

BIE5782

Unidade 3:

**OBJETOS DE DADOS
E SUA MANIPULAÇÃO**

DESAFIOS

1. Importar os dados
2. Manipular os dados

Salvar como texto

The screenshot shows Microsoft Excel in Compatibility Mode. The spreadsheet contains the following data:

	A	B	C	D	E	F	G
1		nascimento	estado	vivo	altura		
2	Didi	1936	CE	s	1.68		
3	Dedé	1936	RJ	s	1.79		
4	Mussum	1941	RJ	n	1.81		
5	Zacarias	1934	MG	n	1.61		

The 'Salvar como' dialog box is open, showing the file path: Ale_2009 > AulasUSP > Introd_R > Aulas > dados. The file name is 'trapa.xls' and the type is 'Texto (separado por tabulações) (*.txt)'. The 'Authors' list includes various Excel file formats, with 'Texto (separado por tabulações) (*.txt)' selected.

Arquivo .txt ou .csv

The image shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		nascimento	estado	vivo	altura														
2	Didi	1936	CE	s	1.68														
3	Dedé	1936	RJ	s	1.79														
4	Mussum	1941	RJ	n	1.81														
5	Zacarias	1934	MG	n	1.61														

Two Notepad windows are overlaid on the spreadsheet:

- trapa.txt - Notepad** displays the data in a tab-separated format:

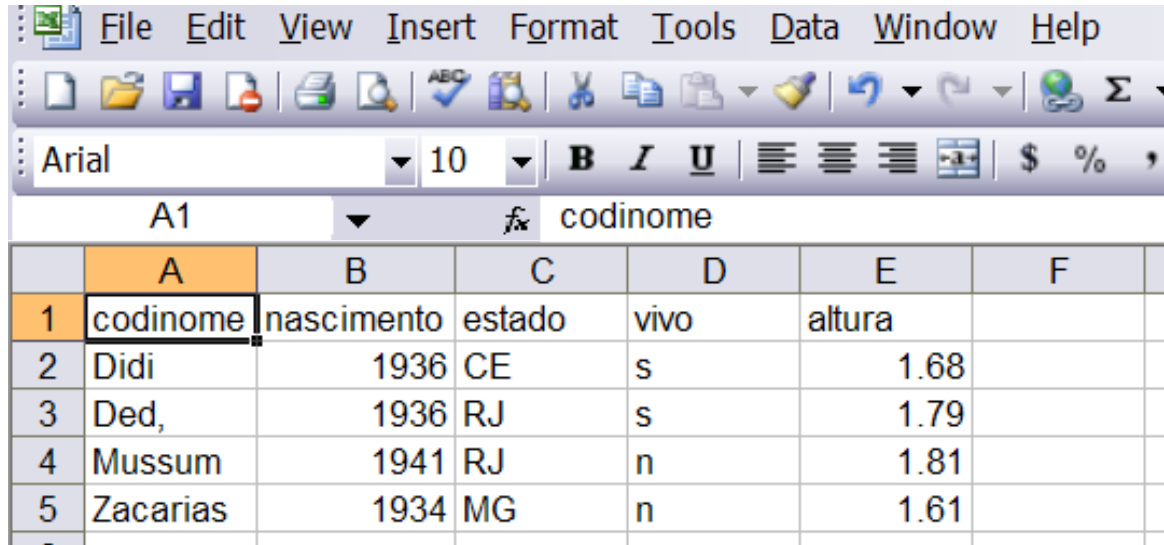
```
nascimento estado vivo altura
Didi 1936 CE s 1.68
Dedé 1936 RJ s 1.79
Mussum 1941 RJ n 1.81
Zacarias 1934 MG n 1.61
```
- trapa.csv - Notepad** displays the data in a comma-separated format:

```
,nascimento,estado,vivo,altura
Didi,1936,CE,s,1.68
Dedé,1936,RJ,s,1.79
Mussum,1941,RJ,n,1.81
Zacarias,1934,MG,n,1.61
```

The Excel interface shows the ribbon with tabs for Início, Inserir, Layout da Página, Fórmulas, Dados, Revisão, Exibição, and Suplementos. The status bar at the bottom indicates 'Pronto' and a zoom level of 100%.

