



Bacharel em Ciências Biológicas com ênfase em Biologia Marinha (UNIVILLE - 2013). Tem experiência em ecologia de pequenos cetáceos, tendo desenvolvido pesquisas sobre dinâmica populacional e ecologia trófica. No momento é pesquisador colaborador do Projeto Toninhas/UNIVILLE e Mestrando do Programa de Pós-graduação em Ecologia/UFSC.

Exercícios [preaula.r](#) [exercicio_4_-_wiki.r](#) [exercicio_5_-_renan.r](#) [exercicio_7a.2_7b.1_7b.2_-_renan.r](#) [exercicio_8.2_wiki.r](#)

Trabalho final

Proposta A - Nearest Neighbor Index (NNI): Em ecologia espacial, identificar se existe ou não padrões de agrupamento de pontos pode ser muito útil em uma variedade de aplicações. O método de Nearest Neighbor (NN - “vizinho mais próximo” em inglês) foi inicialmente introduzido por J.G. Skellam, onde a relação entre os valores médios esperado e observado das distâncias entre pontos vizinhos é utilizado para determinar se um conjunto de dados possui agrupamento. A partir do método obtém-se o Nearest Neighbor Index (NNI), sendo $NNI = 1$ distribuição aleatória; $NNI < 1$ distribuição agrupada; e $NNI > 1$ distribuição uniforme. Minha primeira proposta de função consiste em obter o valor do NNI a partir de um conjunto de coordenadas geográficas. Para isso os dados de entrada deverão ser dois objetos vetoriais contendo as coordenadas de x (latitude) e y (longitude) respectivamente, e um argumento contendo o tamanho da área amostral (A). Os dois objetos devem necessariamente ter o mesmo length e A deve ser expresso em m². O sistema de coordenadas deve ser em UTM (coordenadas planas), que é baseado no plano cartesiano (eixo x,y) e usa o metro (m) como unidade para medir distâncias e determinar a posição de um objeto. Haverá um aviso de alerta quanto à isso! Assim, no primeiro momento a função irá determinar todas as distâncias existentes entre os pontos através do Teorema de Pitágoras. Em seguida, deverá selecionar qual a distância do vizinho mais próximo para cada ponto.



*O modelo aleatório utilizado é derivado de um modelo CSR (Complete Spatial Randomness), que assume que os pontos são distribuídos de forma aleatória, baseados em uma distribuição de Poisson.

O output da função será uma matriz contendo todas as distâncias entre os pontos com os valores de NN destacados; e o valor do NNI.

Complemento: eventualmente daria ainda para inserir um teste de significância do valor de NNI... essa seria minha proposta B, mas eu me dei conta que para simular modelos nulos eu teria que utilizar o mesmo formato de entrada de dados e necessariamente passar por todos os passos da proposta A. :/

<http://www.public.iastate.edu/~pdixon/stat406/NearestNeighbor.pdf>
<http://www.mcs.sdsmt.edu/rwjohnso/html/sofiya.pdf>

Proposta B - Variação temporal/individual da dieta: Em uma matriz de espécies predadas (contendo a abundância de cada espécie) por estômago (predador), a função teria por objetivo simular cenários nulos, aleatorizando as presas encontradas nos estômagos, porém apenas dentro de uma variável categórica, como estação do ano em que o estômago foi coletado, sexo ou faixa etária do predador. Ex:



Assim, a função iria no primeiro momento verificar se cada “sub-matriz” é significativa em relação à uma série de cenários nulos formulados com os dados do sub-conjunto, com reposição (bootstrap). No segundo momento iria comparar as sub-matrizes utilizando testes de Mantel.

O output seria os valores de significância das sub-matrizes em relação aos cenários nulos e dos testes de Mantel.

Autocrítica: Não tenho certeza se esse seria o caminho mais adequado para se fazer as comparações que eu gostaria...

