

# Introdução ao software R

## Comandos básicos e aplicações em economia

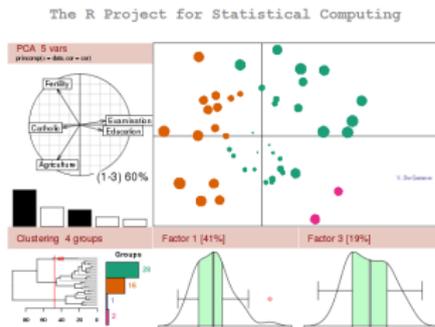
Aluno PAE: Elder Generozo Sant' Anna  
Prof. Dr. Luiz Guilherme Scorzafave

# Sumário

- 1 O Programa
- 2 Instalação do programa
- 3 Instalação de pacotes
- 4 Importante dados
- 5 Manipulando os dados

# O Programa

- O R é software livre destinado para análises estatísticas e gráficas.
- É resultado do esforço, colaboração e contribuições de todo o mundo.
  - Arne Henningsen desenvolveu o pacote **micEcon** com ferramentas de análise microeconómica e modelagem
  - William Revelle desenvolveu o **psych** com procedimentos para análise psicométrica



## Instalação do programa

### Instalando o programa

- Para instalar o programa basta visitar **www.r-project.org**
- Escolher o CRAN mirror - uma espécie de servidor de onde o R será baixado (escolher o mais próximo possível)
- Escolha o sistema operacional e na próxima página opte por *install R for the first time*
- Por fim clique em *Download R 3.0.2 for Windows*

# Instalação de pacotes

## Instalando pacotes

- Na interface principal do R clique em: *Packages* → *Install package(s) ...*
- Escolha o CRAN da onde você baixará o pacote
- Escolha o pacote (**Rcmdr**) e clique em OK
- Outra opção: *Install package(s) from local zip files ...*
- Atentar para a versão para o qual o pacote foi construído
  - Alguns pacotes não funcionam em versões antigas do programa
- Toda vez que se for utilizar um pacote precisa "chamá-lo"
  - `library(maptools)`
- Alguns pacotes, para serem instalados, precisam da instalação de outros.

## Importante dados

- O R lê quase todas as extensões de base de dados
  - **read.mtp** importa 'Minitab Portable Worksheet'
  - **read.xport** lê arquivos no formato SAS
  - **read.S** para objetos produzidos no S-PLUS
  - **read.spss** para arquivos de dados no formato do SPSS
  - **read.dta** lê base de dados produzidas no STATA 11
  - **read.csv** lê dados separados por vírgulas
  - **read.table** mais genérico, lê arquivos em formato de tabela e cria um quadro de dados a partir dele.

### Exemplo

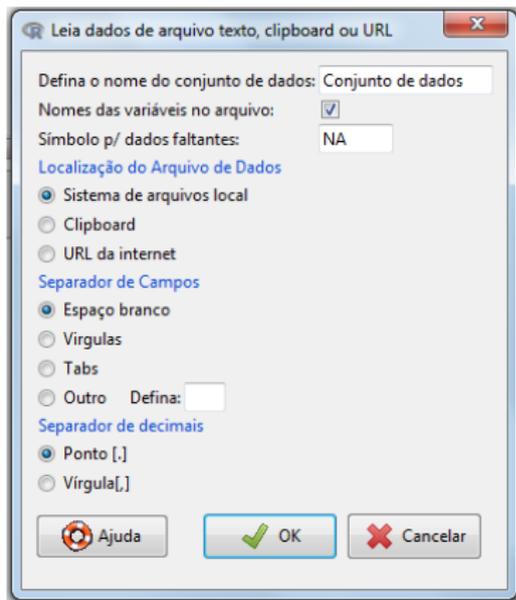
- `dados <- read.table("C:/brasil.CSV", header=T, dec=".", sep=",")`
- `header`: se `TRUE` indica que a primeira linha é o nome das variáveis
- `dec`: indica qual o separador da casa decimal
- `sep`: indica qual o separador dos dados

# O pacote Rcmdr

- Esse pacote pode facilitar a importação dos dados para o programa

## Importante arquivos .raw

- `library(Rcmdr)` → Abrirá uma tela do Rcommander
- Clique em Dados → Importar arquivos de dados → de arquivo texto, clipboard ou URL.



