# Estatísticas Descritivas

Gustavo Jun Yakushiji

Setembro de 2022

## Contents

# Pacotes utilizados

```
# Instalação de pacotes (execute esse comando apenas uma única vez)
install.packages(c("tidyverse", "readxl", "kableExtra", "summarytools"))

## Obs: ao instalar o pacote tidyverse, é instalado juntamente o pacote ggplot2
```

```
# Carregando os pacotes (execute esse comando a cada inicialização do RStudio)
library(tidyverse)
library(ggplot2)
library(readxl)
library(kableExtra)
library(summarytools)
```

A estatística descritiva se apresenta como a etapa inicial da maioria das análises estatísticas. Consiste em **organizar**, **descrever** e **resumir** os aspectos importantes de um conjunto de dados. Além disso, permite identificar valores atípicos (conhecidos também como *outliers*) presentes no conjunto de dados, possibilitando a realização de possíveis ajustes necessários.

Ao resumir um conjunto de dados, inevitavelmente, acaba-se perdendo algumas informações. Contudo, a utilização correta das ferramentas descritivas acabam gerando mais benefícios à análise como um todo. Por apenas descrever os dados, não é possível realizar generalizações ou conclusões acerca desses, sendo necessárias outras análises estatísticas.

# 1 Tipos de variáveis

Uma variável consiste em características e medidas de interesse (numéricas ou não numéricas) referentes a um indivíduo, uma amostra ou uma população, podendo ser divididas em **Qualitativas** e **Quantitativas**.

| Altura | Classificação |
|---|---|
| <1,60 | Baixo |
| 1,60-1,80 | Médio |
| 1,80-2,00 | Alto |
| >2,00 | Muito alto |

## 1.1 Variáveis Qualitativas (Categóricas)

Variáveis qualitativas são definidas por categorias que representam uma classificação. Pode ser subdividida em **Nominal** e **Ordinal**.

- **Variável Qualitativa Nominal**: característica de qualidade (atributo) sem nenhuma ordenação - Ex: gênero, etnia;
- **Variável Qualitativa Ordinal**: característica de qualidade (atributo) com possível ordenação a partir de algum critério - Ex: grau de escolaridade, renda.

## 1.2 Variáveis Quantitativas

Variáveis que apresentam valores numéricos que podem ser medidos em uma escala quantitativa. São subdivididas em **Discreta** e **Contínua**.

- **Variável Quantitativa Discreta**: quantificação de valores que se repetem, sendo normalmente números inteiros e que resultam de uma contagem - Ex: número de filhos;
- **Variável Quantitativa Contínua**: quantificação de valores com grandes intervalos, podendo assumir qualquer valor dentro deste mesmo intervalo e que resultam de uma mensuração - Ex: peso e altura.

Uma variável quantitativa pode ser transformada em qualitativa, de acordo com o tipo de dado e intuito da análise. Um exemplo seria dividir alturas em categorias.

```
tibble::tibble(
  Altura = c("<1,60", "1,60-1,80", "1,80-2,00", ">2,00"),
  Classificação = c("Baixo", "Médio", "Alto", "Muito alto")
) %>%
  kableExtra::kbl() %>%
  kableExtra::kable_styling(full_width = F, position = "c")
```

Por fim, podemos perceber nem todo número é uma variável quantitativa, como por exemplo, o número de CPF, RG e número de telefone, pois cada registro numérico representa um único indivíduo ou observações exclusivas.

No R, podemos utilizar as funções `str()` ou `dplyr::glimpse()` para observarmos as características do banco de dados e suas variáveis. Para isso, utilizaremos dados hipotéticos presentes na seguinte planilha excel:

```
dados <- readxl::read_xlsx("~/GitHub/PAP_Bioestatistica/dados/dados.xlsx")

str(dados)
```

```
## tibble [30 x 8] (S3: tbl_df/tbl/data.frame)
##  $ sexo          : chr [1:30] "M" "F" "M" "M" ...
##  $ grau_instrucao: chr [1:30] "Superior" "Superior" "Superior" "Ens Fundamental" ...
```

```
##  $ cidade        : chr [1:30] "urbano" "rural" "urbano" "urbano" ...
##  $ filhos        : num [1:30] 1 0 0 0 0 2 0 0 1 2 ...
##  $ idade         : num [1:30] 31 25 33 20 23 37 38 37 34 40 ...
##  $ altura        : num [1:30] 1.75 1.67 1.7 1.73 1.83 1.8 1.9 1.6 1.62 1.64 ...
##  $ peso          : num [1:30] 80 65 90 87 71 80 90 55 55 60 ...
##  $ salario       : num [1:30] 4.1 2.65 4.7 1.45 1.85 2.2 2.35 2.7 2.9 1.6 ...
```

```
dplyr::glimpse(dados)
```

```
## Rows: 30
## Columns: 8
## $ sexo           <chr> "M", "F", "M", "M", "M", "M", "M", "F", "F", "F", "M", ~
## $ grau_instrucao <chr> "Superior", "Superior", "Superior", "Ens Fundamental", ~
## $ cidade         <chr> "urbano", "rural", "urbano", "urbano", "urbano", "urban~
## $ filhos         <dbl> 1, 0, 0, 0, 0, 2, 0, 0, 1, 2, 0, 3, 0, 0, 1, 1, 0, 0, 2~
## $ idade          <dbl> 31, 25, 33, 20, 23, 37, 38, 37, 34, 40, 41, 46, 26, 41,~
## $ altura         <dbl> 1.75, 1.67, 1.70, 1.73, 1.83, 1.80, 1.90, 1.60, 1.62, 1~
## $ peso           <dbl> 80, 65, 90, 87, 71, 80, 90, 55, 55, 60, 88, 60, 68, 64,~
## $ salario        <dbl> 4.10, 2.65, 4.70, 1.45, 1.85, 2.20, 2.35, 2.70, 2.90, 1~
```

Ambas as funções informam a dimensão do banco de dados (número de linhas e colunas) e a classe das variáveis (quantitativa ou qualitativa). Note que, para cada coluna, adotou-se um tipo de variável. As colunas `sexo`, `grau_escolaridade` e `cidade` foram classificadas como `chr` - sigla para *charactere* - portanto, variáveis qualitativas; já `filhos`, `idade`, `altura`, `peso` e `salario`, foram classificados como `num` ou `dbl` - siglas para *numeric* e *double*, respectivamente - portanto, variáveis quantitativas.

# 2 Distribuições de frequência

Como visto anteriormente, cada tipo de variável apresenta suas particularidades. Com isso, as ferramentas **tabulares** e **gráficas** devem ser dispostas do modo mais adequado possível a cada caso. Estas técnicas permitem que se observe a disposição do conjunto de dados a partir de contagens.

As **Tabelas de frequências** podem apresentar tanto dados de frequência absoluta (contagem dos valores), como de frequência relativa (porcentagem da frequência absoluta em relação a um total). As frequências relativas possibilitam a comparação entre amostras ou populações com valores absolutos diferentes, pois se resumem a um mesmo total.

Assim sendo, o conjunto formado pelas categorias de variáveis e as suas respectivas frequência são denominadas **distribuição de frequências**.

No R, podemos utilizar funções nativas, no caso a `table()` (para frequências absolutas) e a `prop.table()` (para frequências relativas); mas também é possível utilizar o pacote `summarytools`, este sendo mais específico para a temática.

Exemplificaremos novamente com os mesmos dados hipotéticos utilizados anteriormente:

```
dados <- readxl::read_xlsx("~/GitHub/PAP_Bioestatistica/dados/dados.xlsx")
```

## 2.1 Variáveis Qualitativas

Para as variáveis qualitativas - tanto nominais, quanto ordinais - são utilizadas tabelas de frequências de uma determinada categoria ou de combinações de categorias (também conhecidas por tabelas de contingência/de

dupla entrada/cruzadas). Neste caso, a frequência das categorias é dada pelo *número de observações* da variável no conjunto de dados.

Graficamente, podem ser utilizados gráficos de barras e de setores (pizza), a partir da contagem ou das proporções (porcentagens) de variáveis categóricas.

A seguir, veremos exemplos de tabelas e gráficos de distribuições de frequências univariadas, bivariadas e multivariadas.

### 2.1.1 Univariada

Distribuições de frequências univariadas tratam de apenas uma única variável categórica.

- **Funções `table()` e `prop.table()`**

```
# Frequência absoluta
dados$grau_instrucao %>% table()
```

```
## .
## Ens Fundamental    Ensino Médio        Superior
##               6              10              14
```

```
# Frequência relativa
dados$grau_instrucao %>% table() %>% prop.table() %>% round(2)
```

```
## .
## Ens Fundamental    Ensino Médio        Superior
##            0.20            0.33            0.47
```

- **Pacote `summarytools`**

A função `summarytools::freq()` apresenta a frequência absoluta (`Freq`), a frequência relativa (`% Valid`), as frequências acumuladas (`%_Cum.`) e o total (`% Total`). Além disso, mostra a quantidade de valores ausentes (`NA`).

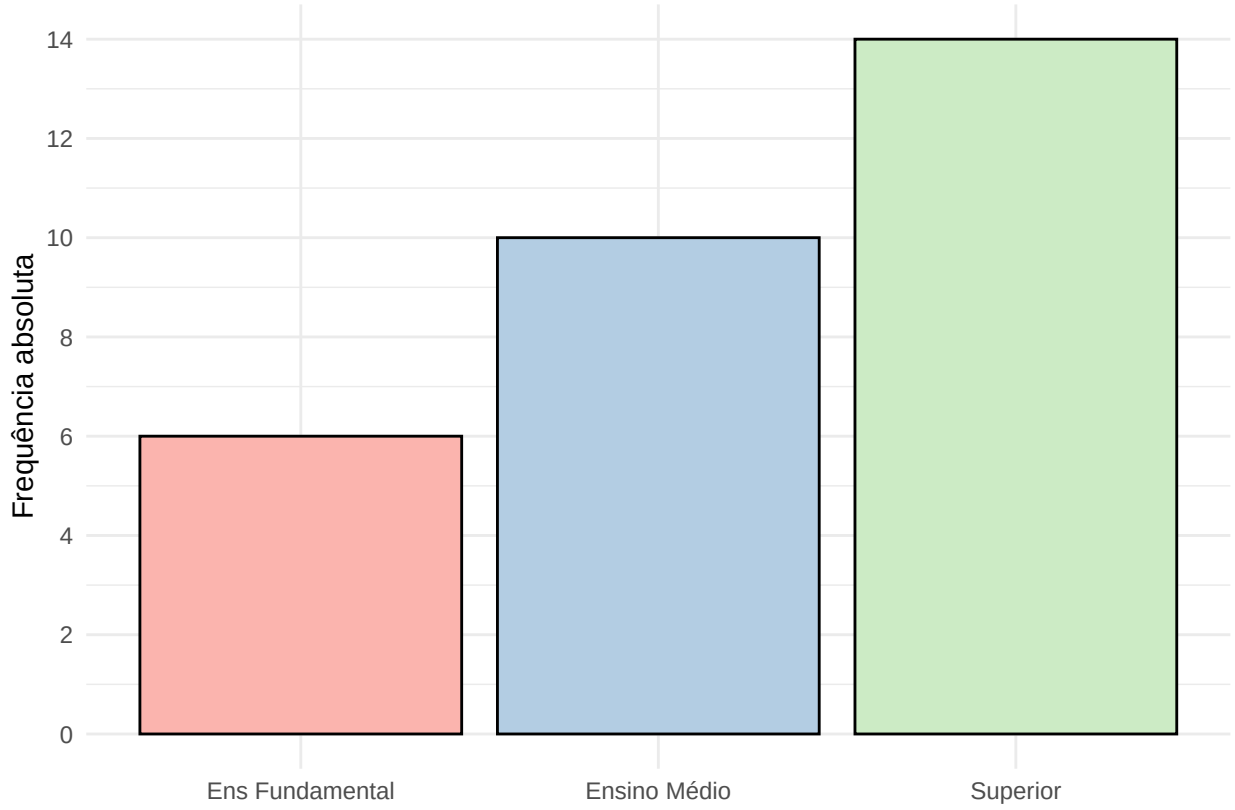
```
dados$grau_instrucao %>% freq()
```

```
## Frequencies
## dados$grau_instrucao
## Type: Character
##
##                        Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
## --------------------- ------ --------- -------------- --------- --------------
##       Ens Fundamental      6     20.00          20.00     20.00          20.00
##          Ensino Médio     10     33.33          53.33     33.33          53.33
##              Superior     14     46.67         100.00     46.67         100.00
##                  <NA>      0                                0.00         100.00
##                 Total     30    100.00         100.00    100.00         100.00
```

