

Pâmela Friedemann



Doutoranda em Ecologia, no Instituto de Biociências da USP. Faço parte do [Guimarães Lab](#).

O título da minha tese (até agora) é: 'A evolução de frugivoria em Thraupidae e Tyrannidae: correlatos biológicos e história evolutiva'.

Lattes: <http://lattes.cnpq.br/6244763451022215>

[Meus exercícios](#) 😊

Proposta de trabalho final

Proposta A

Em uma das diversas definições o Antropoceno pode ser definido como o período em que o Homo sapiens tem se tornado uma força evolutiva dominante (Pievani, 2014), no qual observamos que as atividades humanas (como caça, desmatamento, fragmentação de habitat, etc) são as principais fontes do declínio do número de indivíduos e/ou espécies (Dirzo et al., 2014). Apesar deste padrão ser bastante generalizado, onde há declínio na abundância ou riqueza de espécies com o aumento de ações antropogênicas, as respostas a estas ações são diferentes de acordo com o grupo de organismos ou regiões avaliados. Neste contexto, é importante visualizar e compreender como as espécies respondem às alterações da paisagem em um gradiente de distúrbio. Sendo assim, após realizar o levantamento da diversidade do grupo alvo em paisagens com diferentes graus de distúrbio, o primeiro passo da análise exploratória dos dados coletados seria visualizar como a abundância e a riqueza de espécies se apresenta em cada uma das paisagens. A partir de um data frame com o número de indivíduos por espécie em cada uma das paisagens podemos visualizar e acessar quais paisagens apresentaram um maior número de espécies e indivíduos, e partir então para as análises de interesse. Assim, o objetivo desta função é possibilitar uma primeira visualização uma vez que os dados foram coletados. Esta função poderá ser utilizada considerando a abundância e espécies em paisagens com diferentes níveis/quantidade/intensidade de distúrbio.

Planejamento da função

Nome da função: explor.an

Entrada: explor.an (x, y, tabela=TRUE, graph=TRUE)

Parâmetros:

x: data frame com os dados do número de indivíduos por paisagem, onde as espécies estão nas colunas e as paisagens estão nas linhas. Exemplo abaixo:



y: data frame com as informações quantitativas das paisagens. Exemplo abaixo:



tabela: parâmetro lógico. Se "TRUE" a função cria um arquivo csv no diretório de trabalho com o resumo do data frame com as informações dos números e proporções dos indivíduos e espécies.

graph : parâmetro lógico. Se "TRUE" a função retorna o gráfico de distribuição das espécies ao longo do gradiente da variável ambiental.

Verificando os parâmetros:

x: é um data frame, contendo somente o nome das parcelas, das espécies e os número de indivíduos, dados das classe "integer". Se não, retorna mensagem de erro.

y: é um data frame contendo somente o nome das parcelas, das variáveis e os dados quantitativos de cada paisagem para cada variável, dados da classe "numeric". Se não, retorna a mensagem de erro.

Saída: A função permitirá que a usuária tenha acesso a um data frame com a abundância total, riqueza de espécies, e as respectivas proporções, de cada paisagem, e se @ usuári@ desejar pode ser gerado um arquivo csv com estas informações. A função também gera um gráfico da distribuição das espécies ao longo do gradiente da variável ambiental. Segue abaixo uma imagem de exemplo do gráfico:



Pseudo-código:

1. if Verificar se os parâmetros da função estão sendo obedecidos, e avisar caso não estejam.
2. data.frame Criar o data frame onde serão registradas as informações do número total de espécies e indivíduos, bem como as respectivas proporções por paisagem
3. rowSums Somar o número indivíduos por paisagem, no qual a soma é feita por linha
4. data.frame[i] Guardar no data frame a informação do número de indivíduos total por paisagem
5. soma n^o de espécies/paisagem A cada linha (paisagem), soma-se as espécies que têm valores diferentes de zero nas colunas
6. n^o indivíduos na paisagem/sum total de indivíduos Calcular as proporções de indivíduos por paisagem
7. data.frame[i] Guardar no data frame a informação das proporções de indivíduos por paisagem
8. n^o spp na paisagem/sum total de spp Calcular as proporções de espécies por paisagem
9. data.frame[i] Guardar no data frame a informação das proporções de espécies por paisagem
10. if Caso o parâmetro graphic=TRUE, plotar um histograma da seguinte maneira:
11. hist Histograma com as diferentes paisagens no eixo x e o número de indivíduos no eixo y, e com o valor da proporção representado em cada barra
12. hist Histograma com as diferentes paisagens no eixo x e o número de espécies no eixo y, e com o valor da proporção representado em cada barra
13. return Finaliza a função, retornando o data frame e os dois histogramas.

Proposta B

No mundo acadêmico, enquanto passamos pelo doutorado, é comum termos como modelos outras pesquisadoras que já passaram pelas etapas as quais estamos passando. Apesar de cada pessoa ter uma personalidade e trabalhar de maneiras específicas, em geral é importante organizar o tempo disponível para o desenvolvimento do trabalho mantendo metas a serem cumpridas e para determinar estas metas podemos nos espelhar em pesquisadoras mais experientes. A rotina de uma pesquisadora no doutorado têm alguns aspectos que são comuns a toda estudante, no qual temos que pensar no tempo a ser investido em cada atividade: escrita, análise de dados, estudo (leitura de artigos e livros), reunião de laboratório e reunião com orientadora (ou outras reuniões). Sendo assim, baseado na rotina semanal de cada pós-graduanda seria interessante saber quantas horas semanais teríamos de passar em cada tipo de atividade (escrita, estudo, e outras) para conseguirmos finalizar o doutorado com êxito tendo boas pesquisadoras como inspiração.

