Graphic Libraries

Gordon Erlebacher

Objectives

- Examine the myriad of possibilities of graphic display in R

How

- Step 1
  - What does R make available by default
- Step 2
  - run example(cmd) on some of the commands we find to explore possibilities
- Step 3
  - examine the list of available packages
- Step 4
  - Use ?? to find out what graphic tools exist
- Step 5
  - Use Google or other search engine to find literature or more examples

Approach

- Before becoming submerged in details
  - first find out what exists
  - what have people done?
    - do not reinvent the wheel
  - what have people written about?

Graphics
What is available

- ??graphics   # packages already on my computer
- library(sos)   # list of functions in “sos”
  library(help=sos)
  findFn(“graphics”)
  # Downloaded 307 links in 158 packages.
- Let us explore findFn some more

findFn

- fn = findFn("graphics")
- names(fn)
  # examine "package" and "function"
- fn$package
  fn$function
  fn$link

? versus ??

- ?xxx :   help on the command xxx
- ??xxx : what are all the commands that involve xxx?

List of packages

http://cran.r-project.org/src/contrib/Archive/

Each library contains a list of functions

stats library

```
help("stats")    or    help(stats)
```

List of functions in stats library

```
library(help=stats)

# does not work in RStudio
# works on the command line
```

??graphics

```
grDevices::palette      Set or View the Graphics Palette
grDevices::pdf          PDF Graphics Device
grDevices::pictex       A PicTeX Graphics Driver
grDevices::postscript   PostScript Graphics
grDevices::recordGraphics
                        Record Graphics Operations
grDevices::xfig         XFig Graphics Device
grDevices::svg          Cairo-based SVG, PDF and PostScript Graphics
                        Devices
grDevices::png          BMP, JPEG, PNG and TIFF graphics devices
grDevices::x11          X Window System Graphics
grid::Grid              Grid Graphics
grid::gpar              Handling Grid Graphical Parameters
grid::grid-package      The Grid Graphics Package
grid::grid.add          Add a Grid Graphical Object
grid::grid.collection   Create a Coherent Group of Grid Graphical
                        Objects
grid::grid.copy         Make a Copy of a Grid Graphical Object
grid::grid.edit         Edit the Description of a Grid Graphical Object
grid::grid.get          Get a Grid Graphical Object
grid::grid.grob         Create a Grid Graphical Object
grid::grid.null         Null Graphical Object
grid::grid.remove       Remove a Grid Graphical Object
grid::grid.set          Set a Grid Graphical Object
gWidgets::gWidgets-classes
```

and many more packages

Current version of RStudio only lists packages on my computer, so the list with current version of RStudio will be shorter.

Use findFn for more extensive information

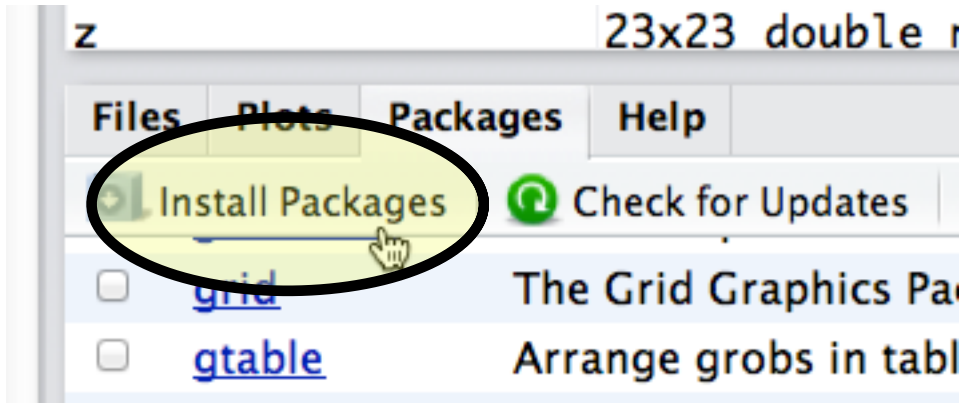
Some packages

- Some packages related to graphics:
  - graphics   (included in R)
  - lattice (very popular)
  - ggplot2
  - grid
  - gWidgets
  - maps

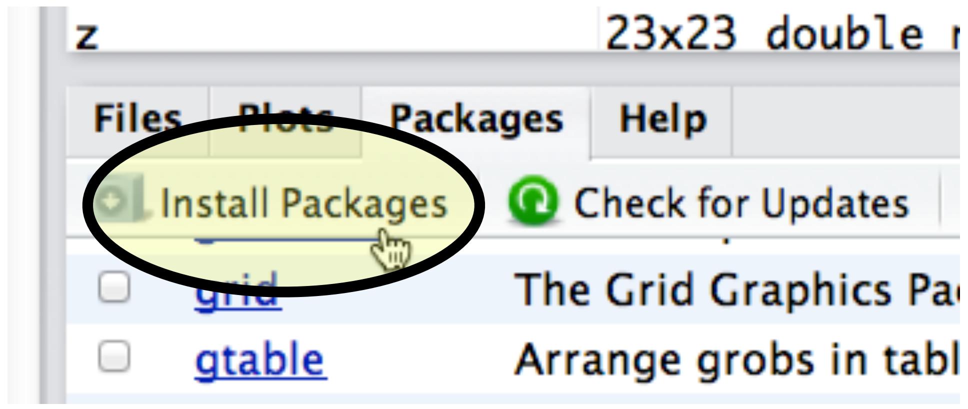
In RStudio

```
> library(maps)

Error in library(maps) : there is no package called 'maps'
```

Library must be retrieved some archive located in some repository on the internet



In RStudio

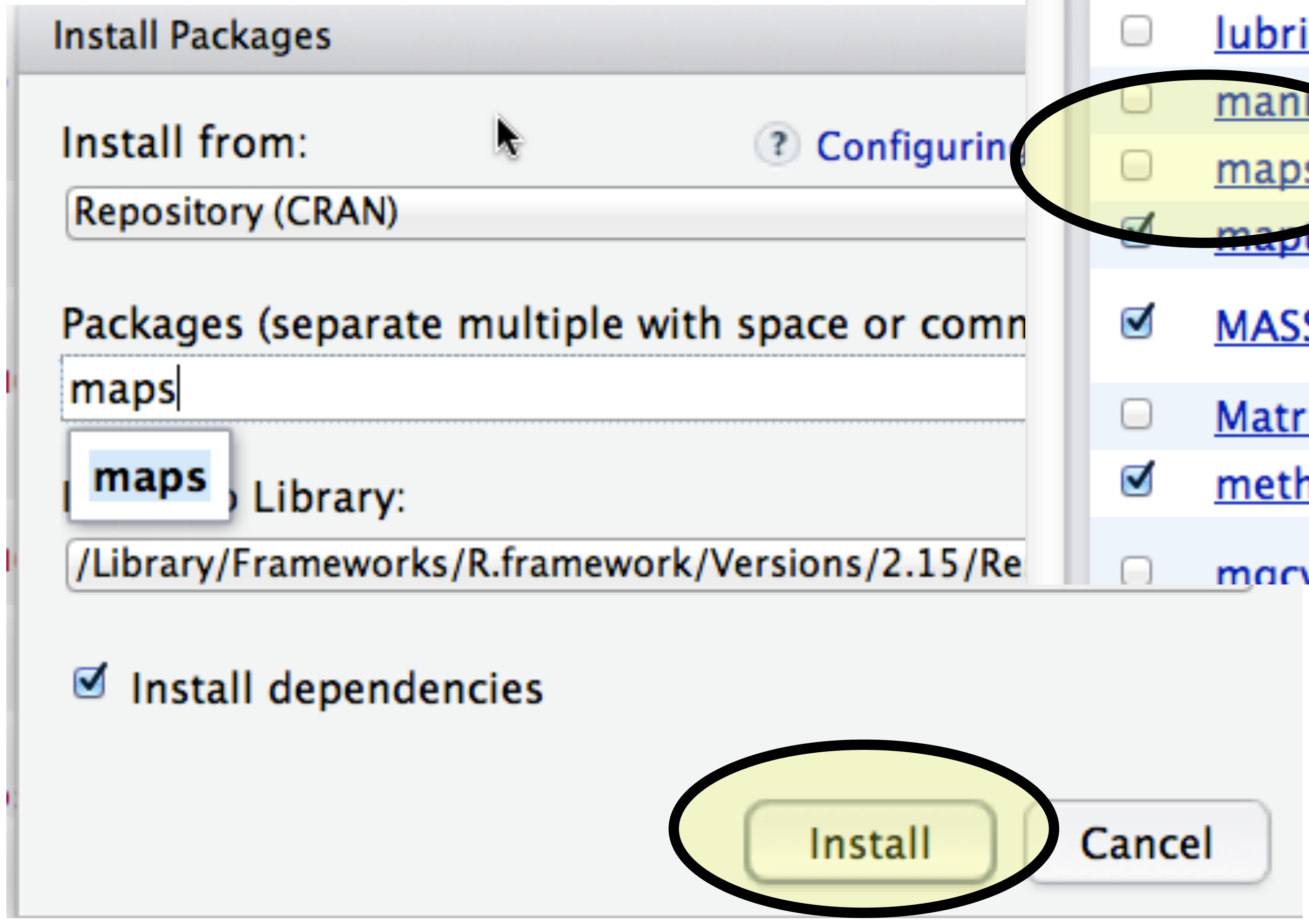


```
z                23x23 double
```

Files | Plots | Packages | Help

Install Packages | Check for Updates

☐	grid	The Grid Graphics Pa
☐	gtable	Arrange grobs in tabl



Install Packages

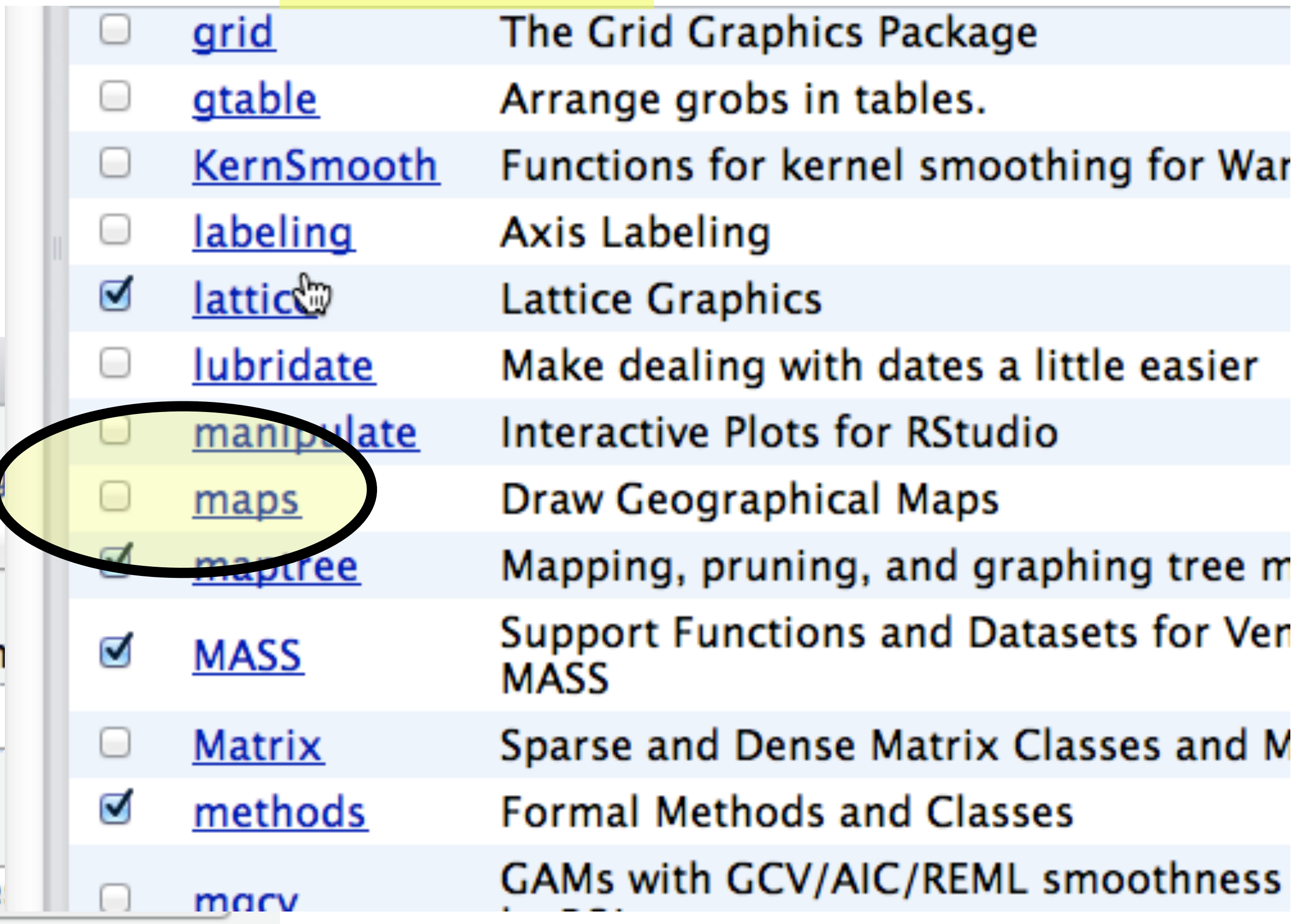
Install from: Repository (CRAN)

