Propostas Trabalho Final:

Proposta A: Orçamento de navios oceanograficos da Universidade de São Paulo.

O Instituto Oceanográfico da Universidade de São Paulo (IOUSP) opera uma das principais frotas de pesquisa e ensino da América Latina. As embarcações contam com modernos equipamentos e instalações, que permitem seu múltiplo emprego nas mais diversas demandas da Oceanografia, seja na área da biologia, geologia, física ou química. O navio Alpha Crusis, com 64 metros de comprimento e 11 metros de largura, opera com uma tripulação de 19 pessoas e 21 pesquisadores e tem capacidade para deslocar 972 toneladas e uma autonomia (até 70 dias) que permite viagens transoceânicas. Já o Alpha Delphini tem 27 metros de comprimento, pode transportar até 12 pesquisadores e seis tripulantes, e apresenta uma autonomia de 10 a 15 dias. Dada a grande utilização dos dois navios nas pesquisas do IOUSP, e outras universidades, os orçamentos destes dois navios são bem comuns nas propostas de projetos, justificativas das contas dos projetos, etc. Estes orçamentos são bem pesados de calcular sendo que normalmente incluem mais de 30 itens em função do navio a utilizar, o número de dias e do orçamento obtido pelas diferentes agências de pesquisa que podem incluir o preço da mão de obra e/ou do combustível, entre outras.

Tarefa: Criar uma função visando facilitar a tarefa de prestação de contas, orçamentos de projetos, etc. em relação aos navios do IOUSP. Esta função poderá ser utilizada comumente nos diferentes departamentos do IOUSP e também em outras unidades ou Universidades. A função terá argumentos para poder escolher entre os dois navios, número de dias, pesquisadores fora do IOUSP, tipo de financiamento obtido pelas agências de orçamento, etc. A função fornecerá detalhadamente o custo total e o custo por dia.

Proposta B: Analise de biodiversidade em amostras.

O mar profundo é um ambiente difícil de amostrar e a pesquisa relacionada com ele está estreitamente relacionada com a tecnologia. É por isso que obter amostras a partir de métodos padronizados e em habitats iguais ou similares é as vezes impossível. Métodos como a rerefação então não são adequados precisando de outros métodos como medidas de riqueza de espécies.

Tarefa: criar uma função para calcular a riqueza de espécies em amostras. obtidas com diferentes metodologias a partir de índices como o de Margalef ou o de Menhiniek e a geração de um gráfico para a visualização dos datos para cada amostra.

Joan, a proposta A é factível e útil, mas a execução parece simples demais. Se você conseguir incluir loops e colocar argumentos gerais nela, talvez ela se torne mais interessante. A principal fraqueza desta proposta é que ela é bastante específica para as questões financeiras do IO.

Dessa maneira, a segunda proposta parece mais interessante, pois poderia ser mais amplamente utilizada. Não tenho familiaridade com os índices em questão, mas tente brincar com os argumentos utilizados (condições de coleta das amostras por exemplo) e gerar vários gráficos descritivos das amostras analisadas, trazendo mais informações além da riqueza de espécies, se possível. Qualquer dúvida, entre em contato.

—-*Glaucia Del-Rio*

PROPOSTA B MELHORADA: A medição da diversidade de espécies é calculada comumente mediante índices, entre outros métodos. Estes índices muitas vezes não são equivalentes entre eles pois medem a diversidade desde diferentes pontos de vista e dando diferente peso aos componentes dela. É por isso que comumente diversos índices são gerados para explicar a diversidade dentro de um mesmo conjunto de dados ou projeto. A medição da diversidade de espécies mediante índices pode ser dividida em duas grandes categorias (Magurran,1988):

1. **Índices de Riqueza de Espécies:** estes índices mesuram o número de espécies em uma unidade de amostra definida.

