

MARATONA DE ANÁLISE DE **BIODIVERSIDADE NO R**

UMA APOSTILA DE EXEMPLOS

MARCOS V. C. VITAL
ACADEMIA DO R

OI, PESSOAL!

Esta apostila foi criada para que você, que está dando seus primeiros passos no R (ou até já tem um pouco de prática nele) possa ver um pouquinho de seu potencial para análises de dados de biodiversidade. Nela, vou dar alguns exemplos bem legais de como trabalhar com estes dados no R, e mostrar algumas funções e análises que com certeza serão úteis para você!

A apostila funciona de forma independente, você não precisa fazer download de dados ou outras coisas, certo? Os dados são exemplos que existem na própria memória do R – é bem prático para quem quer aprender! A única necessidade será você instalar os pacotes ‘vegan’ e ‘iNEXT’, e se você não sabe muito bem como fazer isso, não tem problema, veja este meu tutorial sobre o assunto, no qual eu falo como se instala um pacote e ainda dou dicas de como encontrar pacotes novos. Ah, e se você está bem no começo, pode conferir também este meu outro tutorial aqui, sobre a instalação do R e do Rstudio. ;)

E se você ainda não fez isso, não deixe de seguir o perfil da Academia do R, pois estou sempre compartilhando muita coisa legal sobre o R por lá, tanto sobre este tipo de tema como sobre outras coisas também, viu?

COMO FUNCIONA ESTA APOSTILA

Ao longo deste material, eu vou te mostrar e explicar um pouco sobre procedimentos no R relacionados à dados de Biodiversidade. Para acompanhar fazendo tudo no seu computador, você deverá instalar o R, instalar os pacotes ‘vegan’ e ‘iNEXT’ e, se quiser, instalar o RStudio (ele torna o uso do R bem mais prático, mas é opcional, ok?). Se tiver dúvidas de como fazer estas coisas, confira o canal da Academia do R no Youtube, pois gravei vários tutoriais lá, e eles vão te ajudar com todos estes passos.

Na medida em que avançarmos, todos os procedimentos que você pode repetir no R estarão em destaque com um fundo escuro. Então, você pode copiar e colar os procedimentos para executar no seu R e ver o que acontece, ou então, se preferir, pode digitar tudo por conta própria (a segunda opção é muito boa para quem está aprendendo). Tente repetir o código a seguir:

```
#Um exemplo simples, vou criar números ao acaso e criar um gráfico com eles:  
numeros <- rnorm (100)  
hist (numeros, las=1)
```


Viu como é legal? Mas lembre-se de uma coisa importante: todos os procedimentos que vamos fazer aqui tem uma ordem correta, ok? Se você tentar executar algo lá do meio da apostila, sem ter feito os passos anteriores, provavelmente vai dar errado... Então preste atenção nisso, certinho?

ANTES DE COMEÇAR A TRABALHAR: CARREGANDO OS PACOTES

O R é modular: você pode instalar pacotes (são milhares deles!) que são específicos para as análises que você quer fazer. Nesta apostila, vamos brincar com dois pacotes, o 'vegan' e o 'iNEXT'. Então, quando der início à sua sessão do R, comece carregando os dois, assim:

```
#Carregando os pacotes:  
library (vegan)  
library (iNEXT)
```

E AGORA, OS DADOS!

Para começar, vamos pedir para o R chamar um conjunto de dados de exemplo. Estes dados não estão em um arquivo, então você vai carregá-los da memória do R mesmo. Esta tarefa usa uma função feita só pra isso, a função `data()`. Vamos lá:

Para começar, vamos pedir para o R chamar um conjunto de dados de exemplo. Estes dados não estão em um arquivo, então você vai carregá-los da memória do R mesmo. Esta tarefa usa uma função feita só pra isso, a função `data()`. Vamos lá:

```
#Carregando os dados:  
data (BCI)
```

Prontinho, era só isso mesmo! Fácil, né? Os dados agora estão carregados no R, e existem como um objeto com nome 'BCI'. Quer saber mais sobre os dados? É fácil também, faça assim:

```
#Saber mais sobre os dados:  
?BCI
```

Os dados são de árvores na ilha de Barro Colorado (fica no canal do Panamá). Cada linha da planilha representa uma parcela de 1 hectare, e cada coluna representa uma espécie (os valores são contagens, o número de árvore por espécie em cada parcela). No total, são 50 parcelas e 225 espécies ao todo. Este formato (unidades amostrais nas linhas, espécies nas colunas) é a forma mais comum (e a mais recomendada) de se organizar dados de comunidades biológicas.

