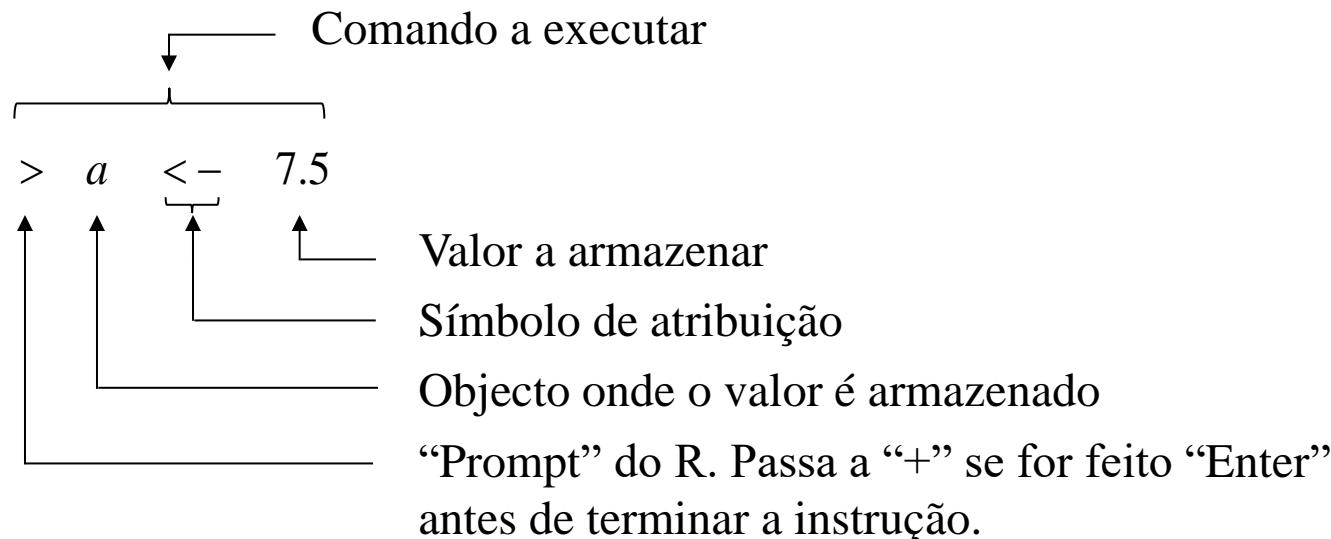


# Introdução ao R

# Introdução

- **Linguagem R:**
  - **O que é?** É uma *linguagem de programação* grátis.
  - **Aplicações:** É usada na análise estatística de dados.
  - **Instalação:** Download a partir do site <http://www.r-project.org>
- **Armazenamento dos dados no R:**
  - Os dados são armazenados em *objectos de várias classes* (*vectores, matrizes, etc.*)
  - **Ex 1:**



# Introdução

- **Ex 2:** `> x <- 5`
- **Ex 3:** `> y <- (2*x^2+7)/2`
- **Algumas funções úteis:**
  - **q()** – Termina a execução do R, abrindo uma janela com uma pergunta.  
Se a resposta for:
    - “Yes” – Guarda a informação criada na sessão de trabalho nos ficheiros “.Rhistory” e “.Rdata” , no directório de trabalho actual.
    - “No” – Sai sem guardar a informação.
  - **getwd()** – Mostra o directório de trabalho actual.
  - **setwd()** – Faz com que o directório especificado se torne o actual.

**Ex:** `> setwd('C:/USER/Documents/AulasR')`

**Nota:** Uma alternativa a `setwd()` é fazer **File → Change dir ...**

# Introdução

- **help()** – Mostra ajuda sobre a função ou tema especificados.  
*Ex:*  $> help(distributions)$
- **ls()** ou **objects()** – Mostra lista de objectos armazenados na memória.
- **rm()** – Remove o objecto especificado.  
*Ex 1:*  $> rm(a)$   
*Ex 2:*  $> rm(list = ls())$
- **Scripts:**
  - **O que são?** São ficheiros com código.
  - **Criação:** *File → New script*
  - **Execução:** *Edit → Run all*  
*Edit → Run line or selection*
  - **Ex:** Criar script com exemplos desta aula.

