

Dylan Padilla



Mestrando em Ecologia e Evolução pela Universidade Federal de São Paulo (UNIFESP). Tenho interesse na área de ecologia evolutiva, atuando em pesquisas que visam entender o comportamento animal e modelar a evolução da coloração em répteis.

<https://www.facebook.com/labanimal15/>

Exercícios resolvidos durante a disciplina

[Meus exercícios](#)

Trabalho final

- **Plano A:** função `diver()`

Introdução

Ao entrarmos no Antropoceno, a humanidade está reorganizando a biosfera. Processos que vão desde a caça excessiva de grandes vertebrados e a deslocação de espécies exóticas através das barreiras biogeográficas até à eliminação indiscriminada de habitats naturais para a agricultura continuam a corroer a biodiversidade. A sociedade valoriza a biodiversidade em múltiplas escalas espaciais, com preocupações que variam desde a provisão local de serviços ecossistêmicos até a preservação global do valor intrínseco e instrumental das espécies. Por exemplo, a Convenção das Nações Unidas sobre Diversidade Biológica procura deter a perda de biodiversidade a nível regional, nacional e global. Para avaliar a melhor forma de conservar a biodiversidade em escalas espaciais, precisamos entender a relação entre dados de monitoramento coletados localmente e dinâmicas de diversidade regional e como os mecanismos que mantêm a diversidade variam de escalas espaciais locais a regionais. Os conservacionistas, portanto, enfrentam uma questão fundamental: como as mudanças medidas localmente aumentam até os impactos regionais?

A solução está na nossa compreensão da diversidade alfa, beta e gama. Portanto, o desenvolvimento de uma ferramenta que possa quantificar essas propriedades das comunidades e ecossistemas se torna muito importante.

A importância central desta função reside no cálculo dos “números efetivos” ou “números equivalentes” dos índices de diversidade convencionalmente usados (shannon entropy ou simpson concentration) para estimar riqueza de espécies. Os números equivalentes de um índice de diversidade correspondem ao número de elementos igualmente prováveis necessários para produzir o valor dado do índice de diversidade. Os números equivalentes do índice de Shannon e o Índice de Simpson são calculados da seguinte forma:

- $Shannon = \exp(-\sum p_i \times \ln(p_i))$
- $Simpson = 1/(\sum \ln(p_i)^2)$

onde:

- p_i = proporção da espécie i relativa ao numero total de indivíduos da amostra.

Para cada medida dos anteriores indices se estimará a incerteza por meio de reamostragem com reposição (BOOSTRAP). Assim, os intervalos de confiança ao 95% serão calculados por cada valor observado do índice.

A diversidade beta será calculada usando o índice de similaridade de Bray-Curtis:

- $(A+B-2*J)/(A+B)$

onde:

- A = numero de especies in j
- B = numero de especies in k
- J = espécies compartilhadas

Porfim, a diversidade gama será calculada como o numero total de especies da paisagem em estudo.

Planejamento da função

- pi = matriz contendo as amostras nas colunas e as espécies nas linhas (classe `matrix()`).
- $alfa$ = índice de diversidade a ser usado, fica a escolha do usuário. Pode ser o exponencial do shannon ou o inverso do simpson.
- $beta.gam$ = diversidade beta calculada por meio da equação definida na introdução (o default é TRUE) e calcula também o valor da diversidade gama.
- $perm$ = número de permutações usado na reamostragem para estimar os intervalos de confiança de cada valor do índice calculado no argumento $alfa$.

Verificando os parâmetros

- pi deve ser um objeto da classe `matrix`, pode conter 0 ou NA's.
- $alfa$ pode ser "shannon" ou "simpson", para calculá-los deve-se especificar alguns deles. Pode-se usar letras iniciais de cada índice.
- $beta.gam$ se TRUE, será calculado. Se FALSE, não será calculado.
- $perm$ deve ser um numero inteiro (classe `integer`).

Pseudo-código

1. Indica que os indices dentro do parametro $alfa$ poderão ser especificados por meio das letras iniciais.
2. Cria o objeto gam e calcula o valor da diversidade gama
3. `if(alfa=="shannon" | alfa=="simpson")`, cria objeto shannon ou simpson e calcula o exponencial do shannon ou o inverso do simpson para cada coluna de pi .
4. Cria um objeto res e constrói uma matriz preenchida de NA's cujas colunas correspondem as amostras e as linhas correspondem ao numero de permutações usado para calcular os intervalos de confiança.

5. Entra em um ciclo for (`i in 1:perm`).
6. Aleatoriza-se as linhas da matriz `pi` com reposição.
7. Aplica-se o índice definido pelo usuário toda vez que a matriz seja aleatorizada.
8. Guarda-se os valores do índice nas linhas da matriz `res`.
9. Fecha o ciclo
10. `if(beta.gam==TRUE)`:
11. Entra em um ciclo for (`i in 1:(dim(pi)[2]-1)`)
12. Entra em um novo ciclo for (`j in (i+1):dim(pi)[2]`)
13. Calcula diversidade beta por cada par de amostras.
14. Imprime os cálculos em um objeto de classe `matrix` criado antes do ciclo, onde o `upper.tri` corresponde aos cálculos da diversidade beta.
15. Fecha o ciclo.
16. Cria vários loops usando o `if` para maior flexibilidade da função.