Mas chega de papo, vamos começar a trabalhar com os dados!

EXPLORANDO OS DADOS

Vamos começar com uma tarefa simples: calcular a riqueza de espécies (ou seja, o número de espécies) para cada uma das parcelas. O `vegan` tem uma função pra isso, a `specnumber()`, e é bem fácil de usar:

```
#Calculando a riqueza de cada parcela:  
riqueza.parcela <- specnumber (BCI)  
#Vendo os valores para cada uma das parcelas:  
riqueza.parcela
```

Agora, são muitos valores, não é? Então vamos olhar para a média e o desvio padrão (o desvio é uma medida de variação em torno da média). Vamos lá:

```
#Calculando a riqueza média e seu desvio padrão:  
mean (riqueza.parcela)  
sd (riqueza.parcela)
```

Bom, já vimos o número de espécies em cada parcela, mas se quisermos, podemos calcular o número total de árvores em cada uma também! Assim podemos saber se existem parcelas mais ou menos densas em relação à quantidade de árvores (e no fim, vamos calcular a média também, só pela curiosidade). Lá vai:

```
#Quantas árvores forem registradas em cada parcela?  
arvores.parcela <- rowSums (BCI)  
arvores.parcela  
  
#Qual o número médio de árvores por parcela?  
mean (arvores.parcela)
```

Perceba que desta vez não usamos uma função específica (como foi o caso da função `specnumber`, que serve para se calcular o número de espécies). Foi mais simples, usamos a função `rowSums()` para somar o valor total de cada linha! Se cada valor é o total de árvore de cada espécie, então a soma nos dá o número de indivíduos de cada parcela. ;)

Uma das coisas bacanas do R é que podemos aprender a “fazer perguntas” para ele. Por exemplo, eu posso olhar para estes dados, com 50 parcelas, e querer identificar quais delas se destacam tendo mais ou menos árvores que a maioria. Veja como podemos fazer isso:

```
#Quais parcelas se destacam?  
  
#Vamos ver quais são as parcelas com menos de 400 árvores:  
arvores.parcela [which (arvores.parcela<400)]  
  
#E quais tem mais de 500 árvores:  
arvores.parcela [which (arvores.parcela>500)]
```

Aqui o truque foi pedir para o R mostrar apenas uma parte do objeto `arvores.parcela` (foi nele que salvamos os totais de árvores por parcela). A parte entre colchetes é a que cria uma espécie de “filtro”, e dentro deles nós usamos a função `which()` para perguntar ao R quais parcelas tem menos de 400 árvores e, depois, quais tem mais de 500. Bacana, não é?

DIAGRAMA DE RANK-ABUNDÂNCIA

Certo, agora vamos para uma ferramenta super prática, e muito usada em estudos de Biodiversidade: o diagrama de rank-abundância. É uma representação gráfica que mostra a abundância de cada espécie no total, em ordem decrescente. Ajuda a nos dar um vislumbre da estrutura da comunidade biológica, e é super útil para se mostrar os dados coletados de uma forma resumida e na comparação entre amostras.

Para começar, são duas etapas: primeiro, pediremos para o R calcular o total de cada espécie (é uma soma das colunas, então usamos a função `colSums()`). Depois, usaremos a função `sort()` para organizar tudo bonitinho em ordem decrescente


```
#Primeiro, o total de indivíduos para cada espécie:  
somas.especies <- colSums (BCI)  
  
#Agora, colocando em ordem descrescente:  
somas.especies.ordem <- sort (somas.especies, decreasing = T)
```

E agora só falta o gráfico! Dependendo do contexto, podemos usar gráficos de barras, mas neste caso, com muitas espécies (são 225, lembra?), pode ficar mais legal de se entender usando um gráfico com pontos e linhas. Aqui não é o momento de explicar os detalhes do comando, mas qualquer coisa é só chamar o help das funções envolvidas para se entender melhor, ok? Lá vamos nós:

```
#Vamos colocar em um gráfico:  
plot (somas.especies.ordem, las=1, lty=2, type="l", xlab="Ordem das  
espécies", ylab="Total de indivíduos")  
points (somas.especies.ordem, pch=1)
```

Só preste atenção porque são duas funções, uma usada depois da outra. A função `points()` só vai funcionar se o gráfico gerado pela função `plot()` estiver aberto, ok? Ela adiciona os pontos em cima do gráfico feito com a linha.

