Michelle Cristine

Doutoranda em Ciências Biológicas (Botânica)
Laboratório de Algas Marinhas “Édison José de Paula”
Instituto de Biociências - USP

Tema de pesquisa - Diversidade de Macroalgas verdes formadoras de *turf* de Ilhas Oceanicas Brasileiras.

Meus Exercicios

TRABALHO FINAL

PROPOSTA PRINCIPAL

SuperSumario - SS()

Para análises de concentrações basais de metais em organismos marinhos, é necessário calcular o limite superior que pode ser considerado natural, é calculado através de Me+MAD (mediana mais o dobro dos desvios absolutos da mediana) e o limite inferior por Me15 + MAD15 (mediana dos 15 % + o dobro dos desvios absolutos dos menores valores a proporção de 15%). Além disso, é necessário obter valores estatísticos básicos como o tamanho da amostra (n), a média, o desvio-padrao, valor mínimo e valor máximo, mediana, coeficiente de variação (Chernova e Kozhenkova, 2016). O cálculo de cada um destes parâmetros é relativamente fácil(Com exceção do MAD15 que exige a utilização de alguns IF’s) no R, porém bastante demorado quando temos muitos metais e temos que fazê-los 1 a 1. A função SuperSumário visa reunir todos estes parâmetros em uma única lista e calculá-los para cada metal de forma rápida.

Planejamento da Funçao

Entrada: `SS(x)`

x = Vetor (matriz) com as concentrações que serão utilizadas para os cálculos (classe matriz)

Verificacao

x é uma matriz numéria? caso não seja, retornar mensagem de erro *“O vetor precisar ser uma matriz”*

Pseudo-código

Criar as fórmulas para calcular cada Parametro: `MMAD` = `Mediana + 2*(vetor-mediana)` – valores absolutos (utilizar **`abs()`**)

`Mediana15` – Selecionar os valores que representam **15%** da amostra e analisar a mediana destes valores. **Verificaçao:** Se o valor dos 15% primeiros valores for **igual a 0**, **considerar** a mediana **0** e seguir o calculo.

```
Se o valor for diferente de 0, utilizar este valor para seguir o calculo.
```

`MMAD15` – Calcular o `MAD` para os valores que representam **15%** das amostras. Verificaçao: Se o valor da Mediana for zero, retornar **0**.

`Coeficiente de Variaçao` – Dado pela divisão do desvio padrão pela média multiplicado por 100 = `sd()/mean()*100`

~~Agora que a parte difícil já passou, criaremos a função `SS()`.~~

`SS()` – utiliza a função `apply` para gerar um resultado para cada metal e parâmetro, realizando o calculo para cada um dos parâmetros pedidos na função.

`N` = calcular através de length(vetor 1)
`Media` = calcular através de sum(vetor1)/N
`Desvio_Padrao`= sd(vetor 1)
`Minimo` = min(vetor1)
`Maximo`= max(vetor1)
`Mediana` = median(vetor1)
`MAD` = Calculado como descrito na funcao MAD
`q15`= quantile(vetor1,c(.15)) #Para cálculo da mediana15
`mediana15`= median(q15)
`MAD15`= 2*(q15-median(q15))
`cv` = desvpad/media*100

Salvar os resultados em um vetor = resultados (data.frame).
`Resultado = data.frame(N,Minimo,Media,Mediana,Mediana15,Desvio_Padrao,Maximo,MAD,MAD15,Coeficiente_Variacao)`

Comentários Lucas Freitas

Apesar de ser razoavelmente simples a sua proposta se enquadra dentro do que está sendo pedido Dentre as duas propostas eu sugeriria essa uma vez que é mais desafiadora. No entanto aqui seguem algumas considerações:

1. É preciso identificar como os dados de input estão organizadas. Cada coluna é um metal e cada linha é referente a uma concentração deste?

2. Sugiro que todas as estatísticas básicas que serão calculadas sejam obtidas antes do cálculo dos limites inferiores e superiores (uma vez que a mediana é necessária no cálculo desta) para não ficar repetindo variáveis

3. Vc colocou que o resultado final seria um vetor contendo as estatísticas. Porém como essas estatísticas estariam repetidas para cada metal? Não seria melhor fazer uma matriz?

