

Ana Hermínia Andrade e Silva
Gilmara Alves Cavalcanti
Juliana Freitas Pires
Maria Lídia Coco Terra
organizadoras

INTRODUÇÃO À ESTATÍSTICA NO **SOFTWARE R**



EJ Editora
UFPB

INTRODUÇÃO À ESTATÍSTICA NO ***SOFTWARE R***



Reitor
Vice-Reitora

UNIVERSIDADE FEDERAL DA PARAÍBA

Valdiney Veloso Gouveia
Liana Filgueira Albuquerque



Direção
Gestão de Editoração
Gestão de Sistemas

EDITORA UFPB

Natanael Antonio dos Santos
Sâmella Arruda Araújo
Ana Gabriella Carvalho

Conselho Editorial

Adailson Pereira de Souza (Ciências Agrárias)
Eliana Vasconcelos da Silva Esval (Linguística, Letras e Artes)
Fabiana Sena da Silva (Interdisciplinar)
Gisele Rocha Côrtes (Ciências Sociais Aplicadas)
Ilda Antonieta Salata Toscano (Ciências Exatas e da Terra)
Luana Rodrigues de Almeida (Ciências da Saúde)
Maria de Lourdes Barreto Gomes (Engenharias)
Maria Patrícia Lopes Goldfarb (Ciências Humanas)
Maria Regina Vasconcelos Barbosa (Ciências Biológicas)

Editora filiada à:



Ana Hermínia Andrade e Silva
Gilmara Alves Cavalcanti
Juliana Freitas Pires
Maria Lúcia Coco Terra

INTRODUÇÃO À ESTATÍSTICA NO *SOFTWARE R*

João Pessoa
Editora UFPB
2021

Direitos autorais 2021 – Editora UFPB.

TODOS OS DIREITOS RESERVADOS À EDITORA UFPB.

É proibida a reprodução total ou parcial, de qualquer forma ou por qualquer meio.

A violação dos direitos autorais (Lei nº 9.610/1998) é crime estabelecido no artigo 184 do Código Penal.

O conteúdo e a revisão de texto/normatização desta publicação são de inteira responsabilidade do(s) autor(es).

Projeto Gráfico
Editoração Eletrônica

Editora UFPB
Emmanuel Luna

Catálogo na fonte:

Biblioteca Central da Universidade Federal da Paraíba

I61 Introdução à estatística no software R [recurso eletrônico] / Ana Hermínia Andrade e Silva ... [et al.]. - João Pessoa : Editora UFPB, 2021.

E-book.

ISBN: 978-65-5942-112-1

1. Estatística. 2. Software R. 3. Inferência estatística. 4. Estatística descritiva. I. Silva, Ana Hermínia Andrade e. II. Título.

UFPB/BC

CDU 311

Livro aprovado para publicação através do Edital Nº 01/2020/Editora Universitária/ UFPB - Programa de Publicação de E-books.

EDITORA UFPB

Cidade Universitária, Campus I
Prédio da Editora Universitária, s/n
João Pessoa – PB
CEP 58.051-970
<http://www.editora.ufpb.br>
E-mail: editora@ufpb.br
Fone: (83) 3216.7147

Aos monitores do projeto “Estatística Aplicada em Software Livre”, Caroline Assis, Davi Batista, Giovani Nucci e João Marcelo Galiza.

Ao Departamento de Estatística da Universidade Federal da Paraíba.

À Jane Torelli e toda a equipe da assessoria de extensão do Centro de Ciências Exatas e da Natureza.

À Pró-reitora de extensão da Universidade Federal da Paraíba.

À Editora UFPB.

SUMÁRIO

1 INTRODUÇÃO	9
2 PRIMEIROS PASSOS	11
2.1 Instalação	11
2.2 Usar o script para digitar os comandos	13
2.3 Instalação de Pacotes	15
2.4 Como acessar a ajuda	17
2.5 O RStudio como calculadora	21
2.6 Criando objetos	21
2.7 Algumas funções úteis	24
2.7.1 Demonstrações	24
2.7.2 Como gerar sequências	25
2.7.3 Funções: sort, order e rank.....	25
2.7.4 Função apply.....	27
2.8 Transformar vetores em matrizes	27
2.8.1 Somando linhas e somando colunas	29
2.9 Função na.omit	29
2.10 Comandos de lógica	30
2.10.1 Função which.....	31
2.10.2 Função ifelse.....	32
2.10.3 Função for.....	33
2.11 Criando funções	33
2.12 Importando dados	34
2.13 Função factor	36
3 ESTATÍSTICA DESCRITIVA	37
3.1 Representação Tabular	38
3.2. Medidas Resumo	42
3.2.1 Média aritmética	43
3.2.2 Mediana	44
3.2.3 Quantis	44
3.2.4 Moda	45
3.2.5 Variância.....	45
3.2.6 Desvio-padrão	46
3.2.7 Outras formas de obter medidas resumo	46
3.3 Representação Gráfica	48
3.3.1 Gráfico em Barras ou colunas	48

3.3.2 Gráfico de Pizza ou Setores.....	50
3.3.3 Gráfico box-plot	51
3.3.4 Gráfico de dispersão	52
3.3.5 Histograma.....	53
3.3.6 Gráfico de linhas.....	54
3.4 Mais recursos gráficos	55
3.5 Gráficos 3D.....	57
3.6 Gramática dos gráficos	60
4 INFERÊNCIA ESTATÍSTICA	66
4.1 Intervalos de Confiança	67
4.1.1 Intervalo de Confiança para a Média Populacional..	68
4.1.1.1 Intervalo de Confiança para a Média Populacional com Variância Populacional Conhecida.....	68
4.1.1.2 Intervalo de Confiança para a Média Populacional com Variância Populacional Desconhecida.....	70
4.1.2 Intervalo de Confiança para a Proporção Populacional.....	72
4.2 Testes de Hipóteses	74
4.2.1 Teste de Hipótese para a Média Populacional.....	77
4.2.1.1 Teste de Hipótese para a Média Populacional com a Variância Populacional Conhecida.....	78
4.2.1.2 Teste de Hipótese para a Média Populacional com Variância Populacional Desconhecida.....	79
4.4 Teste de Hipótese para a Proporção Populacional....	83
4.5 Teste de Hipótese para a Diferença entre Duas Médias Populacionais (Duas Populações Independentes)	86
4.5.1 Teste de Hipótese para a Diferença entre Duas Médias Populacionais com Variâncias Populacionais Iguais ($\sigma_1^2 = \sigma_2^2$) e Desconhecidas	87
4.5.2 Teste de Hipótese para Diferença entre Duas Médias Populacionais com Variâncias Populacionais Diferentes ($\sigma_1^2 \neq \sigma_2^2$) e Desconhecidas	89
4.5.3 Teste de Hipótese para Diferença entre Duas Proporções Populacionais.....	91
4.5.4 Teste de Hipótese para a Diferença entre Duas Médias Populacionais (Populações Dependentes ou Pareadas)	93
4.6 Teste Qui-Quadrado de Independência	96
5 CORRELAÇÃO E REGRESSÃO.....	99

5.2 Análise Gráfica	100
5.3 Coeficiente de correlação linear de Pearson	102
5.4 Regressão linear simples.....	105
5.4.1 O Coeficiente de Determinação (R^2)	111
5.4.2 Teste de Hipóteses para o Coeficiente β	111
6 REFERÊNCIAS	114
7 SOBRE AS AUTORAS.....	115

1 INTRODUÇÃO

A experiência adquirida ao longo dos anos de docência permite constatar que há dificuldades inerentes no desempenho do alunado nas aplicações técnicas, visto que nas disciplinas básicas de estatística o conteúdo é basicamente teórico não havendo tempo hábil para as aplicações. Partindo dessa motivação, em 2018 nasceu o projeto *Estatística Aplicada em Software Livre*, com o objetivo de ministrar cursos práticos que integrem um conhecimento teórico ao prático, incentivando uma melhoria na aplicação da estatística nas pesquisas, artigos, dissertações e teses da comunidade acadêmica, resultando em um profissional de excelência para o mercado de trabalho. Iniciamos o projeto ofertando turmas práticas de introdução à estatística no *software R*, sendo este o *software* mais utilizado pela comunidade estatística atualmente.

O *software R* é um ambiente computacional integrado e ao mesmo tempo uma linguagem de programação orientada, desenvolvido para análise de dados, realização de cálculos e modelos estatísticos. Desenvolvida em 1996 por Ross Ihaka e Robert Gentleman, a linguagem R foi baseada na linguagem S, já bastante utilizada na época para análise estatística, com a diferença de ser gratuita. O *software* está disponível para *download* em <http://cran.r-project.org> para sistemas operacionais Linux, Windows e MacOS. A grande variedade de técnicas disponíveis se deve ao fato de se tratar de um *software* livre. O R vem então conquistando um lugar importante no âmbito da análise de dados, sendo atualmente o *software* estatístico mais utilizado mundialmente, ganhando cada vez mais espaço em todas as áreas do conhecimento.

Apesar do *software R* ter uma interface gráfica, novos usuários podem ter alguma dificuldade em utilizá-lo. Visto isso, podemos fazer uso de uma interface mais intuitiva integrada ao *software R*, chamada RStudio. O RStudio é encontrado em duas versões: RStudio Desktop, para aqueles usuários que desejam rodar o programa localmente como um *software* comum e RStudio Server, que permite o acesso remoto usando um navegador web e que facilita o desenvolvimento de grandes projetos que exijam o trabalho de diversas pessoas simultaneamente. Assim como o R, o RStudio é gratuito e está disponível para *download* em <https://www.rstudio.com> para os sistemas operacionais Windows, MacOS, e Linux.

O presente livro é um produto deste curso básico do projeto, escrito pelas professoras do departamento de Estatística da UFPB, Ana Hermínia Andrade e Silva, Gilmara Alves Cavalcanti, Juliana Freitas Pires e Maria Lídia Coco Terra. É possível acompanhar outros materiais e cursos ofertados pelo projeto *Estatística Aplicada em Software Livre* em nossas redes sociais: <https://www.instagram.com/estatisticalivre/> e <https://www.youtube.com/channel/UC8QTeEyzHqYRjojKneTgLBa>.

2 PRIMEIROS PASSOS

O *software* R é um ambiente computacional integrado gratuito e ao mesmo tempo uma linguagem de programação orientada, desenvolvido para análise de dados, realização de cálculos e modelos estatísticos. Como iremos basear as nossas aplicações no software RStudio, o primeiro passo será como instalar este *software* R no computador. Mas antes de instalar o RStudio é necessário instalar o *software* R. Para isto é necessário fazer o *download* em <http://cran.r-project.org> de acordo com o seu sistema operacional. Depois fazer a instalação do mesmo.

Com o *software* R instalado vamos agora instalar o RStudio. O RStudio é encontrado em duas versões: RStudio Desktop, para aqueles usuários que desejam rodar o programa localmente como um *software* comum; e RStudio Server, que permite o acesso remoto usando um navegador web e que facilita o desenvolvimento de grandes projetos que exijam o trabalho de muitas pessoas simultaneamente. Assim como o R, o RStudio é gratuito e está disponível para *download* em <https://www.rstudio.com/products/rstudio/download/> para os sistemas operacionais Windows, MacOS, e Linux.

2.1 INSTALAÇÃO

Entre na página da web <https://www.rstudio.com/products/rstudio/download> e clique na versão gratuita do RStudio Desktop (ver Figura 3.1.1 a seguir).

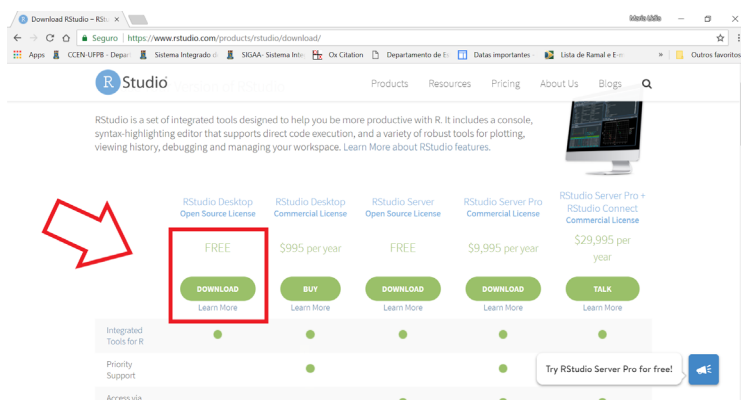


Figura 2.1.1: Primeiro passo para instalar o RStudio.

Esta versão é gratuita. Depois é só selecionar o arquivo de acordo com o seu sistema operacional (ver Figura 2.1.2 abaixo).

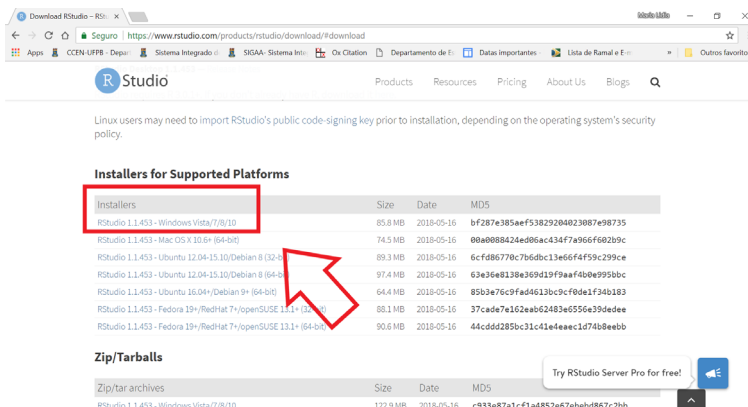


Figura 2.1.2: Segundo passo para instalar o RStudio.

Depois de baixar o instalador, clique no instalador e siga os passos de instalação. Após a instalação, será criado um ícone do RStudio em sua área de trabalho. Para iniciar o programa, dê um duplo-clique no ícone do RStudio localizado na área de trabalho. Ao fazer isso, o programa será executado e uma janela com o programa será aberta.

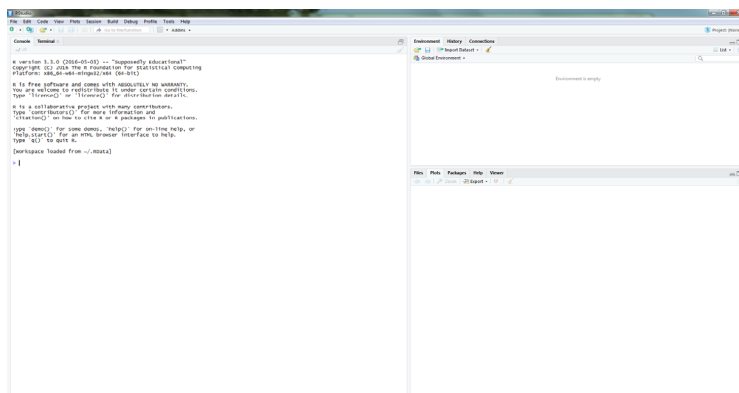


Figura 2.1.3: Abrindo o RStudio pela primeira vez.

Você deve notar que o programa se parece bastante com outros programas que você já deve estar habituado a usar. O programa é disposto da seguinte forma:

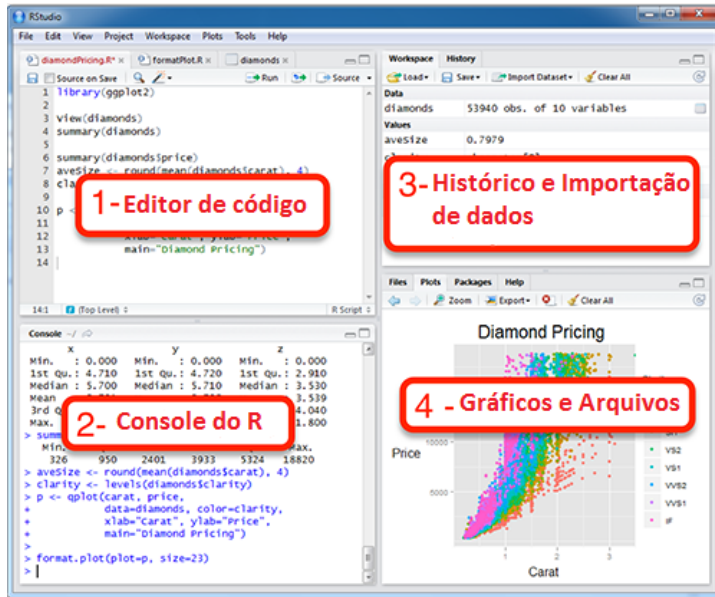


Figura 2.1.4: Disposição do RStudio.

2.2 USAR O SCRIPT PARA DIGITAR OS COMANDOS

Uma maneira que otimiza o uso do RStudio e que poupa muito tempo é usar um script (um arquivo .txt) para digitar seus comandos. Neste caso, os comandos não são digitados diretamente na linha de comandos e sim em um editor de texto (por exemplo: R editor, notepad, tinn-R).

Um script do RStudio é apenas um arquivo .txt em que você digitará todos os comandos e análises. Recomenda-se usar o RStudio editor, que já vem com o RStudio e é bem simples de usar. Nele, seus arquivos podem ser salvos com a extensão .R, e ao ser aberto dentro do RStudio, o script é visualizado sem problemas.

Ao usar um script você pode facilmente fazer alterações e correções em seus comandos, pois salvando o script você poderá refazer rapidamente suas análises, por exemplo, caso algum revisor de um de seus artigos solicite uma mudança em uma de suas análises ou em um de seus gráficos. Para utilizar o RStudio editor basta clicar na barra inicial no ícone de uma folha em branco com um mais verde que se localiza no canto esquerdo na parte de cima.

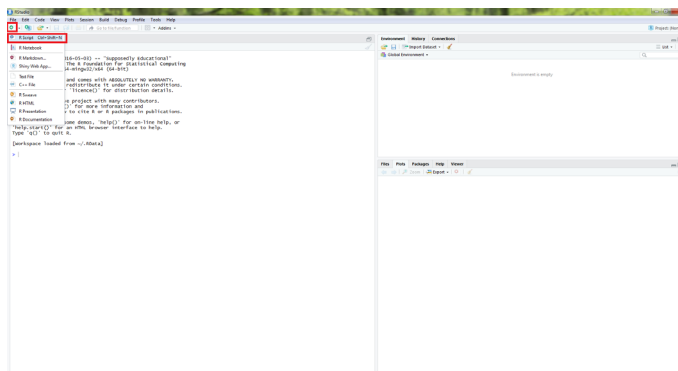


Figura 2.2.1: Abrindo um script no RStudio.

Outra forma seria clicar na barra inicial *File > New File > R Script* ou utilizar o atalho do teclado *Ctrl+Shift+N*. Então, a parte do console irá descer abrindo um espaço em branco no canto esquerdo na parte de cima. Digite 3+3 no script e aperte *Run*.

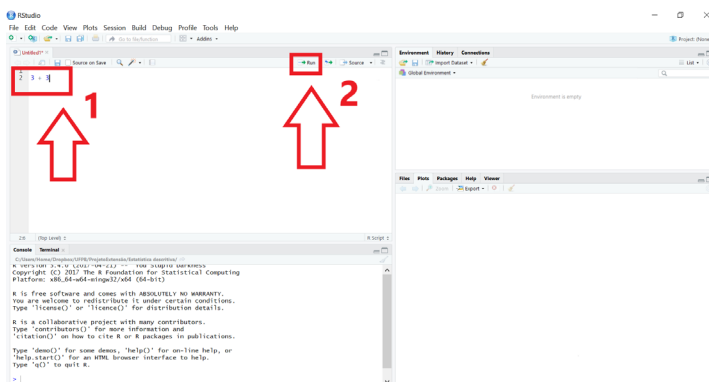


Figura 2.2.2: Rodando um script no RStudio.

O `3+3` será enviado para o console do RStudio e o resultado aparecerá na tela do canto esquerdo na parte de baixo. Para fazer outro comando, vá para o script e selecione a linha do último comando digitado, aperte *Enter* para passar para a linha de baixo e escreva outro comando (cada comando deve ser digitado em uma linha diferente).

No script você também pode inserir observações sobre o que foi feito, usando `#` para indicar a presença de um comentário. Por exemplo:

```
x = sample(1:10,20) # gerando a amostra
mean(x) # mean calcula a média
```

Para salvar o seu script, basta clicar no disquete abaixo acima dos comandos do seu script, ou clicar na barra inicial *File > Save*. O atalho do teclado é *Ctrl+S*. Se preferir, clique na barra inicial *File > Save As...* e selecione o local de seu computador em que o script deve ser salvo.

2.3 INSTALAÇÃO DE PACOTES

O R é um programa leve (ocupa pouco espaço e memória) e geralmente roda rápido, até em computadores não muito bons. Isso porque ao instalarmos o R apenas as configurações mínimas para seu funcionamento básico são instaladas (pacotes que vem na instalação “base”). Para realizar tarefas mais complicadas pode ser necessário instalar pacotes adicionais (packages).

Digamos que você está interessado em instalar um pacote específico. Para isso, clique na aba *Packages* do canto inferior direito do RStudio e depois clique em *Install*.

Aparecerá uma janela (ver Figura 2.3.2). Nela você deve escrever o nome do pacote desejado, por exemplo vamos instalar o pacote CARS. Após digitar o nome do pacote, clique em *Install* (deve-se ter acesso a internet para este procedimento). Após clicar em *Install*, o pacote será instalado. O tempo de espera depende da conexão de Internet e das especificações de cada pacote.

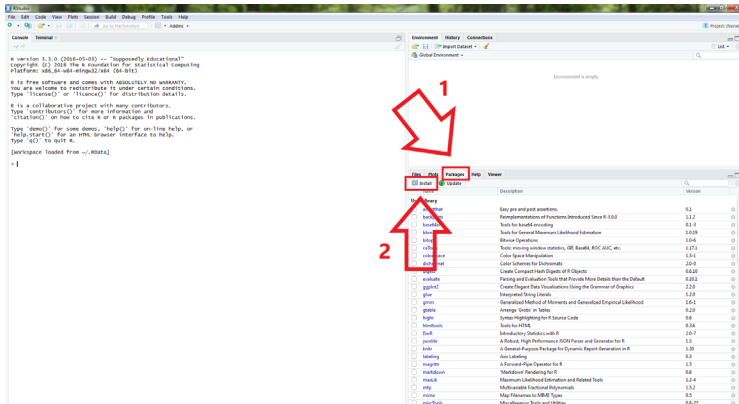


Figura 2.3.1: Primeiro passo para a instalação de pacotes no RStudio.

