Introdução ao R: aula 5

Alexandre Rademaker

EMAp/FGV

February 2, 2011

Agenda

- Revisão
- Entrada e saída
- Programação

Importação de dados

- Excel to R: copy/paste
- Excel to R: `http://cran.r-project.org/web/packages/gdata/`
- CSV e TXT: read.table e write.table
  - Arquivos comprimidos ou não
  - Arquivos locais ou não
- `http://www.omegahat.org/RGoogleDocs/`
- `http://cran.r-project.org/web/packages/foreign/`
- `http://cran.r-project.org/web/packages/RODBC/`
- `http://en.wikipedia.org/wiki/NetCDF`

Mais sobre importação

Inspecionando o ambiente:

- ls, rm
- str
- summary

Mais sobre importação

Entrada e saída:

- dump
- dput
- load
- save, save.image
- source
- sink(file) e sink()

Tabelas

A função `table` conta combinações:

```
> idades <- cut(sample(1:70, 30), c(1, 15,
     24, 60, 70), c("criança", "adolescente",
     "adulto", "idoso"))
> sexo <- factor(sample(1:2, 30, replace = TRUE),
     levels = c(1, 2), labels = c("F",
         "M"))
> t <- table(idades, sexo)
```

Tabelas

```
> t
```

```
             sexo
idades         F  M
  criança      2  2
  adolescente  0  2
  adulto      13  5
  idoso        4  2
```

Inspecionando t:

```
> attributes(t)
> mode(t)
> class(t)
```

Tabelas

Margens:

```
> margin.table(t, 1)

idades
    criança adolescente      adulto       idoso
          4           2          18           6

> prop.table(t)

             sexo
idades                 F          M
  criança     0.06666667 0.06666667
  adolescente 0.00000000 0.06666667
  adulto      0.43333333 0.16666667
  idoso       0.13333333 0.06666667
```

Tabelas

Veja mais, addmargins:

```
> addmargins(t, margin = c(1, 2), FUN = mean,
     quiet = TRUE)

             sexo
idades            F     M  mean
  criança      2.00  2.00  2.00
  adolescente  0.00  2.00  1.00
  adulto      13.00  5.00  9.00
  idoso        4.00  2.00  3.00
  mean         4.75  2.75  3.75
```

Algumas funções úteis

```
> gl(2, 5, labels = c("nao", "sim"))
```

```
 [1] nao nao nao nao nao sim sim sim sim sim
Levels: nao sim
```

Dez números de uma distribuição normal com média 10 e desvio padrão 3.

```
> rnorm(10, mean = 10, sd = 3)
```

```
 [1] 14.384000  7.891334  5.560686  7.509737
 [5]  9.891460  9.025678 10.443924  7.376030
 [9]  9.922945  8.648665
```

Matrizes

```
> (m1 <- matrix(c(45, 23, 66, 77, 33, 44),
     2, 3))

     [,1] [,2] [,3]
[1,]   45   66   33
[2,]   23   77   44

> (m2 <- matrix(c(5, 3, 466, 54.5, 3.2,
     -34), 3, 2))

     [,1]  [,2]
[1,]    5  54.5
[2,]    3   3.2
[3,]  466 -34.0
```

Matrizes

```
> m1 %*% m2
      [,1]   [,2]
[1,] 15801 1541.7
[2,] 20850    3.9
```

Matrizes

```
> t(m1)
     [,1] [,2]
[1,]   45   23
[2,]   66   77
[3,]   33   44
```

Datas

```
> (hoje <- Sys.Date())

[1] "2011-02-02"

> format(hoje, "%d %b %Y")

[1] "02 Feb 2011"

> seq(hoje, len = 10, by = "1 week")

 [1] "2011-02-02" "2011-02-09" "2011-02-16"
 [4] "2011-02-23" "2011-03-02" "2011-03-09"
 [7] "2011-03-16" "2011-03-23" "2011-03-30"
[10] "2011-04-06"

> c(months(hoje), weekdays(hoje))

[1] "February"  "Wednesday"
```

Datas

```
> as.Date("2011-01-26") - as.Date("2010-04-02")

Time difference of 299 days

> ISOdate(2001, 1, 1) - ISOdate(2000, 6,
      14)

Time difference of 201 days

> table(cut(seq(ISOdate(2011, 1, 1), to = ISOdate(2011,
      12, 31), by = "day"), "month"))

2011-01-01 2011-02-01 2011-03-01 2011-04-01
        31         28         31         30
2011-05-01 2011-06-01 2011-07-01 2011-08-01
        31         30         31         31
2011-09-01 2011-10-01 2011-11-01 2011-12-01
        30         31         30         31
```

Datas

Dois tipos de datas/horas: (1) `POSIXct`, número de segundos desde 1970; e (2) `POSIXlt` datas/horas como uma lista dos componentes.

```
> as.POSIXlt(Sys.time(), "GMT")
```

```
[1] "2011-02-02 16:55:35 GMT"
```

```
> a <- as.POSIXct("2011-01-26 11:00")
> b <- as.POSIXct("2011-01-01 00:00")
> a - b
```

```
Time difference of 25.45833 days
```

```
> format(Sys.time(), "%a %b %d %X %Y %Z")
```

```
[1] "Wed Feb 02 14:55:35 2011 BRST"
```

