

# Padrões de diversidade e extração de variáveis marinhas

*Juan P. Quimbayo, Thiago Mendes, Murilo S. Dias, Mariana Bender, Augusto Flores*

## Contents

<b>1</b>	<b>Obtendo dados e criando mapas</b>	<b>2</b>
1.1	Instalando pacotes . . . . .	2
1.2	Dados de peixes recifais . . . . .	2
1.2.1	Onde encontrar os dados . . . . .	2
1.2.2	Preparando para baixar o arquivo . . . . .	3
1.2.3	Distribuição das espécies peixes borboletas . . . . .	3
<b>2</b>	<b>Riqueza de espécies na América</b>	<b>6</b>
<b>3</b>	<b>Preditores climáticos globais</b>	<b>10</b>
3.1	Baixando, carregando e conferindo os dados marinhos . . . . .	10
<b>4</b>	<b>Checando projeções de dados biológicos e climáticos</b>	<b>19</b>
<b>5</b>	<b>Extraindo médias climáticas por pixel</b>	<b>22</b>
<b>6</b>	<b>Calculando relações simples entre riqueza e os preditores</b>	<b>23</b>
6.1	Calculando a riqueza de espécies . . . . .	23
	<b>Bibliography</b>	<b>25</b>

# 1 Obtendo dados e criando mapas

## 1.1 Instalando pacotes

Para realizarmos esta pratica, você precisará instalar alguns pacotes. Utilize o código abaixo para instalar os pacotes *devtools*, *maptools* e *letsR*. Pode ser que demore um pouco, mas se você tiver uma conexão internet o R vai baixar e instalar todo o necessário. Copie e cole os seguintes códigos em teu script ou terminal

```
# instalando pacotes
install.packages("devtools")
install.packages("rgdal")
install.packages("maptools")
install.packages("maps")
library(devtools)
install_github("macroecology/letsR", dependencies = T)
# install.packages('letsR')#o pacote do repositório Git é o
# mais atual
```

```
# carregando pacotes usados
library(letsR)
```

```
## Loading required package: raster
```

```
## Loading required package: sp
```

```
## Loading required package: maps
```

```
library(maptools)
```

```
## Checking rgeos availability: TRUE
```

```
library(maps)
```

```
library(rgdal)
```

```
## rgdal: version: 1.4-4, (SVN revision 833)
```

```
## Geospatial Data Abstraction Library extensions to R successfully loaded
```

```
## Loaded GDAL runtime: GDAL 2.2.2, released 2017/09/15
```

```
## Path to GDAL shared files: /usr/share/gdal/2.2
```

```
## GDAL binary built with GEOS: TRUE
```

```
## Loaded PROJ.4 runtime: Rel. 4.9.2, 08 September 2015, [PJ_VERSION: 492]
```

```
## Path to PROJ.4 shared files: (autodetected)
```

```
## Linking to sp version: 1.3-1
```

## 1.2 Dados de peixes recifais

### 1.2.1 Onde encontrar os dados

Muitos são os repositórios de dados espaciais. Dados geográficos para os organismos mais comumente acessados (e.g., *mamíferos*, *anuros*, etc) estão disponíveis no seguinte link: [\\*http://www.iucnredlist.org/technical-documents/spatial-data\\*](http://www.iucnredlist.org/technical-documents/spatial-data). Verifique as possibilidades disponíveis.

Nós utilizaremos aqui os dados de *peixes borboletas* (*Chaetodontidae*) do Novo Mundo pois são poucas as espécies e os dados são leves o suficiente para uma aula prática rápida. Se quiser testar algo mais real, tente baixar os dados de *mamíferos* (cerca de 350Mb!) em casa e utilizá-lo no lugar dos peixes Mas antes, tenha certeza que teu computador é potente o suficiente para tratar tamanho volume de dados, pois você poderá perder **algumas horas (ou até dias)** para executar os exercícios descritos abaixo.

### 1.2.2 Preparando para baixar o arquivo

Inicialmente, precisamos criar uma pasta no diretório de trabalho que receberá o arquivo depois de terminado o download. Para criar a pasta, fazer o download do arquivo, descomprimir e finalmente carregá-lo no R será preciso executar os seguintes códigos.

```
# criar pasta
dir.create(file.path(getwd(), "/data"))
```

### 1.2.3 Distribuição das espécies peixes borboletas

Para importar os dados e ver o que tem dentro deles, faça

```
# Importante o shapefile
borb <- readOGR(dsn = "data/Chaetodontidae_NewWorld.shp")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "/home/murilo/Dropbox/Disciplina UnB_2sem2019/6_Macroecologia e Biogeografia Marinh
## with 29 features
## It has 27 fields
## Integer64 fields read as strings: id_no
```

```
# gymno <-readOGR(dsn = 'data/GYMNOPHIONA.shp')
```

```
# visualizando o conteúdo dos dados
head(borb@data)
```

```
##      id_no      binomial presence origin seasonal compiler
## 0 165627 Prognathodes aculeatus      1      1          1      IUCN
## 1   4364      Chaetodon robustus      1      1          1      IUCN
## 2 165688 Chaetodon flavirostris      1      1          1      IUCN
## 3   4363 Prognathodes obliquus      1      1          1      IUCN
## 4 165646 Prognathodes guyanensis      1      1          1      IUCN
## 5 165621      Prognathodes aya      1      1          1      IUCN
##      yrcompiled      citation source
## 0      2010 International Union for Conservation of Nature (IUCN) <NA>
## 1      2010 International Union for Conservation of Nature (IUCN) <NA>
## 2      2010 International Union for Conservation of Nature (IUCN) <NA>
## 3      2010 International Union for Conservation of Nature (IUCN) <NA>
## 4      2010 International Union for Conservation of Nature (IUCN) <NA>
## 5      2010 International Union for Conservation of Nature (IUCN) <NA>
##      dist_comm island subspecies subpop      legend tax_comm kingdom
## 0      <NA>      <NA>      <NA>      <NA> Extant (resident)      <NA> ANIMALIA
```

```

## 1      <NA>    <NA>          <NA>    <NA> Extant (resident)    <NA> ANIMALIA
## 2      <NA>    <NA>          <NA>    <NA> Extant (resident)    <NA> ANIMALIA
## 3      <NA>    <NA>          <NA>    <NA> Extant (resident)    <NA> ANIMALIA
## 4      <NA>    <NA>          <NA>    <NA> Extant (resident)    <NA> ANIMALIA
## 5      <NA>    <NA>          <NA>    <NA> Extant (resident)    <NA> ANIMALIA
##      phylum      class      order_      family      genus      category
## 0 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Prognathodes      LC
## 1 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Chaetodon      LC
## 2 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Chaetodon      LC
## 3 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Prognathodes      DD
## 4 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Prognathodes      LC
## 5 CHORDATA ACTINOPTERYGII PERCIFORMES CHAETODONTIDAE Prognathodes      LC
##      marine terrestrial freshwater SHAPE_Leng SHAPE_Area
## 0   True      False      False      555.8      221.5
## 1   True      False      False      191.2      70.2
## 2   True      False      False      676.2      327.2
## 3   True      False      False      5.6      2.5
## 4   True      False      False      623.9      213.9
## 5   True      False      False      283.8      74.7

```

Este último comando te permitirá ver tudo todos os descritores (i.e., colunas) dentro do arquivo *.shp*.

