

BIE5782

Aula 1 :

APRESENTAÇÃO DA DISCIPLINA
E DA LINGUAGEM R

ROTEIRO

1. Estrutura e funcionamento da disciplina
2. Motivações para aprender R
3. Estrutura básica do R:
 1. Sintaxe dos comandos
 2. Obtendo ajuda
 3. Orientação a objeto
 4. Organizando e gravando seu trabalho no R
 5. Manipulações básicas de objetos
 6. Criação de objetos simples e suas classes

Uso da Linguagem R em Ciências Biológicas e Ambientais



Alexandre Adalardo de Oliveira (IB-USP)
João Batista Ferreira (ESALQ-USP)
Rodrigo Santinello Pereira (FFCLRP-USP)
Paulo Inácio Prado (IB-USP)

BIE5782

Uso da Linguagem R para Análise de Dados em Ecologia



Alexandre



Paulo Inácio



Rodrigo



Barbara
(Eco)



Fábio
(Zoo)



Guilherme
(Genética)



Gustavo
(Botânica)



Tatiana
(Fisio)

BIE5782

Estratégia de Ensino

- Aulas teóricas introdutórias
- Tutoriais com a presença da equipe
- Estudo (individual ou grupo):
 - Leia a apostila
 - Conclua os tutoriais
 - Faça o exercícios
 - Busque e dê auxílio no fórum

BIE5782

Avaliações

- Exercícios:
 - Postar no wiki até a aula seguinte.
- Trabalho final:
 - Criar uma função em R e sua página de ajuda.
 - Postar proposta no wiki até 29/03
 - Postar trabalho no wiki até 09/04

BIE5782

Wiki

Índice



▼ BIE 5782

▼ Curso 2010

▼ Área dos Alunos

▶ Trabalho Final

▶ Arquivos de Apoio

▼ Tutoriais

▼ Apostila on line

▼ Cursos Anteriores

▶ Alunos 2009

▶ bRog 2009

▼ Arquivos de Dados

▼ DokuWiki

Forum 2010

Forum R

[[Curso 2010]]

USO DA LINGUAGEM R PARA ANÁLISES DE DADOS ECOLÓGICOS

Visão: » Apresentação » Alunos 2009 » Barbara Henning » Talita M. Zupo » Trabalho Final » Tatiana Hideko Kawamoto » Trabalho Final » Arquivos de Apoio » Área dos Alunos » Curso 2010

Mostrar código fonte Revisões anteriores

Alterações recentes Índice Autenticar-se

Pesquisar

Índice



- ▼ BIE 5782
 - ▼ Curso 2010
 - ▶ Área dos Alunos
 - ▶ Arquivos de Apoio
 - ▼ Tutoriais

Curso 2010

Tabela de conteúdos

- Índice
- Fórum 2010
- Linques

Bem vindo(a). Aqui você encontrará a programação de nossa disciplina, ministrada em março de 2010 no Instituto de Biociências da USP, e outras informações importantes. Além disto, há algumas atividades prévias a cumprir antes do início das aulas.

INFORMAÇÕES IMPORTANTES

INICIO DO CURSO Faça as atividades preparatórias!!! [Atividades Preparatórias](#)

Informações Gerais

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

O que é o R?

- Um ambiente de programação: sistema coerente e integrado.
- Ferramentas de software para manipulação de dados, cálculos e apresentação gráfica.
- Linguagem computacional S: flexibilidade e alta qualidade (Bell Labs.)
- Código aberto: projeto colaborativo
- Permite implementação de muitas técnicas estatísticas, entre outras coisas.

Por que uma disciplina de R?

"Uma das coisas mais importantes que você pode fazer é dedicar um tempo para aprender uma linguagem de programação de verdade.

Aprender a programar é como aprender outro idioma: exige tempo e treinamento, e não há resultados práticos imediatos. Mas se você supera essa primeira subida íngreme da curva de aprendizado, os ganhos como cientista são enormes.

Programar não vai apenas livrar você da camisa de força dos pacotes estatísticos, mas também irá aguçar suas habilidades analíticas e ampliar os horizontes de modelagem ecológica e estatística."

Tradução um tanto livre de Gotelli & Ellison, 2004. A Primer of Ecological Statistics. Sunderland, Sinauer. (Grifo nosso)

Um ambiente pouco amigável ...

```
R version 2.8.1 (2008-12-22)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> █
```

.. ou apenas pouco familiar ?