1.1. **Índice de Margalef** (Clifford&Stephenson, 1975): Se as medidas amostrais (tamanho e esforço) são iguais ou parecidas. Dmg=(S-1)/lnN

1.2. **Índice de Menhinick** (Whittaker, 1977): Se as medidas amostrais (tamanho e esforço) são iguais ou parecidas. Dmn=S/√N

1.3. **Rarefação** (Hulbert, 1971): medidas amostrais desiguais. Calcula o número de espécies esperado em cada amostra se todas elas estivessem estandardizadas.

2. **Índices baseados na proporção das abundâncias das espécies ou heterogeneidade:** estes índices mesurar a riqueza combinando a riqueza e as abundâncias.

2.1. **Índice de Shannon**: este índice assume que os indivíduos são coletados aleatoriamente de uma população indefinidamente grande. Também assume que todas as espécies estão presentes na amostra. H'= -∑pi lnpi

2.2. **Índice de Simpson** (Simpson,1949): este índice é uma mesura da dominância pois dá mais peso nas espécies com mais abundâncias. 1/D= 1/(∑(ni(ni-1))/(N(N-1)))

2.3. **Índice U de MacIntosh** (McIntosh,1967): McIntosh teorizou que uma comunidade pode ser prevista como um ponto em um hipervolume S dimensional e que a distancia Euclineana da comunidade com o origem pode ser usado como uma mesura de diversidade. U= √ (∑niˆ2)

2.4. **Índice de Berger-Parker** (Berger&Parker,1970): expressa a importância proporcional da espécie mais abundante. d=Nmax/N

2.5. **Índex de Brillouni** (Pielou, 1969): parecido ao índex de Shannon mas mais apropriado para o calculo de comunidades completamente amostrada ou quando a aleatoriedade das amostras não pode ser garantida. HB= (lnN! – ∑ln ni!)/N

TAREFA: realizar uma função para obter facilmente os diferentes índices mais utilizados na medição da diversidade. A função gerará um dataframe com os diferentes índices, ou seja tanto os índices de riqueza de espécies como os baseados na heterogeneidade. A função terá a capacidade de ler tabelas de abundância de espécies de qualquer número de amostras e calcular cada índice para cada amostra (Figura abaixo). A função terá também o argumento para indicar se as medidas amostrais são homogêneas ou heterogêneas para gerar os índices de riqueza de espécies correspondentes.

Exemplo de tabela para ser introduzida na funcão:

Tabela "species"

```
especie site1 site2 site3 site4 site5
```

```
   sp1     9     1     9     0     2
   sp2     3     0     5     2     1
   sp3     0     1     7     3     2
   sp4     4     0     3     1     2
   sp5     2     0     1     0     3
   sp6     1     0     2     2     1
   sp7     1     1     8     5     6
   sp8     0     2     0     2     1
   sp9     1     0     0     1     1
  sp10     0     5     4     2     3
  sp11     1     3     3     2     2
  sp12     1     0     2     1     1
```

Exemplo de um possível dataframe gerado pela função (valores ficticios):

diversity(species)

```
      Index site1 site2 site3 site4 site5
   Margalef   2.3   3.1   2.4   4.1   2.5
  Menhinick   4.2   3.1   5.2   5.3   4.1
    Shannon   2.4   3.2   1.3   4.3   3.5
    Simpson   2.3   3.1   2.4   4.1   2.5
  MacIntosh   1.2   2.2   3.1   2.2   1.5
  Brillouni   4.2   3.1   5.2   5.3   4.1
```

