Guto Hashimoto de Mendonça

Engenheiro Ambiental

Mestrando em Ciências da Engenharia Ambiental.

Universidade de São Paulo - Escola de Engenharia de São Carlos.

Centro de Recursos Hídricos e Ecologia Aplicada

Desenvolvendo projeto de investigação dos efeitos de borda em fragmentos de cerrado stricto sensu.
Orientadora: Giselda Durigan

Propostas

Proposta A:

Estatistica Q

Proposta por Kempton e Taylor, a estatistica Q é uma inovadora e interessante aproximação para medida de diversidade. Baseada na distribuição de abundância das espécies sem necessitar de um modelo ajustado de observações empíricas. Utiliza a curva de abundância de espécies acumuladas e estima a diversidade através da declividade interquantil dessa curva. Na teoria, toda a curva poderia ser utilizada para estimar a diversidade, mas na prática, restringindo a medida aos interquantis garante-se que nem as super abundantes, nem as muito raras, influenciem siginificativamente o resultado.

Comentários

Alexandre: Aqui vc. apresenta uma estatística, imagino que sua idéia seja implementá-la em uma função no R. Acho interessante! Gostaria que postasse na proposta o formato de entrada dos dados e qual será a saída. Boa sorte!

Proposta B:

Jackknifing

Jackknifing é utilizado para se estimar a diversidade, mas também pode ser utilizado para estimar a riqueza de espécies. Foi desenvolvida inicialmente por Quenouille em 1956 e modificada por Tukey em 1958, O método genérico foi descrito por Sokal e Rohlf. A beleza do método é que não se faz nenhuma restrição quanto a distribuição dos dados. Em seu lugar, uma série de “pseudo valores” é

produzida. Essa série de valores é usualmente normal, e suas medias são os melhores estimativas da estatistica.

Comentários

Paulo

Também um ótimo exercício implementar uma função de Jacknife genérica no R!

Página de Ajuda

```
 Qstat                              package:BIE5782-2009
R Documentation


Diversidade entre áreas utilizando a estatística Q


Description:

Produz os gráficos de espécies acumuladas por abundância e adiciona linhas
verticais relativas
aos quartis de 25% e 75% e uma linha ligando os pontos extremos dentro de
intervalo de quartis
delimitado pelas retas verticais. Através da declividade da reta
interquartis é estimada a
diversidade para cada uma das áreas.


Usage:

   Qstat(lista, ...)
Arguments:

  lista: data frame. Constituído pelas espécies na primeira coluna, seguida
por colunas de abundâncias de cada espécie.
  Obs: Espécies não observadas em determinada área deve ser assinaladas com
0.
  ...: Outros argumentos relacionados à função plot,

Details:

Os valores de abundacia do data.frame lista são ordenados em ordem
crescente. Através da
abundância é construído um data.frame de espécies acumuladas para cada área.
```

```
Os valores
de abundancia e de espécie acumuladas são transportados para uma lista a fim de facilitar
a manipulação dos dados. Outra lista é gerada contendo o número de espécies.
Da lista de espécies acumuladas e da de abundância são plotados os gráficos que representam
graficamente os calcúlos da estatística Q.
Dentro de cada gráfico são inseridas retas verticais que representam os quartis de 25% e de 75%.
Outra linha é adicionada ligando os pontos cartesianos (Espécies Acumuladas vs Abundância) referentes
aos quartis de 25% e de 75%.

A declividade da reta corresponde ao valor de diversidade da área. A declividade é calculada
utilizando-se a seguinte equação:

Q(x) = (0.5*r1 + nr + 0.5*r2)/ln(R2/R1)

onde:

Q = diversidade da área x
r1 = número de espécies da classe onde correspondente ao valor do quartil 25%
r2 = número de espécies da classe onde correspondente ao valor do quartil 75%
R1 = número de indivíduos do quartil 25%
R2 = número de indivíduos do quartil 75%
nr = número total de espécies entre os quartis de 25% e 75%

Value:

Para cada área é gerado um gráfico de espécies acumuladas vs abundância. Uma lista contendo os
valores de Q é retornada ao fim da execução da função

Warning:

No caso de não ocorrem observações para determinada espécies no data.frame é necessário que o valor 0
seja assinalado. Valores de NA são automaticamente convertidos em 0.

Caso o número de áreas seja maior que 6, aconselha-se a fazer data.frames diferentes contendo no máximo
6 áreas cada um e rodar a análise para cada um deles, evitando "nublar" a plotagem dos gráficos das área,
pois este são plotados em uma mesma janela.

Author(s):

Augusto Hashimoto de Mendonça
```

```
e-mail: gutohm@gmail.com

References:


See Also:

Examples:

species = paste("especie", 1:29)
abundance =
c(0,0,0,133,63,52,38,37,33,33,29,20,15,15,14,13,11,10,9,8,7,6,6,4,4,3,3,3,2)
abundance2 =
c(170,140,133,63,52,38,37,33,33,29,20,15,15,14,13,11,10,9,8,7,6,6,4,4,3,3,3,
3,2)
abundance3 =
c(170,140,133,63,52,38,37,33,33,29,20,15,15,14,13,11,10,9,8,7,6,6,4,4,3,3,3,
2,2)
abundance4 =
c(170,140,133,62,52,38,37,33,33,0,20,15,15,14,13,11,10,9,8,7,6,6,4,4,3,3,3,2
,2)
abundance5 = round(runif(29,0,170))

lista = data.frame(species, abundance, abundance2, abundance3, abundance4,
abundance5)

Qstat(lista)
```

