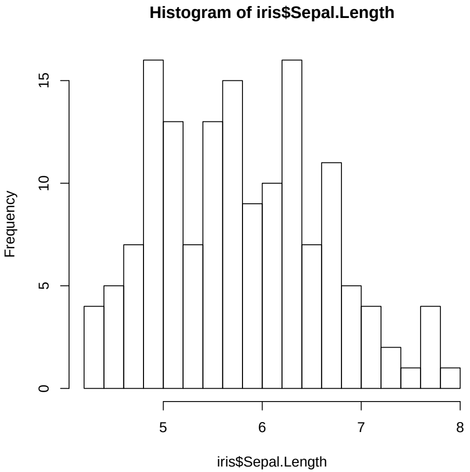
Scatterplot matrix (pairs)

```
R> data(iris)
R> head(iris, n=1)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
R> plot(iris[,-5], col= iris[,5])
```



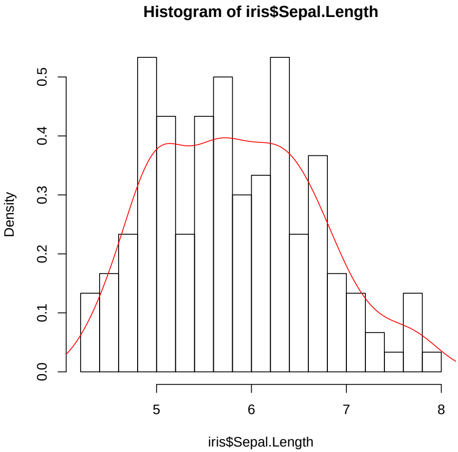
hist - Histogram

```
R> hist(iris$Sepal.Length, breaks=20)
```



hist - Histogram with estimated density

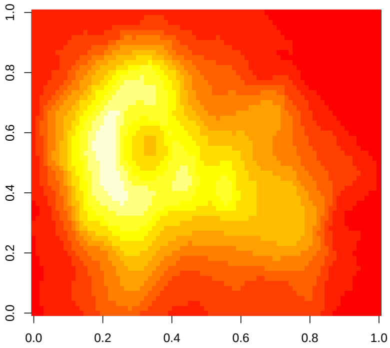
```
R> hist(iris$Sepal.Length, breaks=20, prob=TRUE)
R> lines(density(iris$Sepal.Length), col="red")
```



image

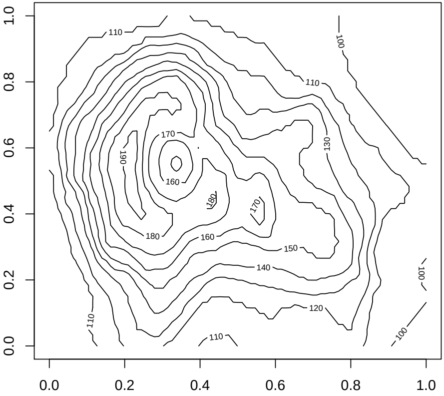
volcano is a R data set with elevation measurements of Maunga Whau on a 10m by 10m grid.

```
R> dim(volcano)
[1] 87 61
R> image(volcano)
```

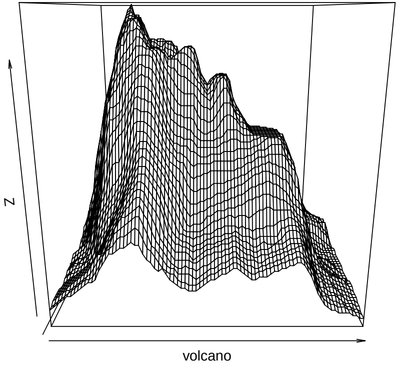


contour

```
R> contour(volcano)
```



```
R> persp(volcano)
```



Typical Arguments for plot functions

- `add=TRUE`: Add to an existing plot?
- `axes=FALSE`: Plot axes?
- `log="x"`, `log="y"` or `log="xy"`: Log. axes?
- `type="l"`: Plot lines instead of points
- `xlab, ylab`: Axis labels
- `main`: Figure title
- `sub`: Sub-title

Table of Contents

Some low-level functions

These functions can be used to add elements to a plot.

- `points(x, y)`
- `lines(x, y)`
- `text(x, y, labels, ...)`
- `abline(a, b)` or `abline(h=y)` or `abline(v=x)`
- `polygon(x, y, ...)`
- `legend(x, y, legend, ...)`
- `title(main, sub)`
- `axis(side, ...)`

Graphical parameter list: par

R maintains a list of graphics parameters to control line style, colors, figure arrangement and text justification.
A separate list of graphics parameters is maintained for each active device.

```
R> oldpar <- par(col=4, pch=4)
R> par(oldpar)
```

Many parameters from `par()` can also be passed to `plot()`.
Try `par()` and `?par`

Important parameters in par

- `pch=4`: Plotting symbol (0-25)
- `lty=2`: Line type
- `lwd=2`: Line width
- `col=2`: Color for points, lines, etc.
- `cex=1.5`: Character expansion (e.g., 50% larger than default text size)
- `mai=c(1, 0.5, 0.5, 0)`: Widths of the bottom, left, top and right margins, respectively, measured in inches.

Saving a plot as an image

```
R> png(file="plot.png")
R> plot(1:10)
R> dev.off()
pdf
  2
```

Other devices are `jpeg()`, `tiff()`, `pdf()`, `postscript()`, `win.metafile()` (Windows).
Use `?Devices` for a complete list.

Interactive and Advanced Graphics

Interactive Graphics are available via several extension packages:

- **ggplot2**: Grammar of graphics.
- **rggobi**: GGobi interactive graphics system.
- **iplots**: Java based plotting (alpha blending, brushing, selection, etc.)
- **playwith**: Build interactive versions of R graphics
- **rgl**: OpenGL

Advanced Graphics

- **ggplot2**: Grammar of graphics. Produces elegant visualizations (see `http://ggplot2.org/`).
- **grid**: Advanced graphics can be programmed using flexible low level ploting functions (viewports, different coordinate systems and units, lines, points, text, etc.) See also package **lattice**.

Table of Contents

Exercises

1. Plot a `sin(x)/x`. Hint: Trigonometric functions in R use angles in radians (see `sin`)
2. The "cars" data set gives the speed of cars and the distances taken to stop. Note that the data were recorded in the 1920s. Plot the "cars" data set as a scatter plot. Plot all data points with distances taken to stop greater than 80 in red.
3. Plot histograms for speed and dist in "cars".