

# 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

Equipe 2009:

Alexandre Adalardo de Oliveira

Paulo Inácio Prado

Cristina Banks-Leite

Página:

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

# BIE5782

Unidade 1 :  
APRESENTAÇÃO DA DISCIPLINA

Unidade 2 :  
INTRODUÇÃO À LINGUAGEM R

# 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 conhecido ?

The image displays a collage of software interfaces used for data analysis and visualization. The top window is 'Untitled - SYSTAT Output Organizer', showing a file tree with 'SYSTAT Output' and 'File C:\DOCL' containing a 'Plot TEST1'. Below it is the 'SYSTAT Graph' window, which displays a scatter plot of 'TEST2' with a y-axis ranging from 80 to 100. The plot shows several data points. To the right of the graph are controls for '3-D Rotation' and 'X-Power' and 'Y-Power', both set to 1.000.

The bottom-left window is 'SPSS Data Editor' (version 10 for Macintosh), showing a data table with columns 'Name', 'Type', 'Width', 'Decimals', 'Label', 'Values', 'Missing', 'Columns', and 'Align'. The data includes variables like 'status', 'q1a', 'q1b', 'q1c', 'q1d', 'q1e', and 'q1f'. A 'Syntax Editor' window is open over the table, showing a script that includes commands like 'end if.', 'loop.', 'select if status=1.', and 'freq /var=q1o.'. Below the table is a frequency table for '1a. I have access to any needed training.' with columns for 'Valid', 'Frequency', 'Percent', 'Valid Percent', and 'Cumulative Percent'.

The bottom-middle window is 'D - SYSTAT Data', showing a data table with columns 'ID', 'SEX\$', and 'TEST1'. The data includes rows for '2 f', '3 m', '4 m', and '5 m' with corresponding 'TEST1' values of 65, 60, 88, and 84.

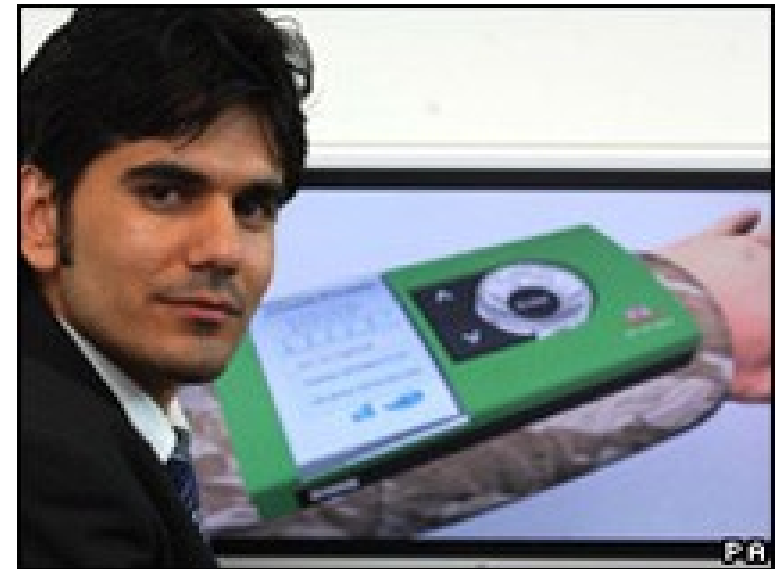
The bottom-right window is 'STATISTICA Basic', showing a workflow diagram with 'INPUT:', 'STATISTICA BASIC:', and 'OUTPUT:' sections. The 'STATISTICA BASIC:' section contains a script with commands like 'dim w(200), os(200);', 'scr:=-NewScrollSheet;', 'scrollsheet=distplot (4 for i=1 to 200 do os scatterplot of wa i);', 'distplot of read scrollsheetGoMater;', 'graph1:=-NewGraph (9C);', 'distplot of read scrollsheetGoMater;', 'graph2:=-NewGraph (LI);', and 'graph3=distplotLineStyle (some statistic o'. The 'OUTPUT:' section shows various output types like 'DATA FILES', 'SCROLLSHEETS', 'GRAPHS', 'DISK FILES', 'DISK', 'TEXT & GRAPHICS REPORTS', and 'INTERACTIVE INPUT/OUTPUT'.