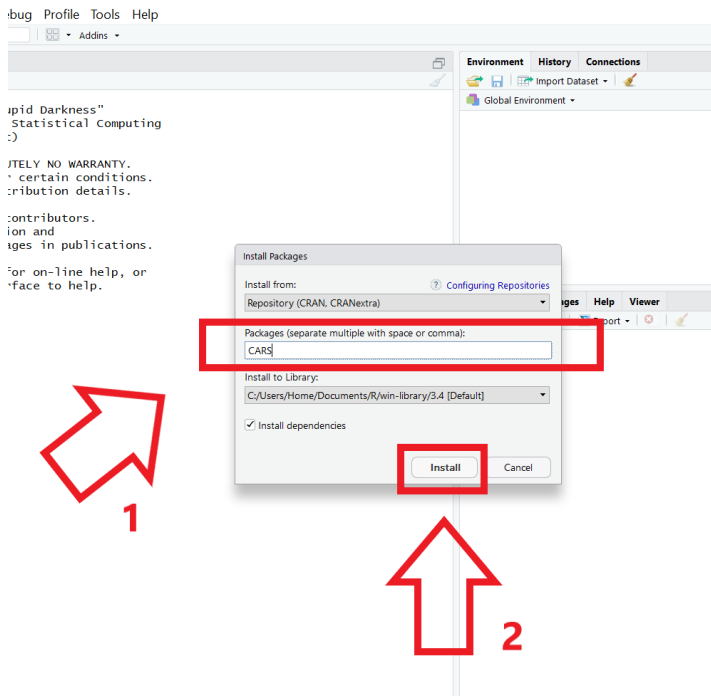


Figura 2.3.2: Segundo passo para instalação de pacotes no RStudio.

Para abrir um pacote e utilizar as funções dele, basta ir no console do RStudio e digitar o comando *library* ou *require* com o nome do pacote entre parênteses e clicar enter:

```
library(CARS)
require(CARS)
```

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal x
~/
R version 3.4.0 (2017-04-21) -- "You Stupid Darkness"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> require(CARS)
Carregando pacotes exigidos: CARS
Warning message:
package 'CARS' was built under R version 3.4.4
> library(CARS)
```

Figura 2.3.3: Utilizando um pacote.

2.4 COMO ACESSAR A AJUDA

Em diversos casos você irá querer fazer uma análise cujo nome da função você ainda não conhece. Nestes casos existem três formas básicas para descobrir uma função que faça aquilo que você deseja. A primeira é pesquisar dentro do RStudio usando palavras chave utilizando a função *help.search()*. Por exemplo, vamos tentar descobrir como calcular logaritmos no RStudio:

```
help.search("logaritmo")
```

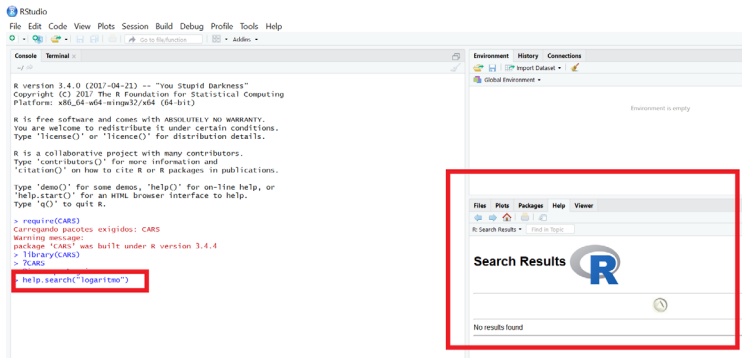


Figura 2.4.1: Acessando ajuda.

Veja que o RStudio não encontrou nenhuma função, pois o RStudio é desenvolvido em língua inglesa, portanto, precisamos buscar ajuda usando palavras em inglês.

> `help.search("Logarithms")`

Com esse comando o RStudio irá procurar, dentro dos arquivos de ajuda, possíveis funções para calcular logaritmos. Uma janela irá se abrir contendo possíveis funções.

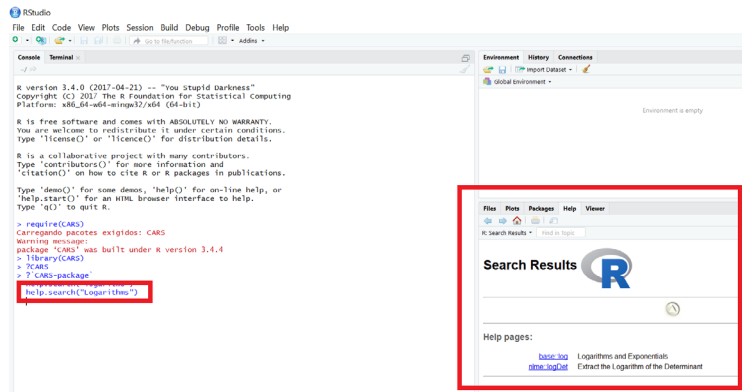


Figura 2.4.2: Acessando ajuda corretamente.

Nas versões mais recentes do RStudio a função `help.search()` pode ser substituída por apenas `??` (duas interrogações).

`??Logarithms`

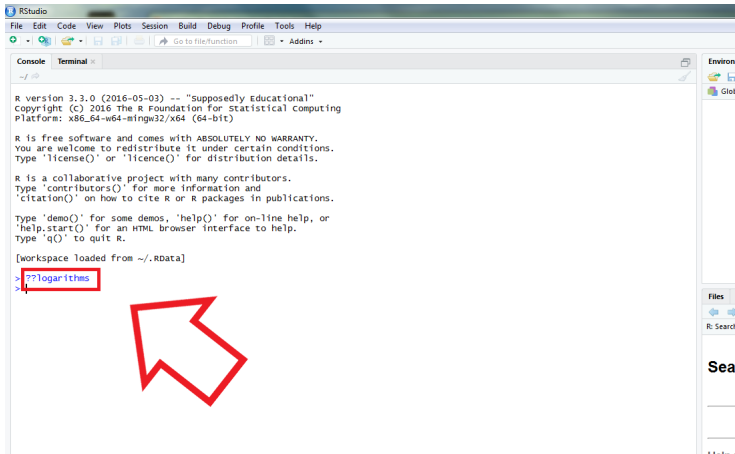


Figura 2.4.3: outra forma de acessar ajuda.

Também é possível buscar ajuda na internet, no site do RStudio, com a função `RSiteSearch()`.

`RSiteSearch("logarithms")`

Este comando irá abrir uma página na internet, mas só funcionará se seu computador estiver conectado à internet. Para ver os arquivos de ajuda do RStudio de uma função de algum pacote que você adquiriu use o comando `help(nome.da.função)` ou `?nome.da.função`. Por exemplo, vamos ver a ajuda da função `log`:

`help(log)`

ou simplesmente

`?log`

Geralmente, o arquivo de ajuda do RStudio possui 10 tópicos básicos:

1. **Description** - faz um resumo geral sobre o uso da função.
2. **Usage** - mostra como a função deve ser utilizada e quais argumentos podem ser especificados.
3. **Arguments** - explica o que é cada um dos argumentos.
4. **Details** - explica alguns detalhes sobre o uso e aplicação da função (geralmente poucos).
5. **Value** - mostra o que sai no output após usar a função (os resultados).
6. **Note** - notas sobre a função.
7. **Authors** - lista os autores da função (quem escreveu os códigos).
8. **References** - referências para os métodos usados.
9. **See also** - mostra outras funções relacionadas que podem ser consultadas.
10. **Examples** - exemplos do uso da função. Copie e cole os exemplos no console para ver como funciona.

Um arquivo de ajuda aparece no canto inferior direito do programa.

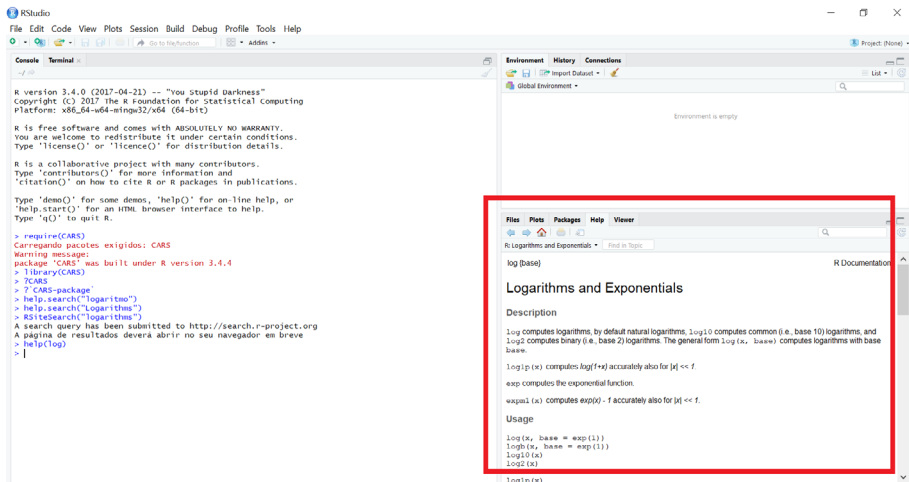


Figura 2.4.4: Arquivo de ajuda.

2.5 O RSTUDIO COMO CALCULADORA

A forma de uso mais básica do RStudio é usá-lo como calculadora. Os operadores matemáticos básicos são: + para soma, - subtração, * multiplicação, / divisão e ^ exponenciação. Digite as seguintes operações na linha de comandos do RStudio:

```
2 + 2
2 * 2
2 - 2
2 / 2
2^2
```

Use parênteses para separar partes dos cálculos, por exemplo, para fazer a conta 4+16, dividido por 4, elevado ao quadrado:

```
((4+16)/4)^2
```

2.6 CRIANDO OBJETOS

Um objeto pode ser criado com a operação de “atribuição”, o qual se denota como uma flecha, com o sinal de menos e o símbolo > ou <, dependendo da direção em que se atribui o objeto. Ou com um único sinal de igual. É importante dizer que o nome de um objeto deve começar com uma letra qualquer, maiúscula ou minúscula, que pode ser seguida de outras letras, números, ou caracteres especiais como o ponto.

```
x <- 10 # x receberá o valor 10
15 -> y # y receberá o valor 15
X <- 6 # X receberá o valor 6
Y = 15 # Y receberá o valor 15
```

Observe que existe diferença entre maiúscula e minúscula. Observe também que o símbolo # indica um comentário. Pode-se fazer cálculos com esse objetos criados. Você também pode armazenar o resultado de um cálculo em um objeto qualquer.

```
t <- sqrt(4) #objeto t irá receber o valor da operação indicada
```

Para mostrar o conteúdo do objeto criado `t`, digite apenas o nome do objeto na linha de comando do RStudio, como abaixo:

```
t
```

Vamos criar um conjunto de dados que contém o número de espécies de aves (riquezas) coletadas em 10 locais. As riquezas são 22, 28, 37, 34, 13, 24, 39, 5, 33, 32.

```
aves <- c(22, 28, 37, 34, 13, 24, 39, 5, 33, 32)
```

A letra minúscula `c` significa concatenar (colocar junto). Entenda como “agrupe os dados entre parênteses dentro do objeto” que será criado neste caso no objeto `aves`. A função `length` fornece o número de observações (`n`) dentro do objeto.

```
length(aves)
```

Também podemos criar objetos que contêm letras ou palavras ao invés de números. Porém, as letras ou palavras devem vir entre aspas.

```
letras <- c("a", "b", "c", "da", "ew")
palavras <- c("Manaus", "Boa Vista",
              "Belém", "Brasília")
```

Crie um objeto “misto”, com letras e com números. Funciona? Esses números realmente são números? Note a presença de aspas, isso indica que os números foram convertidos em caracteres. Evite criar vetores “mistos”, a menos que tenha certeza do que está fazendo. Podemos fazer diversas operações usando o objeto `aves`, criado anteriormente.

```
max(aves) #valor máximo
min(aves) #valor mínimo
sum(aves) #soma dos valores
aves^2
aves/10
```

Caso queira acessar apenas um valor do conjunto de dados use colchetes []. Isto é possível porque o RStudio salva os objetos como vetores, ou seja, a sequência na qual você incluiu os dados é preservada. Por exemplo, vamos acessar o quinto valor do objeto aves.

```
aves[5] # Qual o quinto valor de aves?
palavras[3] # Qual a terceira palavra?
```

Para acessar mais de um valor use c para concatenar dentro dos colchetes:

```
aves[c(5,8,10)]
# acessa o quinto, oitavo e décimo valor de aves
```

Para excluir um valor, por exemplo, o primeiro, use

```
# note que o valor 22 foi excluído
aves[-1]
```

Caso tenha digitado um valor errado e queira corrigir o valor, especifique a posição do valor e o novo valor. Por exemplo, o primeiro valor de aves é 22, caso estivesse errado, por exemplo, deveria ser 100, basta alterarmos o valor da seguinte maneira.

```
# O 1o valor de aves deve ser 100
aves[1]<-100

# Vamos voltar ao valor antigo
aves[1]<-22
```

Em alguns casos é necessário, ou recomendado, transformar os dados antes de analisá-los. Transformações comumente utilizadas são log e raiz quadrada.

```
#Raiz quadrada dos valores de aves
sqrt(aves)
#Logaritmo na base 10
log10(aves)
#Logaritmo natural de aves
log(aves)
```


Para salvar os dados transformados dê um nome ao resultado. Por exemplo:

```
aves.log <- log10(aves)
```

Para listar os objetos que já foram salvos use `ls()`, que significa listar. Para remover objetos use `rm()` para remover o que está entre parênteses.

```
rm(aves.log)
aves.log # você verá a mensagem:
Error in eval(expr, envir, enclos): objeto 'aves.log' não encontrado
```

2.7 ALGUMAS FUNÇÕES ÚTEIS

Nas seções a seguir, serão comentadas algumas funções que você poderá necessitar em suas análises.

2.7.1 Demonstrações

Algumas funções do R possuem demonstrações de uso. Estas demonstrações podem ser vistas usando a função `demo()`. Vamos ver algumas demonstrações de gráficos que podem ser feitos no R. Digite o seguinte na linha de comandos:

```
demo(graphics)
```

Vai aparecer uma mensagem pedindo que você tecle *Enter* para prosseguir, depois clique na janela do gráfico para ir passando os exemplos. Outras demonstrações:

```
demo(persp)
demo(image)
```

Ver também a função `example()`, que é bem parecida com a função `demo()`.

2.7.2 Como gerar seqüências

Existe duas formas de gerar seqüências de números no RStudio. A forma mais simples é feita com `:`, e a outra forma é usando o comando `seq()`. Dois pontos `:` é usado para gerar seqüências de um em um, por exemplo a seqüência de 1 a 10:

```
1:10 # Seq de 1 a 10
5:16 # Seq de 5 a 16
```

A função `seq()` é usada para gerar seqüências especificando os intervalos. Vamos criar uma seqüência de 1 a 10 pegando valores de 2 em 2.

```
seq( 1, 10, 2)
# o default é em intervalos de 1
```

A função `seq()` funciona assim:

```
seq(from = 1, to = 10, by = 2)
#seq(de um, a dez, em intervalos de 2)
```

Se você quiser criar uma seqüência de números repetidos, basta usar a função `rep` para repetir algo `n` vezes.

```
rep(5, 10) # repete 5 dez vezes
```

A função `rep()` funciona assim:

```
rep(x, times=y)
# rep(repita x, y vezes)
```

2.7.3 Funções: sort, order e rank

Primeiro vamos criar um vetor desordenado para servir de exemplo:

```
aves <- c(22, 28, 37, 34, 13,
          24, 39, 5, 33, 32)
```

A função `sort()` coloca os valores de um objeto em ordem crescente ou em ordem decrescente. Para colocar em ordem decrescente basta acrescentar o argumento `decreasing = TRUE`, e para colocar em ordem crescente não é necessário acrescentar este argumento, pois por default a função já ordena em ordem crescente.

```
# para colocar em ordem crescente
sort(aves)
[1] 5 13 22 24 28 32 33 34 37 39

# para colocar em ordem decrescente
sort(aves, decreasing = TRUE)
[1] 39 37 34 33 32 28 24 22 13 5
```

A função `order()` retorna a posição original de cada valor do objeto `aves` caso os valores do objeto `aves` sejam colocados em ordem.

```
aves
[1] 22 28 37 34 13 24 39 5 33 32
order(aves)
[1] 8 5 1 6 2 10 9 4 3 7
```

Note que o primeiro valor acima é 8, isso indica que se quisermos colocar o objeto `aves` em ordem crescente o primeiro valor deverá ser o oitavo valor de `aves`, que é o valor 5 (o menor deles). Na sequência devemos colocar o quinto valor do objeto `aves`, que é 13, depois o primeiro, depois o sexto até que o objeto `aves` fique em ordem crescente.

A função `rank()` atribui postos aos valores de um objeto.

```
aves
[1] 22 28 37 34 13 24 39 5 33 32
rank(aves)
[1] 3 5 9 8 2 4 10 1 7 6
```

Veja que 39 é o maior valor do exemplo, portanto recebe o maior `rank`, no caso 10.

2.7.4 Função apply

Esta função é útil para aplicar outra função nas linhas ou colunas de uma matriz ou vetor.

```
apply(x, margin, funcao, ...)
```

Por exemplo, vamos aplicar a soma nas linhas de uma matriz X.

```
# soma nas colunas
col.sums <- apply(X, 2, sum)
# soma nas Linhas
row.sums <- apply(X, 1, sum)
```

2.8 TRANSFORMAR VETORES EM MATRIZES

Além de importar tabelas, existe a opção unir vetores em um arquivo dataframe ou matriz. Para criar uma matriz use *cbind* (column bind) ou *rbind* (row bind). Vamos ver como funciona. Vamos criar três vetores e depois juntá-los em uma matriz.

```
aa<-c(1,3,5,7,9)
bb<-c(5,6,3,8,9)
cc<-c("a","a","b","a","b")
# juntar os vetores em colunas
cbind(aa,bb)
# juntar os vetores em Linhas
rbind(aa,bb)
```

Lembre que matrizes podem conter apenas valores numéricos ou de caracteres. Por isso, se juntarmos o vetor *cc*, nossa matriz será transformada em valores de caracteres.

```
# junta os vetores em colunas, mas
# transforma números em caracteres.
cbind(aa,bb,cc)
```

Para criar uma dataframe com valores numéricos e de caracteres use a função `data.frame`:

```
data.frame(aa,bb,cc)
```

Os comandos para se criar vetores e matrizes são:

```
A <- matrix(c(3, -1, 2, -2, 3, 1, 1, 4, 1, 4, 0, 3,
              0, 4, 0, 3), nrow=4,ncol=4, byrow=TRUE)
```

Também é possível fazer as operações com matrizes da seguinte forma:

```
B <- matrix(c(4, 0, 3, 0, 4, 1, 3, 1,2, 4, 0,
              3, 6, 4, 0, 3), nrow=4,ncol=4, byrow=TRUE)
t(B) # transposta de B
det(B) # determinante de B
solve(B) # inversa de B
C <- A+B # soma de matrizes
D <- A%*%B # produto de matrizes
```

Agora vamos aprender a selecionar (extrair) apenas partes do nosso conjunto de dados `A` usando `[]` colchetes. O uso de colchetes funciona assim: `[linhas, colunas]`, onde está escrito linhas você especifica as linhas desejadas, na maioria dos casos cada linha indica uma unidade amostral. Onde está escrito colunas, você pode especificar as colunas (atributos) que deseja selecionar. Veja abaixo:

```
# extrair a 1a coluna
# de todas as linhas
A[,1]
# extrair a 2a coluna
# de todas as linhas
A[,2]
# extrai a 1a linha
# de todas as colunas
A[1,]
# extrair a 3a linha e a 3a coluna
A[3,3]
```

```
# extrai o valor da linha 1 e coluna 3
A[1,3]
#extrai somente as linhas 1 a 4
# e as colunas 2 e 3
A[c(1:4),c(2,3)]
```

2.8.1 Somando linhas e somando colunas

Somar os valores de colunas ou linhas usando as funções *colSums* para somar colunas e *rowSums* para somar linhas.

```
# Somando apenas as informações
# sobre as colunas 2 a 3
colSums(A[,2:3])
# Somando apenas as informações
# sobre as linhas 1 a 4
rowSums(A[1:4,])
```

Você pode escolher quais colunas (ou linhas) você deseja somar. Lembrando que, para as matrizes no RStudio, o primeiro número do colchete se refere as linhas e o segundo se refere as colunas. Você também pode somar colunas (ou linhas) contínuas, por exemplo de 2 a 4, fazendo *colSums(A[,2:4])* ou pontuando as colunas (ou linhas) necessárias, por exemplo, se você quisesse somar as colunas 1,3 e 4, *colSums(A[,c(1,3,4)])*.

2.9 FUNÇÃO NA.OMIT

Muitas vezes, em alguns bancos de dados, se tem informações faltando. Quando isso ocorre, o RStudio lê essas informações faltantes como NA. Porém, no momento de analisar os dados a presença desses NAs pode causar erros.

```
na.omit(object, ...)
```

Esta função retira os NAs para que a análise seja feita. Por exemplo, vamos criar um banco de dados com informação faltante e observar como a função funciona:

```
DF <- data.frame(x = c(1, 2, 3),
                 y = c(0, 10, NA))
na.omit(DF$y)
[1] 0 10
attr(,"na.action")
[1] 3
attr(,"class")
[1] "omit"
na.omit(DF)
  x y
1 1 0
2 2 10
```

Observe a diferença nos dois casos acima: em `na.omit(DF$y)`, nós especificamos que se deva retirar as observações não informadas apenas do objeto `y` do banco de dados `DF`. No segundo exemplo (`na.omit(DF)`), nós retiramos as observações faltantes do banco de dados, então o R elimina as observações faltantes e deixa todas as variáveis do banco de dados com o mesmo número de observações. Por isso, muito cuidado ao usar esta função, e pense bem qual das duas formas você deseja utilizar para eliminar as observações faltantes.

2.10 COMANDOS DE LÓGICA

Primeiro vamos ver o significado dos comandos abaixo.

> maior que >= maior ou igual a

< menor que <= menor ou igual a

== igualdade != diferença

Então se tivermos dois vetores `x` e `y`, vamos ver o que acontece se utilizarmos estes operadores lógicos.

```
x <- c(1,2,9,4,5)
y <- c(1,2,6,7,8)
x > y
# Retorna TRUE para os maiores
# e FALSE para os menores
x == y
# Retorna TRUE para os x
# que sao iguais a y
x != y
# Retorna TRUE para os x
# que sao diferentes de y
```

O mesmo ocorre se utilizarmos os operadores `>=` e `<`. Sempre será retornado TRUE ou FALSE, correspondente ao operador utilizado. Também existem algumas funções que utilizam estes operadores.

2.10.1 Função which

A função `which` funciona como se fosse a pergunta: Quais?

```
a<-c(2,4,6,8,10,12,14,16,18,20)
a>10
[1] FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE TRUE
# Retorna um vetor contendo TRUE se
# for maior e FALSE se for menor
which(a>10)
[1] 6 7 8 9 10
# Equivale a pergunta: Quais valores
# de a sao maiores que 10?
```

Note que a resposta é a posição dos valores (o sexto, o sétimo) e não os valores que são maiores que 10.

```
a[6]
[1] 12
# selecionamos o sexto valor de a
a[c(6:10)]
[1] 12 14 16 18 20
# selecionamos do sexto
# ao décimo valores
```


Para saber os valores de a maiores que 10, simplesmente faça:

```
# Para saber os valores de a maiores que 10:
a[which(a>10)]
[1] 12 14 16 18 20
```

2.10.2 Função ifelse

Agora vamos aprender a usar o comando *ifelse* que significa: se for isso, então seja aquilo, se não, seja aquilo outro. O comando funciona da seguinte maneira:

```
ifelse(aplicamos um teste, especificamos
       o valor caso o teste for
       verdade, e o valor caso falso).
```

Complicado? Vamos ver alguns exemplos para facilitar as coisas.