- **Gráficos**

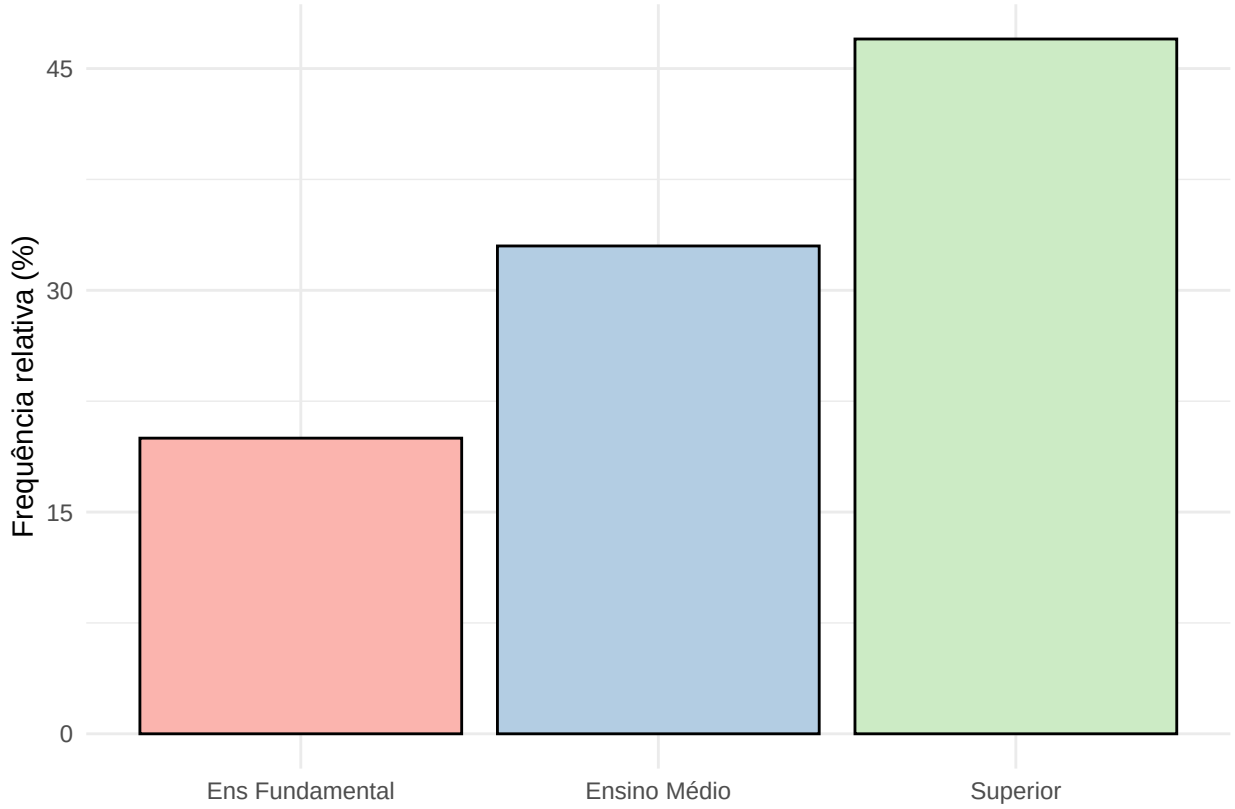
```r
educ <- dados %>%
  count(grau_instrucao) %>%
  mutate(porcentagem = prop.table(n)*100,
         porcentagem = round(porcentagem, 2))
```

```r
# Gráfico de barras/colunas - Frequência absoluta
ggplot(data = educ,
       aes(x = grau_instrucao,
           y = n,
           fill = grau_instrucao))+
  geom_col(show.legend = FALSE, color = "black")+
  labs(x = "", y = "Frequência absoluta")+
  theme_minimal()+
  scale_y_continuous(breaks = seq(0, 14, 2))+
  scale_fill_brewer(palette = "Pastel1")
```

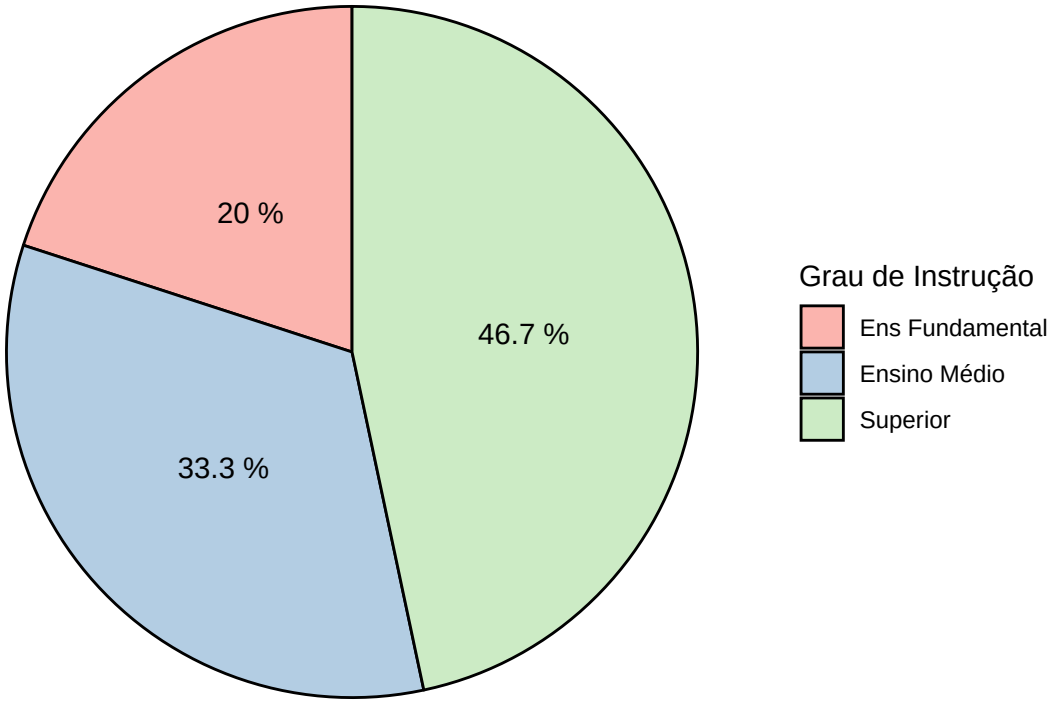



```r
# Gráfico de barras/colunas - Frequência relativa
ggplot(data = educ,
       aes(x = grau_instrucao,
           y = porcentagem %>% round(),
           fill = grau_instrucao))+
  geom_col(show.legend = FALSE, color = "black")+
  labs(x = "", y = "Frequência relativa (%)")+
  theme_minimal()+
```

```
  scale_y_continuous(breaks = seq(0, 45, 15))+
  scale_fill_brewer(palette = "Pastel1")
```




```
# Gráfico de setores (pizza) - Frequência relativa
ggplot(data = educ,
       aes(x = "",
           y = porcentagem,
           fill = grau_instrucao))+
  geom_bar(stat = "identity",
           color="black")+
  geom_text(aes(label = porcentagem %>% round(1) %>% paste("%")),
            position = position_stack(vjust = 0.5))+
  coord_polar(theta = "y", start = 0)+
  theme_void()+
  labs(fill = "Grau de Instrução")+
  scale_fill_brewer(palette = "Pastel1")
```




### 2.1.2 Bivariada

Distribuições de frequências bivariadas relacionam duas variáveis categóricas.

- **Funções `table()` e `prop.table()`**

Para a construção de tabelas de contingência, basta colocar as variáveis desejadas dentro da função `table()`. Além disso, na função `prop.table()`, o argumento `margin =` indica o total referente ao conjunto de dados como um todo, ao total de linhas (`margin = 1`) ou ao total de colunas (`margin = 2`).

```
# Frequência absoluta
table(dados$sexo, dados$grau_instrucao)
```

```
##
##     Ens Fundamental Ensino Médio Superior
##   F               1            4        8
##   M               5            6        6
```

```
# Frequência relativa ao total do conjunto de dados
table(dados$sexo, dados$grau_instrucao) %>% prop.table() %>% round(2)
```

```
##
##     Ens Fundamental Ensino Médio Superior
##   F            0.03         0.13     0.27
##   M            0.17         0.20     0.20
```

```r
# Frequência relativa ao total das linhas
table(dados$sexo, dados$grau_instrucao) %>% prop.table(margin = 1) %>% round(2)
```

```
##
##     Ens Fundamental Ensino Médio Superior
##   F            0.08         0.31     0.62
##   M            0.29         0.35     0.35
```

```r
# Frequência relativa ao total das colunas
table(dados$sexo, dados$grau_instrucao) %>% prop.table(margin = 2) %>% round(2)
```

```
##
##     Ens Fundamental Ensino Médio Superior
##   F            0.17         0.40     0.57
##   M            0.83         0.60     0.43
```

- **Pacote `summarytools`**

A função `summarytools::ctable()` monta a tabela de contingência. Por padrão, a proporção é feita pelo total das linhas (`r`). Para fazer outros tipos de proporções, deve-se utilizar o argumento `prop =`, sendo `t` para o total do conjunto de dados e `c` para as colunas.

```r
# Proporção por linhas (prop = "r")
ctable(x = dados$sexo,
       y = dados$grau_instrucao,
       prop = "r")
```

```
## Cross-Tabulation, Row Proportions
## sexo * grau_instrucao
## Data Frame: dados
##
## ------- ---------------- ----------------- -------------- ------------ -------------
##           grau_instrucao   Ens Fundamental   Ensino Médio     Superior         Total
##    sexo
##       F                          1 ( 7.7%)      4 (30.8%)    8 (61.5%)   13 (100.0%)
##       M                          5 (29.4%)      6 (35.3%)    6 (35.3%)   17 (100.0%)
##   Total                          6 (20.0%)     10 (33.3%)   14 (46.7%)   30 (100.0%)
## ------- ---------------- ----------------- -------------- ------------ -------------
```

```r
# Proporção por colunas (prop = "c")
ctable(x = dados$sexo,
       y = dados$grau_instrucao,
       prop = "c")
```

```
## Cross-Tabulation, Column Proportions
## sexo * grau_instrucao
## Data Frame: dados
##
## ------- ---------------- ----------------- -------------- ------------- -------------
##           grau_instrucao   Ens Fundamental   Ensino Médio      Superior         Total
```

```
##    sexo
##       F                          1 ( 16.7%)     4 ( 40.0%)    8 ( 57.1%)   13 ( 43.3%)
##       M                          5 ( 83.3%)     6 ( 60.0%)    6 ( 42.9%)   17 ( 56.7%)
##   Total                          6 (100.0%)    10 (100.0%)   14 (100.0%)   30 (100.0%)
## ------- ---------------- ----------------- -------------- ------------- -------------
```

