

APS Estatística I - Analisando os times do Brasil

Carregando os dados e pacotes

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --
## v ggplot2 3.3.3      v purrr  0.3.4
## v tibble  3.1.0      v dplyr  1.0.5
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   1.4.0      v forcats 0.5.1

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

```
library(janitor)

##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##   chisq.test, fisher.test
```

```
library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##   combine
```

```
library(viridis)

## Loading required package: viridisLite

library(RColorBrewer)
theme_set(theme_minimal())

dados <- readxl::read_excel("C:/Users/alves/OneDrive - Insper - Instituto de Ensino e Pesquisa/Estudos/
                             col_types = c("text", "text", "numeric",
                                             "date", "text", "text", "text", "numeric",
                                             "numeric", "numeric", "text", "numeric",
                                             "numeric", "numeric")) %>% clean_names() %>%
  mutate('somagols' = golcasa+golvisitante)
```

Considerando a contagem de resultados por time e temporadas.

Contagem dos resultados por ano

```
resultado_ano <- dados %>% group_by(res, temporada) %>%
  summarise(n = n()) %>% mutate(freq = round((n / sum(n))*100,2))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

Considerando apenas a contagem de cada possibilidade de resultado

```
resultado_ano_casa <- dados %>% group_by(res, temporada, casa) %>%
  summarise(n = n()) %>% mutate(freq = round((n / sum(n))*100,2))
```

`summarise()` has grouped output by 'res', 'temporada'. You can override using the `.groups` argument.

```
resultado_ano_visitante <- dados %>% group_by(res, temporada, visitante) %>%
  summarise(n = n()) %>% mutate(freq = round((n / sum(n))*100,2))
```

`summarise()` has grouped output by 'res', 'temporada'. You can override using the `.groups` argument.

Calculando algumas medidas de posição e dispersão da contagem dos resultados por time e por ano

```
medidas_res_c <- as.tibble(resultado_ano_casa %>% filter(res == 'C') %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

Warning: `as.tibble()` was deprecated in tibble 2.0.0.

Please use `as_tibble()` instead.

The signature and semantics have changed, see `?as_tibble`.

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_res_v <- as.tibble(resultado_ano_casa %>% filter(res == 'V') %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_res_e <- as.tibble(resultado_ano_casa %>% filter(res == 'E') %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_res_casa <- as.tibble(resultado_ano_casa %>% group_by(casa, res, temporada) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'casa', 'res'. You can override using the `.groups` argument.

criando as estações do ano

```
inverno_2012 <- c(dados %>% select(data) %>% filter(data > '2012-03-20' & data < '2012-09-22'))
verao_2012 <- dados %>% select(data) %>% filter(data > '2012-09-22' & data < '2013-03-20')

inverno_2013 <- dados %>% select(data) %>% filter(data > '2013-03-20' & data < '2013-09-22')
verao_2013 <- dados %>% select(data) %>% filter(data > '2013-09-22' & data < '2014-03-20')
```

```

inverno_2014 <- dados %>% select(data) %>% filter(data > '2014-03-20' & data < '2014-09-22')
verao_2014 <- dados %>% select(data) %>% filter(data > '2014-09-22' & data < '2015-03-20')

inverno_2015 <- dados %>% select(data) %>% filter(data > '2015-03-20' & data < '2015-09-22')
verao_2015 <- dados %>% select(data) %>% filter(data > '2015-09-22' & data < '2016-03-20')

inverno_2016 <- dados %>% select(data) %>% filter(data > '2016-03-20' & data < '2016-09-22')
verao_2016 <- dados %>% select(data) %>% filter(data > '2016-09-22' & data < '2017-03-20')

inverno_2017 <- dados %>% select(data) %>% filter(data > '2017-03-20' & data < '2017-09-22')
verao_2017 <- dados %>% select(data) %>% filter(data > '2017-09-22' & data < '2018-03-20')

inverno_2018 <- dados %>% select(data) %>% filter(data > '2018-03-20' & data < '2018-09-22')
verao_2018 <- dados %>% select(data) %>% filter(data > '2018-09-22' & data < '2019-03-20')

inverno_2019 <- dados %>% select(data) %>% filter(data > '2019-03-20' & data < '2019-09-22')
verao_2019 <- dados %>% select(data) %>% filter(data > '2019-09-22' & data < '2020-03-20')

inverno_2020 <- dados %>% select(data) %>% filter(data > '2020-03-20' & data < '2020-09-22')
verao_2020 <- dados %>% select(data) %>% filter(data > '2020-09-22' & data < '2021-03-20')

