Curso 02

Helena

07/12/2021

No GIT: https://github.com/HelenaDEspindula/Omega_Course_02.git

Leitura de dados (tidyverse)

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidyverse)
ls("package:tidyverse")
```

```
## [1] "tidyverse_conflicts" "tidyverse_deps"      "tidyverse_logo"
## [4] "tidyverse_packages"  "tidyverse_sitrep"    "tidyverse_update"
```

```
tb <-
  read_csv2(file = "http://leg.ufpr.br/~walmes/data/renda-idh-alfab.csv",
            comment = "#")
```

```
## i Using "’,’" as decimal and "’.’" as grouping mark. Use ‘read_delim()‘ for more control.

## New names:
## * ‘‘ -> ...7

## Rows: 5592 Columns: 7

## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr (2): ???Sigla, Município
## dbl (3): Código, renda, alfab
## lgl (2): idh, ...7
```

```
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
str(tb)
```

```
## spec_tbl_df [5,592 x 7] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ???Sigla : chr [1:5592] "AC" "AC" "AC" "AC" ...
##  $ Código   : num [1:5592] 1200013 1200054 1200104 1200138 1200179 ...
##  $ Município: chr [1:5592] "Acrelândia" "Assis Brasil" "Brasiléia" "Bujari" ...
##  $ idh      : logi [1:5592] NA NA NA NA NA NA ...
##  $ renda    : num [1:5592] 137 115 132 119 108 ...
##  $ alfab    : num [1:5592] 73.3 71 75.5 61.2 62.3 ...
##  $ ...7     : logi [1:5592] NA NA NA NA NA NA ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ‘???Sigla‘ = col_character(),
##   ..   Código = col_double(),
##   ..   Município = col_character(),
##   ..   idh = col_logical(),
##   ..   renda = col_double(),
##   ..   alfab = col_double(),
##   ..   ...7 = col_logical()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
tb_gp <-
  read_tsv(file = "http://leg.ufpr.br/~walmes/data/allgrandprix.txt",
           locale = locale(date_names = "en"))
```

```
## Rows: 22240 Columns: 8

## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (6): circuito, data, corredor, equipe, tempo, pontos
## dbl (2): rank, polip

##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
str(tb_gp)
```

```
## spec_tbl_df [22,240 x 8] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ circuito: chr [1:22240] "Monza" "Monza" "Monza" "Monza" ...
##  $ data    : chr [1:22240] "03 September 1950" "03 September 1950" "03 September 1950" "03 September
##  $ rank    : num [1:22240] 1 2 2 3 4 5 6 7 8 9 ...
##  $ corredor: chr [1:22240] "Giuseppe Farina" "Dorino Serafini" "Alberto Ascari" "Luigi Fagioli" ...
##  $ equipe  : chr [1:22240] "Alfa Romeo" "Ferrari" "Ferrari" "Alfa Romeo" ...
##  $ tempo   : chr [1:22240] "2:51:17.400" "-01:18.600" "-01:18.600" "-01:35.600" ...
##  $ polip   : num [1:22240] 3 6 6 3 13 16 17 18 24 21 ...
##  $ pontos  : chr [1:22240] "8" "3" "3" "4" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   circuito = col_character(),
##   ..   data = col_character(),
##   ..   rank = col_double(),
##   ..   corredor = col_character(),
##   ..   equipe = col_character(),
##   ..   tempo = col_character(),
##   ..   polip = col_double(),
##   ..   pontos = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
tb_ss <-
  read_fwf(file = "saosilvestre_fwf.txt",
           col_positions = fwf_widths(c(5, 6, 30, 100), col_names = letters[1:4]))
```

```
## Rows: 10141 Columns: 4
```

```
## -- Column specification --------------------------------------------------------
##
## chr (2): c, d
## dbl (2): a, b
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
str(tb_ss)
```

```
## spec_tbl_df [10,141 x 4] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ a: num [1:10141] 1 2 3 4 5 6 7 8 9 10 ...
##  $ b: num [1:10141] 2 6 4 7 12 1 134 25 14 20 ...
##  $ c: chr [1:10141] "PAUL TERGAT" "HENDRICK RAMAALA" "ELIJAH KIPTARBEI LAGAT" "SILVIO GUERRA" ...
##  $ d: chr [1:10141] "(69)     M2529 0:44:47         FILA                          (KENYA
##  - attr(*, "spec")=
##   .. cols(
##   ..   a = col_double(),
##   ..   b = col_double(),
##   ..   c = col_character(),
##   ..   d = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
url <- "http://leg.ufpr.br/~walmes/data/euro_football_players.txt"

tb_efp <- read_tsv(file = url, comment = "#", col_names =  TRUE)
```

```
## Rows: 1528 Columns: 17
## -- Column specification -------------------------------------------------------
## Delimiter: "\t"
## chr  (5): country, team, name, pos, apps
## dbl (12): age, cm, kg, goal, ass, yel, red, spg, ps, aw, mom, rt
```

```
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
glimpse(tb_efp)
```

```
## Rows: 1,528
## Columns: 17
## $ country <chr> "Austria", "Austria", "Austria", "Austria", "Austria", "Austri~
## $ team    <chr> "Salzburg", "Salzburg", "Salzburg", "Salzburg", "Salzburg", "S~
## $ name    <chr> "Sadio Mané", "Kevin Kampl", "Alan", "André Ramalho", "Stefan ~
## $ pos     <chr> "M(L)", "M(R)", "FW", "D(C)", "M", "M(C)", "FW", "M(C)", "D(C)~
## $ age     <dbl> 21, 23, 24, 22, 23, 28, 28, 24, 21, 28, 29, 27, 28, 23, 28, 21~
## $ cm      <dbl> 175, 180, 182, 182, 180, 172, 180, 186, 184, 175, 177, 180, 19~
## $ kg      <dbl> 69, 63, 73, 77, 71, 69, 71, 74, 78, 69, 73, 68, 90, 75, 77, 75~
## $ apps    <chr> "9", "9", "8(1)", "9", "1(1)", "7(1)", "7", "8(1)", "8(1)", "8~
## $ goal    <dbl> 4, 2, 4, 1, NA, NA, 8, NA, NA, NA, NA, NA, NA, NA, NA, NA, 1, ~
## $ ass     <dbl> 3, 4, 2, NA, NA, 1, 1, NA, NA, NA, 1, NA, NA, NA, NA, 2, NA, 1~
## $ yel     <dbl> 1, 2, NA, 4, NA, 3, 1, 1, 3, NA, 3, 1, 1, NA, NA, NA, NA, NA, ~
## $ red     <dbl> 1, NA, 1, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
## $ spg     <dbl> 2.0, 2.0, 4.2, 0.9, 0.5, 1.6, 4.1, 0.8, 1.1, 0.9, 0.5, 1.0, 0.~
## $ ps      <dbl> 77.0, 83.9, 60.8, 72.3, 86.3, 79.4, 72.8, 74.7, 69.1, 64.5, 71~
## $ aw      <dbl> 1.2, 0.3, 3.8, 3.2, 3.0, 0.5, 0.3, 4.0, 3.4, 2.8, 1.6, 1.4, 1.~
## $ mom     <dbl> 3, 1, 2, 1, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 1,~
## $ rt      <dbl> 7.98, 7.93, 7.91, 7.67, 7.59, 7.55, 7.54, 7.46, 7.38, 7.27, 7.~
```

```
str(tb_efp)
```

```
## spec_tbl_df [1,528 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ country: chr [1:1528] "Austria" "Austria" "Austria" "Austria" ...
##  $ team   : chr [1:1528] "Salzburg" "Salzburg" "Salzburg" "Salzburg" ...
##  $ name   : chr [1:1528] "Sadio Mané" "Kevin Kampl" "Alan" "André Ramalho" ...
##  $ pos    : chr [1:1528] "M(L)" "M(R)" "FW" "D(C)" ...
##  $ age    : num [1:1528] 21 23 24 22 23 28 28 24 21 28 ...
##  $ cm     : num [1:1528] 175 180 182 182 180 172 180 186 184 175 ...
##  $ kg     : num [1:1528] 69 63 73 77 71 69 71 74 78 69 ...
##  $ apps   : chr [1:1528] "9" "9" "8(1)" "9" ...
##  $ goal   : num [1:1528] 4 2 4 1 NA NA 8 NA NA NA ...
##  $ ass    : num [1:1528] 3 4 2 NA NA 1 1 NA NA NA ...
##  $ yel    : num [1:1528] 1 2 NA 4 NA 3 1 1 3 NA ...
##  $ red    : num [1:1528] 1 NA 1 NA NA NA NA NA NA NA ...
##  $ spg    : num [1:1528] 2 2 4.2 0.9 0.5 1.6 4.1 0.8 1.1 0.9 ...
##  $ ps     : num [1:1528] 77 83.9 60.8 72.3 86.3 79.4 72.8 74.7 69.1 64.5 ...
##  $ aw     : num [1:1528] 1.2 0.3 3.8 3.2 3 0.5 0.3 4 3.4 2.8 ...
##  $ mom    : num [1:1528] 3 1 2 1 NA NA NA NA NA NA ...
##  $ rt     : num [1:1528] 7.98 7.93 7.91 7.67 7.59 7.55 7.54 7.46 7.38 7.27 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   country = col_character(),
##   ..   team = col_character(),
##   ..   name = col_character(),
##   ..   pos = col_character(),
##   ..   age = col_double(),
##   ..   cm = col_double(),
##   ..   kg = col_double(),
##   ..   apps = col_character(),
##   ..   goal = col_double(),
##   ..   ass = col_double(),
##   ..   yel = col_double(),
##   ..   red = col_double(),
##   ..   spg = col_double(),
##   ..   ps = col_double(),
##   ..   aw = col_double(),
##   ..   mom = col_double(),
##   ..   rt = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
attr(tb_efp, which = "spec") <- NULL
str(tb_efp)
```

```
## spec_tbl_df [1,528 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ country: chr [1:1528] "Austria" "Austria" "Austria" "Austria" ...
##  $ team   : chr [1:1528] "Salzburg" "Salzburg" "Salzburg" "Salzburg" ...
##  $ name   : chr [1:1528] "Sadio Mané" "Kevin Kampl" "Alan" "André Ramalho" ...
##  $ pos    : chr [1:1528] "M(L)" "M(R)" "FW" "D(C)" ...
##  $ age    : num [1:1528] 21 23 24 22 23 28 28 24 21 28 ...
##  $ cm     : num [1:1528] 175 180 182 182 180 172 180 186 184 175 ...
##  $ kg     : num [1:1528] 69 63 73 77 71 69 71 74 78 69 ...
##  $ apps   : chr [1:1528] "9" "9" "8(1)" "9" ...
##  $ goal   : num [1:1528] 4 2 4 1 NA NA 8 NA NA NA ...
##  $ ass    : num [1:1528] 3 4 2 NA NA 1 1 NA NA NA ...
##  $ yel    : num [1:1528] 1 2 NA 4 NA 3 1 1 3 NA ...
##  $ red    : num [1:1528] 1 NA 1 NA NA NA NA NA NA NA ...
##  $ spg    : num [1:1528] 2 2 4.2 0.9 0.5 1.6 4.1 0.8 1.1 0.9 ...
##  $ ps     : num [1:1528] 77 83.9 60.8 72.3 86.3 79.4 72.8 74.7 69.1 64.5 ...
##  $ aw     : num [1:1528] 1.2 0.3 3.8 3.2 3 0.5 0.3 4 3.4 2.8 ...
##  $ mom    : num [1:1528] 3 1 2 1 NA NA NA NA NA NA ...
##  $ rt     : num [1:1528] 7.98 7.93 7.91 7.67 7.59 7.55 7.54 7.46 7.38 7.27 ...
##  - attr(*, "problems")=<externalptr>
```