Packages (separate multiple with space or comn: maps

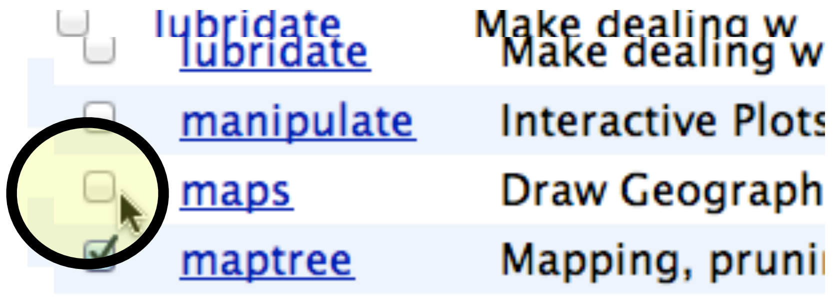
Library: /Library/Frameworks/R.framework/Versions/2.15/Re

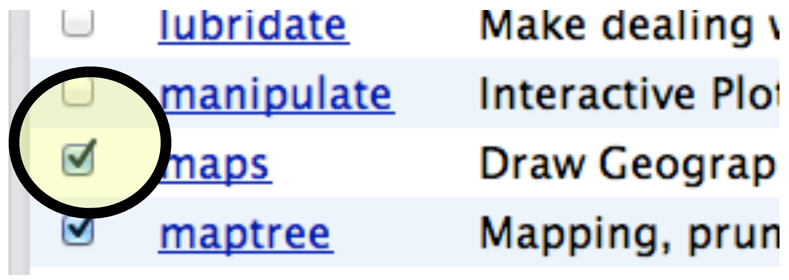
☑ Install dependencies

Install | Cancel



☐	grid	The Grid Graphics Package
☐	gtable	Arrange grobs in tables.
☐	KernSmooth	Functions for kernel smoothing for War
☐	labeling	Axis Labeling
☑	lattice	Lattice Graphics
☐	lubridate	Make dealing with dates a little easier
☐	manipulate	Interactive Plots for RStudio
☐	maps	Draw Geographical Maps
☑	maptree	Mapping, pruning, and graphing tree m
☑	MASS	Support Functions and Datasets for Ver MASS
☐	Matrix	Sparse and Dense Matrix Classes and M
☑	methods	Formal Methods and Classes
☐	mgcv	GAMs with GCV/AIC/REML smoothness





```
> library("maps")
Warning message:
package ‘maps’ was built under R version 2.15.2
```

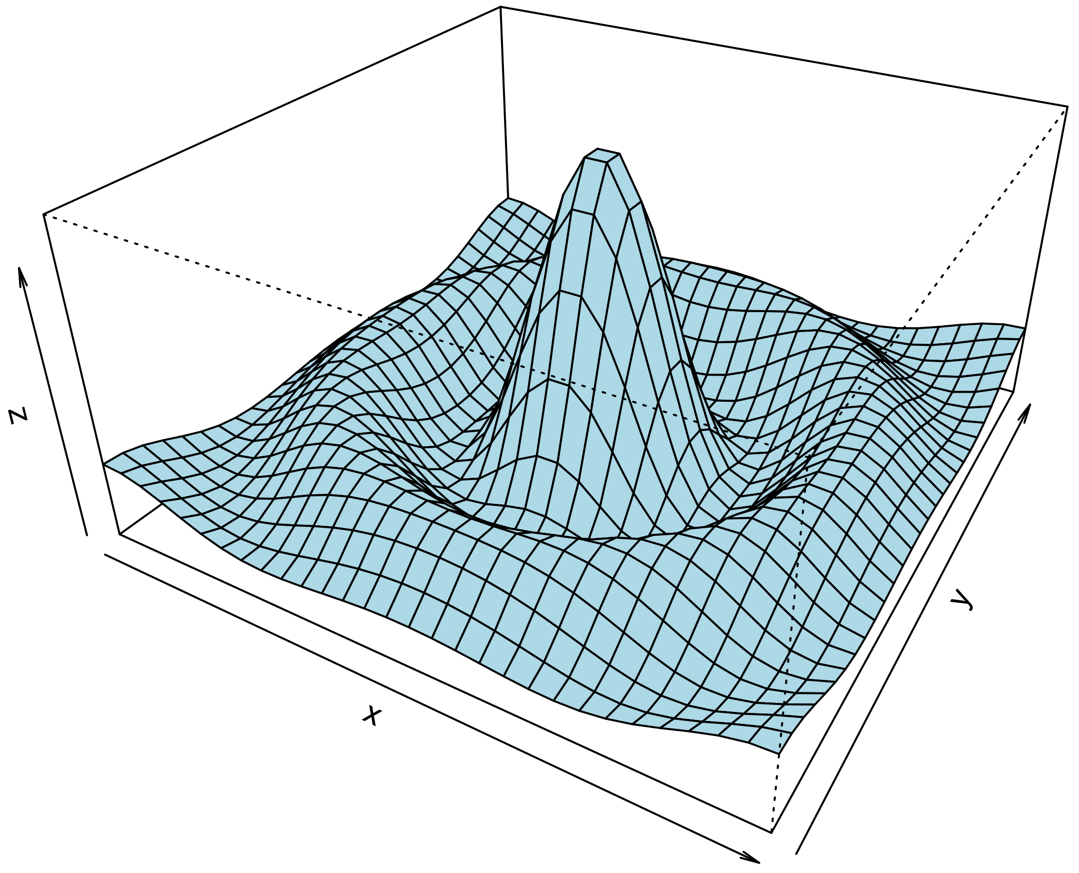
Scattergrams

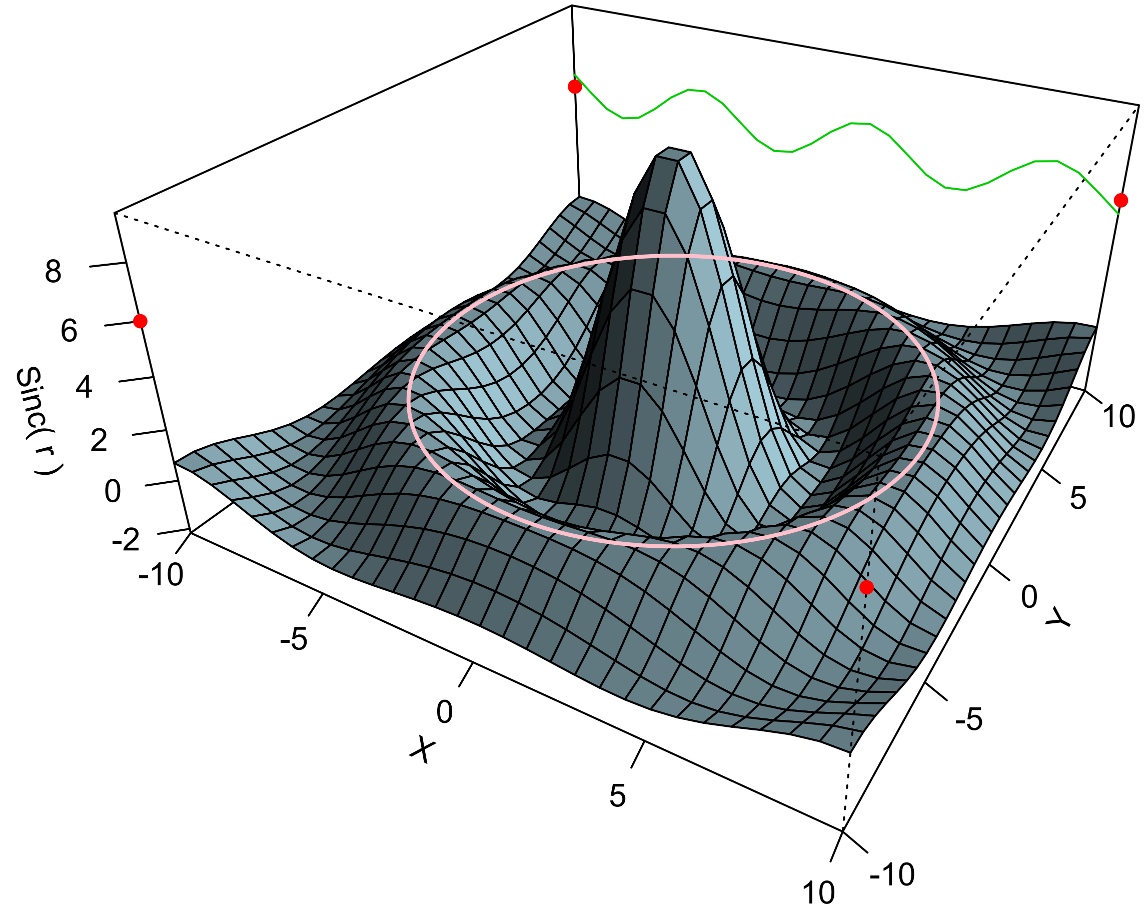
- Scattergrams appear in the packages:
  - lattice
  - scatterplot3d
  - gplots

Functions for 3D graphics built into R by default

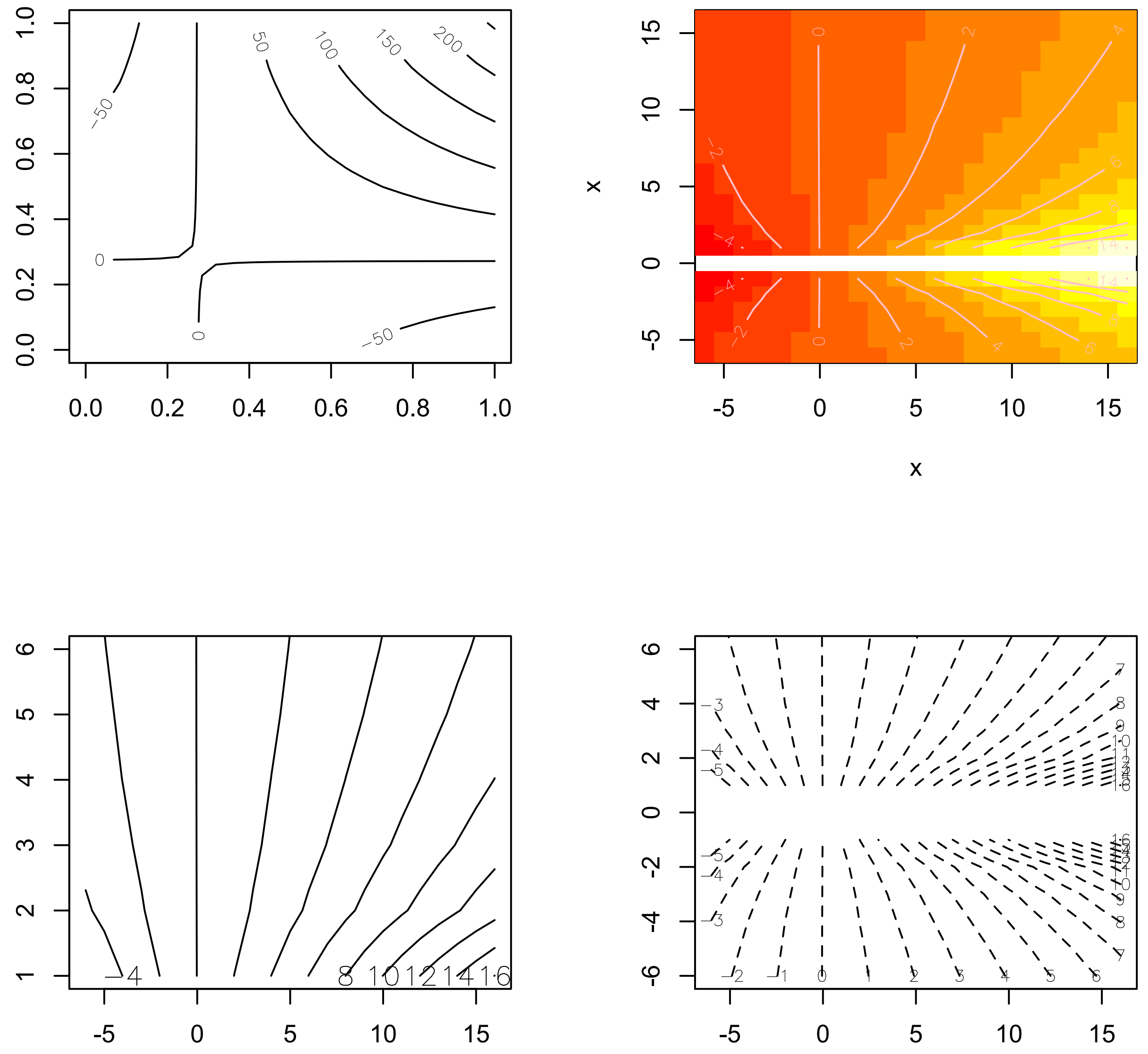
- persp(…)
- ‘contour’ and ‘image’; ‘trans3d’.
  - found in text of ?persp

