

Marcela Vedolin



Graduada em Ciências Biológicas pela Unesp - Bauru. Mestre em Oceanografia na área de Química. Atualmente doutoranda na mesma área pelo Instituto Oceanográfico - USP

Meus exercícios

[Exercícios](#)

[Propostas](#)

Proposta final: função PPI

A função proposta irá produzir uma imagem que irá representar um índice de poluição por pellets de plásticos criado com base na equação proposta por Fernandino et al., 2015: $PPI = [n/a]^p$ Sendo: PPI - Pellet Pollution Index; n - número de pellets encontrados na área; a- área amostrada em m^2 ; p - coeficiente de correlação arbitrariamente determinado como sendo 0,02.

Esta função utilizará ferramentas do pacote: ggplot2.

O objeto de entrada da função deverá conter os dados necessários para o cálculo do PPI (serão vetores numéricos quantitativos denominados com as praias estudadas e o vetor denominado área) a partir do qual será gerada a representação gráfica.

Os resultados serão classificados dentro de um intervalo de valores que representem os índices de poluição de “muito baixo” a “muito alto” com variações de cores em relação ao resultado do índice.

Código da função

```
PPI= function(x, rmNA=TRUE, grafico=TRUE) # Cria a função PPI que aceita os
argumentos data.frame(x), remoção de NAs (default=TRUE) e criação de um
gráfico (default=TRUE)
{
  library(ggplot2) # Carrega o pacote "ggplot2", considerando que já
tenha sido instalado.
  if(rmNA==TRUE) # Se "rmNA= TRUE", a função irá retirar os NAs, caso
tenha nos dados
  {
    dados=(na.omit(x)) # Cria novo objeto com NAs omitidos, caso tenha
    dif=length(x)-length(dados) # Cria novo objeto com os NAs excluídos,
caso tenha
    cat("\t", dif, " valores NA excluídos\n") # Este argumento retorna a
quantidade de NAs que foram excluídos no resultado
```

```
}
else # Caso não tenha sido excluído nenhum NAs,
{
dados=x # os dados originais são mantidos
}
if (grafico==TRUE) # Se "grafico=TRUE", a função irá retornar um
gráfico com os valores do índice plotados, além de retornar os valores em um
objeto da classe vetor dentro do data.frame
{
indice=(x$n/x$area)*0.02 # Cálculo do índice desejado
indice # Resultados dos índices
x$indice=indice #Adicionar os valores obtidos ao data.frame
x$local <- factor(x$local, levels = x$local[order(x$indice)]) # Cria
um fator (local), que irá resultar na plotagem em ordem crescente, de acordo
com índice calculado
str(x) # Confere a estrutura dos dados (com os novos valores)
x11() # Abre o dispositivo gráfico
resultado= ggplot(data=x, aes(x$local, fill=x$indice)) # Plota os
valores referentes aos índices de acordo com cada local. O argumento fill é
referente a escolha de qual fator será preechido, o índice, no caso.
resultado= resultado + geom_bar() # Plota o gráfico de barras
resultado= resultado + scale_fill_continuous(low="#FFFFFF",
high="#333333", breaks=c(8.0,2.5,1.5,1.0,0.5), labels= c("muito
alto","alto", "moderado", "baixo","muito baixo")) # A função
scale_fill_continuous permite o degradê de cores (do branco ao cinza escuro)
e a edição dos intervalos que variam de 0.5 a 8.0 e título para as
legendas, que representam os níveis de poluição
resultado= resultado + xlab("Locais") + scale_y_discrete(name="") +
guides(fill=guide_legend(title="PPI")) # Nomeia os eixos e denomina a
legenda
resultado= resultado + theme(axis.text=element_text(size = 12,
family= "serif"),
panel.background = element_rect(fill="white"), axis.title =
element_text (size =14, family="serif")) # O gráfico terá essas definições
pré definidas com a função theme(): tamanho e fonte dos textos e background
do gráfico na cor branca
resultado= resultado + coord_flip() # Esta função permite que o
gráfico fique na horizontal
ggsave(filename="grafico_ppi.jpg", plot=resultado) #Salva o gráfico
no formato jpeg no diretório
dev.off() # Fecha o dispositivo
}
else # Se "gráfico= FALSE", a função irá retornar apenas os
valores dos índices calculados e inseridos no data.frame
{
indice=(x$n/x$area)*0.02 # Cálculo do índice desejado
resultado=data.frame(x,indice) # Adicionar os valores obtidos ao
data.frame
}
return(resultado) # Retorna resultado
```

```
}  
##Fim da função
```

Help da função

PPI package:unknown R Documentation

Essa função calcula o Pellet Pollution Index, índice que indica o nível de poluição por pellets de plástico em uma determinada área.

Description:

Essa função calcula o Pellet Pollution Index e retorna um gráfico de barras que indica os níveis de poluição por cores e acrescenta uma nova coluna ao data.frame de dados com os valores dos índice calculados.

Usage:

```
PPI= function(x, rmNA=TRUE, grafico=TRUE)
```

Arguments:

x: dados utilizados na forma de data.frame (local, número de pellets e área amostrada)

rmNA=TRUE: retira os NAs dos dados

grafico=TRUE: retorna um gráfico de barras

Details:

Os nomes no objeto de entrada da função (data.frame) devem ser mantidos. Recomenda-se retirar os valores NAs.

Value:

A função retorna um gráfico de barras com os valores dos índices plotados em função dos locais amostrados, o qual as cores escuras representam níveis altos de poluição e as mais claras, níveis mais baixos.

Este gráfico será salvo no formato jpg no diretório de trabalho.

A função também retorna um data.frame que integra os dados originais e os calculados pela função.

Warning:

A confecção do gráfico requer a instalação do pacote ggplot2, caso o pacote não esteja instalado a função não rodará.

Author(s):

Marcela Corrêa Vedolin

(ma.vedolin@gmail.com)

References:

Crawley, M. 2007. The R book. John Wiley & Sons Ltd, England.

Fernandino, Gerson, Elliffa, Carla I., Silva, Iracema R., & Bittencourt, Abílio C.S.P.. (2015). How many pellets are too many? The pellet pollution index as a tool to assess beach pollution by plastic resin pellets in Salvador, Bahia, Brazil.

Revista de Gestão Costeira Integrada, 15(3), 325-332.

Examples:

Dados de entrada: objeto da classe data.frame

```
local= c("Ilha do Cardoso","Guarau","Sonho","Vila
```

```
SP","Gonzaga","Enseada","Santiago","Massaguacu","Praias Grande","Fazenda")
```

```
#Locais (praias) amostrados
```

```
n=c(40,24,37,43,220,113,57,31,62,11) #número de pellets encontrados
```

```
area=rep(0.5, 10) #área amostrada (m2), caso as áreas sejam as mesmas
```

```
recomenda-se usar a função rep()
```

```
x=data.frame(local,n,area) #criar objeto de entrada com a função
```

```
data.frame()
```

```
x #conferir objeto criado
```

Aplicar função: PPI(x)

Arquivo da função

[funcao.r](#)

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2016:alunos:trabalho_final:ma.vedolin:start

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