```
head(tb_efp)
```

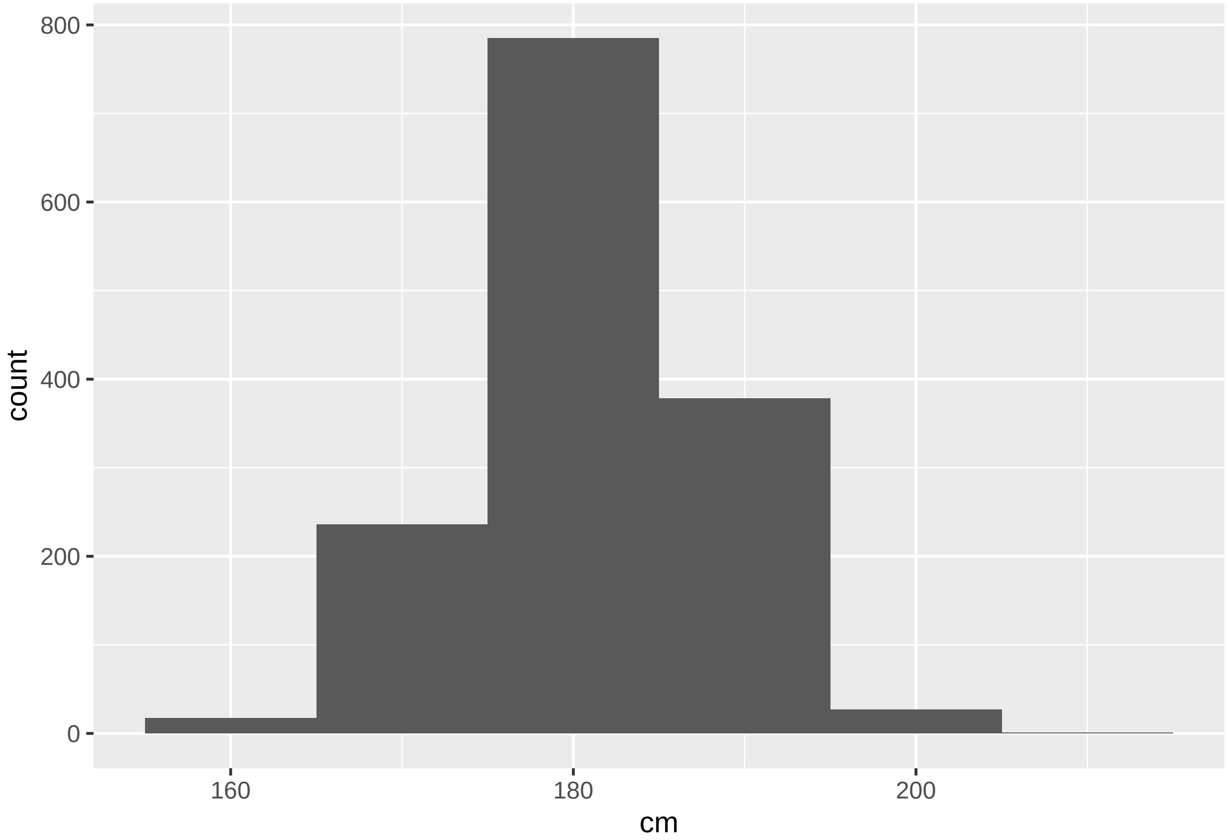
```
## # A tibble: 6 x 17
##   country team     name    pos     age    cm    kg apps   goal   ass   yel   red
##   <chr>   <chr>    <chr>   <chr> <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl>
## 1 Austria Salzburg Sadio ~ M(L)     21   175    69 9         4     3     1     1
## 2 Austria Salzburg Kevin ~ M(R)     23   180    63 9         2     4     2    NA
## 3 Austria Salzburg Alan    FW       24   182    73 8(1)      4     2    NA     1
## 4 Austria Salzburg André ~ D(C)     22   182    77 9         1    NA     4    NA
## 5 Austria Salzburg Stefan~ M        23   180    71 1(1)      NA     NA     NA     NA
## 6 Austria Salzburg Christ~ M(C)     28   172    69 7(1)      NA      1      3     NA
## # ... with 5 more variables: spg <dbl>, ps <dbl>, aw <dbl>, mom <dbl>, rt <dbl>
```

GGPLOT2

«««< HEAD Ver: https://www.rstudio.org/links/data_visualization_cheat_sheet

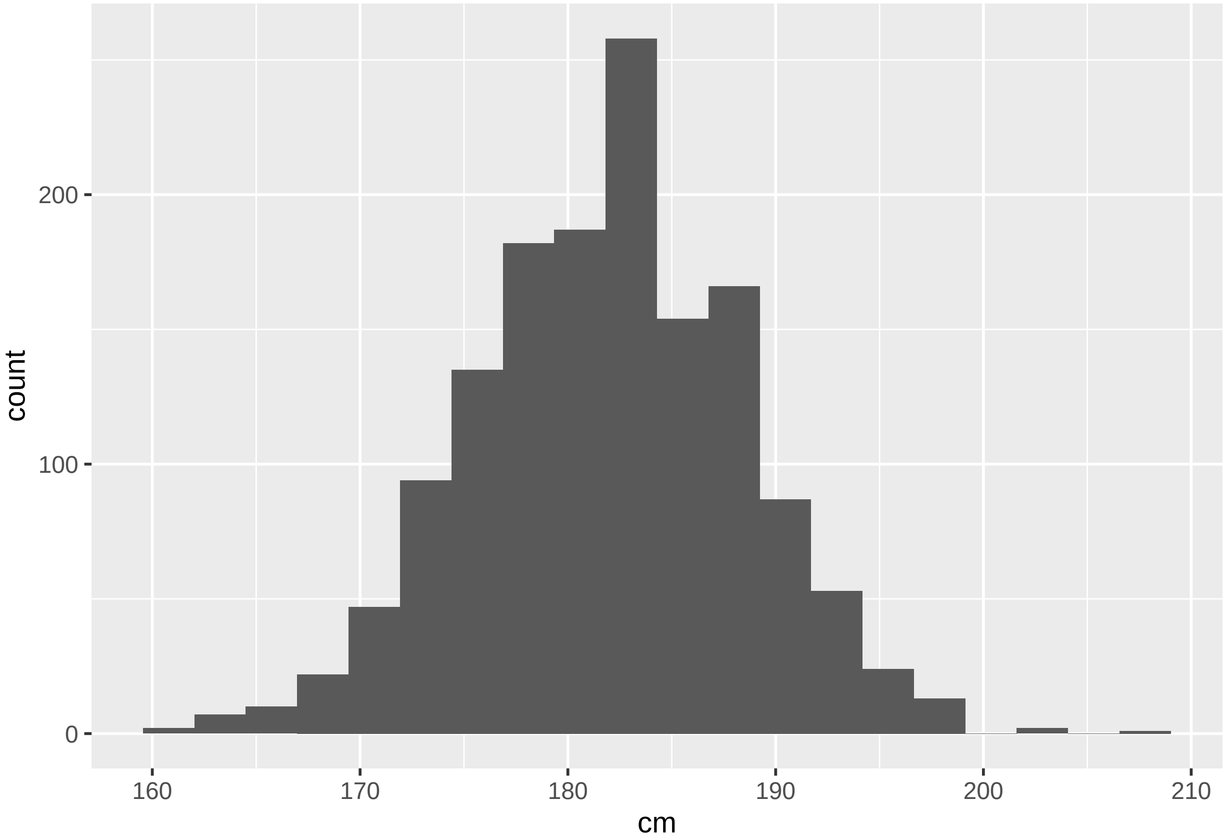
```
ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_histogram(binwidth = 10)
```

```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



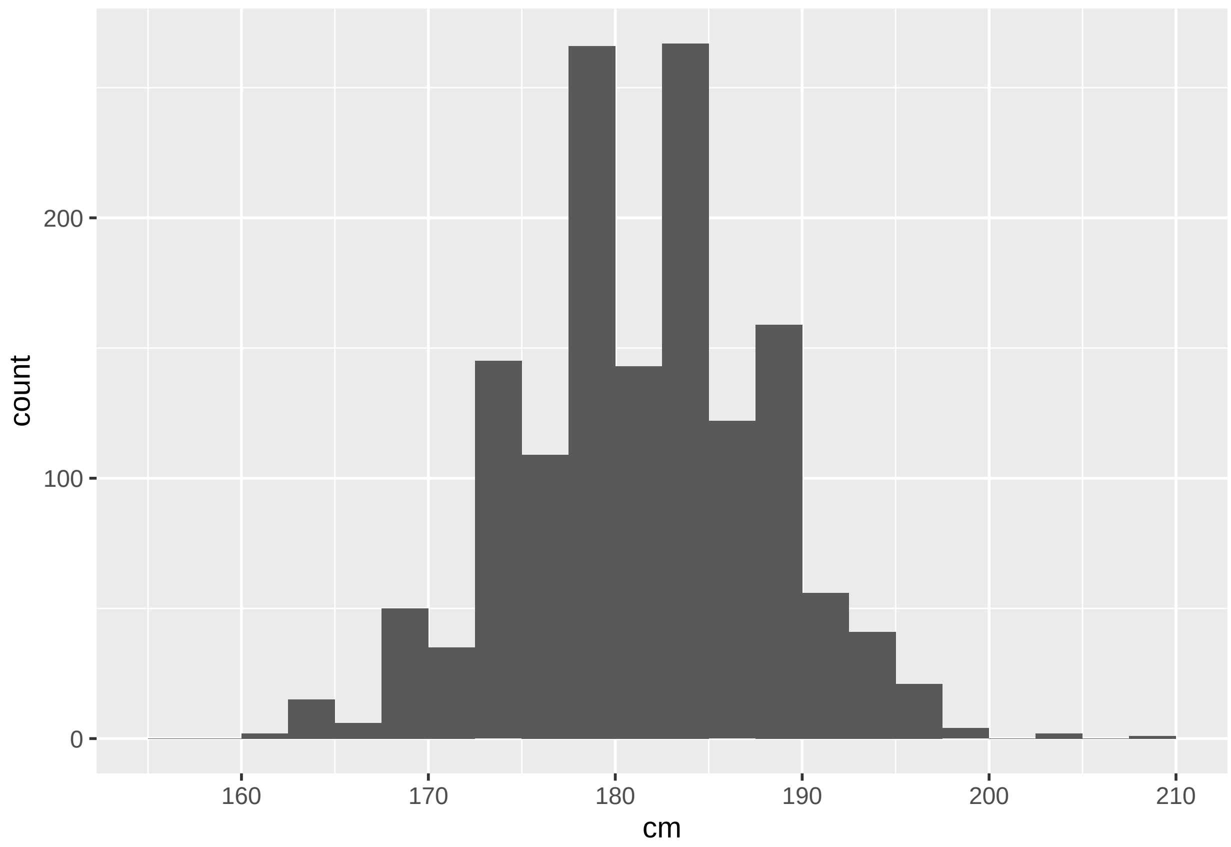
```
ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_histogram(bins = 20)
```

```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



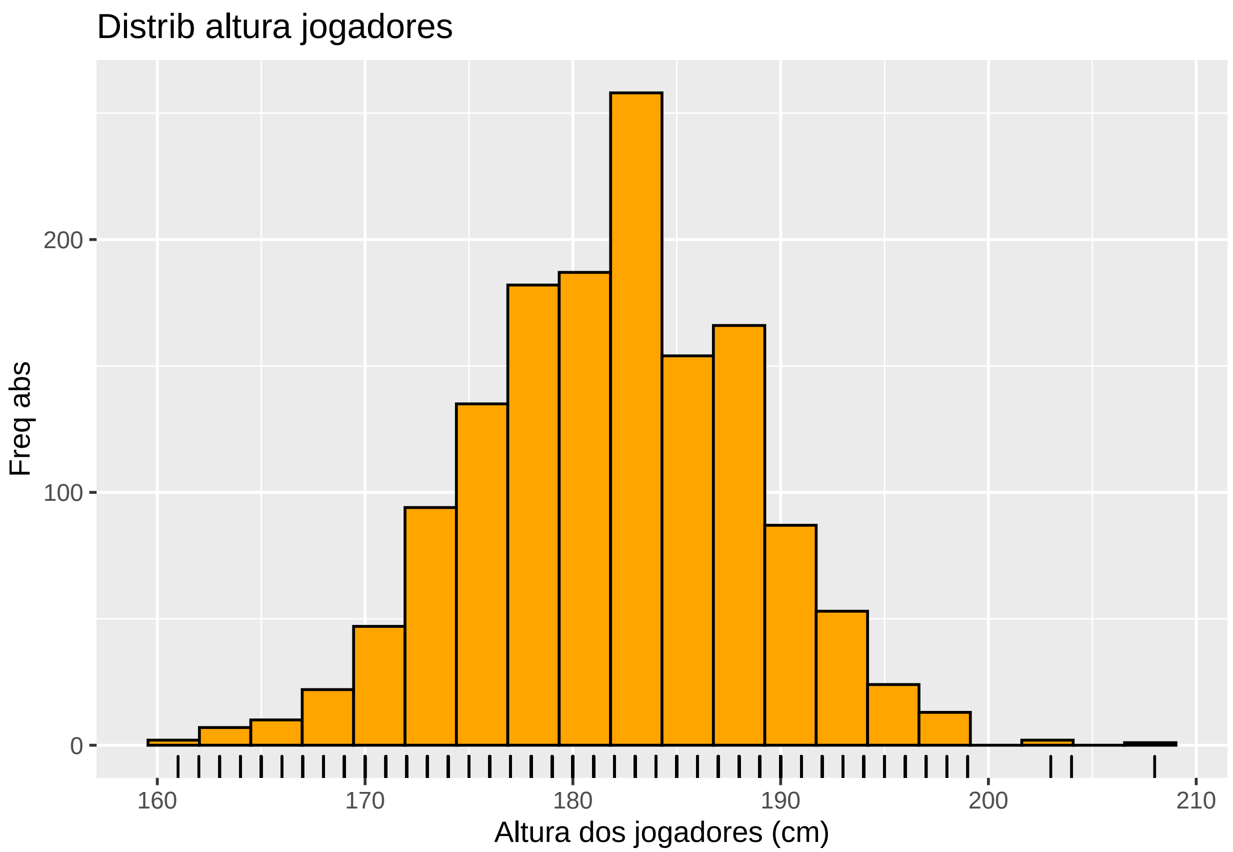
```
ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_histogram(breaks = seq(155, 210, by = 2.5))
```

```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



```
#dados
#mapeamento
#estatistica e geometria


ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_histogram(bins = 20,
                 fill = "orange",
                 color = "black") +
  geom_rug() +
  labs(x = "Altura dos jogadores (cm)",
       y = "Freq abs",
       title = "Distrib altura jogadores")
```

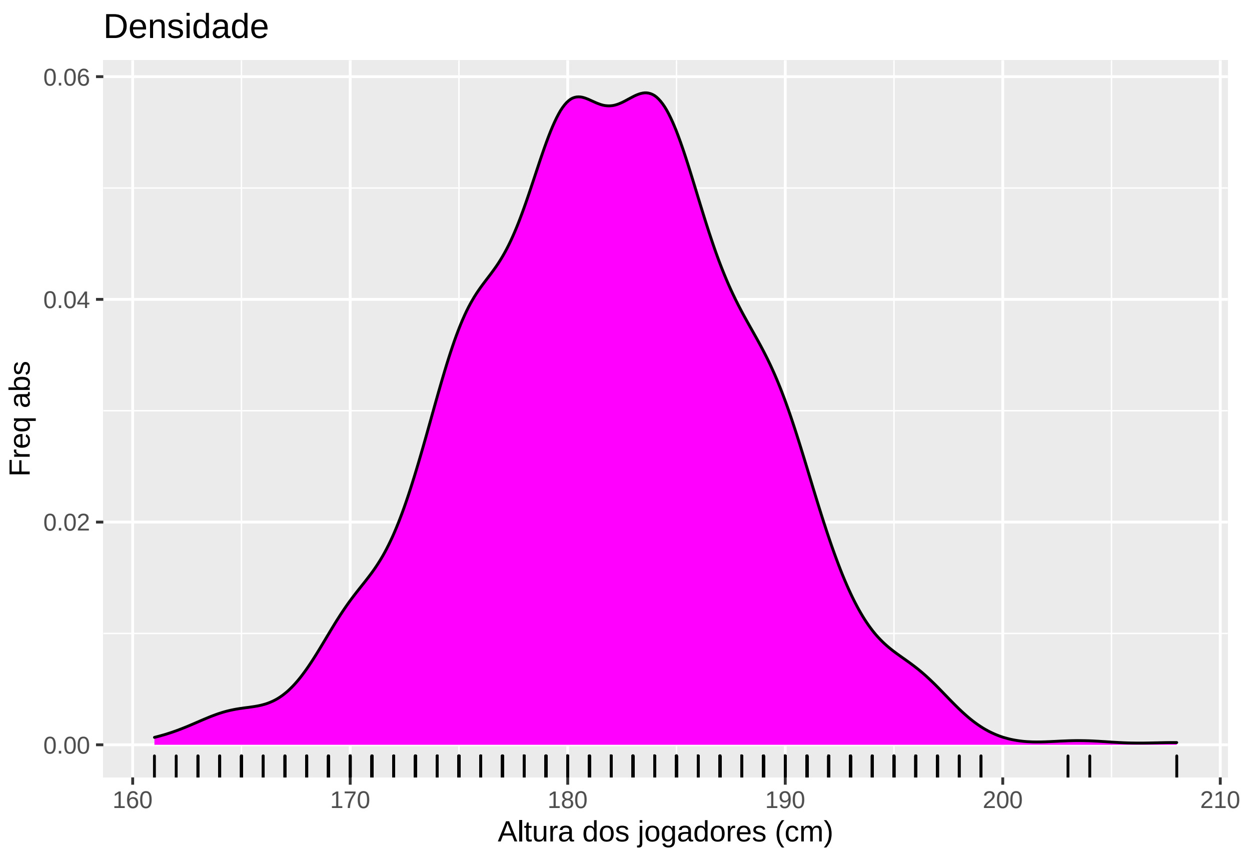
```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