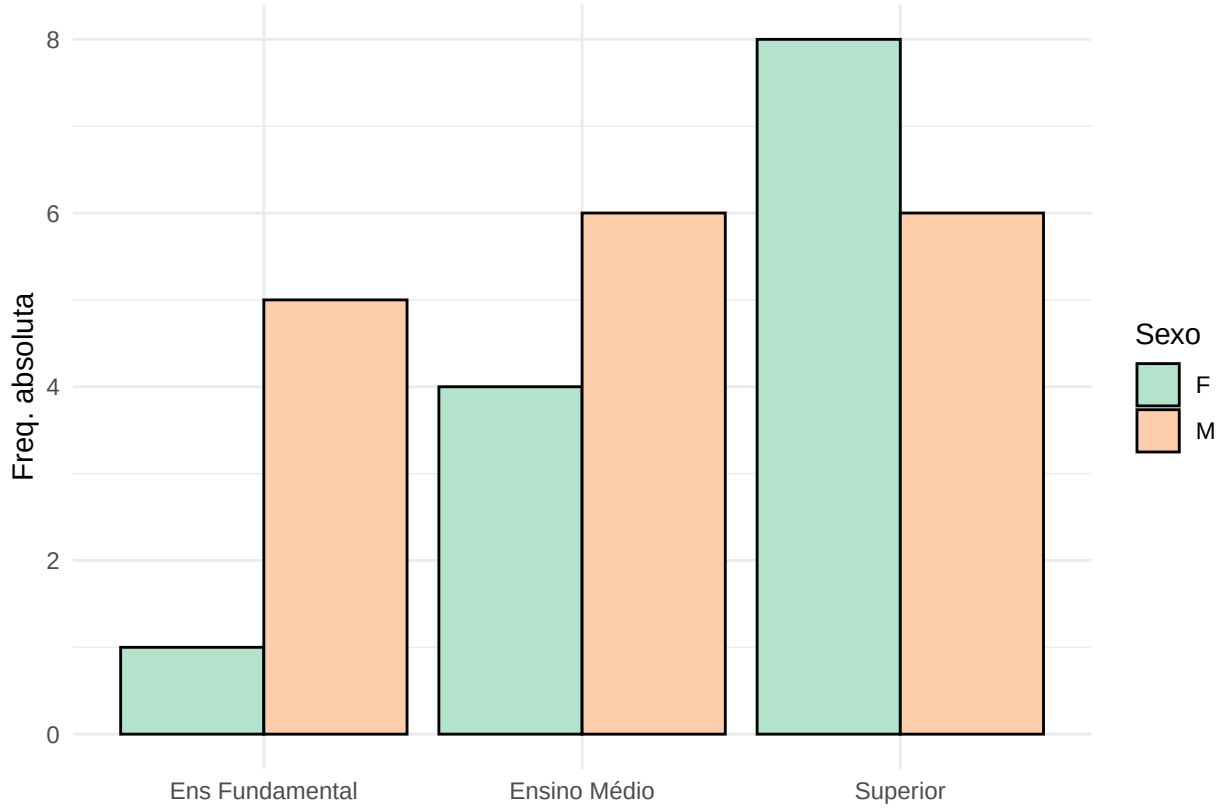
```
# Proporção total (prop = "t")
ctable(x = dados$sexo,
       y = dados$grau_instrucao,
       prop = "t")
```

```
## Cross-Tabulation, Total Proportions
## sexo * grau_instrucao
## Data Frame: dados
##
## ------- ---------------- ----------------- -------------- ------------ -------------
##           grau_instrucao   Ens Fundamental   Ensino Médio     Superior         Total
##    sexo
##       F                          1 ( 3.3%)      4 (13.3%)    8 (26.7%)   13 ( 43.3%)
##       M                          5 (16.7%)      6 (20.0%)    6 (20.0%)   17 ( 56.7%)
##   Total                          6 (20.0%)     10 (33.3%)   14 (46.7%)   30 (100.0%)
## ------- ---------------- ----------------- -------------- ------------ -------------
```

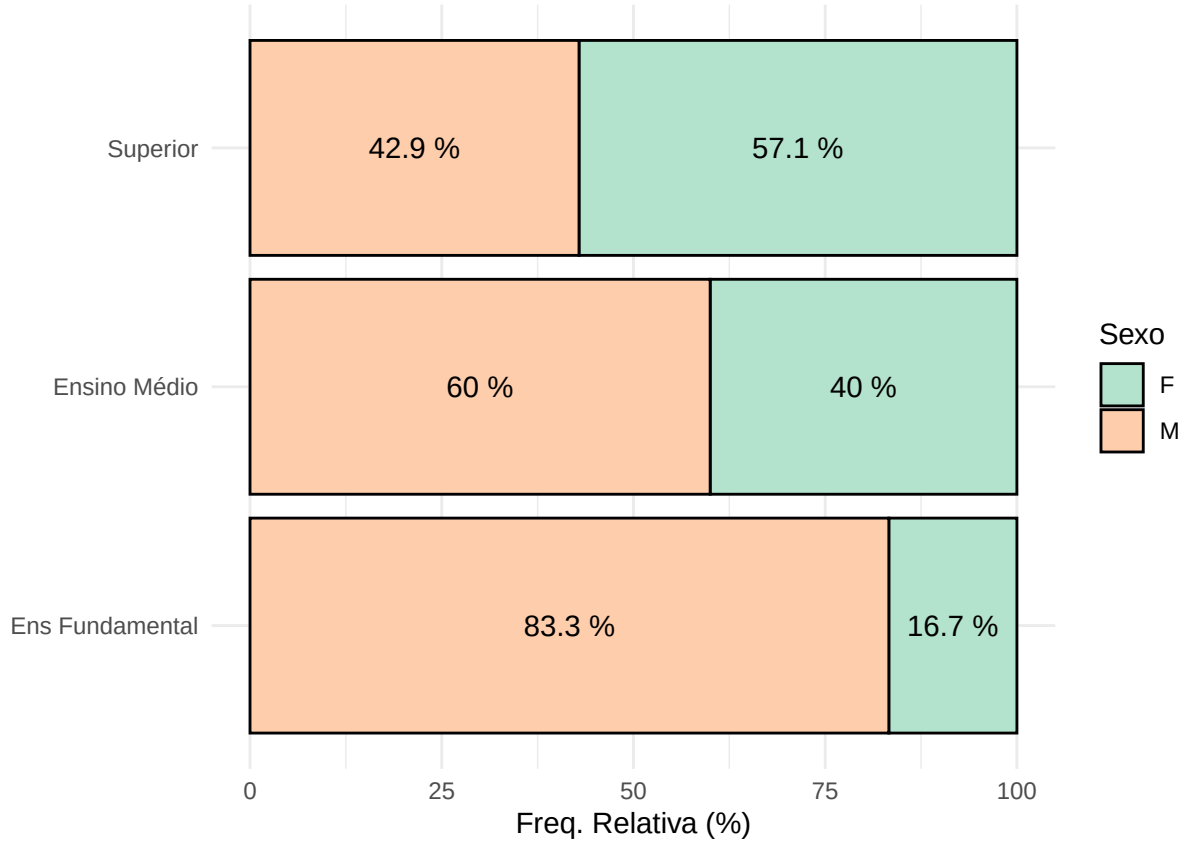
- **Gráficos**

```
# Gráfico de barras/colunas - Frequência absoluta
dados %>%
  select(sexo, grau_instrucao) %>%
  group_by(sexo) %>%
  count(grau_instrucao) %>%
  ggplot(aes(x = grau_instrucao,
             y = n,
             fill = sexo))+
  geom_col(position = "dodge", color = "black")+
  labs(x = "", y = "Freq. absoluta", fill = "Sexo")+
  scale_fill_discrete(labels = c("Feminino", "Masculino"))+
  theme_minimal()+
  scale_fill_brewer(palette = "Pastel2")
```

```
FALSE Scale for 'fill' is already present. Adding another scale for 'fill', which
FALSE will replace the existing scale.
```




```
# Gráfico de barras/colunas - Frequência relativa
dados %>%
  select(sexo, grau_instrucao) %>%
  group_by(grau_instrucao) %>%
  count(sexo) %>%
  mutate(perc = prop.table(n)*100,
         perc = round(perc, 1)) %>%
  ggplot(aes(x = grau_instrucao,
             y = perc,
             fill = sexo))+
  geom_bar(stat="identity", color = "black")+
  labs(x = "", y = "Freq. Relativa (%)", fill = "Sexo")+
  geom_text(aes(label = perc %>% paste("%")),
            position = position_stack(vjust = 0.5))+
  coord_flip()+
  theme_minimal()+
  scale_fill_brewer(palette = "Pastel2")
```




### 2.1.3 Multivariada

Distribuições de frequências multivariadas combinam de mais de duas variáveis categóricas.

- **Pacote `summarytools`**

Utiliza-se a combinação das funções `summarytools::stby()` e `summarytools::ctable()`.

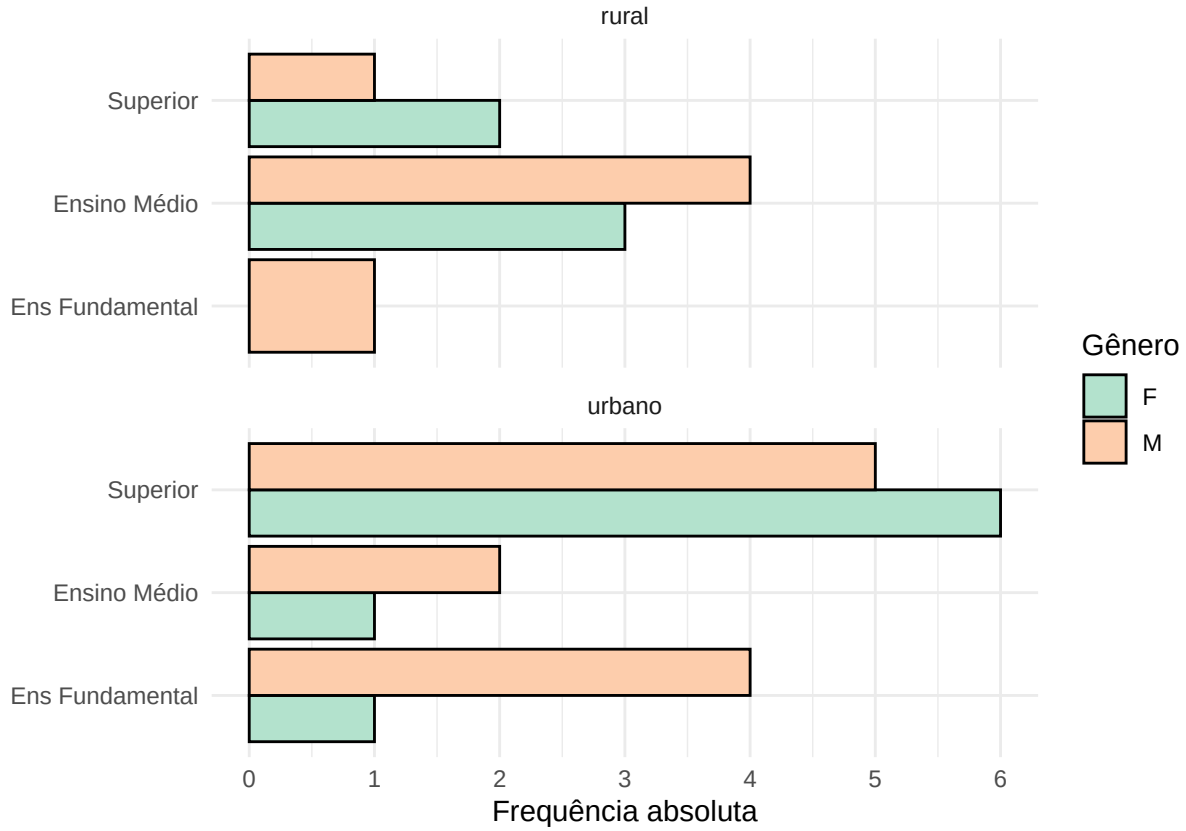
```
stby(list(x = dados$cidade,
          y = dados$grau_instrucao),
     INDICES = dados$sexo,
     FUN = ctable,
     prop = "t")
```

```
## Cross-Tabulation, Total Proportions
## cidade * grau_instrucao
## Data Frame: dados
## Group: sexo = F
##
## -------- ---------------- ----------------- -------------- ------------ -------------
##            grau_instrucao   Ens Fundamental   Ensino Médio    Superior         Total
##   cidade
##    rural                          0 (0.0%)      3 (23.1%)   2 (15.4%)    5 ( 38.5%)
##   urbano                          1 (7.7%)      1 ( 7.7%)   6 (46.2%)    8 ( 61.5%)
```

```
##    Total                          1 (7.7%)      4 (30.8%)   8 (61.5%)   13 (100.0%)
## -------- ---------------- ----------------- -------------- ----------- -------------
##
## Group: sexo = M
##
## -------- ---------------- ----------------- -------------- ----------- -------------
##            grau_instrucao   Ens Fundamental   Ensino Médio    Superior         Total
##   cidade
##    rural                          1 ( 5.9%)      4 (23.5%)   1 ( 5.9%)    6 ( 35.3%)
##   urbano                          4 (23.5%)      2 (11.8%)   5 (29.4%)   11 ( 64.7%)
##    Total                          5 (29.4%)      6 (35.3%)   6 (35.3%)   17 (100.0%)
## -------- ---------------- ----------------- -------------- ----------- -------------
```

- **Gráficos**

```
# Gráfico de barras/colunas - Frequência absoluta
ggplot(data = dados) +
  aes(x = grau_instrucao, fill = sexo) +
  geom_bar(position = "dodge", color = "black") +
  facet_wrap(~cidade, ncol = 1)+
  scale_y_continuous(breaks = seq(0, 7, 1))+
  labs(x = "", y = "Frequência absoluta", fill = "Gênero")+
  theme_minimal() +
  scale_fill_brewer(palette = "Pastel2")+
  coord_flip()
```



## 2.2 Variáveis Quantitativas

Para variáveis quantitativas, pode-se utilizar tabelas de distribuição de frequências para representar a ocorrência de valores em **classes** pré-estabelecidas, podendo conter intervalos iguais ou diferentes, de acordo com o intuito da análise e distribuição dos dados.