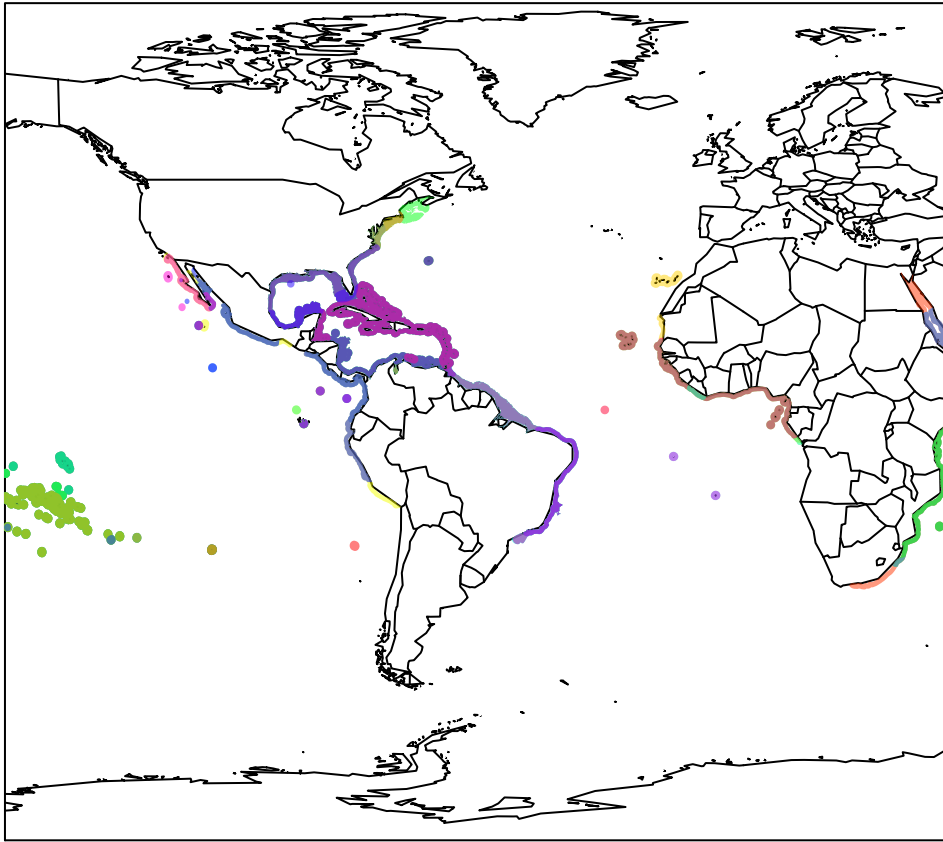
Agora precisaremos fazer um mapa para visualizar o que há dentro deste objeto *borb* que acabamos de criar. Utilizaremos algumas linhas de código a mais para fazer um mapa colorido para melhorar a compreensão dos dados. Utilizaremos a coluna referente ao nome científico de cada uma das espécies (i.e. *binomial*) para criar cores aleatórias e utilizá-las no nosso mapa. O parâmetro **alfa** dentro da função **rainbow** faz referência ao nível de transparência de cada cor; assim, poderemos ver polígonos mesmo que eles estejam cobertos por outros polígonos.

```

## usando cores aleatórias nos nossos mapas
colors <- rainbow(length(levels(borb@data$binomial)), alpha = 0.5)
position <- match(borb@data$binomial, levels(borb@data$binomial))
colors <- colors[position]

## Plot call
map("world", xlim = c(-150, 40))
box()
plot(borb, col = colors, lty = 0, add = T)

```



Cada polígono colorido se refere à área de distribuição de cada espécie.

## 2 Riqueza de espécies na América

A manipulação e tratamento de dados espaciais tipo *.shp* exigem um profundo conhecimento sobre os dados e a forma como eles são armazenados. Se quiserem saber mais sobre como utilizar o R para tratar este tipo de dados, recorram a livros como **Applied spatial data analysis with R** (Bivand et al. 2013) ou **An introduction to R for spatial analysis and mapping** (Brunsdon & Comber 2015). Felizmente, existe um pacote chamado **letsR** (Vilela & Villalobos 2015) que possui uma grande variedade de ferramentas úteis para macroecólogos. Nós já instalamos e carregamos este pacote no início da aula e podemos utilizar suas funções para fazer mais alguns mapas.

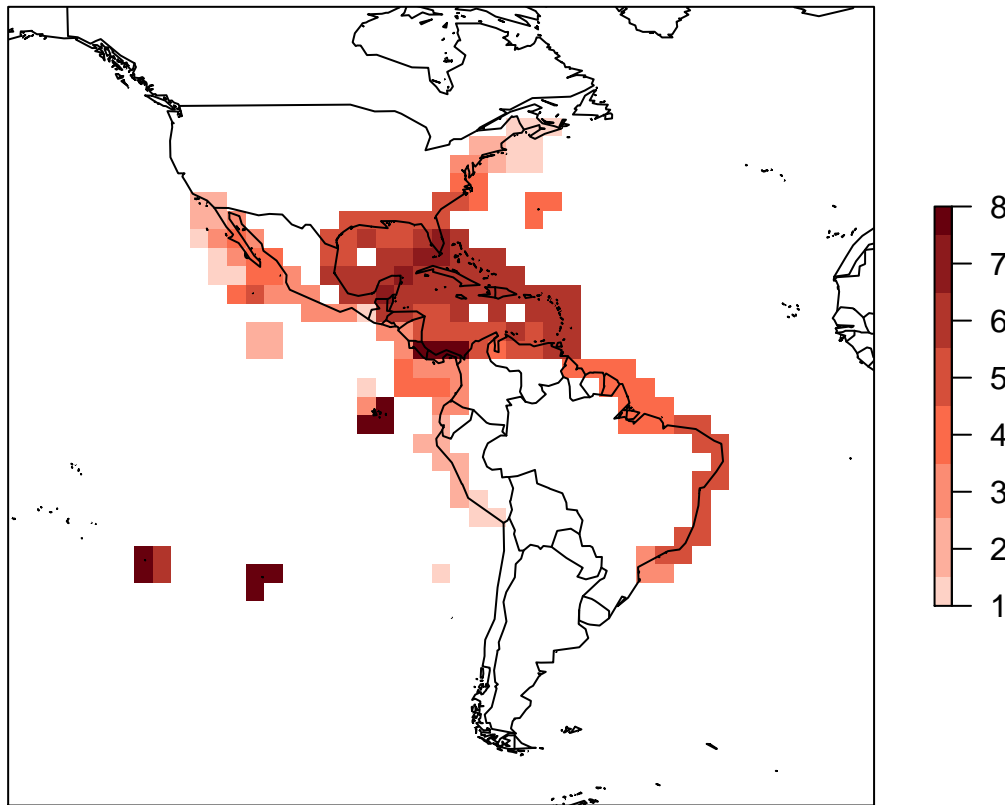
Vamos utilizar a função **lets.presab** para criar um raster (i.e., arquivo tipo imagem que armazena informações em cada um de seus pixels; digite ‘arquivo raster’ no Google para mais detalhes) resumindo a riqueza de espécie de peixes borboletas na América do Sul. Neste caso, a resolução do nosso arquivo de raster será de 3x3 graus (i.e., cada pixel ou célula tem tamanho 3 por 3 graus); mude a *resol* para valores 4 ou 5 e você verá a diferença na figura final. Já o argumento *cover* garante que somente pixels que tenham mais de 1% de sua área sobre do continente sejam mantidos (mude estes valores e veja a diferença no mapa resultante). Como nosso objetivo é um mapa da América, as coordenadas limites foram definidas para realizar o exercício somente ao longo da extensão desejada (argumentos *xmn*, *xmx*, *ymn* e *ymx* da função **lets.presab**).

```
borb_maps <- lets.presab(borb, resol = 3,
                        xmn = -130, xmx = -30, #América range
                        ymn = -65, ymx = 65, #América range
                        cover = 0.01)
```

```
## Warning in `proj4string<-`(`*tmp*`, value = <S4 object of class structure("CRS", package =
## +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## without reprojecting.
## For reprojection, use function spTransform
```

```
plot(borb_maps, axes=F, main="Riqueza de Peixes Borboletas")
```

## Riqueza de Peixes Borboletas



Aproveite e use o seguinte código para ver o que há no objeto criado pela função

```
# see what we have
```

```
summary(borb_maps)
```

```
##  
## Class: PresenceAbsence  
## _ _  
## Number of species: 23  
## Number of cells: 190  
## Cells with presence: 190  
## Cells without presence: 0  
## Species without presence: 0  
## Species with the largest range: Chaetodon ocellatus  
## _ _  
## Grid parameters  
## Resolution: 3, 3 (x, y)  
## Extention: -130, -31, -64, 65 (xmin, xmax, ymin, ymax)  
## Coord. Ref.: +proj=longlat +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0
```

```
str(borb_maps)
```

```
## List of 3  
## $ Presence_and_Absence_Matrix: num [1:190, 1:25] -68.5 -65.5 -62.5 -74.5 -71.5 -68.5 -65.  
## ..- attr(*, "dimnames")=List of 2  
## .. ..$ : NULL
```

```

## .. ..$ : chr [1:25] "Longitude(x)" "Latitude(y)" "Amphichaetodon melbae" "Chaetodon auri
## $ Richness_Raster :Formal class 'RasterLayer' [package "raster"] with 12 slots
## .. ..@ file :Formal class '.RasterFile' [package "raster"] with 13 slots
## .. .. . . .@ name : chr ""
## .. .. . . .@ datanotation: chr "FLT4S"
## .. .. . . .@ byteorder : chr "little"
## .. .. . . .@ nodatavalue : num -Inf
## .. .. . . .@ NChanged : logi FALSE
## .. .. . . .@ nbands : int 1
## .. .. . . .@ bandorder : chr "BIL"
## .. .. . . .@ offset : int 0
## .. .. . . .@ toptobottom : logi TRUE
## .. .. . . .@ blockrows : int 0
## .. .. . . .@ blockcols : int 0
## .. .. . . .@ driver : chr ""
## .. .. . . .@ open : logi FALSE
## .. ..@ data :Formal class '.SingleLayerData' [package "raster"] with 13 slots
## .. .. . . .@ values : num [1:1419] 0 0 0 0 0 0 0 0 0 0 ...
## .. .. . . .@ offset : num 0
## .. .. . . .@ gain : num 1
## .. .. . . .@ inmemory : logi TRUE
## .. .. . . .@ fromdisk : logi FALSE
## .. .. . . .@ isfactor : logi FALSE
## .. .. . . .@ attributes: list()
## .. .. . . .@ haveminmax: logi TRUE
## .. .. . . .@ min : num 0
## .. .. . . .@ max : num 8
## .. .. . . .@ band : int 1
## .. .. . . .@ unit : chr ""
## .. .. . . .@ names : chr ""
## .. ..@ legend :Formal class '.RasterLegend' [package "raster"] with 5 slots
## .. .. . . .@ type : chr(0)
## .. .. . . .@ values : logi(0)
## .. .. . . .@ color : logi(0)
## .. .. . . .@ names : logi(0)
## .. .. . . .@ colortable: logi(0)
## .. ..@ title : chr(0)
## .. ..@ extent :Formal class 'Extent' [package "raster"] with 4 slots
## .. .. . . .@ xmin: num -130
## .. .. . . .@ xmax: num -31
## .. .. . . .@ ymin: num -64
## .. .. . . .@ ymax: num 65
## .. ..@ rotated : logi FALSE
## .. ..@ rotation:Formal class '.Rotation' [package "raster"] with 2 slots
## .. .. . . .@ geotrans: num(0)
## .. .. . . .@ transfun:function ()
## .. ..@ ncols : int 33
## .. ..@ nrows : int 43
## .. ..@ crs :Formal class 'CRS' [package "sp"] with 1 slot