```
ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_density( fill = "magenta", alPHa = 0.3) +
  geom_rug() +
  labs(x = "Altura dos jogadores (cm)",
       y = "Freq abs",
       title = "Densidade")
```

```
## Warning: Ignoring unknown parameters: alPHa
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```

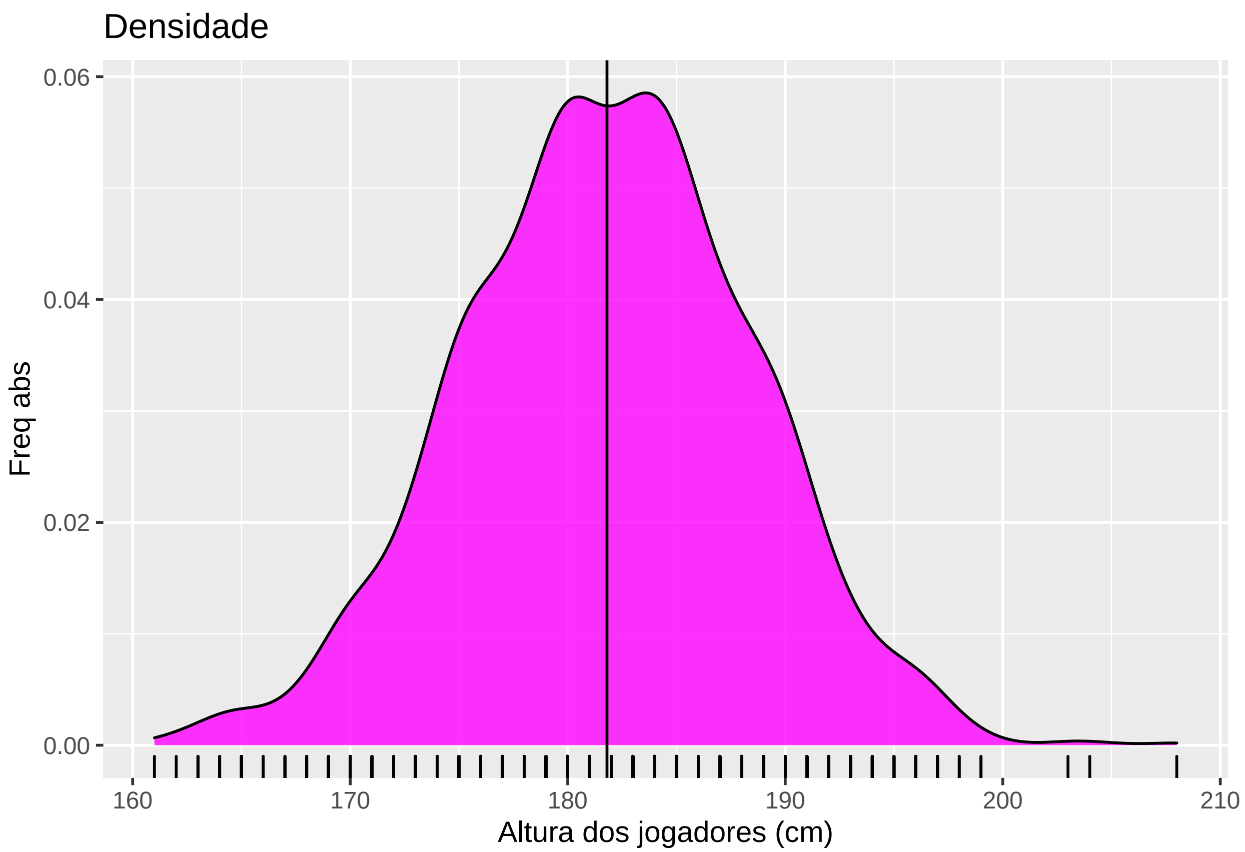


```
media_tb <- mean(tb_efp$cm, na.rm = TRUE)
media_tb
```

```
## [1] 181.8054
```

```
ggplot(data = tb_efp,
       mapping = aes(x= cm)) +
  geom_density( fill = "magenta", alpha = 0.8) +
  geom_rug() +
  geom_vline(xintercept = media_tb) +
  labs(x = "Altura dos jogadores (cm)",
       y = "Freq abs",
       title = "Densidade")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```

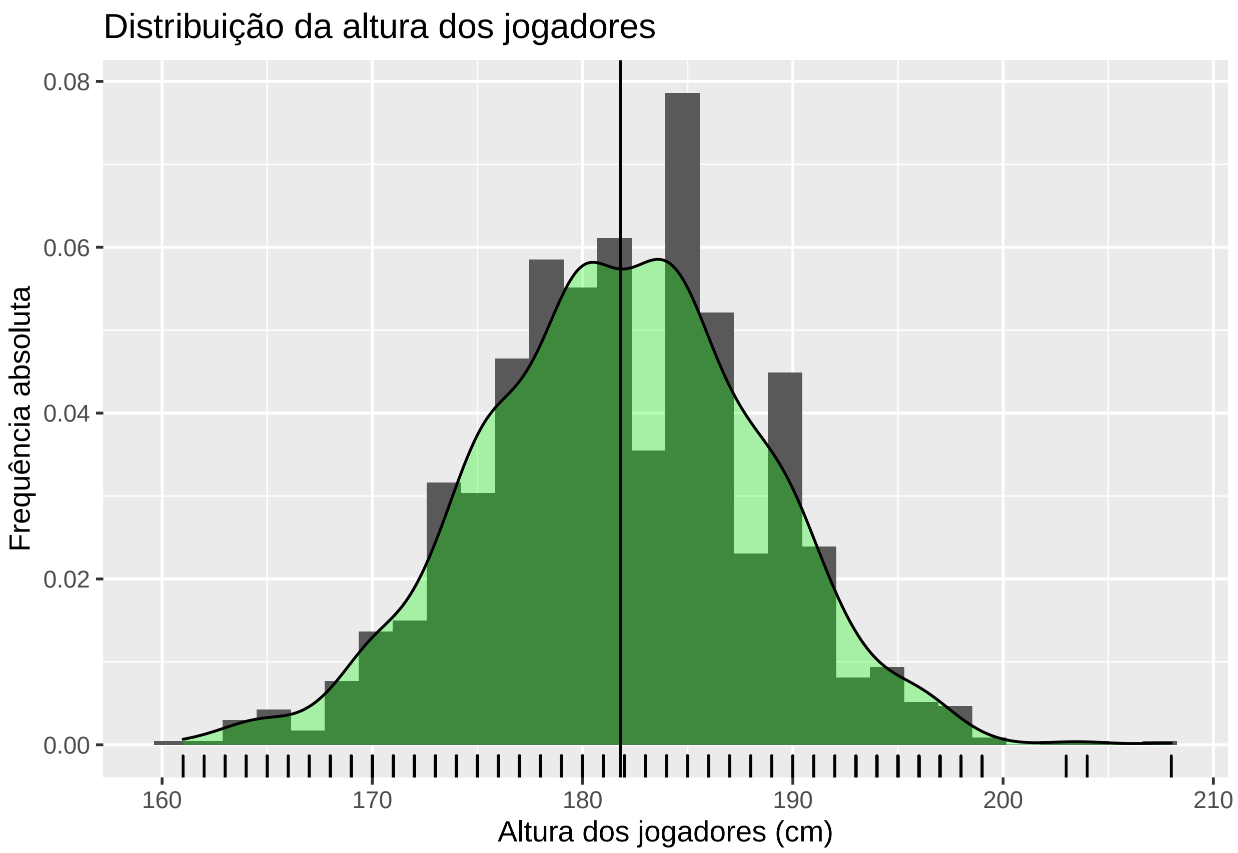


```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
    geom_histogram(mapping = aes(y = ..density..)) +
    geom_density(fill = "green", alpha = 0.3) +
    geom_rug() +
    geom_vline(xintercept = media_tb) +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```

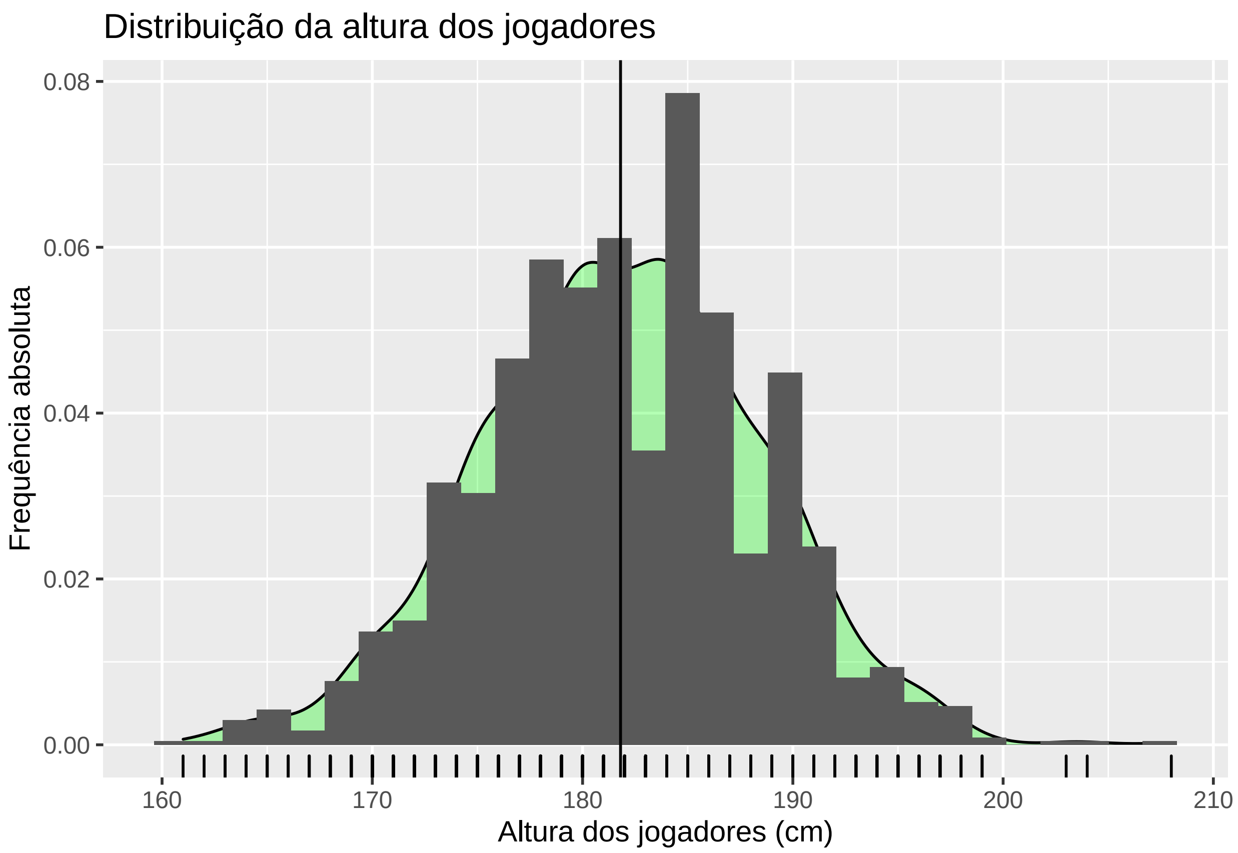


```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
  geom_density(fill = "green", alpha = 0.3) +
  geom_histogram(mapping = aes(y = ..density..)) +
    geom_rug() +
    geom_vline(xintercept = media_tb) +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

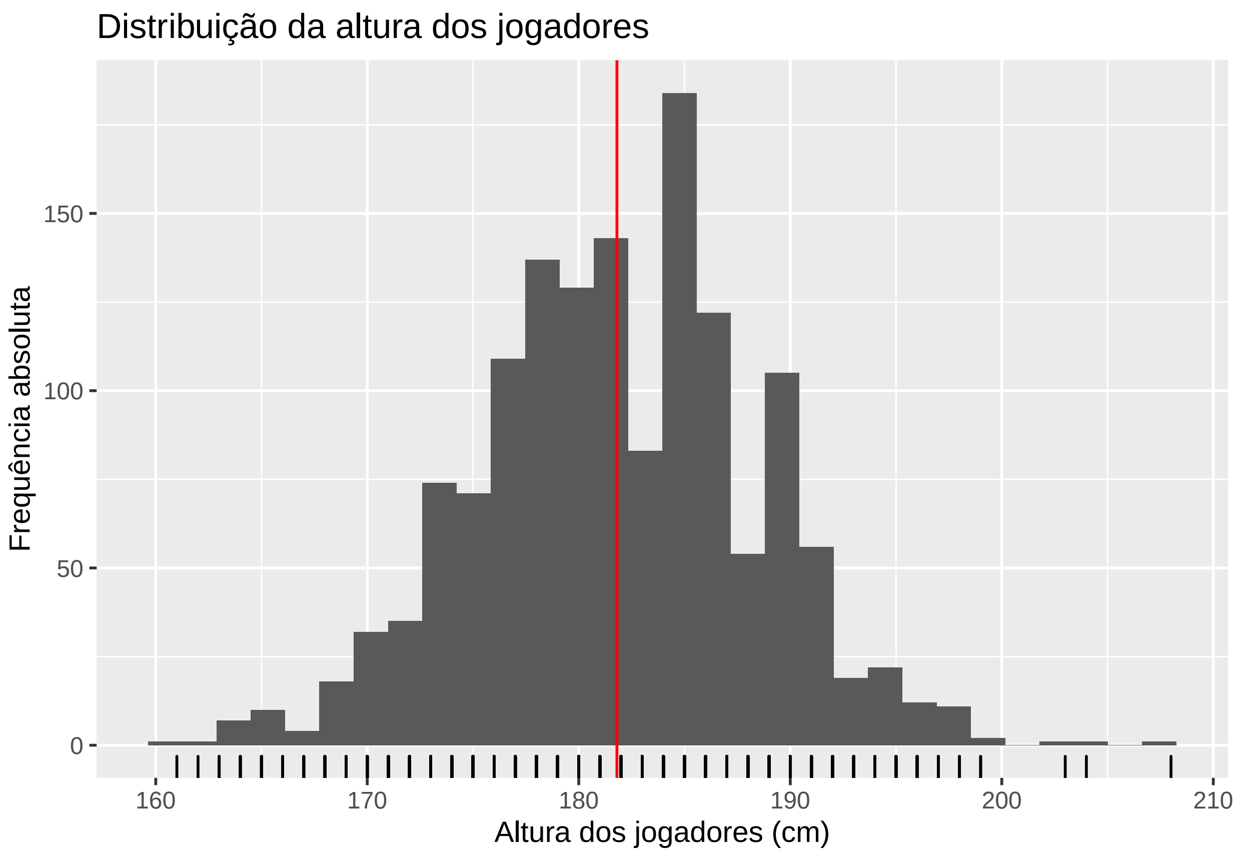
```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



```
#feq = densidade / larg.relat

ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
    geom_histogram(mapping = aes(y = ..count..)) +
    #geom_density(fill = "green") +
    geom_rug() +
    geom_vline(xintercept = media_tb, color = "red") +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

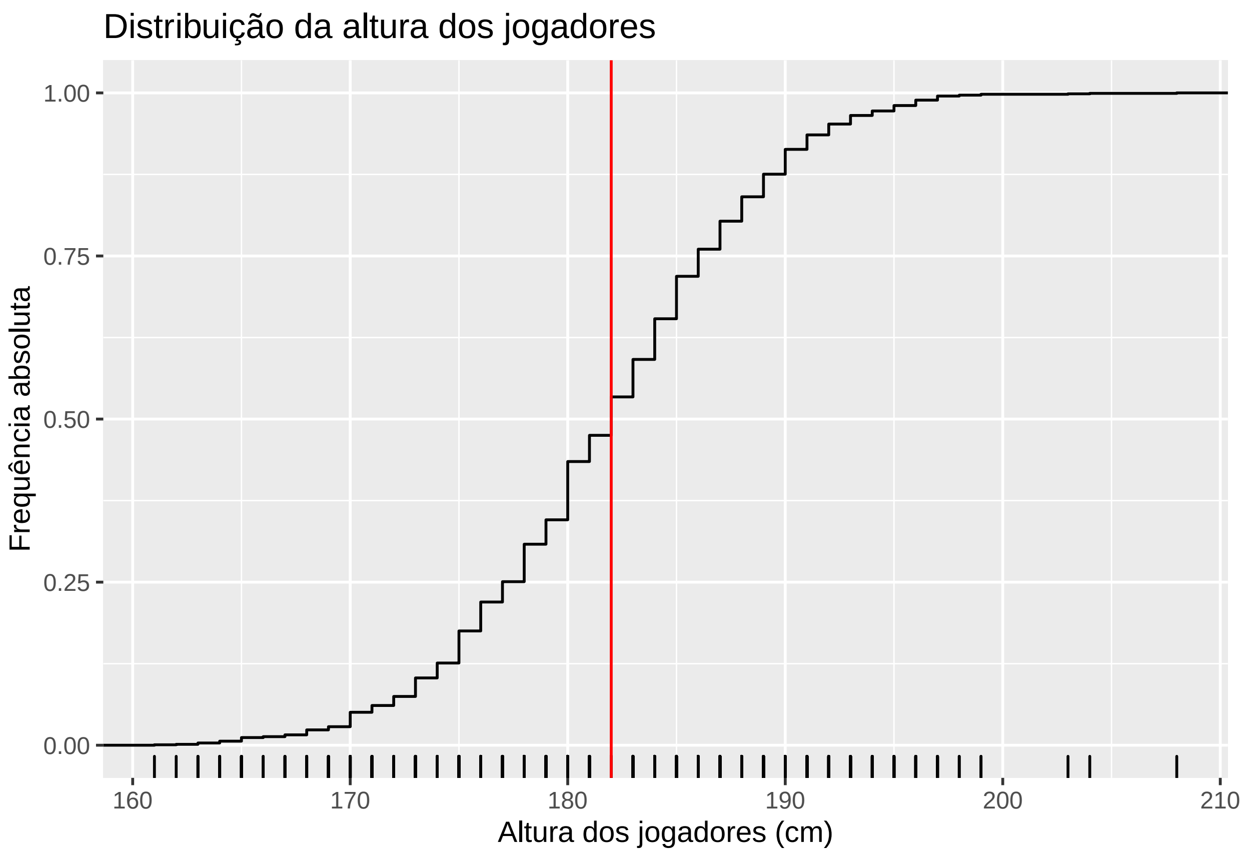
```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

```
## Warning: Removed 84 rows containing non-finite values (stat_bin).
```