persp



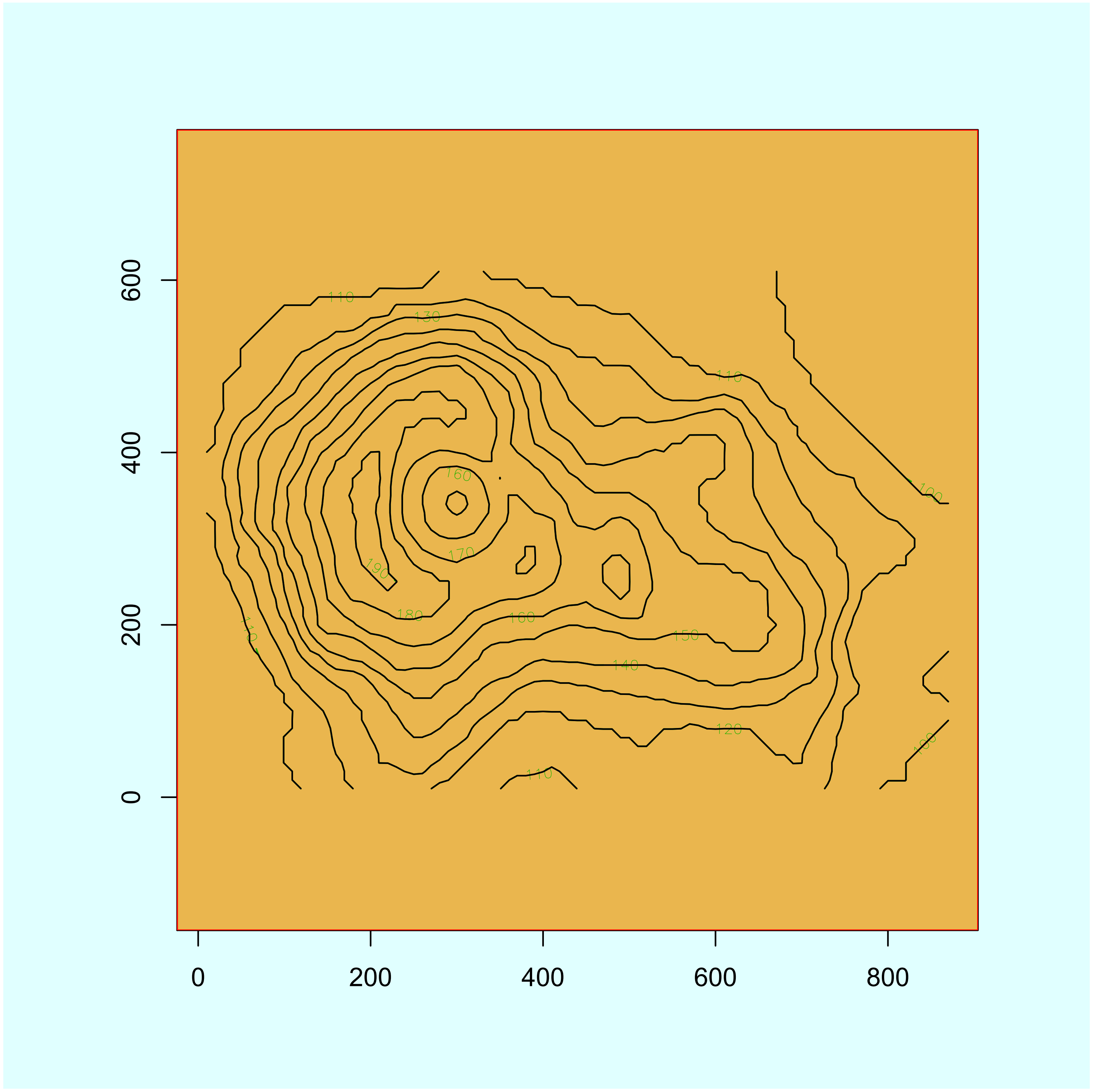


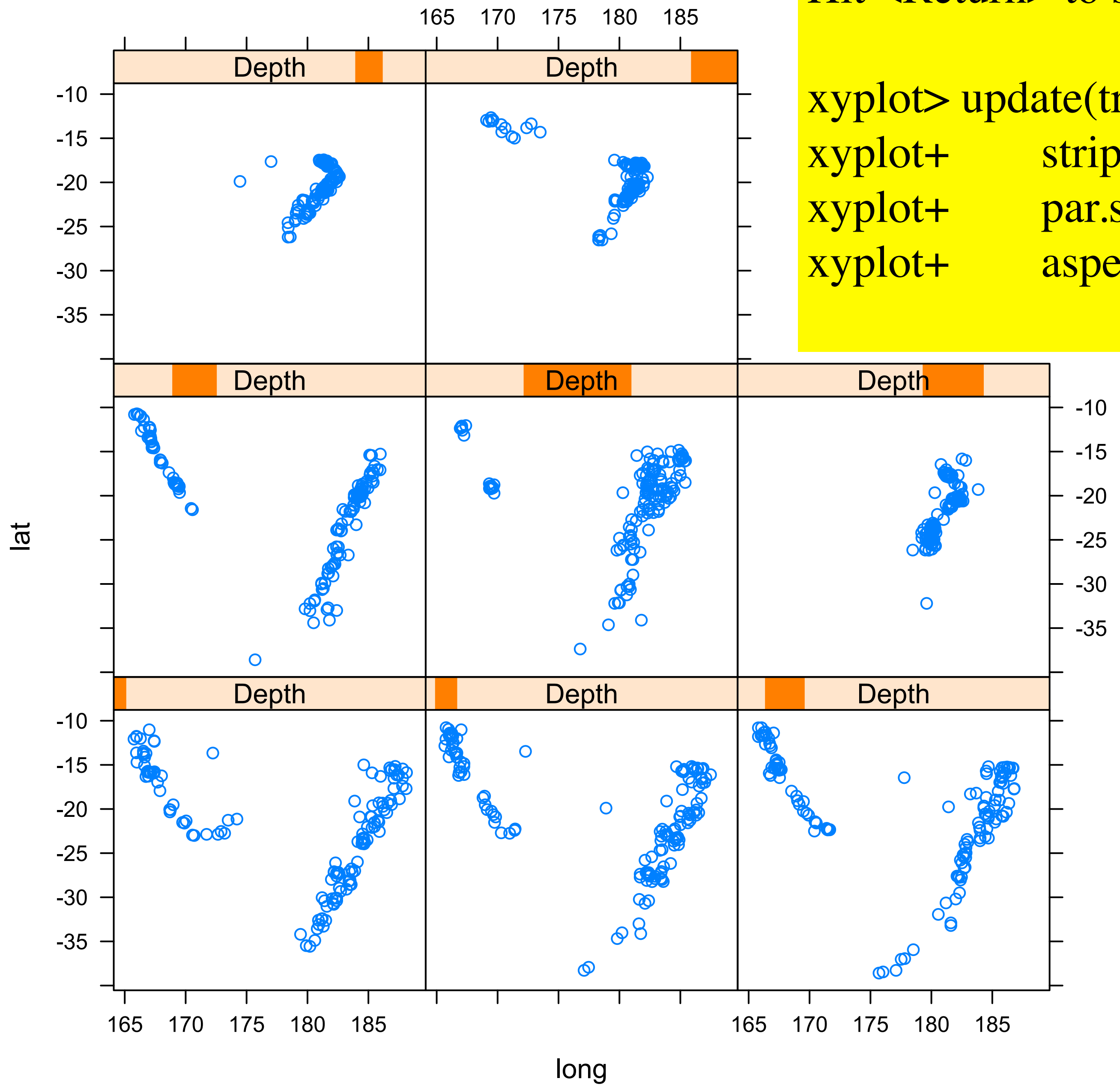
contour



0.0
0.2
0.4
0.6
0.8
1.0
-50
0
50
100
150
200
x
-5
5
10
15
-4
-2
2
4
6
8 10 12 14 16
-6
-3
-1
1
3
7

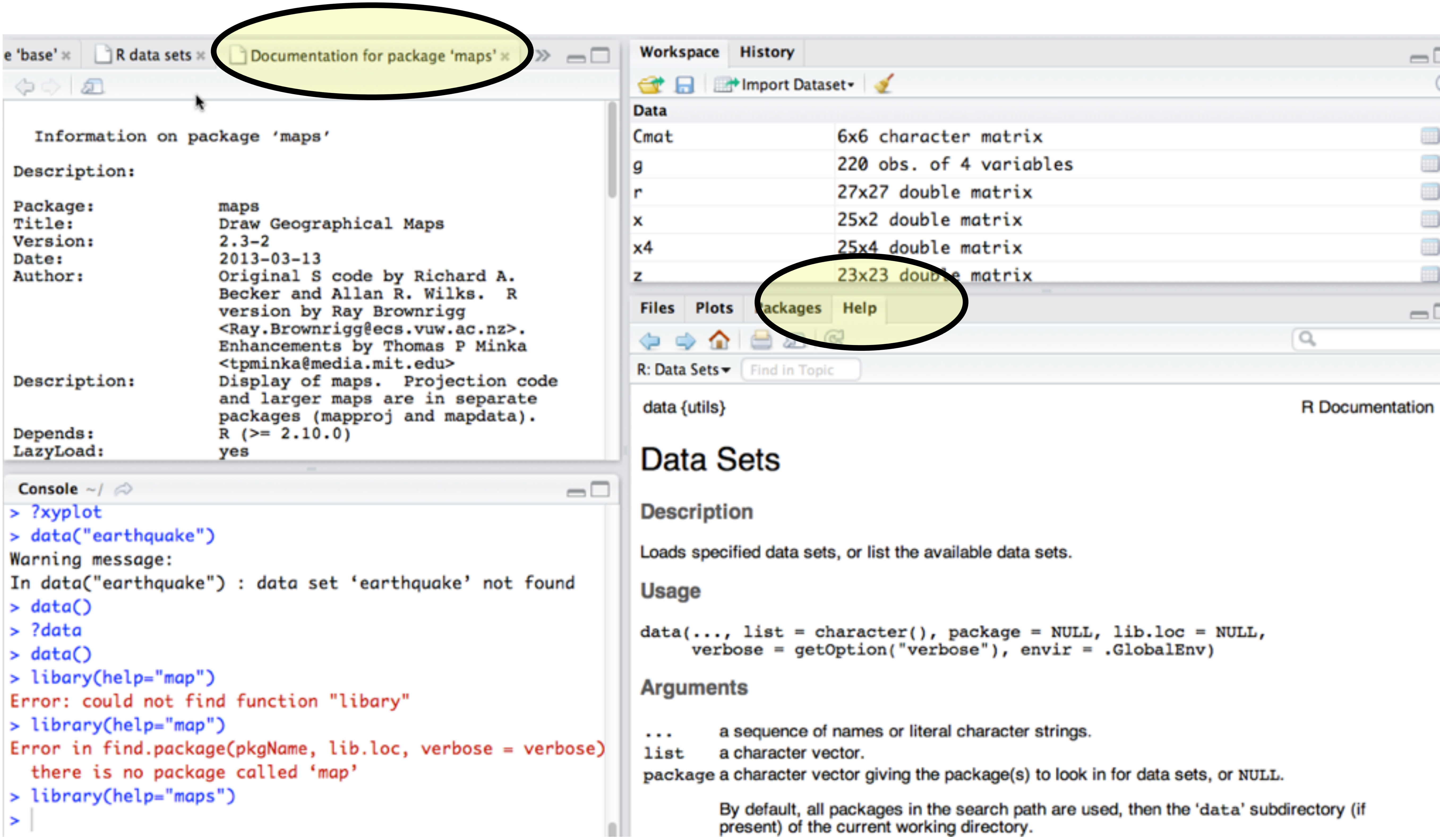
Topographic map





```
xyplot> require(stats)
xyplot>
xyplot> Depth <- equal.count(quakes$depth, number=8, overlap=.1)

xyplot> xyplot(lat ~ long | Depth, data = quakes)
Hit <Return> to see next plot:

xyplot> update(trellis.last.object(),
xyplot+        strip = strip.custom(strip.names = TRUE, strip.levels = TRUE),
xyplot+        par.strip.text = list(cex = 0.75),
xyplot+        aspect = "iso")
```

lattice::xyplot



```
  Information on package 'maps'

Description:

Package:            maps
Title:              Draw Geographical Maps
Version:            2.3-2
Date:               2013-03-13
Author:             Original S code by Richard A.
                    Becker and Allan R. Wilks.  R
                    version by Ray Brownrigg
                    <Ray.Brownrigg@ecs.vuw.ac.nz>.
                    Enhancements by Thomas P Minka
                    <tpminka@media.mit.edu>
Description:        Display of maps.  Projection code
                    and larger maps are in separate
                    packages (mapproj and mapdata).
Depends:            R (>= 2.10.0)
LazyLoad:           yes
```