As duas propostas estão claras, e me parece que vc pode seguir com qualquer uma das duas que vc escolher. Na proposta A, é melhor pedir que a entrada seja um objeto com as duas coordenadas em colunas diferentes, assim não precisa checar o comprimento, e há menor problemas de erros de entradas de dados e NAs.

— *Vitor Rios*

Resposta: Valeu pela dica Vitor. Resolvi por seguir com a proposta A. Gostei muito do resultado, teve momentos que pensei que não conseguiria converter dados de coordenadas em uma matriz de distância entre pontos, mas acabou saindo! Infelizmente não consegui pensar em nenhuma forma para destacar na matriz do output os valores de NN como eu gostaria... mas me dei por satisfeito.

TRABALHO FINAL

HELP DA FUNÇÃO

Introdução à Linguagem R - IBUSP
Trabalho Final – Renan Lopes Paitach
Função: Nearest Neighbor Index (NNI)

Description:

Produz uma matriz das distâncias entre cada par de registros geográficos e o NNI do conjunto de registros, além de uma frase contendo a interpretação do índice.

Usage:

`NNI(coordenadas, area)`

Arguments:

`coordenadas`: data frame ou matriz, com duas colunas contendo as

coordenadas geográficas em UTM de cada registro, sendo a primeira coluna de longitude e a segunda de latitude.

area: numérico, contendo o tamanho da área de estudo em metros quadrados (m^2).

Details:

O Nearest Neighbor Index (NNI – “Índice do Vizinho mais Próximo” em tradução livre), para análise de distribuição espacial pode ser calculado através de:

$$NNI = rA / rE$$

onde:

rA = é o valor médio das distâncias observadas de cada registro geográfico em relação ao “vizinho” mais próximo (NN), calculado como:

$$rA = \sum r / N$$

onde:

r = valores de NN observados

N = número de registros geográficos

rE = é o valor esperado da distância NN para o modelo aleatório(*), calculado como:

$$rE = 1/(2*\sqrt{(N/A)})$$

onde:

A = tamanho da área de estudo em metros quadrados (m^2)

(*) O modelo aleatório utilizado é derivado de um modelo CSR (Complete Spatial Randomness), que assume que os pontos são distribuídos de forma aleatória, baseados em uma distribuição de Poisson.

Value:

A matriz produzida apresenta os valores das distâncias entre todos os registros geográficos.

O Índice de Nearest Neighbor é apresentado numericamente, juntamente com sua interpretação: $NNI = 1$, distribuição aleatória; $NNI < 1$, distribuição agrupada; e $NNI > 1$, distribuição uniforme.

Warning:

O sistema de coordenadas do objeto de entrada “coordenadas” deve ser em UTM (coordenadas planas), que é baseado no plano cartesiano (eixo x,y) e usa o metro (m) como unidade para medir distâncias e determinar a posição de um objeto.

Consequentemente, a unidade do tamanho da área de estudo expressa pelo objeto de entrada “área” deve ser o metro (m^2).

Author(s):

Renan Lopes Paitach
renan_ptch@hotmail.com

References:

Clark, P. and F. Evans (1954). Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations, *Ecology*, 35, 445-453.
Diggle, P. J. (2003). *Statistical Analysis of Spatial Point Patterns*. New York: Oxford University Press Inc.
Dixon, P. M. (1994). Testing spatial segregation using a nearest-neighbor contingency table, *Ecology* 75, 1940-1948.