The image displays a collage of SPSS software windows. At the top, the 'SYSTAT Output Organizer' window shows a file tree with 'SYSTAT Output' and 'File C:\DOCL' containing a 'Plot TEST1'. Below it, the 'SYSTAT Graph' window shows a scatter plot of 'TEST2' with a y-axis from 80 to 100. The 'SPSS Data Editor' window shows a data table with columns 'ID', 'SEX\$', and 'TEST1'. A 'Syntax Editor' window is also visible. At the bottom, the 'STATISTICA Basic' window shows a flowchart of the software's architecture, with 'INPUT' leading to 'STATISTICA BASIC' and 'OUTPUT' leading to 'DATA FILES', 'SCROLLSHEETS', 'GRAPHS', 'DISK FILES', and 'TEXT & GRAPHICS REPORTS'.

Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align
2 status	Numeric	1	0		(1, Members)...	None	8	Right
3 q1a	Numeric	1	0	1a. I have access to an	(1, Strongly agr	0, 9	8	Right
4 q1b	Numeric	1	0	1b. There are sufficien	(1, Strongly agr	0, 9	8	Right
5 q1c	Numeric	1	0	1c. Provides Stewards	(1, Strongly agr	0, 9	8	Right
6 q1d	Numeric	1	0				8	Right
7 q1e	Numeric	1	0				8	Right
8 q1f	Numeric	1	0				8	Right

ID	SEX\$	TEST1
2	f	65
3	m	60
4	m	88
5	m	84

1a. I have access to any needed training.					
Valid		Frequency	Percent	Valid Percent	Cumulative Percent
1	Strongly agree	2209	90.2	91.1	91.1
2	Agree	193	7.9	8.0	99.0
3	Neutral	18	.7	.7	99.8
4	Disagree	4	.2	.2	99.9
5	Strongly disagree	2	.1	.1	100.0
Total		2426	99.1	100.0	
Missing	System	22	.9		
Total		2448	100.0		

Controle

" One of the objectives of statistical analysis is to distil a long and complicated set of data into a small number of meaningful descriptive statistics."

" Many of the modern computer statistical packages, however, do exactly the opposite of this." [Their] "copious output has several major shortcomings: it is open to uncritical acceptance; it can lead to over interpretation of data; and it encourages the bad habit of data trawling."

"R, by other hand, tells you nothing unless you explicitly ask for it."

Crawley, M. The R Book.



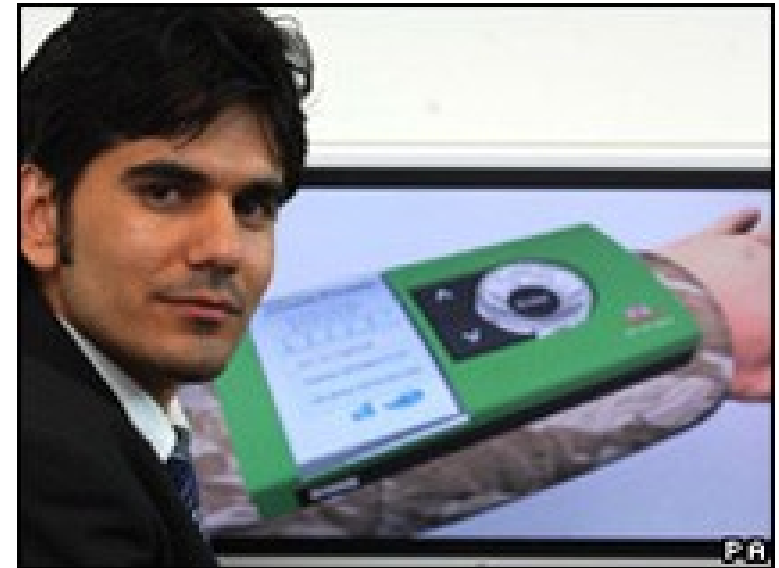
Mick Crawley

A Camisa de Força das Interfaces

O aparelho deve traduzir frases como "não atire", "não se aproxime" e "a ajuda vai chegar logo".

A idéia foi de Amin Ismail, um ex-estudante iraquiano que fugiu de seu país em 2002 e vive na Grã-Bretanha.