Data

Cmat	6x6 character matrix
g	220 obs. of 4 variables
r	27x27 double matrix
x	25x2 double matrix
x4	25x4 double matrix
z	23x23 double matrix

```
> ?xyplot
> data("earthquake")
Warning message:
In data("earthquake") : data set 'earthquake' not found
> data()
> ?data
> data()
> libary(help="map")
Error: could not find function "libary"
> library(help="map")
Error in find.package(pkgName, lib.loc, verbose = verbose)
  there is no package called 'map'
> library(help="maps")
>
```

data {utils}   R Documentation

Data Sets

Description

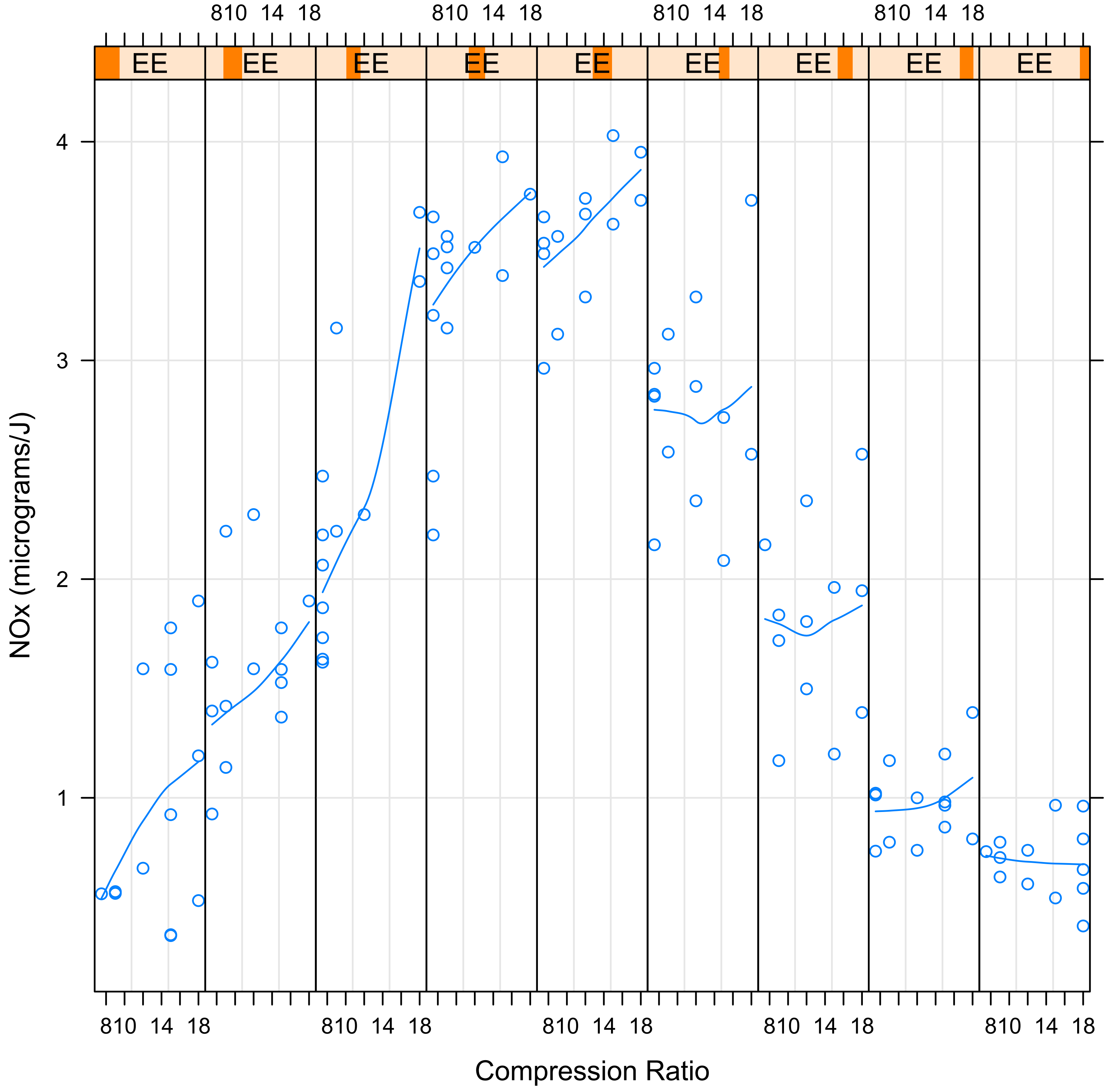
Loads specified data sets, or list the available data sets.

Usage

```
data(..., list = character(), package = NULL, lib.loc = NULL,
     verbose = getOption("verbose"), envir = .GlobalEnv)
```

Arguments

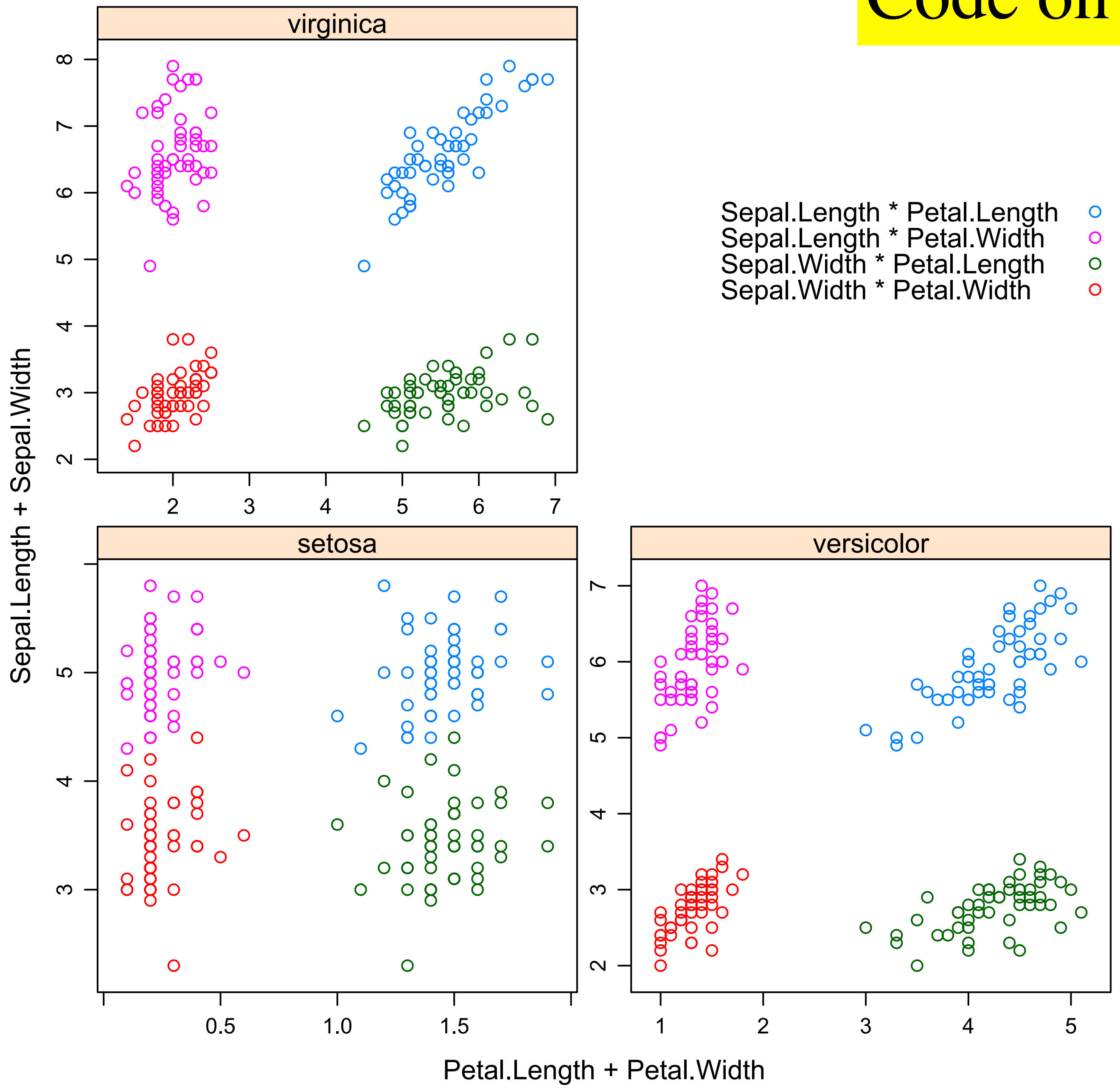
`...`	a sequence of names or literal character strings.
`list`	a character vector.
`package`	a character vector giving the package(s) to look in for data sets, or `NULL`. By default, all packages in the search path are used, then the '`data`' subdirectory (if present) of the current working directory.



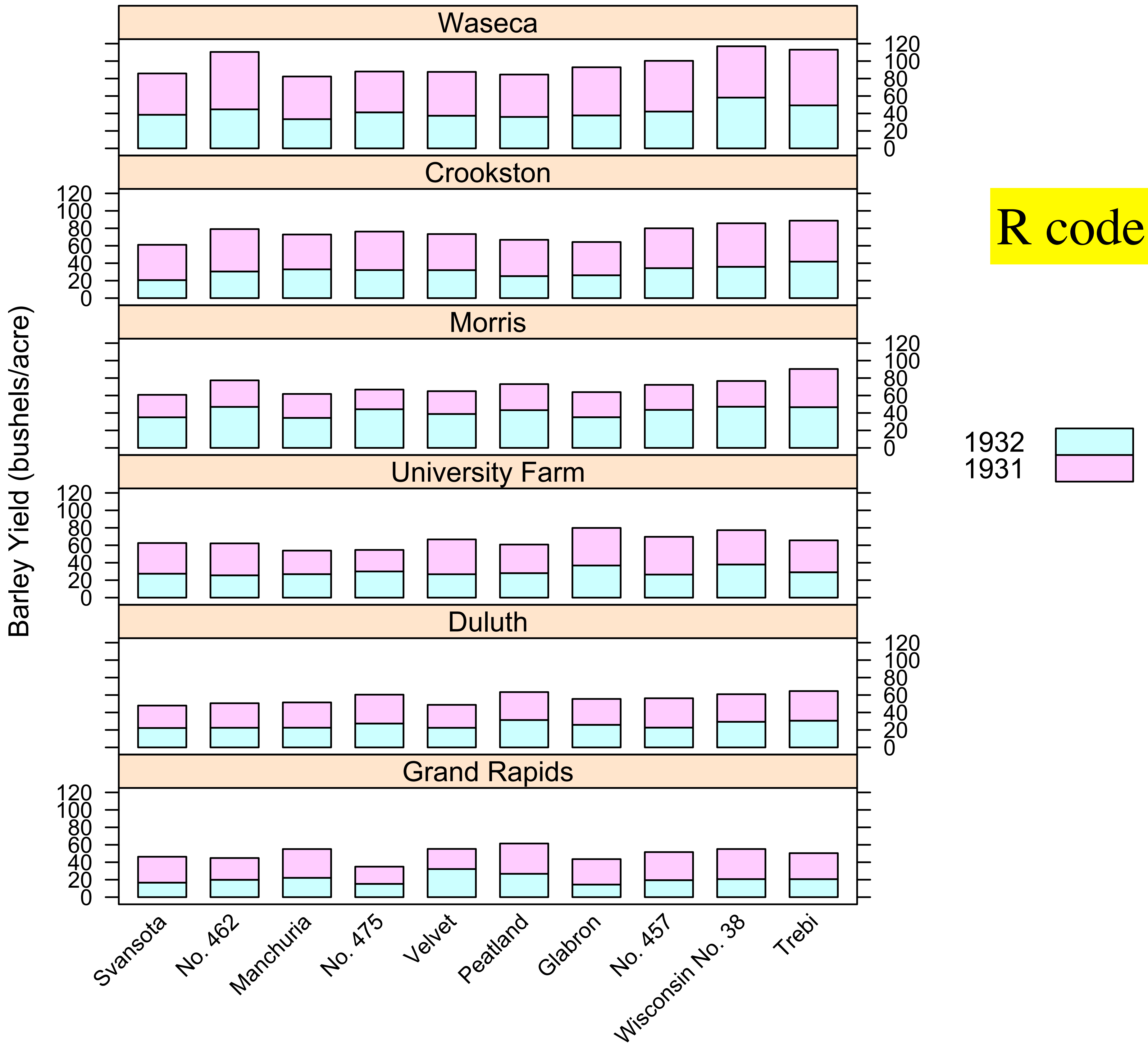
EE
EE
EE
EE
EE
EE
EE
EE
EE
810 14 18
NOx (micrograms/J)
4
3
2
1
Compression Ratio

Clusters (with Lattice)