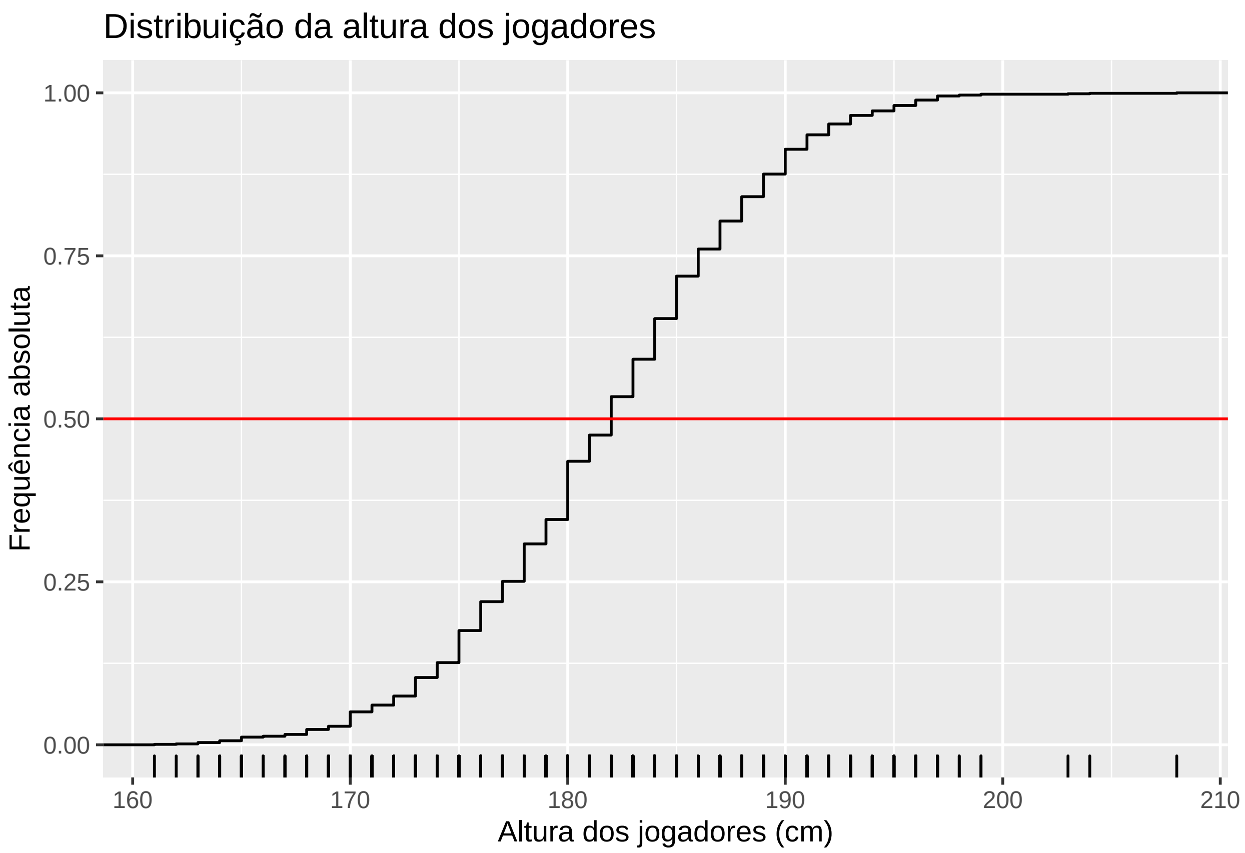
```
mediana_tb <- median(tb_efp$cm, na.rm = TRUE)


ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
  stat_ecdf() +
  geom_rug() +
  geom_vline(xintercept = mediana_tb, color = "red") +
  labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_ecdf).
```



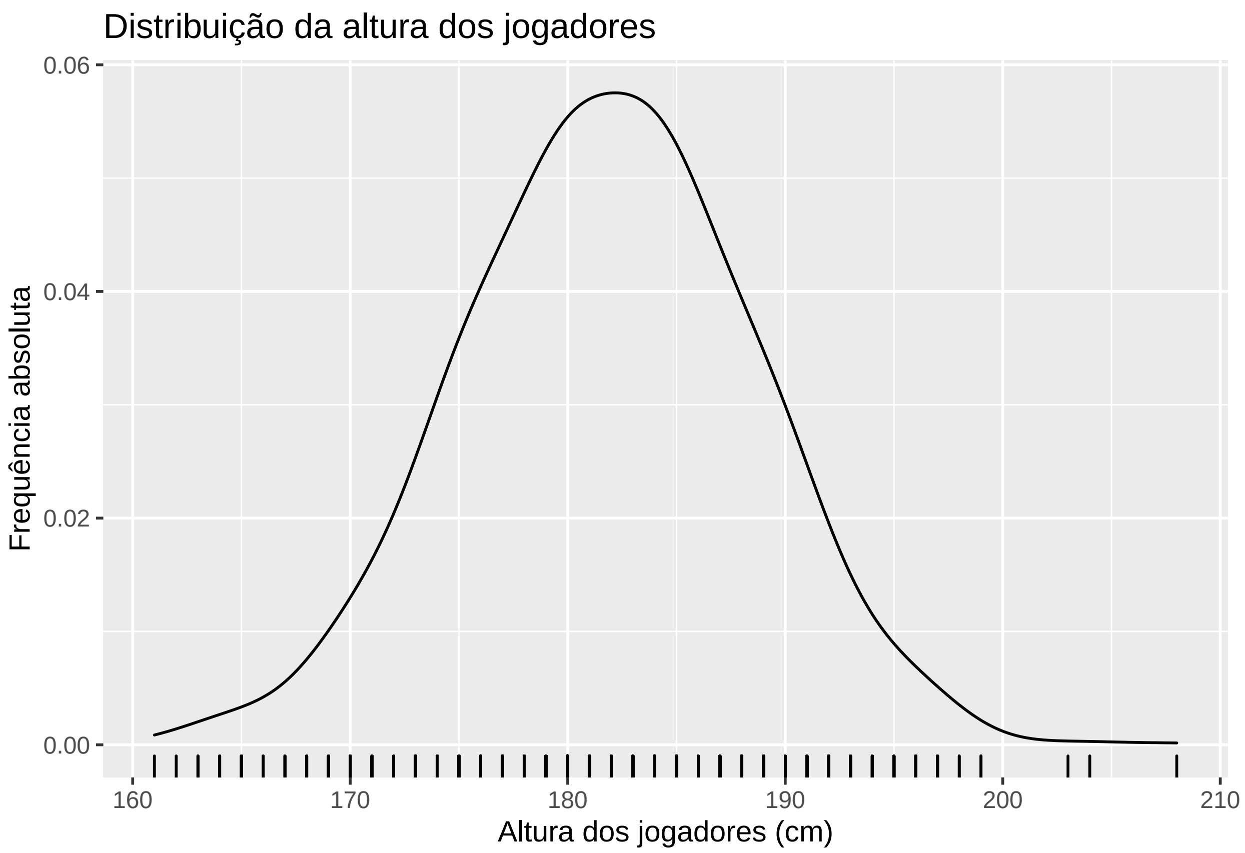
```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
  stat_ecdf() +
  geom_rug() +
  geom_hline(yintercept = 0.5, color = "red") +
  labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_ecdf).
```



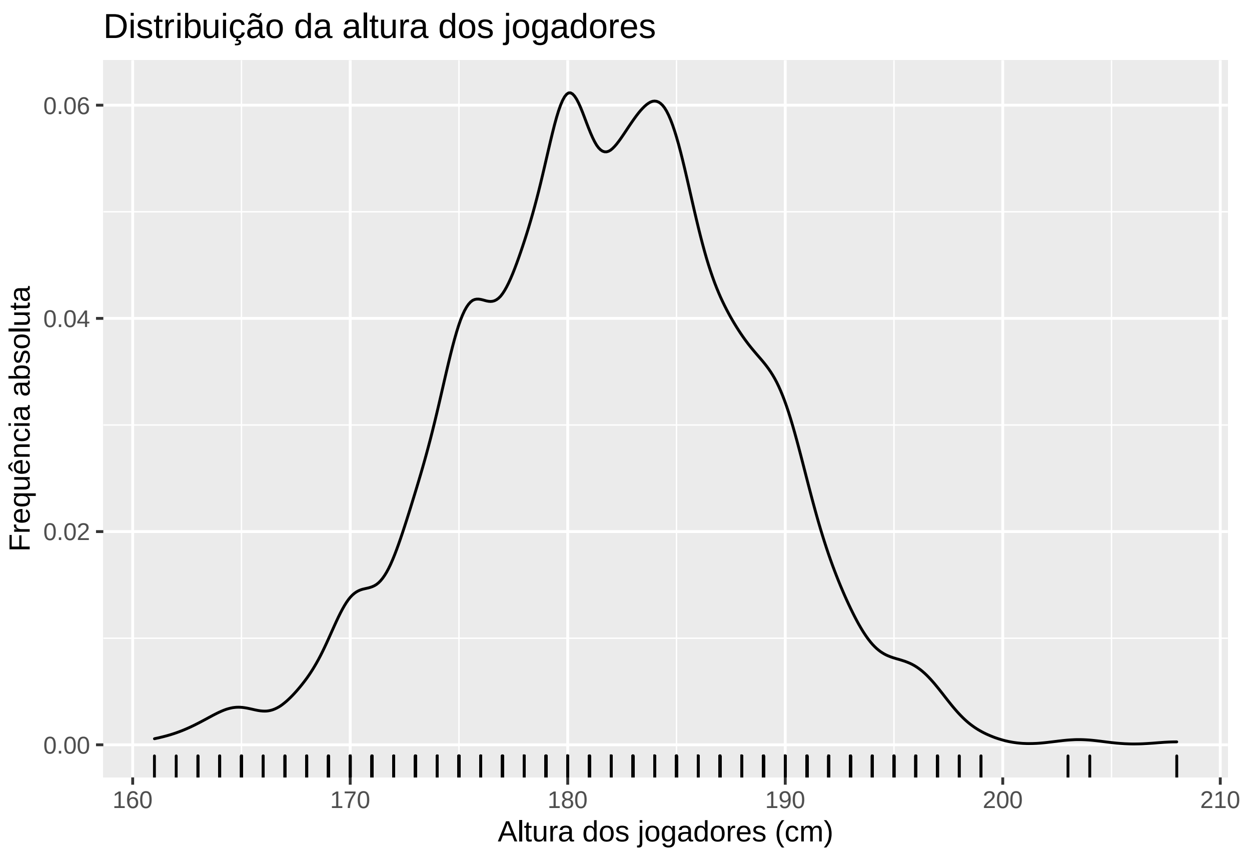
```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
    geom_density(bw = 2) +
    geom_rug() +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```



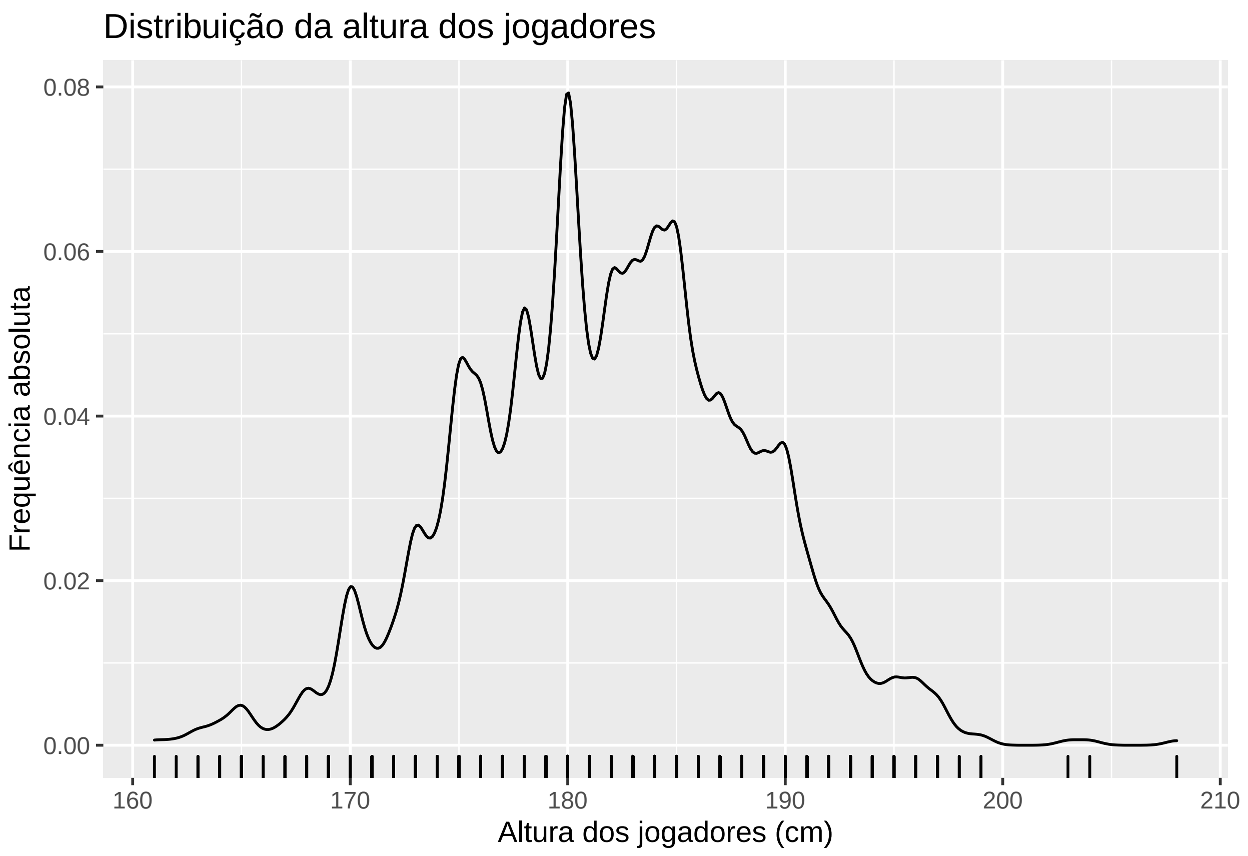
```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
    geom_density(bw = 1) +
    geom_rug() +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```