## Abrindo ceosal1.raw

- Antes de tudo, vamos criar um *script*
  - File → New script
- Verifique o arquivo ceosal1.xls - tem 12 colunas, 209 linhas e não possui nome das variáveis
- Salve o arquivo na extensão .csv
- ```
ceosal1 <- read.csv("diretório/ceosal1.csv", header=FALSE, dec="," , sep=";")
```
- ```
names(ceosal1) <- c("salary", "pcsalary", "sales", "roe", "pcroe", "ros", "indus", "finance", "consprod", "utility", "lsalary", "lsales")
```

## Descrevendo os dados

- Lembrando o R trabalha com objetos, logo a base é um objeto chamado *ceosal1*
- Para descobrir quais variáveis a base contém usa-se **names**
  - `names(ceosal1)` → mostrará o nome de todos os objetos (variáveis) contidos no objeto em questão (base de dados)
- Para estaísticas descritivas da base como um todo usamos o comando **summary**
  - `summary(ceosal1)` → descrição de todas as variáveis da base
  - `summary(ceosal1$roe)` → descrição apenas de uma variável
- Podemos fixar um objeto e não mais precisar ficar se referindo a ele com o comando **attach**
  - `attach(ceosal1)` → fixa o objeto *ceosal1*
  - `summary(roe)` → descrição apenas de uma variável
  - Perceba que não precisei especificar que a *col4* esta na *ceosal1*

## Operando ou Construindo variáveis

- Suponha que o salário está em mil dólares e queremos ele em dólares
  - `ceosal1$sal <- ceosal1$salary*1000`
- Suponha que queremos identificar aqueles que ganham menos de um salário mínimo
  - `ceosal1$menosmin <- ifelse(ceosal1$salary<450, 1, 0)`
- Podemos somar, subtrair, multiplicar e dividir objetos, desde que tenham mesma dimensão
  - Se for um vetor linha ou coluna usamos a função **length**
  - `length(salary)`
  - Caso seja uma matriz  $n \times m$  usamos a função **dim**
  - `dim(ceosal1)`
- Podemos criar um subconjunto da base
  - `ceosal2 <- subset(ceosal1,ceosal1$menosmin==0)`

## Outros comandos úteis

- Podemos verificar a média, variância e o desvio padrão das variáveis
  - `mean(ceosal$salary)`
  - `var(ceosal$salary)`
  - `sd(ceosal$salary)`
- Podemos verificar a correlação entre as variáveis
  - `library(Hmisc)`
  - `corr <- rcorr(cbind( $X_1, X_2, \dots, X_n$ ) , type=c("pearson"))`

```
> corr <- rcorr(cbind(ceosal1$salary, ceosal1$roe), type=c("pearson"))  
> corr
```

```
      [,1] [,2]  
[1,] 1.00 0.11  
[2,] 0.11 1.00
```

```
n= 209
```

```
P  
      [,1] [,2]  
[1,]      0.0978  
[2,] 0.0978  
> |
```

# Fazendo gráficos

- Podemos visualizar graficamente a associação entre duas variáveis
  - `plot(ceosal1$salary, ceosal1$roe)`
  - O comando `plot` tem vários *options* (internet)