read.table

Leitura de Arquivos-texto



	A	B	C	D	E	F
1	codinome	nascimento	estado	vivo	altura	
2	Didi	1936	CE	s	1.68	
3	Dedé	1936	RJ	s	1.79	
4	Mussum	1941	RJ	n	1.81	
5	Zacarias	1934	MG	n	1.61	

```
> trapa <- read.table("trapalhoes.csv",  
header=T, sep=",")
```

```
> trapa  
codinome      nascimento estado vivo  
Didi          1936      CE      s  
Dedé          1936      RJ      s  
Mussum        1941      RJ      n  
Zacarias      1934      MG      n
```

Padrão dos dados no wikisite

Conjunto de Dados: Dados de Biomassa de Árvores de Eucalyptus Saligna [Uso da Linguagem R para Análises de dados Ecológicos] - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://ecologia.ib.usp.br/bie5782/doku.php?id=dados:dados-esaligna

Comenzar a usar Fire... Últimas notícias

Google Pesquisa Pesquisar Favoritos Verificar Traduzir Enviar para Configurações

LABTROP [Laboratório de Ecologia de ...] Conjunto de Dados: Dados de Bio... Floresta da Amazônia Central [Labora... dokuwiki [DokuWiki]

[[Conjunto de Dados: Dados de Biomassa de Árvores de Eucalyptus Saligna]]


USO DA LINGUAGEM R PARA ANÁLISES DE DADOS ECOLÓGICOS

Visitou: » Arquivos de Apoio » Área Restrita » Curso 2009 » Conjunto de Dados: Levantamento em Caixetais » Índice » Apresentação » Arquivos de Dados » Conjunto de Dados: Dados de Biomassa de Árvores de Eucalyptus Saligna


[Editar esta página](#) [Revisões anteriores](#) [Administrar](#) [Atualizar o perfil](#) [Alterações recentes](#) [Índice](#) [Sair](#)

Pesquisar

Índice



- ▼ BIE 5782
 - ▼ Curso 2009
 - ▶ Área dos Alunos
 - ▶ Arquivos de Apoio
 - ▼ Tutoriais
 - ▼ Apostila On-Line
 - ▼ Área Restrita
- ▼ bRog
- ▼ Arquivos de Dados
- ▼ DokuWiki



Uso da Linguagem R para Análise de Dados em Ecologia

Conjunto de Dados: Dados de Biomassa de Árvores de Eucalyptus Saligna

Dados de biomassa de árvores de *Eucalyptus saligna* que foram abatidas e medidas: [arquivo CSV](#) (apagar extensão .pdf). [Editar](#)

Visão das primeiras linhas do arquivo CSV

```
"arvore","classe","talhao","dap","ht","tronco","sobra","fol
6,"c",22,19.9,21.5,183.64,20.42,8.57,212.64
8,"b",23,12.4,15.74,42.29,6.58,2.52,51.4
7,"c",32,16.5,11.74,60.61,11.35,48.52,120.49
8,"a",32,9,7.72,12.28,9.99,27.67,49.95
9,"a",32,7,6.55,11.86,7.97,7.76,27.61
9,"b",32,10.5,8.79,26.1,7.48,23.36,56.95
1,"c",22,13,12.86,47.84,12.14,2.33,62.32
2,"c",22,20,20.05,183.07,32.18,14.56,229.83
1,"a",22,7,11.6,16.19,1.81,0.65,18.66
2,"a",23,6.3,6.36,6.4,5.92,0.93,13.27
1,"b",23,15.5,18.2,103.39,14.17,5.26,122.83
2,"b",23,8.5,10.96,13.71,3.28,0.9,17.91
3,"a",18,7,11.3,6.11,8.23,1.12,15.47
```

Concluído

LEARNING FOR A COMPLEX WORLD



O mundo é mais complexo

1. Nem sempre o padrão é aquilo com que vcs. irão ter! ele pode ser a exceção.
2. Não estamos aqui para reproduzir ações, apertar botões todos já sabem!
3. Precisamos entender as complexidades existentes para saber confrontá-las.
4. A comunidade R já se confrontou com essas complexidades, portanto vamos tirar proveito!
5. Como tirar proveito?

read.table

Entender Argumentos

```
> trapa <- read.table("trapa.csv", header=T,  
sep="," , row.names=1)
```

```
> trapa <- read.table("trapa_nomes.txt",  
header=T, sep="\t", as.is=T)
```

USE O HELP!!

