

# Bianca Tocci



Mestranda em Oceanografia Biológica, Instituto Oceanográfico, USP. Trabalho com taxonomia e ecologia do fitoplâncton marinho.

## Exercícios

Linque para a página com os meus exercícios resolvidos [exec](#)

## Proposta de Trabalho Final

### Plano A

A proposta da função consiste em automatizar uma rotina para calcular índices ecológicos (Margalef, Shannon, Pielou), a partir de arquivos contendo dados de densidade de espécies fitoplanctônicas (cel/L), e a criação de histogramas da variação das classes taxonômicas em um intervalo de tempo.

Especificamente:

1-Cada arquivo refere-se a um evento oceanográfico diferente, onde cada coluna possui a quantidade de cel/L de cada espécie identificada em um período da coleta.

2-As espécies estão listadas dentro de classes taxonômicas (diatomáceas, dinoflagelados, cianobactérias, flagelados, cocolitoforídeos).

3-A função consistiria em calcular os índices de arquivos data.frame em um mesmo diretório, construindo tabelas com os valores resultantes.

4-A função também calcularia a densidade total por classe taxonômica de cada coluna, plotando em um histograma a variação da comunidade nesse intervalo de tempo.

Essa função facilitaria a análise dos meus dados de coletas mensais, ao longo de um ano em 2 pontos.

### Plano B

Criar uma função para calcular quantas células fitoplanctônicas aproximadamente ficariam grudadas no corpo de uma pessoa depois de um mergulho no mar. Como cada ambiente apresenta um valor de densidade fitoplanctônica (cel/L) de acordo com as condições físicas locais, a primeira informação necessária seria saber qual a média de cel/L. Sabe-se que a variação da cor do oceano depende do que tem na água em suspensão, como as microalgas e outras partículas. Sendo assim, atribuindo valores médios conhecidos de cel/L as cores de água do mar mais comuns em nosso litoral de acordo

com a escala de Forel-Ule, seria possível começar a função.

Para isso, a ideia seria pedir ao usuário que respondesse algumas perguntas.

1-Qual a cor da água do mar na hora do mergulho? A resposta do usuário resultaria na média de células de fitoplâncton por litro, de acordo com a cor escolhida.

2-Para calcular a superfície de contato (o corpo da pessoa) o usuário precisaria responder sua altura e o tamanho do quadril.

A partir daí, realizar transformações de unidade de medida da superfície de contato, para calcular hipoteticamente a quantidade aproximada de células de fitoplâncton que ficariam grudadas no corpo. O cálculo eu preciso formular melhor.

A proposta A está bem estruturada e constituirá um bom desafio para o trabalho final. Uma sugestão é que você deixe a proposta um pouco mais genérica, ou seja, permita que os índices sejam calculados para diferentes situações amostrais (e não só para espécies do fitoplâncton). Na verdade, é simples de generalizar a análise pra qualquer situação que tu tenha uma amostra de diferentes espécies e uma unidade de esforço ou densidade. Outra sugestão é começar com um índice mais simples e conforme você for implementando os índices mais simples, passe para os mais complexos. Enfim, não precisa implementar todos os índices caso haja muita dificuldade. Com relação à proposta B achei um pouco mais simples e menos desafiante. Também tenho dúvidas se a água do mar é um bom indicador da densidade de fitoplâncton porque é um critério altamente subjetivo. ---

[Leonardo](#)

## Plano A - Concluído

obs.: A proposta foi reformulada para se tornar mais genérica, foi utilizado somente um índice devido ao meu nível de dificuldade, e ao invés de um histograma a função plotará um gráfico de barras para melhor visualização da variação do período ou situação amostral.

A função consiste em automatizar uma rotina para calcular o índice de diversidade de Shannon Wiener, a partir de arquivos contendo dados de densidade de espécies, e a criação de um gráfico de barras da variação dos grupos em um período ou situação amostral.

Especificamente:

1-Cada arquivo refere-se a um período ou uma situação amostral diferente, onde cada coluna possui a densidade de cada espécie identificada.

2-As espécies estão listadas dentro grupos.