4. Uma vez que o seu resultado final visado é uma matrix ou data frame com cada linha/coluna sendo um tipo de metal e cada coluna/ linha sendo uma estatística básica associada a este .Seria interessante que esse resultado possua cada metal nomeando sua respectiva linha/coluna e cada estatística básica nomeando sua respectiva coluna/linha.

Resposta Michelle

Boa Noite Lucas,

Obrigada pelas dicas e sugestoes.
Vou seguir trabalhando na proposta Principal.

PROPOSTA ALTERNATIVA

Índices de Complementariedade com dados de Presenca e Ausencia (Diversidade B)

Quando analisando comunidades de macroalgas, nem sempre é possível obtermos dados quantitativos, sendo então gerados dados de presença (1) e ausência(0) de cada espécie para os pontos amostrados. Esse fato gera dificuldades em comparar estes locais de coleta e gerar índices de diversidade. Uma forma de resolver isso é utilizando índices de similaridade/dissimilaridade. Índices, antigos e bastante uteis são: ''Sorensen/Jaccard/Bsim''. Cada um com suas particularidades. Desta forma a função ''simi()'' tem visa estimar com sucesso os 3 índices para uma dada amostra de presença e ausência (Vianna,2013).

Jaccard
é definida por $C_J a/(a+b+c)$
Sorensen
é definida por $C_S = 2a/(2a+b+c)$
BSim
é definida por $B_{sim} = 1-[a/(a+min(b,c)$

Onde:

a = numero de espécies nas duas parcelas
b = numero de espécies apenas na parcela 1
c = numero de espécies apenas na parcela 2

VERIFICACAO

Verificar se é um vetor numérico, caso contrário retornar "`O vetor precisa ser numérico`".

Verificar se o comprimento dos vetores b e c é o mesmo, caso contrario retornar a mensagem: "`Os vetores precisam ter o mesmo comprimento`".

PSEUDO CÓDIGO:

Criar a função Jaccard:
Criar o Objeto `Both` com um vetor que apresente as espécies que aparecem nas duas parcelas.
Criar `objeto` B com vetor de soma das espécies exclusivas de B
Criar `objeto` C com o vetor de soma das espécies exclusivas de C
Aplicar a fórmula do índice de Jaccard.
''$C_J a/(Both+B+C)$''

Criar a função Sorensen:
Criar o Objeto `Both` com um vetor que apresente as espécies que aparecem nas duas parcelas.
Criar `objeto` B com vetor de soma das espécies exclusivas de B
Criar `objeto` C com o vetor de soma das espécies exclusivas de C
Aplicar a fórmula do índice de Sorensen.
''$C_S = 2Both/(2Both+B+C)$''

Criar a função BSim
Criar o Objeto `Both` com um vetor que apresente as espécies que aparecem nas duas parcelas.

Criar `objeto` B com vetor de soma das espécies exclusivas de B
Criar `objeto` C com o vetor de soma das espécies exclusivas de C
Aplicar a fórmula do índice de Bsim.
''B_{sim} = 1-[Both/(Both+min(B,C) ''

Criando ~~finalmente~~ a função ''Simi()''
Argumentos: **dados** – `data.frame` \ **índice** = `Jaccard,Sorensen,Bsim`
Return (**`índice`**).

Comentários Lucas Freitas

Novamente apesar de ser razoavelmente simples a sua proposta se enquadra dentro do que está sendo pedido. No entanto aqui seguem algumas considerações:

1. Na checagem de dados é necessário que o vetor além de ser numérico seja uma matriz de duas colunas (ou duas linhas) e o mesmo deve ser especificado nas documentações referentes a função.

2. Como A B e C serão os mesmos para idependente do índice seria interessante se fossem calculados como podendo ser usados em qualquer um dos cenários.

3. Deste modo o seu teste lógico que decide qual índice deve ser calculado apresentaria apenas o cálculo do índice dentro dele

4.O resultado calculado em cada índice pode ser "guardado numa variável de mesmo nome, uma vez que só um deles será executado.

5.Seria interessante que além do índice calculado o resultado retorne o método utilizado

Peço desculpas se entendi algo errado . Qualquer coisa estou a disposição.

Email: rodriguesdefreitaslucas@gmail.com Boa sorte.

Links Trabalho Final

Trabalho Final

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2019:alunos:trabalho_final:mcristine:start

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