No R, deve-se seguir alguns passos para criar as classes, cujos intervalos são definidos da seguinte maneira:

- Determinar valor máximo e mínimo do conjunto de dados;
- Número de classes: determinada pela Regra de Sturges, que define o número de classes em que os intervalos serão divididos. Calculada por $k = 1 + 3,222 \log_{10}(n)$, onde $k$ é o número de classes e $n$ é o número de observações. No R, utilizamos a função `nclass.Sturges()`;
- Construção da tabela de distribuição de frequências.

Caso possível, pode-se aplicar as mesmas funções utilizadas para as variáveis qualitativas para as variáveis quantitativas.

Os gráficos mais usuais utilizados são o histograma, densidade e ramo-e-folhas.

### 2.2.1 Variáveis Quantitativas Discretas

- **Funções `table()` e `prop.table()`**

```
# Determinando valor máximo e mínimo
range(dados$idade)
```

```
## [1] 20 46
```

```
# Calculando o número de classes (Regra de Sturges)
(nclasse <- nclass.Sturges(dados$idade))
```

```
## [1] 6
```

```
# Frequência absoluta
table(cut(dados$idade, seq(20, 50, l = nclasse)))
```

```
##
## (20,26] (26,32] (32,38] (38,44] (44,50]
##       4       7      10       7       1
```

```
# Frequência relativa
round(prop.table(table(cut(dados$idade, seq(20, 50, l = nclasse)))), 2)
```

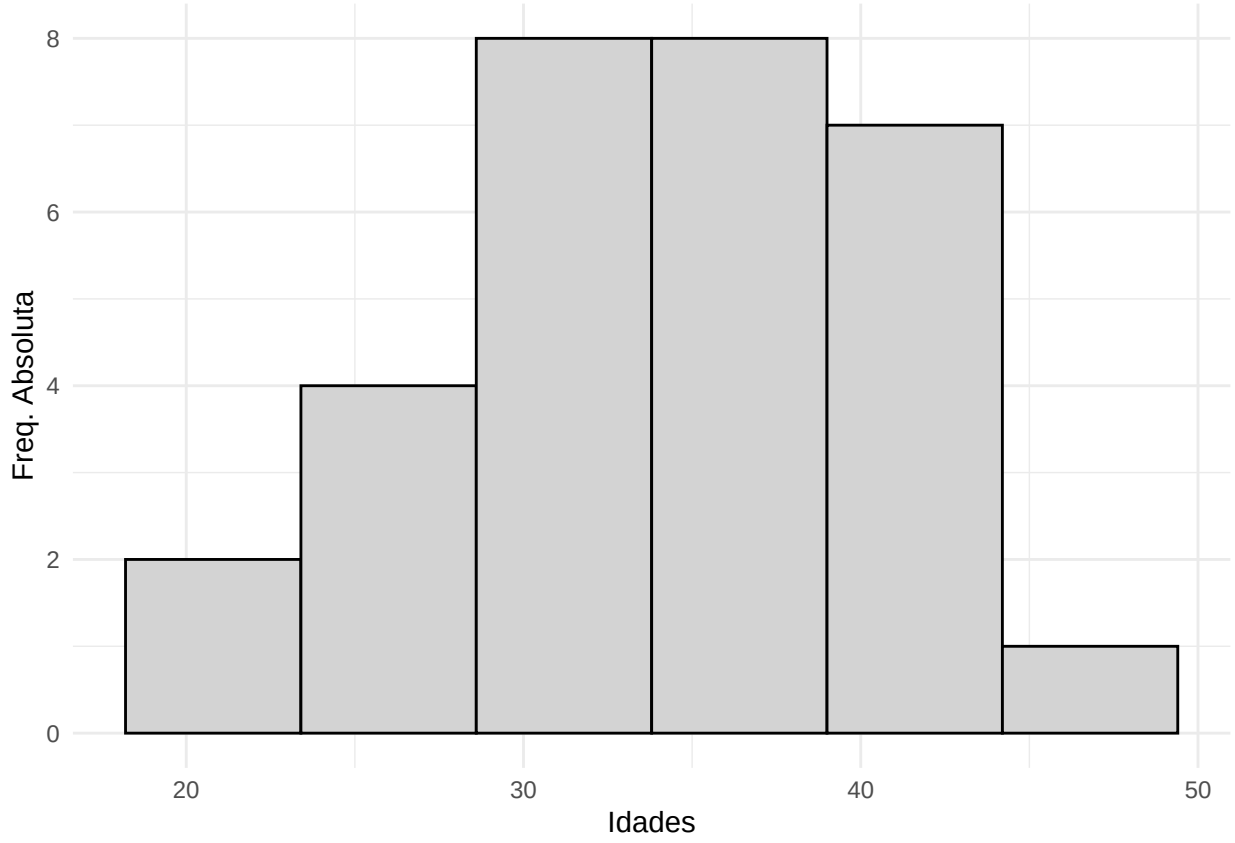
```
##
## (20,26] (26,32] (32,38] (38,44] (44,50]
##    0.14    0.24    0.34    0.24    0.03
```

Analisando a construção da tabela de distribuição de frequências, observa-se que a amplitude dos dados varia entre 20 e 46 anos. Arbitrariamente, podemos definir os valores que adotaremos como mínimos e máximos, desde que haja uma justificativa plausível. Neste caso, foi adotado 20 como idade mínima e 50 como idade máxima, devido a proximidade dos valores extremos a estes valores arredondados.

Aplicando a Regra de Sturges, definimos que o número de classes é 6. Posteriormente, montou-se as tabelas de frequências absolutas e relativas, divididas em 6 classes (`cut()`).

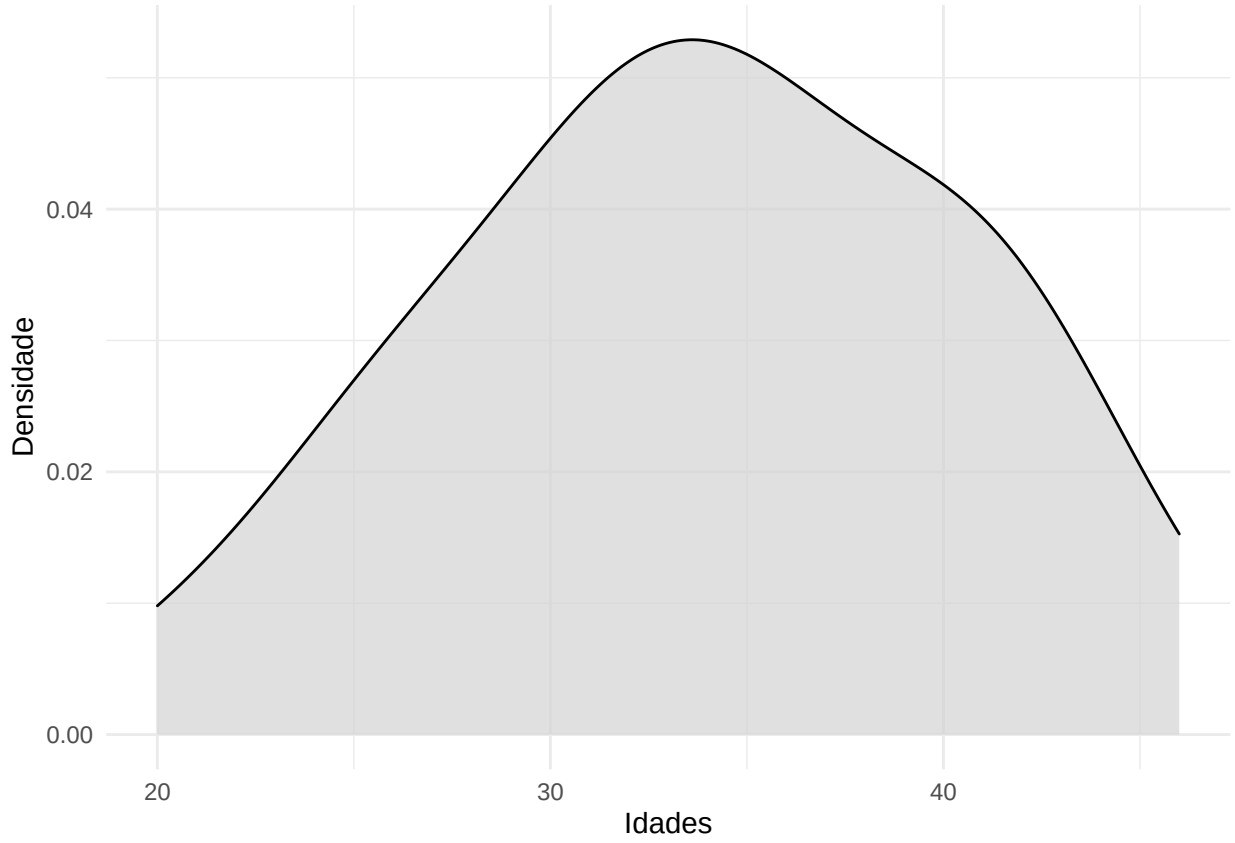
- **Gráficos**

```
# Histograma
ggplot(data = dados,
       aes(x = idade))+
  geom_histogram(bins = nclasse,
                 show.legend = FALSE,
                 color = "black",
                 fill = "lightgrey")+
  labs(x = "Idades", y  = "Freq. Absoluta")+
  theme_minimal()
```




```
# Densidades (Simetria)
ggplot(data = dados,
       aes(x = idade))+
  geom_density(show.legend = FALSE,
               color = "black",
```

```
               fill = "lightgrey",
               alpha = 0.7)+
  labs(x = "Idades", y  = "Densidade")+
  theme_minimal()
```




```
# Ramo-e-folhas
stem(dados$idade, scale = 0.5)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   2 | 0356679
##   3 | 001123344556778
##   4 | 00112336
```

### 2.2.2 Variáveis Quantitativas Contínuas

- **Funções `table()` e `prop.table()`**

```
# Determinando valor máximo e mínimo
range(dados$salario)
```

```
## [1] 1 8
```

```r
# Calculando o número de classes (Regra de Sturges)
(nclasse <- nclass.Sturges(dados$salario))
```

```
## [1] 6
```

```r
# Frequência absoluta
table(cut(dados$salario, seq(1, 8, l = nclasse)))
```

```
##
##   (1,2.4] (2.4,3.8] (3.8,5.2] (5.2,6.6]   (6.6,8]
##        11        12         4         1         1
```

