

title

5 octobre 2007

<http://Imq.esaiq.usp.br/wiki/>

Table des matières

1	9. Solução dos Exercícios	6
1.1	Exercícios 7 : Noções de Programação em Linguagem S	6
1.1.1	Frequência de Espécies	6
1.1.2	Classe da Classe	7
1.1.3	Logaritmo na Base 2	8
1.1.4	Somatório dos n Primeiros Números Naturais	8
1.1.5	Índices de Dispersão I	8
1.1.6	Gráfico de Whittake	9
1.1.7	Editando Funções Externamente	9
1.1.8	Índices de Diversidade de Espécies	9
1.1.9	Loop para Representar o TCL	10
1.1.10	Tabela de Fitossociologia	10

<http://lmg.esalq.usp.br/wiki/>

<http://Imq.esalq.usp.br/wiki/>

Uso da Linguagem R para Análise de Dados em Ecologia

<http://Imq.esalq.usp.br/wiki/>

Chapitre 1

9. Solução dos Exercícios

São apresentados abaixo os códigos para solução dos exercícios do curso. Lembre-se, entretanto, que, em se tratando de código, sempre é possível mais que um caminho para se chegar ao mesmo fim.

1.1 Exercícios 7 : Noções de Programação em Linguagem S

1.1.1 Freqüência de Espécies

```
> cax <- read.csv("caixeta.csv",header=T)
> cax$especie[1:5]
[1] Myrcia sulfiflora      Myrcia sulfiflora      Syagrus romanzoffianus
[4] Tabebuia cassinoides  indet.1
43 Levels: Alchornea triplinervia Andira fraxinifolia ... Tibouchina nutticeps
>
> class( table( cax$especie ) )
[1] "table"
> attributes( table( cax$especie ) )
$dim
[1] 43

$dimnames
$dimnames[[1]]
 [1] "Alchornea triplinervia"  "Andira fraxinifolia"
 [3] "bombacaceae"            "Cabranea canjerana"
 [5] "Callophyllum brasiliensis" "Callophyllum brasiliensis"
 [7] "Cecropia sp"            "Coussapoa macrocarpa"
 [9] "Coussapoa micropoda"    "Cryptocaria moschata"
[11] "Cyathea sp"             "eugenia3"
[13] "Eugenia oblongata"      "fabaceae1"
[15] "Ficus sp"               "Gomidesia sp"
[17] "Ilex durososa"          "Ilex sp"
[19] "indet.1"                "indet.2"
[21] "indet.3"                "Inga sp"
[23] "Jacaranda puberula"     "jussara"
[25] "Matayba sp"             "Mela 1"
[27] "Mela 2"                 "Myrcia sulfiflora"
[29] "myrtaceae1"             "myrtaceae2"
[31] "Myrtaceae 3"            "myrtaceae4"
[33] "Pera glabrata"          "Persea sp"
[35] "Pisonia sp"             "Psidium sp"
[37] "Simplocos sp"          "Solanum sp1"
[39] "Solanum sp2"            "Syagrus romanzoffianus"
[41] "Tabebuia 1"             "Tabebuia cassinoides"
[43] "Tibouchina nutticeps"
```

```

$class
[1] "table"

>

```

A função 'table' produz um objeto da classe 'table'. No exemplo acima, o resultado é uma **tabela unidimensional**, pois apenas uma variável foi dada como argumento.

O objeto da classe 'table' tem como atributos :

- 'dim' que indica as dimensões da tabela;
- 'dimnames' que indica os nomes associados às frequências em cada dimensão da tabela.

As tabelas podem ser **multidimensionais** :

```

> x = round(runif(100) * 10)
> y = round(runif(100) * 10)
> x = round(runif(100) * 10)
> z = round(runif(100) * 10)
> tab3 <- table(x, y, z)
> class(tab3)
[1] "table"
> attributes(tab3)
$dim
[1] 11 11 11

$dimnames
$dimnames$x
 [1] "0" "1" "2" "3" "4" "5" "6" "7" "8" "9" "10"

$dimnames$y
 [1] "0" "1" "2" "3" "4" "5" "6" "7" "8" "9" "10"

$dimnames$z
 [1] "0" "1" "2" "3" "4" "5" "6" "7" "8" "9" "10"

$class
[1] "table"

>

```

1.1.2 Classe da Classe

A maneira mais direta é testar o que acontece quando fazemos o comando 'class(class(x))' :

```

> x = 1:10
> class(x)
[1] "integer"
> class( class(x) )
[1] "character"
>
> y = c("maão", "laranja", "uva")
> class( y )
[1] "character"
> class( class( y ) )
[1] "character"
>

```

```
> z = 3 >= 1:10
> class( z )
[1] "logical"
> class( class( z ) )
[1] "character"
>
```

Note que a classe que a função 'class' retorna é sempre 'character', pois trata-se de um **nome** (palavra), quer seja, o nome da classe do objeto argumento.