Saída

A função retorna uma lista contendo os seguintes objetos:

1. `shan` ou `simp`: Valores do exponencial do índice shannon ou o inverso do simpson por cada amostra (colunas) de `pi`.
2. `confidence.inter`: Os valores dos intervalos de confiança ao 95%.
3. `beta`: Um objeto da classe `matrix` contendo os valores de diversidade beta por pares de amostras.
4. `gam`: O valor da diversidade gama.
5. Gráfico de diversidade alfa ~ amostras

Bibliografia

- Hill, M. 1973. Diversity and evenness: a unifying notation and its consequences. *Ecology* 54:427-432.
- Jost, L. (2007) Partitioning diversity into independent alpha and beta components. *Ecology*, 88, 2427-2439.
- Socolar, J. B., Gilroy, J. J., Kunin, W. E., & Edwards, D. P. (2016). How should beta-diversity inform biodiversity conservation?. *Trends in Ecology & Evolution*, 31(1), 67-80.

Oi Dylan,

A proposta A parece interessante, mas não consigo entender o que há de original além de chamar o pacote `vegan` pra fazer as partes interessantes. Do jeito que está, parece que a função só facilita que o usuário escolha o tipo de análise, mas o trabalho é feito pelas funções do `vegan`. Isso não é o objetivo da função a programar.

De maneira alternativa, eu sugiro manter o tópico sobre estimadores de diversidade mas tentando propor uma implementação de algum estimador para alguma escala de diversidade, e talvez se desafiar tentando implementar estimação de incerteza sobre a mesma por meio de alguma ferramenta (e.g., reamostragem).

Dito isto, você deve propor uma modificação radical da proposta A para poder continuar com ela como trabalho final. Por favor lembre-se que o aceite da proposta B é obrigatório, então, veja meus comentários sobre a mesma na seção a seguir. Próximas conversas terão lugar pelo email e uma vez a gente consiga um consenso você deverá fazer as mudanças que tenham lugar no seu wiki.

—*Gustavo A. Ballen*

- **Plano B:** função `darwin()`

Introdução

A idéia básica da evolução biológica é que populações de espécies de organismos mudam com o tempo. Hoje, quando pensamos em evolução, é provável que vinculemos essa ideia a uma pessoa específica: o naturalista britânico Charles Darwin.

Na década de 1850, Darwin escreveu um livro influente e controverso chamado *A origem das Espécies*. Nele, sugeriu um mecanismo para a evolução: a seleção natural, na qual traços hereditários que ajudam os organismos a sobreviver e se reproduzir se tornam mais comuns em uma população ao longo do tempo.

Existem vários tipos de seleção como a seleção direcional, seleção disruptiva, seleção estabilizadora, entre outros. Por meio do uso da função `darwin` será possível simular só o primeiro tipo de seleção. Este é um dos tipos mais frequente de seleção no qual o ótimo do traço é deslocado por uma pressão de seleção (p.e., predação).



Planejamento da função

- `N` tamanho populacional (classe `integer` ≥ 0)
- `optimal` ótimo do traço (classe `integer` ≥ 0)
- `pressure` intensidade de seleção (classe `integer` ≥ 0)
- `g` tempo em gerações (classe `integer` ≥ 0)

Verificando os parâmetros

- `N`, `optimal`, `pressure`, `g` ≥ 0 caso contrario (< 0), escreve: "O valor do parametro deve ser positivo"

Pseudo-código

1. Cria uma matriz de NA's com `nrow = N` e `ncol = g`
2. Cria a condição inicial da população usando `rnorm()`
3. Põe a condição inicial da população na primeira coluna da matriz
4. Gera um ciclo usando `for (i in 2:g)`

- Calcula a probabilidade de sobrevivência dos indivíduos da condição inicial
 - Simula eventos de reprodução aleatórios dos indivíduos que sobreviveram
 - calcula o valor do traço dos filhotes (ótimo)
 - Põe o valor do traço de cada indivíduo na posição i da matriz
 - Fecha o ciclo
5. Retorna a matriz preenchida contendo as gerações em colunas e os indivíduos nas linhas.
 6. Faz um `boxplot` no qual y = mediana do traço (ótimo) da população e x = tempo `seq(1, g)`.

Saída

A função retorna:

- Uma matriz com o resultado da seleção natural ao longo de gerações
- Um grafico mostrando o deslocamento do ótimo ao longo do tempo

Observações

Parte da ideia desta função se originou como resultado de um curso de simulações empregado pelo Danilo Muniz na UNIFESP.

Bibliografia

- Losos, J. B., Schoener, T. W., & Spiller, D. A. (2004). Predator-induced behaviour shifts and natural selection in field-experimental lizard populations. *Nature*, 432(7016), 505.

Oi Dylan,

A proposta A parece interessante, mas não consigo entender exatamente qual modelo quantitativo estaria sendo implementado aqui. Por favor desenvolva em maior detalhe sua idéia e talvez possa continuar com ela como plano B. Neste ponto, as duas propostas parecem interessantes mas precisam de um pouco de trabalho.

Próximas conversas terão lugar pelo email e uma vez a gente consiga um consenso você deverá fazer as mudanças que tenham lugar no seu wiki.

—[Gustavo A. Ballen](#)

Links para o trabalho final

De acordo com as sugestões do monitor/orientador, resolvi seguir com o plano A. Trata-se da função `diver`, a qual estima a diversidade biológica em diferentes escalas.

- Link para a pagina da função: [diver](#)
- Link para a pagina do help da função: [help](#)

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:dylan.padilla94:start



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