Código da Função

```
###Função que estima a diversidade de diferentes área através da Estatística
Q,
###a partir de dados de abundância de cada espécie para cada uma das áreas.

###Função desenvolvida por Augusto Hashimoto de Mendonça
###Refenrente ao trabalho final da disciplina BIE5782/2009,
###fornecida pelo Instituto de Biociências da Universidade de São Paulo


Qstat <- function(lista)
{
  dim(lista)
  col = dim(lista)[2]  #captura numero de colunas
  lin = dim(lista)[1]  #captura numero de linhas
  num = rep(0,col)     #cria objeto num onde serao armazenados as o numero
de observacoes de cada area sem repeticao
```

```
###Remove os NA do data.frame de entrada

for(j in 2:col)
{
    for(i in 1:lin)
    {
        if (is.na(lista[i,j])==TRUE)
        {
        lista[i,j]=0
        }
    }
}


####Cria o data.frame de Especies Aculumadas a ser utilizada no loop
  acumul.t = data.frame(lista)
  acumul.t[,1]=seq(1:lin)
  names(acumul.t)=c("Atrib",paste("Area",1:(col-1)))


  for (i in 2:col)
  {
  num[i] = length(unique(lista[,i])) #armazena o tamanho de observacoes de
cada area em um objeto
  lista[,i] = sort(lista[,i], decreasing = FALSE) #Ordena as colunas de
observacoes em ordem crescente
  acumul.t[,i]=rep(0,lin) #Cria o data.frame de especies acumuladas com um
numero de linhas igual ao do data.frame de entrada
  }

  lista.c = lista[,-1] #lista sem a primeira coluna de especies

####Logica de calculo de especies acumuladas
  for(h in 2:col)
  {
  j=1 #indexador auxilar utilizado para o objeto acumul.t
    for(i in 1:lin)
    {
    index=lista[i,h]
    if(index==0) #If para especies nao observadas. Simplesmente acrescenta 0
a numero de especie acumulada e pula para a proxima indexacao
    {
    acumul.t[j]=acumul.t[j]
    j+1
    }
      if (index!=0)
      {
       if(i==1) #Quando o i==1 iniciasse a acumulacao de especies
        {
        acumul.t[1,h]=1
        j=1
```

```
        }
       if(i==lin) #Quando o i igual ao ultimo elemento do data.frame de
entrada:
        {
        o=lista[lin,h]==lista[lin-1,h] #compara o elemento com o elemento
anterior
        if(o=="TRUE") # se os valores forem iguais
        {
        acumul.t[j,h]=acumul.t[j,h]+1 #soma mais um na mesma posicao j
        }
        if(o=="FALSE") #se forem diferentes
        {
        j=j+1
        acumul.t[j,h]=acumul.t[j-1,h]+1 #soma um na posicao j+1
        }
        }
        if(i!=1&i!=lin)
        {
        ####objetos logicos para auxilio do calculo de especies acumuladas
        r=lista[i,h] != lista[i+1,h]&lista[i,h] != lista[i-1,h]
            l=lista[i,h] == lista[i+1,h]&lista[i,h] != lista[i-1,h]
            m=lista[i,h] == lista[i+1,h]&lista[i,h] == lista[i-1,h]
            n=lista[i,h] != lista[i+1,h]&lista[i,h] == lista[i-1,h]
            if (r=="TRUE"|l=="TRUE") #Caso o valor da posicao atual de i
seja diferente da posicao anterior e posterior a ele
        {                        # ou o valor da posicao atual de i seja
igual ao posterior e diferente do anterior
        j=j+1 #soma 1 ao indexador j
