



Brenda

Mestranda em Ecologia no Instituto de Biociências da Universidade de São Paulo, Laboratório de Ecologia da Paisagem e Conservação, sob orientação da Prof^a Rozely F. dos Santos.

[Meus exercícios](#)

Trabalho Final: Propostas

Plano A: Análises ecológicas para diagnósticos de fauna

Contextualização e Objetivo

O diagnóstico de fauna a partir de dados coletados em campo é uma atividade muito recorrente em estudos de impacto ambiental com fins de licenciamento e, então, muito aplicados em empresas de consultoria ambiental. Normalmente, os dados brutos coletados em campo são planilhados para, posteriormente, serem analisados. Algumas análises ecológicas básicas são recorrentes em praticamente todos os diagnósticos de fauna, como:

- Abundância relativa das espécies
- Diversidade de Shannon
- Equitabilidade de Pielou

Em geral, estas análises são realizadas por meio do uso de ferramentas como Microsoft Office Excel ou softwares de análise de dados como PAST¹ ou EstimateS². Apesar de serem ótimas ferramentas, as mesmas possuem limitações de análises ou demandam um significativo esforço para sistematização dos dados em diferentes formatos de input.

Assim, a criação de uma função no R que permita as análises acima apresentadas, juntamente com a confecção de um gráfico de abundância relativa das espécies, tornará mais fácil, ágil e prática as análises ecológicas frequentemente presentes nos diagnósticos de fauna.

Planejamento da função

Entrada:

O input dos dados será feito por meio de um dataframe contendo minimamente uma coluna com a informação de nome científico e linhas preenchidas com os registros individuais (nomes científicos das espécies às quais os espécimes registrados são pertencentes). Além da coluna de nome científico poderá haver outras colunas com informações diversas e variáveis a cada estudo, as quais não entrarão na função proposta. Como exemplo:



Verificando os parâmetros:

Nome científico é um objeto composto por caracteres? Se não, retorna: "Nome Científico deve ser um objeto composto por caracteres".

Pseudocódigo:

1. Calcular riqueza total da comunidade amostrada;
2. Calcular abundância total da comunidade amostrada;
3. Calcular a abundância por espécie;
4. Graficar a abundância relativa das espécies;
5. Calcular o Índice de Diversidade de Shannon, considerando todas as espécies da comunidade;
6. Calcular o Índice de Equitabilidade de Pielou, considerando todas as espécies da comunidade;
7. Apresentar os resultados de Abundância e Riqueza total e dos índices de Diversidade e Equitabilidade em um dataframe.

Saída:

- Gráfico de abundância relativa das espécies;
- Dataframe contendo os resultados de abundância total, riqueza total, diversidade de Shannon e Equitabilidade de Pielou.

¹ Disponível em: <https://palaeo-electronica.org/2001_1/past/issue1_01.htm>. Acesso em: 03/05/2018. ² Disponível em: <<http://viceroy.colorado.edu/estimates/>>. Acesso em: 03/02/2018.

Comentários gerais: O plano está bem organizado, o que permitiu que eu acompanhasse sem problemas a sua proposta. A generalidade dela é apropriada e, ao que parece, ela seria útil no seu laboratório (o que é ótimo). Acho que você deve complexificá-la para que o usuário tenha mais liberdade e opções para analisar a diversidade. Mais adiante, darei algumas sugestões de como você pode enriquecê-la.

Para pensar:

- A qual tipo de classe devem pertencer os nomes das espécies?
- Para fazer a manipulação dos dados, o R vai precisar agrupar os indivíduos que pertencem à mesma espécie. O critério para ele fazer esse agrupamento será pelos nomes das espécies. O que vai acontecer se houver um erro de digitação (lembre do Tutorial 4¹)? Você consegue pensar em alguma maneira de fazer essa verificação ou fornecer dados para que o próprio usuário possa fazê-la?
- Como será o gráfico com as abundâncias relativas das espécies? Faça um esboço (na mão mesmo) de como você gostaria que ele ficasse e tente criá-lo no R.