```
salarios <- c(1000, 400, 1200, 3500, 380,
              3000, 855, 700, 1500, 500)
```

As pessoas que ganham menos de 1000 ganham pouco, concorda? Então aplicamos o teste e pedimos para retornar “pouco” para quem ganha menos de 1000 e “muito” para quem ganha mais de 1000.

```
# Se o salario é menor que 1000,
# seja pouco, se for maior seja muito.
ifelse(salarios < 1000, “pouco”, “muito”)
[1] “muito” “pouco” “muito” “muito” “pouco” “muito” “pouco”
“pouco” “muito”
[10] “pouco”
```

Este comando irá retornar um vetor composto pelas palavras “pouco” e “muito”. Então você poderá ver quantos terão em cada categoria.

2.10.3 Função for

O comando *for* é usado para fazer loopings, e funciona da seguinte maneira:

```
for(i in 1:n){comandos}
```

Isso quer dizer que: para cada valor i o RStudio vai calcular os comandos que estão entre as chaves {comandos}. O i in $1:n$ indica que os valores de i serão $i = 1$ até $i = n$. Ou seja, na primeira rodada do *for* o $i = 1$, na segunda $i = 2$, e assim por diante até $i = n$. Para salvar os resultados que serão calculados no *for*, precisamos criar um objeto vazio que receberá os valores calculados.

```
resu<-numeric(0)
for(i in 1:5){
  resu[i]<-i^2
} # Fim do for (i)
resu ## Para ver os resultados
```

2.11 CRIANDO FUNÇÕES

A sintaxe básica é:

```
function(lista de argumentos)
{corpo da funcao}
```

Você pode criar uma função que faça o que você quiser. Vamos ver como criar funções começando por uma função simples, que apenas simula a jogada de moedas (cara ou coroa).

Neste caso a função terá dois argumentos (x e n), em que x será a “moeda” com resultados “cara” e “coroa” e n será o número de vezes que deseja jogar a moeda. Vamos dar o nome a esta função de *jogar.moeda*.

```
# Criando a função
jogar.moeda<-function(x,n){
  sample(x,n, replace=T)
} # Fim da função
```

```
# Criando a moeda
moeda<-c("Cara","Coroa")
# Jogando a moeda 2 vezes
jogar.moeda(moeda,2)
# Jogando a moeda 10 vezes
jogar.moeda(moeda,10)
# Jogando a moeda 1000 vezes
jogar.moeda(moeda,1000)
```

Observe que após a criação da função (*jogar.moeda*), já podemos utilizá-la. No exemplo acima, utilizamos a função para jogar a moeda 2, 10 e 1000 vezes.

2.12 IMPORTANDO DADOS

Muitas vezes o seu banco de dados será um arquivo grande e que não poderá ser digitado na forma de lista de objetos. Será necessário então importar os dados, em formato de texto (.txt) ou planilha. Clique no botão *Import Dataset* no canto direito na parte de cima na aba *Environment*. Selecione o tipo de arquivo. Vamos importar o banco de dados *fsaude* (disponível em: <https://github.com/anaherminia88/EstatisticaLivre/blob/master/1.%20fsaude.xls>).

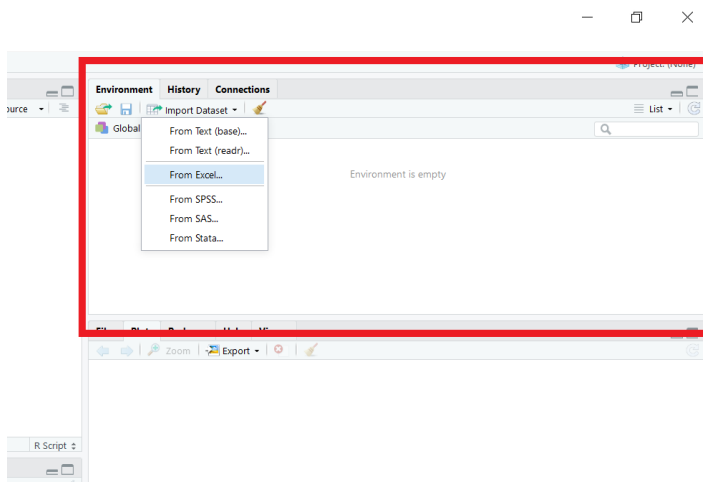


Figura 2.12.1: Importando um banco de dados.

Abrirá uma janela (ver Figura 2.12.2). Clique em *Browse*, selecione o caminho para encontrar o arquivo e depois clique em *Import*.

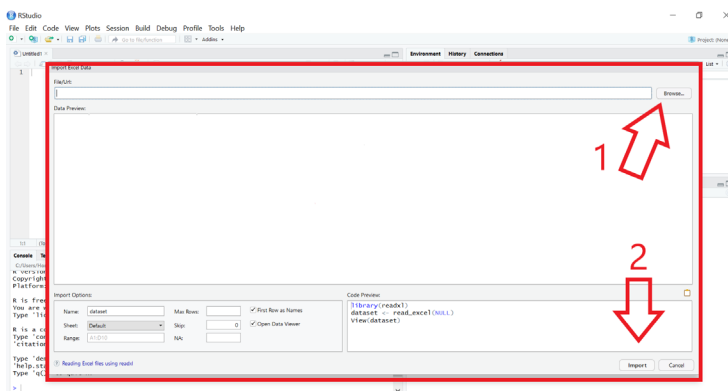


Figura 2.12.2: Janela para importação de dados.

Seus dados irão aparecer na parte superior, no canto esquerdo do ambiente do RStudio (ver Figura 2.12.3).

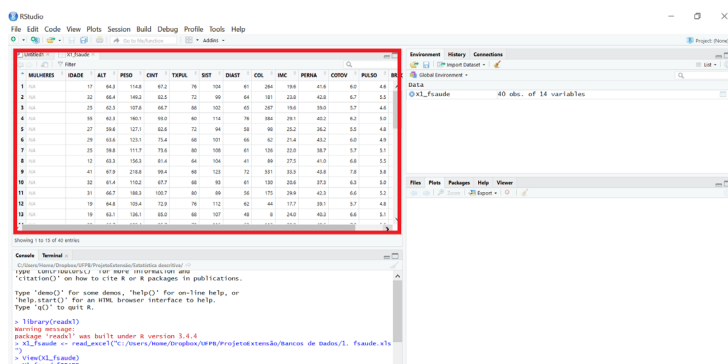


Figura 2.12.3: Dados importados no ambiente do RStudio.

Para acessar alguma variável basta digitar no console *nome do arquivo\$nome da variável*.

X1_fsaudef\$IDADE

Após a importação dos dados, para facilitar a análise, é importante atachar os dados para que o RStudio encontre as variáveis contidas neste banco de dados com mais facilidade. Para isso, usar a função `attach(nome do banco de dados)`:

```
attach(X1_fsaude)
```

Após este comando você poderá usar as variáveis do seu banco de dados livremente. Se preferir, depois de importar os dados, ao invés de atachá-los você pode renomeá-los criando um novo objeto pra receber cada variável do banco de dados.

```
idade = X1_fsaude$IDADE
```

2.13 FUNÇÃO FACTOR

Quando uma variável é qualitativa, é possível substituir os valores dos fatores pelos nomes usando a função `factor`.

```
factor(x = character(), levels,
       labels = levels, exclude = NA,
       ordered = is.ordered(x), nmax = NA)
```

Por exemplo, em um banco de dados (*filmes*, que encontra-se no link <https://github.com/anaherminia88/EstatisticaLivre/blob/master/bdfilmes.xlsx>, a variável a variável *Empresa* pode-se substituir os níveis de 1 a 5 pelo nomes das empresas:

```
Empresa <- factor(filmes$Empresa,
                 label = c("Disney", "MGM",
                           "Warner Bros", "Universal",
                           "20th Century Fox"),
                 levels = 1:5)
```

Com este comando, a variável *Empresa* será uma variável qualitativa cujos fatores serão os nome das empresas: "Disney", "MGM", "Warner Bros", "Universal", "20th Century Fox".

3 ESTATÍSTICA DESCRITIVA

A *Estatística Descritiva* é a base da análise de dados, sendo definida como um conjunto de técnicas que permite descrever, analisar e interpretar os dados referentes à uma população ou amostra. Na estatística descritiva o objetivo é resumir os dados coletados de forma a extrair conhecimento útil acerca do problema motivador da coleta de dados. Nessa fase da pesquisa, estamos preocupados em apresentar os dados em forma de tabelas e gráficos e em obter medidas que quantifiquem os resultados do estudo. Os principais elementos para a análise na estatística descritiva são:

- **Representação tabular:** a organização dos dados em tabelas proporciona um meio eficaz de estudo do comportamento de características de interesse. **Ex:** Distribuição de Frequências.
- **Representação gráfica:** proporciona uma interpretação imediata dos resultados devido à sua simplicidade e clareza.
- **Medidas resumo:** possibilitam representar um conjunto de dados relativo à observação de determinado fenômeno de forma resumida. São classificadas em medidas de posição, dispersão, assimetria e curtose.

Neste capítulo, vamos aprender a analisar os dados descritivamente. Para tanto, vamos utilizar os dados hipotéticos de 36 funcionários da companhia “Milsa”, retirados do livro *Estatística Básica-W. Bussab e P. Morettin*, disponível em <https://github.com/anaherminia88/EstatisticaLivre/blob/master/milsa.txt>. Com base nos conhecimentos adquiridos no Capítulo Primeiros Passos, primeiramente devemos fazer o *download* do banco de dados e importá-lo para o Rstudio.

Após a importação do banco de dados, vamos prepará-lo para ser analisado. Primeiramente vamos criar um *dataframe* chamado *dados* e colocar nosso banco de dados neste *dataframe*. Dessa forma, teremos nossos dados originais disponíveis e vamos editar apenas o *dataframe*.

```
dados = data.frame(milsa)
```

As variáveis *civil*, *instrucao* e *regiao* são variáveis do tipo qualitativas (atenção: *instrucao* é do tipo qualitativa ordinal), porém, no banco de dados elas estão representadas por números. Vamos associar cada um desses números a uma categoria.

```
dados$civil = factor(dados$civil, label = c("solteiro", "casado"),
  levels = 1:2)
dados$instrucao = factor(dados$instrucao, label = c("1º Grau", "2º
  Grau", "Superior"), lev = 1:3, ord= T)
dados$regiao = factor(dados$regiao, label = c("capital", "interior",
  "outro"), lev = c(2, 1, 3))
```

Adicionalmente, vamos criar a variável *idade* como sendo a soma dos anos inteiros e os meses divididos por doze. Dessa forma teremos a idade completa de cada indivíduo.

```
dados$idade <- dados$ano + dados$mes/12
```

Por fim, vamos utilizar o comando *attach* para que o programa reconheça todas as variáveis dentro do *dataframe dados*.

```
attach(dados)
```

Imprimir um resumo sobre o nosso banco de dados utilizando o comando *str*.

```
str(dados)
```

```
'data.frame': 36 obs. of 9 variables:
funcionario: int 1 2 3 4 5 6 7 8 9 10 ...
civil: Factor w/ 2 levels "solteiro","casado": 1 2 2 1 1 2 1 1 2 1 ...
instrucao: Ord.factor w/3 levels "1ºGau" < "2ºGau" < "...: 1 1 1 2 1 1 1 1 2 2
filhos : int NA 1 2 NA NA 0 NA NA 1 NA ...
salario : num 4 4.56 5.25 5.73 6.26 6.66 6.86 7.39 7.59 7.44 ...
ano : int 26 32 36 20 40 28 41 43 34 23 ...
mes : int 3 10 5 10 7 0 0 4 10 6 ...
regiao : Factor w/3 levels "capital","interior",...: 2 1 1 3 3 2 2 1 1 3
idade : num 26.2 32.8 36.4 20.8 40.6 ...
```

3.1 REPRESENTAÇÃO TABULAR

A representação tabular é a organização dos dados em tabelas, proporcionando um meio eficaz de estudo do comportamento de características de

interesse. Para tanto, utilizaremos a distribuição de frequências. No RStudio pode-se calcular as frequências simples e as relativas. A tabela de frequências simples é mais recomendada para variáveis do tipo qualitativas. As tabelas de frequências simples para variáveis do tipo quantitativa discreta podem ficar um pouco cansativas se esta assumir muitos valores e por isso é mais recomendável se calcular uma tabela de frequências por classes. Inicialmente, vamos calcular a frequência simples da variável qualitativa *civil* utilizando o comando *table*. Iremos colocar esta frequência dentro de um objeto chamado *freq*, pois iremos utilizar a frequência simples para calcular as demais frequências.

```
freq = table(civil)
freq
```

```
civil
solteiro  casado
      16      20
```

Podemos calcular a tabela de frequências relativas desta mesma variável, utilizando o comando *prop.table*. Observe que utilizamos a frequência simples para este cálculo, não os dados.

```
freq_rel = prop.table(freq)
freq_rel
```

```
civil
solteiro  casado
0.4444444 0.5555556
```

Caso queira calcular a porcentagem da variável em questão, basta multiplicar o resultado anterior por 100.

```
p_freq_rel = 100*prop.table(freq)
p_freq_rel
```

```
civil
solteiro  casado
44.44444  55.55556
```


Uma outra opção disponível no RStudio é construir uma tabela de frequências mais completa, utilizando o pacote *descr*. Vamos construir uma tabela de frequências para a variável *civil* utilizando o comando *freq*. Além da tabela, este comando nos retorna um gráfico de barras ou colunas. Saberemos mais sobre este gráfico em uma seção mais adiante.

```
library(descr)
```

```
freq(civil)
```



```
civil
      Frequência Percentual
solteiro      16      44.44
casado        20      55.56
Total         36     100.00
```

Como citado anteriormente, fazer uma tabela de frequências por valor para uma variável quantitativa com muitos valores pode não ser tão interessante. Podemos

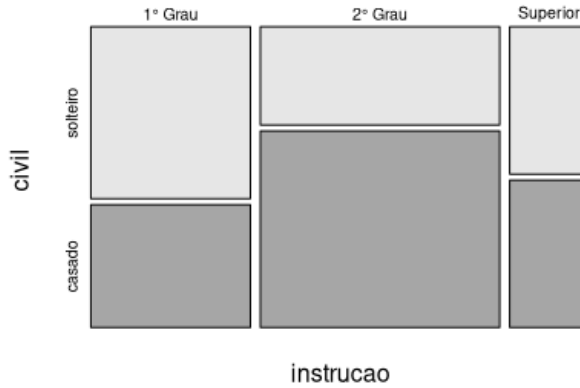
então construir uma tabela de frequências por classes, utilizando o comando *fdt* do pacote *fdth*. Vamos construir uma tabela de frequências por classes para a variável *salario*. Esta função retorna a tabela de frequências por classes por meio do método de Sturges por padrão (*default*).

```
library(fdth)
tabClasses= fdt(salario)
tabClasses
```

```
Class limits  f   rf rf(%)  cf  cf(%)
 [3.96,6.7561) 6 0.17 16.67  6 16.67
 [6.7561,9.5523) 10 0.28 27.78 16 44.44
 [9.5523,12.348) 7 0.19 19.44 23 63.89
 [12.348,15.145) 6 0.17 16.67 29 80.56
 [15.145,17.941) 4 0.11 11.11 33 91.67
 [17.941,20.737) 2 0.06  5.56 35 97.22
 [20.737,23.533) 1 0.03  2.78 36 100.00
```

Caso o interesse resida em descrever a relação entre duas variáveis qualitativas, podemos construir uma tabela de frequências bivariada ou cruzada. Vamos construir uma tabela cruzada para as variáveis *civil* e *instrucao*, utilizando o comando *crossstab* do pacote *descr*. Observe que além da tabela, a função retorna um gráfico de barras ou colunas bivariado. Saberemos mais sobre este gráfico mais adiante.

```
library(descr)
crossstab(civil, instrucao)
```



Conteúdo das células

```

|-----|
|               |
|               | Contagem |
|-----|

```

```

=====
civil      instruaao
          1º Grau  2º Grau  Superior  Total
-----
solteiro      7      6      3      16
-----
casado        5     12      3     20
-----
Total         12     18      6     36
=====

```

3.2 MEDIDAS RESUMO

Para resumir dados, em geral quantitativos, podemos fazer uso de algumas medidas, sendo estas medidas de posição ou de tendência central e medidas de dispersão. Inicialmente utilizaremos o comando *summary* para imprimir uma sumarização das variáveis do nosso banco de dados. Com isto, obteremos a frequência

simples de variáveis qualitativas, enquanto para as variáveis quantitativas obtemos os valores mínimo e máximo, os quartis e a média. A seguir temos o sumário das variáveis do banco de dados *Milsa*.

summary(dados)

```
funcionario  civil      instrucao      filhos      salario
Min.   : 1.00 solteiro:16  1º Grau:12    Min.:0.00   Min.: 4.000
1st Qu.: 9.75 casado  :20  2º Grau:18    1st Qu.:1.00 1st Qu.: 7.553
Median :18.50                Superior0: 6    Median :2.00 Median :10.165
Mean   :18.50                NA's      :16    Mean   :1.65 Mean   :11.122
3rd Qu.:27.25                NA's      :16    3rd Qu.:2.00 3rd Qu.:14.060
Max.   :36.00                NA's      :16    Max.   :5.00 Max.   :23.300

      ano      mes      regioao      idade
Min.   :20.00 Min.   : 0.000 capital :11 Min.   :20.83
1st Qu.:30.00 1st Qu.: 3.750 interior:12 1st Qu.:30.67
Median :34.50 Median : 6.000 outro  :13 Median :34.92
Mean   :34.58 Mean   : 5.611                Mean   :35.05
3rd Qu.:40.00 3rd Qu.: 8.000                3rd Qu.:40.52
Max.   :48.00 Max.   :11.000                Max.   :48.92
```

Agora que já temos uma ideia geral do nosso banco de dados, vejamos o cálculo isolado dessas e de outras medidas.

3.2.1 Média aritmética

A medida de posição mais utilizada é a média aritmética. É calculada somando-se os valores das observações da amostra ou população e dividindo-se o resultado pelo tamanho da amostra ou população. Assim, a média amostral é dada por

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} = \frac{X_1 + \dots + X_n}{n}.$$

No Rstudio, usaremos o comando *mean* para calcular a média aritmética da variável *salario*.

mean(salario)

```
[1] 11.12222
```

3.2.2 Mediana

A mediana é o valor que encontra-se no meio da distribuição ordenada dos dados. Esta medida é bastante utilizada para representar dados assimétricos, pois é menos influenciada por valores destoantes na distribuição de dados do que a média. Para calcular a mediana devemos, em primeiro lugar, ordenar os dados de forma crescente. Se o número de observações for ímpar, a mediana será a observação central. Se o número de observações for par, a mediana será a média aritmética das duas observações centrais. No Rstudio, usaremos a função *median* para calcular a mediana da variável *salario*.

```
median(salario)
```

```
[1] 10.165
```

3.2.3 Quantis

Os quantis são valores dados a partir do conjunto de observações ordenado de forma crescente, que dividem a distribuição em partes iguais. O mais usual é dividir a distribuição em quatro partes. Neste caso temos os quartis, em que 25% das observações estão abaixo do primeiro quartil, Q1, e 75% estão acima de Q1. O segundo quartil, Q2, divide as observações no meio, estando 50% abaixo de Q2 e 50% acima de Q2. Observe que Q2 nada mais é do que a mediana. Por fim, 75% das observações abaixo do terceiro quartil, Q3, e 25% estão acima de Q3. Se dividirmos os dados em dez partes iguais, teremos os decis, em cem partes iguais teremos os centis e assim por diante. No Rstudio, para calcular os quartis da variável *salario*, usaremos a função *quantile*.

```
quantile(salario)
```

```
  0%    25%    50%    75%   100%
4.0000 7.5525 10.1650 14.0600 23.3000
```

Caso queiramos calcular outros quantis, faremos uso do atributo *probs*. Por exemplo, se quisermos calcular os decis:

```
quantile(salario,probs = seq(0, 1, 0.1))
```

```
 0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
4.000 5.995 7.390 8.290 9.130 10.165 11.590 13.415 14.710 16.935 23.300
```

3.2.4 Moda

A moda de um conjunto de dados é o valor que apresenta a maior frequência. Para se calcular a moda no RStudio, é por meio de uma função, sem a necessidade de instalar um pacote.

```
names(table(idade))[table(idade)==max(table(idade))]
```

```
[1] "41"
```

Observação: A moda pode não existir, caso todos os elementos se repitam com a mesma frequência, ou pode não se única, caso um ou mais valores se repitam com a mesma frequência máxima. Além disso a moda também pode ser calculada para variáveis qualitativas, basta observar a categoria ou categorias com maior frequência.

3.2.5 Variância

A variância de uma amostra X_1, \dots, X_n de n elementos é definida como a soma dos desvios quadráticos dos elementos em relação à sua média \bar{x} dividida por $(n-1)$ dada por

$$s^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}.$$

No Rstudio, usaremos o comando *var* para calcular a variância amostral da variável *salario*.

```
var(salario)
```

```
[1] 21.04477
```

Observação: Como utilizamos a soma dos desvios quadráticos para calcular a variância, esta terá unidade de medida quadrática, o que pode ser de difícil interpretação.

3.2.6 Desvio-padrão

O desvio padrão amostral de um conjunto de dados é a raiz quadrada da variância amostral. No Rstudio, usaremos o comando *sd* para calcular o desvio-padrão amostral da variável *salario*.

```
sd(salario)
```

```
[1] 4.587458
```

Observação: A unidade do desvio padrão é a mesma dos dados.

3.2.7 Outras formas de obter medidas resumo

No Rstudio, podemos utilizar a função *tapply* do pacote *descr* para calcular medidas de uma variável quantitativa, para cada categoria de uma outra variável qualitativa do banco de dados. Por exemplo, vamos calcular algumas medidas para a variável *salario* para cada uma das categorias da variável *instrucao*.