E vamos de novo fazer perguntas, que tal? De repente queremos saber os nomes de algumas espécies que estão nos extremos do gráfico, e podemos fazer isso facilmente assim:

```
#Vamos ver quem são as espécies com mais de 500 indivíduos registrados:  
somas.especies.orden [which (somas.especies.orden>500)]  
  
#E vamos ver quem são as que ocorreram com apenas um indivíduo:  
somas.especies.orden [which (somas.especies.orden==1)]
```

E agora só falta o gráfico! Dependendo do contexto, podemos usar gráficos de barras, mas neste caso, com muitas espécies (são 225, lembra?), pode ficar mais legal de se entender usando um gráfico com pontos e linhas. Aqui não é o momento de explicar os detalhes do comando, mas qualquer coisa é só chamar o help das funções envolvidas para se entender melhor, ok? Lá vamos nós:

UMA OLHADINHA NA DIVERSIDADE DE ESPÉCIES

Se você trabalha há algum tempo com dados de ecologia de comunidades, certamente conhece um pouquinho sobre os famosos índices de diversidade. Então vamos dar uma passadinha bem rápida por eles, calculando aqui o índice de diversidade de Shannon. Vamos de novo começar com o valor para cada parcela:

```
#Vamos calcular o índice de diversidade de Shannon de cada parcela:  
diversidade.parcela <- diversity (BCI, index="shannon")  
diversidade.parcela
```

Se quiser, depois chame o help da função `diversity()`, que usamos acima, e veja que alterando o argumento 'index' você pode calcular outros índices!

E agora, vamos calcular o valor médio e seu desvio padrão:

```
#E agora, a diversidade média e seu desvio padrão:  
mean (diversidade.parcela)  
sd (diversidade.parcela)
```

Bem simples, não é? O legal do R é que, na medida em que vamos aprendendo a lógica de funcionamento dele, as coisas vão fluindo de uma maneira fácil. O “segredo” é encontrar o caminho certo para se aprender passo a passo como ele funciona. :)

QUANTO FALTA COLETAR?

Esta é a famosa “pergunta que não quer calar”, que surge todas as vezes em que amostramos dados de diversidade biológica, não é? Existem várias formas de se lidar com isso, e vamos ver aqui algumas maneiras de abordar o problema. Para começar, é sempre muito informativo criarmos uma curva de acúmulo de espécies.

Ela tenta nos mostrar quantas espécies esperamos encontrar para cada unidade amostral a mais que coletamos (mas sem extrapolar, apenas com o que coletamos de fato).

Calcular a curva e visualizar os números por trás dela é bem simples e direto no R, basta usar a função `specaccum()` do `vegan`:

```
#Calculando uma curva de acúmulo  
arvores.curva <- specaccum (BCI)  
arvores.curva
```

Não se preocupe com a mensagem de aviso que pode surgir, ok? Dê uma “namorada” nos resultados, eles contam bastante sobre a expectativa do que poderia ser encontrado se você continuasse a coletar. Neste dados das árvores da ilha de Barro Colorado, por exemplo, podemos ver que com 44 parcelas nós já poderíamos ter encontrado 222 espécies, do total de 225 que foram encontradas com as 50 parcelas.

Perceba que aqui não existe fórmula mágica! Você tem que interpretar os dados, à luz do seu conhecimento sobre as espécies e o ambiente. Então não pense em achar uma resposta simples aqui: a ideia é refletir sobre o que os dados estão nos contando, ok?

Podemos fazer um gráfico meio “rápido e sujo” assim:

Fazer um gráfico para mostrar a curva de fato é sempre bem legal, mas perceba que a interpretação dos valores é bem mais importante do que ficar refletindo apenas sobre o formato do gráfico. :)

```
#Gráfico simples:  
plot (arvores.curva)
```

Pessoalmente, acho bem feio... Mas aí está mais uma coisa fantástica no R: podemos sempre fazer diferente. Neste caso, eu mesmo criei uma função para criar este gráfico com o jeitinho que eu gosto. Nós vamos carregar esta função na memória do R “puxando” diretamente da internet. Veja só como vai ser:

```
#Vamos criar um gráfico com função personalizada  
  
#Primeiro, carregando a função:  
source  
("https://raw.githubusercontent.com/marcosvital/scriptsR/master/curvacum.  
R")  
  
#Agora é só chamar:  
curvacum (arvores.curva, "Número de parcelas", "Número de espécies")
```

