---

project:

title: "House Of Excellence"

output-dir: resultados/

lang: pt-BR

format:

pdf:

geometry:

- left=3cm

- right=2cm

- top=3cm

- bottom=2cm

documentclass: estat

toc: true

toc-title: "Sumário"

mainfont: Arial

linkcolor: black

filecolor: black

citecolor: black

urlcolor: black

number-sections: true

fig-width: 6.2

fig-height: 3.66

crossref:

fig-title: Figura

fig-prefix: ""

tbl-title: Tabela

tbl-prefix: ""

title-delim: ":"

execute:

message: false

warning: false

echo: false

error: false

quiet: true

include-in-header:

- file: arquivos\_tex/nome\_dos\_alocados\_e\_do\_cliente.tex # Arrumando para adicionar os autores e o cliente

- file: arquivos\_tex/outros\_detalhes.tex

include-before-body:

- file: arquivos\_tex/logo.tex

editor: visual

---

{r}

library(kableExtra)

library(DT)

library(knitr)

library(pacman)

library(tidyverse)

library(ggplot2)

library(dplyr)

library(readxl)

library(plotly)

athenas = read\_excel("OlimpiadasPS.xlsx",

sheet = "Athina")

rj = read\_excel("OlimpiadasPS.xlsx",

sheet = "Rio de Janeiro")

londres = read\_excel("OlimpiadasPS.xlsx",

sheet = "London")

sydney = read\_excel("OlimpiadasPS.xlsx",

sheet = "Sydney")

pequim = read\_excel("OlimpiadasPS.xlsx",

sheet = "Beijing")

# Organizando o banco de dados

colnames(sydney) = c("Nome", "Sexo", "Idade", "Altura", "Peso", "Pais", "Esporte", "Evento", "Medalha")

colnames(rj) = c("Nome", "Sexo", "Idade", "Altura", "Peso", "Pais", "Esporte", "Evento", "Medalha")

colnames(londres) = c("Nome", "Sexo", "Idade", "Altura", "Peso", "Pais", "Esporte", "Evento", "Medalha")

colnames(pequim) = c("Nome", "Sexo", "Idade", "Altura", "Peso", "Pais", "Esporte", "Evento", "Medalha")

colnames(athenas) = c("Nome", "Sexo", "Idade", "Altura", "Peso", "Pais", "Esporte", "Evento", "Medalha")

olimpiadas = rbind(athenas,rj,londres,sydney,pequim)

estat\_colors <- c(

"#A11D21", "#003366", "#CC9900",

"#663333", "#FF6600", "#CC9966",

"#999966", "#006606", "#008091",

"#041835", "#666666" )

theme\_estat <- function(...) {

theme <- ggplot2::theme\_bw() +

ggplot2::theme(

axis.title.y = ggplot2::element\_text(colour = "black", size = 12),

axis.title.x = ggplot2::element\_text(colour = "black", size = 12),

axis.text = ggplot2::element\_text(colour = "black", size = 9.5)

,

panel.border = ggplot2::element\_blank(),

axis.line = ggplot2::element\_line(colour = "black"),

legend.position = "top",

...

)

return(

list(

theme,

scale\_fill\_manual(values = estat\_colors),

scale\_colour\_manual(values = estat\_colors)

)

)

}

# ANALISE 1

{r}

medalW = subset(olimpiadas, Sexo== "F")

medalWg = subset(medalW, Medalha== "Gold")

medalWs = subset(medalW, Medalha== "Silver")

medalWb = subset(medalW, Medalha== "Bronze")

medalWgsb = rbind(medalWg,medalWs,medalWb)

tabelamedalhasgeral= table(medalWgsb$Pais)

top5medalhas = sort(tabelamedalhasgeral, decreasing = TRUE)

head( top5medalhas, 5)

top5= data.frame(Team = names(head(top5medalhas, 5)),

Medalhas = as.numeric(head(top5medalhas, 5)))

ggplot(top5, aes(x = reorder(Team, -Medalhas), y = Medalhas)) +

geom\_bar(stat = "identity", fill = "#A11D21") +

labs(title = "Top 5 países com mais medalhas entre as mulheres", x = "Países", y = "Número de Medalhas")+

theme\_estat()

# ANALISE 2

{r}

olimpiadas$Peso <- as.numeric(olimpiadas$Peso)

olimpiadas$pesokg = olimpiadas$Peso \* 0.453592

olimpiadas$alturam = olimpiadas$Altura / 100

# Supondo que as colunas 'peso\_kg' e 'altura\_m' existam

olimpiadas$imc <- olimpiadas$pesokg / (olimpiadas$alturam^2)

judo = subset(olimpiadas, Esporte == "Judo")

badminton = subset(olimpiadas, Esporte == "Badminton")

futebol = subset(olimpiadas, Esporte == "Football")

ginastica = subset(olimpiadas, Esporte == "Gymnastics")

atletismo = subset(olimpiadas, Esporte == "Athletics")

espotes=rbind(judo,badminton,futebol,ginastica,atletismo)

espotes$Esporte[espotes$Esporte == "Football"] = "Futebol"

espotes$Esporte[espotes$Esporte == "Gymnastics"] = "Ginastica"

espotes$Esporte[espotes$Esporte == "Athletics"] = "Atletismo"

ggplot(espotes) +

aes(x = reorder(Esporte, imc, FUN = median), y = imc) +

geom\_boxplot(aes(fill = "Mediana"), width = 0.5) +

stat\_summary(

aes(shape = "Média"), fun = "mean", geom = "point", size = 3, fill = "white"