Datas

Conversão de tipos:

```
> x <- c("1jan1960", "2jan1960", "31mar1960",
     "30jul1960")
> strptime(x, "%d%b%Y")
```

```
[1] "1960-01-01" "1960-01-02" "1960-03-31"
[4] "1960-07-30"
```

```
> dates <- c("02/27/92", "02/27/92", "01/14/92")
> times <- c("23:03:20", "22:29:56", "01:03:30")
> x <- paste(dates, times)
> strptime(x, "%m/%d/%y %H:%M:%S")
```

```
[1] "1992-02-27 23:03:20" "1992-02-27 22:29:56"
[3] "1992-01-14 01:03:30"
```

Séries temporais regulares

Com uma variável:

```
> ts(rnorm(10), start = 1990, frequency = 1)

Time Series:
Start = 1990
End = 1999
Frequency = 1
 [1]  0.007401613 -0.726940765 -0.469221845
 [4]  0.963798493 -0.598979238 -0.198941036
 [7]  0.322653622  0.720672402 -1.257902497
[10] -1.221034176
```

Séries temporais regulares

```
> (k <- ts(rnorm(10), frequency = 12, start = c(1959,
     2)))

             Feb         Mar         Apr
1959 -0.81482515 -1.90777088  0.67789554
             May         Jun         Jul
1959  0.66744498  1.50902163 -0.11849516
             Aug         Sep         Oct
1959  0.01063749  0.48308075 -1.78855045
             Nov
1959  1.27365121
```

Séries temporais regulares

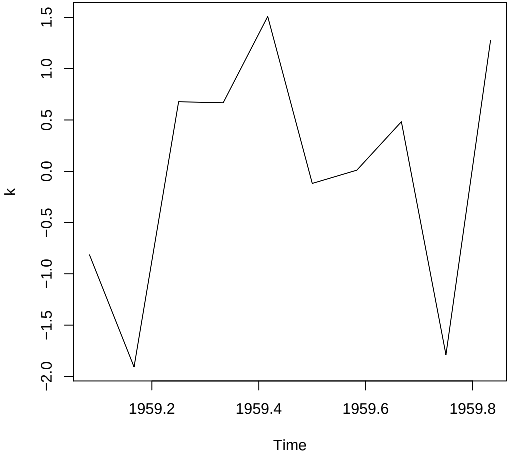
Com mais de uma variável:

```
> (m <- ts(matrix(rnorm(30), 10, 3), start = c(1961,
     6), frequency = 12))

           Series 1   Series 2    Series 3
Jun 1961  2.9220266  0.9721777  0.45087795
Jul 1961 -1.1894378  0.6397086 -0.42912402
Aug 1961 -1.2965092  0.1567075  0.07729965
Sep 1961 -0.0411518  0.4917292 -0.20341952
Oct 1961 -2.5608640 -0.1250718  0.35181090
Nov 1961 -1.3954284 -1.0325372  0.23694719
Dec 1961 -0.3789048 -0.9430397 -1.72054388
Jan 1962  1.0475009 -0.1309345 -0.18597275
Feb 1962  0.9546303  0.7500802  0.87557424
Mar 1962  1.1306772 -0.7085573 -0.34824707
```

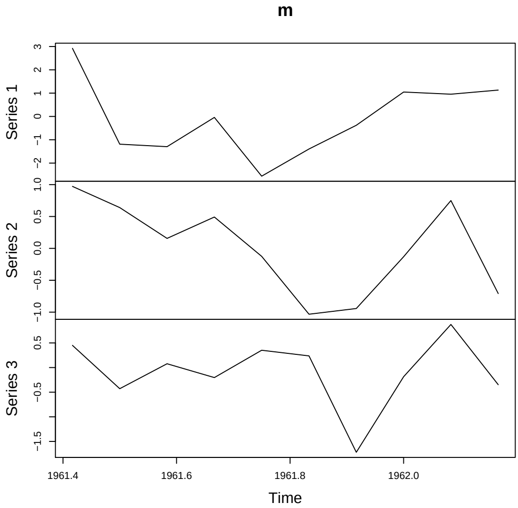
Séries temporais regulares

```
> plot(k)
```



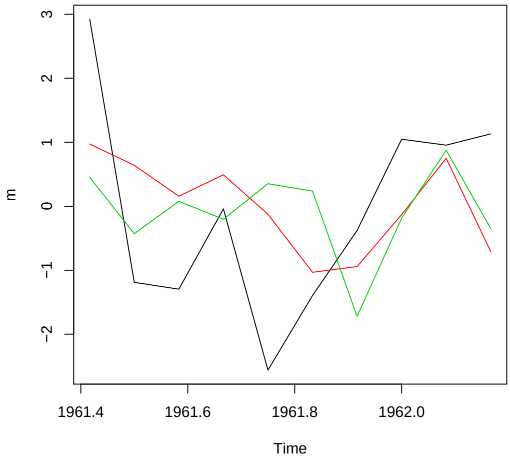
Séries temporais regulares

```
> plot(m)
```