Ismail disse que o projeto é "uma maneira de agradecer aos soldados britânicos por seu trabalho honroso"



Flexibilidade

A pergunta certa sobre uma análise em R não é se é possível fazê-la, e sim como fazê-la.



Paulo Justiniano Ribeiro

Estrutura Básica da Linguagem R

```
R version 2.8.1 (2008-12-22)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> |
```



Sintaxe Básica das Funções

Isto é uma EXPRESSÃO



função(argumento1=valor, argumento2=valor, ...)

> plot(x=area, y=riqueza, log="xy")

> plot(area, riqueza)

> plot(area, riqueza, log="xy")

help , help.start e args

Ajuda sobre funções

Texto de Ajuda

```
> help(mean)  
> ?mean
```

Hipertexto de Ajuda

```
> help.start()
```

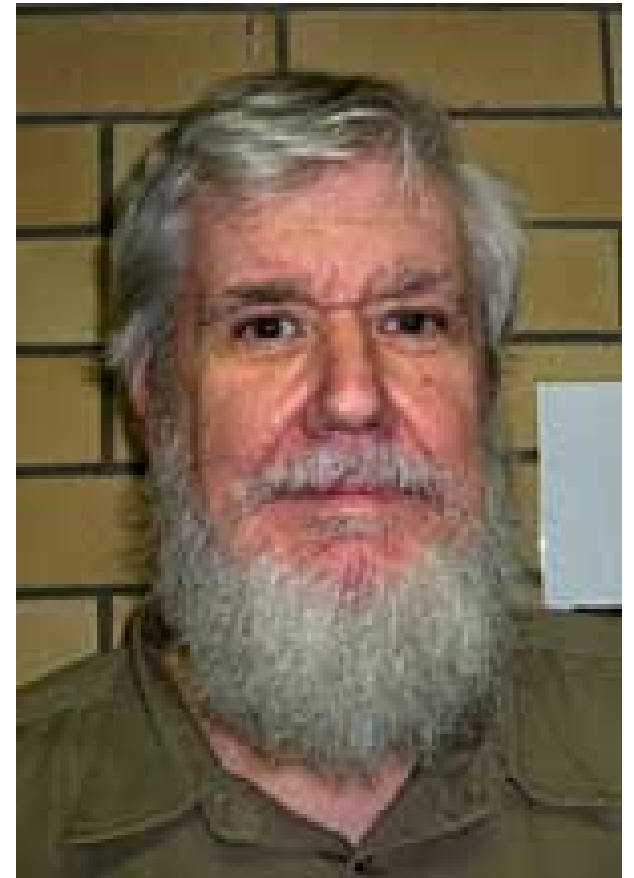
Argumentos de uma função

```
> args(lm)  
function (formula, data, subset, weights, na.action,  
method = "qr", model = TRUE, x = FALSE, y = FALSE,  
qr = TRUE, singular.ok = TRUE, contrasts = NULL,  
offset, ...)
```



A chave para entender o R é que trata-se de uma linguagem.

Uma linguagem para manipular objetos.



Bill Venables

Operadores `<-`, `=`, `->`

Objetos e Atribuição

objeto `<-` expressão

```
> area <- c(303, 379, 961, 295, 332, 47, 122, 11, 53, 2749)
```

objeto `=` expressão

```
> area = c(303, 379, 961, 295, 332, 47, 122, 11, 53, 2749)
```

Pouco Usado:

expressão `->` objeto

```
> c(303, 379, 961, 295, 332, 47, 122, 11, 53, 2749) -> area
```

Digite o nome do objeto para exibí-lo

```
> area
```

```
[1] 303 379 961 295 332 47 122 11 53 2749
```

```
> riqueza
```

```
[1] 3 10 20 7 8 4 8 3 5 23
```

Um erro comum

```
> help
function (topic, offline = FALSE, package = NULL,
lib.loc = NULL,
    verbose = getOption("verbose"), try.all.packages =
getOption("help.try.all.packages"),
    chmhelp = getOption("chmhelp"), htmlhelp =
getOption("htmlhelp"),
    pager = getOption("pager"))
{
    if (!missing(package))
        if (is.name(y <- substitute(package)))
            package <- as.character(y)
```



Funções são objetos, como tudo mais no R. Se você digita uma função sem os parênteses, o R entende que vc quer exibir seu conteúdo, no caso seu código.