Code on next slide

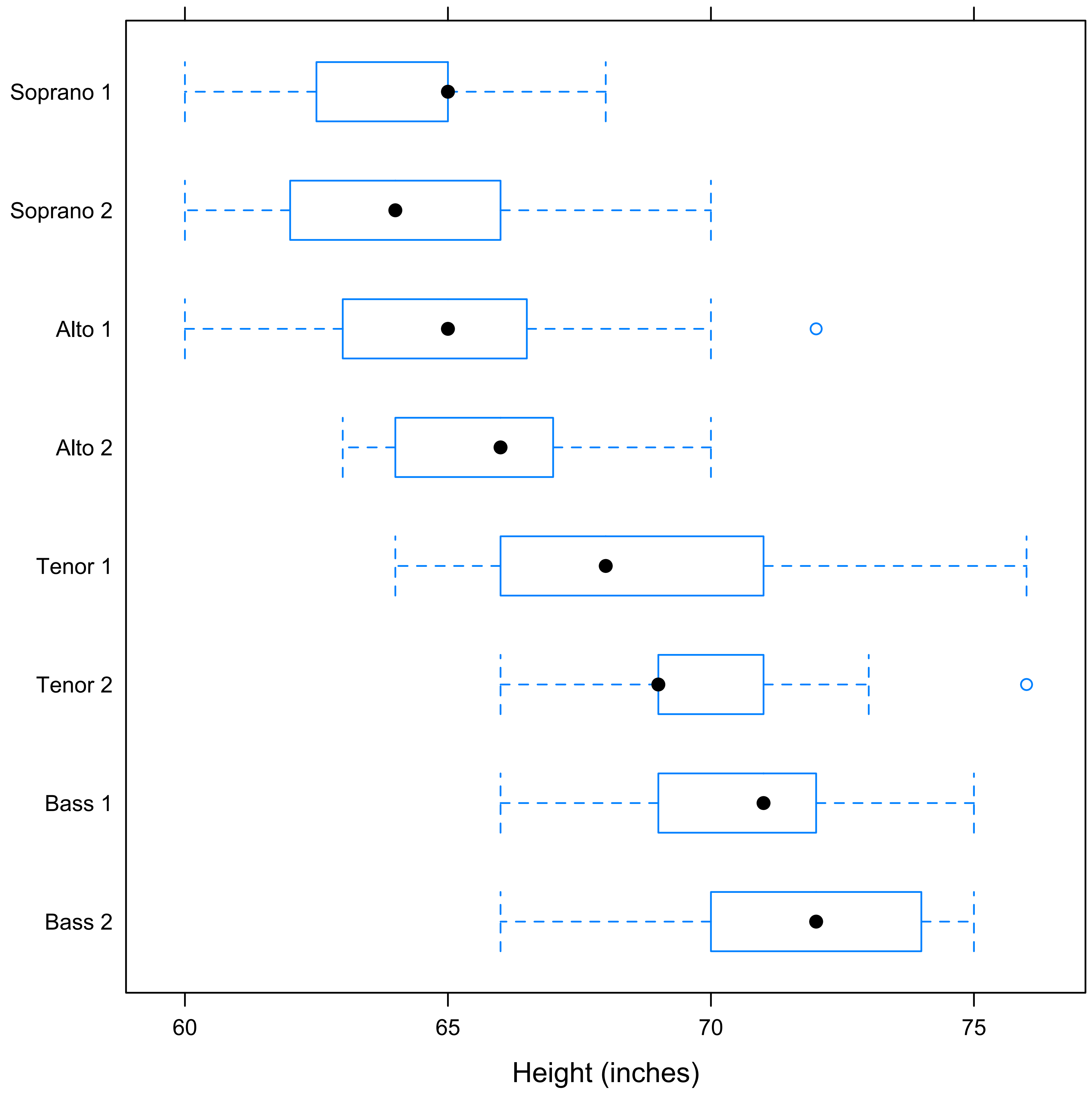


```
xyplot> ## Extended formula interface
xyplot>
xyplot> xyplot(Sepal.Length + Sepal.Width ~ Petal.Length + Petal.Width | Species,
xyplot+        data = iris, scales = "free", layout = c(2, 2),
xyplot+        auto.key = list(x = .6, y = .7, corner = c(0, 0)))
Hit <Return> to see next plot:

xyplot> ## user defined panel functions
xyplot>
xyplot> states <- data.frame(state.x77,
xyplot+                      state.name = dimnames(state.x77)[[1]],
xyplot+                      state.region = state.region)

xyplot> xyplot(Murder ~ Population | state.region, data = states,
xyplot+        groups = state.name,
xyplot+        panel = function(x, y, subscripts, groups) {
xyplot+            ltext(x = x, y = y, labels = groups[subscripts], cex=1,
xyplot+                  fontfamily = "HersheySans")
xyplot+        })
```

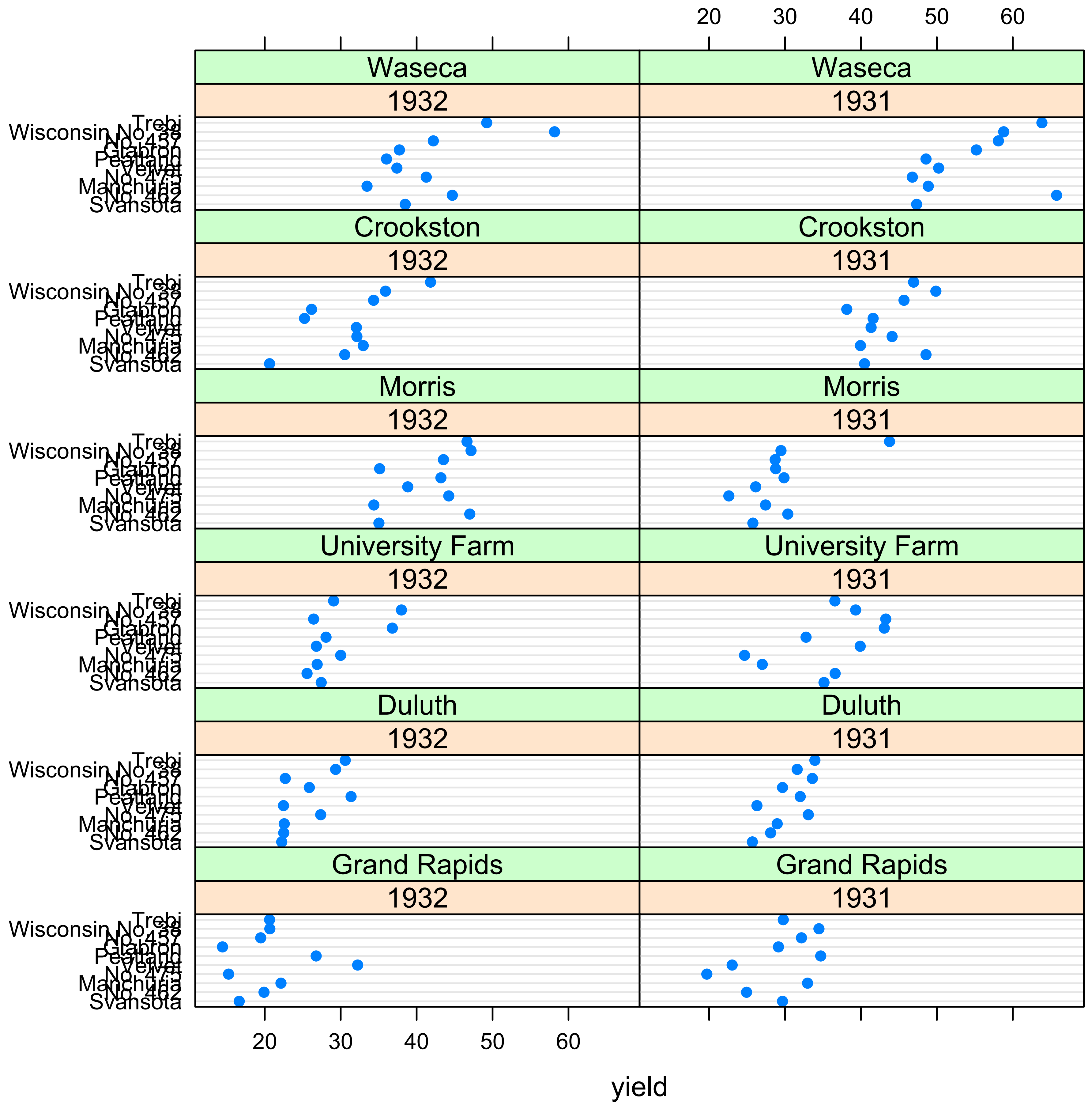


R code on next slide

```
xyplot> ## Extended formula interface
xyplot>
xyplot> xyplot(Sepal.Length + Sepal.Width ~ Petal.Length + Petal.Width | Species,
xyplot+        data = iris, scales = "free", layout = c(2, 2),
xyplot+        auto.key = list(x = .6, y = .7, corner = c(0, 0)))
Hit <Return> to see next plot:

xyplot> ## user defined panel functions
xyplot>
xyplot> states <- data.frame(state.x77,
xyplot+                      state.name = dimnames(state.x77)[[1]],
xyplot+                      state.region = state.region)

xyplot> xyplot(Murder ~ Population | state.region, data = states,
xyplot+        groups = state.name,
xyplot+        panel = function(x, y, subscripts, groups) {
xyplot+            ltext(x = x, y = y, labels = groups[subscripts], cex=1,
xyplot+                  fontfamily = "HersheySans")
xyplot+        })
```

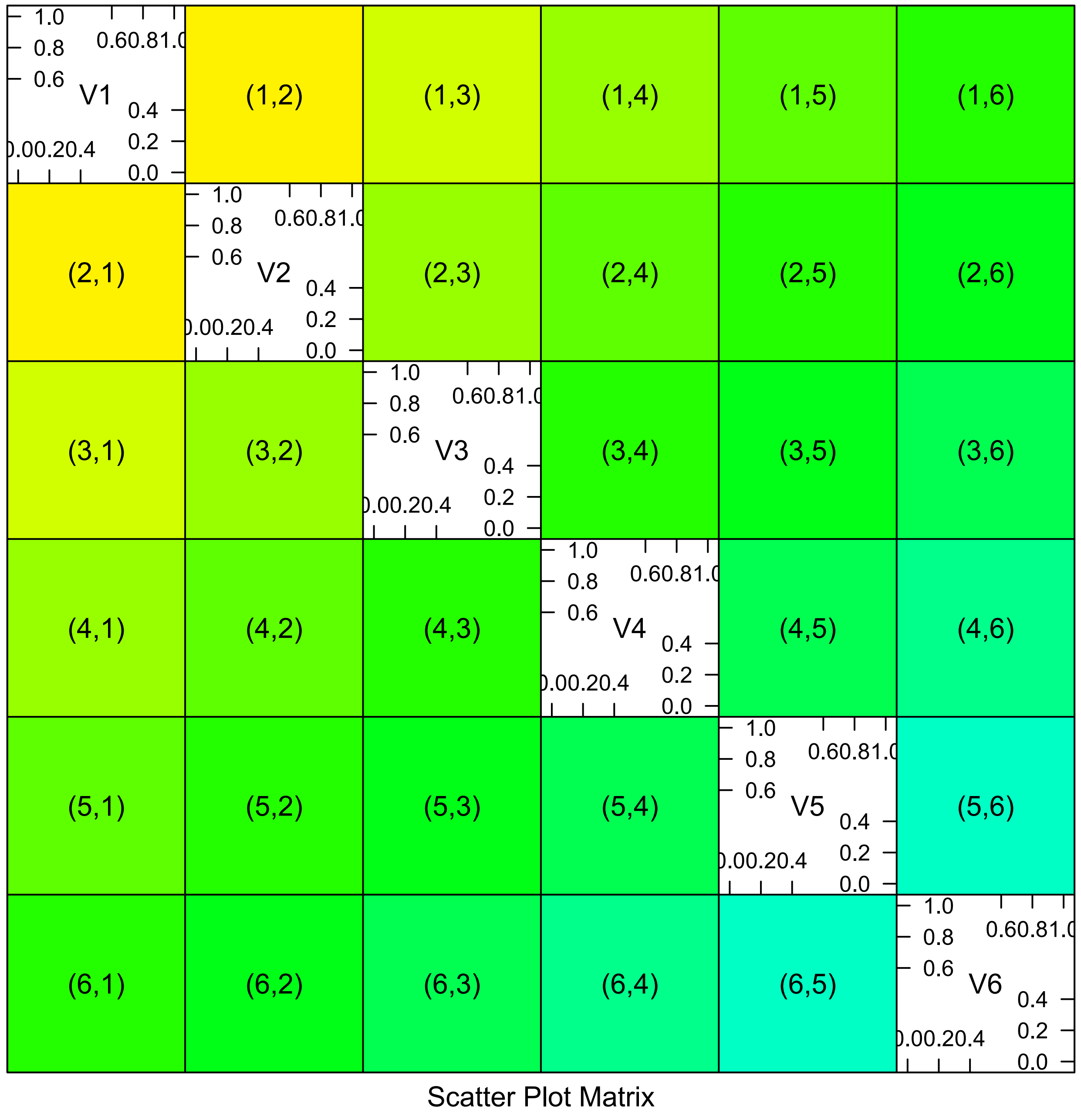

Soprano 1
Soprano 2
Alto 1
Alto 2
Tenor 1
Tenor 2
Bass 1
Bass 2
60
65
70
75
Height (inches)

```
xyplot> bwplot(voice.part ~ height, data=singer, xlab="Height (inches)")
```