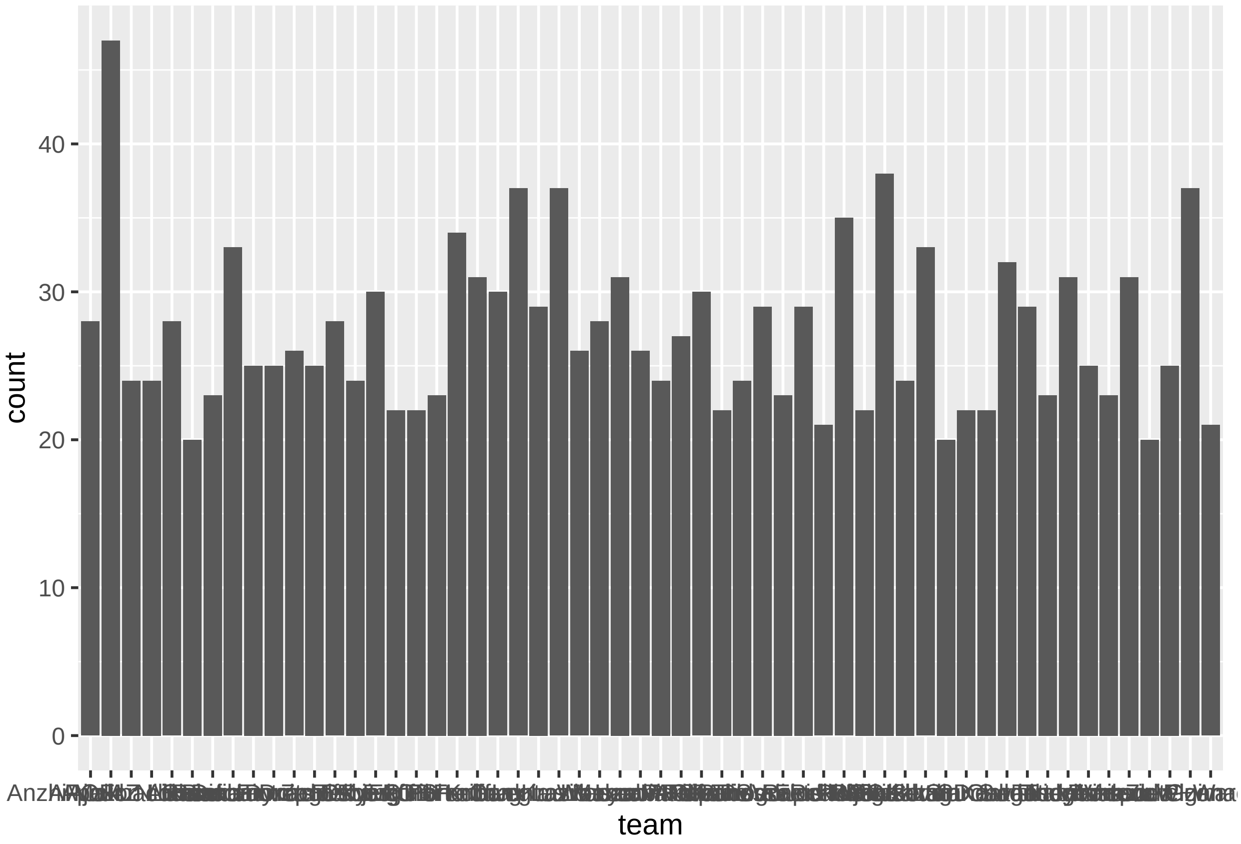


```
ggplot(data = tb_efp,
       mapping = aes(x = cm)) +
    geom_density(bw = 0.5) +
    geom_rug() +
    labs(x = "Altura dos jogadores (cm)",
         y = "Frequência absoluta",
         title = "Distribuição da altura dos jogadores")
```

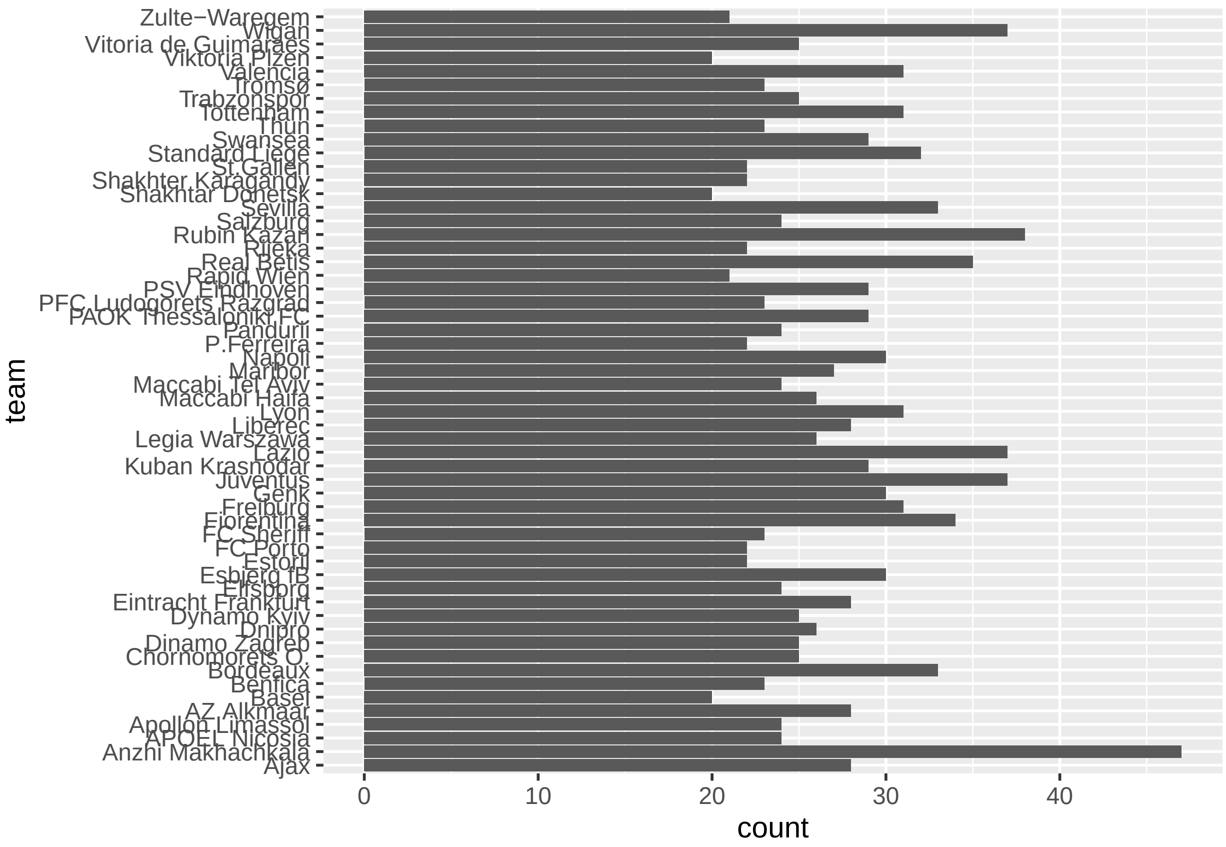
```
## Warning: Removed 84 rows containing non-finite values (stat_density).
```



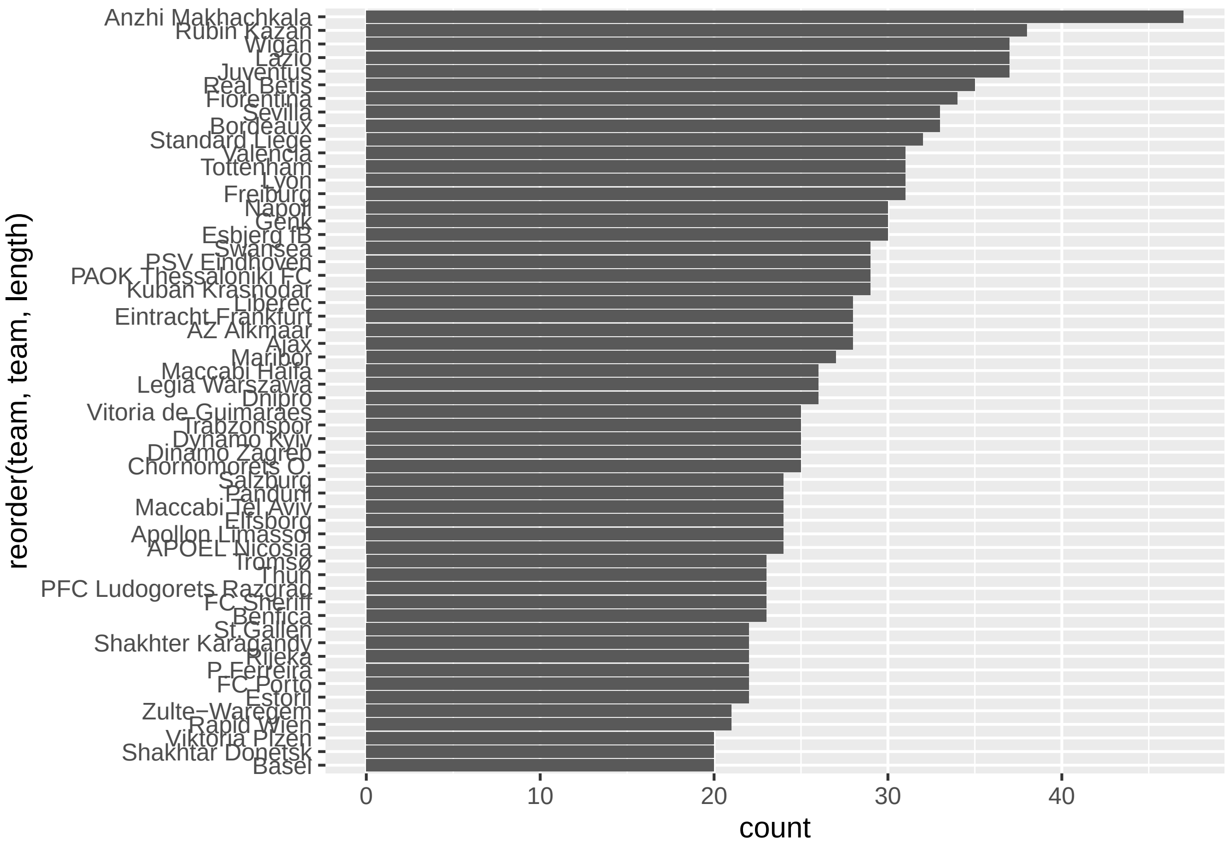
```
ggplot(data = tb_efp,
       mapping = aes(x = team)) +
  geom_bar()
```