Para os argumentos de “read.table” consulte a ajuda. Faça o mesmo para para “write.table”.

Go to R

Símbolo do decimal

“ , ” OU “ . ”

The screenshot shows Microsoft Excel with a CSV file named 'trapaVirgula.csv' open in compatibility mode. The spreadsheet contains the following data:

	A	B	C	D	E
1		nascimento	estado	vivo	altura
2	Didi	1936	CE	s	1,68
3	Dedé	1936	RJ	s	1,79
4	Mussum	1941	RJ	n	1,81
5	Zacarias	1934	MG	n	1,61

Two Notepad windows are overlaid on the spreadsheet:

- trapaVirgula.txt - Notepad**: Shows the original CSV data with commas as decimal separators.

```
File Edit Format View Help
nascimento estado vivo altura
Didi 1936 CE s 1,68
Dedé 1936 RJ s 1,79
Mussum 1941 RJ n 1,81
Zacarias 1934 MG n 1,61
```
- trapaVirgula.csv - Notepad**: Shows the corrected CSV data with semicolons as decimal separators.

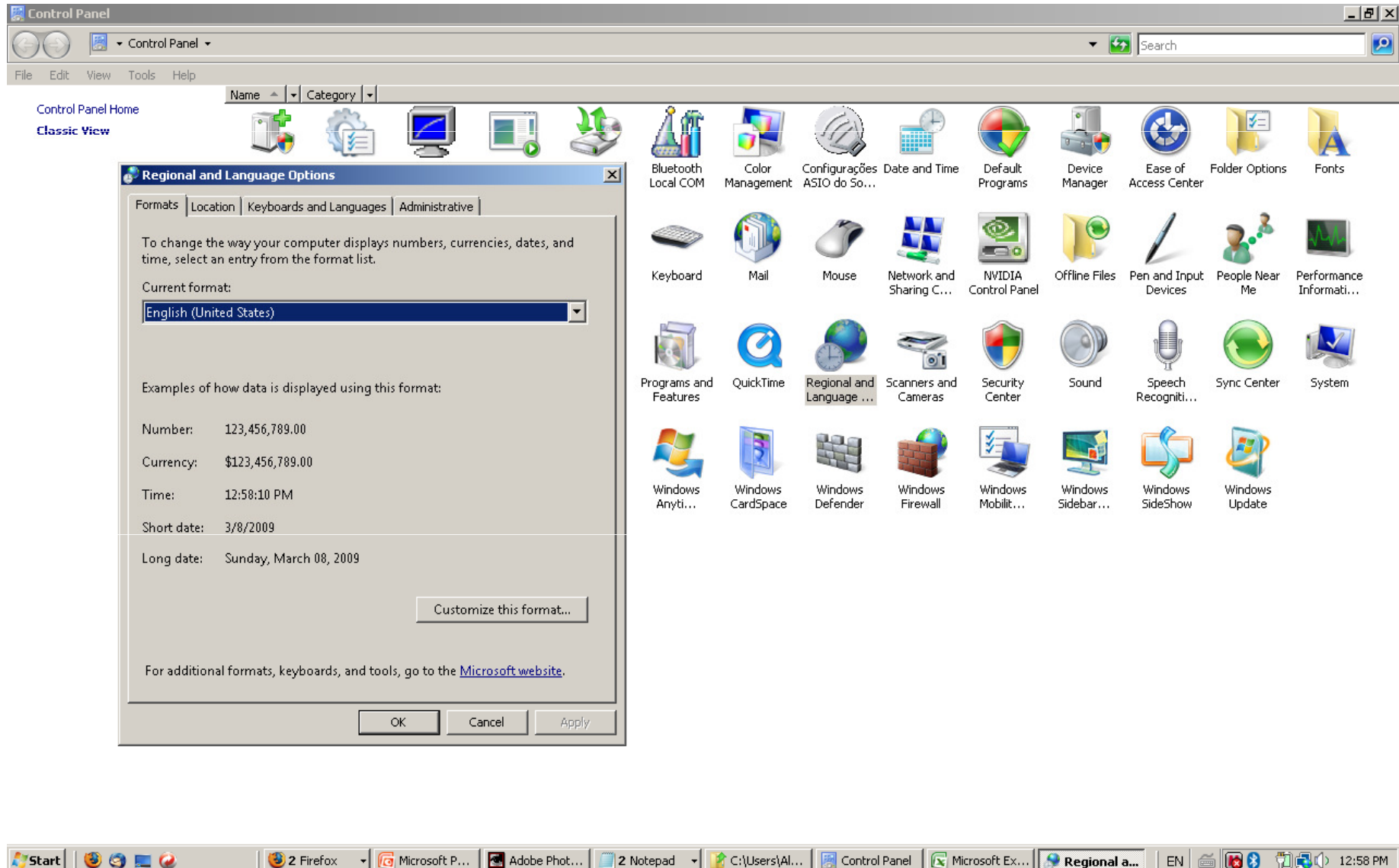
```
File Edit Format View Help
;nascimento;estado;vivo;altura
Didi;1936;CE;s;1,68
Dedé;1936;RJ;s;1,79
Mussum;1941;RJ;n;1,81
Zacarias;1934;MG;n;1,61
```