```



```
## .. .. .@ projargs: chr "+proj=longlat +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0"
## .. ..@ history : list()
## .. ..@ z : list()
## $ Species_name : chr [1:23] "Amphichaetodon melbae" "Chaetodon auriga" "Cha
## - attr(*, "class")= chr "PresenceAbsence"
```

E se precisarmos criar somente uma matriz para saber qual espécie ocorre em cada uma das células do raster, você pode acessar a matriz direto do objeto **borb\_maps** ou refazer a mesma função pedindo como resultado somente a matriz de presença/ausência (*show.matrix=TRUE* dentro da função *lets.presab*; digite *?lets.presab* no console do R e leia o help desta função e seus argumentos).

```
# Presence/absence matrix
borb_pa <- borb_maps$Presence_and_Absence_Matrix
```

Se quisermos calcular a área de distribuição de cada uma das espécies, é só utilizar a seguinte função. Porém, como ela retorna o valor em m<sup>2</sup>, precisamos converter em km<sup>2</sup> dividindo por 1000000 (1000 m x 1000 m).

```
# range size in m2
range_gym <- lets.rangesize(borb_maps, units = "squaremeter")

# transformar m2 para km2
range_gym <- range_gym/1e+06
range_gym
```

```
##
## Range_size
## Amphichaetodon melbae 3.9e+05
## Chaetodon auriga 7.3e+05
## Chaetodon capistratus 8.9e+06
## Chaetodon flavirostris 7.0e+05
## Chaetodon humeralis 5.5e+06
## Chaetodon litus 3.0e+05
## Chaetodon lunula 9.6e+05
## Chaetodon mertensii 5.0e+05
## Chaetodon meyeri 7.6e+05
## Chaetodon ocellatus 1.2e+07
## Chaetodon pelewensis 7.0e+05
## Chaetodon sedentarius 1.1e+07
## Chaetodon striatus 1.1e+07
## Chaetodon unimaculatus 1.0e+06
## Forcipiger flavissimus 4.6e+06
## Hemitaurichthys multispinosus 5.0e+05
## Johnrandallia nigrirostris 5.3e+06
## Prognathodes aculeatus 6.7e+06
## Prognathodes aya 3.2e+06
## Prognathodes brasiliensis 1.4e+06
## Prognathodes carlhubbsi 7.7e+05
## Prognathodes falcifer 1.8e+06
## Prognathodes guyanensis 7.0e+06
```

## 3 Preditores climáticos globais

### 3.1 Baixando, carregando e conferindo os dados marinhos

Agora que já conhecemos como a riqueza de espécies varia ao longo do espaço, precisamos tentar descobrir quais são os mecanismos por trás dos padrões espaciais. Para isso, vamos testar o efeito de variáveis climáticas na riqueza de espécies em questão por meio de modelos lineares simples. Antes disso, devemos recorrer a uma base de dados previamente estabelecida que nos forneça as variáveis climáticas de interesse. Para isso, você pode procurar no site do BioOracle<sup>1</sup> e encontrará muitos dos dados de que vai precisar. Você também precisará de uma conexão internet para baixar os dados do BioOracle (~ 11MB).

Vamos utilizar o pacote *sdmpredictors* para obter diretamente estes dados. Assim, nossas análises tanto espaciais (SIG quanto estatísticas) ficarão todas dentro do R. Isto facilita a manipulação das planilhas e a realização dos modelos. Vamos explorar algumas das variáveis marinhas disponíveis para o mundo. Execute as linhas abaixo e gaste um tempo para ler as informações e variáveis disponíveis.

```
# caso não tenha instalado e carregado o pacote  
# install.packages('sdmpredictors') Carregue o package  
library(sdmpredictors)
```

```
# Explore os dados do pacote  
list_datasets()
```

```
## dataset_code terrestrial marine url  
## 1 WorldClim TRUE FALSE http://www.worldclim.org/  
## 2 Bio-ORACLE FALSE TRUE http://bio-oracle.org/  
## 3 MARSPEC FALSE TRUE http://marspec.org/  
## 4 ENVIREM TRUE FALSE https://envirem.github.io/  
## 5 Freshwater TRUE FALSE https://www.earthenv.org/streams
```

```
##  
## 1  
## 2  
## 3 MARSPEC is a set of high resolution climatic and geophysical GIS data layers for the wor  
## 4  
## 5 The dataset consists of near-global  
##  
## 1 Hijmans, R.J., S.E. Cameron, J.L. Parra,  
## 2 Tyberghein L., Verbruggen H., Pauly K., Troupin C., Mineur F. & De Clerck O. Bio-ORACLE:  
## 3  
## 4 Title, P.O., Bemmels, J.B. 2017. ENVIREM: An expanded  
## 5 Domisch, S., Amatulli, G., and Jetz, W. (20
```

```
list_datasets(terrestrial = FALSE, marine = TRUE)
```

```
## dataset_code terrestrial marine url  
## 2 Bio-ORACLE FALSE TRUE http://bio-oracle.org/  
## 3 MARSPEC FALSE TRUE http://marspec.org/  
##  
## 2
```

---

<sup>1</sup><http://www.bio-oracle.org/>

```
## 3 MARSPEC is a set of high resolution climatic and geophysical GIS data layers for the wor
##
## 2 Tyberghein L., Verbruggen H., Pauly K., Troupin C., Mineur F. & De Clerck O. Bio-ORACLE:
## 3
```

```
# Vejam a quantidade de raster disponivel!
```

```
list_layers(c("Bio-ORACLE"))$layer_code
```

```
## [1] "B0_calcite" "B0_chlomap"
## [3] "B0_chlomean" "B0_chlomin"
## [5] "B0_chlorange" "B0_cloudmax"
## [7] "B0_cloudmean" "B0_cloudmin"
## [9] "B0_damax" "B0_damean"
## [11] "B0_damin" "B0_dissox"
## [13] "B0_nitrate" "B0_parmax"
## [15] "B0_parmean" "B0_ph"
## [17] "B0_phosphate" "B0_salinity"
## [19] "B0_silicate" "B0_sstmax"
## [21] "B0_sstmean" "B0_sstmin"
## [23] "B0_sstrange" "B0_bathymax"
## [25] "B0_bathymean"
## [27] "B02_chlomap_bdmax" "B02_chlomap_bdmean"
## [29] "B02_chlomap_bdmin" "B02_chlomean_bdmax"
## [31] "B02_chlomean_bdmean" "B02_chlomean_bdmin"
## [33] "B02_chlomin_bdmax" "B02_chlomin_bdmean"
## [35] "B02_chlomin_bdmin" "B02_chlorange_bdmax"
## [37] "B02_chlorange_bdmean" "B02_chlorange_bdmin"
## [39] "B02_chloltmax_bdmax" "B02_chloltmax_bdmean"
## [41] "B02_chloltmax_bdmin" "B02_chloltmin_bdmax"
## [43] "B02_chloltmin_bdmean" "B02_chloltmin_bdmin"
## [45] "B02_curvelmax_bdmax" "B02_curvelmax_bdmean"
## [47] "B02_curvelmax_bdmin" "B02_curvelmean_bdmax"
## [49] "B02_curvelmean_bdmean" "B02_curvelmean_bdmin"
## [51] "B02_curvelmin_bdmax" "B02_curvelmin_bdmean"
## [53] "B02_curvelmin_bdmin" "B02_curvelrange_bdmax"
## [55] "B02_curvelrange_bdmean" "B02_curvelrange_bdmin"
## [57] "B02_curvelltmax_bdmax" "B02_curvelltmax_bdmean"
## [59] "B02_curvelltmax_bdmin" "B02_curvelltmin_bdmax"
## [61] "B02_curvelltmin_bdmean" "B02_curvelltmin_bdmin"
## [63] "B02_dissoxmax_bdmax" "B02_dissoxmax_bdmean"
## [65] "B02_dissoxmax_bdmin" "B02_dissoxmean_bdmax"
## [67] "B02_dissoxmean_bdmean" "B02_dissoxmean_bdmin"
## [69] "B02_dissoxmin_bdmax" "B02_dissoxmin_bdmean"
## [71] "B02_dissoxmin_bdmin" "B02_dissoxrange_bdmax"
## [73] "B02_dissoxrange_bdmean" "B02_dissoxrange_bdmin"
## [75] "B02_dissoxltmax_bdmax" "B02_dissoxltmax_bdmean"
## [77] "B02_dissoxltmax_bdmin" "B02_dissoxltmin_bdmax"
## [79] "B02_dissoxltmin_bdmean" "B02_dissoxltmin_bdmin"
## [81] "B02_ironmax_bdmax" "B02_ironmax_bdmean"
```