# Interfaces x Linguagens

*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"



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



Paulo Justiniano Ribeiro

"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.



# Sintaxe Básica das Funções

**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

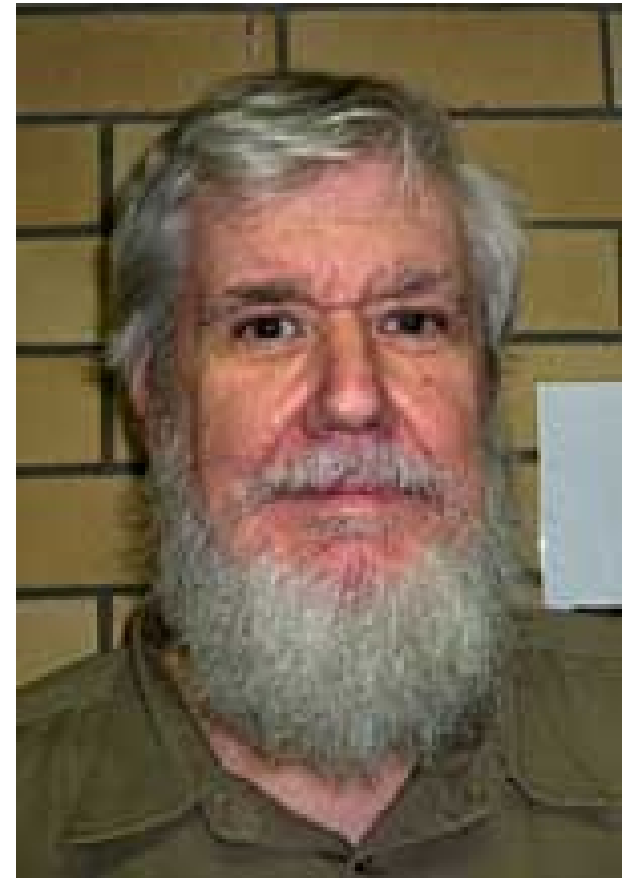
```
> help.start()
```

### 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?

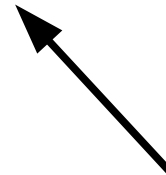
- Os objetos criados são mantidos na memória, até que sejam salvos.
- Por *default*, todos os objetos são salvos em um único arquivo oculto no diretório de trabalho, chamado **.RData**.
- Ao sair do programa com a função **q**, o R pergunta se deve salvar as alterações.
- É recomendável salvar seu trabalho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

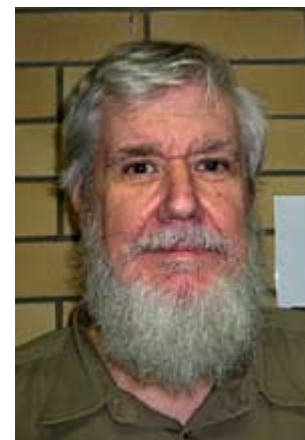
Carrega um arquivo de área de trabalho

```
> 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"
```