HELP DA FUNÇÃO:

```
diversity                       package:unknown
RDocumentation

Data Input

Description:

    Índices de diversidade α de amostras. Lê tabelas de amostras em formato
data.frame e calcula os principais índices de diversidade α para diferentes
amostras. Os possíveis índices calculados nesta função são Número de
indivíduos (N), Número de espécies (S), Índice de Margalef (Dmg), Índice de
Menhinick (Dmn), Rarefacão,         Índice de Shannon (H'), Índice de
Brillouni (HB), Índice de Simpson (1/D), Índice de McIntosh (U) e Índice de
Berger Parker (d).

Usage:

    diversity(x, size=TRUE,  random=TRUE)

Arguments:

x   dataframe com os dados

size    se TRUE (default) calcula os índices de Margalef e Menhinick. Se
```

```
FALSE calcula rarefação.

random  se TRUE (default) calcula o índice de Shannon. Se FALSE calcula o
índice de Brillouni.

Details:

    Os dados introduzidos na função devem ser um data.frame. Todas as
tabelas devem ter na primeira coluna o nome/código das espécies e o
cabeçalho com o número/código/nome das diferentes amostras.
    As equações dos índices desta função são calculados seguindo a
metodologia proposta em Magurran (1991).

Value:
    Retorna uma matriz onde a primeira coluna mostra os índices calculados,
e o cabeçalho mostra as diferentes amostras respeitando a ordem da tabela
(data.frame) original introduzida na função.

Author:

    Joan Manel Alfaro Lucas
    joan@usp.br

References:

    Magurran, A. E. (1991) Ecological Diversity and its Measurement.
Princeton University Press. Princeton. Pp. 179.

Examples:

>species <- (read.table("species1.txt",head=T))
>species

  Index site1 site2 site3 site4 site5
1   sp1     0     3     2     4     2
2   sp2     4     3     0     5     4
3   sp3     1     0     1     2     1
4   sp4     2     3     0     4     3
5   sp5     2     3     0     0     2
6   sp6     1     2     1     0     0
7   sp7     4     3     1     0     0
8   sp8     2     0     1     4     3

> diversity(species,size=T,random=F)

                           1          2          3          4          5
Individuals (N)   16.0000000 17.0000000  6.0000000 19.0000000 15.0000000
Species (S)        7.0000000  6.0000000  5.0000000  5.0000000  6.0000000
Margalef (Dmg)     2.1640426  1.7647806  2.2324425  1.3584931  1.8463469
Menhinick (Dmn)    1.7500000  1.4552138  2.0412415  1.1470787  1.5491933
```

```
Brillouni (HB)     1.3897694  1.4031252  0.9810173  1.2802676  1.3167603
Simpson (1/D)      8.0000000  8.5000000 15.0000000  5.8965517  7.5000000
McIntosh (U)       0.7681392  0.7765848  0.8932724  0.6983789  0.7587444
Berger Parker (d)  0.2500000  0.1764706  0.3333333  0.2631579  0.2666667

> diversity(species,size=F,random=T)

                           1          2          3          4          5
Individuals (N)   16.0000000 17.0000000  6.0000000 19.0000000 15.0000000
Species (S)        7.0000000  6.0000000  5.0000000  5.0000000  6.0000000
Rarefy             4.3942308  4.3823529  4.3166667  3.8797730  4.2527473
Shannon (H')       1.8195113  1.7823028  1.5607104  1.5723855  1.7140875
Simpson (1/D)      8.0000000  8.5000000 15.0000000  5.8965517  7.5000000
McIntosh (U)       0.7681392  0.7765848  0.8932724  0.6983789  0.7587444
Berger Parker (d)  0.2500000  0.1764706  0.3333333  0.2631579  0.2666667
```