Pessoalmente, acho que fica bem melhor assim! E o legal é que você pode ou não concordar comigo, pois o R permite você personalizar o gráfico da forma como preferir. Na medida em que aprendemos mais sobre o R, fica cada vez mais fácil fazer isso. :)

Mas vamos seguindo, ok? Se a curva de acúmulo apenas nos dá uma visão geral do quanto estamos encontrando de espécies na medida em que aumentamos nosso esforço amostral, o que podemos fazer para saber quantas espécies ainda podem existir lá fora para encontrarmos na nossa área de estudo? A resposta está nos estimadores de riqueza!

Existe um conjunto “clássico” de estimadores de riqueza de espécies, que existem há algumas décadas na literatura. O R tem uma função que calcula vários deles de uma vez, e que é bem fácil de se usar:

```
#Calculando os "estimadores clássicos" de riqueza de espécie:  
specpool (BCI)
```

Neste exemplo, o estimador Jackknife de primeira ordem estima 245 espécies. Mas note que cada estimador tem sua própria lógica, e no exemplo acima, cada um deles chegou a um resultado distinto.

Uma das coisas chatas dos estimadores clássicos, é que eles não nos dão um ponto de referência. No caso acima, por exemplo, eu poderia me perguntar: quanto preciso coletar para encontrar as tais 245 espécies? Infelizmente o Jackknife não nos diz nada a respeito disso... Mas felizmente existem outros métodos!

O pacote 'iNEXT' do R, desenvolvida pela equipe da Anne Chao (sim, o nome dela é dado a um dos estimadores clássicos!), traz uma série de novos métodos para se estimar riqueza de espécies. Uma das formas legais de se fazer isso é extrapolar a curva de acúmulo de espécies, aquela que vimos agora há pouco. A matemática por trás pode ser complexa, mas a ideia é mais ou menos simples: você “estica” a curva até um ponto de interesse. Pode perguntar, por exemplo, quantas espécies espera encontrar se coletasse o dobro que coletou!

O método que vamos usar a seguir não vai usar as parcelas, e sim o total de indivíduos, ok? Isto porque este método usa a quantidade de indivíduos coletados como medida de esforço – então vamos aproveitar que já temos o total de árvores para cada espécie calculado, e aplicar a função iNEXT:

```
#Extrapolando a curva de acúmulo:  
extrapolação <- iNEXT (somas.especies, datatype = "abundance")  
#Vendo um resumo dos resultados:  
extrapolação
```

A função nos mostra quantas espécies são esperadas para cada medida de esforço (no caso, medido pelo número de árvores coletadas). Este dados tem um total de 21457 árvores coletadas, somando as 50 parcelas. A estimativa para o dobro disso, ou seja, para um total de 42914 árvores, é de 235 espécies

O bacana deste tipo de método é que a estimativa é bem mais concreta, e você pode de fato pesar melhor as coisas. Neste exemplo, você poderia se perguntar se, dentro dos objetivos do estudo, faria sentido se coletar mais 50 parcelas (o que é muita coisa!) para se encontrar, potencialmente, 10 espécies a mais do que foi encontrado até então. Ei, e não se esqueça de fazer o gráfico! Aqui vamos fazer apenas a versão rápida que o iNext fornece, ok? Lá vai:

```
#Um gráfico rápido da extrapolação:  
plot (extrapolação)
```

Mas é claro que, como tudo no R, você pode fazer diferente. ;)

Ufa, acho que por hoje é só, pessoal. Espero que este material sirva para vocês terem uma visão do potencial do R para análises de Biodiversidade, e que seja útil para quem for estudar o assunto. Mas esta é apenas a pontinha do universo que o R nos oferece para lidar com dados desta natureza (e para vários outros tipos de dados!).

O mais importante é que eu espero que este material sirva de estímulo para você aprender mais, tanto sobre o R como sobre as análises. O R, aliás, é uma excelente ferramenta de aprendizado, pode acreditar! Basta conhecê-lo por um caminho que faça sentido, passo a passo. Se quiser conhecer mais sobre o R e sobre método de ensino que eu desenvolvi para ensiná-lo para qualquer pessoa, nos acompanhe na Academia do R!