

Catarina Marcolin



Doutoranda em Oceanografia Biológica, Instituto de Oceanografia, USP. Fiz mestrado em Ecologia e Biomonitoramento na Universidade Federal da Bahia. Trabalho com Zooplâncton e sempre tive afinidade com as disciplinas relacionadas a técnicas estatísticas e delineamento amostral. Como parte do doutorado, tenho obtido uma quantidade enorme de dados pra tratar e tenho pedido “ajuda aos universitários” para utilizar ferramentas como o MatLab. Espero após esta disciplina alcançar independência!

Exercícios

[execercicio_1.r](#) [exercicio2_bloco.txt](#) [exercicio3.r](#) [exercicio4.r](#) [e52.r](#) [e62.r](#) [e72.r](#) [8.r](#) [9.r](#)

Proposta de Trabalho Final

Calcular o espectro de biomassa normalizada do zooplâncton em cada uma de minhas amostras. As amostras de zooplâncton foram escaneadas e um software de reconhecimento semi-automático gerou planilhas em formato .csv com muitas variáveis coletadas sobre as imagens dos organismos. Desejo utilizar o R para ler minhas planilhas em formato .csv e extrair a informação das colunas ESD e espécies; excluir as “espécies” que não interessam (ex: detritos, fibras); converter os valores em ESD de pixel para micrômetros; atribuir os indivíduos à classes de tamanho pré-estabelecidas que são equivalentes às classes de biomassa de interesse; calcular a biomassa a partir dos valores do ESD (um) para cada indivíduo; somar a biomassa acumulada em cada classe e normalizar (dividir pelo peso da classe). Logaritimizar. Um complicador é que para cada amostra eu tenho 2 planilhas cujos valores precisam ser somados e antes de logaritimizar, a soma da biomassa acumulada em cada classe deve ser multiplicada por um valor (valor da alíquota) e dividido pelo volume filtrado (para cada planilha tenho valores diferentes de alíquota e Vfilt). Só depois os valores podem ser somados e logaritimizados. Plotar um gráfico de dispersão (Log Biomassa Normalizada x Log Classes de Biomassa).

