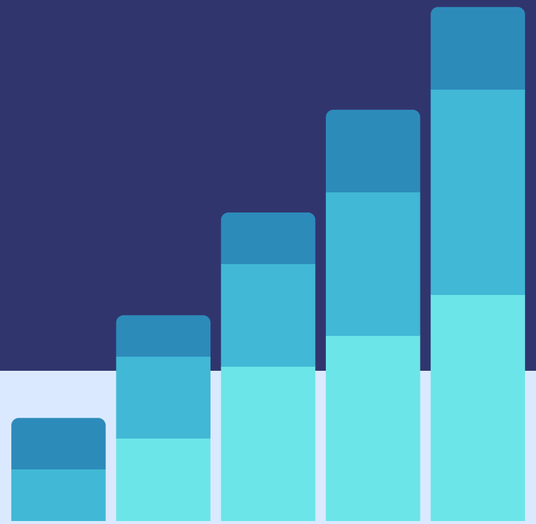
# Aula 03

## Análise exploratória de dados

# Repositório do github do minicurso:

# github.com/gvheisler/minicurso-R

# ANÁLISE EXPLORATÓRIA DE DADOS



- Estatísticas descritivas
- Média, mediana, moda, desvio padrão
- Visualização de dados
- Histogramas, boxplots, scatter plots
- Limpeza de dados
- Tratamento de valores ausentes
- Transformações de dados

PET CC

# NA

Em R, dados NA representam valores ausentes ou indefinidos em um dataset. Eles indicam que a informação para uma determinada posição está indisponível ou não foi registrada

| | Nome | Idade | Salario | Departamento | Ano_de_Contratacao | Genero |
|---|---|---|---|---|---|---|
| **1** | Ana | 28 | 3500 | Marketing | 2015 | Feminino |
| **2** | Bruno | 34 | 4800 | TI | *NA* | masculino |
| **3** | Carla | 29 | *NA* | TI | 2018 | Feminino |
| **4** | Daniel | *NA* | 5200 | Financerio | 2010 | Masculino |
| **5** | Eduarda | 31 | 4500 | Marketing | 2014 | Fememino |
| **6** | Felipe | 25 | 3200 | Financeiro | 2019 | Masculino |

https://gvheisler.github.io/ds/funcionarios.csv

# NA

Existem maneiras diferentes de lidar com NAs

- Remover valores ausentes:
  - Usar na.omit() ou na.exclude() para excluir linhas que contenham NA.
- Substituir valores ausentes:
  - Substituir por valores plausíveis (deve-se ser analisado caso a caso). Pode ser por média, mediana, 0, manualmente.
- Ignorar durante cálculos.

# Estatística

Diversas funções estatísticas estão disponíveis em R.
No material do curso está disponibilizado um link que mostra algumas delas

# Limpeza de dados

Muitas vezes é necessário fazer uma limpeza nos dados que temos.
A limpeza pode ter que ser feita de muitas maneiras diferentes.

| | id | nome | idade | departamento | salario | data_admissao | cidade |
|---|---|---|---|---|---|---|---|
| **1** | 1 | Maria Silva | 35.0 | Vendas | 4500.5 | 2015-06-22 | Sao Paulo |
| **2** | 2 | João Souza | 28.0 | Tecnologia | 5300.0 | 2019-03-12 | São paulo |
| **3** | 3 | Pedro Lima | 45.0 | RH | 5200.75 | 2011-08-05 | Rio de Janeiro |
| **4** | 4 | Ana Costa | | Financeiro | 4800.0 | 2018-01-18 | RiodeJaneiro |
| **5** | 5 | Lucas Pereira | 32.0 | Vendas | | 2017-05-27 | Sao Paulo |

# Limpeza de dados

Diferenciação entre letras maiúsculas e minúsculas, acentos, espaços no meio das palavras, etc.

Também é importante manter as colunas no tipo certo. Uma coluna numérica não pode ter palavras nela, e etc.

# Prática

- Carregue o arquivo https://gvheisler.github.io/ds/funcionarios.csv
- Arrume a coluna de Idade
- Arrume a coluna de departamento
- Arrume a coluna de salário
- Faça um gráfico de barras da média de salário por departamento
- Histograma do ano de contratação

# Prática

```r
media_salario <- aggregate(Salario ~ Departamento, data = df, FUN = mean)

barplot(media_salario$Salario,
        names.arg = media_salario$Departamento,
        main = "Média Salarial por Setor",
        ylab = "Média Salarial",
        xlab = "",
        col = "lightblue",
        las = 2,
        cex.names = 0.8)
```

# Prática

```r
df$Ano_de_Contratacao <- as.numeric(df$Ano_de_Contratacao)
df_ano_Contratacao <- df$Ano_de_Contratacao[!is.na(df$Ano_de_Contratacao)]

hist(df_ano_Contratacao,
     main = "Distribuição do Ano de Contratação",
     xlab = "Ano de Contratação",
     ylab = "Frequência",
     col = "lightgreen",
     border = "black")
```

# Prática

- Carregue o arquivo https://gvheisler.github.io/ds/vendas.csv
- Arrume a coluna de Data
- Faça um gráfico de linhas da quantidade de cada produto vendida
- Gráfico de linha da quantidade total de produtos vendidos por dia

```r
df <- read.csv(url("https://gvheisler.github.io/ds/vendas.csv"))

df$Data <- as.Date(df$Data, format = "%Y-%d-%m")

plot(x = df$Data[which(df$Produto=="Produto A")],
     y = df$Vendas[which(df$Produto=="Produto A")],
     type = 'l', ylim = c(50,200))
lines(x = df$Data[which(df$Produto=="Produto B")],
      y = df$Vendas[which(df$Produto=="Produto B")], col = 'red')
lines(x = df$Data[which(df$Produto=="Produto C")],
      y = df$Vendas[which(df$Produto=="Produto C")], col = 'blue')
```

```r
library(dplyr)

vendas_por_dia <- df %>%
  group_by(Data) %>%
  summarise(Vendas_Totais = sum(Vendas))

plot(vendas_por_dia, type ='l')
```

# Sua vez:

Leia o arquivo https://gvheisler.github.io/ds/funcionarios2.csv

Faça as mudanças que achar necessárias
Faça gráficos

MINICURSO

# BÁSICO DE R

PET CC