



# Sofia Marques Silva

Doutoranda em Biologia/Genética no LABEC-IB-USP. O projeto de doutorado centra-se na investigação de incertezas taxonómicas sobre *Bradypus variegatus* (preguiça-comum), focando-se em contribuições para conservação, nomeadamente na definição dos processos históricos e demográficos que teriam moldado a diversidade nuclear e especiação.

## Proposta de Trabalho Final

### Principal

Uma função que calcule índices de diversidade genética com base na escolha aleatória e gradual de um número crescente de marcadores moleculares nucleares do tipo microsatélite. Este cálculo deverá incluir teste de significância para a diferença de cada valor calculado.

Plano B - Exexutar pelo menos parte da proposta principal.

### Comentário Leandro

Sofia, como serão os dados de entrada para esta função e como será feito esse cálculo de diversidade? Você pode detalhar um pouco mais, pois apesar de achar interessante, não consegui compreender o nível de complexidade da função para saber se ela é factível no tempo disponível.

### Resposta Sofia

O pacote adegenet contém algumas funções interessantes e dá uma ajuda em como realizar esta proposta. Começando por aí, os dados de entrada teriam de estar organizados num `data.frame` com os genótipos em linha e os marcadores em coluna. Proponho utilizar os dados disponibilizados no adegenet também (microbov) em que temos 704 indivíduos, de 15 raças, genotipados para 30 marcadores.

O primeiro passo seria sortear aleatoriamente uma amostra de 10 marcadores, ou seja selecionar 10 colunas (numero minimo geralmente aceite para publicação) e calcular, por exemplo, as frequências alélicas (função `makefreq` do adegenet), a heterozigotia esperada (função `Hs`) e o `Fst` (função `pairwise.fst`). Depois repetir o procedimento para 15, 20, 25, 30 marcadores. Mais marcadores tivessemos, mais continuaríamos...

O mesmo poderá ser aplicado ao número de indivíduos por raça (ou seja linhas), por exemplo de 5 em 5 ou 10 em 10 tb, uma vez que a amostragem o permite. Os índices calculados seriam os mesmos (`Hs` e `Fst`).

Finalmente calcular-se-ia se a diferença entre os índices é estatisticamente diferente.

A função poderia incluir ainda a representação gráfica da variação.

Esta função tem como objectivo avaliar se o numero de marcadores e/ou individuos que estamos utilizando nas nossas análises é suficiente para as análises de genética populacional que pretendemos.

Também seria interessante incluir um argumento na função que permitisse ao utilizador seleccionar sequencialmente marcadores ou individuos de interesse. Isso também pode ser bastante interessante para algumas questões da genética populacional.

## Resposta Leandro

Gostei da proposta. Parece que você já tem bem claras as etapas necessárias para realizar a função. Não sei quanto tempo levará para escrever os cálculos de todos os índices propostos. Uma coisa que achei interessante é o sorteio das amostras dos marcadores ao acaso. Você poderá gerar milhares de sorteios dessas amostras para gerar os “intervalos de confiança” da variabilidade genética para seleção de marcadores aleatórios e comparados com os selecionados por você a partir de algum critério. Era esse o teste estatístico proposto? Acho que só essa primeira etapa dá um bom trabalho. Depois você poderá focar em diferenciar por raças.

[exec](#)

## Página de ajuda

```
choosing.micros                                R Documentation
      Escolha do número mínimo de microssatélites
Description:
  Esta função, para análise exploratória dos dados, seleciona
  sequencialmente marcadores moleculares do tipo microssatélite com base no
  seu número de alelos e avalia a performance de cada pool no cálculo da
  heterozigotia esperada (Hs), por população, relativamente ao total de
  marcadores. Esta função permite a comparação da diversidade genética entre
  populações.
Usage:
  choosing.micros <- function ( x , n , missing.data=NA , npop )
Arguments:
  x          data.frame (ver detalhes)
  n          número máximo de microssatélites a incluir no teste, sendo que
o total
              também é sempre analisado
  missing.data tal como nos pacotes de adegenet pode ser "NA" ou "0"
(ver detalhes)
  npop       número de populações
Details:
  x          0 data.frame deverá ser do tipo:
              pop      L01      L02      ...
```

```
I1    POP1    999/999    999/999
```

```
I2    POP2    ...
```

Para simplicidade da função é essencial obedecer à nomenclatura da primeira linha e a separação dos alelos de um mesmo loci deverá ser feita com "/"

n        Serão feitos os cálculos de Hs com 10 microssatélites e depois vão sendo adicionados mais 10 microssatélites sequencialmente até ao maior valor múltiplo de 10 e menor ou igual a "n". Os cálculos com o total de microssatélites também serão sempre realizados.

missing.data       Estes valores obedecem aos requisitos do pacote adegenet.

