

Antônio Ralph



Mestrando em Epidemiologia, Faculdade de Saúde Pública, USP. O título de minha dissertação é: “Indicadores ecológicos de mosquitos (Diptera: Culicidae) em parques municipais da cidade de São Paulo”, orientado pelo professor Dr. Mauro Toledo Marrelli.

[exercícios](#)

Minha proposta

Plano A

Minha Proposta é criar uma função que descreva e compare comunidades a partir dos dados amostrais, fornecendo a riqueza observada, estimativa da riqueza total (pode ser o Jackknife 1), índice de diversidade de Simpson (1-D) e similaridade entre as comunidades pelo índice qualitativo de Sorensen.. Os dados contendo as unidades amostrais nas linhas e as espécies nas colunas serão lançados no R. Cada comunidade será representada por um objeto e, obviamente, para que a função também forneça o índice de Sorensen será necessário a presença de, pelo menos, uma segunda comunidade. Os índices de riqueza e diversidade aparecerão nas linhas e as comunidades testadas nas colunas, abaixo uma matriz contendo os índices de similaridade. Estou trabalhando com comunidades de Culicideos (mosquitos) e estou analisando 10 áreas. Imagino que inúmeros pacotes do R contenham funções semelhantes para para o estudo de diversidade, mas ainda sim penso que seria interessante e desafiador (para alguém que esta começando a conhecer o R) construir a função proposta, obtendo com um único comando os indicadores de interesse.

Plano B

O segundo plano é obter uma função que com base nos dados amostrais me forneça um gráfico contendo uma curva de rarefação e seu intervalo de confiança de 95%, com uma lógica igual ao “Sobs Mao Tau” obtido no [EstimateS](#). A curva será formada por pontos com seus respectivos intervalos de confiança, sendo o eixo x representado pelas amostras e o y pelo número de espécies. O último ponto plotado no gráfico será um estimador de riqueza total também contendo o intervalo de confiança.

Comentário - Anna

Oi, Antônio.

Se você acha que vai ser legal o exercício de montar essas funções, ele é factível. Você pode também fazer a sua proposta B junto!

Ale

Tem realmente pacotes e funções que realizam a tarefa que sugere. Isso não é um problema se vc. fizer a sua função a partir do zero. Faça a A e se tiver confortável e com tempo, inclua B! — [Alexandre Adalardo de Oliveira](#) 2013/03/24 11:21

Página de ajuda da função e script

indiv	package:sem pacote	R Documentation
-------	--------------------	-----------------

###Fornece alguns indicadores de diversidade e exibe a curva de rarefação de espécies###

Description:

A função "indiv" fornece o número de indivíduos, riqueza observada, estimador Jackknife 1, índice de Simpson (1-D), matriz de similaridade (índice qualitativo de Sorensen) e curva de rarefação. Tem como objetivo comparar assembléias de um mesmo táxon coletados em diferentes locais com um mesmo esforço amostral, considera-se número de indivíduos.

Usage:

```
indiv(x,nsim=300,replace=F)
```

Arguments:

x: é um objeto do tipo lista (list) contendo matrizes com dados amostrais (amostragens nas linhas e espécies nas colunas)

nsim: número de simulações para cada rodada de reamostragem (default=300)

replace: reposição de indivíduos amostrados, se F=falso (não repõe, default), se T=verdadeiro (repõe)

Details:

o número de colunas (correspondentes às espécies) deve ser o mesmo e a posição das espécies deve corresponder entre as matrizes para permitir a comparação da similaridade (exemplo: spl deve ser a mesma para matriz 1, matriz 2, matriz,3...mesmo que o valor correspondente para esta espécie seja zero em alguma destas matrizes). Recomenda-se nomear cada elemento (matriz) da lista com o um nome que defina cada assembléia para melhor visualização dos resultados.

Caso haja valores faltantes (NA's), estes serão desconsiderados.

É possível analisar apenas uma assembléia desde que esta também esteja no formato "list"

Value:

Indicadores de diversidade: contem o numero de indivíduos, espécies, jackknifel e índice de Simpson (1-D)

matriz de similaridade (ind. qual. Sorensen): matriz contendo o índice qualitativo de Sorensen para cada par de assembléia

curva de rarefação: curva formada com base em simulações dos dados amostrais contendo número de indivíduos coletados no eixo x e

número de espécies no eixo y, fornecendo um indicativo da suficiência amostral.

Warning:

Quanto maior o número de indivíduos coletados maior o tempo para processamento das simulações, pois aumenta-se o número de amostragens possíveis.

neste caso recomenda-se não se elevar o número de simulações (nsim) por amostragem.

Note:

#####

Author(s):

Antônio Ralph Medeiros de Sousa

email: aralphms@usp.br

References:

MAGURRAN, Anne E. Measuring biological diversity. 2004.