```
tapply(salario, instrucao, mean)
```

```
1° Grau 2° Grau Superior
7.836667 11.528333 16.475000
```

```
tapply(salario, instrucao, sd)
```

```
1° Grau 2° Grau Superior
2.956464 3.715144 4.502438
```

```
tapply(salario, instrucao, quantile)
```

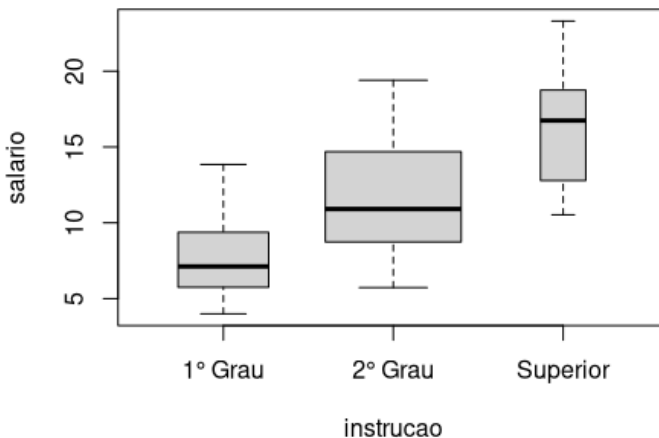
```
$`1° Grau`
  0%    25%   50%   75%  100%
4.0000 6.0075 7.1250 9.1625 13.8500

$`2° Grau`
  0%    25%   50%   75%  100%
5.7300 8.8375 10.9100 14.4175 19.4000

$Superior
  0%    25%   50%   75%  100%
10.5300 13.6475 16.7400 18.3775 23.3000
```

Temos ainda a função *compmeans* do pacote *descr* que nos retorna a média e o desvio padrão da variável quantitativa para o total da amostra e para cada categoria da variável qualitativa, além do tamanho da amostra total e de cada categoria. Adicionalmente, a função gera um gráfico box-plot da variável quantitativa para cada classe da variável qualitativa. Por exemplo, vamos utilizar esta função para calcular medidas da variável *salario*, bem como para cada classe da variável *instrucao*.

```
compmeans(salario, instrucao)
```



Valor médio de “salario” segundo “instrucao”

	Média	N	Desv. Pd.
1° Grau	7.836667	12	2.956464
2° Grau	11.528333	18	3.715144
Superior	16.475000	6	4.502438
Total	11.122222	36	4.587458

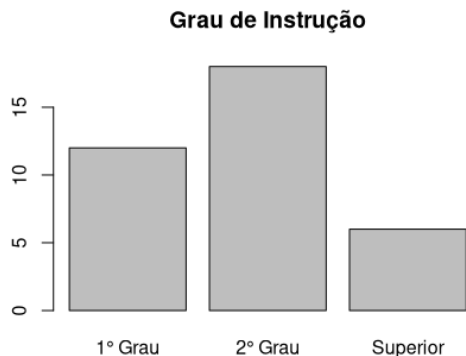
3.3 REPRESENTAÇÃO GRÁFICA

Como foi dito anteriormente, a representação gráfica proporciona uma interpretação imediata dos resultados, devido a sua simplicidade e clareza. Nesta seção iremos aprender a gerar alguns gráficos simples no RStudio.

3.3.1 Gráfico em Barras ou colunas

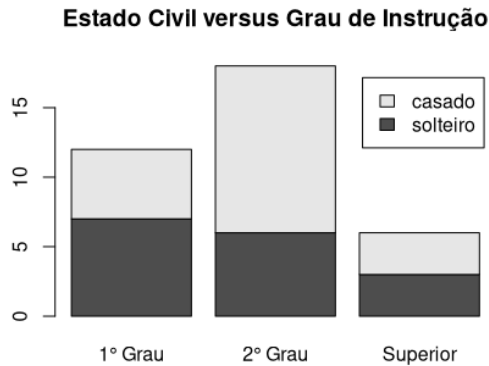
O gráfico de barras ou colunas tem a finalidade de comparar grandezas por meio de barras de igual largura e alturas proporcionais às respectivas grandezas. Em geral, utilizamos como grandeza a frequência simples. É apropriado para representar variáveis qualitativas e quantitativas discretas. Vamos construir este gráfico para a variável *instrucao*.

```
barplot(table(instrucao), main = “Grau de Instrução”)
```

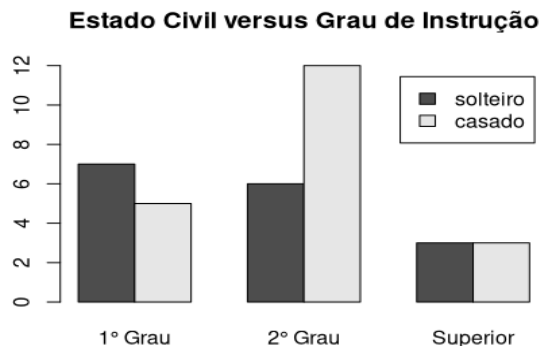


Este gráfico também pode ser gerado para mais de uma variável, com as barras sobrepostas ou lado a lado, respectivamente.

```
barplot(table(civil, instrucao), legend = T,
        main = "Estado Civil versus Grau de Instrução")
```

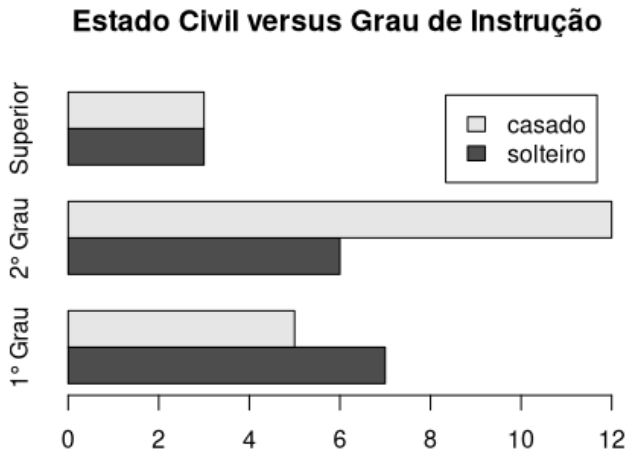


```
barplot(table(civil, instrucao), beside = T, legend = T,
        main = "Estado Civil versus Grau de Instrução")
```



Caso queira fazer qualquer um dos gráficos anteriores com as colunas horizontais, devemos adicionar o argumento *horiz = T*.

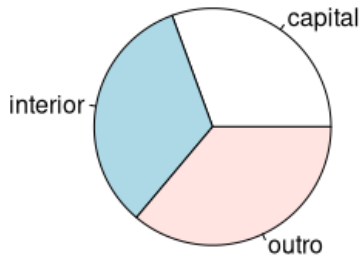
```
barplot(table(civil, instrucao), beside = T, legend = T,
        horiz = T,
        main = "Estado Civil versus Grau de Instrução")
```



3.3.2 Gráfico de Pizza ou Setores

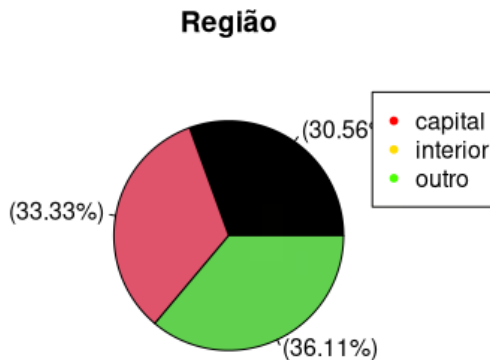
O gráfico de pizza ou setores é apropriado para representar variáveis qualitativas e quantitativas discretas quando o número de categorias é relativamente baixo. A construção de um gráfico de setores parte do fato que o número total de graus de um arco de circunferência é 360°. Vamos construir este gráfico para a variável *regiao*.

```
pie(table(regiao))
```



Podemos ainda inserir os percentuais no gráfico.

```
porc = round(table(regiao)*100/sum(table(regiao)),2)
rotulos = paste("'",porc,"%'",sep="")
pie(table(regiao), main="Região",labels = rotulos, col=palette())
legend(1,1,levels(regiao),col=rainbow(7),pch = rep(20,6))
```



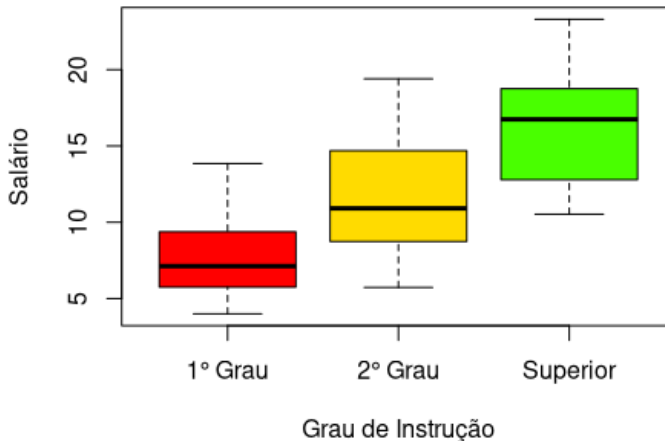
3.3.3 Gráfico box-plot

Para construir um box-plot utilizamos os quartis. Com este gráfico, podemos observar a assimetria da variável, a dispersão e a presença de pontos aberrantes ou outliers. Adicionalmente podemos utilizá-lo para comparar visualmente dois ou mais

grupos. As hastes inferiores e superiores do gráfico se estendem, respectivamente, do primeiro quartil até o limite inferior e do terceiro quartil até o limite superior.

Os pontos fora destes limites são considerados valores discrepantes (outliers). Outro ponto importante é amplitude interquartilica, definida como sendo a diferença entre os quartis ($Q_3 - Q_1$), sendo esta uma medida de variabilidade dos dados. Vamos construir um box-plot para a variável *salario* para cada uma das categorias da variável *instrucao*.

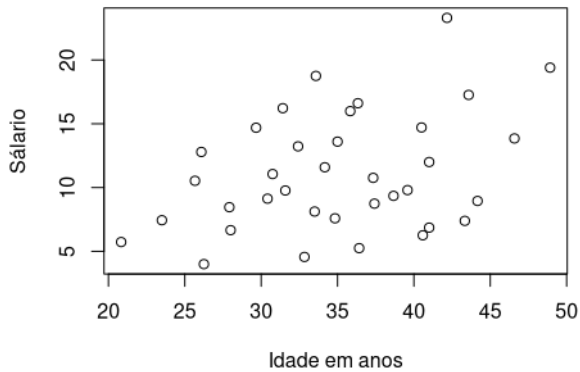
```
boxplot(salario ~ instrucao,col= rainbow(7),
xlab = "Grau de Instrução",ylab = "Salário")
```



3.3.4 Gráfico de dispersão

Os diagramas ou gráficos de dispersão são representações de duas (tipicamente) ou mais variáveis. Este gráfico é construído utilizando coordenadas cartesianas para exibir os valores como uma coleção de pontos. Para construir um gráfico de dispersão no RStudio, podemos utilizar o comando *plot*. Vamos construir um gráfico de dispersão para a variável *idade*.

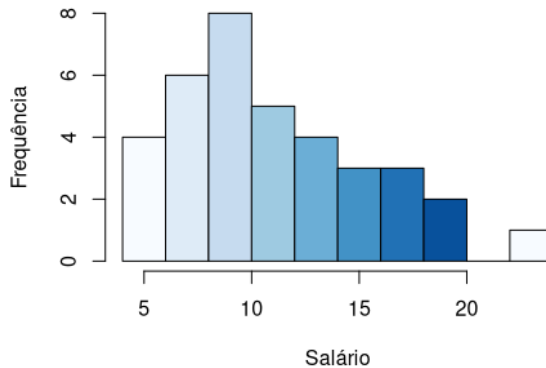
```
plot(idade, salario,xlab = "Idade em anos",ylab = "Sálario")
```



3.3.5 Histograma

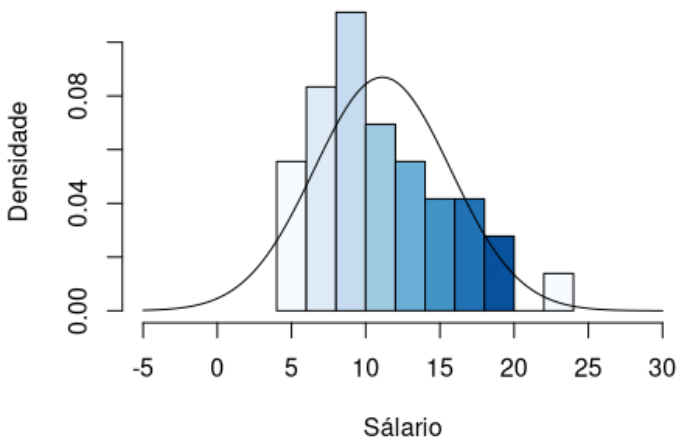
O histograma é a representação gráfica de uma distribuição de frequências por meio de retângulos justapostos, cujas áreas são proporcionais às frequências das classes. Vale mencionar que, podemos utilizar tanto as frequências absolutas simples quanto as relativas simples para construir o histograma. Vamos construir um histograma para a variável *salario* utilizando o comando *hist*.

```
hist(salario,xlab = "Salário",ylab = "Frequência",
     col = blues9,main = "")
```



O histograma é frequentemente utilizado para verificar a simetria dos dados. Para visualizar melhor a simetria dos dados, podemos incluir a curva gaussiana neste gráfico.

```
hist(salario,xlab = "Sálario",ylab = "Densidade",col = blues9,freq = F,xlim
= c(-5,30),main = "")
curve(dnorm(x,mean = mean(salario),sd = sd(salario)), add = T)
```



3.3.6 Gráfico de linhas

O gráfico de linhas é frequentemente utilizado na representação de séries de tempo. As linhas são mais eficientes neste tipo de gráfico, pois permitem a detecção de flutuações ou mudanças intensas nas séries e também possibilitam a representação de várias séries no mesmo gráfico. Para construir um gráfico em linhas no RStudio, vamos utilizar função *plot* com o argumento *type = l* e acrescentar outras linhas

utilizando o comando `lines`. Considere a seguir os dados acerca da taxa de ocupação dos hotéis no Rio de Janeiro por trimestre no período de 2001 a 2009.

```
ano = 2001:2009
tri1 = c(72.8,66.2,69.2,65.9,62.4,67.8,61.3,68.5,70.4)
tri2 = c(60.6,53.7,55.3,56.7,56.4,57.8,57.5,59.8,63.3)
plot(ano, tri1,type="l",main="Taxa de ocupação dos hotéis-
RJ",xlab="ano",ylab="Taxa de ocupação (%)",
col="blue",ylim=c(50,80))
lines(ano,tri2,col="red")
legenda1 = c("1ºtrimestre","2ºtrimestre")
legend(x="topright",legend=legenda1,fill=c("blue","red"), bty="n")
```

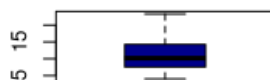
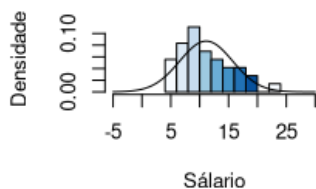
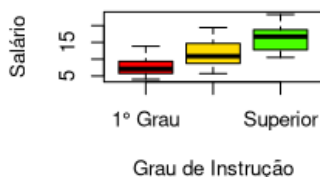
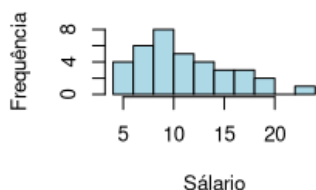


3.4 MAIS RECURSOS GRÁFICOS

Nesta seção iremos aprender mais alguns recursos gráficos disponíveis no RStudio, iniciando pela impressão dos gráficos. Com o comando `par` é possível imprimir mais de um gráfico ao mesmo tempo. Com o argumento `mfrow` determinamos o

número de gráficos que seremos capazes de imprimir, como em uma matriz. Se temos `mfrow = c(2,2)`, isso significa que teremos espaço para imprimir quatro gráficos.

```
par(mfrow = c(2,2));hist(salario, col = "lightblue",
  main = "",xlab = "Salário",ylab = "Frequência")
boxplot(salario ~ instrucao,col=rainbow(7),
  xlab = "Grau de Instrução",ylab = "Salário")
hist(salario,xlab = "Salário",main = "",col = blues9,
  freq = F,xlim = c(-5,30),ylab = "Densidade")
curve(dnorm(x,mean = mean(salario),sd = sd(salario)),
  add = T);boxplot(salario,col="darkblue")
```



Outro recurso disponível é a função `colors()`. Observe que nos gráficos que construímos até então utilizamos o argumento `col` para modificar a cor de alguns gráficos. Com a função `colors()` é possível ter acesso ao nome de 657 cores disponíveis, listadas em ordem alfabética.

`colors()`

Para saber mais sobre parâmetros gráficos, como outras cores, eixos, títulos, símbolos, acesse o site <https://www.statmethods.net/advgraphs/parameters.html>.

3.5 GRÁFICOS 3D

Por vezes, surge a necessidade de analisar a relação de mais de duas variáveis. No RStudio é possível construir gráficos em terceira dimensão, utilizando o pacote *scatterplot3d*.

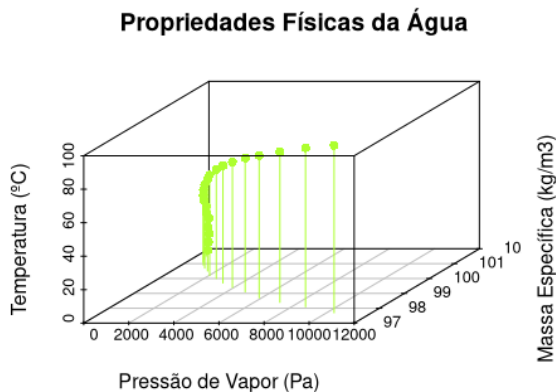
```
library(scatterplot3d)
```

Considere o banco de dados sobre as propriedades físicas da água, retirado do livro “Fundamentos da Engenharia Hidráulica - M. Baptista e M. Lara”. Os dados estão disponíveis em <https://github.com/anaherminia88/EstatisticalLivre/blob/master/agua.txt>.

```
attach(agua)
```

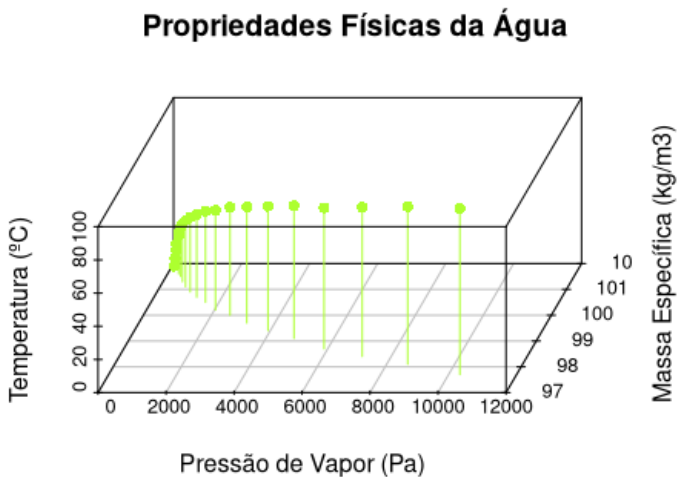
Observe que todas as variáveis são quantitativas. Vamos construir um gráfico tridimensional para as variáveis *pressao*, *massa* e *temp*. As variáveis ocuparão os eixos X, Y e Z respectivamente por ordem de inserção.

```
scatterplot3d(pressao, massa, temp, xlab = "Pressão de Vapor (Pa)", ylab = "Massa Específica (kg/m3)", zlab = "Temperatura (°C)", main = "Propriedades Físicas da Água", pch = 16, color = "greenyellow", type = "h")
```



As linhas de cada ponto até a superfície inferior do gráfico foram inseridas utilizando o argumento `type = "h"`. É possível ainda modificar o ângulo de visualização, utilizando o argumento `angle`.

```
scatterplot3d(pressao, massa, temp, xlab = "Pressão de Vapor (Pa)",
  ylab = "Massa Específica (kg/m3)", zlab = "Temperatura (°C)",
  main = "Propriedades Físicas da Água", pch = 16,
  color = "greenyellow", type = "h", angle = 70)
```



Podemos gerar uma versão interativa do gráfico anterior, utilizando os pacotes `rgl` e `car`.

```
library("car")
library("rgl")
scatter3d(pressao, massa, temp, point.col = "greenyellow",
  surface = FALSE, xlab = "Pressão Vapor (Pa)",
  ylab = "Massa Específica (kg/m3)",
  zlab = "Temperatura (C)")
```

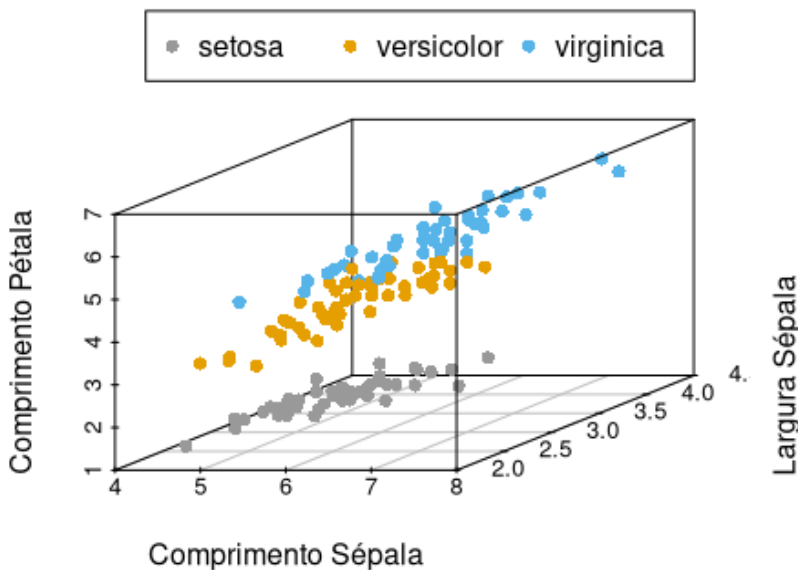
Considere agora um conjunto de dados que contém informações acerca das dimensões em centímetros da pétala e sépala de 50 flores de três diferentes

espécies de íris. Não há necessidade de fazer o *download* desses dados, pois eles fazem parte do conjunto de bancos de dados do *software*. Iremos acessá-los utilizando o comando *data*.

```
data(iris)
```

Vamos construir agora um gráfico em terceira dimensão para as variáveis *Sepal.Length*, *Sepal.Width* e *Petal.Length*, indicando pela cor as classes de uma variável qualitativa, neste caso a espécie das flores.