O que acontece com decimal “,”

- A seguir, Os Trapalhões em:
Detonando total como “,”

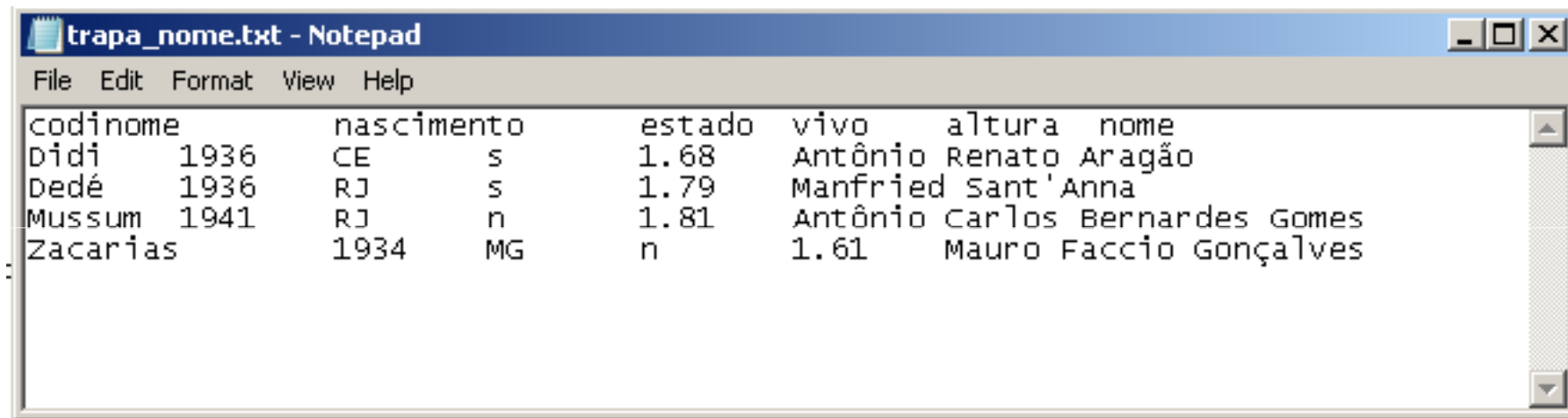
Go to R

Modificar o padrão decimal no computador



read.table

Padrão Prof. Alexandre



codinome	nascimento	estado	vivo	altura	nome
Didi	1936	CE	s	1.68	Antônio Renato Aragão
Dedé	1936	RJ	s	1.79	Manfried Sant'Anna
Mussum	1941	RJ	n	1.81	Antônio Carlos Bernardes Gomes
Zacarias	1934	MG	n	1.61	Mauro Faccio Gonçalves

```
>trapa.ale= read.table("trapa_nomes.txt",  
header=T, sep="\t", dec=".", as.is=T)
```

Leitura dos dados

DICAS

1. Estabelecer um padrão para salvar os dados

- qual o separador txt?
- qual símbolo decimal?
- variáveis com nomes?
- classes de variáveis automático?
- registros de dados com nomes?