## [83]	"B02_ironmax_bdmin"	"B02_ironmean_bdmax"
## [85]	"B02_ironmean_bdmean"	"B02_ironmean_bdmin"
## [87]	"B02_ironmin_bdmax"	"B02_ironmin_bdmean"
## [89]	"B02_ironmin_bdmin"	"B02_ironrange_bdmax"
## [91]	"B02_ironrange_bdmean"	"B02_ironrange_bdmin"
## [93]	"B02_ironltmax_bdmax"	"B02_ironltmax_bdmean"
## [95]	"B02_ironltmax_bdmin"	"B02_ironltmin_bdmax"
## [97]	"B02_ironltmin_bdmean"	"B02_ironltmin_bdmin"
## [99]	"B02_phosphatemax_bdmax"	"B02_phosphatemax_bdmean"
## [101]	"B02_phosphatemax_bdmin"	"B02_phosphatemean_bdmax"
## [103]	"B02_phosphatemean_bdmean"	"B02_phosphatemean_bdmin"
## [105]	"B02_phosphatemin_bdmax"	"B02_phosphatemin_bdmean"
## [107]	"B02_phosphatemin_bdmin"	"B02_phosphaterange_bdmax"
## [109]	"B02_phosphaterange_bdmean"	"B02_phosphaterange_bdmin"
## [111]	"B02_phosphateltmax_bdmax"	"B02_phosphateltmax_bdmean"
## [113]	"B02_phosphateltmax_bdmin"	"B02_phosphateltmin_bdmax"
## [115]	"B02_phosphateltmin_bdmean"	"B02_phosphateltmin_bdmin"
## [117]	"B02_lightbotmax_bdmax"	"B02_lightbotmax_bdmean"
## [119]	"B02_lightbotmax_bdmin"	"B02_lightbotmean_bdmax"
## [121]	"B02_lightbotmean_bdmean"	"B02_lightbotmean_bdmin"
## [123]	"B02_lightbotmin_bdmax"	"B02_lightbotmin_bdmean"
## [125]	"B02_lightbotmin_bdmin"	"B02_lightbotrange_bdmax"
## [127]	"B02_lightbotrange_bdmean"	"B02_lightbotrange_bdmin"
## [129]	"B02_lightbotltmax_bdmax"	"B02_lightbotltmax_bdmean"
## [131]	"B02_lightbotltmax_bdmin"	"B02_lightbotltmin_bdmax"
## [133]	"B02_lightbotltmin_bdmean"	"B02_lightbotltmin_bdmin"
## [135]	"B02_nitratemax_bdmax"	"B02_nitratemax_bdmean"
## [137]	"B02_nitratemax_bdmin"	"B02_nitratemean_bdmax"
## [139]	"B02_nitratemean_bdmean"	"B02_nitratemean_bdmin"
## [141]	"B02_nitratemin_bdmax"	"B02_nitratemin_bdmean"
## [143]	"B02_nitratemin_bdmin"	"B02_nitraterange_bdmax"
## [145]	"B02_nitraterange_bdmean"	"B02_nitraterange_bdmin"
## [147]	"B02_nitrateltmax_bdmax"	"B02_nitrateltmax_bdmean"
## [149]	"B02_nitrateltmax_bdmin"	"B02_nitrateltmin_bdmax"
## [151]	"B02_nitrateltmin_bdmean"	"B02_nitrateltmin_bdmin"
## [153]	"B02_tempmax_bdmax"	"B02_tempmax_bdmean"
## [155]	"B02_tempmax_bdmin"	"B02_tempmean_bdmax"
## [157]	"B02_tempmean_bdmean"	"B02_tempmean_bdmin"
## [159]	"B02_tempmin_bdmax"	"B02_tempmin_bdmean"
## [161]	"B02_tempmin_bdmin"	"B02_temprange_bdmax"
## [163]	"B02_temprange_bdmean"	"B02_temprange_bdmin"
## [165]	"B02_templtmax_bdmax"	"B02_templtmax_bdmean"
## [167]	"B02_templtmax_bdmin"	"B02_templtmin_bdmax"
## [169]	"B02_templtmin_bdmean"	"B02_templtmin_bdmin"
## [171]	"B02_carbonphytomax_bdmax"	"B02_carbonphytomax_bdmean"
## [173]	"B02_carbonphytomax_bdmin"	"B02_carbonphytomean_bdmax"
## [175]	"B02_carbonphytomean_bdmean"	"B02_carbonphytomean_bdmin"
## [177]	"B02_carbonphytomin_bdmax"	"B02_carbonphytomin_bdmean"
## [179]	"B02_carbonphytomin_bdmin"	"B02_carbonphytorange_bdmax"

