Spatial analysis in R for Marine Scientists

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Selected online databases

European Marine Observation and Data Network (EMODnet):
http://www.emodnet.eu/
Copernicus Marine Environment Monitoring Service (CMEMS):
http://marine.copernicus.eu/
GlobColour : http://www.globcolour.info/
General Bathymetric Chart of the Oceans (GEBCO):
http://www.gebco.net/
Integrated Climate Data Center (UHH):
http://icdc.zmaw.de/projekte/easy-init/easy-init-ocean.html

Extract and overlay information

Visually overlay two layers :

```
plot(raster)
plot(vector, add=TRUE)
```

Between vector and raster:

```
extract(raster, vector)
```

`fun`: if specified, aggregate the values per object

Between two vectors (points and polygons):

```
over(points, polygons)
```

NetCDF and multidimensional data

If **one** NetCDF file, create a **brick** object :

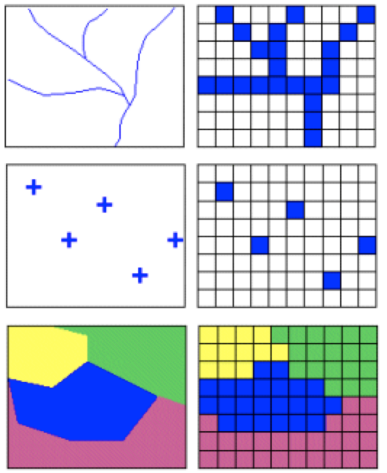
```
dir <- "folder/raster.nc"
brc <- brick(dir, varname="", lvar= )
```

If **multiple** NetCDF files, create a **stack** object :

```
dir.list<-list.files("folder/", pattern=".nc$")
stc <- stack(dir.list, varname="")
```

Loading vector and raster data in R

Vectors are made of points, lines or polygons. It is the natural format for the description of sites, either defined as areas (polygons) or points taken by a GPS. Vectors conserve the shape and spatial accuracy.



Raster is a matrix, i.e. a regular grid containing information for all the pixels uniformly distanced. The distance between the pixel is called the resolution. It is the natural format for remote sensing, model and interpolated dataset.

Load :

Shapefile (.shp) that can be read in R with the `rgdal` package:

```
dir <- "folder/shape.shp"
name <- " shape"
shp <- readOGR(dir,name)
```

Load:

Most of the formats (.tif, .ascii, .bil, .nc) can be open with the `raster` package :

```
dir <- "folder/raster.tif"
rst <- raster(dir)
```

Visualize:

```
plot(...)
```

`col`: define the color
`pch`: define the symbol
`cex`: size of the symbol

`xlim`, `ylim`: limit the extent

`col`: define the palette
`breaks`: manually define the color breaks
`maxpixels`: number of pixel to be plotted

Access information

```
proj4string()
bbox()
dim()
```

Cheatsheet created by R. Frelat, 10.07.2017

base graphics

SET GRAPHICAL PARAMETERS

*the following can **only** be set with* **par()**

par (…)

multiple plots	`mfcol = c(nrow,ncol)` `mfrow = c(nrow,ncol)`	*plot margins (outer)*	`oma = c(bottom, left, top, right)` default: `c(0, 0, 0, 0)` lines
plot margins	`mar = c(bottom, left, top, right)` default: c(5.1, 4.1, 4.1, 2.1) lines	*query x & y limits*	`par ("usr")`

CREATE A NEW PLOT

Bar charts	**barplot**(*height, …*)	**Histograms**	**hist**(*x, …*)
bar labels	`names.arg =`	*breakpts*	`breaks =`
border	`border =`		
fill color	`col =`	**Line charts**	**plot**(*x,* `type = "l"`)
horizontal	`horiz = TRUE`	*line type*	`lty =` "blank"|0 "solid"|1 "dashed"|2 "dotted"|3
Box plots	**boxplot**(*x, …*)	*line width*	`lwd =`
horizontal	`horizontal = TRUE`		
box labels	`names =`	**Scatterplots**	**plot**(*x, …*)
Dot plots	**dotchart**(*x, …*)	*symbol*	`pch =`
dot labels	`labels =`		

REMOVE

axis labels	`ann = FALSE`
axis, tickmarks, and labels	`xaxt = "n"` `yaxt = "n"`
plot box	`bty = "n"`

NOTE: Many of the parameters here can be also be set in `par()`*. See R help for more options.*

ADJUST

allow plotting out of plot region	`xpd = TRUE`
aspect ratio	`asp =`
axis limits	`xlim =, ylim =`
axis lines to match axis limits	`xaxs = "i"` , `yaxs = "i"` *(internal axis calculation)*

ADD TEXT

location

axis labels	`xlab =, ylab =`
subtitle	`sub =`
title	`main =`

style

font face	`font = 1` *(plain)* 2 *(bold)* 3 *(italic)* 4 *(bold italic)*
font family	`family = "serif"` "sans" "mono"

size

(magnification factor)

all elements	`cex =`
axis labels	`cex.lab =`
subtitle	`cex.sub =`
tick mark labels	`cex.axis =`
title	`cex.main =`

position

text direction	`las = 1` (horizontal)
justification	`adj = 0 .5 1` (left, center, right)

ADD TO AN EXISTING PLOT

Add new plot [***any plot function***] (…, `add = TRUE`)

ex. `barplot(x, add = TRUE)`

Axes	**axis** (*side,…)*
location	`side = 1 2 3 4` (bottom, left, top, right)
tick mark:	
labels	`labels =`
location	`at =`
remove	`tick = FALSE`
rotate text	`las = 1` (horizontal)
Axis labels	**mtext** (*text,…)*
location	`side = 1 2 3 4` (bottom, left, top, right)
lines to skip	`line =` *(from plot region, default = 0)*
position	`at =` *x or y-coord* (depending on side)
justification	`adj = 0 .5 1` (left, center, right)

Lines	**lines** (*x,…*)
line style	`lty =`
line width	`lwd =`
color	`col =`
Points	**points** (*x,…*)
symbol	`pch =`

color	`col =`
fill color	`bg =` `(pch: 21-25 only)`
Text	**text** (*x, y, text,…*)
position (rel. to x,y)	`pos = 1 2 3 4` (below, left, above, right) (default=center)
Title	**title** (*main,…*)
axis labels	`xlab =, ylab =`
subtitle	`sub =`
title	`main =`

Joyce Robbins, joycerobbins1@gmail.com

Source: http://publish.illinois.edu/johnrgallagher/files/2015/10/BaseGraphicsCheatsheet.pdf