FUNÇÃO:

```
diversity<-function(x, size=TRUE, random=TRUE) ## a função aceita dataframes com os nomes das espécies na primeria columna e as abundâncias delas nas outras columnas. Cada columna tem que ser uma amostra. A função size=TRUE será utilizada quando as amostras tivessem sido coletadas com o mesmo esforço amostral para calcular os índices de Maragalef e Menhinick. Sera utilizado size=FALSE para calcular rarefração quando o esforço amostral seja diferente entre amostras.
{
  N<-as.vector(rep(NA,length=(length(x)-1))) ##Cria um vetor da mesma longitude que o número de amostras (todas as columnas menos a primeira que são as espécies), que será onde ficaram a suma do número de individuos pra cada amostra.
  S<-as.vector(rep(NA,length=(length(x)-1))) ##Cria um vetor da mesma longitude que o número de amostras(todas as columnas menos a primeira que são as espécies), que será onde ficaram a suma do número de espécies pra cada amostra.
  for(i in 1:(length(x)-1)) ##Cria o loop pra calcular a soma de individuos de cada amostra (columna) e colocar a soma no vetor na posição correspondente.
  {
    N[i]<-(sum(x[,i+1])) ## Sumamos as columnas,i+1 serve para pular a columna 1 que são os nomes das espécies. Número total de individuos por amostra.
    S[i]<-((length(x[,1]))-sum(species[,i+1]==0)) ##Restamos os zeros de cada columna a longitude da columna 1 (espécies totais) para calcular o número de espécies em cada amostra e colocamos na posição correspondente do vetor criado (S).
  }
  matrix.base<- as.matrix(x[,2:(length(x))]) ## Criamos uma matriz só com abundâncias de cada espécie pra um manejo mais fácil na hora de calcular os diferentes índices.
  matrix.mcintosh<- (matrix.base)^2 ## Criamos uma matriz e aplicamos a primeira parte da formula do índice de McIntosh.
```

```
  vect.mcintosh <-as.vector(rep(NA,length=(length(x)-1))) ## Criamos um
vector para colocar os resultados da primeira parte do índice de McIntosh.
  for(i in 1:(length(x)-1)) ## Cria o loop para colocar os resultados do
índice de McIntosh no vector criado acima.
  {
    vect.mcintosh[i]<-sqrt(sum(matrix.mcintosh[,i]))  ## Colocamos os
resultados das operações do índice de Brillouni no vector.
    McIntosh<- (N-vect.mcintosh)/(N-(sqrt(N))) ## Criamos o vector com os
resultados finais do índice de McIntosh.
  }
  vect.berger <-as.vector(rep(NA,length=(length(x)-1))) ## Criamos um vector
para colocar os resultados da primeira parte do índice de Berger-Parker.
  for(i in 1:(length(x)-1)) ## Cria o loop para colocar os resultados do
índice de Berger-Parker no vector criado acima.
  {
    vect.berger[i]<- max(matrix.base[,i]) ## Colocamos os resultados das
operações iniciais do índice de Berger-Parker no vector.
    BergerParker<- vect.berger/N ## Criamos o vector com os resultados
finais do índice de Berger-Parker.
  }
  matrix.simpson.1<-(matrix.base)*((matrix.base)-1) ## Criamos uma matriz
onde colocamos os primeiros resulatdos do îndice de Simpson (numeradores da
equação do índice).
  vect.simpson<- (N*(N-1)) ## Criamos um vector onde colocar os
denominadores da equação
  matrix.simpson.2<-as.matrix(x[,2:(length(x))]) ## Criamos uma matriz para
colocar os resultados de dividir o número de individuos de cada espécie em
cada amostra pelo número total de individuos de cada amostra.
  Simpson <-as.vector(rep(NA,length=(length(x)-1))) ## Criamos o vector para