```

Segregando a contagem dos resultados por estação por ano nos estados ‘quentes’

```

#### CLIMA NOS ESTADOS QUENTES ####
estados_quentes <- c('Alagoas', 'Bahia', 'Ceara', 'Pernambuco', 'Goias', 'Rio de Janeiro')

climao_quente <- dados %>% group_by(res, data, periodo) %>% filter(estado %in% estados_quentes) %>%
  summarise(n = n())

## `summarise()` has grouped output by 'res', 'data'. You can override using the `.groups` argument.
medidas_frio_quentes_2014 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2014$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_calor_quentes_2014 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2014$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_frio_quentes_2015 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2015$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```

```
medidas_calor_quentes_2015 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2015$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_quentes_2016 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2016$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_quentes_2016 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2016$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_quentes_2017 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2017$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_quentes_2017 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2017$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_quentes_2018 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2018$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_quentes_2018 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2018$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_quentes_2019 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2019$data) %>%
```

```
summarise('media' = mean(n),
          'mediana' = median(n),
          'desvpad' = sd(n))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_quentes_2019 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2019$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_quentes_2020 <- as.tibble(climao_quente %>%
  filter(data %in% inverno_2020$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_quentes_2020 <- as.tibble(climao_quente %>%
  filter(data %in% verao_2020$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

Segregando a contagem dos resultados por estação por ano nos estados ‘não quentes’

```
climao_gelado <- dados %>% group_by(res, data, periodo) %>% filter(estado %in% estados_quentes) %>%
  summarise(n = n())
```

`summarise()` has grouped output by 'res', 'data'. You can override using the `.groups` argument.

```
medidas_frio_gelado_2014 <- as.tibble(climao_gelado %>%
  filter(data %in% inverno_2014$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_gelado_2014 <- as.tibble(climao_gelado %>%
  filter(data %in% verao_2014$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_gelado_2015 <- as.tibble(climao_gelado %>%
  filter(data %in% inverno_2015$data) %>%
  summarise('media' = mean(n),
```

```

                                'mediana' = median(n),
                                'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_calor_gelado_2015 <- as.tibble(climao_gelado %>%
  filter(data %in% verao_2015$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_frio_gelado_2016 <- as.tibble(climao_gelado %>%
  filter(data %in% inverno_2016$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_calor_gelado_2016 <- as.tibble(climao_gelado %>%
  filter(data %in% verao_2016$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_frio_gelado_2017 <- as.tibble(climao_gelado %>%
  filter(data %in% inverno_2017$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_calor_gelado_2017 <- as.tibble(climao_gelado %>%
  filter(data %in% verao_2017$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_frio_gelado_2018 <- as.tibble(climao_gelado %>%
  filter(data %in% inverno_2018$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

## `summarise()` has grouped output by 'res'. You can override using the `.groups` argument.
medidas_calor_gelado_2018 <- as.tibble(climao_gelado %>%
  filter(data %in% verao_2018$data) %>%
  summarise('media' = mean(n),
            'mediana' = median(n),
            'desvpad' = sd(n)))

```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_gelado_2019 <- as.tibble(climao_gelado %>%  
  filter(data %in% inverno_2019$data) %>%  
  summarise('media' = mean(n),  
            'mediana' = median(n),  
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_gelado_2019 <- as.tibble(climao_gelado %>%  
  filter(data %in% verao_2019$data) %>%  
  summarise('media' = mean(n),  
            'mediana' = median(n),  
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_frio_gelado_2020 <- as.tibble(climao_gelado %>%  
  filter(data %in% inverno_2020$data) %>%  
  summarise('media' = mean(n),  
            'mediana' = median(n),  
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

```
medidas_calor_gelado_2020 <- as.tibble(climao_gelado %>%  
  filter(data %in% verao_2020$data) %>%  
  summarise('media' = mean(n),  
            'mediana' = median(n),  
            'desvpad' = sd(n)))
```

`summarise()` has grouped output by 'res'. You can override using the `.groups` argument.