# Vectores

- *Vectores:*

- *O que são?* São os *objectos mais básicos* do R.
- *Tipos (classes) de vectores:* *character, logical, numeric, complex.*
- *c()* – Função usada para criar vectores com mais de 1 elemento.

*Ex:* `> v <- c(4,7,23.5,80)`

`> v`

- *length()* – Função que mostra o tamanho do vector indicado.

*Ex:* `> length(v)`

- *mode()* – Função que mostra o modo (classe) do vector.

*Ex:* `> mode(v)`

- *Operações com vectores:*

*Ex 1:* `> z <- sqrt(v)`

`> z`

# Vectores

**Ex 2:** >  $u <- c(34,32.5,12,8)$

>  $u + v$

>  $u * v$

>  $u / v$

>  $u[1]*v[2]$

- *Atribuição de nomes às posições do vector:*

**Ex 1:** >  $x <- c(\text{Lisboa} = 20, \text{Porto} = 17.5, \text{Faro} = 22)$

>  $x$

**Ex 2:** >  $y <- c(20,17.5,22)$

>  $names(y) <- c("Lisboa", "Porto", "Faro")$

>  $y$

# Factores

- *Factores:*
  - *O que são?* São *objectos* do R usados para *armazenar etiquetas* (“*labels*”), como, por exemplo, o estado civil de um conjunto de pessoas.
  - `factor()` – Função que cria 1 factor.

**Ex 1:** `> s <- factor(c("m","f","f","m"))`  
`> s`

**Ex 2:** `> outro.s <- factor(c("m","m","m"))`  
`> outro.s`

**Ex 3:** `> outro.s <- factor(c("m","m","m"),levels = c("f","m"))`  
`> outro.s`

**Ex 4:** `> e.c <- factor(c(1,1,0,1),labels = c("solteiro","casado"))`  
`> e.c`

# Factores

- **table()** – Função que mostra tabela de valores de um factor.

*Ex 1:* `>table(s)`

*Ex 2:* `>table(e.c)`

- **Tabelas cruzadas (2 variáveis):**

*Ex:* `>table(s,e.c)`

- **Frequências marginais:**

*Ex:* `>tab.cruz <- table(s,e.c)`

`> margin.table(tab.cruz,1)` (somas por linhas)

`> margin.table(tab.cruz,2)` (somas por colunas)

- **Frequências relativas:**

*Ex:* `>prop.table(tab.cruz)`

`>prop.table(tab.cruz,1)`

`>prop.table(tab.cruz,2)`

# Sequências

- *Sequências:*
  - *Sequências de inteiros consecutivos:*

**Ex 1:** `>x<-1:10`

`>x`

**Ex 2:** `>10:15-1`

**Ex 3:** `>10:(15-1)`

**Ex 4:** `>5:0`

- *Outras sequências:*

**Ex 1:** `>seq(-4,1,0.5)`

**Ex 2:** `>seq(from = 0, to = 6, length = 4)`

**Ex 3:** `>seq(length = 4, from = 0, by = 2)`

# Indexação

- *Indexação:*

- *Índices lógicos:*

**Ex 1:**  $x > -c(0, -3, 4, -1, 45)$

$x > 0$

**Ex 2:**  $x[x > 0]$

**Ex 3:**  $x[x > 0 \ \& \ x \leq 20]$

**Ex 4:**  $x[x > 0 \mid x < -1]$

- *Índices numéricos:*

**Ex 1:**  $x[1:4]$

**Ex 2:**  $x[-5]$

# Indexação

- *Índices com nomes:*

**Ex:** `> y <- c(Lisboa = 20, Porto = 17.5, Faro = 22)`

`> y`

`> y["Porto"]`

- *Índices vazios:*

**Ex:** `> x`

`> x[ ] <- 0`

`> x`