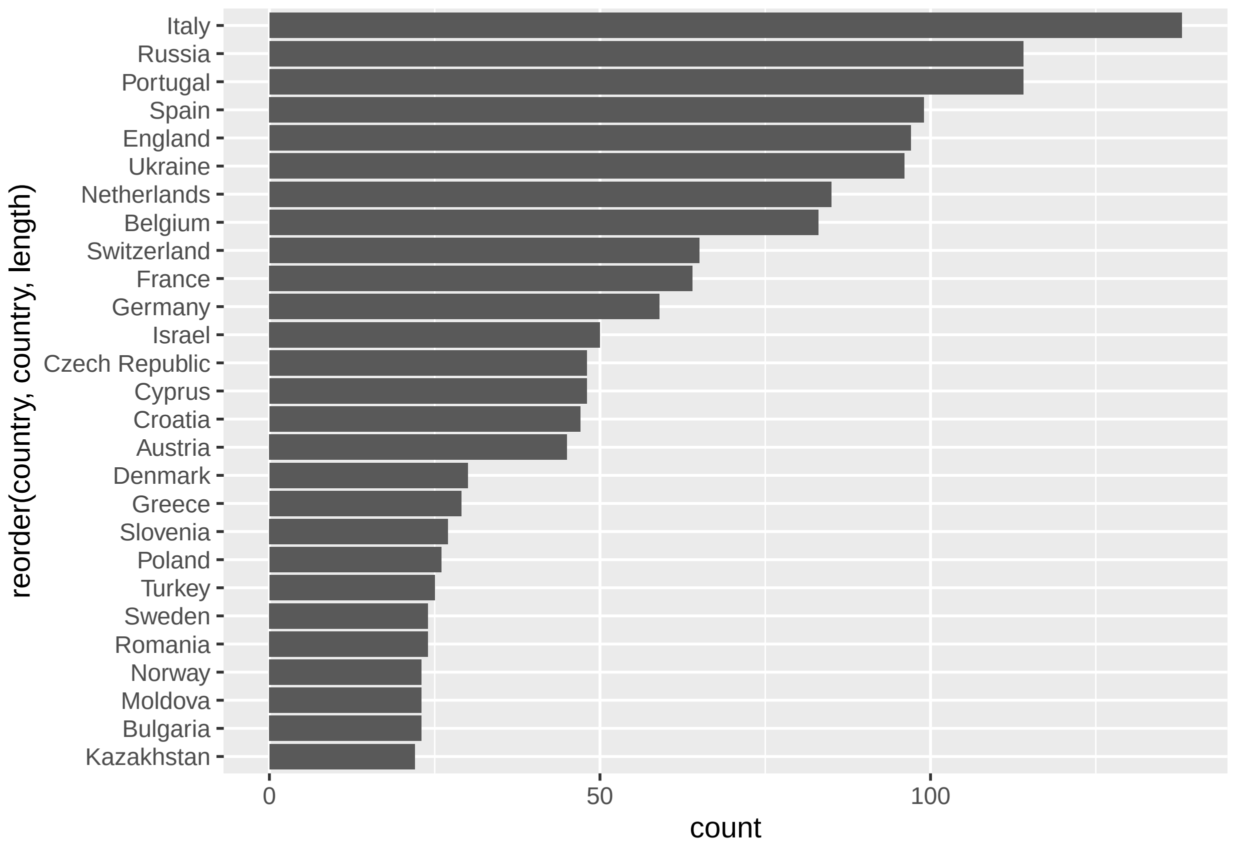
```r
ggplot(data = tb_efp,
       mapping = aes(x = team)) +
  geom_bar() +
  coord_flip()
```



```
ggplot(data = tb_efp,
       mapping = aes(x = reorder(team, team, length))) +
  geom_bar() +
  coord_flip()
```



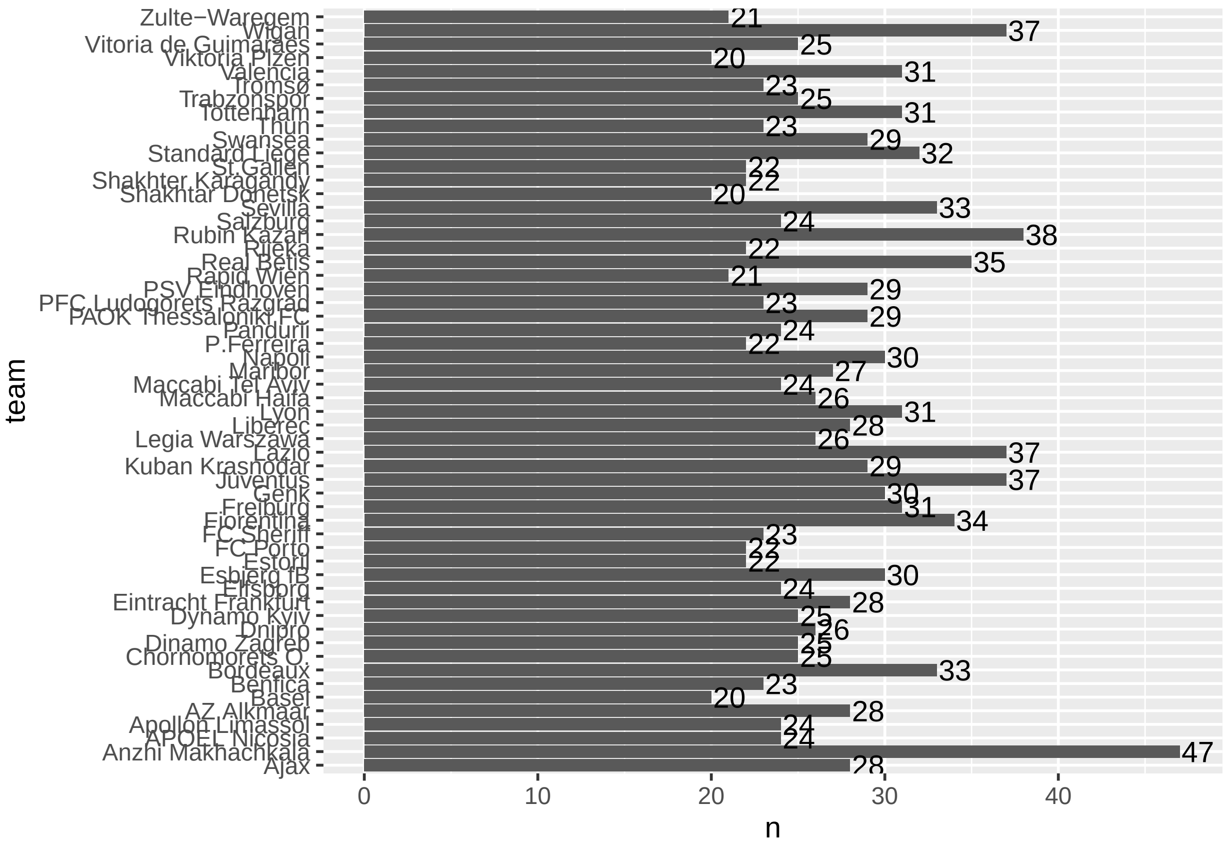
```
ggplot(data = tb_efp,
       mapping = aes(x = reorder(country, country, length))) +
  geom_bar() +
  #geom_text(mapping = aes(x = team, y = ..conut.., label = ..count.., stat = "stat_count")
  coord_flip()
```



```
tb_freq <- tb_efp %>%
  count(team)

tb_freq
```

```
## # A tibble: 56 x 2
##    team                   n
##    <chr>              <int>
##  1 Ajax                  28
##  2 Anzhi Makhachkala     47
##  3 APOEL Nicosia         24
##  4 Apollon Limassol      24
##  5 AZ Alkmaar            28
##  6 Basel                 20
##  7 Benfica               23
##  8 Bordeaux              33
##  9 Chornomorets O.       25
## 10 Dinamo Zagreb         25
## # ... with 46 more rows
```

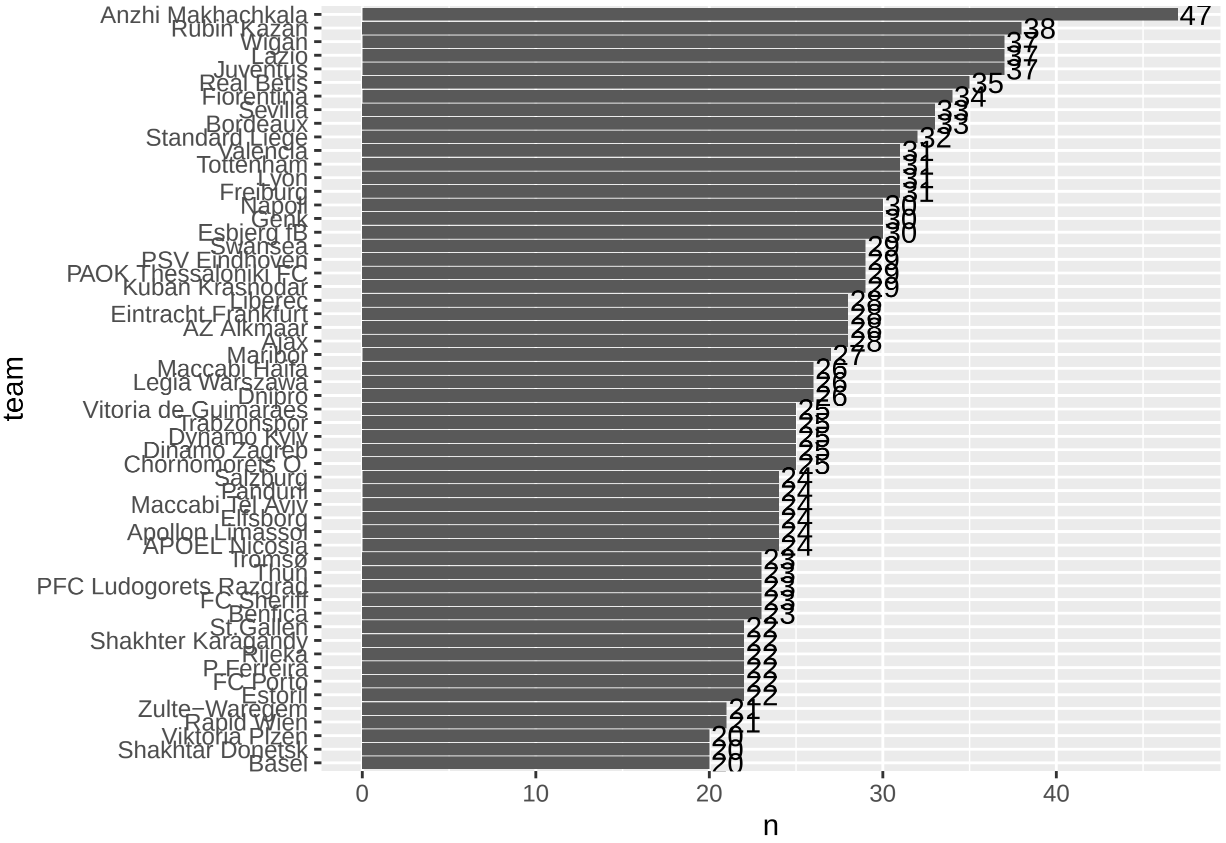
```
ggplot(data = tb_freq,
       mapping = aes(x = n, y = team)) +
  geom_col() +
  geom_text(mapping = aes(label = n),
            hjust = 0, nudge_x = 0.1)
```



```r
tb_freq <- tb_efp %>%
  count(team) %>%
  mutate(team = reorder(team, n))

tb_freq
```

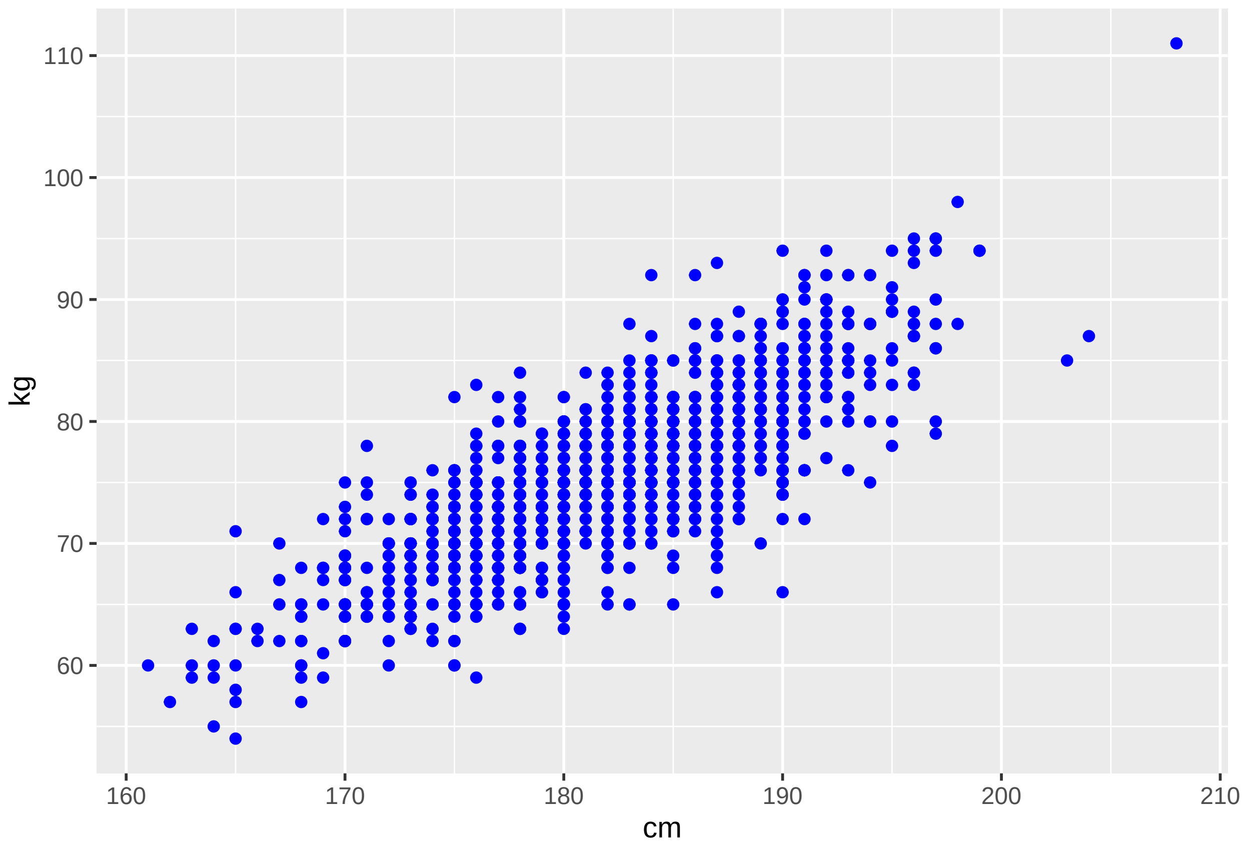
```
## # A tibble: 56 x 2
##    team                  n
##    <fct>             <int>
##  1 Ajax                 28
##  2 Anzhi Makhachkala    47
##  3 APOEL Nicosia        24
##  4 Apollon Limassol     24
##  5 AZ Alkmaar           28
##  6 Basel                20
##  7 Benfica              23
##  8 Bordeaux             33
##  9 Chornomorets O.      25
## 10 Dinamo Zagreb        25
## # ... with 46 more rows
```

```r
ggplot(data = tb_freq,
       mapping = aes(x = n, y = team)) +
  geom_col() +
  geom_text(mapping = aes(label = n),
            hjust = 0, nudge_x = 0.1)
```



```
ggplot(data = tb_efp,
       mapping = aes(x=cm, y=kg)) +
  geom_point( color = "blue")
```

```
## Warning: Removed 111 rows containing missing values (geom_point).
```