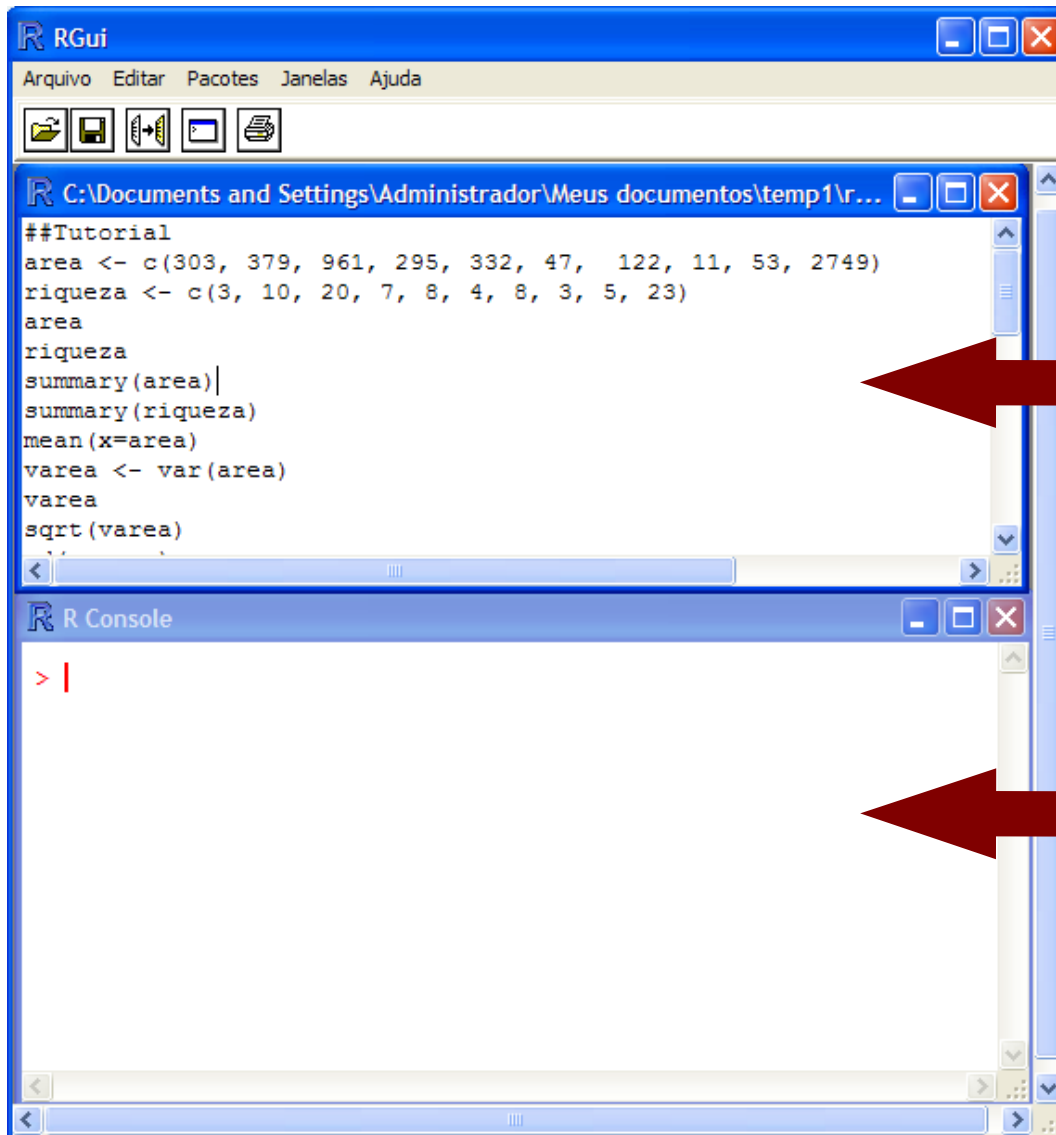
# Como Trabalhar com o R?

## 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 interface. The top window is titled 'R C:\Documents and Settings\Administrador\Meus documentos\temp1\r...' and contains 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 window is titled 'R Console' and shows the R prompt '> |'.

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 algebra 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" "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 DAS UNIDADES 1 e 2

Para a tarde:

## Tutorial da Unidade 1

[http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02\\_tutoriais:start](http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02_tutoriais:start)

## Lista 1 de Exercícios:

[http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:01\\_curso2009:exercicios1](http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:01_curso2009:exercicios1)