1.1.3 Logaritmo na Base 2

```
#
# Função redundante com a função log2
#
logdois <- function(x) log(x, base=2)
```

1.1.4 Somatório dos n Primeiros Números Naturais

```
#
# Versão mais direta da somatório (força bruta)
#
soma.n <-function(x) sum(1:x)
#
# Versão de Frederich Gauss (com 12 anos de idade)
#
soma.gauss <-function(x) x*(x+1)/2
```

1.1.5 Índices de Dispersão I

```
#
# Razao da Variancia Media
#
varmed <- function(x)
{
    var(x) / mean(x)
}
#
# Coeficiente de Green
#
green <- function(x)
{
    vm <- varmed(x)
    (vm - 1) / (sum(x) - 1)
}
#
# Índice de Morisita
#
morisita <- function(x)
{
    n <- length(x)
    yu <- sum(x^2) - sum(x)
    yd <- (sum(x))^2 - sum(x)
    n * yu / yd
}
```



```
}

```

1.1.6 Gráfico de Whittake

```
#
# Gráfico de Whittaker
#
whittaker <- function(sp)
{
  abund <- sort(table(sp), decreasing=TRUE)
  n <- length(abund)
  plot(1:n, abund,
       type="l",
       xlab="Rank de Abundância das Espécies",
       ylab="Abundância",
       log="y",
       las = 1
      )
}

```

1.1.7 Editando Funções Externamente

```
> # Para saber qual é o editor padrão do R:
> getOption( "editor" )
[1] "vi"
>
> # Para mudar o editor padrão do R:
> options( editor = "gedit" )
>
> # Agora basta evocar a função 'edit'
> # para editar um arquivo externo usar o argumento 'file'
> edit( file="minhas-funcoes.R" )
NULL
>

```

Ao finalizar a edição externa do arquivo 'minhas-funcoes.R', o R automaticamente lê o arquivo usando o comando 'source'.

1.1.8 Índices de Diversidade de Espécies

```
#
# Índice de diversidade de Shannon
#
shannon <- function(sp)
{
  densi <- table( sp )
  p <- densi / sum(densi)
  shannon <- - sum( p * log(p) )
  shannon
}
#
# Índice de Diversidade de Simpson
#
simpson <- function(sp)

```

```
{
  densi <- table( sp )
  p <- densi / sum(densi)
  simpson <- sum( p^2 )
  simpson
}
```

1.1.9 Loop para Representar o TCL

Em termos simples, o Teorema Central do Limite (TCL) mostra que independentemente da distribuição de uma variável aleatória, digamos X , a média amostral de X tenderá à distribuição Normal à medida que o tamanho da amostra tende ao infinito. A distribuição da média amostra terá a mesma média de X , mas a variância será a variância de X dividida pelo tamanho do amostra.

Como exemplo, assumamos que X tem distribuição exponencial com média 1. Vamos simular 1000 amostras de tamanhos $n1=10$, $n2=100$ e $n3=1000$

```
> # Gera 1000 amostras aleatórias da exponencial com tamanhos 10, 100 e 1000
> x1 = matrix( rexp(10 * 1000), ncol=1000 )
> x2 = matrix( rexp(100 * 1000), ncol=1000 )
> x3 = matrix( rexp(1000 * 1000), ncol=1000 )
>
> # Calcula as médias das 1000 amostras aleatórias dos vários tamanhos
> m1 = apply(x1, 2, mean)
> m2 = apply(x2, 2, mean)
> m3 = apply(x3, 2, mean)
>
> # Gráfico com as densidades dos três tamanhos de amostra
> plot( density(m3) , col="blue" , xlim=c(0,3))
> lines( density(m2) , col="red" )
> lines( density(m1) , col="darkgreen" )
```

1.1.10 Tabela de Fitossociologia

```
fitosoc <- function(df)
{
# O argumento 'df' é um data.frame que contém as variáveis:
#   'sp' - espécie
#   'dap' - diâmetro das árvores
#   'parcela' - parcela
#
# Extrair as variáveis do data.frame
  sp <- df[, "sp"]
  dap <- df[, "dap"]
  parcela <- df[, "parcela"]

# Calcular a Densidade
  densi <- aggregate( sp, list(especie=sp), function(x) length(x) )
  names(densi)[2] <- "densidade"
  densi$densidade.r <- densi$densidade / sum(densi$densidade) * 100

# Calcular a Dominancia
  domi <- aggregate( (pi/40000)*dap^2, list(especie=sp), sum)
  names(domi)[2] <- "dominancia"
  domi$dominancia.r <- domi$dominancia / sum(domi$dominancia) * 100

# Calcular a Frequencia
```

```
freq <- aggregate( parcela, list(especie=sp), function(x){length(unique(x))})
names(freq)[2] <- "frequencia"
freq$frequencia.r <- freq$frequencia / sum( freq$frequencia) * 100

# Juntar os cálculos
out <- cbind( densi, domi[,-1], freq[,-1] )
out$ivi <- out$densidade.r + out$dominancia.r + out$frequencia.r
out <- out[ order(out$ivi, decreasing=TRUE), ]

out
}
```

<http://lmq.esalq.usp.br/wiki/>