## [181]	"B02_carbonphytorange_bdmean"	"B02_carbonphytorange_bdmin"
## [183]	"B02_carbonphytoltmax_bdmax"	"B02_carbonphytoltmax_bdmean"
## [185]	"B02_carbonphytoltmax_bdmin"	"B02_carbonphytoltmin_bdmax"
## [187]	"B02_carbonphytoltmin_bdmean"	"B02_carbonphytoltmin_bdmin"
## [189]	"B02_ppmax_bdmax"	"B02_ppmax_bdmean"
## [191]	"B02_ppmax_bdmin"	"B02_ppmean_bdmax"
## [193]	"B02_ppmean_bdmean"	"B02_ppmean_bdmin"
## [195]	"B02_ppmin_bdmax"	"B02_ppmin_bdmean"
## [197]	"B02_ppmin_bdmin"	"B02_pprange_bdmax"
## [199]	"B02_pprange_bdmean"	"B02_pprange_bdmin"
## [201]	"B02_ppltmax_bdmax"	"B02_ppltmax_bdmean"
## [203]	"B02_ppltmax_bdmin"	"B02_ppltmin_bdmax"
## [205]	"B02_ppltmin_bdmean"	"B02_ppltmin_bdmin"
## [207]	"B02_salinitymax_bdmax"	"B02_salinitymax_bdmean"
## [209]	"B02_salinitymax_bdmin"	"B02_salinitymean_bdmax"
## [211]	"B02_salinitymean_bdmean"	"B02_salinitymean_bdmin"
## [213]	"B02_salinitymin_bdmax"	"B02_salinitymin_bdmean"
## [215]	"B02_salinitymin_bdmin"	"B02_salinityrange_bdmax"
## [217]	"B02_salinityrange_bdmean"	"B02_salinityrange_bdmin"
## [219]	"B02_salinityltmax_bdmax"	"B02_salinityltmax_bdmean"
## [221]	"B02_salinityltmax_bdmin"	"B02_salinityltmin_bdmax"
## [223]	"B02_salinityltmin_bdmean"	"B02_salinityltmin_bdmin"
## [225]	"B02_silicatemax_bdmax"	"B02_silicatemax_bdmean"
## [227]	"B02_silicatemax_bdmin"	"B02_silicatemean_bdmax"
## [229]	"B02_silicatemean_bdmean"	"B02_silicatemean_bdmin"
## [231]	"B02_silicatemin_bdmax"	"B02_silicatemin_bdmean"
## [233]	"B02_silicatemin_bdmin"	"B02_silicaterange_bdmax"
## [235]	"B02_silicaterange_bdmean"	"B02_silicaterange_bdmin"
## [237]	"B02_silicateltmax_bdmax"	"B02_silicateltmax_bdmean"
## [239]	"B02_silicateltmax_bdmin"	"B02_silicateltmin_bdmax"
## [241]	"B02_silicateltmin_bdmean"	"B02_silicateltmin_bdmin"
## [243]	"B02_icecoverltmax_ss"	"B02_icecoverltmin_ss"
## [245]	"B02_icecovermax_ss"	"B02_icecovermean_ss"
## [247]	"B02_icecovermin_ss"	"B02_icecoverrange_ss"
## [249]	"B02_icethickltmax_ss"	"B02_icethickltmin_ss"
## [251]	"B02_icethickmax_ss"	"B02_icethickmean_ss"
## [253]	"B02_icethickmin_ss"	"B02_icethickrange_ss"
## [255]	"B02_templtmax_ss"	"B02_templtmin_ss"
## [257]	"B02_tempmax_ss"	"B02_tempmean_ss"
## [259]	"B02_tempmin_ss"	"B02_temprange_ss"
## [261]	"B02_chlomax_ss"	"B02_chlomean_ss"
## [263]	"B02_chlomin_ss"	"B02_chlorange_ss"
## [265]	"B02_chloltmax_ss"	"B02_chloltmin_ss"
## [267]	"B02_curvelmax_ss"	"B02_curvelmean_ss"
## [269]	"B02_curvelmin_ss"	"B02_curvelrange_ss"
## [271]	"B02_curvelltmax_ss"	"B02_curvelltmin_ss"
## [273]	"B02_dissoxmax_ss"	"B02_dissoxmean_ss"
## [275]	"B02_dissoxmin_ss"	"B02_dissoxrange_ss"
## [277]	"B02_dissoxltmax_ss"	"B02_dissoxltmin_ss"

```

## [279] "B02_ironmax_ss"          "B02_ironmean_ss"
## [281] "B02_ironmin_ss"         "B02_ironrange_ss"
## [283] "B02_ironltmax_ss"      "B02_ironltmin_ss"
## [285] "B02_phosphatemax_ss"   "B02_phosphatemean_ss"
## [287] "B02_phosphatemin_ss"   "B02_phosphaterange_ss"
## [289] "B02_phosphateltmax_ss" "B02_phosphateltmin_ss"
## [291] "B02_nitratemax_ss"     "B02_nitratemean_ss"
## [293] "B02_nitratemin_ss"     "B02_nitraterange_ss"
## [295] "B02_nitrateltmax_ss"   "B02_nitrateltmin_ss"
## [297] "B02_carbonphytomax_ss"  "B02_carbonphytomean_ss"
## [299] "B02_carbonphytomin_ss" "B02_carbonphytorange_ss"
## [301] "B02_carbonphytoltmax_ss" "B02_carbonphytoltmin_ss"
## [303] "B02_ppmax_ss"          "B02_ppmean_ss"
## [305] "B02_ppmin_ss"          "B02_pprange_ss"
## [307] "B02_ppltmax_ss"        "B02_ppltmin_ss"
## [309] "B02_salinitymax_ss"    "B02_salinitymean_ss"
## [311] "B02_salinitymin_ss"    "B02_salinityrange_ss"
## [313] "B02_salinityltmax_ss"  "B02_salinityltmin_ss"
## [315] "B02_silicatemax_ss"    "B02_silicatemean_ss"
## [317] "B02_silicatemin_ss"    "B02_silicaterange_ss"
## [319] "B02_silicatelmax_ss"   "B02_silicatelmin_ss"

```

Podemos baixar alguns desses layers diretamente no R e utilizar a informação para as análises.

```

# Carregue alguns layers como de Oxigênio dissolvido e a
# batimetria marinha media
bathy <- load_layers(c("B02_dissoxmean_ss", "B0_bathymean"))

```

```

## Warning in get_datadir(datadir): file.path(tempdir(), "sdmpredictors") will
## be used as datadir, set options(sdmpredictors_datadir="<directory>") to
## avoid re-downloading the data in every session or set the datadir parameter
## in load_layers

```

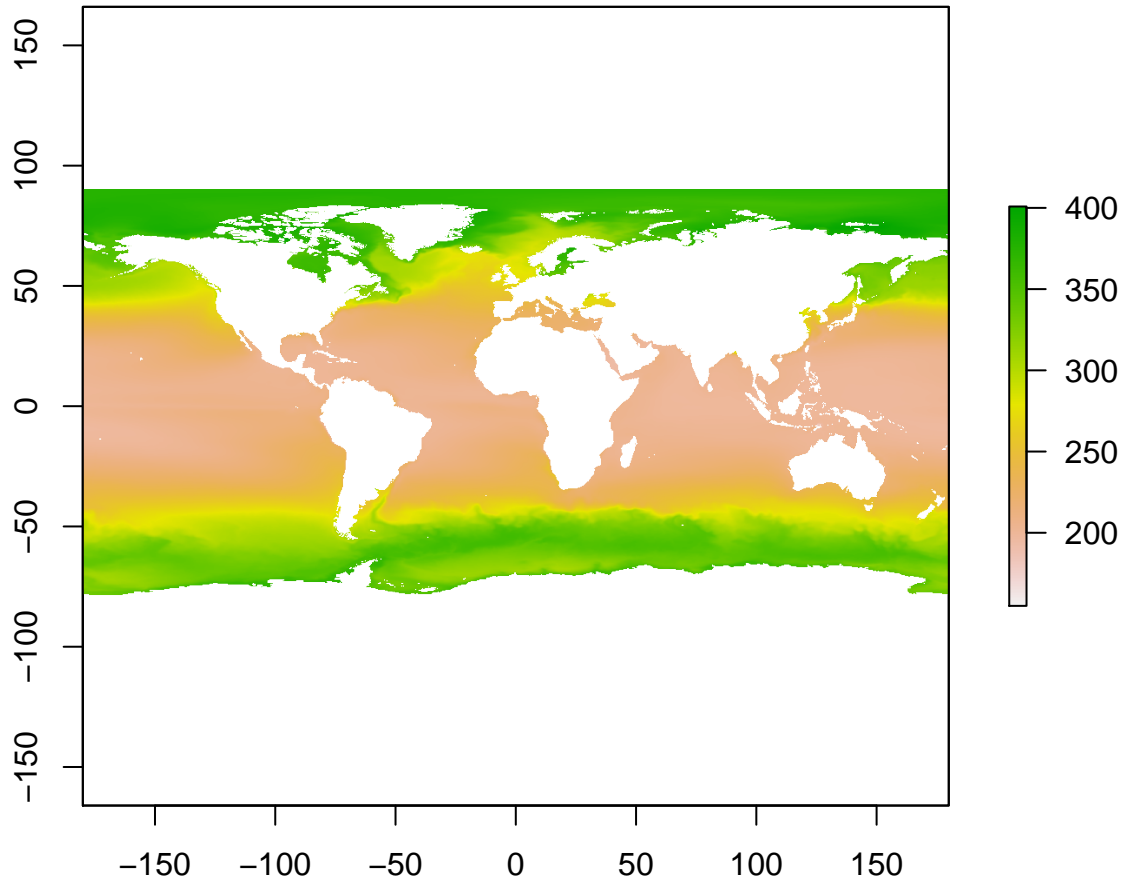
Agora que nós já baixamos os dados e guardamos na forma de um *raster* chamado **bathy**, podemos fazer um mapa das variáveis.

```

plot(bathy$B02_dissoxmean_ss, main = "Annual Mean Dissolved Oxygen")

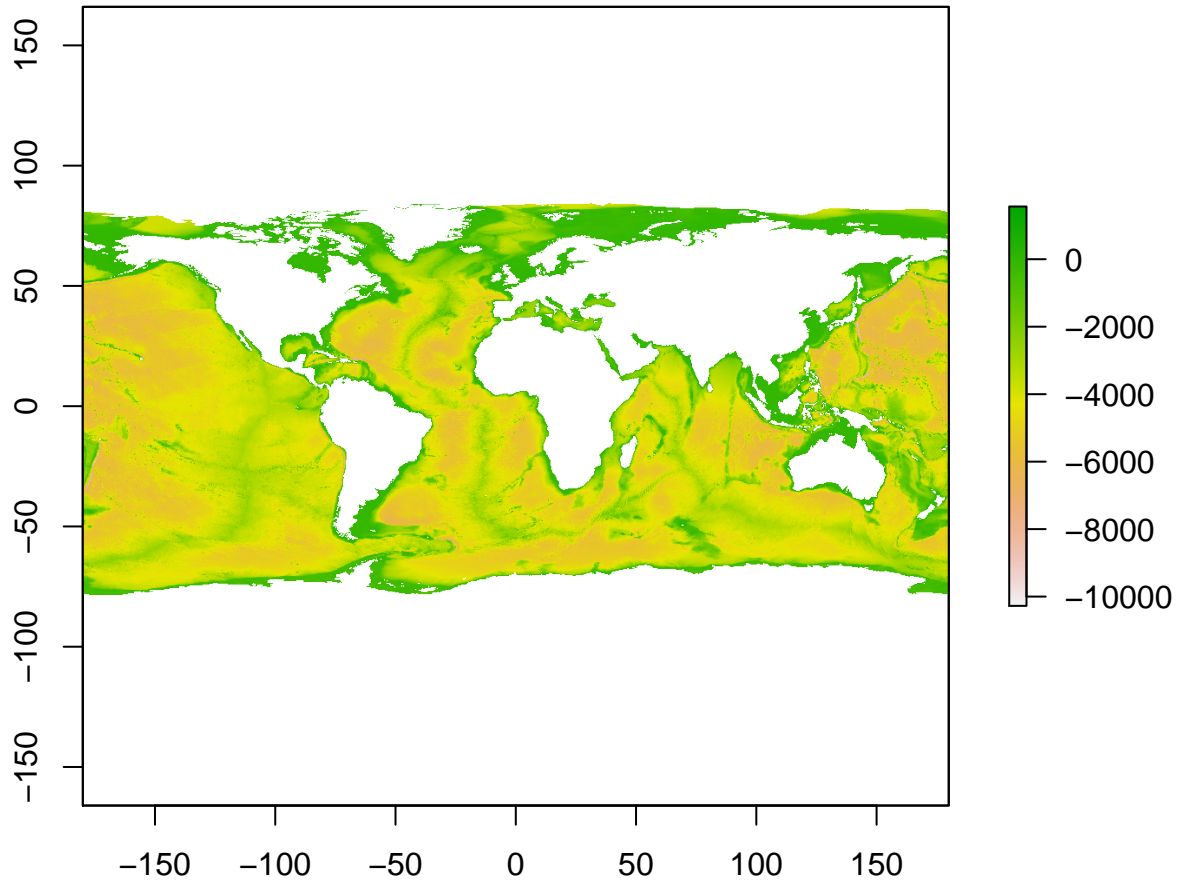
```