3-O calculo dos índices a partir de arquivos data.frame em um mesmo diretório, onde os valores são salvos em tabelas no formato .txt

4-Soma da densidade total de cada grupo de cada coluna, plotando em um gráfico de barras a variação dos grupos em um determinado período ou situação amostral

## Código comentado

```
#Função para automatizar cálculo do índice de Shannon Wiener, a partir de
planilhas em .txt, contidas
#em um mesmo diretório, criando uma tabela com os valores obtidos para cada
arquivo. A função também realiza a #soma das densidades das espécies
contidas em cada grupo amostrado durante um período ou situação amostral
#diferente. A função termina plotando um gráfico da variação da soma dos
grupos em
#um determinado período/coleta. Pode-se inserir outros índices ecológicos.

auto.indices=function(files){
  dir=file.choose() #abre uma janela para a escolha do arquivo.
  #Para MacIntosh file.choose(), para Windows choose.files()
  a=unlist(strsplit(dir, "")) #separa os caracteres do caminho do arquivo
  final=(max(which(a==" /")) #vetor logico para selecionar os caracteres
contido depois da última barra
  #Para MacIntosh (a==" /"), para Windows (a=="\\")
  dir=noquote(paste(a[1:final],sep="",collapse="")) #junta os caracteres do
nome do arquivo
  setwd(dir) #estabelece o diretório na pasta aonde foi escolhido o arquivo
para a análise
  # se não for dado os arquivos de entrada, a função irá rodar para todos os
arquivos .txt do diretório
  if (missing(files)){files=list.files(pattern=".txt")}
  else {files=files}
  for(file in files){ #loop para cada arquivo
    data=read.table(file, header=TRUE, dec=".", sep="\t", fill=TRUE)
    # matriz criada de acordo com o arquivo inicial
    # sendo o número de linhas igual a quantidade de índices e o número de
colunas igual ao arquivo original
    mat=matrix(NA,1,length(data))
    mat[,1]="H"
    mat[,2]="Shannon-Wiener"
    # loop para calcular o índice de Shannon-Wiener
    for(i in seq(3,length(data))){ #calcula o índice da terceira coluna até
o final
      # Shanon-Wiener index (H)
      data2=data #cria um data2 para não sobrescrever o original quando
substituir o 0 por NA
      data2[data2==0]=NA # substitui 0 por NA por causa do índice
      soma=sum(data2[,i], na.rm=T) #soma de cada coluna
```

```
    pi=data2[,i]/soma #calcula a proporção de cada espécie em relação ao
total
    H=-(sum(pi*log(pi), na.rm=T)) #calcula o índice
    mat[,i]=H #cola os valores dos índices em cada tempo respectivo na
matriz criada fora do loop
  } # loop do shannon-wiener acaba aqui
  df=data.frame(mat, row.names=NULL) #cria um data.frame pra poder salvar
em txt
  names=c("Indice",names(data[,3:length(data)])) #puxa os nomes dos tempos
do arquivo original
  write.table(df[,2:length(df)], file=paste(file,"_indice.txt",sep=""),
quote=FALSE, sep="\t", col.names=names,row.names=FALSE, dec=".")
  #esse write.table foi usado da segunda coluna em diante do data.frame
pra remover a primeira coluna que não será utilizada
#####TERMINA O CÁLCULO DOS INDICES#####
####SOMA DOS GRUPOS####
mat2=matrix(NA,length(levels(factor(data[,1]))),length(data))
#mat2 tem a quantidade de linhas igual ao número de grupos
#e a quantidade de colunas igual ao arquivo original
mat2[,2]=levels(factor(data[,1])) # cola os nomes/levels dos grupos
for(i in seq(3,length(data))){ #aqui começa o loop para fazer a
somatória dos grupos
  a=tapply(data[,i],factor(data[,1]),sum) #somatória das
espécies(linhas) que pertencem ao mesmo grupo
  mat2[,i]=a #cola na matriz
} #aqui termina o loop da soma dos grupos
mat.plot=mat2[,2:ncol(mat2)] # exclui a primeira coluna que não será
utilizada
palette(rainbow(nrow(mat.plot))) #padroniza a palheta de cores
jpeg(filename=paste(file,"_plot.jpg",sep=""), 3200, 2000, quality=600,
type="quartz", res=300)
#salva o gráfico em jpeg
#Para MacIntosh "quartz", para Windows "windows" ou "care"
par(mar=c(4,5,1,6)) #margens
colors=factor(c(mat.plot[,2])) #atribui uma cor pra cada grupo dentro da
palheta de cores selecionada
barplot(mat.plot[,2:ncol(mat.plot)],col=colors, ylab="Densidade celular
[cel/L]", names.arg=names(data[,3:length(data)])) #criação barplot
par(xpd=T) # permite que a legenda fique fora do gráfico
x=ncol(mat.plot)+0.1 #posição X da legenda
y=max(as.numeric(mat.plot[,2:ncol(mat.plot)])) #posição Y da legenda
legend(x,y,c(mat.plot[,1]), col=colors, pch=22, pt.bg=colors) #legenda
dev.off()
} #aqui fecha loop de cada arquivo
} #FECHA A FUNÇÃO
```