Gols nos estados por estação por ano

```
medidas_sg_inverno_2014 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% inverno_2014$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2014 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% verao_2014$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_inverno_2015 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% inverno_2015$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2015 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% verao_2015$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_inverno_2016 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% inverno_2016$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2016 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% verao_2016$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_inverno_2017 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% inverno_2017$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2017 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% verao_2017$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_inverno_2018 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% inverno_2018$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2018 <- as.tibble(dados %>% group_by(estado, periodo) %>%  
  filter(data %in% verao_2018$data) %>%  
  summarise('media' = mean(somagols),  
            'mediana' = median(somagols),  
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.


```
medidas_sg_inverno_2019 <- as.tibble(dados %>% group_by(estado, periodo) %>%
  filter(data %in% inverno_2019$data) %>%
  summarise('media' = mean(somagols),
            'mediana' = median(somagols),
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2019 <- as.tibble(dados %>% group_by(estado, periodo) %>%
  filter(data %in% vero_2019$data) %>%
  summarise('media' = mean(somagols),
            'mediana' = median(somagols),
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_inverno_2020 <- as.tibble(dados %>% group_by(estado, periodo) %>%
  filter(data %in% inverno_2020$data) %>%
  summarise('media' = mean(somagols),
            'mediana' = median(somagols),
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

```
medidas_sg_verao_2020 <- as.tibble(dados %>% group_by(estado, periodo) %>%
  filter(data %in% vero_2020$data) %>%
  summarise('media' = mean(somagols),
            'mediana' = median(somagols),
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

Trabalhando com a soma dos gols por período do dia e por temporada

```
somagols_estado_periodo <- dados %>% group_by(somagols, estado, periodo)
```

```
medidas_gols_estado_periodo <- as.tibble(somagols_estado_periodo %>% group_by(estado, periodo) %>%
  summarise('media' = mean(somagols),
            'mediana' = median(somagols),
            'desvpad' = sd(somagols)))
```

`summarise()` has grouped output by 'estado'. You can override using the `.groups` argument.

Calculando medidas dos gols da casa por ano

```
medidas_gols_ano <- as.tibble(dados %>% group_by(temporada, res)%>%
  filter(res == 'C') %>%
  summarise('media' = mean(golcasa),
            'mediana' = median(golcasa),
            'desvpad' = sd(golcasa)))
```

`summarise()` has grouped output by 'temporada'. You can override using the `.groups` argument.

```
medidas_gols_estado <- as.tibble(dados %>% group_by(estado) %>%
  filter(res == 'C') %>%
  summarise('media' = mean(golcasa),
            'mediana' = median(golcasa),
            'desvpad' = sd(golcasa)))
```

Plots

Distribuição dos resultados ao longo dos anos

```
dist_res_2020 <- resultado_ano %>%
  group_by(temporada) %>%
  ggplot(aes(temporada, n, colour = res)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Quantidade de vitórias do mandante por temporada') +
  xlab('Temporada') + ylab('Quantidade') + labs(col = 'Resultado') +
  ggeasy::easy_center_title()
```

Distribuição dos resultados da casa por temporada

```
dist_c_casa_ano <- resultado_ano_casa %>% filter(res == 'C') %>%
  mutate(casa = fct_reorder(casa, n)) %>%
  group_by(casa) %>%
  ggplot(aes(temporada, casa, colour = n)) + geom_point() +
  scale_color_viridis(discrete = FALSE) +
  ggtitle('Quantidade de vitórias da casa por ano') +
  xlab('Temporada') + ylab('Time') + labs(col = 'Quantidade') +
  ggeasy::easy_center_title()
```

```
dist_v_casa_ano <- resultado_ano_casa %>% filter(res == 'V') %>%
  mutate(casa = fct_reorder(casa, n)) %>%
  group_by(casa) %>%
  ggplot(aes(temporada, casa, colour = n)) + geom_point() +
  scale_color_viridis(discrete = FALSE) +
  ggtitle('Quantidade de derrotas da casa por ano') +
  xlab('Temporada') + ylab('Time') + labs(col = 'Quantidade') +
  ggeasy::easy_center_title()
```

```
grid.arrange(arrangeGrob(dist_c_casa_ano, dist_v_casa_ano),
  top = 'Distribuição da quantidade de resultados do time considerando o mandante')
```

caderno_script_files/figure-latex/unnamed-chunk-12-1.pdf

Plot média de gols por período