Minha Função

```
NBSS=function(a1,a2,a3, al1=1, al2=1, al3=1, num.files=2, vf=1, dif.LOPC=1)
{
  classes.um=c(90, 97.2, 104.9, 113.3, 122.3, 132.1, 142.6, 154, 166.3, 179.6,
  193.9, 209.3, 226.1, 244.1, 263.6, 284.6, 307.3,331.8,358.3,386.9, 417.7,
  451.1, 487, 525.9, 567.8, 613.1, 662, 714.9, 771.9, 833.5, 900, 971.8,
  1049.3, 1133, 1223.4, 1321, 1426.3, 1540.1, 1663, 1795.7, 1938.9, 2093.7,
  2260.7, 2441, 2635.6, 2846, 3073, 3318, 3583, 3869, 4177, 4511, 4870, 5259,
  5678)
  dif.classes.um = diff(classes.um)
  dif.classes.w = dif.classes.um*(10^-6) # Converting classes from micrometers
```

```
to ugC.
dif.classes.wm = ((dif.classes.w/2)^3)*(4/3)*pi # Converting classes from
micrometers to ugC.
dif.classes.wum = dif.classes.wm*(10^6) # Converting classes from
micrometers to ugC.
dif.classes.we = (log10(dif.classes.wum)+1.429)/0.808 # Converting classes
from micrometers to ugC.
dif.classes.weight =10^(dif.classes.we)

if (num.files==1)
{
  al$BiXos=al[,length(al)] ## Last column contains information of
particles classification after validation.
  al.clean=al[al$BiXos!="detrito_claro" & al$BiXos!="detrit_filamen" &
al$BiXos!="Detrito_claro" & al$BiXos!="detritos" &
al$BiXos!="detrito_escuro" & al$BiXos!="filamentos" & al$BiXos!="múltiplos"
& al$BiXos!="Múltiplos" & al$BiXos!="muda_crustacea" &
al$BiXos!="gelatinosos_restos" & al$BiXos!="Muda_crustacea",] ## Exclude
particles that are not planktonic organisms such as detritus.
  al.clean$ESD.pixel=al.clean[,57] ## ESD diameter values, always the same
column at plankton identifier files.
  ESD.um=10.58*(al.clean$ESD.pixel) ## Conversion from pixels to
micrometers.
  esferal.oblonga=((ESD.um/10^6)/2)^3)*4/9*pi # Assumes particles have
the shape of an oblong sphere, aspect:ratio=1:3.
  biovol1.ml=esferal.oblonga*10^6
  biomass1.ugC=(10^((log10(biovol1.ml)+1.429)/0.808))*1000 # Converts
biovolume to biomass (ugC).
  data.al=data.frame(ESD.um, biomass1.ugC)

# Attribute each organism to a size interval (size bin):

resulta=rep(NA, 54)
for (j in 1:54)
{
  resulta[j]=sum(data.al$biomass1.ugC[data.al$ESD.um >= classes.um[j]
& data.al$ESD.um < classes.um[j+1]])
}
# Split-ratio correction - aliquot (1/x) of the first file.
resulta.all=resulta*all

# LOPC tunel area and volume filtered correction. Volume in cubic meters.
resulta.corr=(resulta.all)/(dif.LOPC*vf)
}

if (num.files==2)

{
  al$BiXos=al[,length(al)] ## Last column contains information of
particles classification after validation.
```

```
a1.clean=a1[a1$BiXos!="detrito_claro" & a1$BiXos!="detrit_filamen" &
a1$BiXos!="Detrito_claro" & a1$BiXos!="detritos" &
a1$BiXos!="detrito_escuro" & a1$BiXos!="filamentos" & a1$BiXos!="múltiplos"
& a1$BiXos!="Múltiplos" & a1$BiXos!="muda_crustacea" &
a1$BiXos!="gelatinosos_restos" & a1$BiXos!="Muda_crustacea",] ## Exclude
particles that are not planktonic organisms such as detritus.
a1.clean$ESD.pixel=a1.clean[,57] ## ESD diameter values, always the same
column at plankton identifier files.
ESD.um=10.58*(a1.clean$ESD.pixel) ## Conversion from pixels to
micrometers.
esfera1.oblonga=((ESD.um/10^6)/2)^3*4/9*pi
biovol1.ml=esfera1.oblonga*10^6
biomass1.ugC=(10^((log10(biovol1.ml)+1.429)/0.808))*1000 # Converts
biovolume to biomass (ugC).
data.a1=data.frame(ESD.um, biomass1.ugC)

# Attribute each organism to a size interval (size bin):

resulta=rep(NA, 54)
for (j in 1:54)
{
  resulta[j]=sum(data.a1$biomass1.ugC[data.a1$ESD.um >= classes.um[j]
& data.a1$ESD.um < classes.um[j+1]])
}

# Split-ratio correction - aliquot (1/x) of the first file.
resulta.all=resulta*all

# Next file!

a2$BiXos=a2[,length(a2)]
a2.clean=a2[a2$BiXos!="detrit_filamen" & a2$BiXos!="detrito_claro" &
a2$BiXos!="Detrito_claro" & a2$BiXos!="detritos" &
a2$BiXos!="detrito_escuro" & a2$BiXos!="filamentos" & a2$BiXos!="múltiplos"
& a2$BiXos!="Múltiplos" & a2$BiXos!="muda_crustacea" &
a2$BiXos!="gelatinosos_restos" & a2$BiXos!="Muda_crustacea",]
a2.clean$ESD.pixel=a2.clean[,57]
ESD2.um=10.58*(a2.clean$ESD.pixel)
esfera2.oblonga=((ESD2.um/10^6)/2)^3*4/9*pi
biovol2.ml=esfera2.oblonga*10^6
biomass2.ugC=(10^((log10(biovol2.ml)+1.429)/0.808))*1000 # Converts
biovolume to biomass (ugC).
data.a2=data.frame(ESD2.um, biomass2.ugC)

# Attribute each organism to a size interval (size bin):

resulta2=rep(NA, 54)
for (j in 1:54)
{
  resulta2[j]=sum(data.a2$biomass2.ugC[data.a2$ESD2.um >=
classes.um[j] & data.a2$ESD2.um < classes.um[j+1]])
```

```
}  
