

Nathalia Monalisa



Perfil

Doutoranda em Ecologia no Instituto de Biociências, USP.

Integrante do Laboratório de Ecologia de Florestas Tropicais (LABTROP/USP).

Meu perfil: <http://labtrop.ib.usp.br/doku.php?id=equipe:nathaliamonalisa:start>

Lattes: <http://lattes.cnpq.br/7276271320727179>

Exercícios

[Meus exercícios](#)

Trabalho Final

Proposta A: função competition()

1.Descrição da proposta: Considerando que as interações competitivas constituem uma das principais forças estruturadoras de comunidades de plantas, a função proposta tem como objetivo calcular um índice que informa a intensidade de competição que uma árvore sofre das árvores vizinhas (neighborhood crowding effect). Para isso, assume que o efeito competitivo sobre um indivíduo arbóreo focal é uma função da identidade, do tamanho e da distância dos seus vizinhos imediatos. O índice que estima os efeitos espécie-específicos (ou seja, o efeito de cada espécie sobre esse indivíduo) é calculado pela expressão (adaptado de Adler et al. 2010¹):



Onde:

- W_{ijm} : efeito competitivo que um indivíduo i da espécie j experimenta dos vizinhos da espécie m ;
- α_{jm} : coeficiente da escala espacial de efeito. Determina a escala espacial dentro da qual vizinhos de uma espécie m exercem efeitos sobre as taxas vitais de vizinhos da espécie j (calculado previamente a partir de modelos de regressão). É importante ressaltar que não necessariamente $\alpha_{jm} = \alpha_{mj}$, exceto nos casos de indivíduos da mesma espécie (onde $m = j$). Na expressão, esse coeficiente representa o fator de multiplicação da distância;
- d : distância do vizinho k da espécie m ao indivíduo focal;
- u : tamanho do indivíduo k da espécie m ;
- k = indexa todos os indivíduos competidores da espécie m .

O efeito total da competição sobre o indivíduo (W_{ij}) é obtido pela soma dos efeitos que ele sofre de

todas as espécies ao seu redor (calculado pela expressão anterior), ponderado pela abundância. E o valor obtido será diretamente proporcional ao tamanho dos vizinhos competidores, mas inversamente proporcional à distância deles. Ou seja, quanto maior ou mais próximo o vizinho, maior o efeito de competição sofrido. A figura abaixo representa esta ideia:



1.2.Onde a função pode ser usada: o cálculo desse índice pode ser incorporado em modelos espacialmente explícitos de crescimento e sobrevivência que levam em consideração o efeito local das interações competitivas sobre as taxas vitais das espécies (como por exemplo em Chu & Adler, 2015² e Uriarte et al., 2010³).

2.Entradas: O usuário deve fornecer :

a) Um dataframe com as linhas correspondendo aos indivíduos arbóreos (nº de observações) e as colunas correspondendo aos seguintes atributos:

- individuo: variável contendo um rótulo de identificação único para cada indivíduo (ou seja, dois indivíduos não podem receber a mesma numeração);
- species: variável contendo o nome ou o código da espécie;
- tamanho: variável numérica contendo o diâmetro a altura do peito (cm) do indivíduo;
- x: variável com a coordenada espacial no eixo X;
- y: variável com a coordenada espacial no eixo Y;

b) Uma matriz de coeficientes de escala espacial de efeito (α) entre todos os pares de espécies. As espécies que exercem os efeitos (m) devem estar nas linhas e as espécies que recebem o efeito (j) devem estar nas colunas.

3.Saídas: a função deve retornar dois outputs:

- a) um data frame com o valor do índice de competição para cada indivíduo da comunidade (W_{ij});
- b) um dotplot indicando os valores do índice e a localização dos indivíduos.

