

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

Map the two layers overlaid:

```
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 :

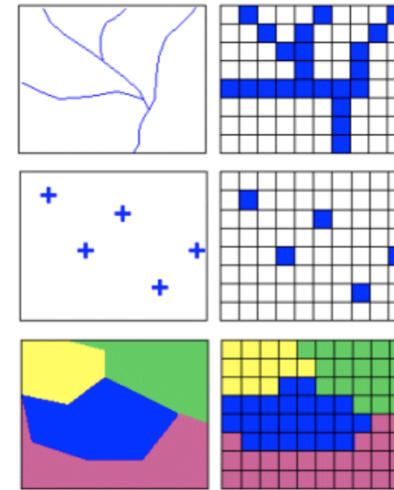
```
dir <- "folder/raster.nc"
brc <- brick(dir, varname="", lvar= )
```

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.



Load :

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

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

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:

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

Get attribute table:

```
@data
```

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

See the resolution:

```
res()
```

base graphics

SET GRAPHICAL PARAMETERS

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

`par(...)`

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

CREATE A NEW PLOT

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

REMOVE

<i>axis labels</i>	<code>ann = FALSE</code>
<i>axis, tickmarks, and labels</i>	<code>xaxt = "n"</code> <code>yaxt = "n"</code>
<i>plot box</i>	<code>bty = "n"</code>

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

ADJUST

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

ADD TEXT

location

<i>axis labels</i>	<code>xlab =, ylab =</code>
<i>subtitle</i>	<code>sub =</code>
<i>title</i>	<code>main =</code>

style

<i>font face</i>	<code>font = 1 (plain)</code> <code>2 (bold) 3 (italic)</code> <code>4 (bold italic)</code>
------------------	---

<i>font family</i>	<code>family = "serif"</code> "sans" "mono"
--------------------	--

size

(magnification factor)

<i>all elements</i>	<code>cex =</code>
<i>axis labels</i>	<code>cex.lab =</code>
<i>subtitle</i>	<code>cex.sub =</code>
<i>tick mark labels</i>	<code>cex.axis =</code>
<i>title</i>	<code>cex.main =</code>

position

<i>text direction</i>	<code>las = 1 (horizontal)</code>
<i>justification</i>	<code>adj = 0 .5 1</code> (left, center, right)

ADD TO AN EXISTING PLOT

Add new plot [*any plot function*]
(..., `add = TRUE`)
ex. `barplot(x, add = TRUE)`

Axes
location `axis(side, ...)`
`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
location `mtext(text, ...)`
`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 =`

 0 1 2 3 4 5 6 7 8 9 10 11 12
 13 14 15 16 17 18 19 20 21 22 23 24 25
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