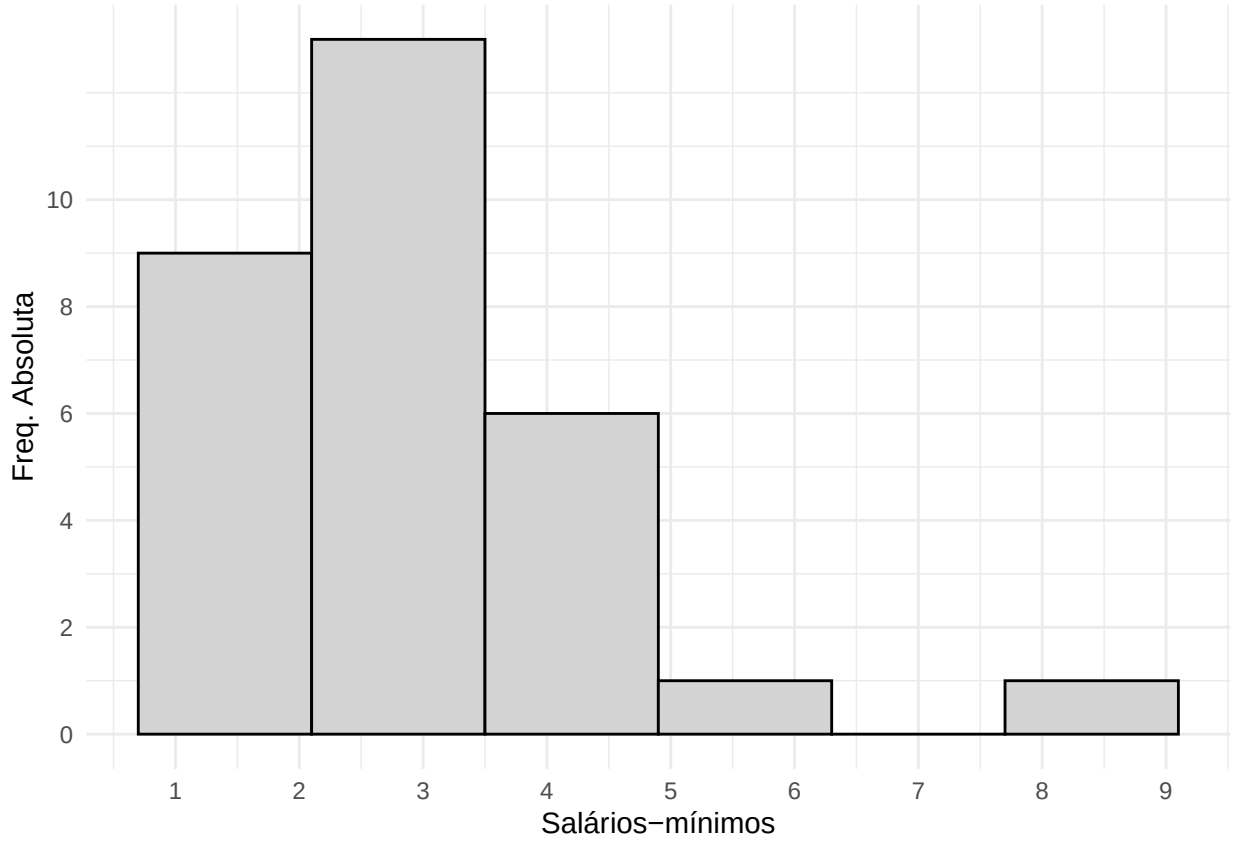
```r
# Frequência relativa
round(prop.table(table(cut(dados$salario, seq(1, 8, l = nclasse)))),2)
```

```
##
##   (1,2.4] (2.4,3.8] (3.8,5.2] (5.2,6.6]   (6.6,8]
##      0.38      0.41      0.14      0.03      0.03
```

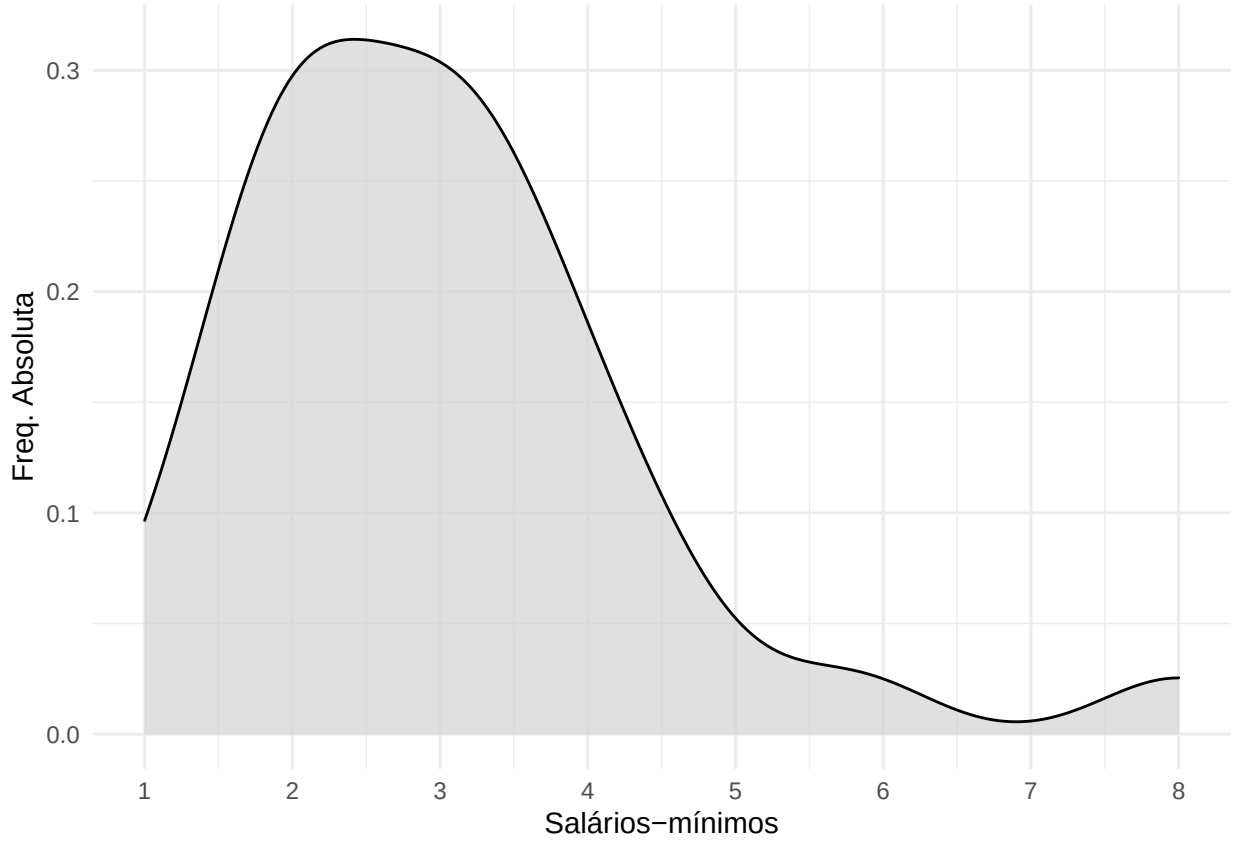
Neste caso, observa-se que a amplitude dos dados varia entre 1 e 8 salários-mínimos, sendo adotados o 1 como valor mínimo e 8 como valor máximo.

- **Gráficos**

```r
# Histograma
ggplot(data = dados,
       aes(x = salario))+
  geom_histogram(bins = nclasse,
                 show.legend = FALSE,
                 color = "black",
                 fill = "lightgrey")+
  scale_x_continuous(breaks = seq(1, 9, 1))+
  scale_y_continuous(breaks = seq(0, 10, 2))+
  labs(x = "Salários-mínimos", y = "Freq. Absoluta")+
  theme_minimal()
```




```
# Densidades (Simetria)
ggplot(data = dados,
       aes(x = salario))+
  geom_density(show.legend = FALSE,
               color = "black",
               fill = "lightgrey",
               alpha = 0.7)+
  scale_x_continuous(breaks = seq(1, 8, 1))+
  labs(x = "Salários-mínimos", y = "Freq. Absoluta")+
  theme_minimal()
```

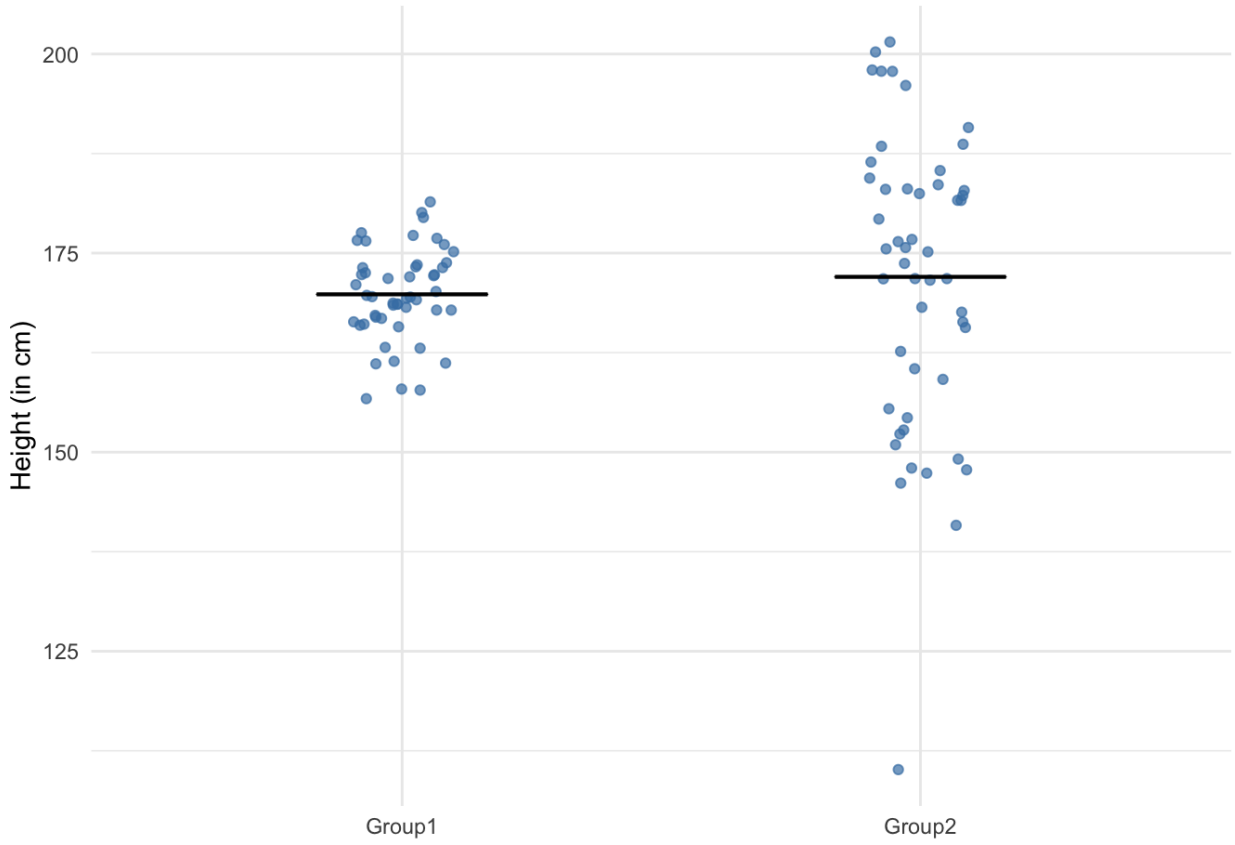



```
# Ramo-e-folhas
stem(dados$salario, scale=0.5)
```

```
##
##   The decimal point is at the |
##
##   0 | 0567999
##   2 | 11234778901234577
##   4 | 01378
##   6 |
##   8 | 0
```

# 3 Medidas-Resumo

Consiste no resumo dos dados para alguns valores que sejam representativos para toda a série. São divididos em medidas de tendência central e medidas de dispersão, sendo complementares entre si, a fim de resumir os dados de maneira concisa, porém de forma mais completa o possível.




No gráfico acima, foram divididos dois grupos, cujas médias das alturas (linhas pretas) são bem próximas. Contudo, pode-se observar que a dispersão dos valores entre os grupo é muito distinto. Sendo assim, esse exemplo ilustra a importância de se observar, de maneira conjunta, as medidas de posição e de dispersão.

## 3.1 Medidas de tendência central (de posição ou de localização)

Uma medida de tendência central informa a localização de um valor em relação a outros valores no conjunto de dados.

### 3.1.1 Média

A média aritmética é a soma das observações dividida pelo número delas. Assumindo que a variável $x$ possua $n$ valores $x_i$, sendo $i = 1, 2, \ldots,$ n, a média aritmética é calculada por meio da expressão:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + ... + x_n}{n}$$

No R, utilizamos a função `mean()` para calcular a média aritmética.

```
mean(dados$idade)
```

```
## [1] 33.93333
```

```r
mean(dados$altura)
```

```
## [1] 1.713667
```

Pode-se também calcular a média ponderada relativa ao número de observações de dada variável, ou à frequência relativa das observações:

- Média ponderada de $n$ observações:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{k} p_i x_i = \frac{p_1 x_1 + p_2 x_2 + ... + p_n x_n}{n}$$

- Média ponderada da Frequência relativa:

$$\bar{x} = \sum_{i=1}^{k} f_i x_i = \frac{f_1 x_1 + f_2 x_2 + ... + f_n x_n}{n}$$

No R, utilizamos a função `weighted.mean()` para calcular a média ponderada, tendo como primeiro argumento os valores e o segundo, as ponderações.

```r
# Ex: Média da Prova 1 (8,5) e Prova 2 (7,6), tendo pesos 0,4 e 0,6, respectivamente
weighted.mean(c(8.5, 7.6), c(0.4, 0.6))
```

```
## [1] 7.96
```

Uma observação pertinente é que a média é bastante influenciada por valores atípicos (*outliers*) presentes no conjunto de dados. Portanto, a média é uma medida recomendada para distribuições simétricas. Como exemplo, substituiremos o primeiro valor de um conjunto de dados por um número dez vezes maior.

```r
# Ex: substituir o primeiro valor de um conjunto de dados por um número 10x maior
mean(c(5,8,3,5,6))
```

```
## [1] 5.4
```

```r
mean(c(50,8,3,5,6))
```

```
## [1] 14.4
```

### 3.1.2 Mediana

A mediana é o valor que ocupa a posição central da série de observações quando estão ordenados (`sort()`), ou seja, o valor tal que 50% dos valores da variável estão acima da mediana e 50% estão abaixo. Diferentemente da média, a mediana não é afetada por valores atípicos, sendo uma medida recomendada para distribuições assimétricas.