Sugestões:

- Na tabela que você colocou como input, os nomes das espécies estão em negrito e itálico. Apesar de ser possível manipular esse tipo de objeto, o R não gosta muito dessas formatações.
- O pseudo-código está bastante geral. Dentro de cada um dos passos enumerados, terão vários passos intermediários. Vale a pena você fazer esse detalhamento antes de iniciar a função, levando em consideração os objetos que serão criados e como cada um deles será manipulado.
- Que tal o usuário poder decidir quais análises a função vai executar? Vamos dizer que ele só tenha interesse na Diversidade de Shannon, seria interessante se o output fosse apenas esse índice. Além de dar mais liberdade ao usuário, essa modificação pouparia manipulações desnecessárias dos dados.
- Depois que os dados forem agrupados de acordo com as espécies, as manipulações serão

simples. Acho que enriqueceria o trabalho se a função executasse mais coisas: (1) realizar outras análises de diversidade (índice de Simpson, por exemplo) e/ou, (2) comparar a diversidade em mais de uma localidade.

¹ http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02_tutoriais:tutorial4:start

Plano B: Treinos de Academia

Contextualização e Objetivo

É recomendado e comum entre os praticantes de academias esportivas a variação de treinos (conjuntos de exercícios físicos) a cada um ou dois meses, ou seja, uma rotatividade de diferentes exercícios que trabalham um mesmo grupo muscular, independentemente do objetivo dos exercícios, se para perda de peso ou para ganho de massa muscular. Por conseguinte, há uma demanda pela produção de diferentes conjuntos de exercícios, os quais normalmente são agrupados em diferentes treinos considerando os seguintes conjuntos musculares: membros superiores e membros inferiores, plano anterior e plano posterior, exercícios de empurrar e de puxar; e considerando os dois principais objetivos de treinos: perda de peso ou ganho de massa muscular.

Assim, considerando uma base de dados contendo diferentes exercícios físicos, que abordam grupos musculares distintos e que podem ser realizados com dois objetivos diferentes, a criação de uma função para elaboração de treinos básicos diversos (randomização) e harmônicos, ou seja, que trabalhem todos os principais grupos musculares, poderá facilitar este trabalho, seja para usuários quanto para instrutores de academia.

Planejamento da função

Entrada:

Para aplicação dessa função deverá ser construído um banco de dados contendo as seguintes informações: nome do exercício; músculo primário trabalhado; localização do membro trabalhado (superior ou inferior); plano corporal trabalhado (anterior ou posterior); e ação realizada (empurrar ou puxar). Além disso, a aplicação dos exercícios também poderá ser feita com dois objetivos distintos: perda de peso (3 séries de 15-20 repetições) ou ganho de massa muscular (4 séries de 6-8 repetições).

O input da função constituirá em um dataframe contendo as opções escolhidas pelo usuário: setorização do treino (ou seja, se por membro trabalhado, plano corporal trabalhado ou ação realizada) e objetivo do treino (perda de peso ou ganho de massa muscular).

Verificando os parâmetros:

- A opção selecionada para setorização do treino é um objeto constante na base de dados? Se não, retorna: Setorização do treino deve ser um objeto existente na base de dados.
- A opção selecionada como objetivo do treino é um objeto constante na base de dados? Se não, retorna: Objetivo do treino deve ser um objeto existente na base de dados.

Pseudocódigo:

1. Criar treinos A e B, conforme opção de segmentação escolhida pelo usuário;
2. Criar exercícios de forma randômica, evitando a repetição do músculo primário trabalhado;

3. Criar a série e repetições dos exercícios escolhida pelo usuário.

Saída:

- Treinos A (membros superiores) e B (membros inferiores), por objetivo (perda de peso ou ganho de massa muscular); ou
- Treinos A (plano anterior) e B (plano posterior), por objetivo (perda de peso ou ganho de massa muscular); ou
- Treinos A (ação de empurrar) e B (ação de puxar), por objetivo (perda de peso ou ganho de massa muscular).