2. Se algo não está funcionando, após a leitura, confira se o arquivo foi lido corretamente, use comandos:

```
str() ; head() ; dim()
```

Leitura dos dados

DICAS continuação...

3. Coloque o arquivo de leitura no diretório de trabalho do R!

```
getwd()
```

```
setwd()
```

4. Caso não queira fazer cópia de arquivos com localização complicada, use:

```
choose.file()
```


names, rownames

Nomes de “Colunas” e de Linhas

```
> names(trapa)
[1] "codinome" "nascimento" "estado"
"vivo" "nome"
> names(trapa) <- c("COD", "NASC", "ESTADO",
+ "VIVO", "NOME")
> trapa
  COD      NASC ESTADO  VIVO  NOME
Didi   1936      CE     s Renato...
Dedé   1936      RJ     s ..
Mussum 1941      RJ     n ..
Zacarias 1934      MG     n ..
> rownames(trapa) <- paste("trap", 1:4)
[1] "trap1"      "trap2"      "trap3"
[2] "trap4"
```

\$

Seleção de “Colunas”

```
> trapa$nascimento  
[1] 1936 1936 1941 1934
```

```
> trapa$vivo  
[1] s s n n  
Levels: n s
```

```
> trapa$estado  
[1] CE RJ RJ MG  
Levels: CE MG RJ
```

\$ com atribuição

Substituição e Criação de “Colunas”

```
> trapa$VIVO <- c("T", "T", "F", "F")
```

```
> trapa$ID.2009 = 2009 - trapalhoes$NASC
```

```
> trapa
```

	NASC	ESTADO	VIVO	ID.2008
Didi	1936	CE	T	73
Dedé	1936	RJ	T	73
Mussum	1941	RJ	F	68
Zacarias	1934	MG	F	75

`class()`

Classes de Objetos

```
> class(trapa)
[1] "data.frame"
> class(trapa$NASC)
[1] "integer"
> class(trapa$VIVO)
[1] "character"
> class(trapa$ESTADO)
[1] "factor"
```

O objeto “trapa” é um objeto da classe “data.frame”, que são conjuntos de vetores de mesmo comprimento, mas que podem ser de classes diferentes.

List ()

Cria Objeto da Classe “Lista”

```
> a
[1] 1 2 3 4 5
> b
[1] a a a b b b c c c
Levels: a b c
> c
  sec inicio
1 XIX    1801
2  XX    1901
3 XXI    2001

> minha.lista <- list (um.vetor=a, um.fator=b,
+ um.data.frame=c)
```

Listas (cont.)

```
> minha.lista
```

```
$um.vetor
```

```
[1] 1 2 3 4 5
```

```
$um.fator
```

```
[1] a a a b b b c c c
```

```
Levels: a b c
```

```
$um.data.frame
```

```
  sec inicio
```

```
1 XIX  1801
```

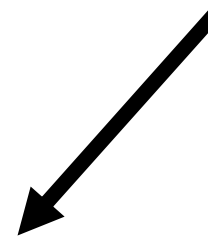
```
2  XX  1901
```

```
3 XXI 2001
```

```
> minha.lista$um.data.frame$inicio
```

```
[1] 1801 1901 2001
```

Uma lista pode conter
outras listas
(**recursividade!**)



matrix()

Cria Objeto da Classe “Matriz”

```
> matrix(1:12, nrow=4, ncol=3)
```

```
      [,1] [,2] [,3]  
[1,]    1    5    9  
[2,]    2    6   10  
[3,]    3    7   11  
[4,]    4    8   12
```

```
> matrix(1:12, 4, 3, byrow=T)
```

```
      [,1] [,2] [,3]  
[1,]    1    2    3  
[2,]    4    5    6  
[3,]    7    8    9  
[4,]   10   11   12
```

matrix()

Cria Objeto da Classe “Matriz”

```
> ilhas=matrix(
round(sample(c(runif(36, 0, 6), rep(0, 4)))) , ncol
=8)
> ilhas
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,]    6    5    4    2    5    2    3    3
[2,]    3    2    4    3    1    5    1    1
[3,]    2    5    1    1    1    3    6    0
[4,]    2    6    0    2    0    3    4    0
[5,]    1    5    0    3    3    6    0    2
> colnames(ilhas) <- paste("ilha", 1:8)
> rownames(ilhas) <- paste("sp", 1:5)
```


matrix()

Cria Objeto da Classe “Matriz”

```
> ilhas
```

```
      ilha 1 ilha 2 ilha 3 ilha 4 ilha 5 ilha 6 ilha 7 ilha 8  
sp 1      6      5      4      2      5      2      3      3  
sp 2      3      2      4      3      1      5      1      1  
sp 3      2      5      1      1      1      3      6      0  
sp 4      2      6      0      2      0      3      4      0  
sp 5      1      5      0      3      3      6      0      2
```

> , < , == , != , & , | , ...