Sendo assim, o objetivo desta função é determinar quanto tempo precisamos trabalhar diariamente para finalizar o doutorado com o máximo de êxito possível (mais artigos publicados e/ou publicação em revistas com alto fator de impacto, ou qualquer outro parâmetro), considerando o tempo que pesquisadoras experientes trabalham semanalmente.

Planejamento da função

Nome da função: meta.dout

Entrada: meta.dout (x=5, y="dat", z=12.50)

Parâmetros:

x: data frame com o número de horas semanais que a usuária está investindo nas atividades relacionadas à pesquisa atualmente e as horas que suas "pesquisadoras modelo" investem em cada atividade. Exemplo:



y: média do número de horas que são direcionadas para outras atividades diárias (atividades domésticas, refeições, descanso, atividade física, terapia, curso etc)



Verificando os parâmetros:

x: é um data frame. Se não, escreve: "vetor não é um data frame"

y: é um data frame. Se não, escreve: "vetor não é data frame"

Saída: A função retorna o número de horas diárias que a usuária precisa trabalhar em cada atividade ligada a sua pesquisa considerando o número de dias disponíveis na semana para esse fim, o número de horas semanais trabalhadas pela pesquisadora modelo e o número de horas diárias direcionadas para outras atividades. Também retorna duas possibilidades para conseguir cumprir essa meta. E caso, em alguma das atividades a usuária estiver investindo menos que 2 desvio padrão do número investido pelas pesquisadoras experientes aparecerá um aviso (exemplo: "Investir mais tempo na escrita, se possível ;)")

Pseudo-código

1. `if` Conferir se os parâmetros da função estão sendo obedecidos e avisar caso não estejam `stop`
2. `mean` Calcular a média de tempo usado pelas pesquisadoras experientes em cada atividade
3. `sd` Calcular o desvio padrão para cada uma das categorias
4. `if` Se na categoria 1 (linha 1 no data frame, exemplo: estudo (artigo/livro)) o valor da usuária for menor que 2 desvio padrão (ou 1, não sei direito, pensar nisso!) das outras pesquisadoras, aparece a mensagem `cat` (“Investir mais tempo na leitura”)
5. `if` Se na categoria 2 (linha 2 no data frame, exemplo: análise se dados) o valor da usuária for menor que 2 desvio padrão (ou 1, não sei direito, pensar nisso!) das outras pesquisadoras, aparece a mensagem `cat` (“Já está na hora de analisar seus dados? Procure investir mais tempo”)
6. `if` E assim por diante para cada categoria avaliada!
7. `return` Retorna a função com as mensagens de aviso.

Comentários da Débora Brandt

Pâmela,

Na proposta A, o que você pretende fazer com o 2o input (y)? A proposta A está um pouco simples, mas se você for fazer alguma coisa a mais usando o objeto y, talvez seja o suficiente. Favor detalhar um pouco mais essa proposta para nós podermos avaliar se é o suficiente.

Na proposta B, o pseudo-código também não descreve como você vai usar o segundo argumento (y), apesar de você indicar o que vai fazer com ele no item “Saída”. Uma outra dúvida: a função só vai aceitar 2 pesquisadoras-modelo, ou pode ter mais?

A proposta A parece um pouco mais interessante, então sugiro seguir com ela, mas descrever melhor como você vai usar o 2o dataframe do input.

Comentários Pâmela:

Oi Débora! Obrigada pelos comentários. Entendi! Assim, realmente deixei a desejar na explicação do que input y da função 'A' vai adicionar. Minha ideia é que gere um histograma + distribuição das espécies (abundância), adicionei uma imagem de exemplo na parte da 'saída' da função. Tu acha que ainda é muito simples? Eu gostaria que fosse uma função que desse uma primeira ideia desse tipo de dado (onde a gente tem gradientes de valores de uma variável e o número de spp de cada paisagem),

apresentando um resumo das informações e como as spp se distribuem nas paisagens, de forma gráfica.

Tem algum outro meio para eu te enviar dúvidas pontuais? Estou começando a fazer a função A, mas algumas coisas já estão gerando dúvidas. Se tu puder me passa o teu e-mail. ;)

Comentários da Débora Brandt

Pâmela,

Pode me escrever: deboraycb@gmail.com A saída gráfica que você usou de exemplo faz algo que parece tornar sua função um pouco mais interessante do que plotar um simples histograma: a figura ordena as paisagens de acordo com a variável ambiental plotada, e imagino que isso sirva para avaliar se a abundância das diferentes espécies é correlacionada com certas variáveis ambientais. Talvez você possa explorar um pouco essa ideia para complementar sua função.

Trabalho final: função 'explor.an'

A função que eu desenvolvi a partir da **proposta A**, e com algumas mudanças realizadas ao longo da criação da função, realiza um conjunto de análises exploratórias a partir dos seguintes tipos de dados:

- Uma matrix de dados com o número de espécies por paisagem
- Uma matrix do gradiente ambiental das paisagens com dados numéricos de cada variável ambiental

Página de ajuda: [Help](#) 😊

Função: [explor.an](#) 😊

A função resulta um resumo com o número total de indivíduos e espécies, bem como as proporções dos mesmos. A função também resulta um gráfico das distribuição das espécies ao longo do gradiente da variável ambiental selecionada pelo usuário.

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

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

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