```
cores = c("#999999", "#E69F00", "#56B4E9"); cores = cores[as.numeric(iris$Species)]
scatterplot3d(iris[,1:3], pch = 16, color=cores, xlab = "Comprimento SÉpala", ylab = "Largura SÉpala", zlab = "Comprimento Pétala")
legend("top", legend = levels(iris$Species), col = c("#999999", "#E69F00", "#56B4E9"), pch = 16, inset = -0.25, xpd = TRUE, horiz = TRUE)
```



3.6 GRAMÁTICA DOS GRÁFICOS

O conceito de gramática dos gráficos foi definido por Leland Wilkinson em seu livro *The Grammar of Graphics*. Este conceito nos possibilita construir gráficos de uma forma diferente, utilizando camadas. Imagine que para escrever e compreender uma frase, precisamos adicionar as palavras e depois fazer a leitura da frase. Aqui utilizaremos o mesmo conceito, em que inserimos os elementos gráficos em camadas e ao final fazemos a compilação das camadas para poder visualizar o gráfico de forma completa. A gramática dos gráficos é composta por sete camadas:

- **Dados:** É a primeira camada do gráfico, sendo sua base. Nesta camada indicamos a base.
- **Estética:** Nesta camada definimos as variáveis e possíveis agrupamentos.
- **Geometria:** Na camada geometria definimos a forma dos elementos gráficos.
- **Facets:** Camada utilizada para dividir o gráfico em mais de uma parte.
- **Estatística:** É possível inserir uma análise de dados utilizando a camada estatística.
- **Coordenadas:** Possibilita a escolha das coordenadas do gráfico.
- **Temas:** É possível definir um estilo para visualização geral do gráfico utilizando a camada temas.

As três primeiras camadas são obrigatórias para a construção do gráfico. No RStudio, o conceito de gramática dos gráficos está implementado no pacote *ggplot2*.

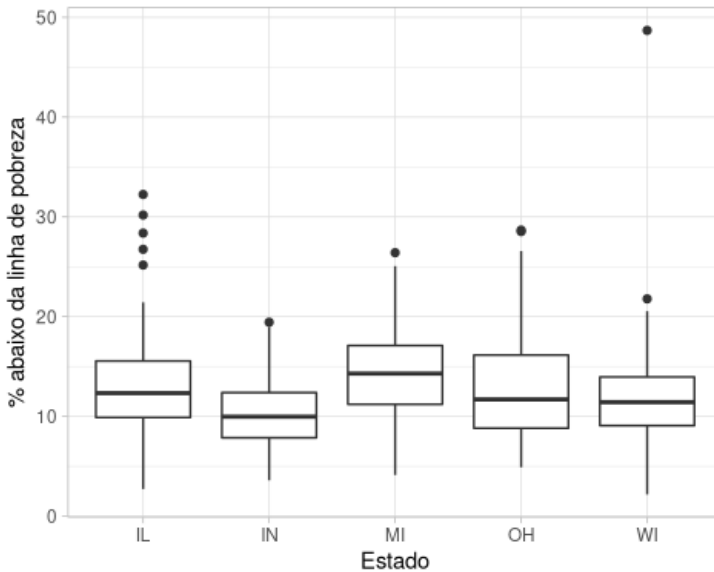
```
library(ggplot2)
```

Para elaborar gráficos utilizando o pacote *ggplot2* iremos considerar a base de dados desse pacote chamada *midwest*, que contém informações demográficas acerca dos estados do centro-oeste dos EUA. Para acessá-la basta carregar o pacote e utilizar o comando *data*.

```
data("midwest")
```

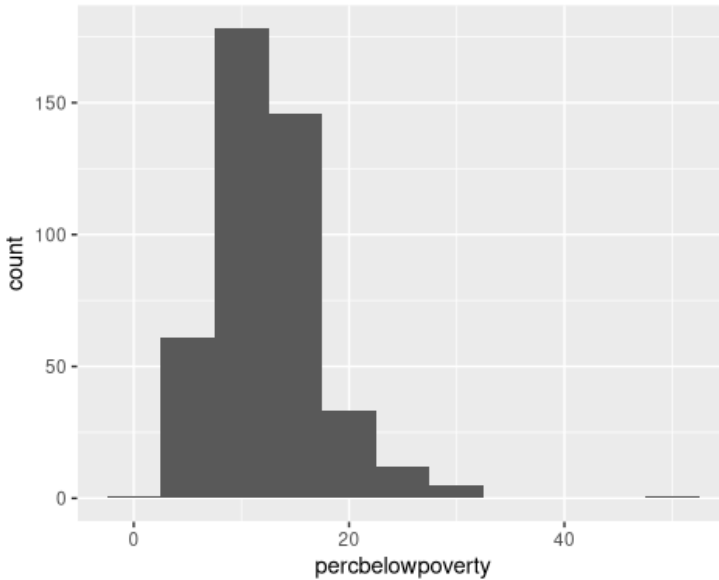
Vamos construir um box-plot para a variável *percbelowpoverty* por estado.

```
ggplot(midwest, aes(state, percbelowpoverty)) +
  geom_boxplot() +
  theme_light() +
  xlab("Estado") +
  ylab("% abaixo da linha de pobreza")
```



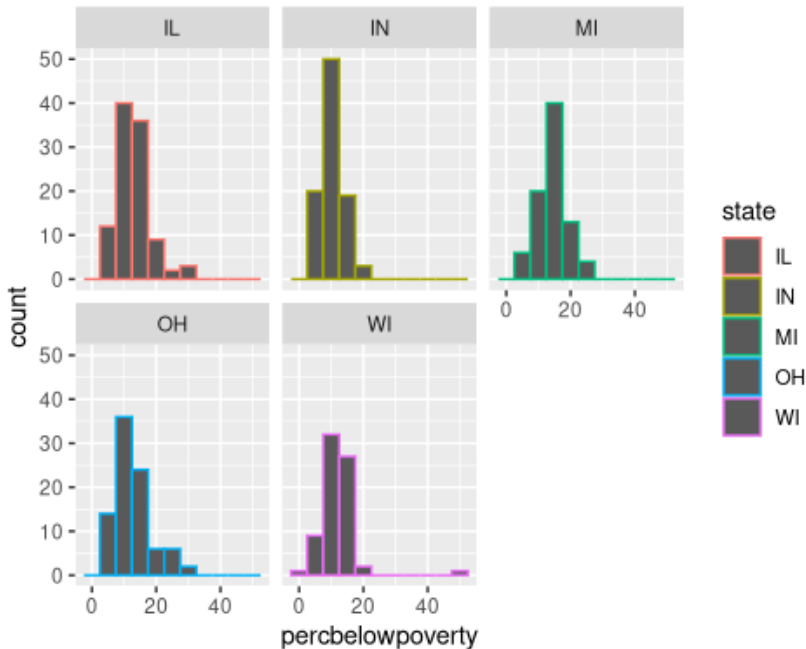
Observe que para acrescentar as camadas dados e estética utilizamos o comando *ggplot*. Para adicionar uma nova camada inserimos o sinal “+”. Na segunda camada, geometria, indicamos que nosso gráfico será um box-plot, utilizando a função *geom_boxplot()*. Dentre diversos temas disponíveis no pacote *ggplot2* escolhemos como ilustração o tema *light* utilizando a função *theme_light()*. Por fim definimos os títulos dos eixos X e Y, respectivamente, utilizando as funções *xlab* e *ylab*. Iremos agora construir um histograma para a variável *percbelowpoverty*.

```
ggplot(midwest, aes(percbelowpoverty)) +
  geom_histogram(binwidth = 5)
```



Podemos observar que neste gráfico utilizamos menos camadas que no anterior. Definimos o banco de dados e a variável utilizando o comando `ggplot`, enquanto indicamos a geometria como histograma utilizando o comando `geom_histogram`. Com o argumento `binwidth` podemos definir a largura do retângulo. Podemos ainda fazer mais de um gráfico por vez, por exemplo, um histograma para a variável `percbelowpoverty` por estado.

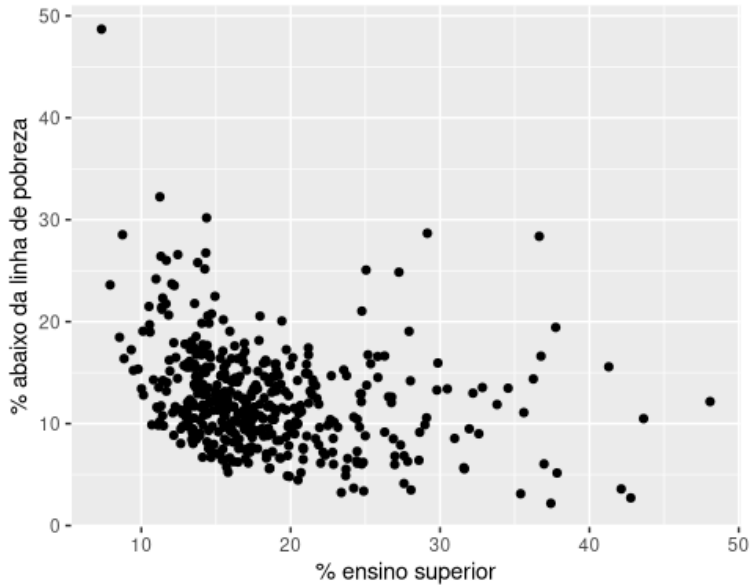
```
ggplot(midwest, aes(percbelowpoverty, color = state)) +
  geom_histogram(binwidth = 5) +
  facet_wrap(~state, nrow = 2, ncol = 3)
```



Observe que na camada estética indicamos não apenas as variáveis, como também um agrupamento por estado, utilizando o argumento `color = state`. Adicionamos também a camada `facets` utilizando a função `facet_wrap`, em que indicamos a variável qualitativa a ser utilizada para agrupar os dados, como também em quantas partes iremos dividir o nosso gráfico, utilizando o sistema matricial.

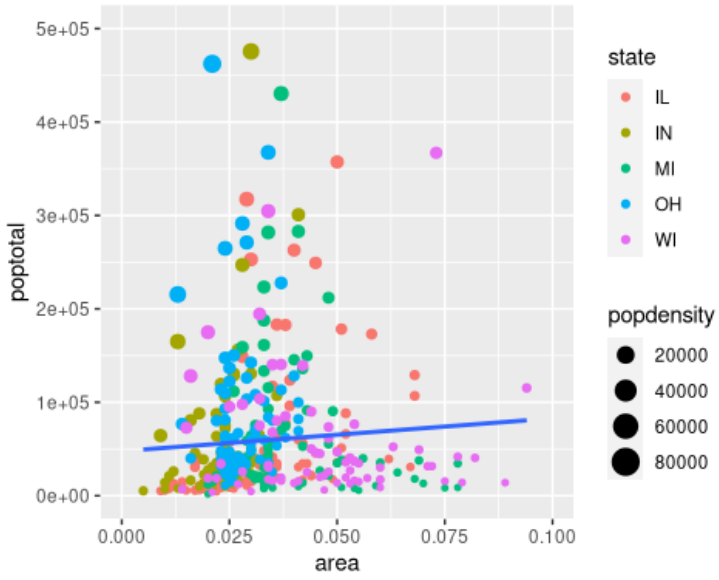
Podemos construir diversos tipos de gráficos com este pacote, definindo o tipo na camada geometria. Outro exemplo é o gráfico de dispersão. Vamos construir um gráfico de dispersão para as variáveis `percollege` e `percbelowpoverty` por estado.

```
ggplot(midwest, aes(percollege, percbelowpoverty)) +
  geom_point() +
  xlab("% ensino superior") +
  ylab("% abaixo da linha de pobreza")
```

Para indicar o gráfico de dispersão utilizamos a função `geom_point`. Considere agora a construção de outro gráfico de dispersão, agora para as variáveis `area` e `poptotal`. Neste novo cenário deseja-se indicar no gráfico o estado por cor e a densidade populacional por tamanho dos pontos. Iremos adicionar ainda um modelo utilizando a função `geom_smooth` que corresponde a camada estatística.

```
ggplot(midwest, aes(area, poptotal)) +
  geom_point(aes(col = state, size = popdensity)) +
  xlim(c(0, 0.1)) +
  ylim(c(0, 500000)) +
  geom_smooth(method = "lm", se = F)
```



4 INFERÊNCIA ESTATÍSTICA

Usualmente é impraticável observar toda uma população seja pelo custo caríssimo, tempo ou por dificuldades diversas. Nesses casos, examina-se então uma amostra. Se essa amostra for bastante representativa, os resultados obtidos poderão ser generalizados para toda a população. O uso de informações da amostra para concluir sobre o todo faz parte da atividade diária da maioria das pessoas. Há inúmeras situações reais em que se procura determinar valores para quantidades desconhecidas como médias e proporções. Por exemplo, empresários que têm interesse em saber a quantia média gasta por um turista em sua cidade; um produtor de televisão que deseja saber qual o índice de audiência de determinados programas, ou ainda, um engenheiro de qualidade que pretende determinar a proporção de itens defeituosos produzidos em uma linha de produção.

Um experimento pode ter por finalidade a determinação da estimativa de um parâmetro de uma função. A estimação pontual ocorre quando, a partir de uma amostra aleatória, um único valor é usado para estimar o parâmetro desconhecido. Esse tipo de estimação não possui uma medida do possível erro cometido na estimação. Portanto, uma maneira de expressar a precisão da estimação é estabelecer limites, que com certa probabilidade incluam o verdadeiro valor do parâmetro da população. Esses limites são chamados de Limites de Confiança e determinam um *intervalo de confiança*, no qual deverá estar o verdadeiro valor do parâmetro.

Ao ser feita determinada afirmação sobre uma população, mais especificamente sobre um parâmetro dessa população, é natural desejar saber se os resultados experimentais provenientes de uma amostra contrariam, ou não, tal afirmação. Nesse contexto é possível a realização de um *teste de hipóteses*. Essa ferramenta estatística permite ao pesquisador levantar hipóteses sobre uma população, a partir de uma dada amostra, e tomar a decisão entre rejeitar, ou não, a hipótese testada levando em consideração uma margem de erro.

Toda conclusão obtida por um procedimento de amostragem, quando generalizada para a população, virá acompanhada de um grau de incerteza ou risco. Ao conjunto de técnicas e procedimentos que permitem dar ao pesquisador

um grau de confiabilidade nas afirmações que faz para toda a população, baseadas nos resultados das amostras, damos o nome de Inferência Estatística. O problema fundamental da Inferência Estatística, portanto, é medir o grau de incerteza ou risco dessas generalizações. No presente capítulo, os instrumentos da estatística inferencial a serem abordados, de modo a permitir a viabilidade das conclusões por meio de afirmações estatísticas, são os **Intervalos de Confiança** e os **Testes de Hipóteses**.

4.1 INTERVALOS DE CONFIANÇA

Um intervalo de confiança é uma amplitude de valores, derivados de estatísticas de amostras, que têm a probabilidade de conter o valor de um parâmetro populacional desconhecido. Devido à sua natureza aleatória, é improvável que duas amostras de uma determinada população produzirá intervalos de confiança idênticos. Mas, se você repetir sua amostra várias vezes, uma determinada porcentagem dos intervalos de confiança resultantes conterá o parâmetro populacional desconhecido. Tais intervalos são determinados calculando-se uma **estimativa de ponto** e, depois, determinando sua **margem de erro**.

- **Estimativa de Ponto:** Este valor único estima um parâmetro populacional usando os seus dados amostrais.
- **Margem de Erro:** Quando você usa estatísticas para estimar um valor, é importante lembrar-se de que não importa quão bem seu estudo foi projetado, sua estimativa está sujeita a erros de amostragem aleatórios. A margem de erro quantifica esse erro e indica a precisão da sua estimativa.

Portanto, pode-se entender que os intervalos de confiança são ferramentas estatísticas utilizadas para indicar a confiabilidade de uma estimativa.

Definição: A partir da amostra procura-se construir um intervalo de variação, $\hat{\theta}_1 \leq \theta \leq \hat{\theta}_2$ com certa probabilidade de conter o verdadeiro parâmetro populacional. Isto é, consiste na fixação de dois valores tais que $(1 - \alpha)$ seja a probabilidade de que o intervalo, por eles determinado, contenha o verdadeiro valor do parâmetro.

4.1.1 Intervalo de Confiança para a Média Populacional

Um intervalo de confiança para uma média específica um intervalo de valores dentro do qual o parâmetro populacional desconhecido, neste caso a média, pode estar. Estes intervalos podem ser usados, por exemplo, por um fabricante que deseja estimar sua produção média diária ou por um pesquisador que deseja estimar a resposta média por pacientes a uma nova droga. Quando deseja-se estimar a média de uma população, através de uma amostra, temos dois casos distintos a considerar: quando a variância da população é conhecida e quando ela é desconhecida. Ambas as situações são abordadas nos tópicos a seguir.

4.1.1.1 Intervalo de Confiança para a Média Populacional com Variância Populacional Conhecida

Consideremos uma amostra aleatória simples, X_1, \dots, X_n , obtida de uma população com distribuição Normal, com média μ e variância σ^2 conhecida. Desta forma, a distribuição amostral da média também é Normal com média μ e variância σ^2/n . Assim, temos que,

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$$

é a variável com distribuição Normal Padrão, $N(0,1)$. Considerando que a probabilidade da variável Z tomar valores entre $-Z_{\alpha/2}$ e $Z_{\alpha/2}$ seja igual a $(1 - \alpha)$, pode-se definir o seguinte intervalo de confiança para a média populacional

$$IC[\mu; 1 - \alpha] = \left[\bar{X} - Z_{\alpha/2} \frac{\sigma}{\sqrt{n}}; \bar{X} + Z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \right]$$

Exemplo 4.1.1.1: Um pesquisador está estudando a resistência de um dado material sob determinadas condições. Ele sabe que essa variável é normalmente

distribuída com desvio padrão de 2 unidades. Utilizando a amostra: 4,9; 7,0; 8,1; 4,5; 5,6; 6,8; 7,2; 5,7; 6,2 unidades, determine o intervalo de confiança para a resistência média com um nível de confiança de 95%.

Resolução 01 (4.1.1.1). Função do R (Utilizando Intervalo de Confiança):

```
library(OneTwoSamples)
dadosR = c(4.9, 7.0, 8.1, 4.5, 5.6, 6.8, 7.2, 5.7, 6.2)
interval_estimate1(dadosR, sigma = 2, alpha = 0.05)
      mean df      a      b
[1] 6.222222 9 4.91558 7.528865
```

Resolução 02 (4.1.1.1). Função do R (Utilizando Teste de Hipótese):

```
library(TeachingDemos)
dadosR2 = z.test(dadosR, mean(dadosR), stdev = 2, conf.level
= 0.95)
```

```
dadosR2$conf.int
[1] 4.915580 7.528865
```

```
attr(,"conf.level")
[1] 0.95
```

Análise do IC 4.1.1.1: Ao nível de confiança de 95% a verdadeira resistência média do material é de no mínimo 4,92 e de no máximo 7,53 unidades.

Observação: Perceba que, as bibliotecas **OneTwoSamples** e **TeachingDemos** são necessárias ao usarmos as funções *interval_estimate1* e *z.test*, respectivamente.

4.1.1.2 Intervalo de Confiança para a Média Populacional com Variância Populacional Desconhecida

No caso de **amostras grandes**, se desconhecermos σ^2 , é possível estimá-la por ponto através de S^2 , baseado em uma amostra aleatória. Nesse caso, o intervalo de confiança para a média populacional é definido por

$$IC[\mu; 1 - \alpha] = \left[\bar{X} - Z_{\alpha/2} \frac{\sigma}{\sqrt{n}}; \bar{X} + Z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \right]$$

Exemplo 4.1.1.2.1: Foram realizados testes glicêmicos em pacientes após um jejum de 8 horas. Os resultados (em mg/dL) são apresentados na tabela abaixo. Construir um intervalo de confiança para a média μ , ao nível de confiança de 95%.

80	117	112	91	100	84	104	80	101	95
77	132	118	102	73	103	140	82	92	120
95	78	88	90	102	121	83	88	107	117

Resolução 01 (4.1.1.2.1). Função do R (Utilizando Intervalo de Confiança):

```
library(OneTwoSamples)
dadosG = c(80, 117, 112, 91, 100, 84, 104, 80, 101, 95, 77, 132,
118, 102, 73, 103, 140, 82, 92, 120, 95, 78, 88, 90, 102, 121, 83,
88, 107, 117)
interval_estimate1(dadosG, sd(dadosG), alpha = 0.05)
      mean df      a      b
1 99.06667 30 92.91812 105.2152
```

Resolução 02 (4.1.1.2.1). Função do R (Utilizando Teste de Hipótese):

```
library(TeachingDemos)
dadosG2 = z.test(dadosG, mean(dadosG), sd(dadosG), conf.level = 0.95)
dadosG2$conf.int
[1] 92.91812 105.21522
attr(,"conf.level")
[1] 0.95
```

Análise do IC 4.1.1.2.1: Ao nível de confiança de 95% o verdadeiro nível médio de glicemia dos pacientes é de no mínimo 92,92 mg/dL e de no máximo 105,22 mg/dL após 8 horas de jejum.

No caso de **amostras pequenas**, quando a variância da população não é conhecida, deve-se estimá-la com base no conjunto amostral. Adota-se como estimador a variância amostral, S^2 . A simples substituição da variância populacional pela variância amostral, somente é justificável para grandes amostras. Para pequenas amostras, torna-se necessário uma correção na distribuição padronizada, que consiste em substituir a distribuição Normal Padrão pela distribuição t-Student. Essa correção surge do fato de que, nesses casos, o valor da variância amostral pode ser totalmente diferente da variância populacional, não sendo adequado, portanto, o uso da aproximação Normal. Sob esse aspecto, para se fazer inferências sobre μ , deve-se usar a variável,

$$t = \frac{\bar{X} - \mu}{S/\sqrt{n}}$$

que tem distribuição t-Student com $(n-1)$ graus de liberdade. Quanto menor a amostra mais necessária se torna a introdução da correção: $t_{(n-1)}$ ao invés de Z . Portanto, o intervalo de confiança para a média populacional μ , ao nível de confiança $(1 - \alpha)$ é definido por

$$IC[\mu; 1 - \alpha] = \left[\bar{X} - t_{[(n-1); \alpha/2]} \frac{S}{\sqrt{n}}; \bar{X} + t_{[(n-1); \alpha/2]} \frac{S}{\sqrt{n}} \right]$$

Exemplo 4.1.1.2.2: Uma amostra de árvores castanheiras, todas com 8 anos de idade, foi observada em uma floresta. Os diâmetros (em polegadas) das árvores foram medidos à uma altura de 3 pés e os resultados foram registrados: 19,4; 21,4; 22,3; 22,1; 20,1; 23,8; 24,6; 19,9; 21,5; 19,1. Construa um intervalo de confiança de 95% para o verdadeiro diâmetro médio das árvores castanheiras dessa idade na floresta.

Resolução 01 (4.1.1.2.2). Função do R (Utilizando Intervalo de Confiança):

```
library(OneTwoSamples)
dadosC = c(19.4, 21.4, 22.3, 22.1, 20.1, 23.8, 24.6, 19.9, 21.5, 19.1)
interval_estimate1(dadosC, sigma = -1, alpha = 0.05)
  mean df      a      b
1 21.42  9 20.10233 22.73767
```

Resolução 02 (4.1.1.2.2). Função do R (Utilizando Teste de Hipótese):