Linguagem Orientada a Objetos

```
> area <- c(303, 379, 961, 295, 332, 47, 122, 11, 53, 2749)
```

```
> area
```

```
[1] 303 379 961 295 332 47 122 11 53 2749
```

```
> summary(area)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
11.00	70.25	299.00	525.20	367.20	2749.00

Linguagem Orientada a Objetos

```
> modelo1 <- lm(riqueza~area)
> summary(modelo1)
```

```
Call:
lm(formula = riqueza ~ area)
```

```
Residuals:
```

Min	1Q	Median	3Q	Max
-4.4614	-2.1245	-0.5101	1.4862	7.6863

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	5.226994	1.352158	3.866	0.00477	**
area	0.007374	0.001431	5.155	0.00087	***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 3.555 on 8 degrees of freedom
```

```
Multiple R-squared: 0.7686, Adjusted R-squared: 0.7397
```

```
F-statistic: 26.57 on 1 and 8 DF, p-value: 0.00086930
```


ls e rm: lista e remove objetos

```
> ls()
[1] "area" "modelo1" "modelo2" "previsto" "riqueza" "varea"

> rm(varea)

> ls()
[1] "area" "modelo1" "modelo2" "previsto" "riqueza"
```

Como o R trabalha?

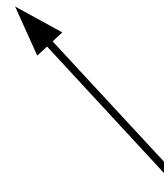
- Os objetos criados são mantidos na memória, até que sejam salvos.
- Por *default*, todos os objetos são salvos em um único arquivo oculto no diretório de trabalho, chamado **.RData**.
- Ao sair do programa com a função **q**, o R pergunta se deve salvar as alterações.
- É recomendável salvar seu trabalho regularmente, com o comando **save.image**

q :

Encerra a sessão do R

**Sai do programa mas salva a área de trabalho em
.RData (se a resposta for “y” !)**

```
> q()  
Save workspace image? [y/n/c]:  
y
```



Se a resposta for “n” você perde
todas as alterações !

save.image , **save** e **load**:

Salva e carrega a área de trabalho

Salva a área de trabalho no arquivo default (.RData)

```
> save.image()
```

Salva a área de trabalho em outro arquivo

```
> save(file="meuworkspace.RData")
```

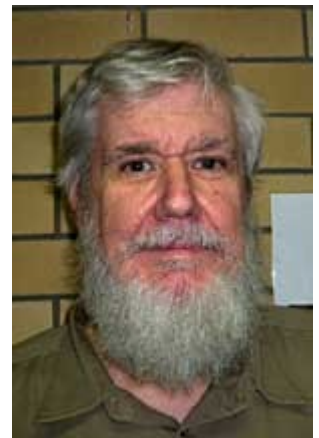
Carrega um arquivo de área de trabalho

```
> load(file="meuworkspace.RData")
```

Como Trabalhar com o R?

Dica 1: Separe cada projeto em um diretório

- R works best if you have a dedicated folder for each separate project – the working folder. Put all data files, &c, in the working folder (or in subfolders of it)
- Start R in the working folder: three ways
 - make an R shortcut pointing to the folder and double-click
 - double-click on the **.RData** file in the folder, when it exists
 - double-click any R shortcut and use **setwd()**
- Work on the project – your objects can be automatically saved in the **.RData** file
- To quit – use **q()**



getwd e setwd:

Mostra e muda o diretório de trabalho

Qual o diretório de trabalho atual?

```
> getwd()  
[1] "/home/paulo/work/Pos_grad/Eco_USP/cursoR/2008"
```

