

R Base

Folha de Consulta

Obtendo Ajuda

Acessando os arquivos de ajuda

?mean

Obter a ajuda de uma função específica.

help.search('weighted mean')

Pesquisar a ajuda por meio de palavra ou frase.

help(package = 'dplyr')

Encontrar a ajuda de um pacote.

Mais sobre um objeto

str(iris)

Obter um resumo da estrutura de um objeto.

class(iris)

Saber a classe de um objeto.

Usando Pacotes

install.packages('dplyr')

Baixar e instalar um pacote do CRAN.

library(dplyr)

Carregar o pacote para dentro da sessão, disponibilizando todas as suas funções.

dplyr::select

Usar uma função específica de um pacote.

data(iris)

Carregar um conjunto de dados interno para dentro da sessão.

Diretório de Trabalho

getwd()

Encontrar o diretório de trabalho atual (onde estão as entradas e aonde vão as saídas).

setwd('C://local/arquivo')

Mudar o diretório de trabalho atual.

Use projetos no RStudio para assinalar o diretório de trabalho à pasta que você está usando.

Vetores			Programação					
Criando Vetores			Repetição "Para"			Repetição "Enquanto"		
c(2, 4, 6)	2 4 6	Juntar elementos em um vetor	for (variavel in sequencia){	Fazer algo	}	while (condicao){	Fazer algo	}
2:6	2 3 4 5 6	Uma sequência de inteiros						
seq(2, 3, by=0.5)	2.0 2.5 3.0	Uma sequência complexa						
rep(1:2, times=3)	1 2 1 2 1 2	Repetir um vetor	for (i in 1:4){	j <- i + 10	print(j)	while (i < 5){	print(i)	i <- i + 1
rep(1:2, each=3)	1 1 1 2 2 2	Repetir elementos de um vetor	}			}		
Funções Vetoriais								
sort(x)	rev(x)	Retornar x ordenado.	if (condicao){	Fazer algo	} else {	if (condicao){	Fazer algo	return(nova_variavel)
		Retornar x revertido.	 Fazer algo diferente		}	 Fazer algo diferente		}
table(x)	unique(x)	Contagens de valores.						
		Ver valores únicos.						
Selecionando Elementos de um Vetor								
Por Posição			Estruturas condicionais			Funções		
x[4]	O quarto elemento.		if (condicao){	Fazer algo	} else {	nome_da_funcao <- function(var){	Fazer algo	
x[-4]	Todos menos o quarto.		 Fazer algo diferente		}	 return(nova_variavel)	}	
x[2:4]	Elementos dois a quatro.						Exemplo	
x[!(2:4)]	Todos elementos exceto 2 a 4.		if (i > 3){	print('Sim')	} else {	quadrado <- function(x){	elevadoadois <- x**2	
x[c(1, 5)]	Elementos um e cinco.		 print('Nao')		}	 return(quadrado)	}	
Por Valor			Exemplo			Exemplo		
x[x == 10]	Elementos que são iguais a 10.		if (i > 3){	print('Sim')	} else {	quadrado <- function(x){	elevadoadois <- x**2	
x[x < 0]	Todos elementos menores que zero.		 print('Nao')		}	 return(quadrado)	}	
x[x %in% c(1, 2, 5)]	Elementos no conjunto 1, 2, 5.						Lendo e Escrevendo Dados	
Vetores com nomes			Entrada			Saída		
x['abacaxi']	Elemento com nome 'abacaxi'.		df <- read.table('arquivo.txt')	write.table(df, 'arquivo.txt')			Descrição	
							Ler e escrever um arquivo de texto delimitado.	
			df <- read.csv('file.csv')	write.csv(df, 'arquivo.csv')			Ler e escrever um arquivo de valores separados por vírgula. Este é um caso especial de read.table/write.table.	
			load('arquivo.RData')	save(df, file = 'arquivo.Rdata')			Ler e escrever um arquivo de dados R, um tipo de arquivo especial para R.	
Condições			Condições			Condições		
			a == b	São iguais	a > b	Maior que	a >= b	Maior que ou igual a
			a != b	Não iguais	a < b	Menor que	a <= b	Menor que ou igual a
								is.na(a)
								É faltante
								is.null(a)
								É nulo

Tipos

Convertendo entre tipos comuns de dados em R. Pode-se sempre ir de um valor mais alto na tabela a um valor mais baixo.

<code>as.logical</code>	TRUE, FALSE, TRUE	Valores booleanos (VERDADEIRO ou FALSO).
<code>as.numeric</code>	1, 0, 1	Números inteiros ou de ponto flutuante.
<code>as.character</code>	'1', '0', '1'	Cadeias de caracteres. Geralmente preferíveis a fatores.
<code>as.factor</code>	'1', '0', '1', levels: '1', '0'	Cadeias de caracteres com níveis pré-estabelecidos. Necessário para alguns modelos estatísticos.