## Annual Mean Dissolved Oxygen



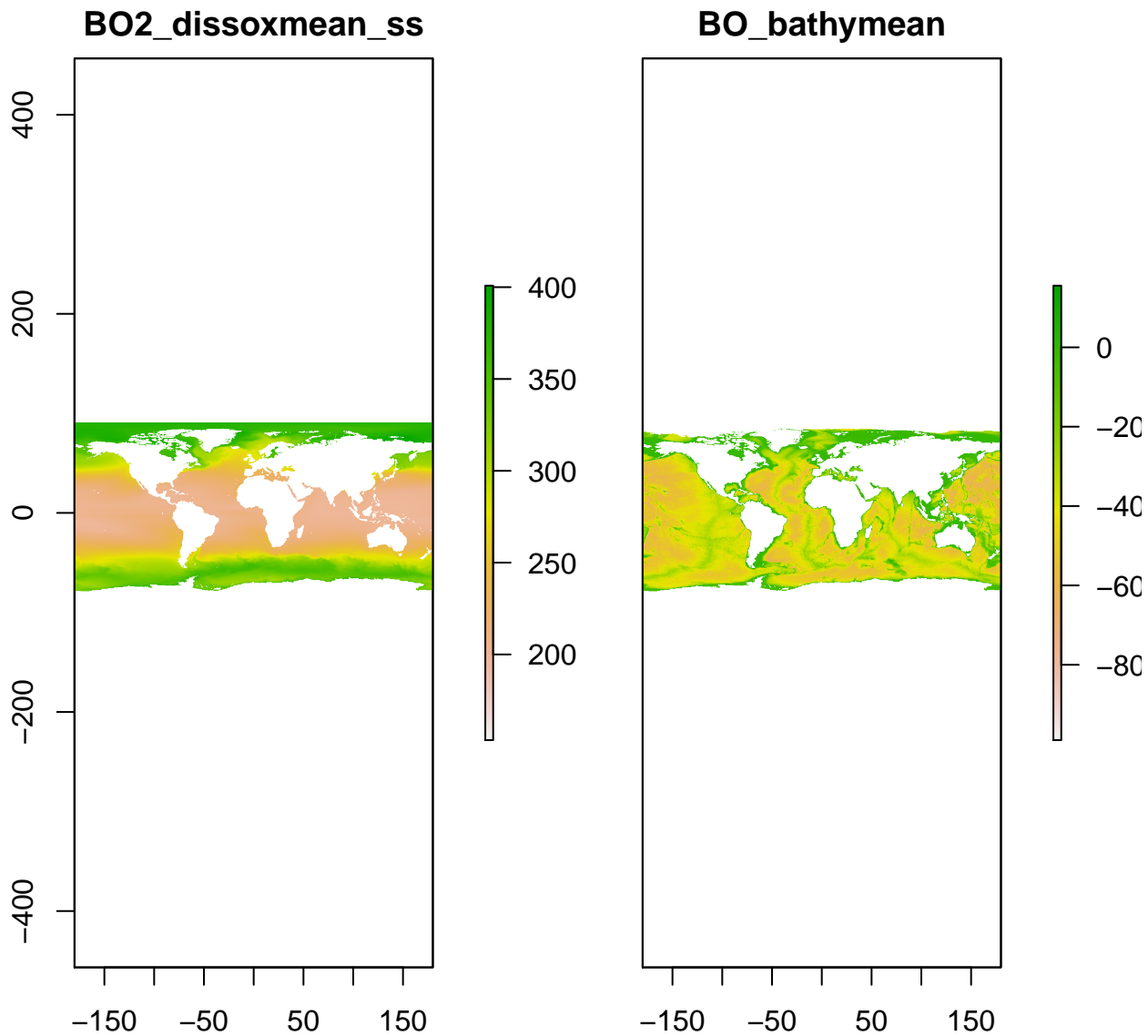
```
plot(bathy$BO_bathymean, main = "Mean Bathymetry")
```

## Mean Bathymetry



```
# ou  
plot(bathy)
```



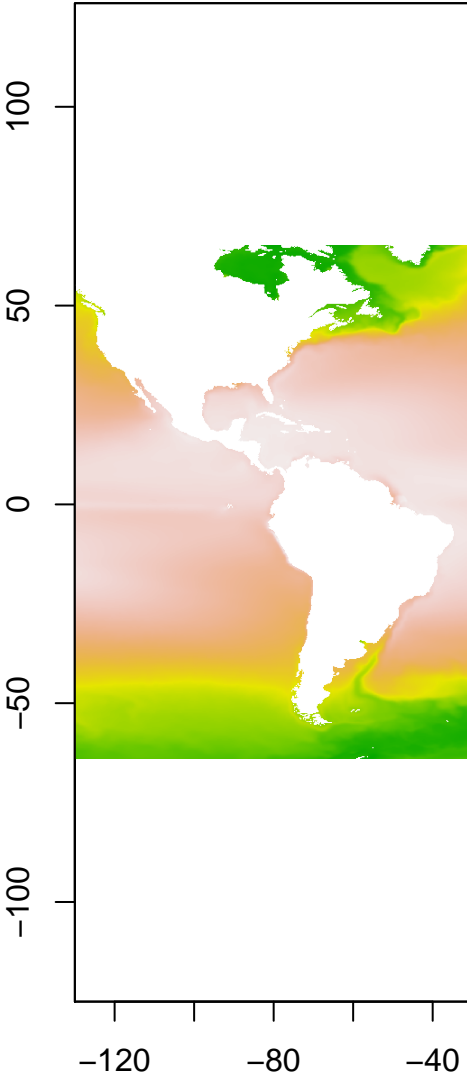


Para facilitar a manipulação, vamos cortar essas variáveis e utilizar somente a extensão do raster utilizado no dado de riqueza de peixes-borboleta. Para isso, vamos utilizar a função `crop` do pacote **raster** de acordo com o dado de riqueza.

