

# Introdução ao R: aula 2

Alexandre Rademaker

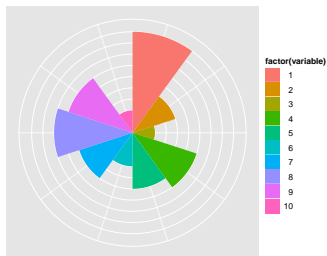
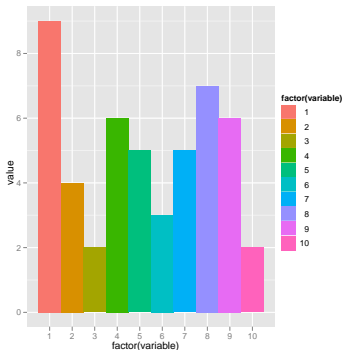
EMAp/FGV

January 26, 2011

```

> DF <- data.frame(variable = 1:10, value = sample(10,
+   replace = TRUE))
> p1 <- ggplot(DF, aes(factor(variable),
+   value, fill = factor(variable))) +
+   geom_bar(width = 1)
> p2 <- p1 + scale_y_continuous(breaks = 0:10) +
+   coord_polar() + labs(x = "", y = "") +
+   opts(axis.text.x = theme_blank(),
+   axis.text.y = theme_blank(), axis.ticks = theme_blank())

```



# Operações aritméticas básicas

- Operações aritméticas com escalares
- soma, subtração, divisão, multiplicação etc.
- Funções e argumentos
- Variáveis

# Exemplos

```
> 5 + 49 - (8 * 5)
```

```
[1] 14
```

```
> -5
```

```
[1] -5
```

```
> 12/4
```

```
[1] 3
```

```
> 31%%7
```

```
[1] 3
```

```
> 31%/%7
```

```
[1] 4
```

# Funções

Para “executar” a função, sempre usar os parênteses, mesmo quando não forem passados argumentos. Argumentos podem ser passados pelos nomes ou pelas posições.

```
> q()  
> q(save = "no", runLast = TRUE)  
> q("no", , TRUE)
```

## Inspecionando funções

```
> q  
  
function (save = "default", status = 0, runLast = TRUE)  
  .Internal(quit(save, status, runLast))  
<environment: namespace:base>  
  
> args(q)  
  
function (save = "default", status = 0, runLast = TRUE)  
NULL  
  
> str(q)  
  
function (save = "default", status = 0, runLast = TRUE)
```

# Variáveis

```
> Wing1 <- 59
```

```
> Wing2 <- 55
```

```
> 2 * Wing1
```

```
[1] 118
```

```
> Wing1 + Wing2
```

```
[1] 114
```

```
> (Wing1 + Wing2)/5
```

```
[1] 22.8
```

Cuidado com a expressão `a<-5`, pode ser uma atribuição de 5 para `a` ou comparação do valor `-5` com o valor da variável `a`.

```
> a <- 5
```

```
> a < -5
```

```
[1] FALSE
```

Use espaços!

## Exemplo: empréstimo

Uma pessoa quer pegar um empréstimo de  $P$  com taxa de  $i/\text{mês}$ . O empréstimo deverá ser pago em  $n$  vezes de valor  $R$ .

$$P = R(1+i)^{-1} + R(1+i)^{-2} + \dots + R(1+i)^{-n}$$

ou

$$P = R \sum_{j=1}^n (1+i)^{-j}$$

A soma da série geométrica é dada por:

$$P = R \left( \frac{1 - (1+i)^{-n}}{i} \right)$$

Onde o valor de  $R$  é obtido por:

$$R = P \frac{i}{1 - (1+i)^{-n}}$$



## Exemplo: empréstimo

```
> intRate <- 0.01
> n <- 10
> principal <- 1500
> (payment <- principal * intRate/(1 - (1 +
+   intRate)^(-n)))

[1] 158.3731
```

# Juntando dados: vetores e algumas funções

- Função “c”
- Operador “:”
- Extraíndo partes de um vetor, operador “[”.
- R é “case-sensitive”
- Operando com valores *NA*, “na.rm”
- Funções “rep” e “seq”.

## Constantes especiais

```
> alguns.pares <- NULL
> alguns.pares[seq(2, 20, 2)] <- seq(2,
+   20, 2)
> alguns.pares

[1] NA  2 NA  4 NA  6 NA  8 NA 10 NA 12 NA 14 NA
[16] 16 NA 18 NA 20

> x <- c(0, 7, 8)
> x/x

[1] NaN    1    1

> 1/x

[1]      Inf 0.1428571 0.1250000

> x[0.4]

numeric(0)
```

# Armadilhas das aproximações

```
> 0.1 == 0.3/3
```

```
[1] FALSE
```

```
> seq(0, 1, by = 0.1) == 0.3
```

```
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
[8] FALSE FALSE FALSE FALSE
```

```
> unique(c(0.3, 0.4 - 0.1, 0.5 - 0.2, 0.6 -  
+ 0.3, 0.7 - 0.4))
```

```
[1] 0.3 0.3 0.3
```

```
> 7/13 - 3/31
```

```
[1] 0.4416873
```

```
> print(7/13 - 3/31, digits = 16)
```

```
[1] 0.4416873449131513
```

# Valores aproximados

Padrão de aproximação do R é de 53 dígitos em binários, aproximadamente 15/16 dígitos em decimais.

$$\forall n \quad (5/4 * n * 4/5) = n$$

```
> n <- 1:10
```

```
> 1.25 * (n * 0.8) - n
```

```
[1] 0.000000e+00 0.000000e+00 4.440892e-16
```

```
[4] 0.000000e+00 0.000000e+00 8.881784e-16
```

```
[7] 8.881784e-16 0.000000e+00 0.000000e+00
```

```
[10] 0.000000e+00
```

```
> 5/4 * (n * 4/5) - n
```

```
[1] 0 0 0 0 0 0 0 0 0 0
```

# Herança de Lisp!

```
CL-USER> (mapcar #'(lambda (n) (- (* 5/4 (* n 4/5)) n))  
  '(1 2 3 4 5 6 7 8 9 10))  
(0 0 0 0 0 0 0 0 0 0)
```

```
CL-USER> 4/5
```

```
4/5
```

```
CL-USER> (defparameter *a* 4/5)
```

```
*A*
```

```
CL-USER> *a*
```

```
4/5
```

```
> a <- 4/5
```

```
> a
```

```
[1] 0.8
```