Ramon Wilk

Doutorando pelo programa de Medicina Tropical, na área de Doenças Tropicais e Saúde Internacional, IMT/USP.

Meu trabalho consiste em utilizar ferramentas moleculares em populações do mosquito vetor da Febre Amarela silvestre (*Haemagogus leucocelaenus*) no estado de São Paulo, e analisar a influência da paisagem em suas populações.

Meus Exercícios

Proposta de Trabalho Final

Proposta 1

Contextualização:

O indice de Shannon é comumente empregado para verificação da diversidade de especies em uma determinada comunidade, de modo a atribuir um maior peso as espécies raras na amostra (SHANNON, 1948). Sendo seu calculo é obito por meio da formula:

Onde: pi é abundância relativa (proporção) da espécie i.

O Índice de Simpson mede a probabilidade de dois indivíduos, selecionados ao acaso na amostra, pertencerem à mesma espécie, de forma a atribuir maior peso as espécies dominantes (SIMPSON, 1949). Sendo seu calculo obtido por meio da formula:

A função tem como objetivo a comparação par a par dos índices de diversidade observados com os valores de N simulações - geradas a partir da exclusão de forma aleatória de uma posição a cada rodada, de modo que possa fornecer o valor de p para os índices observados, bem como seus intervalos de confiança.

Função: índices de diversidade por bootstrap

Entrada (x, índice, n)

x = objeto do tipo data.frame contendo linhas como espécies e colunas como parcelas e/ou localidades.

Índice= índice de diversidade que o usuário deseja calcular: shanonn ou simpson.

n= número de simulações

Verificando parâmetros

data.frame= caso o usuário adicione um outro tipo de objeto, será retornado: “Classe de objeto não corresponde a data.frame”.

Índice= caso não seja informado, por default será calculado o índice de Shannon.

N= caso não seja informado nenhum valor, por default será aplicado 1000.

Pseudo-código

1. Cria um objeto denominado “dados”.

1.1. Converter os valores faltantes “ ” em NA.

1.2. Remove “na.omit”.

2. Condição para calcular os índices de Shannon

2.1. Se if = “shannon”:

2.1.1. Calcula o índice e guarda no objeto denominado “índice.shannon”

2.1.2. Guarda o objeto “índice.shannon” na primeira posição de uma lista denominada “sim.ind.shannon”, contendo 1000 repetições de “NAs”e/ou o n informado.

2.2. Cria um contador “for” que vai de “i” ao comprimento do objeto “sim.ind.shannon”, guardando a partir da posição 2. (a cada rodada será excluída aleatoriamente uma posição do data.frame).

3. Condição para calcular o índice de “Simpson”.

3.1. Se if =”simpson”:

3.1.1. Serão repetidos os passos do item 2, sendo criados os objetos “indide.simpson” e “sim.ind.simpson”.

Saída

Retorna um Boxplot contendo os quartis com os valores simulados na lista para cada uma das parcelas e/ou comunidades.

Retorna uma matriz com os valores do teste t para cada um dos pares das comunidades e/ou parcelas.

Retorna uma matriz com os valores de p.

Retorna um data frame com os valores dos índices e seus intervalos de confiança.

Referências

SHANNON, C. A Mathematical Theory of Communication. **The bell system Technical Jornal**, v. 27, p. 379–423, 1948.

SIMPSON, E. H. Measurement of diversity. **Nature**, v. 163, n. 4148, p. 688, 1949.

Proposta 2

Contextualização:

Unweighted pair-group method using arithmetic averages (UPGMA), permite verificar a proximidade entre os pares de grupos de forma hierárquica, não importando a disposição previa deste em determinada matriz. Apesar da existência de outras metodologias, a exemplo de Neighbour-joining, o método continua sendo empregado tanto em analises filogenéticas, quanto em estudos de variações fenotípicas em caracteres morfológicos.

A função tem como objetivo retornar um dendograma com base em uma matriz de distâncias, ex. distâncias euclidianas tomadas a partir de caracteres morfológico dos espécimes.

Função 2: UPGMA por média aritimética.

Entrada (x, pop=TRUE)

x = objeto do tipo data.frame contendo linhas como espécies e colunas como coordenadas x e y, de dados morfométricos, onde a primeira coluna guarda os indivíduos e a segunda os grupos (populações).

pop= agrupa os indivíduos por seus respectivos grupos.

Verificando parâmetros

data.frame= caso o usuário adicione um outro tipo de objeto, será retornado: “Classe de objeto não corresponde a data.frame”.

pop= caso não seja informado, por default os indivíduos serão agrupados.

Pseudo-código

1. Cria um objeto denominado “matriz”.

1.1. Converte a partir da coluna 3 ao comprimento da matriz em classe numérica.

2. Condição para agrupamento por população.

2.1. Se if = “pop=TRUE”: transforma a segunda coluna em fator.

2.1.1. Calcula a media das colunas por grupo “função aggregate”

2.1.2. Calcula uma matriz de distância e guarda no objeto “dis.matriz”.

2.1.3. Calcula a matrizes reduzidas , comando “for”

2.2. Se if= “pop=FALSE”:

2.2.1. Calcula uma matriz de distância e guarda no objeto “dis.matriz.ind”

2.2.2. Calcula a matrizes reduzidas , comando “for”.

3. Utiliza o objeto “dis.matriz” ou “dist.matriz.ind” para calcular os braches e construir o dendograma UPGMA.

Saída

Retorna um plot do dendograma UPGMA.

Oi Ramon, achei suas duas ideias legais e válidas, mas recomendo seguir com o plano A mesmo.

Sobre o plano A. Pelo que eu entendi, a ideia é testar a hipótese de diferença entre similaridade entre todos os pares de comunidades usando um teste de reamostragem por bootstrap. Digo isso porque achei o texto meio confuso, então tente caprichar mais no help (não entendi porque vc fala de teste t, por exemplo). Se for isso mesmo, acho uma boa proposta. Pode prosseguir com ela.

Entretanto, como bom monitor enxerido, tenho algumas sugestões: (1) ao invés de bootstrap, eu recomendaria fazer testes de permutação. Num teste de permutação, vc embaralharia os indivíduos entre as comunidades e recalcularia o índice. O livro do Manly (Randomization, Bootstrap and Monte Carlo methods in biology) é a principal referência que eu recomendo dar uma olhada. (2) pense com carinho no formato do objeto de saída, pra que ele seja o mais informativo possível. Como são vários testes e várias distribuições nulas, seria legal pensar numa tabela ou matriz que resumisse bem as coisas.

Sobre o plano B, caso vc acabe seguindo por ele, eu recomendaria pensar melhor no objeto de retorno mesmo. Calcular tanta coisa e só devolver um plot me parece um desperdício. Então a dica (2) sobre o plano A vale pro plano B também.

Danilo G Muniz

Links para o trabalho final

Link para página da minha função minha função

Link para a página da ajuda da função Help da função