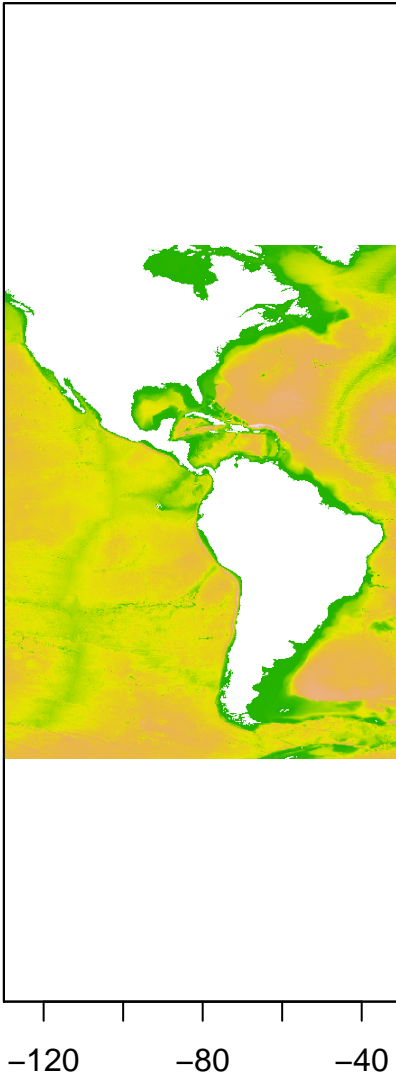
```
# Corta o raster de acordo com o mapa de riqueza
corta_ext <- extent(borb_maps$Richness_Raster)
bathy_ext <- crop(bathy, corta_ext)

plot(bathy_ext)
```

**BO2\_dissoxmean\_ss**



**BO\_bathymean**



## 4 Checando projeções de dados biológicos e climáticos

Vamos checar se as projeções do arquivo dos peixes e do mar são idênticas (digite *projeções cartografia* no Google para ter uma noção da tarefa). A linha **'coord. ref. : +proj=longlat +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0'** tem que ser idêntica em ambos; caso contrário você deverá mudar as projeções de um ou do outro arquivo. Para conferir a olho nu, faça o mapa de ambos os gráficos na mesma janela; se a sobreposição dos mapas não for perfeita, há um problema com a projeção.

```
# projeções de ambos os arquivos
```

```
borb_maps$Richness_Raster
```

```
## class      : RasterLayer
## dimensions  : 43, 33, 1419  (nrow, ncol, ncell)
## resolution  : 3, 3  (x, y)
## extent     : -130, -31, -64, 65  (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0
## data source : in memory
## names      : layer
## values     : 0, 8  (min, max)
```

```
bathy_ext$B02_dissoxmean_ss
```

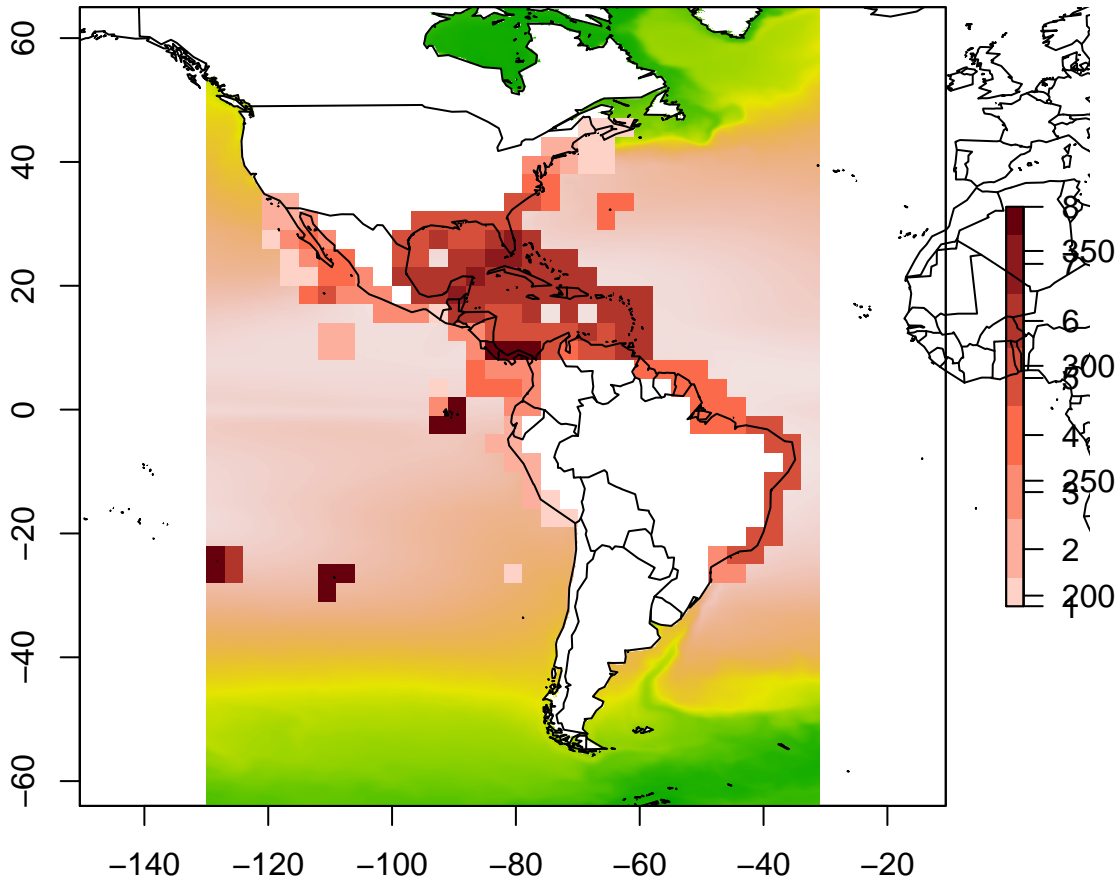
```
## class      : RasterLayer
## dimensions  : 1548, 1188, 1839024  (nrow, ncol, ncell)
## resolution  : 0.083, 0.083  (x, y)
## extent     : -130, -31, -64, 65  (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## data source : in memory
## names      : B02_dissoxmean_ss
## values     : 195, 369  (min, max)
```

```
bathy_ext$B0_bathymean
```

```
## class      : RasterLayer
## dimensions  : 1548, 1188, 1839024  (nrow, ncol, ncell)
## resolution  : 0.083, 0.083  (x, y)
## extent     : -130, -31, -64, 65  (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## data source : in memory
## names      : B0_bathymean
## values     : -8372, 1284  (min, max)
```

```
# mapa único
```

```
plot(bathy_ext$B02_dissoxmean_ss)
plot(borb_maps, add = T)
```



```
# tudo parece OK
```

```
# porém precisamos colocar as projeções com o mesmo nome
```

```
projection(borb_maps$Richness_Raster) <- projection(bathy_ext$B0_bathymean)
```

```
# agora sim tudo idêntico
```

```
borb_maps$Richness_Raster
```

```
## class      : RasterLayer
## dimensions  : 43, 33, 1419 (nrow, ncol, ncell)
## resolution  : 3, 3 (x, y)
## extent     : -130, -31, -64, 65 (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## data source : in memory
## names      : layer
## values     : 0, 8 (min, max)
```

```
bathy_ext$B02_dissoxmean_ss
```

```
## class      : RasterLayer
## dimensions  : 1548, 1188, 1839024 (nrow, ncol, ncell)
## resolution  : 0.083, 0.083 (x, y)
## extent     : -130, -31, -64, 65 (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## data source : in memory
## names      : B02_dissoxmean_ss
```

```
## values      : 195, 369 (min, max)
```

```
bathy_ext$BO_bathymean
```

```
## class       : RasterLayer
```

```
## dimensions  : 1548, 1188, 1839024 (nrow, ncol, ncell)
```

```
## resolution  : 0.083, 0.083 (x, y)
```

```
## extent     : -130, -31, -64, 65 (xmin, xmax, ymin, ymax)
```

```
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
```

```
## data source : in memory
```

```
## names      : BO_bathymean
```

```
## values     : -8372, 1284 (min, max)
```

## 5 Extraindo médias climáticas por pixel

O pixel que criamos para calcular a riqueza de espécies de peixes tem dimensão 3x3 graus, enquanto os pixels das variáveis marinhas são de 0.08333333x0.08333333 graus; ou seja, os dados climáticos são muito mais precisos que *borb\_maps*. A definição do tamanho do pixel da informação biológica depende da tua pergunta, mas, em geral, os macroecólogos utilizam 1x1 grau ou 0.5x0.5 grau. O fato é que para cada célula de mamífero, nós temos muitas células de variáveis marinhas; por isso, vamos *extrair* a média dos valores marinhas dentro de cada célula de peixes; em seguida, precisamos *juntar/fundir* ambas informações para termos uma correspondência entre a riqueza de peixes e, por exemplo, a quantidade média de oxigênio dissolvido por pixel. Para isso usaremos a função **lets.addvar**.

```
# extrair e fundir variáveis climáticas aos mamíferos
borb_mar <- lets.addvar(borb_maps, bathy_ext, fun = mean)
# veja um pedaço da matriz resultante
borb_mar[1:5, c(1:5, 25:27)]

##      Longitude(x) Latitude(y) Amphichaetodon melbae Chaetodon auriga
## [1,]           -68           46                0           0
## [2,]           -66           46                0           0
## [3,]           -62           46                0           0
## [4,]           -74           42                0           0
## [5,]           -72           42                0           0
##      Chaetodon capistratus Prognathodes guyanensis B02_dissoxmean_ss_mean
## [1,]                    0                    0                    293
## [2,]                    0                    0                    303
## [3,]                    0                    0                    313
## [4,]                    1                    0                    280
## [5,]                    1                    0                    279
##      B0_bathymean_mean
## [1,]           -47.1
## [2,]           -34.7
## [3,]           -57.6
## [4,]            -2.3
## [5,]           -32.9
```

Ótimo, agora temos uma planilha pixel vs espécie e uma série de preditores para testar nossas hipóteses. Vamos às análises!