4.Pseudocódigo:

- a) Realiza um teste de premissas (teste lógico) para conferir se os objetos pertencem à classe correta. Se sim, prossegue a execução do código. Se não, exibe uma mensagem de erro sugerindo a correção;
- b) Calcula uma matriz de distância entre todos os pares de indivíduos com base nas coordenadas geográficas;
- c) Usa um loop para calcular o somatório do efeito de todos os indivíduos de uma mesma espécie (W_{ijm}) sobre o indivíduo focal i;
- d) Calcula a soma ponderada dos efeitos de todas as espécies para encontrar o efeito total (W_{ij}) sobre o indivíduo focal i;
- e) Exporta o dataframe com os valores do índice de competição para o diretório de trabalho;
- f) Plota o gráfico.

5.Pacote requerido:

plotly

Referências bibliográficas:

¹Adler, P. B., Ellner, S. P., & Levine, J. M. (2010). Coexistence of perennial plants: An embarrassment of niches. *Ecology Letters*, 13(8), 1019–1029. doi:10.1111/j.1461-0248.2010.01496.x

²Chu, C., & Adler, P. B. (2015). Large niche differences emerge at the recruitment stage to stabilize

grassland coexistence. Ecological Monographs, 85(3), 373–392. doi:10.1890/14-1741.1

³Uriarte, M., Swenson, N. G., Chazdon, R. L., Comita, L. S., John Kress, W., Erickson, D., ... Thompson, J. (2010). Trait similarity, shared ancestry and the structure of neighbourhood interactions in a subtropical wet forest: Implications for community assembly. Ecology Letters, 13(12), 1503–1514. doi:10.1111/j.1461-0248.2010.01541.x

Proposta B: função `bufunfa.ppg()`

1.Descrição da proposta: todos sabemos que a vida financeira dos pós-graduandos residentes na cidade de São Paulo ~~(e em todas as outras cidades brasileiras)~~ não é lá muito saudável. E que precisamos fazer uma ginástica diária para garantir o cafezinho nosso de cada dia. Por isso é importante manter o controle dos gastos ao longo do mês e entender quanto e onde investimos as nossas escassas moedas que sobram do bandejão. Assim, o objetivo é desenvolver uma função que auxilie os pós-graduandos a administrarem suas economias mensais, gerando estatísticas e relatórios detalhados das suas movimentações financeiras.

1.2. Onde a função pode ser usada: em qualquer máquina operada por pós-graduandos nerds interessados em programação e que estejam seriamente preocupados com as suas vidas financeiras. Mas não recomendo o uso para pessoas excessivamente ansiosas ou hipocondríacas, pois o recorrente saldo no vermelho pode causar fortes efeitos colaterais! 😊

2.Entradas: o usuário deve fornecer como base de dados um dataframe onde as linhas representam os dias (variando de 0 a 31) em que foram registrados gastos, enquanto as colunas representam os seguintes atributos:

- descrição: produto(s) ou serviço(s) adquirido naquele dia;
- valor: custo da despesa;
- categoria: cada item deve ser classificado de acordo com uma das seguintes categorias de gasto: transporte, saúde, supermercado, bandejão, despesas de casa, lazer, higiene pessoal, papelaria, outros;
- necessidade: cada gasto deve ser classificado em: *essencial* (itens estritamente necessários, como café e miojo) ou *supérfluo* (itens desnecessários e que poderiam ser evitados, como roupas e sapatos 😊);

Exemplo:



:

2.1.Argumentos: `bufunfa.ppg(dados, bolsa, fapesp,tip)`

- *dados* = base de dados descrita no item anterior (dataframe)
- *bolsa* = variável do tipo string que informa o nível de formação do pós-graduando (“mestrando” ou “doutorando”). Esse argumento será utilizado para inferir o valor da bolsa recebida pelo pós-graduando. Por padrão, mestrandos recebem RS1500 e doutorandos recebem RS2200.
- *fapesp* = variável lógica que informa se o ~~(sortudo-do)~~ aluno é bolsista fapesp. Se sim, o valor da bolsa será corrigido.
- *tip* = variável numérica que informa se o pós-graduando teve alguma renda extra naquele mês, além da bolsa. O valor extra será somado ao valor da bolsa (inferida a partir do argumento anterior) de onde serão deduzidas as despesas diárias para calcular o saldo do mês. O usuário deverá preencher a célula com 0 se não houve entradas extras.