Quando o número de observações for **par**, tendo os dados ordenados, usa-se como mediana a média aritmética das duas observações centrais. Para observações em número **ímpar**, a mediana será o valor da observação central.

- Se $n$ for par:

$$md(x) = \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1})$$

- Se $n$ for ímpar:

$$md(x) = x_{\frac{n+1}{2}}$$

No R, utilizamos a função `median()` para calcular a mediana.

```
median(dados$idade)
```

```
## [1] 34
```

```
median(dados$salario)
```

```
## [1] 2.825
```

Todavia, a mediana possui a limitação de ser pouco específica e genérica para diferentes séries de dados. O exemplo a seguir ilustra tal fato.

```
# Exemplo: Número de questões corretas nas provas de Estatística e Física
estudantes <- tibble(estatistica = c(10,10,10,18,20),
                     fisica = c(5,7,10,10,11))
estudantes
```

```
## # A tibble: 5 x 2
##   estatistica fisica
##         <dbl>  <dbl>
## 1          10      5
## 2          10      7
## 3          10     10
## 4          18     10
## 5          20     11
```

```
# Mediana
## Estatística
median(estudantes$estatistica)
```

```
## [1] 10
```

```
# Mediana
## Física
median(estudantes$fisica)
```

```
## [1] 10
```

```r
# Média
## Estatística
mean(estudantes$estatistica)
```

```
## [1] 13.6
```

```r
# Média
## Física
mean(estudantes$fisica)
```

```
## [1] 8.6
```

No caso, podemos notar que a mediana em ambas as disciplinas é 10, porém, houve mais acertos na prova de estatística do que na de física. A média na prova de estatística foi de 13,6, enquanto que na de física, 8,6.

### 3.1.3 Moda

A moda é a medida que ocorre com maior frequência em um conjunto de dados, sendo possível tanto para variáveis quantitativas (principalmente as quantitativas discretas), como para qualitativas. Há situações em que não existe moda; em que ocorrem duas modas (bimodal); e onde há mais de duas modas (multimodal).

Para determiná-la no R, devemos criar uma tabela de frequência absoluta.

```r
table(dados$filhos)
```

```
##
##  0  1  2  3  4
## 15  7  6  1  1
```

Na primeira linha, observamos o número de filhos e na segunda, suas respectivas frequências. Nesse exemplo, a moda é 0 filho, com frequência absoluta de 15 observações.

## 3.2 Medidas de dispersão

As medidas de dispersão nos dá noção acerca da distribuição dos dados, ou seja, se os dados estão próximos ou dispersos em relação a um ponto de referência, comumente sendo uma medida de tendência central.

### 3.2.1 Amplitude

Consiste na distância entre o maior valor (`max()`) e o menor valor (`min()`) do conjunto de dados. Para calcularmos no R, utilizamos as funções `range()` e `diff()`.

```r
# Determinando os valores máximo e mínimo
range(dados$altura)
```

```
## [1] 1.51 1.90
```

```r
# Determinando a amplitude (Máx - Mín)
range(dados$altura) %>% diff()
```

```
## [1] 0.39
```

A limitação da amplitude se dá pelo fato de considerar somente os valores extremos do conjunto de dados, não dando noção de como estão distribuídos os demais valores ou se existem valores atípicos (*outliers*).

### 3.2.2 Quantil

Os quantis são medidas de posição que dividem os valores, em ordem crescente, em $q$ partes iguais, ou em $q$ partes com a mesma proporção de valores. Esta divisão dá origem a *q-quantis*. Alguns deles possuem nomenclaturas especiais devido a maior utilização:

- **Percentis**: divisão em 100-quantis;
- **Decis**: divisão em 10-quantis;
- **Quintis**: divisão em 5-quantis;
- **Quartis**: divisão em 4-quantis;
- **Tercis**: divisão em 3-quantis.

**3.2.2.1 Percentil** Percentis são medidas que dividem os dados ordenados em 100 partes iguais, ou seja, é possível calcular 99 percentis.

O percentil é representado por $P_\alpha$, sendo $\alpha$ um dos 99 percentis. No R, utilizamos a função `quantile()` para calcular os percentis.

```r
# P13%
quantile(dados$altura, 0.13, type = 3)
```

```
## 13%
## 1.6
```

**3.2.2.2 Quartil** Quartis são medidas de posição que dividem os valores da variável em quatro partes:

- Q1 (primeiro quartil ou quartil inferior): define o valor para qual 25% dos valores estão abaixo dele; equivalente ao percentil 25% ($P_{25}$);
- Q2 (segundo quartil): é o valor que tem 50% dos valores abaixo e 50% acima; equivale à mediana e ao percentil 50% ($P_{50}$);
- Q3 (terceiro quartil ou quartil superior): define o valor que possui 75% dos dados abaixo dele; equivalente ao percentil 75% ($P_{75}$).

Há diversos algoritmos para o cálculo dos quartis. Dependendo do algoritmo, valores ligeiramente diferentes serão obtidos. No R, há 9 tipos programados, definidos a partir do argumento `type =`, dentro da função `quantile()`. Ademais, na função `quantile()`, os quartis devem ser expressos em percentis.

```r
# Quartis
quantile(dados$altura, c(0.25, 0.50, 0.75), type = 3)
```

```
##  25%  50%  75%
## 1.64 1.73 1.75
```

Também podemos utilizar a função `summary()`, que fornece os quartis 0% (min), 25% (Q1), 50%(Q2 - median), 75% (Q3), 100% (max), baseados no algoritmo `type = 7`, além do valor da média.

```
# Função summary
summary(dados$altura)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.510   1.643   1.730   1.714   1.780   1.900
```

**3.2.2.3 Distância interquantil (IQR)** A **distância interquantil** é uma medida de dispersão que não sofre a influência de *outliers*. É dada pela diferença entre o Q3 e o Q1, ou seja, é a amplitude dos valores centrais do conjunto de dados.

$$IQR = Q3 - Q1$$

No R, utilizamos a função `IQR()` para calcular a distância interquantil.

```
# Distância interquantil
IQR(dados$altura)
```

```
## [1] 0.1375
```

**3.2.2.4 Boxplot** O boxplot é um diagrama que fornece noções de posição, dispersão, assimetria, caudas e valores atípicos.

Para construi-lo, consideremos um retângulo onde estão representados os quartis (Q1, Q2 e Q3).

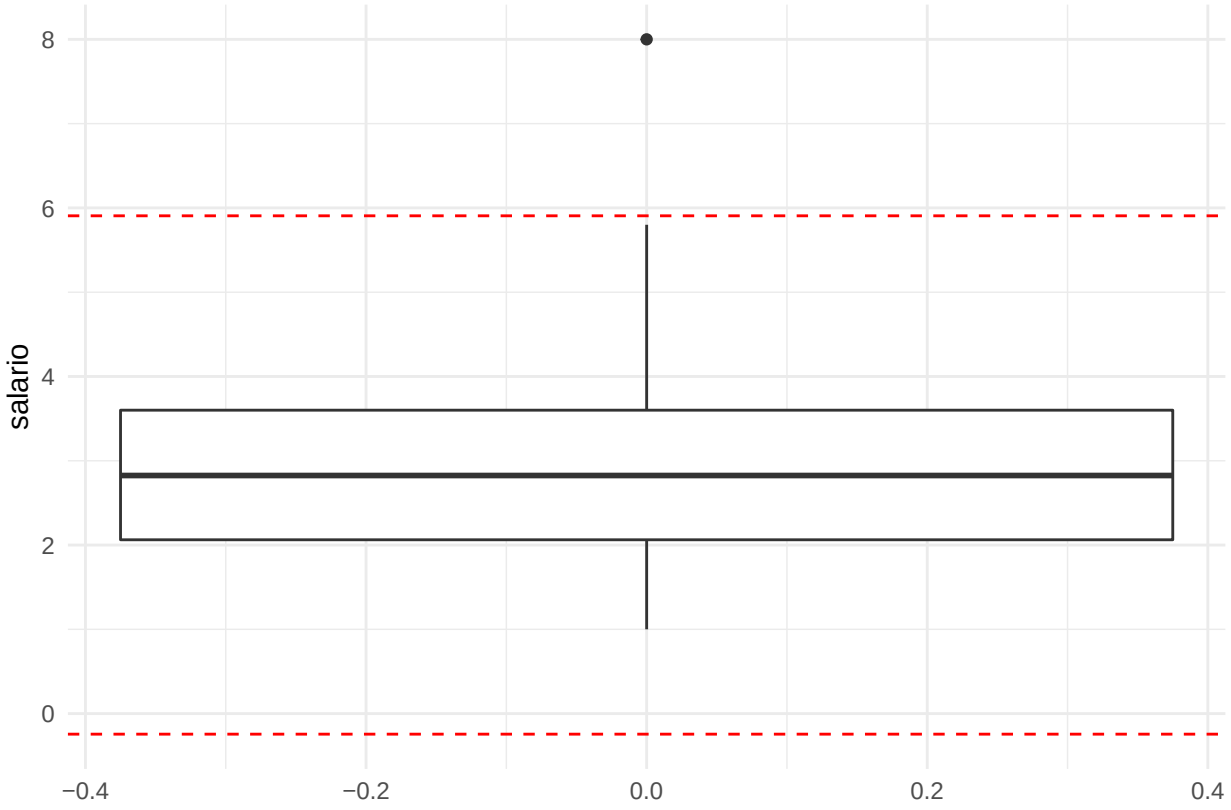
Da parte de cima do retângulo, segue uma linha até o ponto mais remoto que não exceda o **limite superior (LS)**, calculado por: $LS = q3 + (1,5)IQR$. De modo similar, da parte de baixo do retângulo, segue uma linha até o ponto mais remoto que não exceda o **limite inferior (LI)**, calculado por: $LI = q1\check{}(1,5)IQR$.

Os valores compreendidos entre esses dois limites são chamados **valores adjacentes**. As observações que estiverem acima do limite superior ou abaixo do limite inferior são chamados de **pontos exteriores** (valores atípicos ou *outliers*), sendo representados por asteriscos ou pontos no diagrama.

```
# Limite Superior
LS <- quantile(dados$salario, 0.75) + 1.5*IQR(dados$salario)

# Limite Inferior
LI <- quantile(dados$salario, 0.25) - 1.5*IQR(dados$salario)

# Boxplot
ggplot(data = dados,
       aes(y = salario))+
  geom_boxplot()+
  geom_hline(aes(yintercept = LS), color = "red", linetype = 2)+
  geom_hline(aes(yintercept = LI), color = "red", linetype = 2)+
  labs(x = "")+
  theme_minimal()
```




### 3.2.3 Variância e Desvio Padrão

Diferentemente das medidas de dispersão vistas anteriormente, as quais mostravam as dispersões dos dados em **determinadas posições** quando se **ordenava** os dados, o desvio padrão e a variância levam em conta todos os valores do conjunto de dados.