colocar os resultados do índice de Simpson para cada amostra.
  for(i in 1:(length(x)-1)) ## Cria o loop para calcular o resultado de
dividir o número de individuos de cada espécie em cada amostra pelo número
total de individuos de cada amostra.
  {
    matrix.simpson.2[,i]<-(matrix.simpson.1[,i])/(vect.simpson[i]) ##
Aplicamos a equação do índice de Simpson e colocamos os resultados na matriz
criada acima.
    Simpson[i]<-1/(sum(matrix.simpson.2[,i])) ## Colocamos os resultados no
vector criado acima.
  }
  if(size==T) ##Se o esforço amostral é igual pra cada amostra calculamos o
índice de Margalef e Menhinick pra cada amostra.
  {
    Margalef<- ((S-1)/log(N)) ##Índice de Margalef
    Menhinick<- (S/(sqrt(N))) ##Índice de Menhinick
  }
  else ##Se o esforço amostral não é igual entre amostras, então size=FALSE
  {
    MinN<- (min(N)) ## Menor número de individuos de todas as amostras
    FactMinN<-factorial(MinN) ## Factorial do Menor número de individuos de
```

```
todas as amostras
    matrix.rare.1<-
matrix(NA,nrow=(length(x[,1])),ncol=(length(x)-1),byrow=FALSE) ## Criamos
uma matriz das mesmas dimensões que matrix.base para começar fazer as
primeiras operações da rarefaição
    for(i in 1:(length(x)-1)) ## Cria o loop para colocar os resultados das
primeiras operações na matriz criada acima.
    {
      matrix.rare.1[,i]<-(sum(matrix.base[,i]))-(matrix.base[,i]) ## Calcula
o resultado de restar o número de individuos de cada espécie em cada amostra
pelo número total de individuos de cada amostra.
      fact.rare.1<- factorial(matrix.rare.1) ## Fazemos o factorial de cada
factor da matriz criada acima.
    }
    matrix.rare.2<-abs((matrix.rare.1)-MinN) ## Criamos uma segunda matriz
onde colocamos os resultados de restar o menor número de individuos de todas
as amostrasos aos valores da matrix.rare.1 criada acima.
    fact.rare.2<-factorial(matrix.rare.2) ## Fazemos o factorial de cada
valor da mastrix gerada acima.
    matrix.rare.3<- ((fact.rare.1)/((FactMinN)*(fact.rare.2))) ##Criamos a
matriz 3 para colocar os resultados das operações dos objectos criados acima
    denom<- (factorial(N))/((FactMinN)*(factorial((N)-MinN))) ##Calculamos o
denominador da funçao do índice
    matrix.rare.4<-
matrix(NA,nrow=(length(x[,1])),ncol=(length(x)-1),byrow=FALSE) ##Criamos a
matriz 4
    for(i in 1:(length(x)-1))  ## Cria o loop para colocar os resultados das
operações na matriz 4.
    {
      matrix.rare.4[,i:(length(x)-1)]<- ((matrix.rare.3[,i])/denom[i])
##Fazemos as operações do índice com os objectos finais criados.
    }
    matrix.rare.5<-(1-(matrix.rare.4)) ##Criamos a matriz 5 pra colocar os
resultados parciais do índice
    Rarefy <-as.vector(rep(NA,length=(length(x)-1))) ## Criamos um vector
para colocar os resultados da rarefação.
    for(i in 1:(length(x)-1))  ## Cria o loop para colocar os resultados do
índice no vector criado acima.
    {
      Rarefy[i]<-(sum(matrix.rare.5[,i])) ##Vector com os resultados finais
da rarefação.
    }
  }
  if(random==TRUE)
  {
  Shannon<-as.vector(rep(NA,length=(length(x)-1))) ##Criamos o vector para
colocar os futuros resultados do índice para cada amostra.
  list.shannon<-
rep((list(matrix(NA,nrow=(length(x[,1])),ncol=1))),((length(x))-1))
##Criamos uma lista de tantos objetos como amostras e da mesam longitude