3.Saídas:

- a) Tabela com o valor total das despesas para cada categoria de gasto;
- b) Tabela com o valor das despesas essenciais versus despesas supérfluas;
- c) Barplot com a porcentagem de despesas em cada categoria de gasto;
- d) Gráfico do decaimento da renda total ao longo dos dias do mês;
- e) Saldo do mês acompanhado de uma mensagem motivacional para estimular o pós-graduando a gerir melhor suas economias.

4.Pseudocódigo:

- a) Realiza um teste de premissas (teste lógico) para conferir se os objetos e variáveis pertencem à classe correta. Se sim, prossegue a execução do código. Se não, retorna uma mensagem de erro sugerindo a correção;

b)Preenche as células vazias com **na**

- c)Faz um teste lógico para as variáveis “bolsa” e “ fapesp” e atribui o valor da bolsa ao objeto ‘entradas’

Se bolsa=mestrando & fapesp=FALSE, então entradas = 1500

Se bolsa=doutorando & fapesp=FALSE, então entradas = 2200

Se bolsa=mestrando & fapesp=TRUE, então entradas = 2168.70

Se bolsa=doutorando & fapesp=TRUE, então entradas = 3726.30

- d)Se o argumento tip > 0, soma a quantia ao valor da bolsa no objeto ‘entradas’

- e)Soma o total de gastos da coluna “dados\$valor” e atribui ao objeto ‘saidas’

- f)calcula o saldo do mês (entradas - saídas) e guarda em um objeto chamado ‘saldo’

- g)Calcula a tabela de gastos por categoria usando a função tapply

- h)Calcula a tabela de gastos essenciais vs. supérfluos usando a função tapply

- i)Calcula a proporção de cada gasto em relação ao total e guarda na coluna dados\$prop

- j)Abre a janela gráfica e passa os parâmetros gráficos

- k)Plota o gráfico de barras (barplot) com a porcentagem de gastos em cada categoria

- l)Plota o gráfico de decaimento da renda ao longo dos dias

- m)Verifica o saldo com um teste lógico e printa uma mensagem motivacional de acordo com o seu valor:

- Se saldo =< 0: então [escreve em fonte vermelha] “Danger!Danger!Danger!Seu saldo do mês é (saldo). Desse jeito você precisará trabalhar por mais 3 reencarnações para poder se aposentar!Trate de vender alguns livros encalhados para complementar a sua renda. 🤔”

- Se saldo > 0 & saldo < 300, então [escreve em fonte azul] “Muito bem, tigrão. Seu saldo do mês é (saldo). Você está ficando bom com as economias, hein?Mas ainda há um longo caminho até a aposentadoria. Continue trabalhando firme e a partir de agora cobre mais caro para colocar os trabalhos dos IC’s nas regras da ABNT que você vai chegar lá. 😊”

- Se saldo >= 300: [escreve em fonte verde] “WoooW!Seu saldo do mês é (saldo). Você é mesmo o

mestre dos magos, ou melhor, dos malabarismos! Se continuar nesse ritmo você estará aposentado dentro de 127 anos. Parabéns! Assim você vai longe... 😊”

5. Dificuldades e dúvidas:

Ainda não sei bem como fazer o gráfico de decaimento da renda ao longo dos dias.

Realização da proposta

Proposta realizada: A

Página do código: [Código competition\(\)](#)

Página do help: [Help competition\(\)](#)

Arquivo da função: [arquivo_competition](#)

From:

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

Permanent link:

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



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