See Also:

Examples:

```
dados.teste <- read.table("teste.csv", header=TRUE, sep=";", dec=".")
tamanho.baia <- 20000000
NNI(dados.teste, tamanho.baia)
```

CÓDIGO DA FUNÇÃO

```
#### TRABALHO FINAL - RENAN PAITACH ####
```

```
dados <- read.table("trabfinal.csv", header=T, sep=";")
area.baia <- 20000000
```

```
####FUNÇÃO - Nearest Neighbor Index (NNI)####
```

```
NNI <- function(coordenadas, area) #comando para criar a função.
{
  matriz <- matrix(NA, nrow=length(coordenadas$Longitude),
ncol=length(coordenadas$Longitude)) #cria uma matriz contendo NAs com número
de linhas e colunas igual ao número de registros geográficos.
  rownames(matriz) <- rownames(matriz, do.NULL=FALSE, prefix="Registro.")
#modifica o nome das linhas.
  colnames(matriz) <- colnames(matriz, do.NULL=FALSE, prefix="Registro.")
#modifica o nome das colunas.

  for(i in 1:length(coordenadas$Longitude)) #realiza a série de cálculos das
distâncias de acordo com o número de registros geográficos da entrada.
  {
    dist.long <- c(coordenadas$Longitude[i] - coordenadas$Longitude) #obtem
a distância em longitude de cada registro em relação aos demais.
    dist.lat <- c(coordenadas$Latitude[i] - coordenadas$Latitude) #obtem a
distância em latitude de cada registro em relação aos demais.
    pitagoras <- sqrt(dist.long^2+dist.lat^2) #calcula a distância de cada
registro em relação aos demais (hipotenusa do teorema de pitagoras).
    matriz[,i] <- pitagoras #substitui no objeto "matriz" a i'ésima coluna
```

pelo vetor de distâncias calculadas do i 'ésimo registro geográfico em relação aos demais.

```
}
```

```
NN <- rep(NA, time=length(coordenadas$Longitude)) #cria um vetor contendo
NAs com comprimento do tamanho do número de registros geográficos.
```

```
for(i in 1:length(NN)) #seleciona o Nearest Neighbor (NN) de cada registro
geográfico para preencher o vetor "NN".
```

```
{
```

```
ordenamento <- matriz[,i] #cria objeto "ordenamento" contendo os valores
da  $i$ 'ésima coluna do objeto "matriz".
```

```
ordenamento <- as.vector(ordenamento) #transforma "ordenamento" em
vetor.
```

```
ordenamento <- sort(ordenamento) #ordena os valores de "ordenamento" em
ordem crescente.
```

```
ordenamento <- ordenamento[-1] #elimina o primeiro valor que corresponde
à distância do  $i$ 'ésimo registro em relação a ele próprio (=zero)
```

```
ordenamento <- min(ordenamento) #seleciona o valor de NN do  $i$ 'ésimo
registro geográfico.
```

```
NN[i] <- ordenamento #subscrive os NA's do vetor "NN" pelo valor
correspondente ao Nearest Neighbor de cada registro geográfico.
```

```
}
```

```
rA <- sum(NN)/length(NN) #cálculo do valor médio das distâncias de NN
observadas.
```

```
rE <- 1/(2*(sqrt(length(NN)/area))) #cálculo do valor médio esperado das
distância NN para o modelo aleatório.
```

```
NNI <- rA/rE #cálculo do Nearest Neighbor Index.
```

```
if(NNI<1)
```

```
{NNI2 <- "Distribuição agrupada"} #cria um objeto com a interpretação do
teste se o índice for menor que 1.
```

```
if(NNI==1)
```

```
{NNI2 <- "Distribuição aleatória"} #cria um objeto com a interpretação
do teste se o índice for igual a 1.
```

```
if(NNI>1)
```

```
{NNI2 <- "Distribuição uniforme"} #cria um objeto com a interpretação do
teste se o índice for maior que 1.
```

```
resulta <- list(matriz, NNI, NNI2) #cria um objeto do tipo lista contendo
os objetos da saída.
```

```
names(resulta) <- c("Distâncias entre registros:", "Índice de Nearest
Neighbor:", "Interpretação do índice:") #nomeia cada objeto da lista de
saída.
```

```
return(resulta) #output da função: Matriz de distâncias e NNI.
```

```
}
```

```
NNI(dados, area.baia)
```

[trabfinal.csv trabalho_final_-_renan_paitach.r](#)

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2014:alunos:trabalho_final:renan_ptch:start 

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