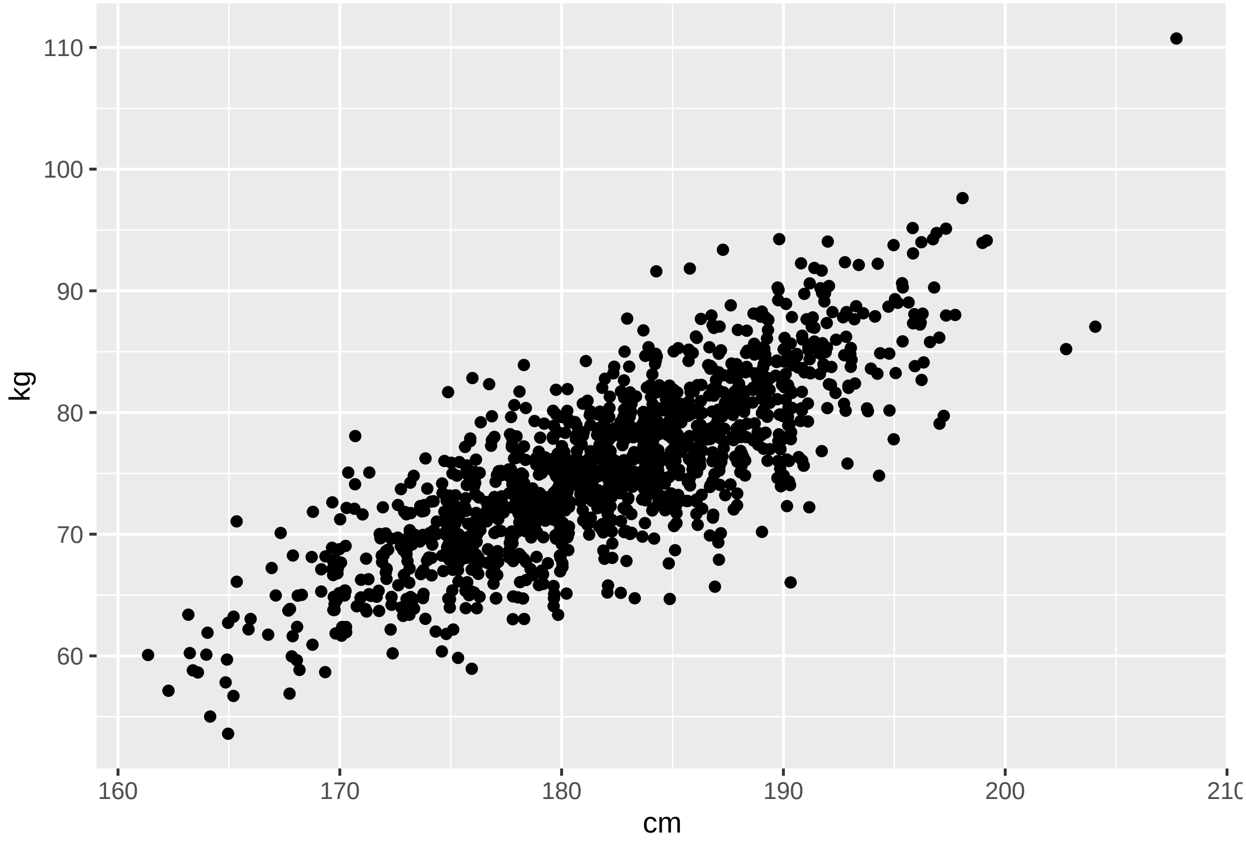


```
tb_efp %>%
  filter(cm == 180, kg == 72)
```

```
## # A tibble: 14 x 17
##    country  team    name   pos     age    cm    kg apps   goal   ass   yel   red
##    <chr>    <chr>   <chr>  <chr> <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl>
##  1 Italy    Juvent~ Ouasi~ M        20   180    72 <NA>     NA    NA    NA    NA
##  2 Turkey   Trabzo~ Soner~ M        23   180    72 1(1)      1    NA    NA    NA
##  3 Russia   Rubin ~ Vladi~ M(C)     29   180    72 13(6)    NA    NA     2    NA
##  4 Belgium  Genk    Piete~ M        19   180    72 1(1)     NA    NA    NA    NA
##  5 France   Lyon    Alass~ M        21   180    72 0(6)     NA    NA    NA    NA
##  6 Czech R~ Liberec Vojte~ D        19   180    72 <NA>     NA    NA    NA    NA
##  7 Russia   Kuban ~ Vladi~ AM(R)    27   180    72 7(1)      1     1    NA    NA
##  8 Russia   Kuban ~ Nikit~ M        23   180    72 1(3)     NA    NA    NA    NA
##  9 England  Wigan   James~ M(CL)    23   180    72 5        NA    NA     1    NA
## 10 England  Wigan   James~ D(CL~    28   180    72 32(2)    NA    NA     7    NA
## 11 Croatia  Rijeka  Ivan ~ M(C)     20   180    72 6        NA    NA     1    NA
## 12 Croatia  Rijeka  Zoran~ M        25   180    72 5(1)     NA    NA     1    NA
## 13 Israel   Maccab~ Gal A~ M(C)     30   180    72 5(1)     NA    NA    NA    NA
## 14 Netherl~ Ajax    Daley~ D(L)~    24   180    72 25        1     1     3    NA
## # ... with 5 more variables: spg <dbl>, ps <dbl>, aw <dbl>, mom <dbl>, rt <dbl>
```

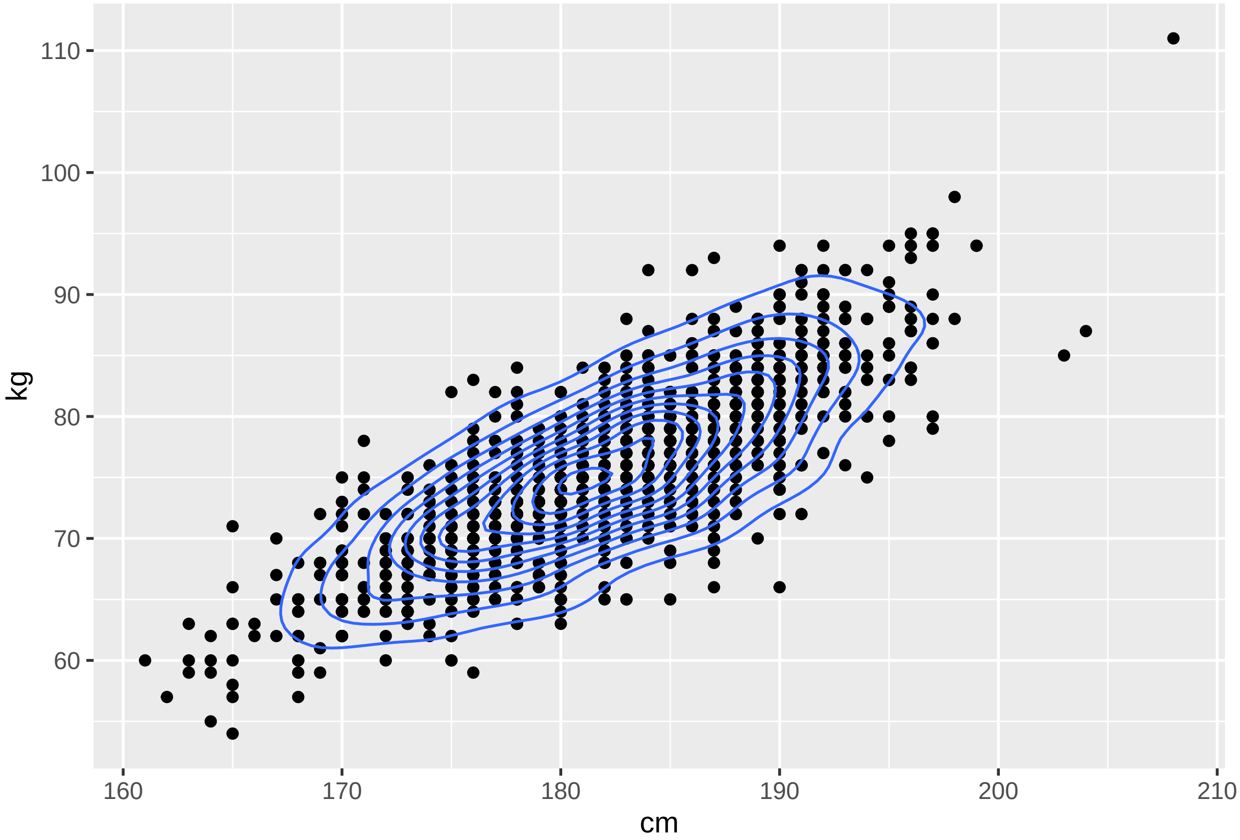
```
ggplot(data = tb_efp,
       mapping = aes(x=cm, y=kg)) +
  geom_jitter()
```

```
## Warning: Removed 111 rows containing missing values (geom_point).
```



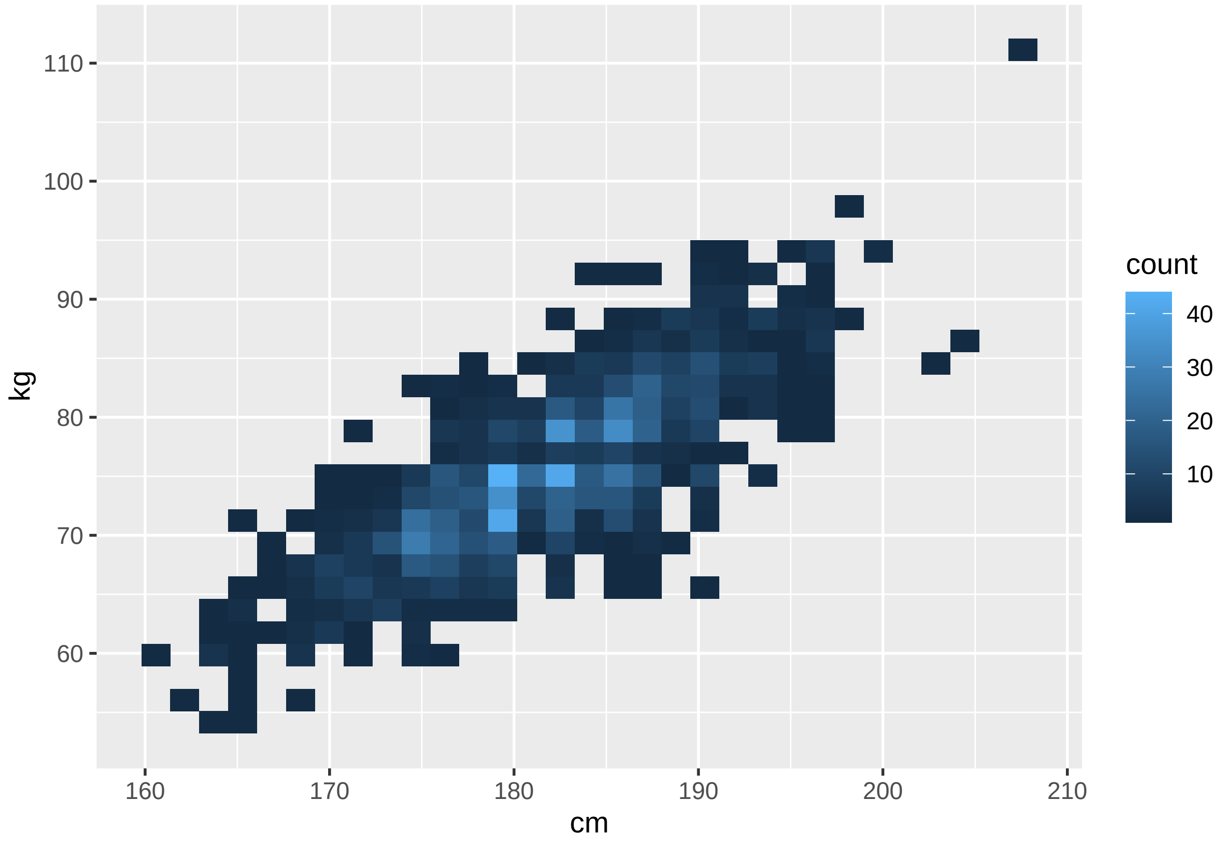
```
ggplot(data = tb_efp,
       mapping = aes(x=cm, y=kg)) +
  geom_point() +
  geom_density2d()
```

```
## Warning: Removed 111 rows containing non-finite values (stat_density2d).

## Warning: Removed 111 rows containing missing values (geom_point).
```



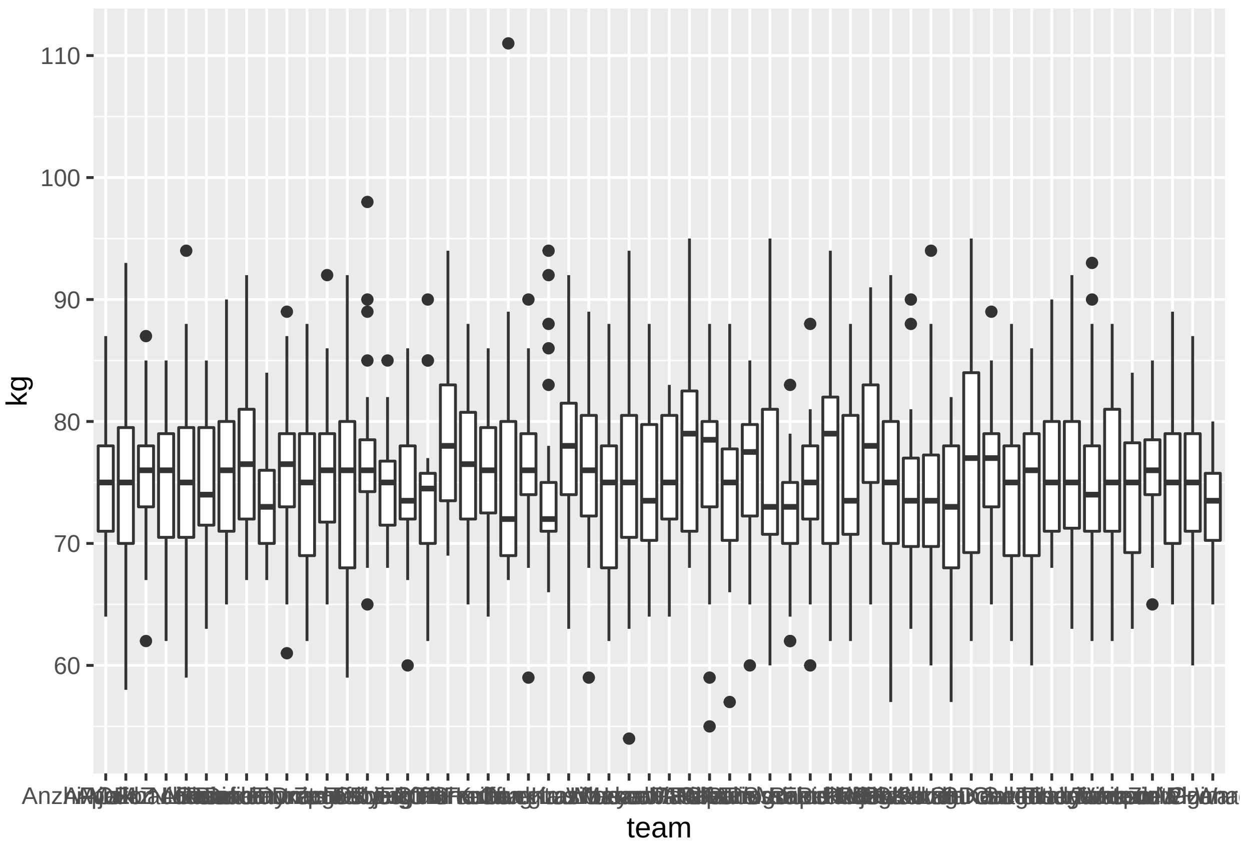
```
ggplot(data = tb_efp,
       mapping = aes(x=cm, y=kg)) +
  geom_bin_2d()
```

```
## Warning: Removed 111 rows containing non-finite values (stat_bin2d).
```