```
dist_med_gols_perodo <- medidas_gols_estado_perodo %>%
  mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols a tarde por estado') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()
```

Plot de gols por estado

```
dist_med_gols_estado <- medidas_gols_estado %>% mutate(estado = fct_reorder(estado, media)) %>%
  ggplot(aes(media, estado, colour = media)) + geom_point() +
  scale_color_viridis(discrete = FALSE) +
  ggtitle('Media de gols da casa por estado') +
  xlab('Ano') + ylab('Media de gols') +
  theme(legend.position="none")
ggeasy::easy_center_title()
```

```
## List of 1
## $ plot.title:List of 11
## ..$ family      : NULL
## ..$ face         : NULL
## ..$ colour       : NULL
## ..$ size         : NULL
## ..$ hjust        : num 0.5
## ..$ vjust        : NULL
## ..$ angle        : NULL
## ..$ lineheight   : NULL
## ..$ margin       : NULL
## ..$ debug        : NULL
## ..$ inherit.blank: logi FALSE
## ..- attr(*, "class")= chr [1:2] "element_text" "element"
## - attr(*, "class")= chr [1:2] "theme" "gg"
## - attr(*, "complete")= logi FALSE
## - attr(*, "validate")= logi TRUE
```

Plots medias gol por estado e média gol por temporada

```
medidas_gols_ano %>%
  ggplot(aes(media, temporada, colour = media)) + geom_point() +
  scale_color_viridis(discrete = FALSE) +
  ggtitle('Media de gols da casa por ano') +
  xlab('Ano') + ylab('Media de gols') +
  theme(legend.position="none")
```

caderno_script_files/figure-latex/unnamed-chunk-15-1.pdf

```
ggeasy::easy_center_title()
```

```
## List of 1
## $ plot.title:List of 11
## ..$ family      : NULL
## ..$ face        : NULL
## ..$ colour      : NULL
## ..$ size        : NULL
## ..$ hjust       : num 0.5
## ..$ vjust       : NULL
## ..$ angle       : NULL
## ..$ lineheight  : NULL
## ..$ margin      : NULL
## ..$ debug       : NULL
## ..$ inherit.blank: logi FALSE
## ..- attr(*, "class")= chr [1:2] "element_text" "element"
## - attr(*, "class")= chr [1:2] "theme" "gg"
## - attr(*, "complete")= logi FALSE
## - attr(*, "validate")= logi TRUE
```

Plot soma de gols por estado

```
dist_sg_al <- dados %>% filter(estado == 'Alagoas') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período em Alagoas') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()
```

```
dist_sg_ba <- dados %>% filter(estado == 'Bahia') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período na Bahia') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()
```

```
dist_sg_ce <- dados %>% filter(estado == 'Ceara') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período no Ceara') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()
```

```
dist_sg_go <- dados %>% filter(estado == 'Goias') %>%
```

```

ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período em Goiás') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_mg <- dados %>% filter(estado == 'Minas Gerais') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período em Minas Gerais') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_pr <- dados %>% filter(estado == 'Parana') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período no Parana') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_pb <- dados %>% filter(estado == 'Pernambuco') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período no Pernambuco') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_rs <- dados %>% filter(estado == 'Rio Grande do Sul') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período no Rio Grande do Sul') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_rj <- dados %>% filter(estado == 'Rio de Janeiro') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período no Rio de Janeiro') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

dist_sg_sc <- dados %>% filter(estado == 'Santa Catarina') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período em Santa Catarina') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +

```

```

ggeasy::easy_center_title()

dist_sg_sp <- dados %>% filter(estado == 'Sao Paulo') %>%
  ggplot(aes(somagols, y = stat(density), fill = periodo)) +
  geom_histogram(color = 'black', bins = 8) +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição da soma dos gols por período em São Paulo') +
  xlab('Soma dos gols por jogos') + ylab('Densidade') +
  ggeasy::easy_center_title()