LEGENDRE, Pierre; LEGENDRE, Louis. Numerical ecology. Elsevier, 2012.

See Also:

####

Examples:

```
> analise.div=list(ambiente1,ambiente2,ambiente3)
> names(analise.div)=c("ambiente 1","ambiente 2","ambiente 3")
> analise.div
$`ambiente 1`
  sp 1 sp 2 sp 3 sp 4 sp 5 sp 6 sp 7 sp 8 sp 9 sp 10
am 1   9   0   0   5   0   0   0   5   0   0
am 2   0   0   0   9   0   0   9   0   0   0
am 3   0   0   0   0   1   5   2   6  12   0
am 4   0   0   0   0   0   0   0   0   0   0
am 5   0   0   0   0   0  11  11   0   0   0

$`ambiente 2`
```

	sp 1	sp 2	sp 3	sp 4	sp 5	sp 6	sp 7	sp 8	sp 9	sp 10
am 1	11	13	14	11	0	0	0	0	0	0
am 2	0	0	0	12	13	0	0	0	0	18
am 3	0	0	0	11	0	0	0	0	0	0
am 4	10	0	1	14	0	1	0	0	2	0
am 5	0	3	0	0	0	3	0	16	0	0

\$`ambiente 3`

	sp 1	sp 2	sp 3	sp 4	sp 5	sp 6	sp 7	sp 8	sp 9	sp 10
am 1	0	0	0	0	0	0	0	0	11	0
am 2	0	0	0	0	0	0	0	0	13	0
am 3	8	0	12	0	1	0	11	0	14	15
am 4	0	0	0	10	0	15	0	0	9	21
am 5	0	0	0	11	4	0	0	0	0	0

```
> indiv(analise.div,nsim=500,replace=F)
```

```
#####  
#####
```

Função Indicadores de diversidade

```
(indiv)#####
```

```
#####  
#####
```

```
indiv<-function(x,nsim=300,replace=F)
```

```
{  
  individ=t(lapply(x,sum,na.rm=T))##somando o número de indivíduos em cada  
matriz de dados  
  rownames(individ)="Indivíduos"  
  sobs<-function(z)#####função para calcular a riqueza em cada matriz  
de dados  
  {  
    z1=apply(z,2,sum,na.rm=T)  
    riqueza=length(z1[z1>0])  
    return(riqueza)  
  }  
  rich<-t(lapply(x,sobs))####calculando a riqueza na lista de comunidades  
  rownames(rich)<- "Riqueza observada"  
  jack1<-function(z)###função para calcular o jackknife 1 em cada matriz de  
dados  
  {  
    am.x=nrow(z)###calculando número de amostras###  
    am.x2=nrow(z)-1  
    bin.x=z  
    bin.x[bin.x>1]=1###transformando em dados binários##  
    sumbin.x=apply(bin.x,2,sum,na.rm=T)  
    apl=length(sumbin.x[sumbin.x==1])###somando número de espécies que  
aparecem em apenas uma amostra  
    riqobs=length(sumbin.x[sumbin.x>0])###riqueza observda
```

```

    jack=riqobs+(ap1*(am.x2/am.x))###calculando o Jackknife 1####
    return(jack)
}
jackkn1<-t(lapply(x,jack1))###calculando o jackknife nas comunidades
rownames(jackkn1)<-"Jackknife1"
simpson<-function(z)####função para calcular o índice de Simpson
{
  tot.x=apply(z,2,sum,na.rm=T)
  tot.ger=sum(tot.x)
  simp=1-(sum((tot.x/tot.ger)^2))###calculando o índice de simpson
  return(simp)
}
ind.simp<-t(lapply(x,simpson))###calculando o índice de Simpson das
comunidades
rownames(ind.simp)<-"Simpson (1-D)"
cont.spp<-function(z)###função para transformar espécies em dados binários
{
  x1=apply(z,2,sum,na.rm=T)
  x1[x1>0]=1
  return(x1)
}
bin<-as.data.frame(lapply(x,cont.spp))###aplicando a função à lista de
transformando os resultados em um data.frame
bin.sp<-as.matrix(bin)###transformando em uma matriz de dados binários
num.col=ncol(bin.sp)
m.similar=matrix(1,ncol=num.col,nrow=num.col)###matriz para abrigar os
dados de similaridade
for(i in 1:num.col-1)###automatizando a verificação de co-ocorrência de
espécies entre as comunidades###
{
  r=i+1
  for(r in r:num.col)
  {
    co.ocor=sum(bin.sp[,i]>0 & bin.sp[,r]>0)
    tot.sp=sum(bin.sp[,i])+sum(bin.sp[,r])
    m.similar[i,r]=(2*co.ocor)/tot.sp
    m.similar[r,i]=(2*co.ocor)/tot.sp
  }
  mat.similar=round(m.similar,digits=3)
  rownames(mat.similar)=names(x)
  colnames(mat.similar)=names(x)
}
rar=function(z)###função para construir uma curva de rarefação para cada
assembléia amostrada###
{
  w1=apply(z,2,sum,na.rm=T)###somando número de indivíduos de cada espécie
  w2=sum(w1)####somando total de indivíduos coletados
  w5=rep(1,w2)###criando um vetor que repet o número 1 tantas vezes quanto
for o total de indivíduos
  w6=cumsum(w5)##fazendo a soma cumulativa deste vetor(com isto se diz
quantos indivíduos serão coletados a cada simulação)##