Muda o diretório de trabalho

```
> setwd("/home/paulo/work/treinos_R/")
```

Verificando ...

```
> getwd()  
[1]  
"/home/paulo/work/treinos_R"
```

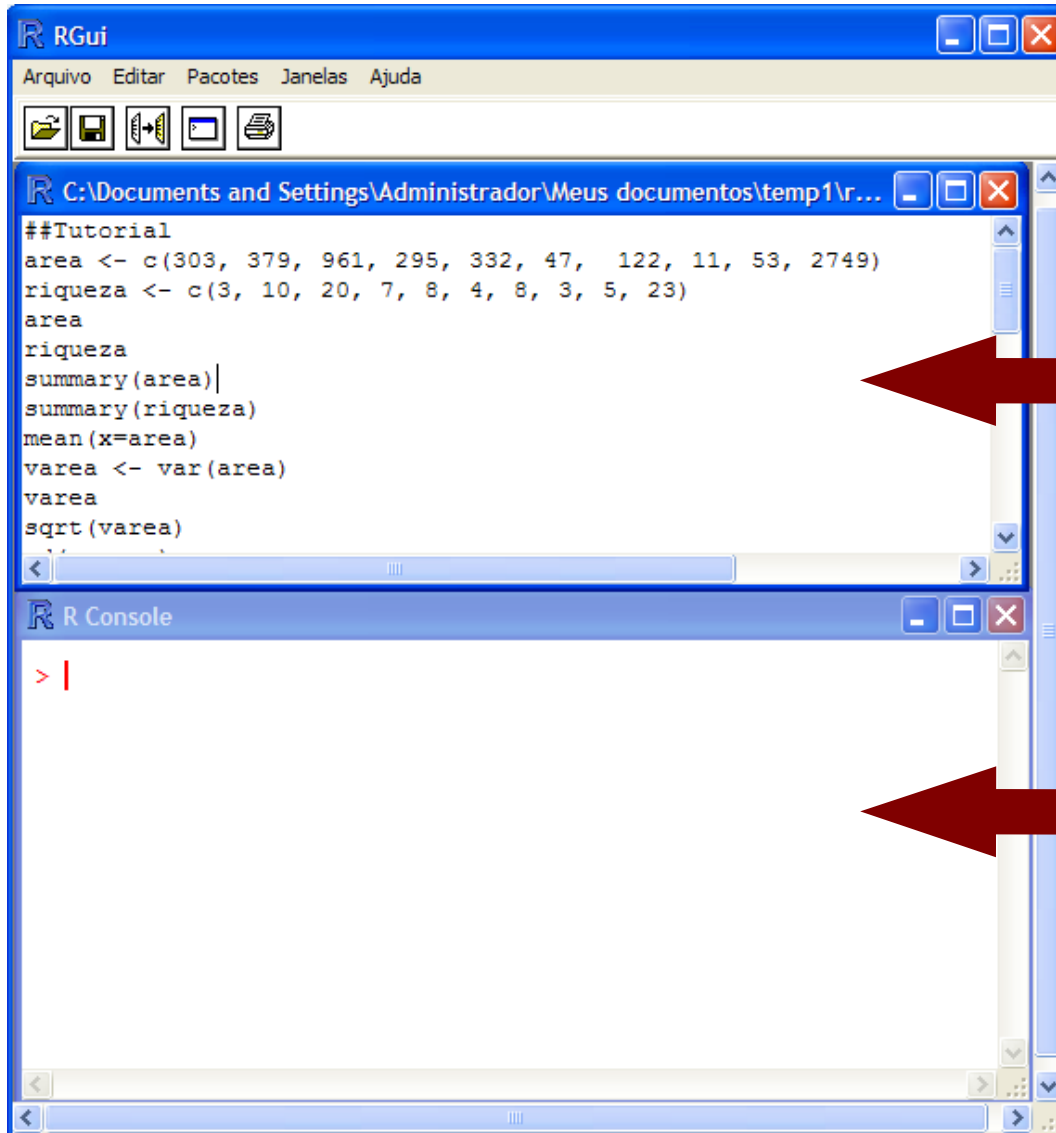
Como Trabalhar com o R?

Dica 2: salve seus comandos em um arquivo

- Use um editor de textos para escrever seus comandos em um arquivo.
- Salve este arquivo em seu diretório de trabalho com extensão `.r` ou `.R`
- Envie seus comandos ao R. Três maneiras:
 - Com o comando **source** ("arquivo.R"): executa todos os comandos.
 - Na interface R-GUI do Windows, use o editor de scripts e **ctrl-R** para enviar linhas ou blocos para.
 - Use editores próprios para programação em R (e.g., Emacs com módulo ESS, Tinn-R).

Como Trabalhar com o R?