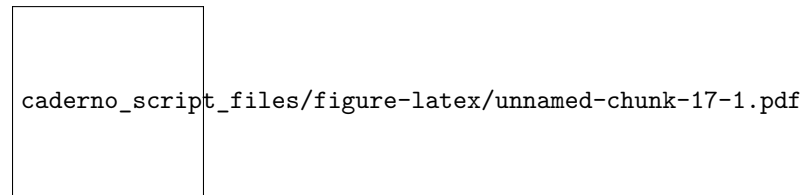
```

Unindo os plots da soma de gols por estado

```

dist_sg1 <- grid.arrange(arrangeGrob(dist_sg_al, dist_sg_ba, dist_sg_ce,
                                     dist_sg_go, dist_sg_mg, dist_sg_pb),
                        top = 'Distribuição da quantidade total de gols por estado e período (1)')

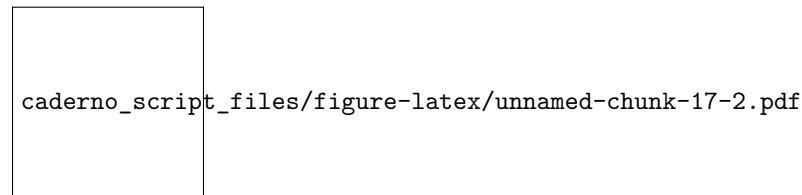
```



```

dist_sg2 <- grid.arrange(arrangeGrob(dist_sg_pr, dist_sg_rj, dist_sg_rs,
                                     dist_sg_sc, dist_sg_sp,
                                     top = 'Distribuição da quantidade total de gols por estado e período (2)'))

```



Plot da quantidade por resultados dos estados “não quentes”

```

dist_frio_gelados2012 <- climao_gelado %>%
  filter(data > '2012-03-20' & data < '2012-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6, color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2012') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2012 <- climao_gelado %>%
  filter(data > '2012-09-22' & data < '2013-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6, color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2012') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

```

```

dist_frio_gelados2013 <- climao_gelado %>%
  filter(data > '2013-03-20' & data < '2013-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2013') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2013 <- climao_gelado %>%
  filter(data > '2013-09-22' & data < '2014-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2013') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_gelados2014 <- climao_gelado %>%
  filter(data > '2014-03-20' & data < '2014-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2014') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2014 <- climao_gelado %>%
  filter(data > '2014-09-22' & data < '2015-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2014') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_gelados2015 <- climao_gelado %>%
  filter(data > '2015-03-20' & data < '2015-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2015') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2015 <- climao_gelado %>%
  filter(data > '2015-09-22' & data < '2016-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +

```

```

ggtitle('Distribuição de resultado nos estados frios no verão de 2015') +
xlab('Quantidade de resultados') + ylab('Ocorrências') +
ggeasy::easy_center_title()

dist_frio_gelados2016 <- climao_gelado %>%
  filter(data > '2016-03-20' & data < '2016-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2016') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2016 <- climao_gelado %>%
  filter(data > '2016-09-22' & data < '2017-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2016') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_gelados2017 <- climao_gelado %>%
  filter(data > '2017-03-20' & data < '2017-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2017') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2017 <- climao_gelado %>%
  filter(data > '2017-09-22' & data < '2018-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2017') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_gelados2018 <- climao_gelado %>%
  filter(data > '2018-03-20' & data < '2018-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2018') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2018 <- climao_gelado %>%

```



```

filter(data > '2018-09-22' & data < '2019-03-20') %>%
ggplot(aes(n, fill = res)) +
geom_bar(alpha = .6,color = 'black') +
scale_fill_brewer(palette = 'Dark2') +
ggtitle('Distribuição de resultado nos estados frios no verão de 2018') +
xlab('Quantidade de resultados') + ylab('Ocorrências') +
ggeasy::easy_center_title()

dist_frio_gelados2019 <- climao_gelado %>%
  filter(data > '2019-03-20' & data < '2019-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2019') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2019 <- climao_gelado %>%
  filter(data > '2019-09-22' & data < '2020-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2019') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_gelados2020 <- climao_gelado %>%
  filter(data > '2020-03-20' & data < '2020-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no inverno de 2020') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_gelados2020 <- climao_gelado %>%
  filter(data > '2020-09-22' & data < '2021-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados frios no verão de 2020') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

```

Plot da quantidade por resultados dos estados mais quentes

```

dist_frio_quentes2012 <- climao_quente %>%
  filter(data > '2012-03-20' & data < '2012-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +

```