Comentários gerais: Eu tive mais dificuldades para entender essa função do que a do Plano A. Certamente, grande parte se deve a minha falta de familiaridade com academias. Mas, além disso, senti falta de exemplos dos dataframes e do algoritmo da função. Acredito que uma função que monte treinos de academias possa ser bastante útil. O input e o output estão muito simples, mas é possível explorá-los mais.

Para pensar:

- A sua função precisa de um banco de dados dos exercícios e suas características. Pelo o que eu entendi, esse não é o input e nem o output da função. Como você faria para colocar esse dataframe dentro da função? É necessário tê-lo completo em todas as vezes que ela for rodada?
- O input da função são apenas dois argumentos. Será que faz sentido organizá-los em um dataframe?
- Para verificar se o usuário inseriu os argumentos corretamente, o teste seria “(...) é um objeto constante na base de dados?”. Será que não tem uma maneira mais simples de fazer essa verificação?
- Se eu entendi corretamente, quando a função rodar, o dataframe com os exercícios será filtrado pela setorização do treino. Em seguida, dentro desse subset, serão montados os treinos A e B sendo que cada um deles trabalha todos os músculos. Como você faria para garantir que todos os músculos sejam trabalhados nos dois treinos?
- É realmente o usuário que escolhe a série e repetições dos exercícios?
- A qual tipo de classe pertence o output da função?

Sugestões:

- O pseudo-código só resume o que a função vai fazer, sem especificar os objetos e nem como eles serão manipulados. Como no plano anterior, também aconselho que você desenvolva o algoritmo antes de começar a escrever o script.
- O usuário só pode determinar duas coisas: a setorização e o objetivo do treino. Não existem outras variáveis que deveriam ser lavadas em conta na hora de montar o treino (por exemplo: dieta, tempo disponível na semana, etc.)?
- No início da descrição, você comentou que é interessante variar os treinos a cada um ou dois meses. Vamos supor que eu vou começar a frequentar uma academia e rodo a sua função no início do primeiro mês para montar um treino. No mês seguinte, eu vou precisar variar o treino e, para isso, rodo sua função novamente. O que garante que os exercícios não serão os mesmos do mês anterior? Por que você não pensa em montar treinos para um bimestre/semestre/ano?

Conclusão: Brenda, os seus dois planos se propõem a resolver problemas relevantes. Em ambos os casos, acho que seria interessante permitir que o usuário interaja mais com a função. Os pseudo-códigos também precisam ser mais trabalhados antes de você começar a montar o script propriamente. Pessoalmente, gosto mais do Plano A (posso estar enviesada por ser bióloga e não frequentar academias): a proposta me parece mais consolidada e o algoritmo vai envolver uma série de manipulações diferentes. Se quiser conversar, pode me mandar um e-mail (carolina.mendonca@usp.br) e a gente combina um horário. Bom trabalho! 😊

Carol Mendonça

Carol, muito obrigada pela avaliação das minhas duas propostas. Seus comentários são muito pertinentes; obrigada pelos pontos levantados para pensar melhor e pelas sugestões de melhoria. Como sugerido, já estou trabalhando melhor no pseudo-código. Vou seguir com o Plano A e tentarei incorporar suas sugestões à minha proposta inicial.

Abraços, Brenda.

TRABALHO FINAL: Para elaboração do trabalho final segui o Plano A, incorporando grande parte das sugestões e pontos de reflexão levantados na avaliação da proposta. Um ponto que não consegui atender, por não saber como fazer, foi a resolução de erros de digitação na planilha de input. Obrigada a todos que fizeram parte da disciplina!

Link para Função: [analisar](#)

Link para Help da Função: [Help\(analisar\)](#)

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:brenda.correa:start&rev=1597223093 

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