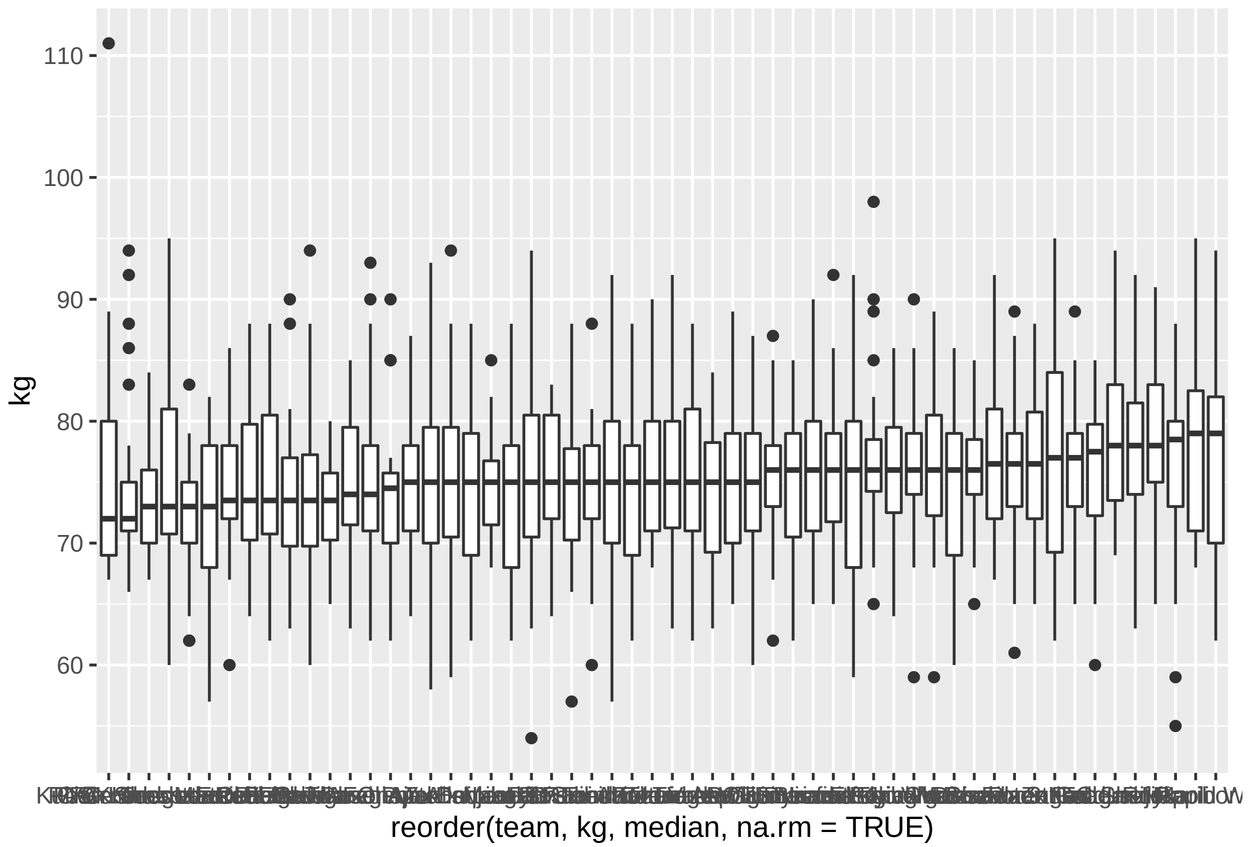
```
ggplot(data = tb_efp,
       mapping = aes(x=team, y=kg)) +
  geom_boxplot()
```

```
## Warning: Removed 111 rows containing non-finite values (stat_boxplot).
```



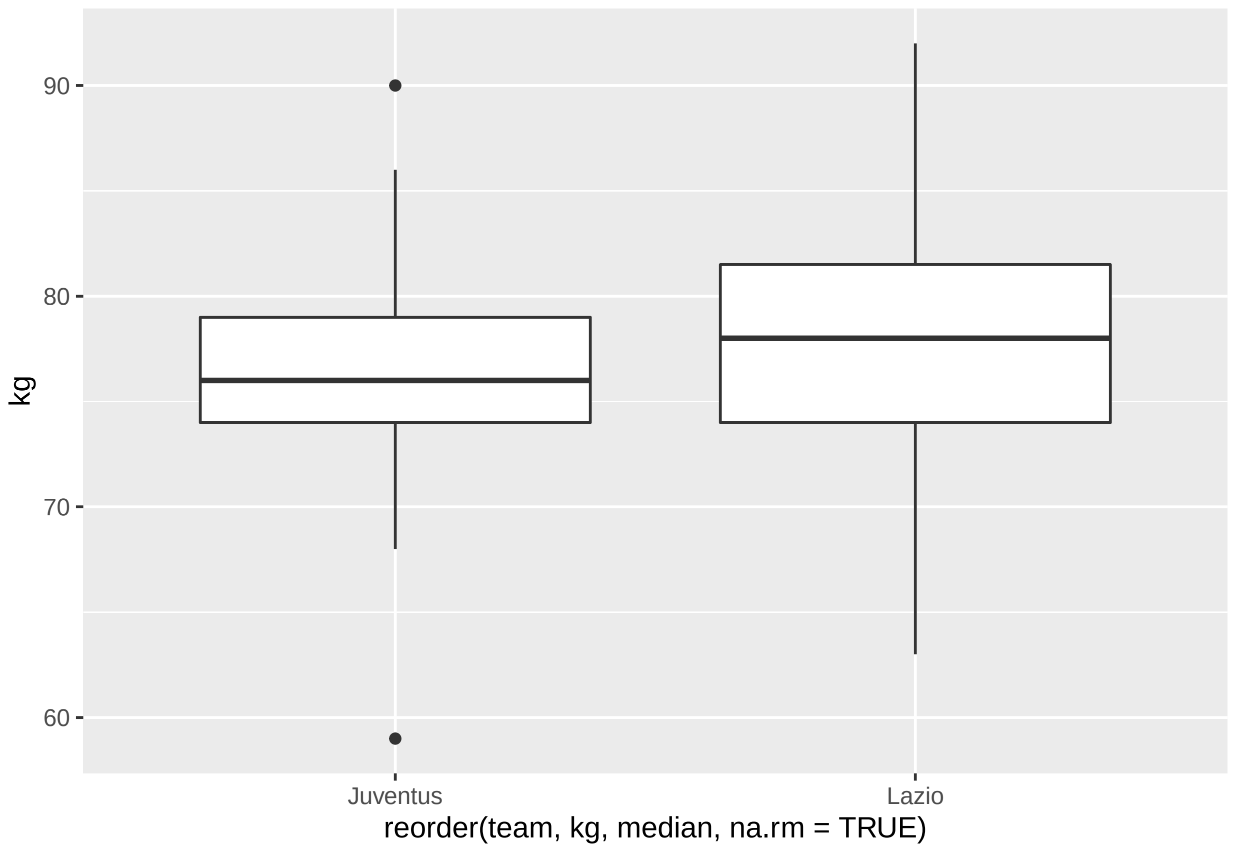
```
ggplot(data = tb_efp,
       mapping = aes(x= reorder(team, kg, median, na.rm = TRUE), y=kg)) +
  geom_boxplot()
```

```
## Warning: Removed 111 rows containing non-finite values (stat_boxplot).
```



```
ggplot(data = filter(tb_efp, team %in% c("Juventus", "Lazio")),
       mapping = aes(x= reorder(team, kg, median, na.rm = TRUE), y=kg)) +
  geom_boxplot()
```

```
## Warning: Removed 6 rows containing non-finite values (stat_boxplot).
```



```
gg <- ggplot2::ggplot(data = tb_freq,
       mapping = aes(x = n, y = team)) +
  geom_col() +
  geom_text(mapping = aes(label = n),
            hjust = 0, nudge_x = 0.1)

#ggsave(file = "meugrafico.png")
#salva o ultimo grafico que vc fez

#ggsave(file = "meugrafico.svg")

apropos("^scale_")
```

```
##   [1] "scale_alpha"               "scale_alpha_binned"
##   [3] "scale_alpha_continuous"    "scale_alpha_date"
##   [5] "scale_alpha_datetime"      "scale_alpha_discrete"
##   [7] "scale_alpha_identity"      "scale_alpha_manual"
##   [9] "scale_alpha_ordinal"       "scale_color_binned"
##  [11] "scale_color_brewer"        "scale_color_continuous"
##  [13] "scale_color_date"          "scale_color_datetime"
##  [15] "scale_color_discrete"      "scale_color_distiller"
##  [17] "scale_color_fermenter"     "scale_color_gradient"
##  [19] "scale_color_gradient2"     "scale_color_gradientn"
##  [21] "scale_color_grey"          "scale_color_hue"
##  [23] "scale_color_identity"      "scale_color_manual"
##  [25] "scale_color_ordinal"       "scale_color_steps"        
##  [27] "scale_color_steps2"        "scale_color_stepsn"       
##  [29] "scale_color_viridis_b"     "scale_color_viridis_c"    
##  [31] "scale_color_viridis_d"     "scale_colour_binned"      
##  [33] "scale_colour_brewer"       "scale_colour_continuous"  
##  [35] "scale_colour_date"         "scale_colour_datetime"    
##  [37] "scale_colour_discrete"     "scale_colour_distiller"   
##  [39] "scale_colour_fermenter"    "scale_colour_gradient"    
##  [41] "scale_colour_gradient2"    "scale_colour_gradientn"   
##  [43] "scale_colour_grey"         "scale_colour_hue"         
##  [45] "scale_colour_identity"     "scale_colour_manual"      
##  [47] "scale_colour_ordinal"      "scale_colour_steps"       
##  [49] "scale_colour_steps2"       "scale_colour_stepsn"      
##  [51] "scale_colour_viridis_b"    "scale_colour_viridis_c"   
##  [53] "scale_colour_viridis_d"    "scale_continuous_identity"
##  [55] "scale_discrete_identity"   "scale_discrete_manual"    
##  [57] "scale_fill_binned"         "scale_fill_brewer"        
##  [59] "scale_fill_continuous"     "scale_fill_date"          
##  [61] "scale_fill_datetime"       "scale_fill_discrete"      
##  [63] "scale_fill_distiller"      "scale_fill_fermenter"     
##  [65] "scale_fill_gradient"       "scale_fill_gradient2"     
##  [67] "scale_fill_gradientn"      "scale_fill_grey"          
##  [69] "scale_fill_hue"            "scale_fill_identity"      
##  [71] "scale_fill_manual"         "scale_fill_ordinal"       
##  [73] "scale_fill_steps"          "scale_fill_steps2"        
##  [75] "scale_fill_stepsn"         "scale_fill_viridis_b"     
##  [77] "scale_fill_viridis_c"      "scale_fill_viridis_d"     
##  [79] "scale_linetype"            "scale_linetype_binned"    
##  [81] "scale_linetype_continuous" "scale_linetype_discrete"  
##  [83] "scale_linetype_identity"   "scale_linetype_manual"    
##  [85] "scale_radius"              "scale_shape"              
##  [87] "scale_shape_binned"        "scale_shape_continuous"   
##  [89] "scale_shape_discrete"      "scale_shape_identity"     
##  [91] "scale_shape_manual"        "scale_shape_ordinal"      
##  [93] "scale_size"                "scale_size_area"          
##  [95] "scale_size_binned"         "scale_size_binned_area"   
##  [97] "scale_size_continuous"     "scale_size_date"          
##  [99] "scale_size_datetime"       "scale_size_discrete"      
## [101] "scale_size_identity"       "scale_size_manual"        
## [103] "scale_size_ordinal"        "scale_type"               
## [105] "scale_x_binned"            "scale_x_continuous"       
## [107] "scale_x_date"              "scale_x_datetime"         
## [109] "scale_x_discrete"          "scale_x_log10"            
## [111] "scale_x_reverse"           "scale_x_sqrt"             
## [113] "scale_x_time"              "scale_y_binned"           
## [115] "scale_y_continuous"        "scale_y_date"             
## [117] "scale_y_datetime"          "scale_y_discrete"         
## [119] "scale_y_log10"             "scale_y_reverse"          
## [121] "scale_y_sqrt"              "scale_y_time"
```

```
apropos("^geom_")
```

```
##   [1] "geom_abline"            "geom_area"              "geom_bar"              
##   [4] "geom_bin_2d"            "geom_bin2d"             "geom_blank"
##  [7] "geom_boxplot"           "geom_col"               "geom_contour"
## [10] "geom_contour_filled"    "geom_count"             "geom_crossbar"
## [13] "geom_curve"             "geom_density"           "geom_density_2d"
## [16] "geom_density_2d_filled" "geom_density2d"         "geom_density2d_filled"
## [19] "geom_dotplot"           "geom_errorbar"          "geom_errorbarh"
## [22] "geom_freqpoly"          "geom_function"          "geom_hex"
## [25] "geom_histogram"         "geom_hline"             "geom_jitter"
## [28] "geom_label"             "geom_line"              "geom_linerange"
## [31] "geom_map"               "geom_path"              "geom_point"
## [34] "geom_pointrange"        "geom_polygon"           "geom_qq"
## [37] "geom_qq_line"           "geom_quantile"          "geom_raster"
## [40] "geom_rect"              "geom_ribbon"            "geom_rug"
## [43] "geom_segment"           "geom_sf"                "geom_sf_label"
## [46] "geom_sf_text"           "geom_smooth"            "geom_spoke"
## [49] "geom_step"              "geom_text"              "geom_tile"
## [52] "geom_violin"            "geom_vline"
```

```
apropos("^stat_")
```

```
##  [1] "stat_bin"               "stat_bin_2d"            "stat_bin_hex"
##  [4] "stat_bin2d"             "stat_binhex"            "stat_boxplot"
##  [7] "stat_contour"           "stat_contour_filled"    "stat_count"
## [10] "stat_density"           "stat_density_2d"        "stat_density_2d_filled"
## [13] "stat_density2d"         "stat_density2d_filled"  "stat_ecdf"
## [16] "stat_ellipse"           "stat_function"          "stat_identity"
## [19] "stat_qq"                "stat_qq_line"           "stat_quantile"
## [22] "stat_sf"                "stat_sf_coordinates"    "stat_smooth"
## [25] "stat_spoke"             "stat_sum"               "stat_summary"
## [28] "stat_summary_2d"        "stat_summary_bin"       "stat_summary_hex"
## [31] "stat_summary2d"         "stat_unique"            "stat_ydensity"
```