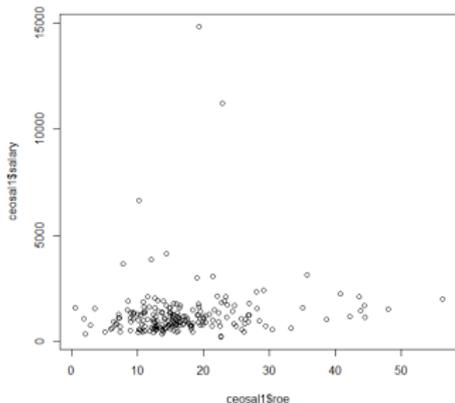
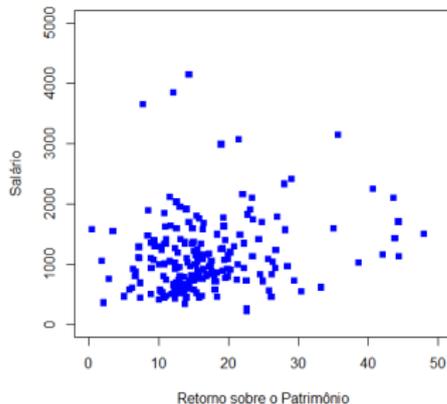


Gráfico 1 - Salário x Retorno sobre o Patrimônio



## Operações com matrizes

- Construindo uma matriz com o **matrix**
  - `A <- matrix(c(1, 2, 3, 4, 5,6), nrow=3, ncol=2, byrow=TRUE)`
  - Para visualizar um elemneto da matriz  $\rightarrow$  `a[1,1]`
  - Para visualizar uma linha da matriz  $\rightarrow$  `a[1, ]`
  - Para visualizar uma coluna da matriz  $\rightarrow$  `a[ ,2 ]`
- Podemos construir uma matriz agrupando vetores com o **cbind** ou o **rbind**
  - `B <- cbind(ceosal1$salary, ceosal1$roe)`  $\rightarrow$  combinando colunas
  - `N <- rbind(V1, V2, ... Vn)`  $\rightarrow$  combinando linhas
- É possível ainda criar uma matriz diagonal com o **diag**
  - `C <- diag(6, nrow=5)`
  - `D <- diag(1, nrow=5)`  $\rightarrow$  Matriz identidade  $5 \times 5$
- Para transpor matrizes basta usar o comando **t**
  - `E <- t(a)`

## Operações com matrizes

- Para inverter matrizes basta usar o **solve**
  - `F <- solve(A)`
  - `G <- solve(A, b)` → gera um vetor  $x$  que resolve a seguinte equação  $b = Ax$  onde  $x = A^{-1}b$
- Multiplicação de matrizes:
  - `H <- C*D` → Multiplicação elemento a elemento
  - `I <- C%*%D` → Multiplicação matricial
  - `J <- C%o%D` → Produto externo  $\vec{c} \wedge \vec{d} = |\vec{c}||\vec{d}|\sin\theta \vec{n}$
  - `K <- crossprod(C,D)` → retorna  $C'D$
  - `L <- crossprod(D)` → retorna  $D'D$

## MQO - Derivação das estimativas por métodos dos momentos

- Sabemos que os estimadores de uma regressão simples são dados por:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (1)$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \quad (2)$$

- Além disso

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \quad (3)$$

$$\hat{u}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i \quad (4)$$

## MQO - Derivação das estimativas por métodos dos momentos

- Exemplo 2.3
- Primeiro vou calcular o  $\bar{x}$  e, posteriormente, criar um vetor de  $\bar{x}$ 's. Ao objeto  $\bar{x}$  eu dei o nome de "x1" e ao vetor coluna  $\bar{x}$  eu dei o nome de "xbar". O mesmo vou fazer para o y. Vale lembrar que, o vetor x é o **roe** e o vetor y é o **salary**
  - `x1 <- mean(roe)`
  - `xbar <- rep(x1,nrow(ceosal1))`
  - `y1 <- mean(salary)`
  - `ybar <- rep(y1,nrow(ceosal1))`
- Agora preciso criar mais dois vetores colunas, um será o  $(x_i - \bar{x})$  e o outro  $(y_i - \bar{y})$ . Ao primeiro eu dei o nome de "xi\_xbar" e ao segundo "yi\_ybar".
  - `xi_xbar <- roe - xbar`
  - `yi_ybar <- salary - ybar`