# Split-ratio correction - aliquot (1/x) of the second file.  
  
resulta.al2=resulta2*al2  
  
# LOPC tunel area and volume filtered correction. Volume in cubic meters.  
resulta.corr=(resulta.al1+resulta.al2)/(dif.LOPC*vf)  
}  
  
if (num.files==3)  
{  
  al$BiXos=al[,length(al)] ## Last column contains information of  
particles classification after validation.  
  al.clean=al[al$BiXos!="detrito_claro" & al$BiXos!="detrit_filamen" &  
al$BiXos!="Detrito_claro" & al$BiXos!="detritos" &  
al$BiXos!="detrito_escuro" & al$BiXos!="filamentos" & al$BiXos!="múltiplos"  
& al$BiXos!="Múltiplos" & al$BiXos!="muda_crustacea" &  
al$BiXos!="gelatinosos_restos" & al$BiXos!="Muda_crustacea",] ## Exclude  
particles that are not planktonic organisms such as detritus.  
  al.clean$ESD.pixel=al.clean[,57] ## ESD diameter values, always the same  
column at plankton identifier files.  
  ESD.um=10.58*(al.clean$ESD.pixel) ## Conversion from pixels to  
micrometers.  
  esfera1.oblonga=((ESD.um/10^6)/2)^3)*4/9*pi  
  biovol1.ml=esfera1.oblonga*10^6  
  biomass1.ugC=(10^((log10(biovol1.ml)+1.429)/0.808))*1000 # Converts  
biovolume to biomass (ugC).  
  data.al=data.frame(ESD.um, biomass1.ugC)  
  
# Attribute each organism to a size interval (size bin):  
  
resulta=rep(NA, 54)  
for (j in 1:54)  
{  
  resulta[j]=sum(data.al$biomass1.ugC[data.al$ESD.um >= classes.um[j]  
& data.al$ESD.um < classes.um[j+1]])  
}  
# Split-ratio correction - aliquot (1/x) of the first file.  
resulta.al1=resulta*al1  
  
# Next file!  
  
a2$BiXos=a2[,length(a2)]  
a2.clean=a2[a2$BiXos!="detrit_filamen" & a2$BiXos!="detrito_claro" &  
a2$BiXos!="Detrito_claro" & a2$BiXos!="detritos" &  
a2$BiXos!="detrito_escuro" & a2$BiXos!="filamentos" & a2$BiXos!="múltiplos"  
& a2$BiXos!="Múltiplos" & a2$BiXos!="muda_crustacea" &  
a2$BiXos!="gelatinosos_restos" & a2$BiXos!="Muda_crustacea",]  
a2.clean$ESD.pixel=a2.clean[,57]  
ESD2.um=10.58*(a2.clean$ESD.pixel)
```

```
esfera2.oblonga=((ESD2.um/10^6)/2)^3)*4/9*pi
biovol2.ml=esfera2.oblonga*10^6
biomass2.ugC=(10^((log10(biovol2.ml)+1.429)/0.808))*1000 # Converts
biovolume to biomass (ugC).
data.a2=data.frame(ESD2.um, biomass2.ugC)

# Attribute each organism to a size interval (size bin):

resulta2=rep(NA, 54)
for (j in 1:54)
{
  resulta2[j]=sum(data.a2$biomass2.ugC[data.a2$ESD2.um >=
classes.um[j] & data.a2$ESD2.um < classes.um[j+1]])
}

# Split-ratio correction - aliquot (1/x) of the second file.
resulta.al2=resulta2*al2

# Next file! again!

a3$BiXos=a3[,length(a3)] ## Last column contains information of
particles classification after validation.
a3.clean=a3[a3$BiXos!="detrit_filamen" & a3$BiXos!="detrito_claro" &
a3$BiXos!="Detrito_claro" & a3$BiXos!="detritos" &
a3$BiXos!="detrito_escuro" & a3$BiXos!="filamentos" & a3$BiXos!="múltiplos"
& a3$BiXos!="Múltiplos" & a3$BiXos!="muda_crustacea" &
a3$BiXos!="gelatinosos_restos" & a3$BiXos!="Muda_crustacea",]
a3.clean$ESD.pixel=a3.clean[,57] ## ESD diameter values, always the same
column at plankton identifier files.
ESD3.um=10.58*(a3.clean$ESD.pixel) ## Conversion from pixels to
micrometers.
esfera3.oblonga=((ESD3.um/10^6)/2)^3)*4/9*pi
biovol3.ml=esfera3.oblonga*10^6
biomass3.ugC=(10^((log10(biovol3.ml)+1.429)/0.808))*1000 # Converts
biovolume to biomass (ugC).
data.a3=data.frame(ESD3.um, biomass3.ugC)

# Attribute each organism to a size interval (size bin):

resulta3=rep(NA, 54)
for (j in 1:54)
{
  resulta3[j]=sum(data.a3$biomass3.ugC[data.a3$ESD3.um >=
classes.um[j] & data.a3$ESD3.um < classes.um[j+1]])
}

# Split-ratio correction - aliquot (1/x) of the third file.
resulta.al3=resulta3*al3

# LOPC tunel area and volume filtered correction. Volume in cubic meters.
```

```
Sum of analysed files.
resulta.corr=(resulta.al1+resulta.al2+resulta.al3)/(dif.LOPC*vf)
}

# Normalization
resulta.corr.norm=resulta.corr/dif.classes.weight
# Log transformation
final=log10(resulta.corr.norm)
# Plot results
log.bins= c(-0.557, -0.457, -0.357, -0.257, -0.157, -0.057, 0.043,
0.143, 0.243, 0.343, 0.443, 0.543, 0.643, 0.743, 0.843, 0.943,
1.043, 1.143, 1.243, 1.343, 1.443, 1.543, 1.643, 1.743, 1.843,
1.943, 2.043, 2.143, 2.243, 2.343, 2.443, 2.543, 2.643, 2.743,
2.843, 2.943, 3.043, 3.143, 3.243, 3.343, 3.443, 3.543, 3.643,
3.743, 3.843, 3.943, 4.043, 4.143, 4.243, 4.343, 4.443, 4.543,
4.643, 4.743)
X11()
par(family="mono")
plot(final~log.bins, pch=20, col=635, cex=1.5, ylab="Log Normalized
Biomass", xlab="Log Biomass Bin", main="NBSS")
return(list(log.bins, final))
}
```