Funções Matemáticas

<code>log(x)</code>	Log natural.	<code>sum(x)</code>	Soma.
<code>exp(x)</code>	Exponencial.	<code>mean(x)</code>	Média.
<code>max(x)</code>	Maior elemento.	<code>median(x)</code>	Mediana.
<code>min(x)</code>	Menor elemento.	<code>quantile(x)</code>	Quantis percentílicos.
<code>round(x, n)</code>	Arredondar para n casas decimais.	<code>rank(x)</code>	Postos dos elementos.
<code>sig.fig(x, n)</code>	Arredondar para n algarismos significativos.	<code>var(x)</code>	A variância.
<code>cor(x, y)</code>	Correlação.	<code>sd(x)</code>	O desvio padrão.

Atribuição de Variável

```
> a <- 'abacaxi'
> a
[1] 'abacaxi'
```

O Ambiente

<code>ls()</code>	Listar todas as variáveis no ambiente.
<code>rm(x)</code>	Remover x do ambiente.
<code>rm(list = ls())</code>	Remover todas as variáveis do ambiente.

Você pode usar o painel Environment no Rstudio para navegar pelas variáveis no seu ambiente.

Matrizes

```
m <- matrix(x, nrow = 3, ncol = 3)
```

Criar uma matriz a partir de x.



`m[2,]` - Selecionar uma linha



`m[, 1]` - Selecionar uma coluna



`m[2, 3]` - Selecionar um elemento

`t(m)`

Transpor

`m %*% n`

Multiplicação de Matrizes

`solve(m, n)`

Determinar x em: $m^* x = n$

Listas

```
l <- list(x = 1:5, y = c('a', 'b'))
```

Uma lista é uma coleção de elementos que podem ser de diferentes tipos.

`l[[2]]`

Segundo elemento de l.

`l[1]`

Nova lista com apenas o primeiro elemento.

`l$x`

Elemento chamado x.

`l['y']`

Nova lista com único elemento chamado y.

Veja também o pacote `dplyr`.

Data Frames

```
df <- data.frame(x = 1:3, y = c('a', 'b', 'c'))
```

Um caso especial de lista onde todos elementos têm o mesmo tamanho.

x	y
1	a
2	b
3	c



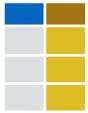
Entendendo um data frame

`View(df)` Ver o data frame inteiro.

`head(df)` Ver as 6 primeiras linhas.

Subagrupamento de matrizes

`df[, 2]`



`df[2,]`



`df[2, 2]`



`nrow(df)`

Número de linhas.

`ncol(df)`

Número de colunas.

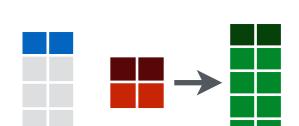
`dim(df)`

Número de colunas e linhas.

`cbind` - Concatenar colunas.



`rbind` - Concatenar linhas.



Strings

```
paste(x, y, sep = ' ')
```

Juntar múltiplos vetores.

```
paste(x, collapse = ' ')
```

Juntar elementos de um vetor.

```
grep(pattern, x)
```

Achar expressões regulares em x.

```
gsub(pattern, replace, x)
```

Substituir ocorrências em x por um string.

`toupper(x)`

Converter para caixa alta.

`tolower(x)`

Converter para caixa baixa.

`nchar(x)`

Número de caracteres em um string.

Fatores

`factor(x)`

Transformar um vetor em um fator. Pode-se definir os níveis do fator e a ordem.

`cut(x, breaks = 4)`

Transformar um vetor numérico em um fator 'cortando'-o em seções.

Estatística

`lm(x ~ y, data=df)`

Modelo linear.

`glm(x ~ y, data=df)`

Modelo linear generalizado.

`summary`

Obter informações mais detalhadas de um modelo.

`t.test(x, y)`

Fazer um teste t para

diferença de

duas

proporções.

`pairwise.t.test`

Fazer um teste t para

dados pareados.

`aov`

Análise de variância.

Distribuições

Variáveis aleatórias	Função Densidade	Distribuição Acumulada	Quantil
Normal	<code>rnorm</code>	<code>dnorm</code>	<code>pnorm</code>
Poisson	<code>rpois</code>	<code>dpois</code>	<code>ppois</code>
Binomial	<code>rbinom</code>	<code>dbinom</code>	<code>pbinom</code>
Uniforme	<code>runif</code>	<code>dunif</code>	<code>qunif</code>

Gráficos

`plot(x)`

Valores de x em ordem.

`plot(x, y)`

Valores de x contra y.

`hist(x)`

Histograma de x.

Veja também o pacote `ggplot2`.

Datas

Saiba mais em [página web](#) ou [vinheta](#) • versão do pacote: 3/15