Operadores Lógicos

```
> ALTURA
```

```
[1] 1.85 1.78 1.92 1.63 1.81 1.55
```

```
> SEXO
```

```
[1] M M M F F F
```

```
Levels: F M
```

```
> ALTURA > 1.80
```

```
[1] TRUE FALSE TRUE FALSE TRUE FALSE
```

```
> homens.altos <- ALTURA >= 1.80 & SEXO ==  
"M"
```

```
> homens.altos
```

```
[1] TRUE FALSE TRUE FALSE FALSE FALSE
```

```
> class(homens.altos)
```

```
[1] "logical"
```

Soma de Vetores Lógicos?

```
> notas.dos.alunos  
[1] 6.0 5.1 6.8 2.8 6.1 9.0 4.3 10.4  
[9] 6.0 7.9 8.9 6.8 9.8 4.6 11.3 8.0  
[17] 6.7 4.5
```

##Quantos aprovados?

```
> sum(notas.dos.alunos>=5)  
[1] 14
```

##Qual a proporção de aprovados?

```
> sum(notas.dos.alunos>=5)  
+ /length(notas.dos.alunos)  
[1] 0.7777778
```

Um Exemplo: Totais Marginais

```
> ilhas
```

	ilha 1	ilha 2	ilha 3	ilha 4	ilha 5	ilha 6	ilha 7	ilha 8
sp 1	2	3	1	6	1	2	6	0
sp 2	0	3	3	1	6	5	6	2
sp 3	6	5	4	0	3	3	4	0
sp 4	5	5	0	3	3	6	2	0
sp 5	0	1	0	1	2	5	3	3

```
> ilhaVF <- ilhas > 0
```

```
> ilhas.vf
```

	ilha 1	ilha 2	ilha 3	ilha 4	ilha 5	ilha 6	ilha 7	ilha 8
sp 1	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
sp 2	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
sp 3	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE
sp 4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE
sp 5	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE

apply()

Um Exemplo: Totais Marginais

```
> ilhas.vf
```

	ilha 1	ilha 2	ilha 3	ilha 4	ilha 5	ilha 6	ilha 7	ilha 8
sp 1	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
sp 2	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
sp 3	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE
sp 4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE
sp 5	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE

```
##Quantas espécies por ilha
```

```
> apply(ilhas.vf, 2, sum)
```

ilha 1	ilha 2	ilha 3	ilha 4	ilha 5	ilha 6	ilha 7	ilha 8
3	5	3	4	5	5	5	2

```
##Quantas ilhas por espécie
```

```
> apply(ilhas.vf, 1, sum)
```

sp 1	sp 2	sp 3	sp 4	sp 5
7	7	6	6	6

Um Exemplo: Totais Marginais

```
> ilhas.vf
      ilha 1 ilha 2 ilha 3 ilha 4 ilha 5 ilha 6 ilha 7 ilha 8
sp 1   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  FALSE
sp 2  FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
sp 3   TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  FALSE
sp 4   TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  FALSE
sp 5  FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
```

##média de espécies por ilha

```
> mean(apply(ilhas.vf, 2, sum))
[1] 4
```

```
> summary(apply(ilhas.vf, 2, sum))
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.0    3.0    4.5    4.0    5.0    5.0
```

E lá vem MAIS história!