```
library(TeachingDemos)
dados2 = t.test(dadosC, conf.level = 0.95)
dados2$conf.int
[1] 20.10233 22.73767
attr(,"conf.level")
[1] 0.95
```

Análise do IC 4.1.1.2.2: Ao nível de confiança de 95% o verdadeiro diâmetro médio das árvores de castanheiras é de no mínimo 20,10 polegadas e de no máximo 22,74 polegadas.

4.1.2 Intervalo de Confiança para a Proporção Populacional

Esse parâmetro pode ser usado para informar sobre: a proporção de pacientes tratados com um novo medicamento e que sofrem efeitos colaterais indesejáveis, a proporção de uma população que é imune a certa enfermidade, a proporção de itens defeituosos em um processo de produção, etc.. Considere uma variável aleatória, X , que representa a presença (ou não) de determinada característica de interesse de uma população. Isto é, X corresponde ao número de repetições independentes de um experimento com dois resultados possíveis: sucesso e fracasso, onde: $P(\text{Sucesso}) = p$ e $P(\text{Fracasso}) = (1 - p)$. Então X segue uma distribuição Binomial, $\text{Bin}(n; p)$, onde $E(X) = np$ e $\text{Var}(X) = np(1 - p)$. Portanto, o intervalo de confiança para a proporção populacional p , com nível de confiança $(1 - \alpha)$, é dado por

$$IC[p; 1 - \alpha] = \left[\hat{p} - Z_{\alpha/2} \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}}; \hat{p} + Z_{\alpha/2} \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}} \right],$$

em que $\hat{p} = x/n$ representa a proporção amostral, a qual é determinada a partir do número de sucessos na amostra (x) e do tamanho da mesma (n).

Exemplo 4.1.2: Para avaliar a taxa de desemprego em uma cidade, coletou-se uma amostra aleatória de 1000 habitantes em idade de trabalho e observou-se que 87 eram desempregados. Estimar a porcentagem de desempregados em toda cidade através de um intervalo de confiança de 95%.

Resolução 01 (4.1.2). Função do R (Utilizando Intervalo de Confiança):

```
library(OneTwoSamples)
x = 87
n = 1000
vetor <- c(rep(1,x), rep(0,n-x))    # Criar um vetor de 0s e 1s

interval_estimate1(vetor, sigma = sqrt(var(vetor)), alpha =
0.05)
  mean  df      a      b
1 0.087 1000 0.06952326 0.1044767
```

Resolução 02 (4.1.2). Função do R (Utilizando Teste de Hipótese):

```
library(TeachingDemos)
teste = z.test(vetor, stdev = sqrt(var(vetor)))
teste$conf.int
[1] 0.06952326 0.10447674
attr(,"conf.level")
[1] 0.95
```

Análise do IC 4.1.2: Ao nível de confiança de 95% a verdadeira proporção de desempregados, em idade de trabalho, em toda cidade é de no mínimo 6,95% e de no máximo 10,45%.

4.2 TESTES DE HIPÓTESES

É uma metodologia estatística que nos auxilia a tomar decisões sobre uma ou mais populações baseada na informação obtida da amostra. Os testes de hipóteses fornecem ferramentas que nos permitem rejeitar, ou não, uma hipótese estatística através da evidência fornecida pela amostra. Isto é, essa ferramenta permite ao pesquisador verificar se os dados amostrais trazem evidência que apoiem ou não uma hipótese estatística formulada.

Ao tentarmos tomar decisões, é conveniente a formulação de suposições ou de conjecturas sobre as populações de interesse, que, em geral, consistem em considerações sobre parâmetros (μ , σ^2 , p) das mesmas. Essas suposições, que podem ser ou não verdadeiras, são denominadas de Hipóteses Estatísticas, e tem como objetivo examinar duas hipóteses opostas sobre uma população: a hipótese nula e a hipótese alternativa. A hipótese nula é a declaração que está sendo testada. Normalmente, é uma declaração de “nenhum efeito” ou “nenhuma diferença”. A hipótese alternativa é a declaração que você quer ser capaz de concluir que é verdadeira com base em evidências fornecidas pelos dados da amostra. O teste será feito de tal forma que deverá sempre concluir na rejeição, ou não, da hipótese nula.

Em muitas situações práticas o interesse do pesquisador é verificar a veracidade sobre um ou mais parâmetros populacionais (**Testes de Hipóteses Paramétricos**), ou então, sobre a natureza da distribuição de uma variável aleatória (**Testes de Hipóteses Não-Paramétricos**). Por exemplo, pode-se estar interessado em verificar se: o salário médio de certa categoria profissional no Brasil é igual a R\$ 1.500,00; a produtividade média de milho no estado de Santa Catarina é de 2.500 kg/ha; 40% dos eleitores votarão em certo candidato nas próximas eleições; a proporção de peças defeituosas por unidade de fabricação é de 0,10; a propaganda produz efeito positivo nas vendas; os métodos de ensino produzem resultados diferentes de aprendizagem, ou ainda, se um medicamento é mais eficaz que outro.

Conceitos Fundamentais:

- **Hipótese Estatística:** Suposição quanto ao valor de um parâmetro ou quanto à natureza da distribuição de uma probabilidade de uma variável populacional.
- **Tipos de Hipóteses:**
 - a) Hipótese Nula (H_0): É aquela que será testada, sendo sempre contrária ao resultado do experimento.
 - b) Hipótese Alternativa (H_1): É qualquer hipótese diferente da hipótese nula, sendo sempre a favor do resultado do experimento.
- **Tipos de Testes:**
 - a) Teste Bilateral: A região crítica (sombreada) localiza-se nas duas extremidades da curva da distribuição amostral da estatística do teste.
 - b) Teste Unilateral à Esquerda: A região crítica do teste (sombreada) é localizada completamente na extremidade à esquerda da curva da distribuição amostral da estatística do teste.
 - c) Teste Unilateral à Direita: A região crítica do teste (sombreada) é localizada completamente na extremidade à direita da curva da distribuição amostral da estatística do teste.

Como estamos tomando uma decisão com base em informações de uma amostra, estaremos sujeitos a cometer dois tipos de erros.

- **Tipos de Erros:**
 - a) Erro Tipo I (α): P(rejeitar H_0 | H_0 é verdadeira)
 - b) Erro Tipo II (β): P(aceitar H_0 | H_0 é falsa)
- **Estatística de Teste:** A decisão de rejeitar ou não a hipótese nula é baseada nos dados amostrais, que são usados para calcular o valor da Estatística de Teste, e que servirá de referência para a tomada de decisão. Portanto, é a estatística utilizada para julgar H_0 .
- **Região Crítica do Teste (RC):** É formada pelo conjunto de valores que levam a rejeição de H_0 . Ela depende do tipo de hipótese alternativa, do nível de significância (α) adotado, e da distribuição de probabilidade da estatística do teste.

Etapas para a Elaboração de um Teste de Hipóteses:

1. Enunciar as hipóteses nula (H_0) e alternativa (H_1);
2. Fixar o nível de significância (α);
3. Determinar a estatística do teste;
4. Determinar a região crítica do teste;
5. Calcular o valor da estatística do teste (com base em uma amostra da população de interesse);
6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar (H_0). Caso contrário, não há motivos para rejeitar (H_0);
7. Concluir o teste (interpretação).

P-valor: Sob o ponto de vista estatístico, o p-valor, também chamado de **nível descritivo**, é a probabilidade de se obter uma estatística de teste igual ou mais extrema que aquela observada em uma amostra, sob H_0 . Por exemplo, em testes de hipóteses, pode-se rejeitar a hipótese nula a 5% caso o p-valor seja menor que 5%. É uma medida de quanta evidência você tem contra a hipótese nula, ou seja, quanto **menor** for o p-valor, maior é a evidência para rejeitar H_0 .

Outra interpretação para o p-valor, é que este é o menor nível de significância com que se rejeitaria a hipótese nula. Em termos gerais, um p-valor pequeno significa que a probabilidade de obter um valor da estatística de teste como o observado é muito improvável, levando assim à rejeição de H_0 . É preciso muita cautela na interpretação desse valor, dado que esta medida é bastante influenciada pelo tamanho da amostra. Amostras grandes tendem a produzir p-valores pequenos, enquanto amostras pequenas tendem a produzir p-valores grandes.

Vimos anteriormente que foram definidas as hipóteses nula (H_0) e alternativa (H_1). Em muitas aplicações da estatística, convencionou-se definir H_1 como a hipótese formulada pelo pesquisador, enquanto H_0 é o seu complemento. A princípio, a hipótese nula é considerada a verdadeira. Ao confrontarmos a hipótese nula com os achados de uma amostra aleatória extraída de uma população de interesse, verifica-se a sua plausibilidade em termos probabilísticos, o que nos leva a rejeitarmos ou não H_0 .

No entanto, por utilizarmos nesta tomada de decisão uma amostra (e não a população), podemos cometer erros, conforme mencionado. A probabilidade de ocorrer um Erro Tipo I é chamada de nível de significância (α), que é geralmente determinado pelo pesquisador antes da coleta dos dados. Em muitas aplicações da estatística, esse nível é tradicionalmente fixado em 5%.

Com base nestes conceitos, define-se o p-valor como a menor escolha que teríamos feito para o nível de significância, de forma a rejeitar H_0 . Por exemplo, suponha que foi fixado um $\alpha = 0,05$. Um p-valor igual a 0,20 indica que nós rejeitaríamos H_0 se tivéssemos escolhido um α igual a 0,20, ao menos. Como escolhemos $\alpha = 0,05$, não rejeitaríamos H_0 . Portanto, uma regra usual pode ser estabelecida, em que rejeita-se H_0 se o p-valor é menor que α , caso contrário, não há motivos para rejeitar H_0 .

A seguir vejamos uma interpretação razoável dos p-valores:

P-Valor (P)	Interpretação
$P < 0,01$	Evidência muito forte contra H_0
$0,01 \leq P < 0,05$	Evidência moderada contra H_0
$0,05 \leq P < 0,10$	Evidência sugestiva contra H_0
$P \geq 0,10$	Pouca ou nenhuma evidência real contra H_0

Tradicionalmente, o valor de corte para rejeitar H_0 é de 0,05, o que significa que, quando não há nenhuma diferença, um valor tão extremo para a estatística de teste é esperado em menos de 5% das vezes. Nos tópicos a seguir, alguns testes de hipóteses paramétricos são apresentados.

4.2.1 Teste de Hipótese para a Média Populacional

Considere uma população da qual retiramos uma amostra aleatória, X_1, X_2, \dots, X_n . Estamos interessados em realizar inferência sobre a média populacional μ . Nesse caso, duas situações serão consideradas: quando σ^2 for conhecida e, quando σ^2 for desconhecida.

4.2.1.1 Teste de Hipótese para a Média Populacional com a Variância Populacional Conhecida

1. Enunciar as hipóteses:

$$\begin{array}{lll} H_0: \mu = \mu_0 & & H_0: \mu = \mu_0 \\ H_1: \mu \neq \mu_0 & \text{ou} & H_1: \mu < \mu_0 \quad \text{ou} \quad H_1: \mu > \mu_0 \end{array}$$

2. Fixar o nível de significância (α);
3. Definir a estatística do teste. Admitindo σ^2 conhecida, sob a hipótese nula, a distribuição da estatística de teste terá distribuição $N(0,1)$;
4. Determinar a região crítica de teste;
5. Calcular o valor da estatística de teste:

$$Z_{cal} = \frac{\bar{X} - \mu_0}{\sigma / \sqrt{n}}$$

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.2.1.1: Suponha uma amostra aleatória com 1000 observações. Ao nível de significância de 5%, testar se a média populacional é igual a 7 supondo desvio padrão populacional conhecido igual a 2.

Resolução 4.2.1.1:

```
library(TeachingDemos)
set.seed(25062020) # Gerar a mesma amostra aleatória desse exemplo
amostra = rnorm(1000, mean = 5, sd = 2)
teste = z.test(amostra, mu = 7, stdev = 2, conf.level = 0.95)
teste

One Sample z-test

data: amostra
z = -30.231, n = 1.0000e+03, Std. Dev. = 2.0000e+00, Std.
```

```

Dev. of the
sample mean = 6.3246e-02, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 7
95 percent confidence interval:
 4.964043 5.211961
sample estimates:
mean of amostra
 5.088002

```

Análise do Teste 4.2.1.1:

1. Enunciar as hipóteses:

$$H_1: \mu \neq 7$$

$$H_0: \mu = 7;$$

2. Nível de significância: $\alpha = 5\%$;

3. Média amostral: $\bar{X} = 4,9260$ com Desvio Padrão da Média Amostral: $S = 0,0632$;

4. Estatística de Teste: $Z_{cal} = -32,793$;

5. P-valor $2,2e-16$ (menor que 1%);

6. Rejeita-se H_0 ao nível de significância de 5%, ou seja, há evidência suficiente para afirmar que a média populacional é diferente de 7.

4.2.1.2.1 Teste de Hipótese para a Média Populacional com Variância Populacional Desconhecida

No caso de **amostras grandes** pode-se substituir σ por S na estatística de teste. Nesse caso, a distribuição t-Student se aproxima da distribuição Normal. Portanto, o procedimento para a construção do teste é similar ao anterior.

Utilizaremos como exemplo os registros do banco de dados referente ao peso (em gramas) de uma amostra de docinhos "M&M", retirado do livro *Estatística Básica* (Autores: W. Bussab e P. Morettin. Disponível no Apêndice B: Conjunto 11, Página: 380).

Exemplo 4.2.1.2.1: Um pacote de confeitos da marca M&M indica no rótulo um conteúdo de 1498 confeitos com o peso total de 1361g, de modo que o peso médio de cada confeito é de 0,9085g (isto é, 1361/1498). Em um teste para determinar se o consumidor está sendo prejudicado, seleciona-se uma amostra aleatória de confeitos M&M. Execute o teste ao nível de significância de 5% e verifique a situação do consumidor. Os dados estão disponíveis em <https://github.com/anaherminia88/EstatisticaLivre/blob/master/13.%20mem.xls>.

Resolução 4.2.1.2.1:

```
library(stats)
mems = c(Vermelha, Laranja, Amarela, Marrom, Azul, Verde) #
Concatenar os dados
length(mems)
[1] 162

mems2 = na.omit(mems) # Retirar os NA's
length(mems2)
[1] 100

testemems2 = z.test(mems2, mu = 0.9085, sd(mems2),
alternative="less")
testemems2

One Sample z-test

data: mems2
z = -10.042, n = 1.0000e+02, Std. Dev. = 5.1794e-02, Std.
Dev. of the
sample mean = 5.1794e-03, p-value < 2.2e-16
alternative hypothesis: true mean is less than 0.9085
95 percent confidence interval:
 -Inf 0.8650094
sample estimates:
mean of mems2
 0.85649
```

Análise do Teste 4.2.1.2.1:

1. Enunciar as hipóteses:

$$H_0: \mu = 0,9085g$$

$$H_1: \mu < 0,9085g;$$

2. Nível de significância: $\alpha = 5\%$;
3. Média amostral: $\bar{X} = 0,85649$ com Desvio Padrão da Média Amostral: $S = 0,00518$;
Estatística de Teste: $Z_{cal} = -10,042$;
4. P-valor $2,2e-16$ (menor que 5%);
5. Rejeita-se H_0 ao nível de significância de 5%, ou seja, há evidência suficiente para desconfiar que o consumidor está sendo prejudicado visto que o verdadeiro peso médio dos confeitos é menor do que o indicado no rótulo, pelo fabricante;

No caso de termos **amostras pequenas**, uma correção se faz necessária e a distribuição t-Student passa a ser utilizada. O procedimento para a realização do teste é dado a seguir.

1. Enunciar as hipóteses:

$$H_0: \mu = \mu_0$$

$$H_0: \mu = \mu_0$$

$$H_0: \mu = \mu_0$$

$$H_1: \mu \neq \mu_0$$

ou

$$H_1: \mu < \mu_0$$

ou

$$H_1: \mu > \mu_0;$$

2. Fixar o nível de significância (α);
3. Definir a estatística de teste. Admitindo σ^2 desconhecida, a distribuição da estatística de teste sob a hipótese nula é: $t_{(n-1, \alpha/2)}$;
4. Determinar a região crítica do teste;
5. Calcular o valor da estatística de teste:

$$T_{cal} = \frac{\bar{X} - \mu_0}{S/\sqrt{n}}$$
6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.2.1.2.2: Os valores relacionados a seguir: 270, 273, 258, 204, 254, 228 e 282, são cargas axiais (em libras) de uma amostra de sete latas de alumínio de 12 oz. A carga axial de uma lata é o peso máximo que seus lados podem suportar, e deve ser superior a 165 libras, porque esta é a pressão máxima aplicada quando se fixa a tampa no lugar. Ao nível de significância de 0,01, teste a afirmação do engenheiro supervisor de que esta amostra provém de uma população com média superior a 165 libras.

Resolução 4.2.1.2.2:

```
library(stats)
latas = c(270, 273, 258, 204, 254, 228, 282)
testelatas = t.test(latas, mu = 165, alternative = "greater")
testelatas
```

One Sample t-test

```
data:  latas
t = 8.3984, df = 6, p-value = 7.761e-05
alternative hypothesis: true mean is greater than 165
95 percent confidence interval:
 232.4193      Inf
sample estimates:
mean of x
 252.7143
```

Análise do Teste 4.2.1.2.2:

1. Enunciar as hipóteses
 $H_0: \mu = 165$
 $H_1: \mu > 165;$
2. Nível de significância: $\alpha = 1\%$;
3. Média amostral: $\bar{X} = 252,7143$;
4. Estatística de Teste: $t_{cal} = 8,3984$ com $(n-1) = 6$ graus de liberdade;
5. P-valor = $7,761e-05 = 0,00007761$ (menor que 1%);
6. Rejeita-se H_0 ao nível de significância de 1%, ou seja, há evidência suficiente para apoiar a afirmação do supervisor de que a amostra provém de uma população com média superior às 165 libras desejadas.

4.4 TESTE DE HIPÓTESE PARA A PROPORÇÃO POPULACIONAL

1. Enunciar as hipóteses

$$\begin{array}{l} H_0: p = p_0 \\ H_1: p \neq p_0 \end{array} \quad \text{ou} \quad \begin{array}{l} H_0: p = p_0 \\ H_1: p < p_0 \end{array} \quad \text{ou} \quad \begin{array}{l} H_0: p = p_0 \\ H_1: p > p_0 \end{array}$$

2. Fixar o nível de significância (α);
3. Definir a estatística de teste, com distribuição normal padrão, sob a hipótese nula;
4. Determinar a região crítica do teste;
5. Calcular o valor da estatística de teste

$$Z_{cal} = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}}$$

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.4.1: Em um estudo sobre a eficácia do air-bag em automóveis, constatou-se que, em 821 colisões de carros de tamanho médio equipados com air-bag, 46 colisões resultaram em hospitalização do motorista. Ao nível de significância de 0,01, teste a afirmação de que a taxa de hospitalização nos casos de air-bag é inferior à taxa de 7,8% para colisões de carros de tamanho médio equipados com cintos automáticos de segurança.

Resolução 01 (4.4.1). Utilizando a aproximação pela Distribuição Normal:

```
library(TeachingDemos)
x = 46
n = 821
pest = x/n
vetor = c(rep(1,x), rep(0,n-x))      # Criar vetor de 0s e 1s
z.test(vetor, mu = 0.078, stdev = sqrt(var(vetor)), conf.
```

```
level = 0.99, alternative = "less")
```

```
One Sample z-test
```

```
data: vetor
```

```
z = -2.7357, n = 8.2100e+02, Std. Dev. = 2.3012e-01, Std.  
Dev. of the
```

```
sample mean = 8.0312e-03, p-value = 0.003113
```

```
alternative hypothesis: true mean is less than 0.078
```

```
99 percent confidence interval:
```

```
  -Inf 0.07471257
```

```
sample estimates:
```

```
mean of vetor
```

```
0.05602923
```

Análise do Teste 4.4.1 (Resolução 01):

1. Enunciar as hipóteses

$$H_0: p = 0,078$$

$$H_1: p < 0,078;$$

2. Nível de significância: $\alpha = 1\%$;
3. Proporção amostral: $\hat{p} = 0,0560 = 5,60\%$ (equivalente ao "pest");
4. Estatística de Teste: $Z_{cal} = -2,7357$;
5. P-valor = 0,003113 (menor que 1%);
6. Rejeita-se H_0 ao nível de significância de 1%, ou seja, há evidência suficiente para apoiar a afirmação de que, para colisões de carros de tamanho médio, a taxa de hospitalização, no caso de haver o air-bag, é inferior à taxa de 7,8% verificada no caso de carros com cintos de segurança automáticos.

Resolução 02 (4.4.1). Utilizando a aproximação pela Distribuição Qui-Quadrado:

Alternativamente, a proporção populacional pode ser testada pela função **prop.test**, em que, sob a hipótese nula, a estatística de teste segue uma distribuição Qui-Quadrado, χ^2 .