## 6 Calculando relações simples entre riqueza e os preditores

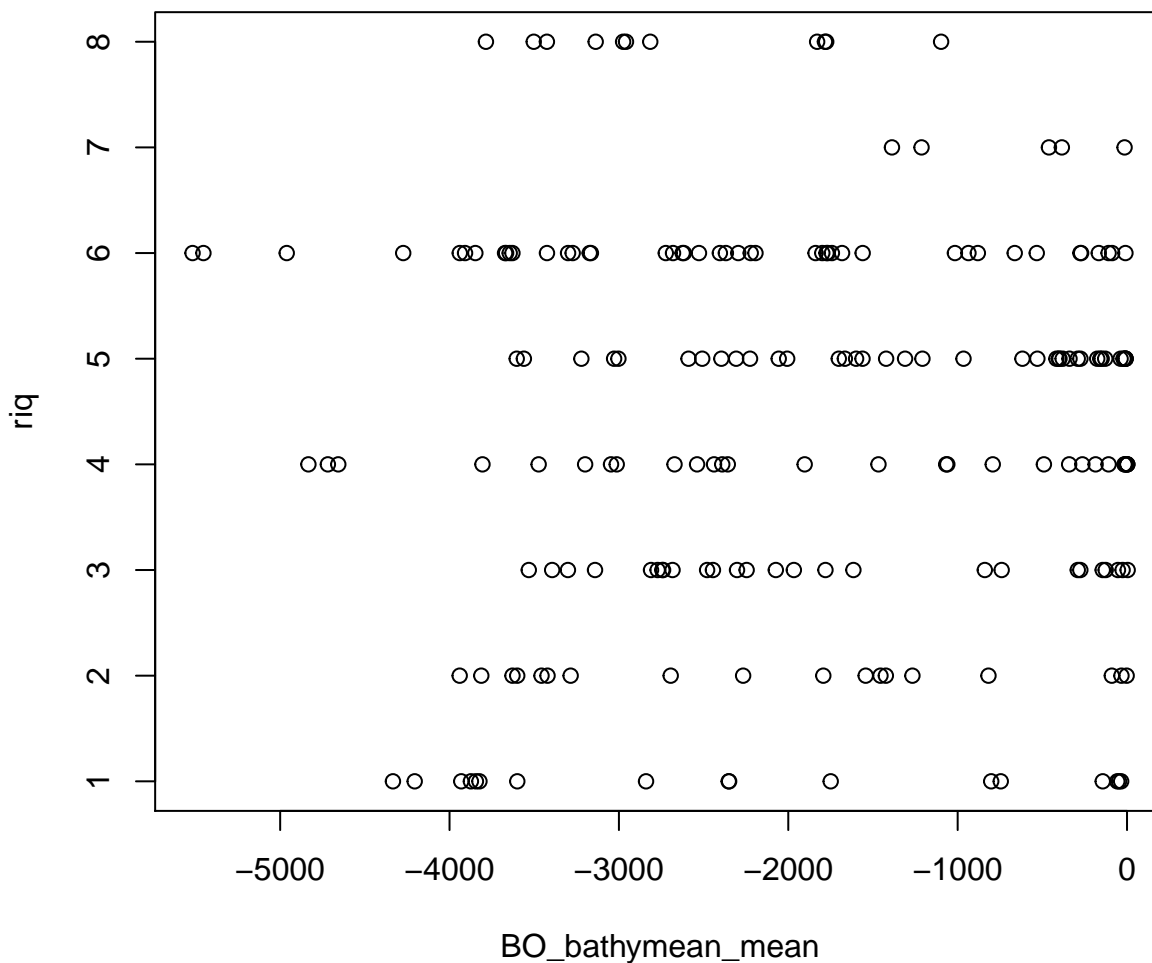
### 6.1 Calculando a riqueza de espécies

Com a matriz final já esta praticamente pronta, nos só precisamos incluir uma nova coluna onde calcularemos a riqueza de espécies. Para isso, basta somar a ocorrência de todas as espécies ao longo das linhas desta matriz (exceto as duas primeiras e as duas últimas que não possuem informação de ocorrência) e teremos ao final a riqueza de espécies.

```
# Precisamos transformar a matriz em um data.frame para  
# facilitar  
borb_mar <- as.data.frame(borb_mar)  
  
# Calcular a riqueza de espécies  
borb_mar$riq <- rowSums(borb_mar[, -c(1:2, 26:27)])
```

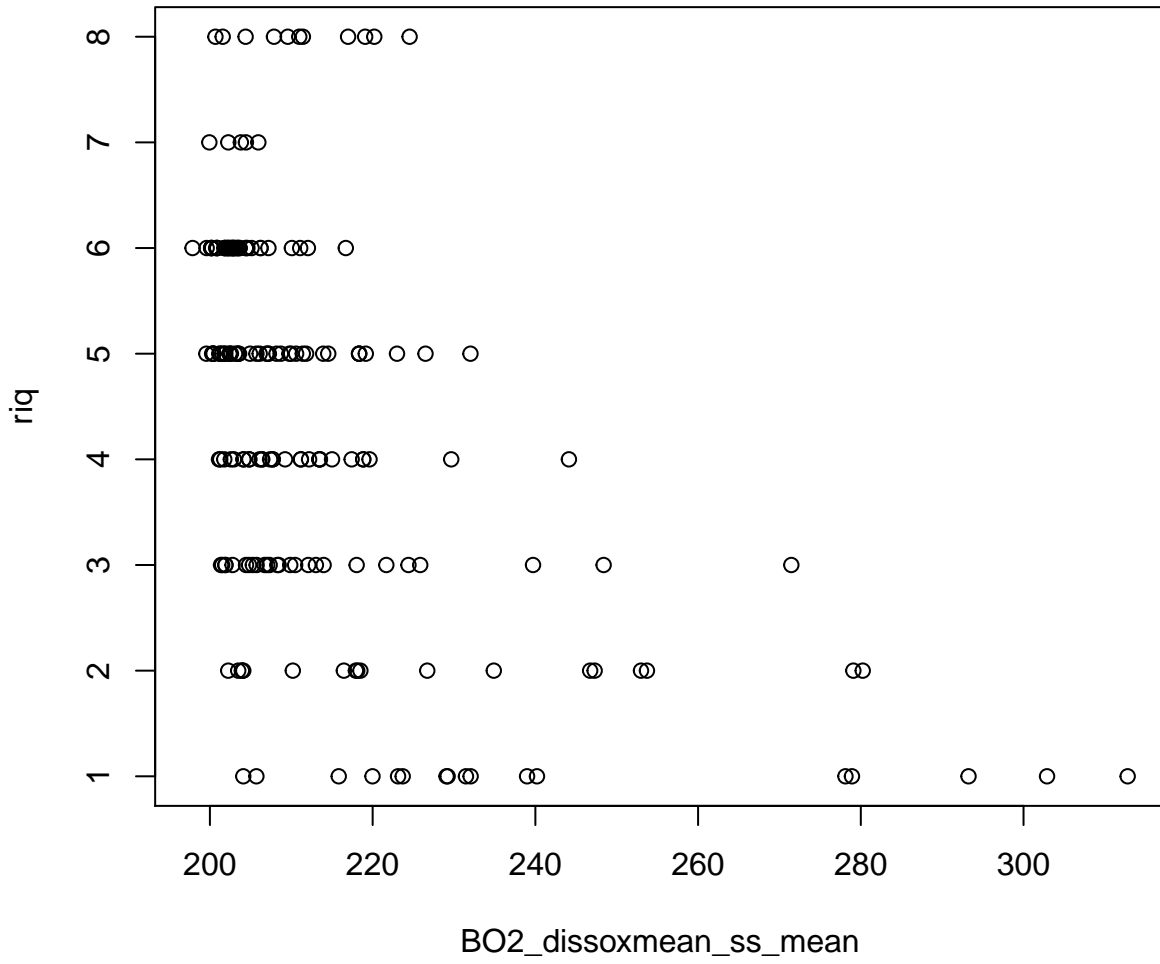
Qual a relação entre a riqueza de espécies e a batimetria marinha?

```
plot(riq ~ BO_bathymean_mean, borb_mar)
```



Qual a relação entre a riqueza de espécies e oxigênio dissolvido?

```
plot(riq ~ BO2_dissoxmean_ss_mean, borb_mar)
```





# Bibliography

1. Vilela, B. & Villalobos, F. (2015). letsR: A new r package for data handling and analysis in macroecology. *Methods in Ecology and Evolution*, 6, 1229–1234.