) +

labs(x = "Esporte", y = "IMC", fill = "", shape = "") +

scale\_shape\_manual(values = c(23)) +

scale\_fill\_manual(values = c("Mediana" = "#A11D21")) +

theme\_estat()

# ANALISE 3

{r}

ouro = subset(olimpiadas, Medalha == "Gold")

prata = subset(olimpiadas, Medalha == "Silver")

bronze = subset(olimpiadas, Medalha == "Bronze")

medalhaalltime = rbind(ouro,prata,bronze)

view(medalhaalltime)

Top3alltime = table(medalhaalltime$Nome)

top3medalhas = sort(Top3alltime, decreasing = TRUE)

top3\_nomes = names(head(top3medalhas, 3))

Top3alltimeOLympic = data.frame(

nome = top3\_nomes,

medalhas = as.numeric(head(top3medalhas, 3))

)

Top3alltimeOLympic$esporte = medalhaalltime$Esporte[match(Top3alltimeOLympic$nome, medalhaalltime$Nome)]

head(Top3alltimeOLympic, 3)

ggplot(Top3alltimeOLympic, aes(x = nome, y = medalhas)) +

geom\_bar(stat = "identity", fill = "#A11D21") +

labs(

title = "TOP 3 MEDALHISTAS SYDNEY 2000 - RIO DE JANEIRO 2016",

x = "",

y = "QUANTIDADE DE MEDALHAS"

) +

theme\_estat()

{r}

michaelphelps = subset(olimpiadas, Nome == "Michael Fred Phelps, II")

natalie = subset(olimpiadas, Nome == "Natalie Anne Coughlin (-Hall)")

ryan = subset(olimpiadas, Nome == "Ryan Steven Lochte")

michaelphelps$Medalha[is.na(michaelphelps$Medalha)] = "Out podium"

natalie$Medalha[is.na(natalie$Medalha)] <- "Out podium"

ryan$Medalha[is.na(ryan$Medalha)] <- "Out podium"

medalha1 <- michaelphelps %>%

count(Medalha) %>%

mutate(

freq = n,

relative\_freq = round((freq / sum(freq)) \* 100, 1),

freq = gsub("\\.", ",", relative\_freq) %>% paste("%", sep = ""),

label = str\_c(n, " (", freq, ")") %>% str\_squish()

)

ggplot(medalha1) +

aes(x = fct\_reorder(Medalha, freq, .desc = TRUE), y = n, label = label) +

geom\_bar(stat = "identity", fill = "#A11D21", width = 0.7) +

geom\_text(

position = position\_dodge(width = .9),

vjust = -0.5,

size = 3

) +

labs(title = " MEDALHA MICAHEL PHELPS", x = "Cor das medalhas", y = "Quantas medalhas ganhou") +

theme\_estat()

ggsave("colunas-uni-freq.pdf", width = 158, height = 93, units = "mm")

medalha2 <- natalie %>%

count(Medalha) %>%

mutate(

freq = n,

relative\_freq = round((freq / sum(freq)) \* 100, 1),

freq = gsub("\\.", ",", relative\_freq) %>% paste("%", sep = ""),

label = str\_c(n, " (", freq, ")") %>% str\_squish()

)

ggplot(medalha2) +

aes(x = fct\_reorder(Medalha, freq, .desc = TRUE), y = n, label = label) +

geom\_bar(stat = "identity", fill = "#A11D21", width = 0.7) +

geom\_text(

position = position\_dodge(width = .9),

vjust = -0.5,

size = 3

) +

labs(title = " MEDALHA NATALIE ANNE", x = "Cor das medalhas", y = "Quantas medalhas ganhou") +

theme\_estat()

medalha3 <- ryan %>%

count(Medalha) %>%

mutate(

freq = n,

relative\_freq = round((freq / sum(freq)) \* 100, 1),

freq = gsub("\\.", ",", relative\_freq) %>% paste("%", sep = ""),

label = str\_c(n, " (", freq, ")") %>% str\_squish()

)

ggplot(medalha3) +

aes(x = fct\_reorder(Medalha, freq, .desc = TRUE), y = n, label = label) +

geom\_bar(stat = "identity", fill = "#A11D21", width = 0.7) +

geom\_text(

position = position\_dodge(width = .9),

vjust = -0.5,

size = 3

) +

labs(title = " MEDALHA RYAN STEVEN", x = "Cor das medalhas", y = "Quantas medalhas ganhou") +

theme\_estat()

### ANALISE 4

{r}

ggplot(olimpiadas, aes(x = Peso, y = Altura)) +

geom\_point(color = "#A11D21") +

geom\_hline(aes(yintercept = mean(Altura, na.rm = TRUE), color = "Média de Altura"), linetype = "dashed", size = 1) +

geom\_vline(aes(xintercept = mean(Peso, na.rm = TRUE), color = "Média de Peso"), linetype = "dashed", size = 1) +

geom\_smooth(method = "lm", aes(color = "Linha de Regressão")) +

labs(

title = "Altura x Peso",

x = "Peso",

y = "Altura",

color = "Legenda"

) +

scale\_color\_manual(values = c("Média de Altura" = "#003366", "Média de Peso" = "#CC9900", "Linha de Regressão" = "#663333"))+

scale\_y\_continuous(breaks = seq(min(olimpiadas$Altura, na.rm = TRUE), max(olimpiadas$Altura, na.rm = TRUE), by = 10)) +

theme\_estat()