Help

NBSS

package:unknown

R Documentation

~Function to calculate the Zooplankton Normalized Biomass Size Spectra (NBSS) from Plankton Identifier files
- a software used for images automated identification.~~

Description:

~~Automated identification of plankton samples are being performed by different kinds of software for image analysis, and files with large amounts of data are provided. This function extracts the information from organisms' equivalent spherical diameter (ESD), provided in pixels and calculates the NBSS for a sample. It will not take into account the classification of the particles, but it can exclude some categories of non-organisms, such as fibers and detritus. You will have to check the names you gave to these classes during your classification (in your learning set) so this function fits your data. It will convert ESD from pixels to micrometers; attributes each particle to a size class, to estimate its biovolume assuming all particles have the shape of an oblong sphere. Then, biovolume will be converted to

biomass (μgC) according to Wieber et al. (1975): $\log_{10} \text{BV} = -1.429 + 0.808 \log_{10} (\text{Carbon})$. The sum of the biomass in each size class is calculated (the bins were chosen in order to compare to NBSS provided by the Laser Optical Plankton Counter (LOPC) and divided by the class weight (this is the normalization). Results are plotted in log-log scale.~~

Usage:

```
~~NBSS(a1,a2,a3, al1=1, al2=1, vf=1)x~~
```

Arguments:

~a1, a2, a3 data frames, preferentially Plankton Identifier output files (.csv or .txt).
 num.files Number of files to be analysed. You can analyse samples splited up to 3 fractions (3 files).
 al1=1 Integer. Split-ratio factor from your first file (motoda splitter fraction: 1, 2, 4, 8...).
 al2=1 Integer. Split-ratio factor from your second file.
 al3=1 Integer. Split-ratio factor from your third file.
 vf=1 Water volume filtered by the net in cubic meters.
 dif.LOPC Conversion factor to address differences among the net's mouth area and the LOPC tunnel. Only if you wish to compare your samples to LOPC data. Default is 1 - no difference.~~

Details:

~~Plankton identifier files provide ESD values in the 57th collum, and the classification of the particles is in the last collum. You must assure your files have this configuration. You also must check if you have particles classes that you do not want to include in your analysis. If this is true, you must remove then manually from your files or adjust the function to your needs. The classes removed by this function are: "detrito_claro", "Detrito_claro", "detritos", "detrit_filamen", "detrito_escuro", "filamentos", "muda_crustacea", "Muda_crustacea", "gelatinosos_restos".~~

Value:

A plot of the NBSS is provided and also a list:

comp1 : The x axis of the plot - the log biomass bins.

comp2 : The y axis of the plot - the normalized biomass size spectra.

...

Warning:

....

Note:

~~further notes~~

~Make other sections like Warning with \section{Warning }{....}~

Author(s):

~Catarina Marcolin~~

References:

~Wieber PH, Boyd S and Cox JL. 1975. Relationships between zooplankton displacement volume, wet weight, dry weight, and carbon. Fishery Bulletin. 73(4): 777-786.~

See Also:

~~objects to See Also as 'help', ~~~

Examples:

```
a1=read.csv("Analysis_menor_maior_St21.txt", header=TRUE, sep=";")
a2=read.csv("Analysis_St21_maior.txt", header=TRUE, sep=";")
NBSS(a1, a2, al1=16,al2=16, num.files=2, vf=52.98637, dif.LOPC=57.7)
```

Arquivos para testar a função

[analysis_menor_maior_st21.txt](#) [analysis_st21_maior.txt](#)

Plano B

Calcular a distribuição de frequência dos tamanhos (a partir do ESD contido na minha planilha .csv) para cada espécie em cada amostra (2 planilhas) e depois em todo meu conjunto de amostras (várias planilhas).

Comentários

Daniel:

Principal - Proposta muito interessante, pois a implementação parece facilitar muito o trabalho de sua dissertação. No entanto, tome cuidado com a complexidade. Acredito que seja melhor somente

implementar a primeira parte com somente uma matriz e depois, se possível, incluir as análises com a segunda.

Ale: concordo plenamente! Bom trabalho!

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2011:alunos:trabalho_final:catarina:start 

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