Waseca
1932
1931
Crookston
Morris
University Farm
Duluth
Grand Rapids
Trebi
Wisconsin No. 38
No. 457
Glabron
Peatland
Velvet
No. 475
Manchuria
No. 462
Svansota
20
30
40
50
60
yield

```
xyplot> ## Grouped dot plot showing anomaly at Morris
xyplot>
xyplot> dotplot(variety ~ yield | site, data = barley, groups = year,
xyplot+         key = simpleKey(levels(barley$year), space = "right"),
xyplot+         xlab = "Barley Yield (bushels/acre) ",
xyplot+         aspect=0.5, layout = c(1,6), ylab=NULL)
```

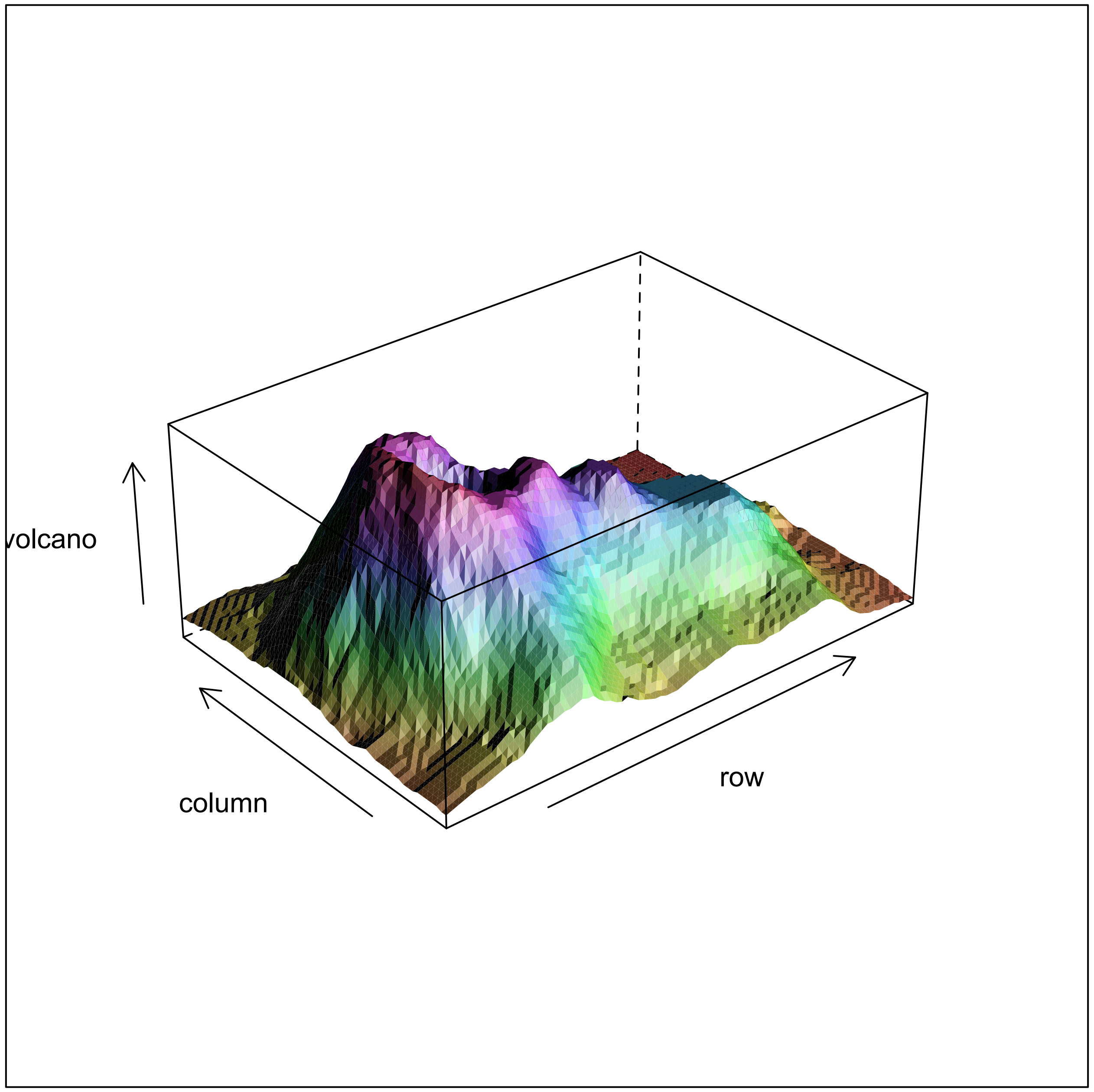
lattice::panel.pairs



R code

```
> example(panel.pairs)

pnl.pr> Cmat <- outer(1:6,1:6,
pnl.pr+              function(i,j) rainbow(11, start=.12, end=.5)[i+j-1])

pnl.pr> splom(~diag(6), as.matrix = TRUE,
pnl.pr+       panel = function(x, y, i, j, ...) {
pnl.pr+           panel.fill(Cmat[i,j])
pnl.pr+           panel.text(.5,.5, paste("(",i,",",j,")",sep=""))
pnl.pr+       })
Hit <Return> to see next plot:
```

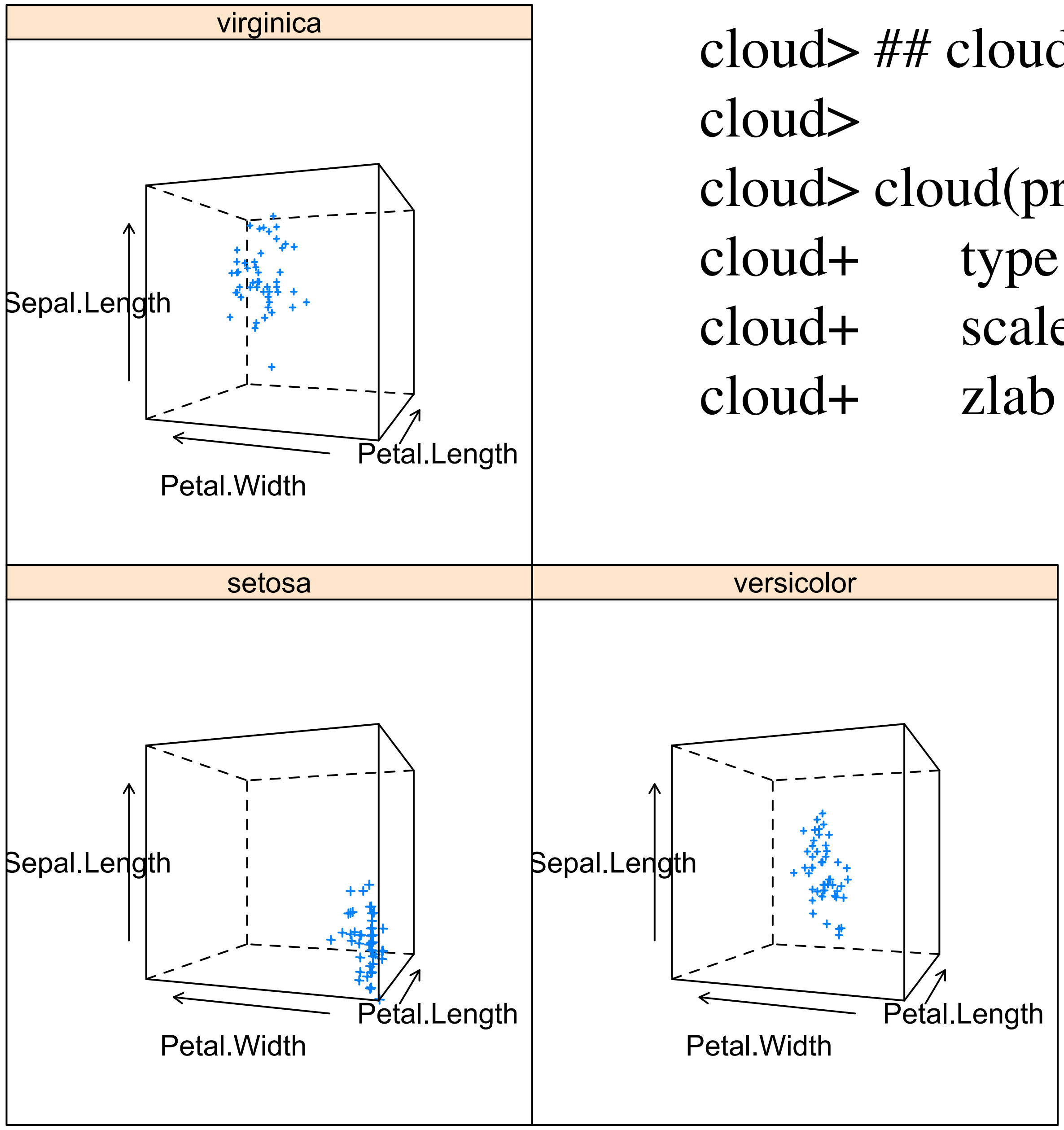
lattice::cloud



R code

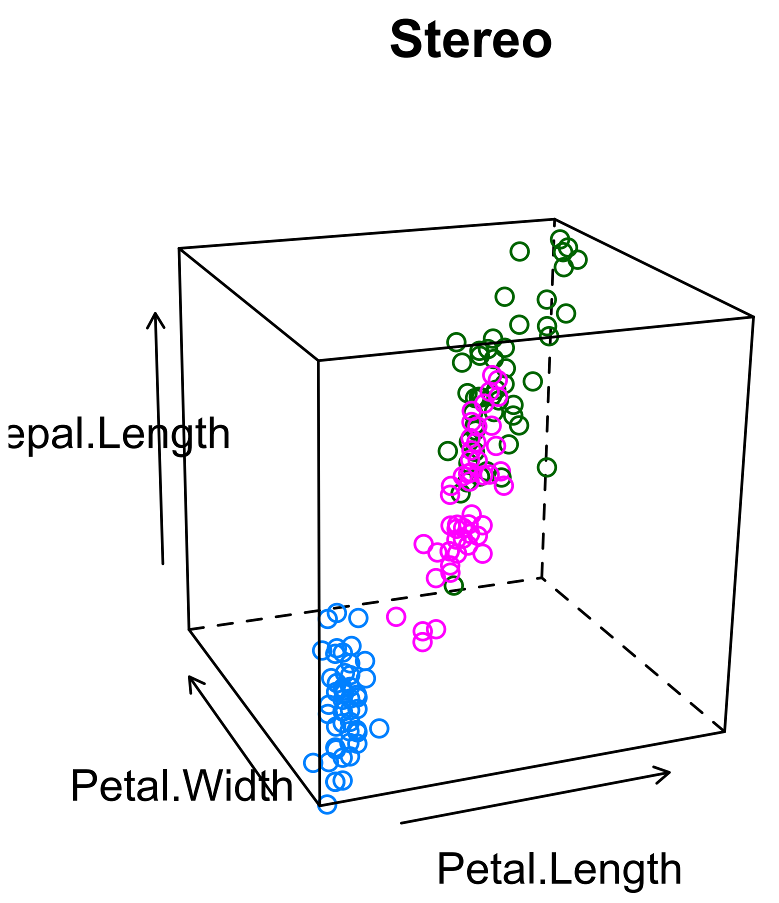
```
cloud> ## volcano  ## 87 x 61 matrix
cloud> wireframe(volcano, shade = TRUE,
cloud+           aspect = c(61/87, 0.4),
cloud+           light.source = c(10,0,10))
```