delas pra fazer mais simples o possível tratamento de NaNs posteriormente.
```

```
  for(i in 1:(length(x)-1)) ##Criamos o loop para calcular o índice para
cada amostra
  {
    list.shannon[[i]]<-matrix.base[,i] ## Colocamos cada amostra como um
objeto da lista acima criada.
    list.shannon[[i]]<-(list.shannon[[i]])/(sum(matrix.base[,i])) ##
Dividimos cada dado de cada amostra pelo número total de individuos de cada
amostra.
    list.shannon[[i]]<-list.shannon[[i]]*(log(list.shannon[[i]])) ## Fazemos
as operações para calcular o índice.
    list.shannon[[i]]<-na.omit(list.shannon[[i]]) ##Tiramos os NaNs
resultantes de fazer os logaritmos.
    list.shannon[[i]]<--1*(sum(list.shannon[[i]])) ## Terminamos de fazer os
calculos do índice.
    Shannon[i]<-list.shannon[[i]] ##Colocamos os resultados no vector criado
anteriormente.
  }
  }
  else
  {
  matrix.brillouni<-log(factorial(matrix.base)) ## Aplicamos a primeira
parte da formula do índice de Brillouni.
  Brillouni<-as.vector(rep(NA,length=(length(x)-1))) ## Criamos um vector
para colocar os resultados do índice de Brillouni.
  for(i in 1:(length(x)-1)) ## Cria o loop para colocar os resultados do
índice de Brillouni no vector criado acima.
  {
    Brillouni[i]<-(log(factorial(N[i]))-sum(matrix.brillouni[,i]))/N[i] ##
Colocamos os resultados das operações finais do índice de Brillouni no
vector.
  }
  }
  if(size==TRUE&random==TRUE)
  {
  result<-
matrix(data=c(N,S,Margalef,Menhinick,Shannon,Simpson,McIntosh,BergerParker),
nrow=8,ncol=(length(species)-1),dimnames=list(c("Individuals (N)","Species
(S)","Margalef (Dmg)","Menhinick (Dmn)","Shannon (H')","Simpson
(1/D)","McIntosh (U)","Berger Parker
(d)"),c(1:(length(species)-1))),byrow=T)
  return(result) ## Se o esforço amostral é igual entre amostras, a função
irá retornar os índices de Margalef, Menhinick,Shannon,Simpson,McIntosh e
Berger-Parker.
  }
  if(size==TRUE&random==FALSE)
  {
  result1<-
matrix(data=c(N,S,Margalef,Menhinick,Brillouni,Simpson,McIntosh,BergerParker
),nrow=8,ncol=(length(species)-1),dimnames=list(c("Individuals (N)","Species
(S)","Margalef (Dmg)","Menhinick (Dmn)","Brillouni (HB)","Simpson
```

```
(1/D)","McIntosh (U)","Berger Parker
(d)"),c(1:(length(species)-1))),byrow=T)
  return(result1) ## Se as amostras não são coletadas ao acaso, a função irá
retornar os índices de Margalef, Menhinick,Brillouni,Simpson,McIntosh e
Berger-Parker.
  }
  if(size==FALSE&random==TRUE)
  {
  result2<-
matrix(data=c(N,S,Rarefy,Shannon,Simpson,McIntosh,BergerParker),nrow=7,ncol=
(length(species)-1),dimnames=list(c("Individuals (N)","Species
(S)","Rarefy","Shannon (H')","Simpson (1/D)","McIntosh (U)","Berger Parker
(d)"),c(1:(length(species)-1))),byrow=T)
  return(result2) ## Se o esforço amostral é diferente entre amostras, a
função irá retornar rarefação,Shannon,Simpson,McIntosh e Berger-Parker.
  }
  if(size==FALSE&random==FALSE)
  {
  result3<-
matrix(data=c(N,S,Rarefy,Brillouni,Simpson,McIntosh,BergerParker),nrow=7,nco
l=(length(species)-1),dimnames=list(c("Individuals (N)","Species
(S)","Rarefy","Brillouni (HB)","Simpson (1/D)","McIntosh (U)","Berger Parker
(d)"),c(1:(length(species)-1))),byrow=T)
  return(result3) ## Se o esforço amostral é diferente entre amostras e
estas não são coletadas ao acaso, a função irá retornar
rarefação,Brillouni,Simpson,McIntosh e Berger-Parker.
  }
}
```

FUNÇÃO: funca_o.r

HELP DA FUNÇÃO: help1.txt