Dica 2: salve seus comandos em um arquivo



The screenshot shows the RGui application window. The top menu bar includes 'Arquivo', 'Editar', 'Pacotes', 'Janelas', and 'Ajuda'. Below the menu is a toolbar with icons for file operations. The main window is divided into two panes. The top pane, titled 'R C:\Documents and Settings\Administrador\Meus documentos\temp1\r...', contains a script with the following R code:

```
##Tutorial
area <- c(303, 379, 961, 295, 332, 47, 122, 11, 53, 2749)
riqueza <- c(3, 10, 20, 7, 8, 4, 8, 3, 5, 23)
area
riqueza
summary(area)
summary(riqueza)
mean(x=area)
varea <- var(area)
varea
sqrt(varea)
```

The bottom pane, titled 'R Console', shows the R prompt '> |' with a vertical cursor.

Arquivo de código
(*script*)

Janela do R
(R- console)

Classes de Objetos

```
> class(modelo1)
[1] "lm"
```

```
> names(modelo1)
[1] "coefficients" "residuals"      "effects"        "rank"
[5] "fitted.values" "assign"         "qr"            "df.residual"
[9] "xlevels"      "call"          "terms"         "model"
```

O objeto `modelo1` pertence à classe `lm` (linear models), que tem resultados de uma regressão linear.

Esta classe de objeto contém outros objetos, como os vetores de valores previstos e de resíduos, a fórmula do modelo ajustado e muito mais.

Classes de Objetos

```
> x.1
[1] 1 2 3 4 5
> class(x.1)
[1] "integer"
> x.2
[1] 0.333333333 0.166666667 0.083333333
> class(x.2)
[1] "numeric"
> x.3
[1] "a" "b" "c" "d" "e"
> class(x.3)
[1] "character"
> x.4
[1] TRUE TRUE FALSE FALSE FALSE
> class(x.4)
[1] "logical"
```

is.[classe] e **as.[classe]**:

Verificação e Coerção de Classes

```
> x.1
[1] 1 2 3 4 5
> x.1t <- as.character(x.1)
> x.1t
[1] "1" "2" "3" "4" "5"
> x.5 <- c("TRUE", "TRUE", "TRUE", "FALSE")
> class(x.5)
[1] "character"
> is.logical(x.5)
[1] FALSE
> x.5 <- as.logical(x.5)
> class(x.5)
[1] "logical"
```

C

Criação de Vetores

```
> a = c(3.4, pi, exp(-1))  
> a  
[1] 3.40000000 3.1415927 0.3678794
```

- Um objeto da classe `vector` no R é um conjunto de elementos de uma mesma classe (números, caracteres, lógicos, etc).
- Um objeto vetor no R não corresponde aos vetores de álgebra matricial (para isso há a classe `matrix`).
- A função `c` combina elementos em um vetor.

factor

Classe de fatores

```
> sexo <- rep(c("F", "M"), each=9)
> sexo
[1] "F" "F" "F" "F" "F" "F" "F" "F" "F" "M"
[11] "M" "M" "M" "M" "M" "M" "M" "M" "M"
> class(sexo)
[1] "character"
> sexo <- factor(sexo)
> sexo
 [1] F F F F F F F F F M M M M M M M M M
Levels: F M
> class(sexo)
[1] "factor"
```

table

Contagens de Observações

```
> sexo
[1] F F F F F F F F F M M M M M M M M M
Levels: F M
> table(sexo)
sexo
F M
9 9
> dieta
[1] L L L D D D N N N L L L D D D N N N
Levels: N D L
> table(sexo,dieta)
      dieta
sexo  N  D  L
  F   3  3  3
  M   3  3  3
```

levels

Níveis de um fator

```
> politicos <- factor(rep("corrupto",10),  
                      levels=c("corrupto","honesto"))  
> levels(politicos)  
[1] "corrupto" "honesto"  
> politicos  
[1] corrupto corrupto corrupto corrupto corrupto  
[6] corrupto corrupto corrupto corrupto corrupto  
Levels: corrupto honesto  
> table(politicos)  
politicos  
corrupto  honesto  
      10      0
```



FIM DA PRIMEIRA PARTE

Em seguida:

Uso do wiki

Hoje à tarde:

Tutorial da Unidade 1 - Sala Multimídia

http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02_tutoriais:tut1

Postar no wiki até a próxima aula:

Lista 1 de Exercícios:

http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:01_curso_atual:exercicios1