## MQO - Derivação das estimativas por métodos dos momentos

- Agora é necessário resolver os somatório. Lembre-se que, para somar um vetor coluna  $1 \times n$  basta pré-multiplicar esse vetor por um vetor de uns  $n \times 1$ . Dessa forma, criei o vetor "aux" que tem a seguinte forma  $[11 \dots 1]_{1 \times n}$  e, em seguida, pré-multipliquei  $(x_i - \bar{x})(y_i - \bar{y})$  e, também  $(x_i - \bar{x})^2$ .
  - `aux <- rep(1, nrow(ceosal1))`
  - `numerador <- aux%*%(xi_xbar*yi_ybar)`
  - `denominador <- aux%*%(xi_xbar^2)`
- Finalmente, dividindo o resultado do somatório do numerador pelo somatório do denominador será obtido o  $\hat{\beta}_1$  e pela equação 2 do slide 15 o  $\hat{\beta}_0$ .
  - `beta1 <- numerador/denominador`
  - `beta0 <- y1 - beta1*x1`

## MQO - Propriedades Algébricas

- Vamos verificar a equação (2.36) do capítulo 2, ou seja, será mesmo que  $SQT = SQE + SQR$
- Primeiro, vamos calcular  $SQT \equiv \sum_{i=1}^n (y_i - \bar{y})^2$ .
  - `SQT <- aux%*%(yi_ybar^2)`
  - `SQT`
- Agora vamos calcular  $SQT \equiv \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$ . Porém, não temos o vetor  $\hat{y}$  então vamos calculá-lo por meio da equação 3 do slide 15. Perceba que precisaremos, antes de tudo criar um vetor para cada um dos *betas* estimatos.
  - `aux0 <- rep(beta0, nrow(ceosal1))`
  - `aux1 <- rep(beta1, nrow(ceosal1))`
  - `yhat <- aux0 + aux1*roe`
  - `SQE <- aux%*%((yhat-ybar)^2)`
  - `SQE`

## MQO - Propriedades Algébricas

- Finalmente, calcula-se o  $SQR \equiv \sum_{i=1}^n \hat{u}_i^2$ . Onde  $\hat{u}_i$  é dado pela equação 4 do slide 15.
  - `uhat <- salary - yhat`
  - `SQR <- aux%*%(uhat^2)`
  - `SQR`
- Por fim, vamos verificar se, de fato,  $SQT = SQE + SQR$ .
  - `TESTE <- SQE + SQR`
  - `TESTE`
  - `SQT`

## Lista de Exercícios - Primeiras aplicações em R

- 1 - Todos os exercícios a serem executados devem ser escritos em um *scrip* como foi ensinado em aula. O nome desse arquivo deverá ser o seu número USP.
- 2 - Qual a proporção de observações salarias da base por tipo de industria?
- 3 - Qual a média das vendas **em dólares**?
- 4 - Verifique se a correlação entre *salary* e *sales* é diferente da correlação entre o Logaritmo do salário e o lagaritmo das vendas.
- 5 - Qual é o 103<sup>o</sup> elemneto do vetor *pcroe*?
- 6 - Crie uma matriz  $m \times 3$  que satisfaça as seguintes condições: a mudança percentual do salário, o retorno sobre o estoque da empresa e o retorno sobre o patrimonio líquido **apenas** para aquelas informações de empresas financeiras.
- 7 - Transponha a matriz criada anteriormente

## Lista de Exercícios - Primeiras aplicações em R

- 8 - Crie uma matriz quadrada a partir da matriz criada anteriormente e, posteriormente, calcule a inversa da mesma.
- 9 - Agora realizem duas multiplicações, uma termo a termo e a outra uma multiplicação vetorial, com os seguinte vetores *pcsalary* e *pcroe*.
- 10 - Verifique as seguintes propriedades algébricas das estatísticas de MQO:

1.  $\sum_{i=1}^n \hat{u}_i = 0$
2.  $\sum_{i=1}^n x_i \hat{u}_i = 0$
3.  $\sum_{i=1}^n \hat{u}_i (\hat{y}_i - \bar{y}) = 0$

Boa Sorte!!!