Assim, os valores para este argumento podem ser os seguintes e terão os respectivos efeitos:

- "NA": se todos os genótipos de uma população para um dado alelo estão em falta, o valor de count será NA

- "0": se todos os genótipos de uma população para um dado alelo estão em falta, o valor de count será 0

npop        0 número de populações deverá ser sempre indicado, a coluna 2 deverá conter o nome da população para cada indivíduo, mesmo que todos pertençam à mesma população e portanto o mesmo nome apareça em toda a coluna. Para que o nome da(s) população(s) apareça no gráfico é necessário que esta variável seja um factor (ver exemplos).

Value:

comp1 : Na função está incluído o sumário de todos os microssatélites por população que é automaticamente retornado.

comp2 : A ordem dos microssatélites utilizados com indicação do respectivo número de alelos é também apresentada no final.

comp3 : E ainda uma matriz com os valores de Hs calculados por pool de microssatélites e por população.

comp4 : O barplot das Hs calculadas, por pool de microssatélites e por população também é obtido.

Warning:

Requer pacote adegenet.

A instalação deste pacote poderá ser demorada em alguns casos, nomeadamente em Windows Vista, pelo que se recomenda a instalação prévia do pacote. O seguinte comando deverá ser utilizado, para que a instalação seja realizada com sucesso:

```
install.packages("adegenet", dep=TRUE)
```

A função poderá demorar alguns longos segundos. Tanto mais longos quanto maior o número de dados.

Note:

Motivação       Num cenário em que o aumento do número de marcadores moleculares é uma realidade em crescimento

exponencial, surge a pergunta se não será redundante utilizar 10 ou 50 microssatélites, por exemplo.

Por um lado, em espécies modelo, para as quais estão descritos de fato centenas de microssatélites,

poderá ser monetariamente vantajosa a escolha de somente alguns desses marcadores.

Por outro lado, para espécies não modelo, atualmente é já também possível obter um grande número de

microssatélites, mas o número de indivíduos amostrados continua a ser muitas vezes limitado. Deste modo, um conhecimento prévio da variabilidade genética dos marcadores moleculares disponíveis é imprescindível.

Author(s):

Sofia Marques Silva  
sofiamarques1@usp.br

References:

Jombart T. (2008) adegenet: a R package for the multivariate analysis of genetic markers. Bioinformatics 24: 1403-1405.

doi: 10.1093/bioinformatics/btn129

Adegenet on the web: <http://adegenet.r-forge.r-project.org/>

See also:

pacote adegenet: genind2genpop, df2genind, Hs

Examples:

```
bradypus<-  
read.table("bradypus.txt",header=TRUE,as.is=TRUE,sep="\t",row.names=1)  
bradypus$pop<-as.factor(bradypus$pop) # para aparecer a legenda no eixo  
do xx  
Hs.bradypus<-choosing.micros(bradypus,48,missing.data=NA,npop=3)  
Hs.bradypus  
bradypus1<-  
read.table("bradypus1.txt",header=TRUE,as.is=TRUE,sep="\t",row.names=1)  
bradypus1$pop<-as.factor(bradypus1$pop) # para aparecer a legenda no  
eixo do xx  
Hs.bradypus1<-choosing.micros(bradypus1,34,missing.data=NA,npop=1)  
Hs.bradypus1
```

[bradypus.txt](#) [bradypus1.txt](#)

### Código da Função

```
choosing.micros<-function (x,n,missing.data=NA,npop)  
{  
  input <- x  
  input.pop <-  
genind2genpop(df2genind(input[-1],sep="/",missing=missing.data,pop=input$pop  
, ploidy=2, type="codom"))  
  sort.nall <- sort(input.pop@loc.nall,decreasing=TRUE)  
  result <-  
matrix(NA,nrow=(n/10+1),ncol=npop,byrow=TRUE,dimnames=list(c(seq(10,n,10),"Total"),levels(input$pop)))  
  for (i in seq(10,n,10))  
  {  
    name.micros <- names(sort.nall[1:i])  
    data <- input[name.micros]  
    data.ind <-  
df2genind(data,sep="/",missing=missing.data,pop=input$pop, ploidy=2,
```

```
type="codom")
  data.pop <- genind2genpop(data.ind)
  result[(i/10),] <- Hs(data.pop)
}
HsTotal<-Hs(input.pop)
result[(n/10+1),]<-HsTotal
par(mar=c(3,5,1,1))
barplot(result,beside=TRUE,ylab="Hs",cex.axis=0.7,cex.names=0.7,ylim=c(0,(max(result)+0.1)))
par(mar=c(5, 4, 4, 2))
summary(input.pop)
to.return <- list (sort.nall,result)
return(to.return)
}
```

From:

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

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2011:alunos:trabalho\\_final:sofia:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2011:alunos:trabalho_final:sofia:start)



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