



Daniel Varajão de Latorre

Iniciação em ecologia. Efeito da área na diversificação de grupos tropicais.

Exercícios

Link para os [exec](#)

Proposta Final

Plano A

Fazer uma função que calcula as distancias dos nós até o topo de uma filogenia que inclua espécies viventes e extintas. Existe uma função chamada `branching.times()` do pacote “ape” que realiza essa tarefa para filogenias que contenham apenas espécies viventes. Porém essa função retorna valores que precisam ser corrigidos quando aplicada a uma filogenia que contenha espécies extintas.

Plano B

Fazer uma função que recebe como entrada os registros de quando uma espécie fóssil foi encontrada e calcular uma estimativa da diversidade “boundary-crossing”. Isso é, uma taxa de diversidade que pode ser calculada para cada período geológico. Essa função pediria como entrada uma lista com os registros de quando cada espécie foi encontrada e retornaria um valor de diversidade para cada mudança de período geológico. O grau de detalhamento do período geológico deve ser informado na chamada da função. Dentro da função haverá uma “tabela” com os tempos geológicos.

Comentários

Olá Daniel,

Se vc. já conhece o pacote APE e já domina a função `branching.times()` é factível modificá-la. Precisa definir bem a entrada de dados, o tipo de arquivo, se os taxa extintos entram em um objeto separado... etc. Nesse sentido, o Plano B me parece melhor estruturado e, ao menos para mim, mais interessante....

— [Alexandre Adalardo de Oliveira](#) 2012/04/03 21:25

Página de Ajuda

branching.times.with.extinct
Documentation

package:unknown

R

Cálculo dos tempos de especiação de uma filogenia com espécies extintas

Descrição:

Essa função computa os tempos de especiação de uma filogenia com espécies viventes e extintas. Isso significa, a distancia entre cada nó e o tempo presente.

Usage:

```
branching.times.with.extinct(phy)
```

Arguments:

phy um objeto no formato phylo

Valores:

Um vetor com os valores entre cada nó e o tempo presente. Se o objeto phy possui um elemento node.label este será usado para nomear o vetor retornado. Caso contrário os nomes são tirados do elemento edge do objeto phy.

Observações:

A função branching.times.with.extinct é uma modificação da função branching.times do pacote ape.

A seguir está o código da função original. A linha indicada por "ERRO!" corresponde ao trecho da função que foi alterado.

```
# branching.times<-function (phy)
# {
#   if (!inherits(phy, "phylo"))
#     stop("object \"phy\" is not of class \"phylo\"")
#   n <- length(phy$tip.label)
#   N <- dim(phy$edge)[1]
#   xx <- numeric(phy$Nnode)
#   interns <- which(phy$edge[, 2] > n)
#   for (i in interns) xx[phy$edge[i, 2] - n] <- xx[phy$edge[i,1] - n] +
phy$edge.length[i]
#   depth <- xx[phy$edge[N, 1] - n] + phy$edge.length[N] #####ERRO!
#   xx <- depth - xx
#   names(xx) <- if (is.null(phy$node.label))
#     (n + 1):(n + phy$Nnode)
#   else phy$node.label
```

```
#      xx
# }
```

A linha indicada por "ERRO!" tem como objetivo calcular o tempo total da filogenia. No entanto quando a filogenia apresenta espécies extintas esse calculo deve ser feito de outra maneira.

A solução proposta na função `branching.times.with.extinct` leva em consideração o tempo de todos os táxons terminais em relação ao nó mais basal da árvore. O tempo usado pela nova função é o valor máximo calculado. Author(s):

Emmanuel Paradis - autor da função original `branching.times`
Daniel Varajão de Latorre - autor das modificações presentes

Examples:

```
library(ape)
phy<-rlineage(1, 0.6, 2)
phy
par(mfrow=c(2,1))
plot(phy)
axisPhylo()
branc<-branching.times.with.extinct(phy)
plot(phy)
axisPhylo()
for(i in 1:length(branc))
  lines(x=c(2-branc[i], 2-branc[i]), y=c(-1000, 1000), col="red")
```

Código da Função

```
branching.times.with.extinct<-function(phy)
{
  n <- length(phy$tip.label)
  N <- dim(phy$edge)[1]
  xx <- numeric(phy$Nnode)
  interns <- which(phy$edge[, 2] > n)
  for (i in interns)
    xx[phy$edge[i, 2] - n] <- xx[phy$edge[i,1] - n] + phy$edge.length[i]
  ##parte nova, novo algoritmo para calcular qual o tempo maximo da
  arvore
  ntip<-Ntip(phy)
  sum<-numeric(ntip)
  index<-numeric()
  x<-numeric()
  for(i in 1:ntip)
  {
    node<-i
    while(node!=ntip+1)
```

```
{
  index<-which(phy$edge[,2]==node)
  sum[i]<-sum[i] + phy$edge.length[index]
  node<-phy$edge[index,1]
}
}
depth <- max(sum)
##volta a ser usado o codigo da funÃ§Ã£o original
xx <- depth - xx
names(xx) <- if (is.null(phy$node.label))
  (n + 1):(n + phy$Nnode)
else phy$node.label
xx
}
```

Arquivo com o Código da Função

[branching.times.with.extinction.r](#)

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2013:alunos:trabalho_final:danielvdelatorre:start

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