            acumul.t[j,h] = acumul.t[j-1,h]+1 #e acrescenta +1 ao objeto de
especies acumuladas
        }
        if (m=="TRUE"|n=="TRUE") #Caso o valor da posicao atual de i seja
igual ao da posicao anterior e posteiror
            {                          #ou o valor da posicao atual de i seja
diferente do posterior e igual ao anterior
        acumul.t[j,h] = acumul.t[j,h]+1 #soma +1 na mesma posicao de j
        }
        }
      }
    }
  }

  ####Cria a lista de especies acumuladas e de abundancia para plotagem e
calculo da inclinacao da reta interquartis
  ind = list(unique(lista[,2]))
  acu.sp = list(unique(lista[,2]))
  for (i in 1:(col-1))
  {
  ind[i] = list(unique(lista[,i+1]))
  acu.sp[i] = list(acumul.t[1:num[i+1],i+1])
```

```
  }


  ####Cria a lista de especies e retorna a posicao do valor dos quantis 25%
e 75% para cada área
  num.esp = list(0)#objeto que guarda o numero de especies com o mesmo valor
de observacao dentro de uma area
  for (j in 1:(col-1))
  {
  num.esp[j] = list(table(lista.c[,j]))
  }

  pos25= rep(0,col-1) #cria objeto que guarda a posicao onde estao alocados
os valores de quartis25% de cada area
  pos75= rep(0,col-1) #cria objeto que guarda a posicao onde estao alocados
os valores de quartis75% de cada area

  for (j in 1:(col-1))
  {
    for (i in 1:num[j+1])
    {
    x = ind[[j]][i]
        if(x<=quantile(lista.c[[j]])[2])
        {
        pos25[j] = pos25[j]+1
        }
        if(x>=quantile(lista.c[[j]])[2])
        {
        pos25[j]=pos25[j]
        }
    }
  }

  for (j in 1:(col-1))
  {
    for (i in 1:num[j+1])
    {
    x = ind[[j]][i]
        if(x<=quantile(lista.c[[j]])[4])
        {
        pos75[j] = pos75[j]+1
        }
        if(x>=quantile(lista.c[[j]])[4])
        {
        pos75[j]=pos75[j]
        }
    }
  }

  ####Logica de plot de Especies acumuladas por abundancia de cada area
```

```
  linhas = floor(col/2) #objeto de auxilio para o mfrow.
  par(mfrow=c(linhas,2))
  for(i in 1:(col-1))
  {
    plot(acu.sp[[i]]~ind[[i]], col="black", main=paste("Area",i),
xlab="Abundancia", ylab="Numero Cumulativo de Especies", ylim =
c(min(acu.sp[[i]]),max(acu.sp[[i]])), xlim = c(min(ind[[i]]),max(ind[[i]])))
    abline(v=quantile(lista[[i+1]])[2],col="blue")
    abline(v=quantile(lista[[i+1]])[4],col="red")
lines(c(ind[[i]][pos25[[i]]],ind[[i]][pos75[[i]]]),c(acu.sp[[i]][pos25[[i]]]
,acu.sp[[i]][pos75[[i]]]), col="orange")
  }
  par(mfrow=c(1,1))



  ####Calculo de Q

  Q = list(0)
  for (i in 1:(col-1))
  {
    r1 = num.esp[[i]][pos25[[i]]]
    r2 = num.esp[[i]][pos75[[i]]]
    R1 = ind[[i]][pos25[[i]]]
    R2 = ind[[i]][pos75[[i]]]
    nr = 0
    for (j in (pos25[[i]]+1):(pos75[[i]]-1))
    {
    nr = nr+num.esp[[i]][j]
    }
    Q[i] = list(round((0.5*r1 + 0.5*r2 + nr)/log(R2/R1),2))
  }
  names(Q) = paste("Area",1:(col-1))
  return(Q)
}
```

Arquivo da Função

Função Qstat

From:
http://ecor.ib.usp.br/ - **ecoR**

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:alunos:trabalho_final:augusto

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