```
library(stats)
x = 46
n = 821
prop.test(x, n, p = 0.078, alternative = "less")
  1-sample proportions test with continuity correction

data:  x out of n, null probability 0.078
X-squared = 5.2094, df = 1, p-value = 0.01123
alternative hypothesis: true p is less than 0.078
95 percent confidence interval:
 0.00000000 0.07142186
sample estimates:
      p
0.05602923
```

Análise do Teste 4.4.1 (Resolução 02):

Perceba que, assim como na resolução anterior, a conclusão desse teste para verificar a eficácia do air-bag é similar a análise realizada por meio dos resultados obtidos da **Resolução 01**, visto que o registro do **P-valor = 0,01123** (encontrado na **Resolução 02**) confirma a rejeição da hipótese nula.

4.5 TESTE DE HIPÓTESE PARA A DIFERENÇA ENTRE DUAS MÉDIAS POPULACIONAIS (DUAS POPULAÇÕES INDEPENDENTES)

Supondo que as variâncias das duas populações são desconhecidas, um teste para comparar as médias de duas populações com amostras independentes pode ser realizado. Na prática, há diversas situações em que desejamos comparar, em termos de média ou proporção, duas populações diferentes.

Exemplos:

- a) Meninos e meninas, gastam o mesmo número de horas semanais navegando na internet?
- b) O desempenho de alunos em um exame nacional melhora depois que eles realizam um curso preparatório?
- c) Irmãos gêmeos respondem da mesma forma a um determinado estímulo?
- d) A proporção de pessoas favoráveis a um projeto estadual é a mesma na zona urbana e na zona rural?

Perceba que nos exemplos, deseja-se comparar duas populações: meninos e meninas, alunos antes e depois do curso preparatório, irmãos gêmeos, zona urbana e zona rural. Nesse contexto, é possível inferir acerca dos parâmetros associados as duas populações, baseado nas amostras aleatórias extraídas das populações de interesse. Há duas situações diferentes:

Situação 01. Quando comparamos meninos e meninas do ensino médio ou zonas urbana e rural.

Situação 02. Quando comparamos irmãos gêmeos ou alunos antes e depois de um curso preparatório.

No primeiro caso, há uma independência entre as amostras, o que não ocorre no segundo caso. Portanto, define-se as **amostras independentes** quando o processo de seleção dos indivíduos na amostra 1 não tem qualquer efeito, ou qualquer relação, com a seleção dos indivíduos na amostra 2. As **amostras dependentes**,

também denominadas de pareadas (ou emparelhadas), são aquelas nas quais os experimentos que envolvem medidas de cada indivíduo antes e depois de algum evento resultam em dados emparelhados. Nesse caso, cada observação **antes** está associada, ou emparelhada, com uma observação **depois**. As variáveis são medidas no mesmo sujeito.

Aplicações de testes de hipóteses no caso de duas populações independentes, assim como, nas situações em que as populações são dependentes (pareadas), serão apresentadas a seguir. Vale ressaltar que serão consideradas apenas as situações em que as variâncias populacionais são desconhecidas, visto que, na prática, a maioria dos estudos se enquadra nessa característica.

4.5.1. Teste de Hipótese para a Diferença entre Duas Médias Populacionais com Variâncias Populacionais Iguais ($\sigma_1^2 = \sigma_2^2$) e Desconhecidas

1. Enunciar as hipóteses nula (H_0) e alternativa (H_1);

$$2. \quad \begin{array}{l} H_0: \mu_1 = \mu_2 \\ H_1: \mu_1 \neq \mu_2 \end{array} \quad \text{ou} \quad \begin{array}{l} H_0: \mu_1 = \mu_2 \\ H_1: \mu_1 < \mu_2 \end{array} \quad \text{ou} \quad \begin{array}{l} H_0: \mu_1 = \mu_2 \\ H_1: \mu_1 > \mu_2 \end{array}$$

3. Fixar o nível de significância (α);
4. Determinar a estatística de teste que tem, sob a hipótese nula, uma distribuição t-student com $(n_1 + n_2 - 2)$ graus de liberdade;
5. Determinar a região crítica do teste;
6. Calcular o valor da estatística de teste (com base em uma amostra da população de interesse);

em que

$$T_{cal} = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{S_c^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}$$

$$S_c^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{(n_1 + n_2 - 2)},$$

\bar{X}_1 é a média da amostra 1, \bar{X}_2 é a média da amostra 2, S_1^2 é a variância da amostra 1, S_2^2 é a variância da amostra 2, n_1 é o tamanho da amostra 1, e n_2 é o tamanho da amostra 2;

7. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
8. Conclusão do teste (interpretação).

Exemplo 4.5.1: Dois tipos diferentes de tecido devem ser comparados. Uma máquina de testes Martindale pode comparar duas amostras ao mesmo tempo. O peso (em miligramas) para sete experimentos foram registrados a seguir:

Tecidos	1	2	3	4	5	6	7
Tecido A	36	26	31	38	28	20	37
Tecido B	39	27	35	42	31	39	22

Teste se um tecido é mais pesado que o outro a um nível de significância de 5%. Admita que a variância populacional é a mesma para os dois tipos de tecido.

Resolução 4.5.1:

```
library(stats)
tecidoA = c(36, 26, 31, 38, 28, 20, 37)
tecidoB = c(39, 27, 35, 42, 31, 39, 22)
testetecido = t.test(tecidoA, tecidoB, var.equal = TRUE)
```

Two Sample t-test

```
data: tecidoA and tecidoB
t = -0.73005, df = 12, p-value = 0.4794
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -10.814996  5.386425
sample estimates:
mean of x mean of y
 30.85714  33.57143
```

Análise do Teste 4.5.1:

1. Enunciar as hipóteses

$$H_0: \mu_A = \mu_B, \text{ ou equivalentemente, } H_0: \mu_A - \mu_B = 0$$

$$H_1: \mu_A \neq \mu_B, \text{ ou equivalentemente, } H_1: \mu_A - \mu_B \neq 0;$$

2. Nível de significância: $\alpha = 5\%$;
3. Médias Amostrais: $\bar{X}_1 = 30,86$ e $\bar{X}_2 = 33,57$;
4. Estatística de Teste: $t_{cal} = -0,73005$ com $(n-2) = 12$ graus de liberdade;
5. P-valor = 0,4794 (maior que 5%);
6. Não há motivos para rejeitar H_0 ao nível de significância de 5%, ou seja, há evidência suficiente para afirmar que os dois tipos de tecido tenham o mesmo peso médio populacional.

4.5.2 Teste de Hipótese para Diferença entre Duas Médias Populacionais com Variâncias Populacionais Diferentes ($\sigma_1^2 \neq \sigma_2^2$) e Desconhecidas

1. Enunciar as hipóteses nula (H_0) e alternativa (H_1):
 - a) Teste Bilateral: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 \neq \mu_2$;
 - b) Teste Unilateral à Esquerda: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 < \mu_2$;
 - c) Teste Unilateral à Direita: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 > \mu_2$;
2. Fixar o nível de significância (α);
3. Determinar a estatística de teste que, sob a hipótese nula, tem distribuição t-student com g graus de liberdade dado pela parte inteira de

$$g = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1-1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2-1}}$$

4. Determinar a região crítica do teste;

5. Calcular o valor da estatística de teste (com base em uma amostra da população de interesse).

$$T_{cal} = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}}$$

em que n_1 é o tamanho da amostra 1, n_2 é o tamanho da amostra 2, \bar{X}_1 é a média da amostra 1, \bar{X}_2 é a média da amostra 2, S_1^2 é a variância da amostra 1 e S_2^2 é a variância da amostra 2;

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.5.2.1: Duas amostras de 10 alunos, de duas turmas distintas, de um mesmo curso apresentam os seguintes totais de pontos em provas de certa disciplina. Ao nível de significância de 5% testar as hipóteses de que as turmas tenham aproveitamento médio diferentes.

Turma 1	51	47	75	35	72	84	45	11	52	57
Turma 2	27	75	49	69	73	63	79	37	84	32

Resolução 4.5.2.1:

```
library(stats)
turma1 = c(51, 47, 75, 35, 72, 84, 45, 11, 52, 57)
turma2 = c(27, 75, 49, 69, 73, 63, 79, 37, 84, 32)
t.test(turma1, turma2)

Welch Two Sample t-test

data:  turma1 and turma2
t = -0.62797, df = 17.998, p-value = 0.5379
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -25.63902  13.83902
sample estimates:
mean of x mean of y
 52.9      58.8
```

Análise do Teste 4.5.2.1:

1. Enunciar as hipóteses

$$H_0: \mu_1 = \mu_2 \text{ ou equivalentemente, } H_0: \mu_1 - \mu_2 = 0$$

$$H_1: \mu_1 \neq \mu_2 \text{ ou equivalentemente, } H_1: \mu_1 - \mu_2 \neq 0;$$

2. Nível de significância: $\alpha = 5\%$;
3. Médias Amostrais: $\bar{X}_1 = 52,9$ e $\bar{X}_2 = 58,8$;
4. Estatística de Teste: $t_{cal} = -0,62797$ com $(n-2) = 17,998$ graus de liberdade;
5. P-valor = $0,5379$ (maior que 5%);
6. Não há motivos para rejeitar H_0 ao nível de significância de 5% , ou seja, há evidência suficiente para afirmar que as duas turmas apresentam verdadeiro aproveitamento médio similar.

Observação: Perceba que, no caso de haver a suposição de que as **variâncias populacionais são IGUAIS, porém desconhecidas** (teste apresentado anteriormente), basta adicionar a essa resolução o argumento **var.equal = TRUE**.

4.5.3 Teste de Hipótese para Diferença entre Duas Proporções Populacionais

Um teste para comparar as proporções de duas populações pode ser realizado da seguinte forma:

1. Enunciar as hipóteses nula (H_0) e alternativa (H_1);
 - a) Teste Bilateral: $H_0: p_1 = p_2$ versus $H_1: p_1 \neq p_2$
 - b) Teste Unilateral à Esquerda: $H_0: p_1 = p_2$ versus $H_1: p_1 < p_2$.
 - c) Teste Unilateral à Direita: $H_0: p_1 = p_2$ versus $H_1: p_1 > p_2$.
2. Fixar o nível de significância (α);
3. Determinar a estatística de teste que, sob a hipótese nula, tem distribuição Normal Padrão;
4. Determinar a região crítica do teste;
5. Calcular o valor da estatística de teste

$$Z_{cal} = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}\hat{q}\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

em que n_1 é o tamanho da amostra 1, n_2 é o tamanho da amostra 2, \hat{p}_1 é a proporção de ocorrências da característica de interesse da amostra 1, \hat{p}_2 é a proporção de ocorrências da característica de interesse da amostra 2, \hat{p} é a proporção de ocorrências da característica de interesse nas duas amostras combinadas, ou seja,

$$\hat{p} = \frac{x_1 + x_2}{n_1 + n_2},$$

em que x_1 e x_2 são o número de ocorrências da característica de interesse nas amostras 1 e 2, respectivamente. Além disso, $\hat{q} = 1 - \hat{p}$;

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.5.3.1: De 400 moradores sorteados de uma grande cidade industrial, 300 são favoráveis a um projeto governamental. Já na cidade vizinha, de 130 moradores, cuja principal atividade é o turismo, 120 são contra o projeto governamental. Considerando um nível de significância de 5% você diria que as opiniões dos moradores, em termos de proporção, nas duas cidades em relação ao projeto governamental é a mesma?

Resolução 4.5.3.1:

```
library(stats)
testprojeto = prop.test(x = c(300, 120), n = c(400,130))
  2-sample test for equality of proportions with continuity
correction

data:  c(300, 120) out of c(400, 130)
```

```
X-squared = 16.833, df = 1, p-value = 4.082e-05
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.2406141 -0.1055397
sample estimates:
  prop 1    prop 2
0.7500000 0.9230769
```

Análise do Teste 4.5.3.1:

1. Enunciar as hipóteses

$$H_0: p_1 = p_2, \text{ ou equivalentemente, } H_0: p_1 - p_2 = 0$$

$$H_1: p_1 \neq p_2, \text{ ou equivalentemente, } H_1: p_1 - p_2 \neq 0;$$

2. Nível de significância: $\alpha = 1\%$;
3. Proporções Amostrais: $\hat{p}_1 = 0,75$ e $\hat{p}_2 = 0,92$;
4. Estatística de Teste: $\chi^2_{cal} = 16,833$ com $(n-1) = 1$ grau de liberdade;
5. P-valor = 0,00004082;
6. Rejeita-se H_0 ao nível de significância de 5%, ou seja, há evidências de que as opiniões dos moradores, para a verdadeira proporção, nas duas cidades em relação ao projeto governamental são diferentes.

4.5.4 Teste de Hipótese para a Diferença entre Duas Médias Populacionais (Populações Dependentes ou Pareadas)

Um teste para comparar as médias de duas populações com amostras emparelhadas pode ser realizado da seguinte forma:

1. Enunciar as hipóteses nula (H_0) e alternativa (H_1);
 - a) Teste Bilateral: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 \neq \mu_2$.
 - b) Teste Unilateral à Esquerda: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 < \mu_2$.
 - c) Teste Unilateral à Direita: $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 > \mu_2$.

2. Fixar o nível de significância (α);
3. Determinar a estatística de teste que, sob a hipótese nula, tem distribuição t-student com $(n-1)$ graus de liberdade;
4. Determinar a região crítica do teste;
5. Calcular o valor da estatística de teste:

Para cada par $(X_{1i}; X_{2i})$ calcule a diferença, $d_i = X_{1i} - X_{2i}$. Então, calcule a diferença média e a variância das diferenças, dadas, respectivamente, por:

$$\bar{d} = \frac{\sum_{i=1}^n d_i}{n} \text{ e } S_d^2 = \frac{\sum_{i=1}^n (d_i - \bar{d})^2}{n - 1};$$

A estatística de teste, que sob a hipótese nula tem distribuição t-student com $n - 1$ graus de liberdade, é definida, por:

$$T_{cal} = \frac{\bar{d}}{S_d / \sqrt{n}};$$

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Exemplo 4.5.4.1: Um consultor que trabalha para o quartel da Polícia Estadual afirma que as armas de serviço dispararão com uma velocidade de boca maior se o cano estiver adequadamente limpo. Obteve-se uma amostra aleatória de armas de 9 mm, e mediu-se a velocidade de boca (em pés por segundo) de um único tiro de cada arma. Cada arma foi profissionalmente limpa e a velocidade de boca de um segundo tiro (com o mesmo tipo de bala) foi medida. Os dados são apresentados na tabela que segue. Verifique se há alguma evidência de que uma arma limpa dispara com velocidade média de boca maior ao nível de significância de 1%.

Disparo da Arma

Antes da limpeza 1505 1419 1504 1494 1510 1506
 Depois da limpeza 1625 1511 1459 1441 1472 1521

Resolução 4.5.4.1:

```
library(stats)
antes = c(1505, 1419, 1504, 1494, 1510, 1506)
depois = c(1625, 1511, 1459, 1441, 1472, 1521)
t.test(antes, depois, paired = T, alternative = "less")

Paired t-test

data: antes and depois
t = -0.49656, df = 5, p-value = 0.3203
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
 -Inf 46.3796
sample estimates:
mean of the differences
 -15.16667
```

Análise do Teste 4.5.4.1:

1. Enunciar as hipóteses

$$H_0: \mu_D = 0, \text{ ou equivalentemente, } H_0: \mu_1 - \mu_2 = 0$$

$$H_1: \mu_D < 0, \text{ ou equivalentemente, } H_1: \mu_1 - \mu_2 < 0;$$

2. Nível de significância: $\alpha = 1\%$;
3. Média das Diferenças Amostrais: $\bar{X}_1 - \bar{X}_2 = -15,16667$;
4. Estatística de Teste: $t_{cal} = -0,49656$ com $(n-1) = 5$ graus de liberdade;
5. P-valor = 0,3203 (maior que 1%);
6. Não há motivos para rejeitar H_0 ao nível de significância de 1%, ou seja, não há evidências de que uma arma limpa dispara com velocidade média de boca maior.

4.6 Teste Qui-Quadrado de Independência

É um teste de hipótese não-paramétrico cujo objetivo é verificar se existe independência entre duas variáveis medidas nas mesmas unidades experimentais. Em outras palavras, o teste qui-quadrado busca evidência estatística de que duas variáveis possuem certo grau de associação.

1. Enunciar as hipóteses

H_0 : As variáveis são independentes (não estão associadas)

H_1 : As variáveis não são independentes (estão associadas);

2. Fixar o nível de significância (α);
3. Determinar a estatística de teste: χ^2 com $(r-1)(c-1)$ graus de liberdade, em que r representa o número de linhas e c corresponde ao número de colunas na tabela de contingência;
3. Determinar a região crítica do teste;
4. Calcular o valor da estatística de teste

$$\chi^2 = \sum_{i=1}^r \sum_{j=1}^c \frac{(O_{ij} - E_{ij})^2}{E_{ij}};$$

5. Supondo-se que as variáveis sejam independentes, o valor esperado de cada célula será

$$E_{ij} = \frac{(\text{totalna linha } i)(\text{totalna coluna } j)}{\text{totalna amostra}};$$

6. Se o valor calculado no passo 5 pertencer a RC deve-se rejeitar H_0 . Caso contrário, não há motivos para rejeitar H_0 ;
7. Conclusão do teste (interpretação).

Critérios para validação do Teste Qui-Quadrado:

1. Exclusivamente para variáveis nominais e ordinais.
2. Seleção aleatória dos dados.
3. Observações independentes.
4. Todas as frequências esperadas devem ser maiores ou igual a 1.

5. Não mais de 20% das frequências esperadas devem ser inferiores a 5.

Exemplo 4.6.1: A seguir são apresentados os registros de uma pesquisa realizada com eleitores quanto a classificação por gênero e a sua identificação partidária. Os indivíduos se identificaram mais fortemente com o partido Democrata ou com o partido Republicano ou com o Independente. Ao nível de significância de 1%, verifique se a preferência partidária está associada ao gênero dos indivíduos.

Gênero	Democrata	Independente	Republicano
Masculino	484	239	477
Feminino	762	327	468
Total	1246	566	945

Resolução 4.6.1:

```
library(stats)
dadosTC = as.table(rbind(c(484, 239, 477), c(762, 327, 468)))
dadosTC
  A  B  C
A 484 239 477
B 762 327 468

dimnames(dadosTC) = list(Genero = c("Masculino", "Feminino"),
  Partido = c("Democrata", "Independente", "Republicano"))
dimnames(dadosTC)
$Genero
[1] "Masculino" "Feminino"

$Partido
[1] "Democrata" "Independente" "Republicano"

Xsq = chisq.test(dadosTC)
Xsq

  Pearson's Chi-squared test

data: dadosTC
X-squared = 30.07, df = 2, p-value = 2.954e-07

Xsq$observed
  Partido
Genero Democrata Independente Republicano
Masculino 484 239 477
Feminino 762 327 468

Xsq$expected
  Partido
Genero Democrata Independente Republicano
```

Masculino	542.3286	246.3547	411.3166
Feminino	703.6714	319.6453	533.6834

Análise do Teste 4.6.1:

1. Enunciar as hipóteses

H_0 : A preferência pelo partido independe do gênero dos eleitores

H_1 : A preferência pelo partido não independe do gênero dos eleitores;

2. Nível de significância: $\alpha = 1\%$;
3. Graus de Liberdade: $(r-1)(c-1) = (2-1)(3-1) = 2$;
4. Estatística de Teste: $\chi^2 = 30,07$;
5. P-valor = $2,954e-07$ (menor que 1%);
6. Rejeita-se H_0 ao nível de significância de 1%, ou seja, há evidência suficiente para afirmar que a preferência pelo partido está associada ao gênero dos eleitores (as variáveis em estudo não são independentes).

5 CORRELAÇÃO E REGRESSÃO

Quando existe o interesse em analisar a relação linear entre duas variáveis quantitativas (X e Y , por exemplo), duas técnicas podem ser consideradas:

Correlação: Quantifica a força da relação linear com uma medida que resume o grau de relacionamento entre duas variáveis.

Regressão: Modela a forma dessa relação, tendo como resultado uma equação matemática que descreve o relacionamento entre variáveis.

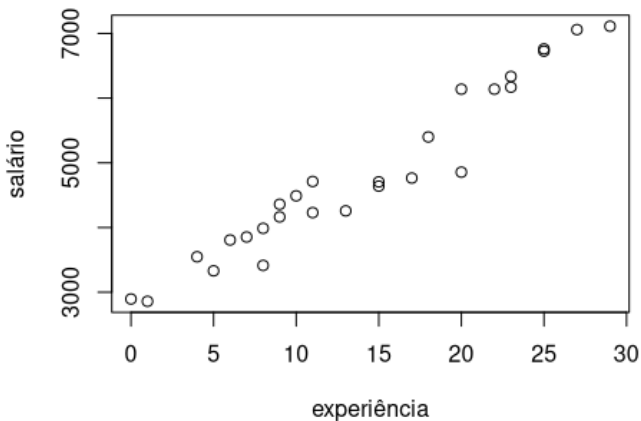
É importante ressaltar que o conceito de correlação se refere a uma associação numérica (relação estatística) entre duas variáveis, não implicando, necessariamente, relação de causa-e-efeito. A ideia de relação causal deve vir de uma análise teórica na área do estudo. Além disso, uma correlação elevada entre variáveis nem sempre indica que faz sentido essa relação, como por exemplo, as variáveis produção de bananas *versus* taxa de natalidade podem apresentar correlação, porém isso não significa que uma influencia no comportamento da outra. Vejamos a seguir alguns exemplos de pares variáveis em que podemos utilizar estas técnicas.

Variável X	Variável Y
Temperatura do forno(°C)	Resistência mecânica da cerâmica (Mpa)
Quantidade de aditivo (%)	Octanagem da gasolina
Renda (R\$)	Consumo (R\$)
Memória RAM do computador (Gb)	Tempo de resposta do sistema (s)
Área construída do imóvel (m ²)	Preço do imóvel (R\$)

5.2 ANÁLISE GRÁFICA

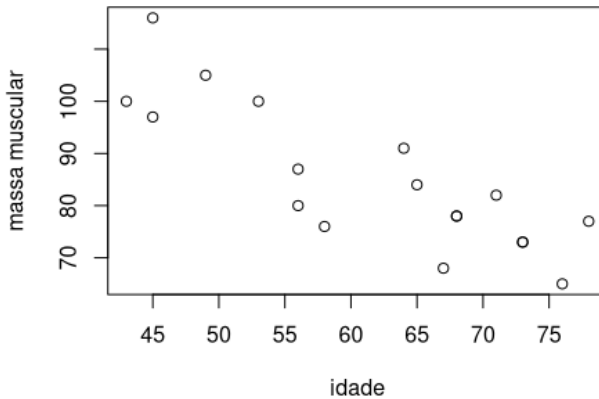
A análise gráfica nos possibilita observar a possível existência e tipo da relação estatística entre duas variáveis. O gráfico utilizado para analisar o comportamento conjunto de duas variáveis quantitativas é chamado *gráfico de dispersão*. A seguir, iremos relembra, com alguns exemplos, este tipo de gráfico, visto no Capítulo Estatística Descritiva. Considere o banco de dados hipotético acerca do salário e anos de experiência de 27 indivíduos, disponível em https://github.com/XXX/EstatisticaLivre/blob/master/dados_salario.txt. Vamos construir um gráfico de dispersão entre as variáveis *salário* (*salario*) e *anos de experiência* (*exp*).

```
attach(dados_salario)
plot(exp, salario, xlab = "experiência", ylab = "salário")
```



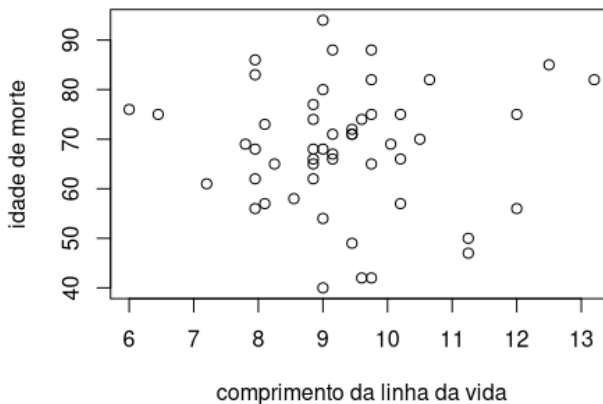
É possível observar uma relação crescente e linear entre as variáveis *salário* e *anos de experiência*. Considere agora o banco de dados hipotético acerca da massa muscular e idade de 18 indivíduos, disponível em https://github.com/anaherminia88/EstatisticaLivre/blob/master/dados_massa_muscular.txt. Vamos construir um gráfico de dispersão entre as variáveis *idade* e *massa muscular* (*mm*).

```
attach(dados_massa_muscular)
plot(idade, mm, xlab = "idade", ylab = "massa muscular")
```



Neste caso, observamos uma relação linear decrescente entre as variáveis *massa muscular* e *idade*. Considere por fim o banco de dados hipotético acerca da idade da morte e comprimento da linha da vida de 51 indivíduos, disponível em https://github.com/anaherminia88/EstatisticaLivre/blob/master/dados_linha_da_vida.txt. Vamos construir um gráfico de dispersão entre as variáveis *idade da morte* (*im*) e *comprimento da linha da vida* (*comp*).

