Introdução ao R & Probabilidades

prof. Andre Luiz Cunha

21/03/2025

Introdução ao R

Instalação

O software R é um pacote estatístico gratuito e *open source*. Possui uma extensa documentação e comunidade ativa de suporte, além de várias bibliotecas de funções para tratamento e análise de dados.

A página oficial do R <https://www.r-project.org/> traz as versões disponíveis para download, assim como o repositório **CRAN mirrors** <https://cran.r-project.org/mirrors.html> que fornece os arquivos necessários para instalações e atualizações.

Tipos de Dados

- Numeric

```
x = 3.5
x
class(x)
```

- Integer

```
y = 1
y
class(y)
is.integer(y)

y = as.integer(y)
y = 1L
class(y)
```

- Complex

```
z = 1 + 2i
class(z)

sqrt(-1)
sqrt(-1 + 0i)
```

- Logical

```r
v1 = 2
v2 = 5
res = v1 > v2
res
!res
TRUE & FALSE
TRUE | FALSE
```

- Character

```r
s = "STT5898 - Estatística"
nchar(s)

x = as.character(3.14)
x
class(x)

name = "Andre"
family = "Cunha"
paste(name, family)
paste(family, name, sep=", ")
```

- Factor (categoria : nominal ou ordinal)

```r
a = factor("A")
a
class(a)

x = factor(1)
y = factor(2)
x + y
```

Vector

Vector é uma sequência de elementos do **MESMO TIPO**.

```r
c(2,3,4)
c(TRUE, FALSE, TRUE, FALSE)
c("aa", "bb", "cc", "dd")
c(2, TRUE, "aa")

# Lenght
length(c("aa", "bb", "cc", "dd"))

# Combine vectors
x = c(1,2,3)
y = c(4,5,6)
z = c(x,y)
z

# Arithmetics
```

```r
x + y
x - y
x * y
x / y
3.14 * x

# Recycling
w = c(10,20,30,40,50,60)
x + w

# Index
w[1]
w[2]
w[1:3]
w[-1]
w[-6]
w[c(1,5,6)]
w[c(1,1,2)]
w[c(3,5,1)]
w[w > 20]
```

List

Uma **List** é um vetor genérico que contém qualquer tipo de dado. É uma estrutura flexível que permite que objetos de diferentes tipos fiquem em um mesmo contêiner.

```r
n = c(2,3,5)
s = c("aa", "bb", "cc", "dd")
b = c(TRUE, FALSE, FALSE, TRUE)
x = list(n,s,b,3L)
x

# List slicing
x[1]
x[c(2,4)]

# Member access
x[[1]]
x[[1]] = c(11,12,13)

# Named Members
dados = list(
  nome=c("Andre", "Maria", "José"),
  titulo=c("Dr", "Eng", "MSc"),
  anoUSP=c(2004,2024,2023)
)
dados
dados['nome']
dados$nome
dados[['nome']]
```

Data Frame

Uma **Data Frame** é usada para manusear dados tabelados (**banco de dados**). É uma **List** de **Vectors** de mesmo tamanho.

```
df = data.frame(
  nome=c("Andre", "Maria", "José"),
  titulo=c("Dr", "Eng", "MSc"),
  anoUSP=c(2004,2024,2023)
)
df
df['nome']
df$nome
df[['nome']]
```

Probabilidades

```
sample(1:7)
sample(1:7, 5)
sample(1:7, 5, replace = TRUE)
sample(1:7, 5, prob = (1:7)/sum(1:7))
globe = c("water", "land")
sample(globe, 10, replace = TRUE, prob = c(0.7, 0.3))
```

Exemplo 1 - Bagagens

Versão 1

```
## Create data frame
bagagem <- data.frame(entrega = c(rep('Normal', 6500), rep('Defeito', 350)),
                      empresa = c(rep('X', 500), rep('Y', 4500), rep('Z', 1500),
                                  rep('X', 30), rep('Y', 270), rep('Z', 50)))
length(bagagem)
dim(bagagem)
nrow(bagagem)
ncol(bagagem)

## Frequencies - table
(tbl <- table(bagagem))
table(bagagem$empresa)
table(bagagem$entrega)

## Proportions
tbl/sum(tbl)
prop.table(tbl)
prop1 = prop.table(tbl, 1)
apply(prop1,1, sum)
prop2 = prop.table(tbl, 2)
apply(prop2,2, sum)
```

```r
proportions(tbl)
proportions(tbl,1)
proportions(tbl,2)
```

Versão 2

```r
df <- as.data.frame.table(tbl)
df

tbl_df <- xtabs(Freq ~ entrega + empresa, df)
tbl_df

proportions(tbl_df)

## P( empresa | entrega )
proportions(tbl_df, 1)
df_prob <- as.data.frame.table(proportions(tbl_df, 1))
df_prob[df_prob$entrega == 'Defeito',]
df_prob[df_prob$entrega == 'Normal',]

## P( entrega | empresa )
proportions(tbl_df, 2)
df_prob <- as.data.frame.table(proportions(tbl_df, 2))
df_prob[df_prob$empresa == 'X',]
df_prob[df_prob$empresa == 'Y',]
df_prob[df_prob$empresa == 'Z',]
```

Exemplo 2 - Emissão

```r
## Emissão de poluentes:
## CARRO : regular, excesso
## TESTE : negativo (não excesso), positivo (excesso)
##
## P(carro = 'excesso') = 25%
## P(teste = 'positivo' | carro = 'excesso') = 99%
## P(carro = 'regular') = 75%
## P(teste = 'positivo' | carro = 'regular') = 17%
##
## P(carro = 'excesso' | teste = 'positivo' ) = ???
## P(carro = E | teste = +) = (P(teste = + | carro = E) * P(carro = E)) / P(teste = +)

## P(teste = +) = P(teste = + | carro = E) * P(carro = E) + P(teste = + | carro = R) * P(carro = R)
.25 * .99 + .17 * .75
0.99 * 0.25 / 0.375

df_tbl <- data.frame(veiculo_emissao = rep(c('excesso', 'regular'),1, each=2),
                     teste_emissao = rep(c('positivo', 'negativo'),2),
                     Freq = c(0.99 * .25, 0.01 * .25, 0.17 * .75, 0.83 * .75)
                     )
proportions(xtabs(Freq ~ teste_emissao + veiculo_emissao, df_tbl),2)
```

Exemplo 3 - Rodovias

```
ex3 <- data.frame(rodovia = rep(1:4, 2),
                  trafego = c(rep('congestionado',4), rep('normal',4)),
                  Freq    = c(0.3, 0.2, 0.60, 0.35,
                              0.7, 0.8, 0.4, 0.65))
ex3$Freq <- ex3$Freq * rep(0.25, 8)
ex3_tbl <- xtabs(Freq ~ trafego + rodovia, ex3)
ex3_tbl
prop.table(ex3_tbl)
prop.table(ex3_tbl,1)
prop.table(ex3_tbl,2)
```

Exemplo 4 - Globo

```
df4_prior <- data.frame(local = c('L', 'W'),
                        Freq = c(0.5, 0.5))

df4 <- data.frame(local = c('W', 'W', 'L', 'W'))
df4_tbl <- table(df4)
df4_likelihood <- as.data.frame.table(proportions(df4_tbl))

df4_posterior <- data.frame(local = c('L', 'W'),
                            Freq = (df4_prior$Freq * df4_likelihood$Freq)/
                              sum(df4_prior$Freq * df4_likelihood$Freq))
df4_prior
df4_likelihood
df4_posterior
```