```

ggtitle('Distribuição de resultado nos estados quentes no inverno de 2012') +
xlab('Quantidade de resultados') + ylab('Ocorrências') +
ggeasy::easy_center_title()

dist_calor_quentes2012 <- climao_quente %>%
  filter(data > '2012-09-22' & data < '2013-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2012') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2013 <- climao_quente %>%
  filter(data > '2013-03-20' & data < '2013-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2013') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2013 <- climao_quente %>%
  filter(data > '2013-09-22' & data < '2014-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2013') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2014 <- climao_quente %>%
  filter(data > '2014-03-20' & data < '2014-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2014') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2014 <- climao_quente %>%
  filter(data > '2014-09-22' & data < '2015-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2014') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

```

```

dist_frio_quentes2015 <- climao_quente %>%
  filter(data > '2015-03-20' & data < '2015-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2015') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2015 <- climao_quente %>%
  filter(data > '2015-09-22' & data < '2016-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2015') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2016 <- climao_quente %>%
  filter(data > '2016-03-20' & data < '2016-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2016') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2016 <- climao_quente %>%
  filter(data > '2016-09-22' & data < '2017-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2016') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2017 <- climao_quente %>%
  filter(data > '2017-03-20' & data < '2017-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2017') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2017 <- climao_quente %>%
  filter(data > '2017-09-22' & data < '2018-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2017') +

```

```

xlab('Quantidade de resultados') + ylab('Ocorrências') +
ggeasy::easy_center_title()

dist_frio_quentes2018 <- climao_quente %>%
  filter(data > '2018-03-20' & data < '2018-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2018') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2018 <- climao_quente %>%
  filter(data > '2018-09-22' & data < '2019-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2018') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2019 <- climao_quente %>%
  filter(data > '2019-03-20' & data < '2019-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2019') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2019 <- climao_quente %>%
  filter(data > '2019-09-22' & data < '2020-03-20') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2019') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_frio_quentes2020 <- climao_quente %>%
  filter(data > '2020-03-20' & data < '2020-09-22') %>%
  ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no inverno de 2020') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()

dist_calor_quentes2020 <- climao_quente %>%
  filter(data > '2020-09-22' & data < '2021-03-20') %>%

```

```
ggplot(aes(n, fill = res)) +
  geom_bar(alpha = .6,color = 'black') +
  scale_fill_brewer(palette = 'Dark2') +
  ggtitle('Distribuição de resultado nos estados quentes no verão de 2020') +
  xlab('Quantidade de resultados') + ylab('Ocorrências') +
  ggeasy::easy_center_title()
```

Unindo os plots para comparação de resultados ano-ano clima-clima

```
dist_res2012 <- grid.arrange(arrangeGrob(dist_calor_gelados2012, dist_calor_quentes2012,
  dist_frio_gelados2012, dist_frio_quentes2012),
  top = 'Distribuição dos resultados em 2012')
```

caderno_script_files/figure-latex/unnamed-chunk-20-1.pdf

```
dist_res2013 <- grid.arrange(arrangeGrob(dist_calor_gelados2013, dist_calor_quentes2013,
  dist_frio_gelados2013, dist_frio_quentes2013),
  top = 'Distribuição dos resultados em 2013')
```

caderno_script_files/figure-latex/unnamed-chunk-20-2.pdf

```
dist_res2014 <- grid.arrange(arrangeGrob(dist_calor_gelados2014, dist_calor_quentes2014,
  dist_frio_gelados2014, dist_frio_quentes2014),
  top = 'Distribuição dos resultados em 2014')
```

caderno_script_files/figure-latex/unnamed-chunk-20-3.pdf

```
dist_res2015 <- grid.arrange(arrangeGrob(dist_calor_gelados2015, dist_calor_quentes2015,
  dist_frio_gelados2015, dist_frio_quentes2015),
  top = 'Distribuição dos resultados em 2015')
```

caderno_script_files/figure-latex/unnamed-chunk-20-4.pdf

```
dist_res2016 <- grid.arrange(arrangeGrob(dist_calor_gelados2016, dist_calor_quentes2016,
  dist_frio_gelados2016, dist_frio_quentes2016),
```

```
top = 'Distribuição dos resultados em 2016')
```

caderno_script_files/figure-latex/unnamed-chunk-20-5.pdf