```
attach(dados_linha_da_vida)
plot(comp, im, xlab = "comprimento da linha da vida",
      ylab = "idade de morte")
```



Já aqui, é possível notar que não há relação entre as variáveis *idade de morte* e *comprimento da linha da vida*.

5.3 COEFICIENTE DE CORRELAÇÃO LINEAR DE PEARSON

O *Coefficiente de Correlação Linear de Pearson* é uma medida do grau e do sinal da correlação linear entre duas variáveis (X, Y) , sendo expresso por

$$r = \frac{Cov(X, Y)}{S_X S_Y},$$

em que S_X e S_Y representam o desvio padrão amostral das variáveis X e Y , respectivamente, e $Cov(X, Y)$ é a covariância entre elas, definida por

$$Cov(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}.$$

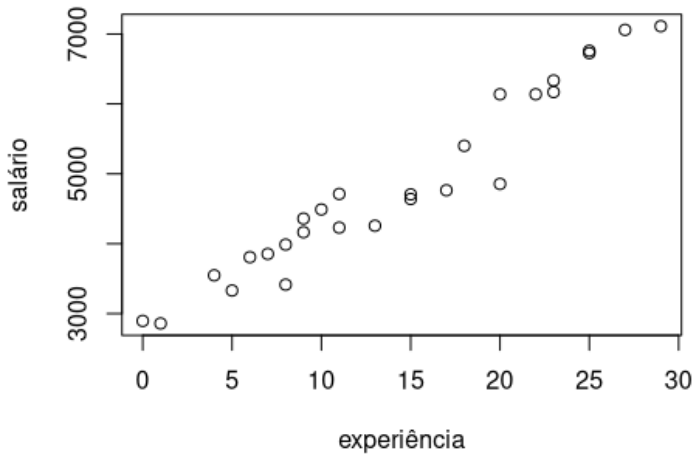
Este coeficiente é adimensional, logo não é afetado pelas unidades de medidas das variáveis X e Y . O sinal *positivo* deste coeficiente indica que a relação entre as variáveis é diretamente proporcional, enquanto que o sinal *negativo* indica relação inversamente proporcional. Temos que

$$-1 \leq r \leq 1$$

- Se $r = -1$, dizemos que a correlação é **perfeita negativa**.
- Se $r = 0$, dizemos que a correlação é **nula**.
- Se $r = 1$, dizemos que a correlação é **perfeita positiva**.
- Se $0 < r < 1$, dizemos que a correlação é **positiva**.
- Se $-1 < r < 0$, dizemos que a correlação é **negativa**.

Retomando o nosso primeiro exemplo gráfico, já observamos que havia uma relação crescente e linear entre as variáveis *salário* e *anos de experiência*.

```
attach(dados_salario)
plot(exp, salario, xlab = "experiência", ylab = "salário")
```



No RStudio, para medir o grau desta relação, vamos calcular o coeficiente de correlação linear entre as variáveis *salário* e *anos de experiência* utilizando o comando *cor*.

```
cor(salario, exp)
```

```
## [1] 0.9704583
```

Obtivemos $r = 0,97$, ou seja, existe uma forte correlação linear entre as variáveis *salário* e *anos de experiência* na **amostra**. Para estendermos este resultado para a **população** de onde essa amostra foi retirada, é preciso aplicar um teste de hipóteses. As etapas do teste de hipóteses para o coeficiente de correlação linear de Pearson estão descritas a seguir.

- Definição das hipóteses.

$H_0: \rho = 0$ (não existe correlação linear)

$H_1: \rho \neq 0$ (existe correlação linear);

- Fixar o nível de significância α ;
- Definir a estatística do teste e sua distribuição sob a hipótese nula.

$$T = r \sqrt{\frac{n-2}{1-r^2}} \sim t_{(n-2)};$$

- Definir a região crítica do teste (RC);

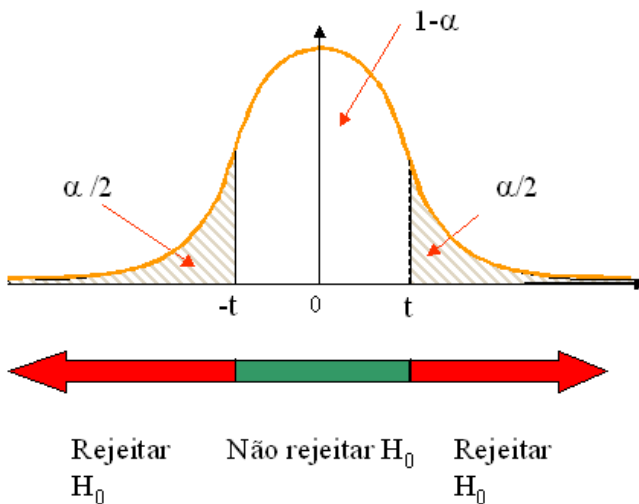


Figura 6.3.1: Região crítica para teste de hipóteses bilateral.

- Calcular a estatística de teste T_c ;
- Se T_c pertence a RC \Rightarrow rejeitar H_0 . Se T_c não pertence a RC \Rightarrow não rejeitar H_0 ;
- Concluir sobre a decisão tomada na etapa anterior.

No RStudio podemos realizar este teste de hipóteses para saber se há correlação linear entre as variáveis *salário* e *anos de experiência* utilizando o comando `cor.test`.

```
cor.test(salario,exp, method = "pearson",
alternative="two.sided")
```

```
Pearson's product-moment correlation

data:  salario and exp
t = 20.112, df = 25, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.9354131 0.9866192
sample estimates:
      cor
0.9704583
```

Observe que $p\text{-valor} < 2,2 * 10^{-16}$, ou seja, rejeita-se a hipótese nula para um nível de significância baixo, como por exemplo $\alpha = 1\%$. Concluímos então que com base na amostra e com confiança de 99% existe relação linear positiva forte entre as variáveis *salário* e *anos de experiência* da qual essa amostra foi retirada.

5.4 REGRESSÃO LINEAR SIMPLES

Iniciaremos o estudo de regressão com a formulação mais simples, relacionando uma *variável Y*, chamada de *variável resposta* ou *dependente*, com uma *variável X*, denominada de *variável explicativa* ou *independente*. O modelo que busca explicar uma variável *Y* como uma função linear de apenas uma variável *X* é denominado de modelo de regressão linear simples. A aplicação da regressão é geralmente feita sob um referencial teórico, que justifique uma relação matemática de causalidade.

Como um exemplo em que a análise de regressão pode ser útil, suponha que um engenheiro está estudando sobre o sistema de abastecimento de máquinas de venda automática de refrigerantes. Ele está interessado em desenvolver um método para prever o tempo necessário para o funcionário abastecer e fazer a manutenção de rotina das máquinas como uma função do número de refrigerantes que serão estocados. Nossas variáveis são

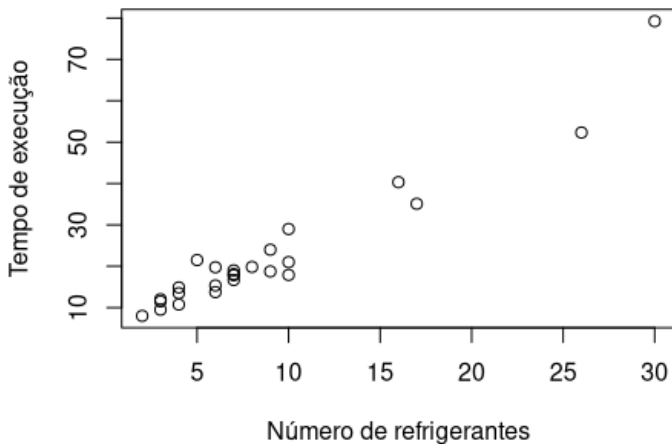
Y : Tempo de execução (minutos) → Variável resposta.

X : Número de refrigerantes → Variável explicativa.

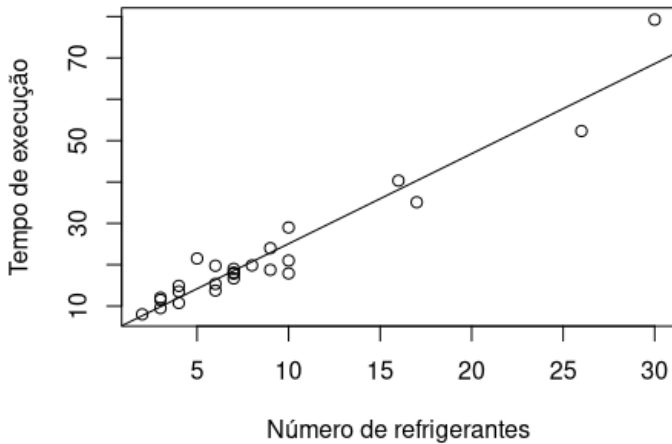
Esses dados estão disponíveis no pacote *GLMsData*. Para acessá-los basta utilizar o comando `data(softdrink)` após carregar o pacote.

```
library(GLMsData)
data(sdrink)
attach(sdrink)
```

```
y = Time
x = Cases
plot(y ~ x, xlab= "Número de refrigerantes", ylab="Tempo de
execução")
```



O gráfico de dispersão sugere claramente uma relação crescente entre o tempo de execução e o número de refrigerantes estocados. Como o objetivo é modelar o relacionamento das variáveis por meio de uma equação matemática, uma forma simples e razoável é por meio da relação linear.



Analisando o gráfico com a ilustração desse relacionamento em linha reta, observamos que os pontos dos dados não caem exatamente em cima da linha reta. Mas, como Y é uma variável aleatória, para o conceito de regressão, é incluído no modelo linear um *componente aleatório*, chamado de erro.

A equação que relaciona a variável resposta Y com a variável independente X pode ser escrita da seguinte maneira

$$Y_i = \alpha + \beta x_i + \epsilon_i,$$

em que

- Y_i é a variável aleatória associada à i -ésima observação de Y .
- x_i é a i -ésima observação do valor fixado para a variável independente (e não aleatória) X .
- ϵ_i é o *erro aleatório* da i -ésima observação, isto é, o efeito de uma infinidade de fatores que estão afetando a observação de Y de forma aleatória.
- α e β são parâmetros que precisam ser estimados.

É importante perceber que, na *análise de regressão*, que o regressor X é uma variável controlada (fixa) e que para cada possível valor de X existe uma distribuição de probabilidade para a variável resposta Y . Além disso, devemos pensar em ϵ como

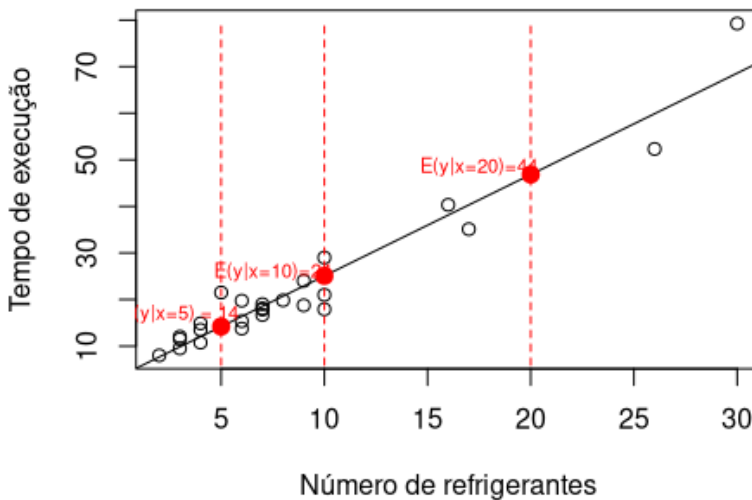
um erro estatístico, ou seja, uma variável aleatória que explica a falha do modelo no ajuste dos dados. Adicionalmente, fazemos a suposição que

$$E(\epsilon) = 0 \quad e \quad Var(\epsilon) = \sigma^2.$$

Assim, interpretamos a reta como sendo a linha de valores médios (ou esperados) da variável resposta y para um dado valor de x . Além disso, para cada valor de x , a variabilidade de y se mantém constante e é σ^2 , a variância do erro.

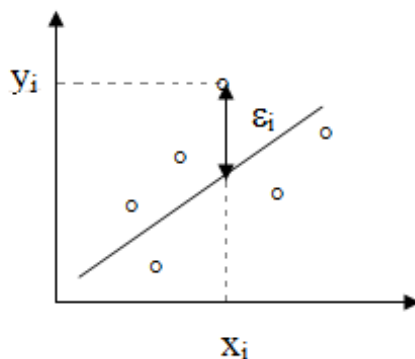
Os parâmetros α e β são geralmente chamados de coeficientes de regressão. Eles têm uma interpretação simples e bastante útil. O intercepto α é a média ($\mu_{Y|x}$) quando $x = 0$. Se o intervalo de x não inclui zero, então α não possui interpretação prática. A inclinação β é a alteração na média da distribuição de y produzida por uma mudança de unidade em x .

Interpretando o modelo do exemplo, a reta representa os valores médios do tempo de abastecimento e manutenção das máquinas (em minutos) para cada número específico de refrigerantes estocados.



Seja a reta $E(Y|x) = 4 + 2X$, então 4 é o intercepto e 2 é a inclinação. Temos que, o tempo médio para manutenção da máquina quando não há a reposição de refrigerante é de 4 minutos e, para cada unidade de refrigerante estocado, haverá um aumento de 2 minutos neste tempo médio.

Como devemos então escolher a reta que **melhor** se ajusta aos dados? Queremos encontrar a reta que passe o mais próximo possível dos pontos observados. Uma ideia inicial seria que nosso modelo envolve erros, então podemos tentar minimizá-los.



O **método de mínimos quadrados** é usado para estimar os parâmetros do modelo (α e β) e consiste em fazer com que a soma dos erros quadráticos seja menor possível, ou seja, este método consiste em obter os valores de α e β que minimizam a expressão

$$S = \sum_{i=1}^n \epsilon_i^2 = \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2.$$

Aplicando-se derivadas parciais à expressão anterior, e igualando-se a zero, acharemos as seguintes estimativas para α e β , as quais chamaremos de a e b , respectivamente

$$b = \frac{n \sum_{i=1}^n x_i Y_i - (\sum_{i=1}^n x_i)(\sum_{i=1}^n Y_i)}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}$$

e

$$a = \frac{\sum_{i=1}^n Y_i - b \sum_{i=1}^n x_i}{n}.$$

A chamada equação (reta) de regressão é dada por

$$\hat{y} = a + bx$$

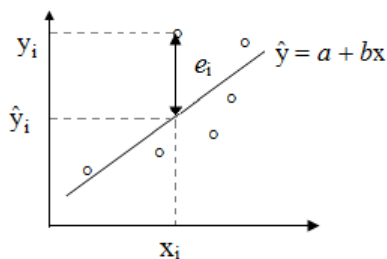
e para cada valor x_i ($i = 1, \dots, n$) temos, pela equação de regressão, o valor predito

$$\hat{y}_i = a + bx_i.$$

A diferença entre os valores observados e os preditos é chamada de resíduo, sendo obtido da seguinte forma

$$e_i = y_i - \hat{y}_i.$$

O resíduo relativo à i -ésima observação (e_i) pode ser considerado uma estimativa do erro aleatório (ϵ_i) desta observação (veja ilustração abaixo).



Para medir a *qualidade do modelo* podemos utilizar o coeficiente de determinação.

5.4.1 O Coeficiente de Determinação (R^2)

O coeficiente de determinação é uma medida descritiva da proporção da variação de Y que pode ser explicada por variações em X , segundo o modelo de regressão especificado. Ele é dado pela seguinte razão

$$R^2 = \frac{\sum_{i=1}^n (\hat{y} - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = \frac{\text{variação explicada pelo modelo}}{\text{variação total}}$$

em que $\bar{y} = \sum_{i=1}^n \frac{y_i}{n}$. Note que $0 \leq R^2 \leq 1$. Se $R^2 = 0$, o modelo não tem nenhum poder explicativo. Se $R^2 = 1$, o poder explicativo do modelo é total.

5.4.2 Teste de Hipóteses para o Coeficiente β

Para verificar se de fato X influencia no comportamento de Y e estender a estimativa b obtida na amostra, para a população, precisamos proceder um teste com as seguintes hipóteses

$$H_0: \beta = 0$$

$$H_1: \beta \neq 0.$$

Se não rejeitarmos H_0 significa que X não influencia no comportamento médio de Y , ou seja, no caso do modelo simples, não existe relação de regressão. Caso contrário, existe e X de fato influencia no comportamento médio de Y . Para tanto iremos utilizar a seguinte estatística de teste

$$T_c = \frac{|b|}{S_b}$$

em que $S_b^2 = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{(n-2) \sum_{i=1}^n (x_i - \bar{x})^2}$ e $T_c \sim t_{(n-2; \alpha/2)}$, sob a hipótese nula. Vamos retomar o exemplo da relação entre as variáveis *salário* e *anos de experiência*. Para obter o modelo de regressão linear no RStudio, podemos utilizar o comando *lm* e em seguida o comando *summary* para obter o resumo do modelo.

```
ajuste = lm(salario~exp)
summary(ajuste)
```

```
Call:
lm(formula = salario ~ exp)

Residuals:
    Min       1Q   Median       3Q      Max
-875.32 -137.49   87.12  237.04  407.18

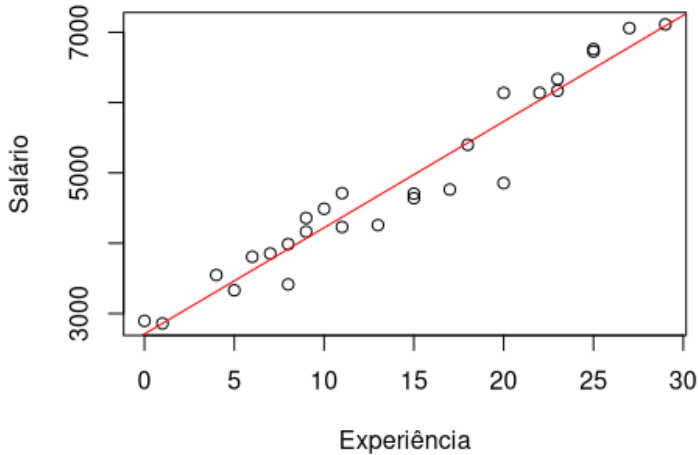
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2708.606    122.296    22.15  <2e-16 ***
exp          151.111     7.514    20.11  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 316.7 on 25 degrees of freedom
Multiple R-squared:  0.9418, Adjusted R-squared:  0.9395
F-statistic: 404.5 on 1 and 25 DF,  p-value: < 2.2e-16
```

O modelo obtido foi $Y = 2708,61 + 151,11X$. Observamos que o p-valor associado à variável *anos de experiência* foi menor que $2 * 10^{-16}$, ou seja, rejeitamos a hipótese de que $\beta = 0$ para um pequeno nível de significância, como por exemplo 5%. Isso indica que existe relação de regressão. Podemos interpretar o modelo da seguinte forma: a cada ano a mais de experiência, o indivíduo recebe a mais, em média, US\$151,11 em seu salário.

Podemos observar ainda que $R^2 = 0,94$, ou seja, o modelo explica 94% da variabilidade total dos dados, o que indica um bom ajuste. Podemos visualizar o ajuste do modelo também de forma gráfica:

```
plot(exp,salario, xlab = "Experiência", ylab = "Salário")
abline(ajuste, col="red")
```



Para fazer previsões utilizando o modelo devemos inserir novos valores da variável X pertencentes ao intervalo da própria variável na função *predict*. Vamos prever o aumento de salário médio para 10 e 11 anos de experiência.

```
predict(ajuste, newdata=data.frame(exp=c(10,11)),
interval="prediction")
```

	fit	lwr	upr
1	4219.712	3552.440	4886.984
2	4370.823	3704.848	5036.797

Observe que os valores salariais foram de US\$4219,71 e US\$4370,82 para 10 e 11 anos de experiência, respectivamente. Adicionalmente, obtemos um intervalo para cada uma dessas previsões.

6 REFERÊNCIAS

CRAWLEY, Michael J. **The R book**. John Wiley & Sons, 2012.

LANDEIRO, Victor Lemes. Introdução ao uso do programa R. **Manaus: Instituto Nacional de Pesquisas da Amazônia, 2011.**

MONTGOMERY, Douglas C.; PECK, Elizabeth A.; VINING, G. Geoffrey. **Introduction to linear regression analysis**. John Wiley & Sons, 2012.

MORETTIN, Pedro Alberto; BUSSAB, WILTON OLIVEIRA. **Estatística básica**. Saraiva Educação SA, 2017.

TRIOLA, Mario F. et al. **Introdução à estatística**. Rio de Janeiro: Itc, 2005.

TEAM, R. Core et al. R: A language and environment for statistical computing. 2020.

RUMSEY, Deborah. **Estatística II para leigos**. Alta Books Editora, 2018.

SHEATHER, Simon. **A modern approach to regression with R**. Springer Science & Business Media, 2009..

WILKINSON, Leland. The grammar of graphics. In: **Handbook of Computational Statistics**. Springer, Berlin, Heidelberg, 2012. p. 375-414.

7 SOBRE AS AUTORAS

Ana Hermínia Andrade e Silva

Doutora em Estatística pela Universidade Federal de Pernambuco - UFPE (2017). Mestre em Estatística pela Universidade Federal de Pernambuco - UFPE (2013). Bacharel em Estatística pela Universidade Federal da Paraíba - UFPB (2010). Atualmente é professora Adjunta II do Departamento de Estatística da UFPB, onde coordena o projeto de extensão “Estatística Aplicada em *Software* Livre” e colabora no projeto de extensão “Brincando e Aprendendo com a Estatística”. Tem experiência nas áreas de análise exploratória de dados, inferência, modelagem e estatística não-paramétrica.

Gilmara Alves Cavalcanti


Doutora em Modelos de Decisão e Saúde pela Universidade Federal da Paraíba - UFPB (2017). Mestre em Estatística e Métodos Quantitativos pela Universidade de Brasília - UnB (2000). Graduada em Estatística pela Universidade Federal do Rio Grande do Norte - UFRN (1997). Atualmente é professora Adjunta II do Departamento de Estatística da UFPB e coordenadora da Graduação em Estatística da UFPB, onde colabora com os projetos de extensão “Estatística Aplicada em *Software* Livre” e “Brincando e Aprendendo com a Estatística”. Tem experiência nas áreas de análise exploratória de dados, probabilidade, modelagem, controle de qualidade e amostragem.

Juliana Freitas Pires

Doutora em Matemática Computacional pela Universidade Federal de Pernambuco - UFPE (2014). Mestre em Estatística pela Universidade Federal de Pernambuco - UFPE (2009). Graduada em Licenciatura em Matemática pela Universidade Estadual de Santa Cruz - UESC (2005). Atualmente é professora adjunta IV do Departamento de Estatística da UFPB, onde colabora com o projeto de extensão “Estatística Aplicada em *Software* Livre”. Tem experiência nas áreas de modelagem e inferência paramétrica, atuando principalmente nos seguintes temas: correção de viés, verossimilhança perfilada, bootstrap, modelo linear hierárquico, influência local, efeitos aleatórios e efeitos fixos.

Maria Lídia Coco Terra

Doutora em Estatística pela Universidade Federal de Pernambuco - UFPE (2013). Mestre em Estatística pela Universidade Federal de Pernambuco - UFPE (2009). Graduada em Matemática pela Universidade Estadual de Santa Cruz - UESC (2006). Atualmente é professora adjunta II do Departamento de Estatística da UFPB e vice-coordenadora da Graduação em Estatística da UFPB, onde coordena o projeto de extensão “Brincando e Aprendendo com a Estatística” e colabora no projeto de extensão “Estatística Aplicada em *Software* Livre. Tem experiência nas áreas de análise exploratória de dados, inferência, probabilidade, regressão e MLG.

 Este livro foi diagramado
pela Editora UFPB em 2020,
utilizando a fonte Myriad Pro.