```

```
w7=dim(z)[2]##contando o número de colunas de espécies
w8=c(1:w7)##criando um vetor que vai de 1 ao total de colunas
w9 = as.vector(rep(w8[which(w1 > 0)], times = w1[which(w1 >
0)]))###criando um vetor que repita o número da espécie quantas vezes for o
número de indivíduos desta##
resultw=rep(NA,times=length(w5))##criando um vetor para abrigar e
orientar os dados da simulação
aq=rep(NA,times=length(w5))###idem
resultsim<- matrix(NA, nrow=nsim, ncol=length(w5), byrow=T)###matriz de
Na's para guardar os dados das simulações
for(i in 1:nsim)###automatizando o processo de aleatorização de
amostragens começando por 1 indiv. nsim vezes até o total de indiv. nsim
vezes
{
  for(r in 1:length(w6))
  {
    for(k in 1:length(w6))
    {
      aq[k]=w6[r]###informando quantos indivíduos serão retirados da
amostra a cada rodada de simulação
    }
    resultw[r]=length(unique(sample(w9, size=aq, replace)))###somando o
número de espécies de cada amostra
  }
  resultsim[i,]=resultw###guardando o resultado na matriz
}
meddados=t(t(apply(resultsim,2,mean)))###obtendo a média de espécies
para cada valor simulado de tamanho de amostragem
vardados=t(t(apply(resultsim,2,var)))###obtendo a variância
guardaic=matrix(NA,length(meddados),2,byrow=T)###criando matriz para
guardar intervalos de confiança
for(d in 1:length(meddados))###automatizando o processo de calculo do
intervalo de confiança
{
  calcic=meddados[d,]+qt(c(0.025,0.975),df=length(resultsim[,d])-1)*sqrt(varda
dos[d,]/length(resultsim[,d]))###calculo do intervalo de confiança (95%)
  guardaic[d,]=calcic##guardando os valores de IC obtidos
}
w10=t(t(w6))##transpondo a soma cumulativa de indivíduos para coluna
dadosrar=cbind(w10,meddados,guardaic)###reunindo número de indivíduos
amostrados, média e IC's
colnames(dadosrar)=c("índividuos","media de espécies","ic 95% inf","ic
95% sup")
rownames(dadosrar)=paste("am",c(1:length(dadosrar[,1])))
dadosrare=as.matrix(dadosrar)
}
individ2=max(as.numeric(individ))##obtendo o maior valor de indivíduos
amostrados
rich2=max(as.numeric(rich))###obtendo o maior valor de espécies amostrado
x11()
```

```
plot(c(1,individ2),c(1,rich2),xlab="indivíduos amostrados",ylab="nº médio
de espécies",type="n",main="curva de rarefação")##plotando gráfico para
construção da curva de rarefação
  resultrare=lapply(x,rar)###aplicando a função rar para as comunidades da
lista
  for(l in 1:length(resultrare))##automatizando a criação da curva de
rarefação com média e IC's
  {
    tran=resultrare[[l]]
    tran1=as.numeric(tran[,1])###indivíduos
    tran2=as.numeric(tran[,2])###espécies
    tran3=as.numeric(tran[,3])###IC
    tran4=as.numeric(tran[,4])###IC
    tran5=max(tran1)
    tran6=max(tran2)
    lines(tran1,tran2,type="l")
    lines(tran1,tran3,col="red",type="l",lty=3)
    lines(tran1,tran4,col="red",type="l",lty=3)
    text(tran5,tran6,labels=names(resultrare[l]),pos=3)
  }
  index=rbind(individ,rich,jackkn1,ind.simp)##juntando o resultado dos
indicadores
  tudo=list(index,mat.similar)###juntando os indicadores e a matriz de
similaridade em uma lista
  names(tudo)=c("indicadores das comunidades","matriz de similaridade (ind.
qual. Sorensen)")
  return(tudo)###retornando no console todos os valores calculados exceto
curva de rarefação que será plotada em uma gráfico aberto automaticamente##
}
```

[script_da_funcao_indiv.txt](#)

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2013:alunos:trabalho_final:aralphms:start

Last update: **2020/08/12 09:04**