```
dist_res2017 <- grid.arrange(arrangeGrob(dist_calor_gelados2017, dist_calor_quentes2017,  
  dist_frio_gelados2017, dist_frio_quentes2017),  
  top = 'Distribuição dos resultados em 2017')
```

caderno_script_files/figure-latex/unnamed-chunk-20-6.pdf

```
dist_res2018 <- grid.arrange(arrangeGrob(dist_calor_gelados2018, dist_calor_quentes2018,  
  dist_frio_gelados2018, dist_frio_quentes2018),  
  top = 'Distribuição dos resultados em 2018')
```

caderno_script_files/figure-latex/unnamed-chunk-20-7.pdf

```
dist_res2019 <- grid.arrange(arrangeGrob(dist_calor_gelados2019, dist_frio_quentes2019,  
  dist_frio_gelados2019, dist_frio_quentes2019),  
  top = 'Distribuição dos resultados em 2019')
```

caderno_script_files/figure-latex/unnamed-chunk-20-8.pdf

```
dist_res2020 <- grid.arrange(arrangeGrob(dist_calor_gelados2020, dist_calor_quentes2020,  
  dist_frio_gelados2020, dist_frio_quentes2020),  
  top = 'Distribuição dos resultados em 2020')
```

caderno_script_files/figure-latex/unnamed-chunk-20-9.pdf

```
## Plot da soma de gols por estação
```

```
somagols_inverno2014 <- medidas_sg_inverno_2014 %>% mutate(estado = fct_reorder(estado, media)) %>%  
  group_by(estado) %>%  
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
```

```

scale_color_viridis(discrete = TRUE) +
ggtitle('Média de gols no inverno por período por estado em 2014') +
xlab('Média de gols') + ylab('Estado') +
ggeasy::easy_center_title()

somagols_verao2014 <- medidas_sg_verao_2014 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2014') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2015 <- medidas_sg_inverno_2015 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2015') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_verao2015 <- medidas_sg_verao_2015 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2015') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2016 <- medidas_sg_inverno_2016 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2016') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_verao2016 <- medidas_sg_verao_2016 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2016') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2017 <- medidas_sg_inverno_2017 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2017') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

```

```

somagols_verao2017 <- medidas_sg_verao_2017 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2017') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2018 <- medidas_sg_inverno_2018 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2018') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_verao2018 <- medidas_sg_verao_2018 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2018') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2019 <- medidas_sg_inverno_2019 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2014') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_verao2019 <- medidas_sg_verao_2019 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2019') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_inverno2020 <- medidas_sg_inverno_2020 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%
  ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2020') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()

somagols_verao2020 <- medidas_sg_verao_2020 %>% mutate(estado = fct_reorder(estado, media)) %>%
  group_by(estado) %>%

```



```
ggplot(aes(media, estado, colour = periodo)) + geom_point() +
  scale_color_viridis(discrete = TRUE) +
  ggtitle('Média de gols no inverno por período por estado em 2020') +
  xlab('Média de gols') + ylab('Estado') +
  ggeasy::easy_center_title()
```

```
gols0 <- grid.arrange(arrangeGrob(somagols_verao2014, somagols_inverno2014,
  somagols_verao2015, somagols_inverno2015),
  top = 'Distribuição da soma dos gols por estação (1)')
```

caderno_script_files/figure-latex/unnamed-chunk-21-1.pdf

```
gols1 <- grid.arrange(arrangeGrob(somagols_verao2016, somagols_inverno2016,
  somagols_verao2017, somagols_inverno2017),
  top = 'Distribuição da soma dos gols por estação (2)')
```

caderno_script_files/figure-latex/unnamed-chunk-21-2.pdf

```
gols2 <- grid.arrange(arrangeGrob(somagols_verao2018, somagols_inverno2018,
  somagols_verao2019, somagols_inverno2019),
  top = 'Distribuição da soma dos gols por estação (3)')
```

caderno_script_files/figure-latex/unnamed-chunk-21-3.pdf

```
gols3 <- grid.arrange(arrangeGrob(somagols_verao2020, somagols_inverno2020),
  top = 'Distribuição da soma dos gols por estação (4)')
```

caderno_script_files/figure-latex/unnamed-chunk-21-4.pdf