## Código sem comentários

```
auto.indices=function(files){

  dir=file.choose()
  a=unlist(strsplit(dir, ""))
  final=(max(which(a==" /"))
  dir=noquote(paste(a[1:final],sep="",collapse=""))
  setwd(dir)
  if (missing(files)){files=list.files(pattern=".txt")}
  else {files=files}
  for(file in files){
    data=read.table(file, header=TRUE, dec=".", sep="\t", fill=TRUE)
    mat=matrix(NA,1,length(data))
    mat[,1]="H"
    mat[,2]="Shannon-Wiener"
    for(i in seq(3,length(data))){
      data2=data
      data2[data2==0]=NA
      soma=sum(data2[,i], na.rm=T)
      pi=data2[,i]/soma
      H=-(sum(pi*log(pi), na.rm=T))
      mat[,i]=H
    }
    df=data.frame(mat, row.names=NULL)
    names=c("Indice",names(data[,3:length(data)]))
    write.table(df[,2:length(df)], file=paste(file,"_indice.txt",sep=""),
    quote=FALSE, sep="\t", col.names=names,row.names=FALSE, dec=".")

    mat2=matrix(NA,length(levels(factor(data[,1]))),length(data))
    mat2[,2]=levels(factor(data[,1]))
    for(i in seq(3,length(data))){
      a=tapply(data[,i],factor(data[,1]),sum)
      mat2[,i]=a
    }
    mat.plot=mat2[,2:ncol(mat2)]

    palette(rainbow(nrow(mat.plot)))
    jpeg(filename=paste(file,"_plot.jpg",sep=""), 3200, 2000, quality=600,
    type="quartz", res=300)
    par(mar=c(4,5,1,6))
    colors=factor(c(mat.plot[,2]))
    barplot(mat.plot[,2:ncol(mat.plot)],col=colors, ylab="Densidade celular
[cel/L]", names.arg=names(data[,3:length(data)]))
    par(xpd=T)
    x=ncol(mat.plot)+0.1
    y=max(as.numeric(mat.plot[,2:ncol(mat.plot)]))
    legend(x,y,c(mat.plot[,1]), col=colors, pch=22, pt.bg=colors)
    dev.off()
  }
}
```

## Página de ajuda

auto.indices

package:unknown

R Documentation

### Automatização de calculo de diversidade

#### Description:

Função para automatizar cálculo de diversidade do índice de Shannon Wiener, de um conjunto de arquivos .txt contidos em um mesmo diretório, criando uma tabela com os valores obtidos para cada arquivo. O arquivo consiste em uma planilha com dados de densidade de espécies amostrados em um determinado período ou a variação desejada, distribuídas em colunas. A função termina plotando um gráfico de barras da variação da soma dos grupos em um determinado período ou situação amostral.

#### Usage:

```
auto.indices=function(files)
```

#### Arguments:

Files lista dos arquivos dos dados a serem lidos. Cada linha da tabela consiste nas densidades das espécies amostradas ao longo de um período (a partir da 3ª coluna). Se <files> não for especificado, <auto.indices()> irá calcular automaticamente a função para todos os arquivos presentes no diretório atual <getwd()> e usá-los para importar os dados.

#### Details:

O arquivo de entrada de dados, DEVE conter as duas primeiras colunas com os nomes dos grupos e das espécies respectivamente. Os arquivos não precisam ter número de colunas e linhas iguais. Ao menos que <files> não seja especificado, todos os arquivos dentro do diretório atual serão lidos.

#### Value:

df: Tabela <data.frame> com os valores do índice de Shannon Wiener. Para cada arquivo, será salvo \_index.txt ao nome original.  
mat.plot: arquivo .txt com os valores das somas de cada grupo para cada arquivo.

```
jpeg: salva o gráfico das somas de cada grupo por período. Para cada
arquivo, será salvo _plot.jpg
      ao nome original.
```

#### Warning:

```
Para MacIntosh file.choose() e para Windows choose.files().
which() para MacIntosh (a=="/"), para Windows (a=="\\").
jpeg(..., type= "quartz") Para MacIntosh "quartz", para Windows
"windows" ou "care"
```

#### Author(s):

```
Bianca Reis Castaldi Tocci
biatocci@gmail.com
```

#### References:

```
Crawley, M. J. (2007). Data Input. Chapter 3 of The R Book. eds M. J.
Crawley. John Wiley & Sons, Ltd.
```

#### See Also:

```
read.table() list.files() tapply() barplot()
```

#### Examples:

```
auto.indices()
auto.indices("arquivo.txt")
auto.indices("C:/Users/windows/Documents")
```

## Scripts e arquivos de exemplo

[auto\\_indices.r](#) [especies\\_exemplo.txt](#) [especies\\_exemplo2.txt](#) [pagina\\_ajuda.txt](#)

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2015:alunos:trabalho\\_final:biatocci:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2015:alunos:trabalho_final:biatocci:start)



Last update: **2020/08/12 06:04**