SUBCONJUNTOS E INDEXAÇÃO

rank() , sort() , order()

Ordenação

```
> ALTURA
```

```
[1] 1.85 1.78 1.92 1.63 1.81 1.55
```

```
> rank (ALTURA)
```

```
[1] 5 3 6 2 4 1
```

```
> sort (ALTURA)
```

```
[1] 1.55 1.63 1.78 1.81 1.85 1.92
```

```
> order (ALTURA)
```

```
[1] 6 4 2 5 1 3
```

```
> ALTURA[order (ALTURA) ]
```

```
[1] 1.55 1.63 1.78 1.81 1.85 1.92
```

```
> ALTURA[c (6, 4, 2, 5, 1, 3) ]
```

```
[1] 1.55 1.63 1.78 1.81 1.85 1.92
```


Operador Colchetes []

Indexação de Vetores

```
> x
[1] "A" "B" "C" "D" "E" "F"
> x[1]
[1] "A"
> x[length(x)]
[1] "F"
> x[1:3]
[1] "A" "B" "C"
> x[c(1, 1, 3, 5)]
[1] "A" "A" "C" "E"
> x[-2]
[1] "A" "C" "D" "E" "F"
> x[-c(2, 4)]
[1] "A" "C" "E" "F"
```

Indexação com Operações Lógicas

```
> ALTURA
[1] 1.85 1.78 1.92 1.63 1.81 1.55
> PESO
[1] 80 100 115 70 65 50
> SEXO
[1] M M M F F F
Levels: F M
> homens.altos <- ALTURA >1.80 & SEXO ==
"M"
> homens.altos
[1] TRUE FALSE TRUE FALSE FALSE FALSE
> PESO[homens.altos]
[1] 80 115
> PESO[ALTURA >1.80 & SEXO == "M"]
[1] 80 115
```

Em uma matrix

```
> ilhas.vf
      ilha 1 ilha 2 ilha 3 ilha 4 ilha 5 ilha 6 ilha 7 ilha 8
sp 1   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  FALSE
sp 2   FALSE TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
sp 3   TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  FALSE
sp 4   TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  FALSE
sp 5   FALSE TRUE  FALSE TRUE  TRUE  TRUE  TRUE  TRUE

> riqueza=apply(ilhas.vf,2,sum)
> riqueza
ilha 1 ilha 2 ilha 3 ilha 4 ilha 5 ilha 6 ilha 7 ilha 8
      3      5      3      4      5      5      5      2

> riqueza.indice=order(riqueza,decreasing=T)
> riqueza.indice
[1] 2 5 6 7 4 1 3 8
```

Indexação [l,c]

```
> riqueza.indice=order(riqueza, decreasing=T)
```

```
> riqueza.indice
```

```
[1] 2 5 6 7 4 1 3 8
```

```
> ilhas[,riqueza.indice]
```

	ilha 2	ilha 5	ilha 6	ilha 7	ilha 4	ilha 1	ilha 3	ilha 8
sp 1	5	5	2	3	2	6	4	3
sp 2	2	1	5	1	3	3	4	1
sp 3	5	1	3	6	1	2	1	0
sp 4	6	0	3	4	2	2	0	0
sp 5	5	3	6	0	3	1	0	2

[] com atribuição

Alteração de Subconjuntos

```
> trapa
```

	codinome	nascimento	estado	vivo	altura
1	Didi	1936	CE	s	1.68
2	Dede	1936	RJ	s	1.79
3	Mussum	1941	RJ	n	1.81
4	Zacarias	1934	MG	n	1.61

```
>trapa[trapa$vivo=="s",]
```

	codinome	nascimento	estado	vivo	altura
1	Didi	1936	CE	s	1.68
2	Dede	1936	RJ	s	1.79

Indexação em mais de uma Dimensão

```
> trapa[c(1,2),1:4]
  codinome nascimento estado vivo
1     Didi         1936     CE    s
2     Dedé         1936     RJ    s

> trapa$vivo=="s"
[1] TRUE  TRUE FALSE FALSE
> which(trapa$vivo=="s")
[1] 1 2
> trapa$vivo[which(trapa$vivo=="s")]<-T
>trapa$vivo[which(trapa$vivo=="n")]<-F
> trapa$vivo
[1] "TRUE"  "TRUE"  "FALSE" "FALSE"
```

FIM DA AULA 3

Para amanhã:

Leia o texto do Wiki e
faça os exercícios da aula 3

<http://ecologia.ib.usp.br/bie5782>

IMPORTANTE: O texto do Wiki tem outras informações necessárias para os exercícios, que não foram detalhadas nesta apresentação.