O desvio padrão fornece uma medida de dispersão ao redor da média. Quanto maior o valor, mais dispersos estarão os dados ao redor da média (menos homogêneos são os dados), e quanto menor o desvio padrão, mais concentrados estarão ao redor da média (mais homogêneos são os dados). Seu cálculo é obtido a partir da raiz quadrada da variância ($s^2$), esta obtida da seguinte maneira:

- Variância ($s^2$)

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

- Desvio Padrão ($s$)

$$s = \sqrt{s^2}$$

No R, utiliza-se a função `sd()` para calcular o desvio padrão e `var()` para cálculo de variância.

```r
# Variância
var(dados$idade)
```

```
## [1] 42.61609
```

```r
var(dados$altura)
```

```
## [1] 0.008872299
```

```r
# Desvio padrão
sd(dados$idade)
```

```
## [1] 6.5281
```

```r
sd(dados$altura)
```

```
## [1] 0.09419288
```

Para calcular o desvio padrão de multiplas variáveis, usa-se a função `lapply`, com o segundo argumento indicando a devida operação (`sd` ou `var`, no caso).

```r
# Variância múltipla
lapply(dados[, 4:8], var)
```

```
## $filhos
## [1] 1.154023
##
## $idade
## [1] 42.61609
##
## $altura
## [1] 0.008872299
##
## $peso
## [1] 265.0989
##
## $salario
## [1] 2.013851
```

```r
# Desvio padrão múltiplo
lapply(dados[, 4:8], sd)
```

```
## $filhos
## [1] 1.074255
##
## $idade
## [1] 6.5281
##
## $altura
## [1] 0.09419288
```

```
##
## $peso
## [1] 16.28186
##
## $salario
## [1] 1.419102
```

Sendo assim, o desvio padrão indica, em média, qual será o desvio (erro) que se comete ao substituir cada observação pela medida-resumo adotada - no caso, substituir pela média. Ainda, se apresenta como uma boa medida para comparar variáveis com mesma unidade e médias próximas (distribuição homogênea/normal).

### 3.2.4 Desvio Absoluto Médio e Mediano

Análogo ao desvio padrão quanto às propriedades. Consiste na somatória dos desvios, em valores absolutos, em relação à uma medida de tendência central, comumente a média, mas também a mediana.

- Desvio Absoluto Médio

$$dm(x) = \frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{x}|$$

- Desvio Absoluto Mediano

$$dm(x) = \frac{1}{n}\sum_{i=1}^{n}|x_i - md|$$

No R, utilizamos a função `mad` para realizar o cálculo. Podendo selecionar a medida de tendência central desejada a partir do argumento `center =`.

```
# Desvio Absoluto Médio
mad(dados$salario, center = mean(dados$salario))
```

```
## [1] 1.30963
```

```
# Desvio Absoluto Mediano
mad(dados$salario, center = median(dados$salario))
```

```
## [1] 1.18608
```

### 3.2.5 Escore padrão

O escore padrão informa quantos desvios padrões ($s$) um dado valor ($x_i$) está distante da média ($\bar{x}$), sendo valores positivos acima da média e valores negativos, abaixo da média. Portanto, o escore padrão nos fornece um parâmetro para comparar valores de diferentes conjunto de dados, mesmo que estes apresentem diferentes médias e desvios padrões.

$$z_i = \frac{x_i - \bar{x}}{s}$$

No R, utilizamos a função `scale()` para calcular o escore padrão das observações.

```r
# Escore padrão da variável "salario"
score_salario <- dados %>%
  select(salario) %>%
  summarise(salario_ord = sort(salario))%>%
  mutate(escore_padrao = scale(salario_ord) %>% round(2))

head(score_salario)
```

```
## # A tibble: 6 x 2
##   salario_ord escore_padrao[,1]
##         <dbl>             <dbl>
## 1        1                -1.43
## 2        1.45             -1.12
## 3        1.6              -1.01
## 4        1.7              -0.94
## 5        1.85             -0.83
## 6        1.85             -0.83
```

```r
tail(score_salario)
```

```
## # A tibble: 6 x 2
##   salario_ord escore_padrao[,1]
##         <dbl>             <dbl>
## 1         4                 0.68
## 2         4.1               0.75
## 3         4.3               0.89
## 4         4.7               1.17
## 5         5.8               1.95
## 6         8                 3.5
```

### 3.2.6 Coeficiente de Variação (CV)

O coeficiente de variação (CV) expressa, em porcentagem, a variabilidade dos dados em relação à média. Por ser um percentual, permite a comparação entre variáveis de ordem de grandezas diferentes. Assim, quanto menor o CV, menor a dispersão (mais homogêneo serão os dados).

$$CV = \frac{s}{\bar{x}}.100$$

```r
# Coeficiente de variação (CV)
sd(dados$salario) / mean(dados$salario) * 100
```

```
## [1] 46.78358
```

## 3.3 Resumo

Temos a disposição algumas funções que nos retornam diversas medidas-resumo em um único comando. Demonstraremos as funções `summary()`, `by()`, `summarytools::aggregate` e `summarytools::dfSummary`.

- Função `summary()`

```
summary(dados)
```

```
##      sexo           grau_instrucao        cidade              filhos
##  Length:30          Length:30          Length:30          Min.   :0.0000
##  Class :character   Class :character   Class :character   1st Qu.:0.0000
##  Mode  :character   Mode  :character   Mode  :character   Median :0.5000
##                                                           Mean   :0.8667
##                                                           3rd Qu.:1.7500
##                                                           Max.   :4.0000
##      idade            altura           peso            salario
##  Min.   :20.00   Min.   :1.510   Min.   : 48.00   Min.   :1.000
##  1st Qu.:30.00   1st Qu.:1.643   1st Qu.: 60.00   1st Qu.:2.062
##  Median :34.00   Median :1.730   Median : 69.00   Median :2.825
##  Mean   :33.93   Mean   :1.714   Mean   : 72.27   Mean   :3.033
##  3rd Qu.:39.50   3rd Qu.:1.780   3rd Qu.: 87.75   3rd Qu.:3.600
##  Max.   :46.00   Max.   :1.900   Max.   :110.00   Max.   :8.000
```

```
summary(dados$filhos)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.5000  0.8667  1.7500  4.0000
```

```
summary(dados$grau_instrucao)
```

```
##    Length     Class      Mode
##        30 character character
```

- Função `by()`

```
# Agrupa as medidas-resumo a partir de uma categoria
by(dados, dados$sexo, summary)
```

```
## dados$sexo: F
##      sexo           grau_instrucao        cidade              filhos
##  Length:13          Length:13          Length:13          Min.   :0.0000
##  Class :character   Class :character   Class :character   1st Qu.:0.0000
##  Mode  :character   Mode  :character   Mode  :character   Median :1.0000
##                                                           Mean   :0.8462
##                                                           3rd Qu.:1.0000
##                                                           Max.   :3.0000
##      idade            altura           peso           salario
##  Min.   :25.00   Min.   :1.510   Min.   :48.00   Min.   :1.000
##  1st Qu.:30.00   1st Qu.:1.600   1st Qu.:53.00   1st Qu.:2.300
##  Median :34.00   Median :1.640   Median :60.00   Median :2.900
##  Mean   :34.23   Mean   :1.636   Mean   :60.38   Mean   :2.965
##  3rd Qu.:40.00   3rd Qu.:1.670   3rd Qu.:65.00   3rd Qu.:3.450
##  Max.   :46.00   Max.   :1.740   Max.   :85.00   Max.   :5.800
## ------------------------------------------------------------
## dados$sexo: M
##      sexo           grau_instrucao        cidade              filhos
```

```
##  Length:17          Length:17          Length:17          Min.   :0.0000
##  Class :character   Class :character   Class :character   1st Qu.:0.0000
##  Mode  :character   Mode  :character   Mode  :character   Median :0.0000
##                                                           Mean   :0.8824
##                                                           3rd Qu.:2.0000
##                                                           Max.   :4.0000
##      idade           altura           peso           salario
##  Min.   :20.00   Min.   :1.670   Min.   : 55.00   Min.   :1.450
##  1st Qu.:30.00   1st Qu.:1.730   1st Qu.: 71.00   1st Qu.:1.900
##  Median :34.00   Median :1.750   Median : 87.00   Median :2.750
##  Mean   :33.71   Mean   :1.773   Mean   : 81.35   Mean   :3.085
##  3rd Qu.:38.00   3rd Qu.:1.830   3rd Qu.: 90.00   3rd Qu.:3.650
##  Max.   :43.00   Max.   :1.900   Max.   :110.00   Max.   :8.000
```

- Função `summarytools::aggregate()`

```
# Agrupa as medidas-resumo a partir de mais de uma categoria
aggregate(data = dados,
          cbind(peso, altura) ~ sexo + grau_instrucao,
          mean)
```

```
##   sexo  grau_instrucao     peso   altura
## 1    F Ens Fundamental 53.00000 1.640000
## 2    M Ens Fundamental 90.20000 1.802000
## 3    F    Ensino Médio 64.50000 1.705000
## 4    M    Ensino Médio 78.00000 1.778333
## 5    F        Superior 59.25000 1.601250
## 6    M        Superior 77.33333 1.743333
```

- Função `summarytools::dfSummary()`

```
dfSummary(dados) %>% view()
# Rode este exemplo em seu computador
```

# 4 Regressão e Correlação

Os modelos de regressão permitem estudar a relação entre duas ou mais variáveis, sendo X a(s) variável(is) independente(s) e Y, dependente. A análise de regressão fornece uma equação matemática que descreve a natureza do relacionamento entre variáveis, permitindo que sejam feitas previsões dos valores de uma delas em função dos valores das outras. Quando envolve somente duas variáveis, é chamada de **Análise de Regressão Simples**. Por outro lado, quando envolve mais de duas variáveis, é denominada **Análise de Regressão Múltipla**. Neste material, trataremos apenas da **Análise de Regressão Linear Simples**, descrita pela seguinte equação simplificada:

$$Y = \beta_0 + \beta_1 X$$

Sendo $\beta_0$ o intercepto e $\beta_1$, o coeficiente angular.

Uma vez que se deseja associar duas variáveis, espera-se identificar o **grau de associação** entre as variáveis, ou seja, o grau de dependência entre elas (se as variáveis são dependentes ou não), de modo a compreender melhor o resultado de uma variável quando conhecemos a realização da outra. Contudo, vale salientar que

a associação estatística entre variáveis não significa, necessariamente, que há uma relação direta de causa e efeito.

Para analisarmos a associação entre duas variáveis quantitativas, além das distribuições de dados em tabelas de frequência, podemos confeccionar um **gráfico de dispersão**, que nos permite observar a presença ou ausência de correlação, além de suas características.

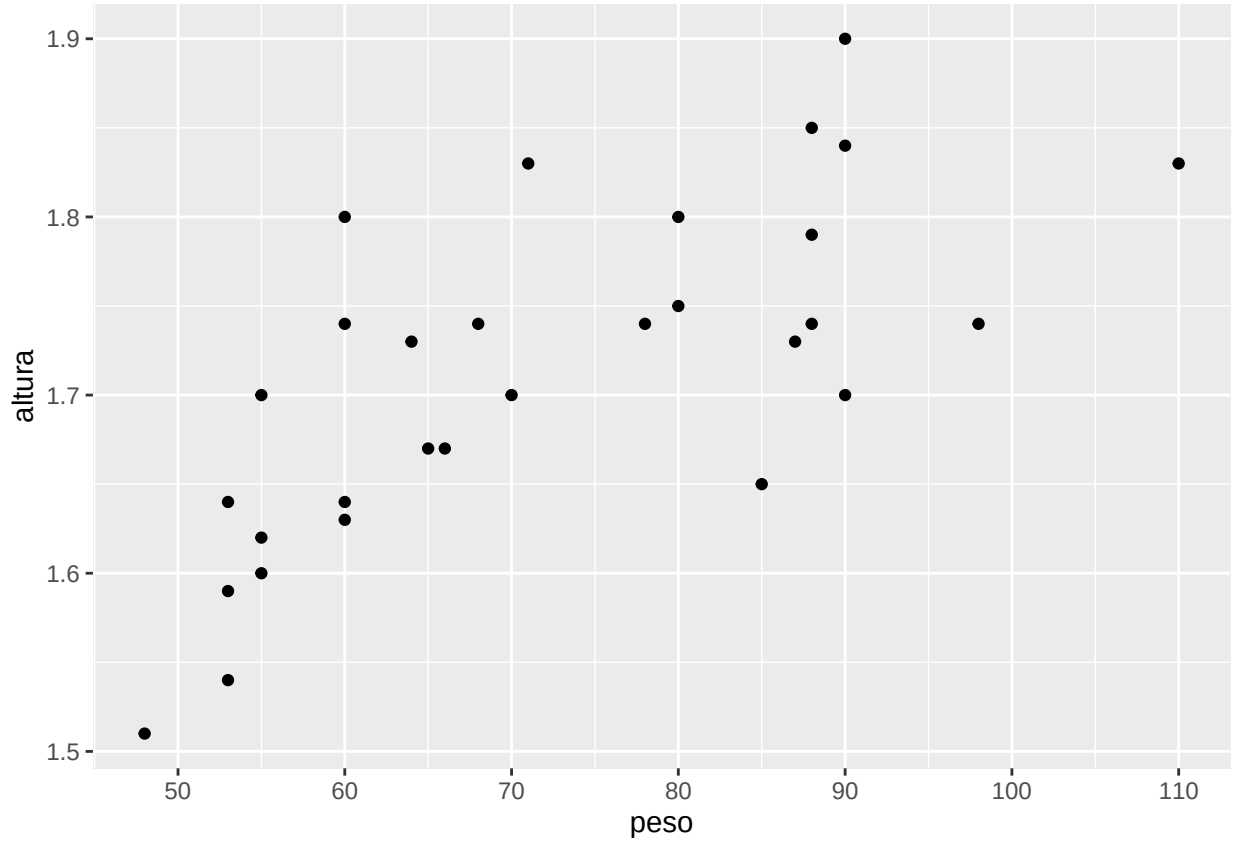
Caso seja constatada uma relação linear entre as variáveis, podemos realizar a **Análise de Correlação Linear** e a **Análise de Regressão**. Em ambas as análises, visa-se verificar **SE** e **COMO** as variáveis quantitativas se relacionam.