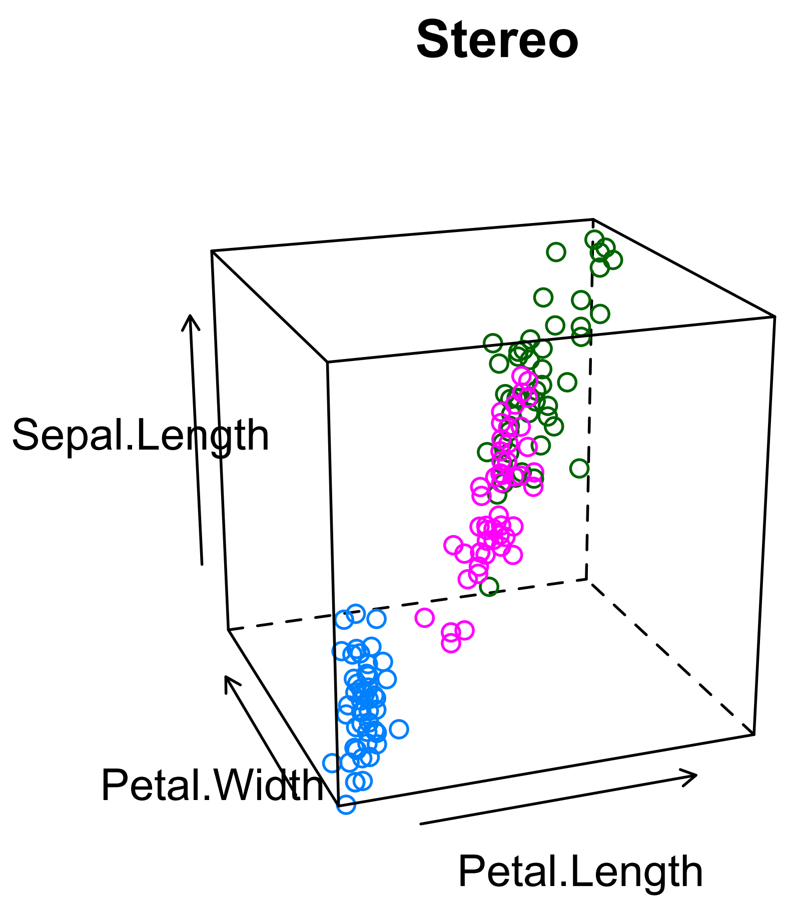
lattice::cloud



```
cloud> ## cloud.table
cloud>
cloud> cloud(prop.table(Titanic, margin = 1:3),
cloud+      type = c("p", "h"), strip = strip.custom(strip.names = TRUE
cloud+      scales = list(arrows = FALSE, distance = 2), panel.aspect =
cloud+      zlab = "Proportion")[, 1]
```

lattice::cloud





clustering packages

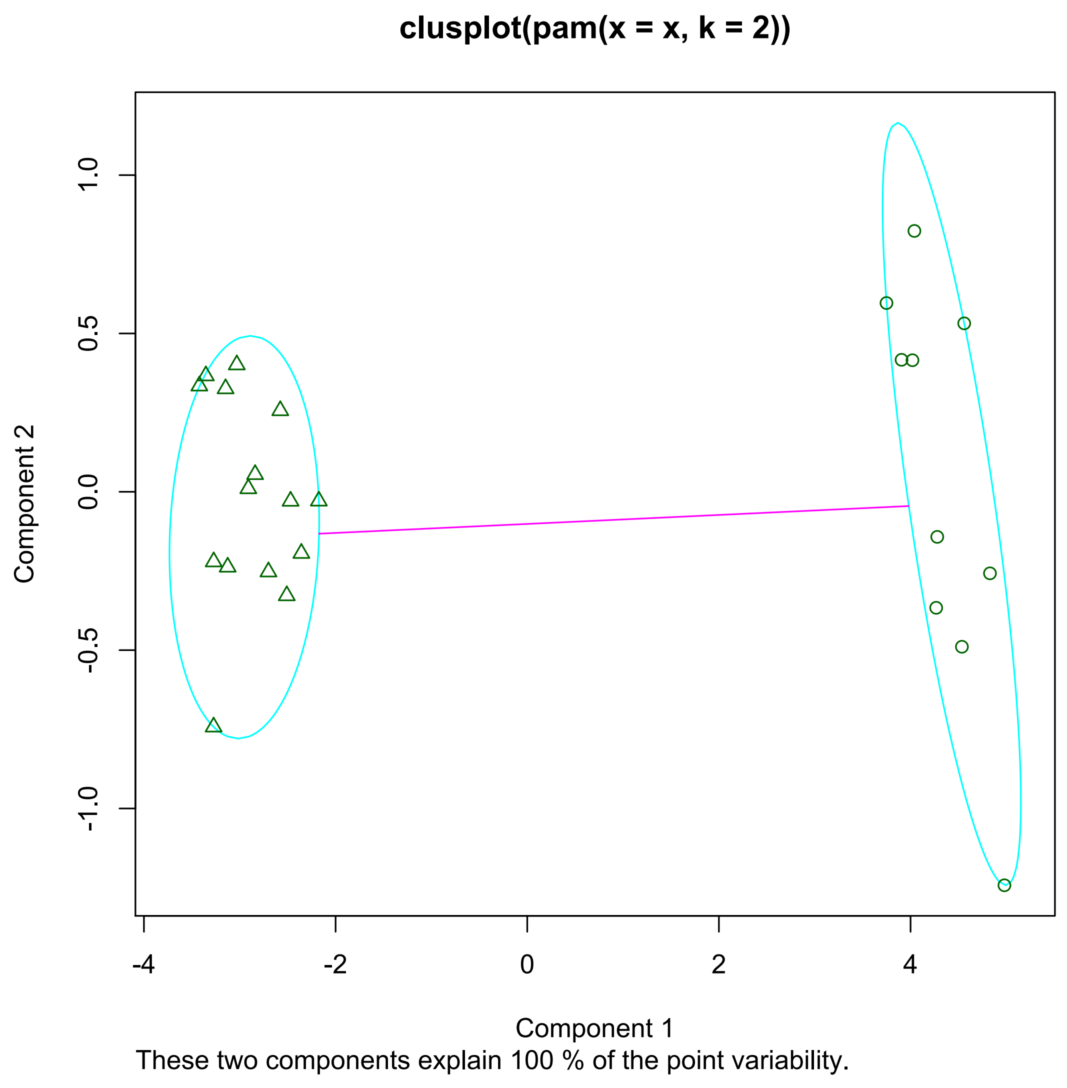
- Some packages with graphics related to clustering:
  - cluster
  - maptree
  - many more (I only considered those that appeared to have graphics)

help(package=cluster)

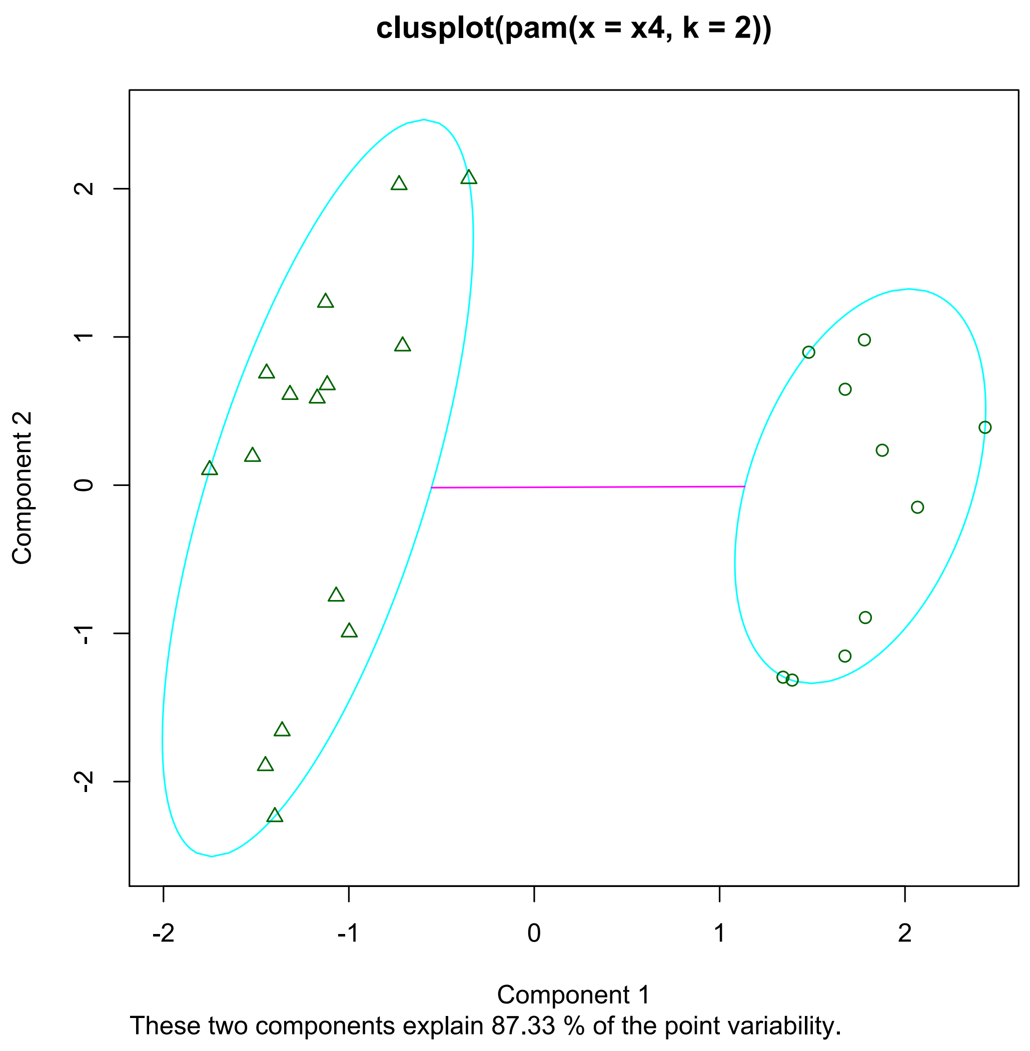
```
clusplot              Cluster Plot - Generic Function
clusplot.default        Bivariate Cluster Plot (Clusplot) Default Method
clusplot.partition     Bivariate Clusplot of a Partitioning Object
coef.hclust            Agglomerative Coefficient for 'hclust' Objects
pltree               Clustering Trees - Generic Function
pltree.twins           Clustering Tree of a Hierarchical Clustering

bannerplot             Plot Banner (of Hierarchical Clustering)
silhouette             Compute or Extract Silhouette Information from
                     Clustering
ellipsoidhull          Compute the Ellipsoid Hull or Spanning Ellipsoid
                     of a Point Set
predict.ellipsoid       Predict Method for Ellipsoid Objects
volume.ellipsoid        Compute the Volume of Planar Object
lower.to.upper.tri.inds Permute Indices for Triangular Matrices

plot.agnes             Plots of an Agglomerative Hierarchical Clustering
plot.diana             Plots of a Divisive Hierarchical Clustering
plot.mona              Banner of Monothetic Divisive Hierarchical Clusterings
plot.partition         Plot of a Partition of the Data Set
print.dissimilarity     Print and Summary Methods for Dissimilarity Objects
print.agnes            Print Method for AGNES Objects
print.clara            Print Method for CLARA Objects
print.diana            Print Method for DIANA Objects
print.fanny            Print Method for FANNY Objects
print.mona             Print Method for MONA Objects
print.pam              Print Method for PAM Objects
summary.agnes          Summary Method for 'agnes' Objects
summary.clara          Summary Method for 'clara' Objects
summary.diana          Summary Method for 'diana' Objects
summary.fanny          Summary Method for 'fanny' Objects
summary.mona           Summary Method for 'mona'  Objects
```

example(clusplot)

Identify clusters in datasets

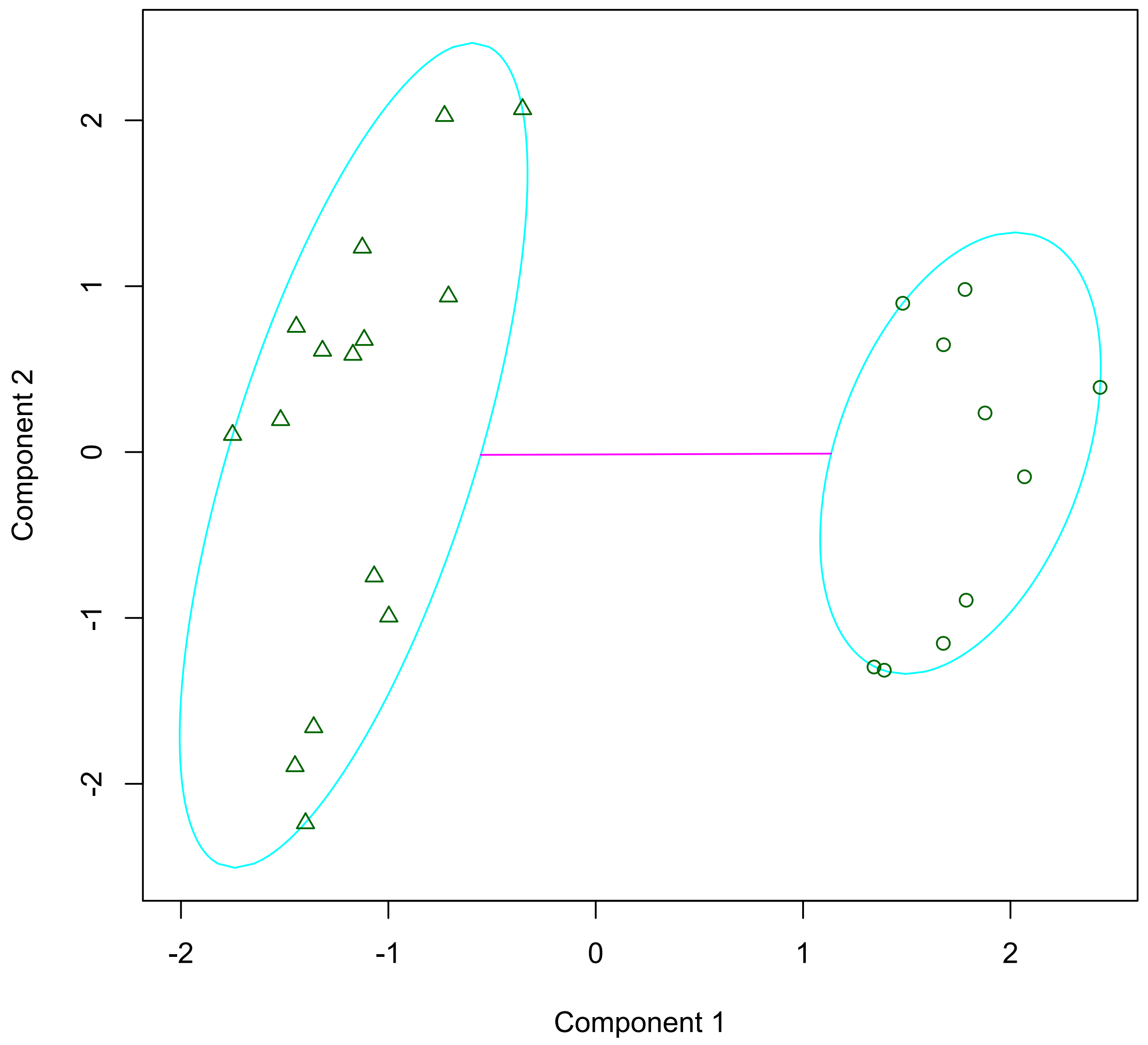


These two components explain 100 % of the point variability.