Séries temporais regulares

```
> plot(m, plot.type = "single", col = 1:3)
```



Séries temporais regulares

Operando com séries temporáis: `diff`, `lag`, `start`, `end`, `window`

```
> a <- ts(rnorm(10), frequency = 4, start = c(1959,
     2))
> b <- ts(rnorm(10), frequency = 4, start = c(1960,
     1))
> cbind(a, b)

                  a           b
1959 Q2 -0.01328483          NA
1959 Q3 -1.35513913          NA
1959 Q4 -0.31264488          NA
1960 Q1 -0.31747694  0.62395131
1960 Q2  1.73284283  2.41993514
1960 Q3 -0.25013843 -0.12187548
1960 Q4 -0.40744921  0.48572371
1961 Q1  0.30571225  0.30083900
1961 Q2 -1.18619405  0.42586877
1961 Q3 -0.56753791  0.29608128
1961 Q4          NA -1.42583654
1962 Q1          NA  0.06325206
1962 Q2          NA  0.31944083
```

Séries temporais irregulares

- Pacote `its`
- Pacote `zoo`
- Pacote `tseries`

Merge

```
> x <- zoo(rnorm(5), seq(as.Date("2011-01-01"),
     length = 5, by = "days"))
> y <- zoo(rnorm(5), seq(as.Date("2011-01-04"),
     length = 5, by = "days"))
> cbind(x, y)
```

```
                    x          y
2011-01-01  1.0302287         NA
2011-01-02  0.2283393         NA
2011-01-03 -0.6680974         NA
2011-01-04  0.9973810 -0.9057784
2011-01-05  1.1139767 -0.1214299
2011-01-06         NA -0.5848263
2011-01-07         NA -0.2162378
2011-01-08         NA  0.2750344
```

Merge

```
> merge(x, y, all = FALSE)
```

```
                   x          y
2011-01-04 0.997381 -0.9057784
2011-01-05 1.113977 -0.1214299
```

Veja mais opções em `?merge.zoo`!

Merge

```
> Sq1 <- read.table(file = "squid1.txt",
     header = TRUE, stringsAsFactors = FALSE)
> head(Sq1)

  Sample     GSI
1      1 10.4432
2      2  9.8331
3      3  9.7356
4      4  9.3107
5      5  8.9926
6      6  8.7707
```

Merge

```
> Sq2 <- read.table(file = "squid2.txt",
     header = TRUE, stringsAsFactors = FALSE)
> head(Sq2)
```

```
  Sample YEAR MONTH Location Sex
1      1    1     1        1   2
2      2    1     1        3   2
3      3    1     1        1   2
4      5    1     1        1   2
5      6    1     1        1   2
6      7    1     1        1   2
```

Merge

Qual a diferença?

```
> dim(Sq1)

[1] 2644    2

> dim(Sq2)

[1] 2643    5

> unique(c(Sq1$Sample[-1], Sq1$Sample[length(Sq1$Sample)]) -
     Sq1$Sample)

[1] 1 0

> unique(c(Sq2$Sample[-1], Sq2$Sample[length(Sq2$Sample)]) -
     Sq2$Sample)

[1] 1 2 0
```

Merge

```
> SquidMerged <- merge(Sq1, Sq2, by = "Sample")
> head(SquidMerged)
```

```
  Sample     GSI YEAR MONTH Location Sex
1      1 10.4432    1     1        1   2
2      2  9.8331    1     1        3   2
3      3  9.7356    1     1        1   2
4      5  8.9926    1     1        1   2
5      6  8.7707    1     1        1   2
6      7  8.2576    1     1        1   2
```

Teoria dos conjuntos:

```
> setdiff(Sq1$Sample, Sq2$Sample)
```

```
[1] 4
```

Veja mais em `?intersect`.

Merge

```
> SquidMerged <- merge(Sq1, Sq2, by = "Sample",
     all = TRUE)
> SquidMerged[1:10, ]

   Sample     GSI YEAR MONTH Location Sex
1       1 10.4432    1     1        1   2
2       2  9.8331    1     1        3   2
3       3  9.7356    1     1        1   2
4       4  9.3107   NA    NA       NA  NA
5       5  8.9926    1     1        1   2
6       6  8.7707    1     1        1   2
7       7  8.2576    1     1        1   2
8       8  7.4045    1     1        3   2
9       9  7.2156    1     1        3   2
10     10  6.8372    1     2        1   2
```

Interação com o ambiente

```
> x <- 35
> cat("x tem o valor de ", x, "\t Estranho?\n")

x tem o valor de  35          Estranho?

> scan()
> scan(4)
> scan(what = character())
```