Uma das suposições básicas de ambas as análises é a de uma hipótese acerca da relação de dependência entre variáveis, ou seja, permite identificar variáveis dependentes e independentes, além de uma associação ser positiva ou negativa e em que grau.

## 4.1 Gráfico de dispersão

A seguir, construiremos um gráfico de dispersão da altura em relação ao peso:

```
dados %>%
  ggplot(aes(x = peso, y = altura))+
  geom_point()+
  scale_x_continuous(breaks = seq(50, 110, 10))
```




Através do diagrama de dispersão é possível ter uma idéia inicial de como as variáveis estão relacionadas:

- **Direção da correlação:** isto é, o que ocorre com os valores de Y (aumentam ou diminuem) quando os valores de X aumentam;

- **Força da correlação:** em que "taxa" (inclinação da reta - $\beta_1$) os valores de Y aumentam ou diminuem em função de X;
- **Natureza da correlação:** se é possível ajustar uma reta, parábola, exponencial, etc., aos pontos.

A imagem a seguir ilustra os tipos de direção da correlação linear.

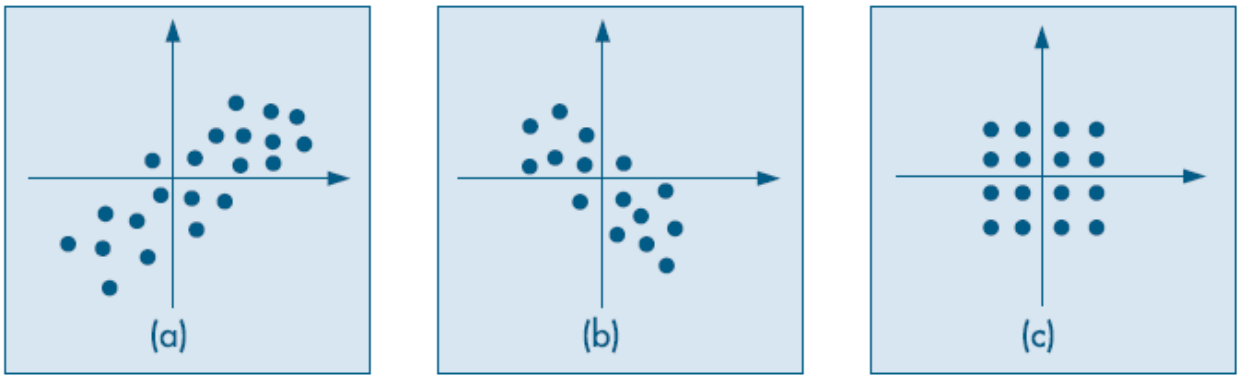



Figure 1: Fonte: Bussab e Morettin (2010).

**(a) Correlação Linear Positiva:** à medida que a variável X aumenta, os valores de Y também tendem a aumentar, sendo possível ajustar uma reta crescente que passe entre os pontos;

**(b) Correlação Linear Negativa:** à medida que a variável X aumenta, os valores de Y tendem a diminuir, sendo possível ajustar uma reta decrescente que passe entre os pontos;

**(c) Correlação não linear:** no caso do exemplo, não existe correlação, pois para cada resultado positivo, tem-se um resultado negativo simétrico, anulando-se na soma, não apresentando correlação linear.

Portanto, através do diagrama de dispersão é possível identificar se há correlação linear, e se a correlação é positiva ou negativa. Quanto mais o diagrama de dispersão aproximar-se de uma reta, mais forte será a correlação linear.

## 4.2 Coeficientes da reta

No R, podemos determinar o intercepto ($\beta_0$) e o coeficiente angular ($\beta_1$) a partir da função `lm()`, declarando a variável dependente (no caso, a altura), em relação à independente (nesse caso, o peso).

```
lm(altura ~ peso, data = dados)
```

```
##
## Call:
## lm(formula = altura ~ peso, data = dados)
##
## Coefficients:
## (Intercept)         peso
##    1.419061     0.004077
```

No exemplo anterior, verificamos que o intercepto apresentou valor de 1,419061 e o coeficiente angular, 0,004077. Assim, podemos dizer que para cada unidade adicional de peso, a altura varia, positivamente, na taxa de 0,004077.

## 4.3 Correlação Linear de Pearson (r)

Para verificar a associação entre variáveis, precisamos quantificar o grau de associação entre as variáveis. Isto pode ser feito a partir dos Coeficientes de Correlação.

A **Análise de Correlação Linear** pode ser realizada a partir do cálculo do **Coeficiente de Correlação Linear de Pearson (r)**, o que permite mensurar a **direção** e a **força** da relação linear entre duas variáveis. Seja $r$ o coeficiente de correlação linear, temos a seguinte equação:

$$r = \frac{\sum_{i=1}^{n}(x_i y_i) - n\bar{x}\bar{y}}{\sqrt{\sum_{i=1}^{n}(x_i^2) - n\bar{x}^2}\sqrt{\sum_{i=1}^{n}(y_i^2) - n\bar{y}^2}}$$

O coeficiente de correlação linear de Pearson varia de -1 a +1 e é adimensional:

- **r = -1**: significa que há uma correlação linear negativa perfeita entre as variáveis;
- **r = +1**: significa que há uma correlação linear positiva perfeita entre as variáveis;
- **r = 0**: significa que não há correlação linear entre as variáveis.

Novamente, um alto coeficiente de correlação linear de Pearson (próximo a +1 ou a -1) não significa uma relação de causa e efeito entre as variáveis, mas apenas que as duas variáveis apresentam uma tendência de variação conjunta.

No R, calculamos a correlação a partir da função `cor()`, declarando como argumentos a variável dependente (y) e independente (x).

```
cor(y = dados$altura, x = dados$peso)
```

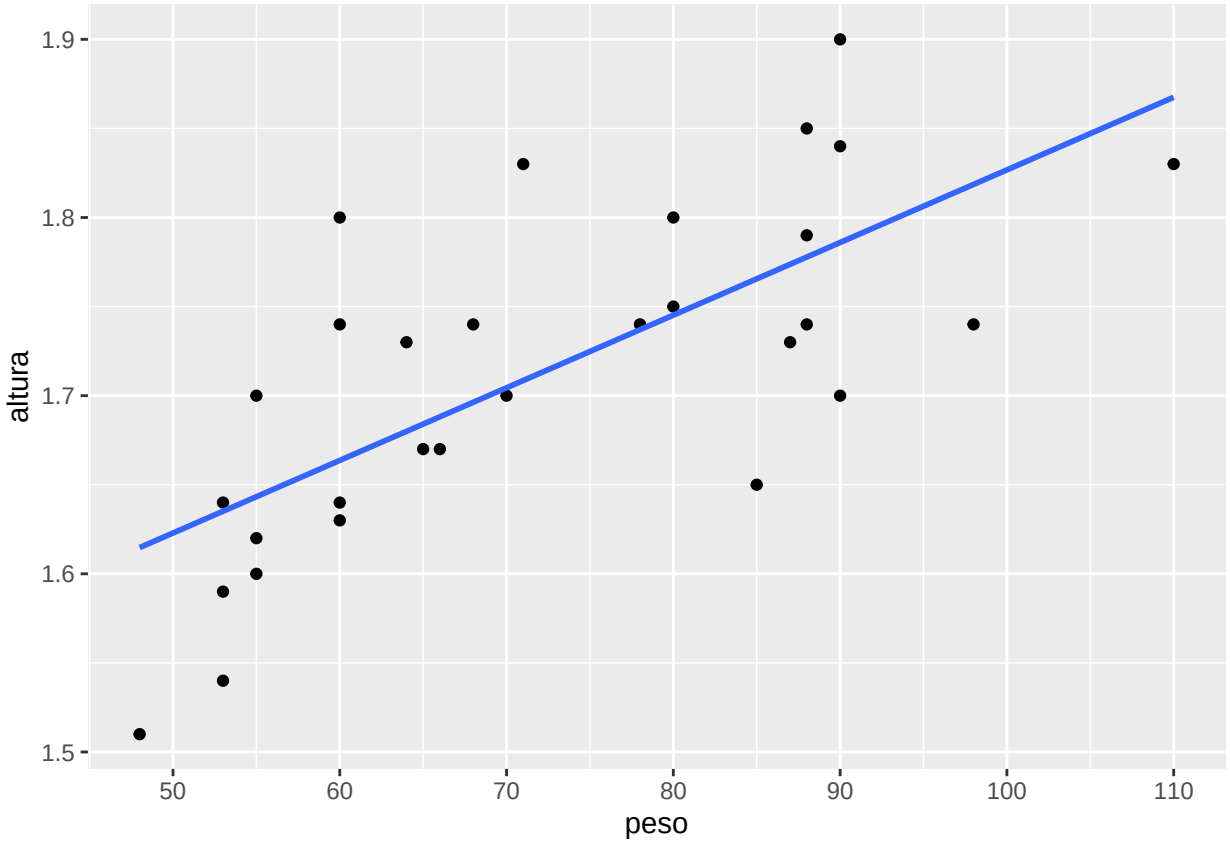
```
## [1] 0.704674
```

No exemplo anterior, obtivemos um valor de $r = 0,705$, portanto, apresentou uma significativa correlação linear positiva entre as variáveis.

## 4.4 Regressão linear simples

No R, utilizamos a função `geom_smooth()` como camada adicional da `ggplot()` para criar a regressão linear simples, sendo calculado, automaticamente, todos os parâmetros necessários para sua construção.

```
dados %>%
  ggplot(aes(x = peso, y = altura))+
  geom_point()+
  geom_smooth(formula = y ~ x, method = "lm", se = FALSE)+
  scale_x_continuous(breaks = seq(50, 110, 10))
```




## Bibliografia consultada

ANJOS, A.. **Estatística Básica com uso do software R**. Universidade Federal do Paraná, 2014. https://docs.ufpr.br/~aanjos/CE002A/estdescritiva.pdf.

BUSSAB, W.O.; MORETTIN, P.A.. Análise exploratória de dados. In: BUSSAB, W.O.; MORETTIN, P.A.. **Estatística Básica**. 6.ed. São Paulo: Saraiva, 2010. Cap. 1. p. 9-102.

FREIRE, S.M.. **Bioestatística Básica**. Universidade do Estado do Rio de Janeiro, 2021. http://www.lampada.uerj.br/arquivosdb/_book/medidasTendenciaDispersao.html#fig:variaveisMultimodais.

REIS, E.A.; REIS, I.A.. **Análise Descritiva de Dados**. Universidade Federal de Minas Gerais, 2002. http://www.est.ufmg.br/portal/arquivos/rts/rte0202.pdf.

SOETEWEY, A.. **Descriptive statistics by hand**. 2020. https://statsandr.com/blog/descriptive-statistics-by-hand/.

SOETEWEY, A.. **Descriptive statistics in R**. 2020. https://statsandr.com/blog/descriptive-statistics-in-r/.

TIERNEY, N.. **RMarkdown for Scientists**. 2020. https://rmd4sci.njtierney.com/math.