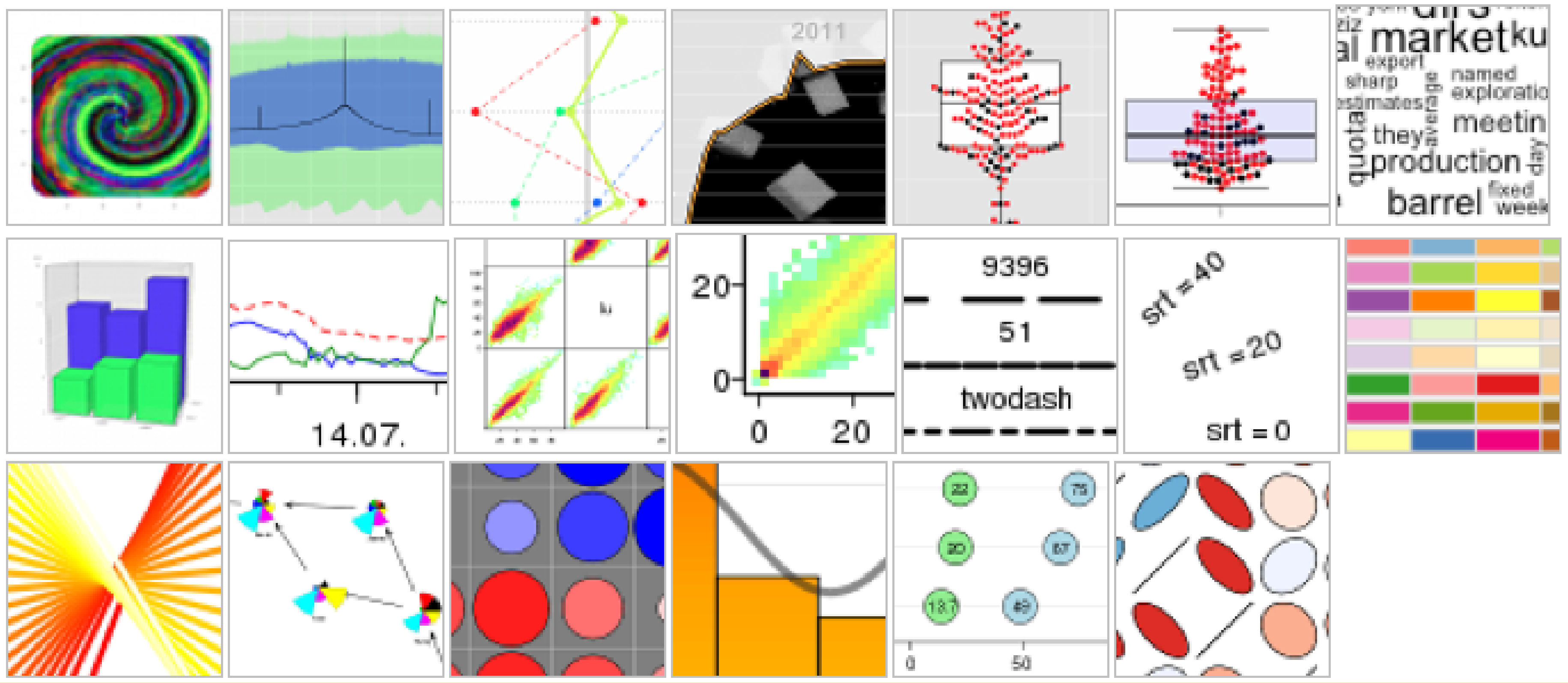


These two components explain 87.33 % of the point variability.

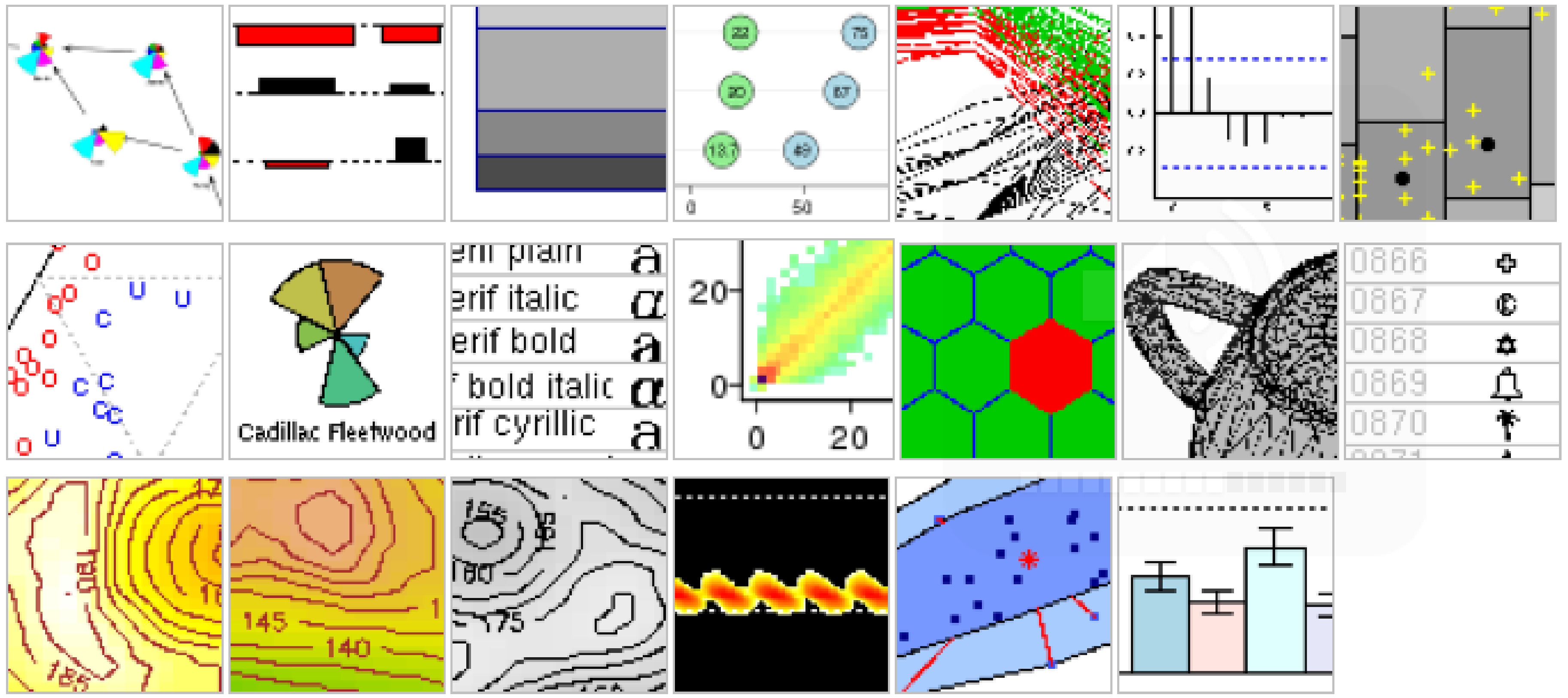
clusplot(pam(x = x4, k = 2))



These two components explain 87.33 % of the point variability.



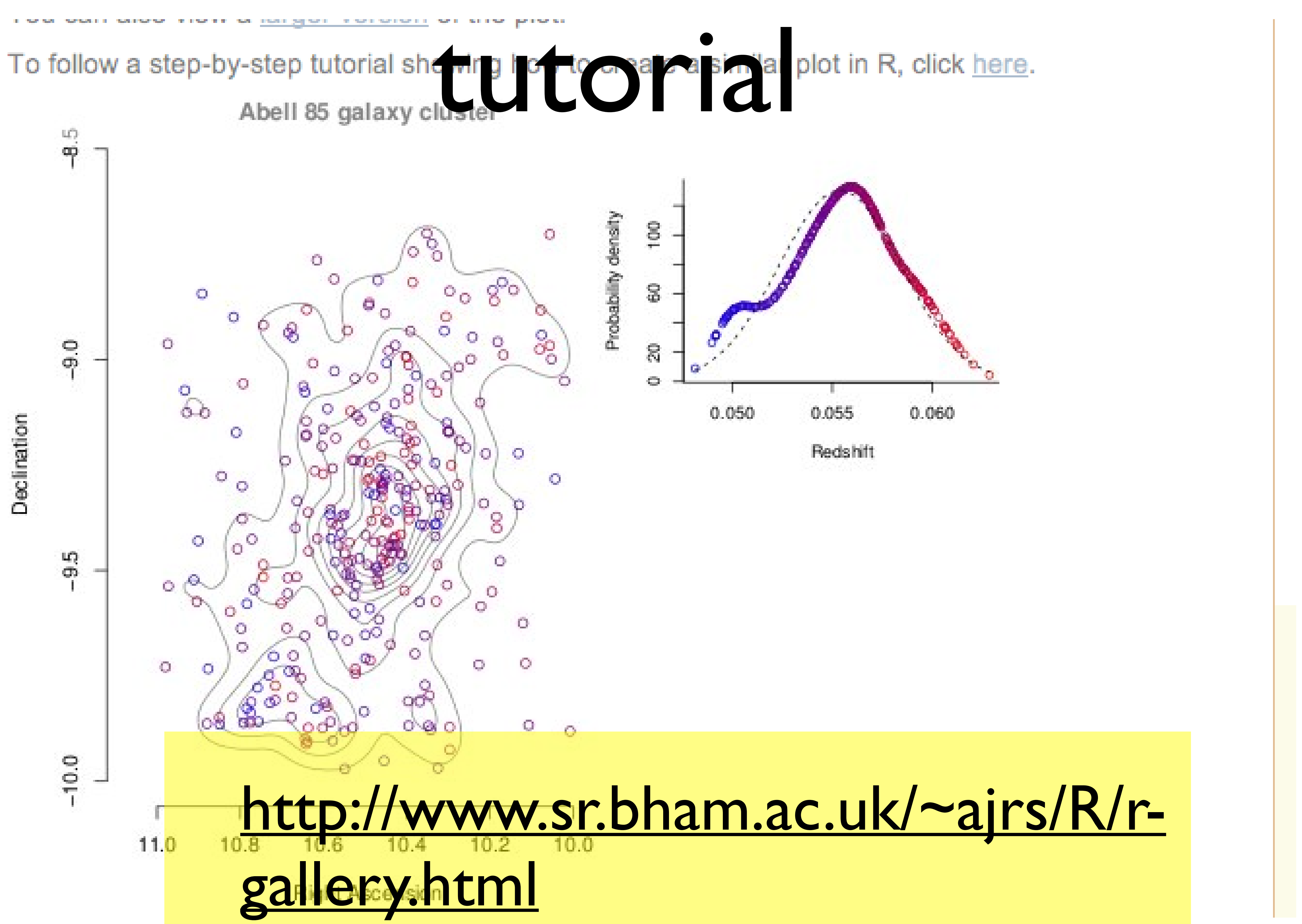
http://addictedtor.free.fr/graphiques/



Graphic Gallery

- With many examples, source code
- http://research.stowers-institute.org/efg/R/

Another Gallery with R tutorial


You can also view a larger version of the plot.

To follow a step-by-step tutorial showing how to create a similar plot in R, click here.

http://www.sr.bham.ac.uk/~ajrs/R/r-gallery.html

Graphics Gallery

- http://www.r-bloggers.com/browse-r-graphics-with-the-r-graph-gallery-and-the-r-graphical-manual/

Wikipedia

- http://en.wikibooks.org/wiki/R_Programming/Graphics

Scatterplots

- http://www.statmethods.net/graphs/scatterplot.html

Interactive Plots

- http://www.rosuda.org/iplots/

R graphics (online book)

- http://www.stat.auckland.ac.nz/~paul/RGraphics/rgraphics.html

R graphics

- http://homepage.univie.ac.at/harald.schilly/R_doku/graphics.html

Graphic Examples

- http://www.stat.auckland.ac.nz/~paul/RGraphics/rgraphics.html

R graphics manuals

- http://bg9.imslab.co.jp/Rhelp/
- thousands of pictures. Link not working.

Next lesson(s)

- Summary of course
- Solicit feedback for class improvement
- Have a great summer!