



INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Data Science com R - II

## Primeira Parte

Pedro Sousa e João Lopes

Setembro 2024





INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Programa





INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Programa



- **Transformação dos dados**
- **Exploração dos dados**
- **Modelação**
- Relatórios e apresentações
- Comunicação





# Programa



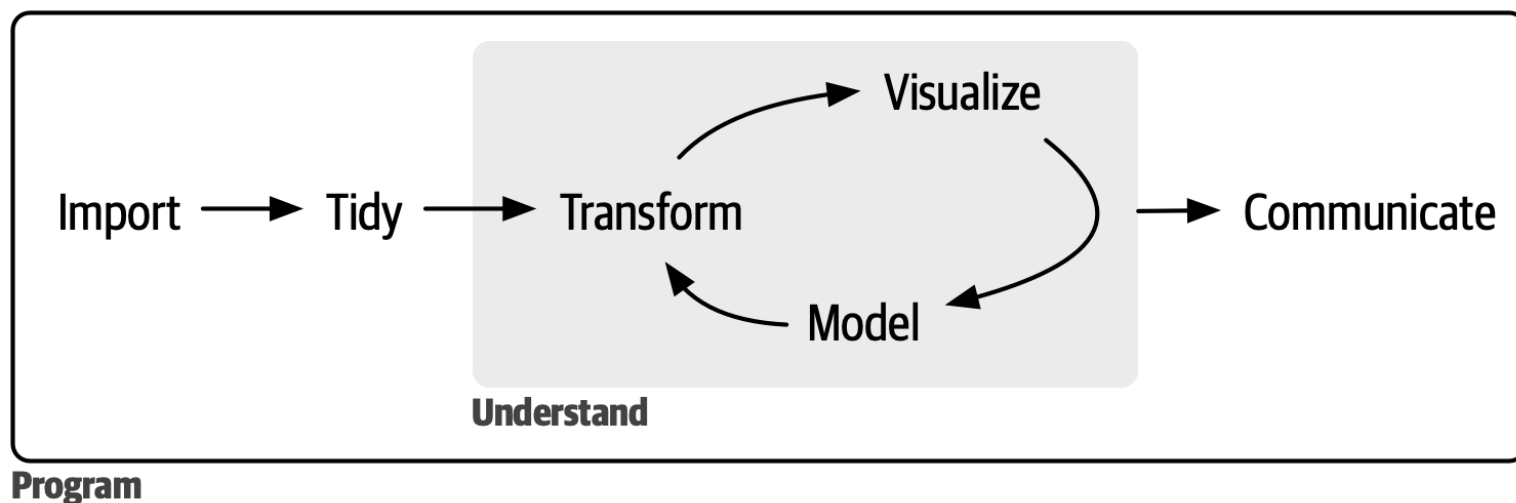
- **Transformação dos dados**
  - Vetores numéricos
  - Fatores
  - Vetores lógicos
- **Exploração dos dados**
  - Variação
  - Valores em falta
  - Covariação
  - Padrões e Modelos
- **Modelação**
  - Construir modelo simples
  - Construir modelo com workflow()
  - Construir vários modelos com workflow()



# Recapitulação



## » Esquema geral



H. Wickham M. Çetinkaya-Rundel & G. Grolemund (2023)

# Dia 1 - Transformação





# Dia 1 - Transformação



- **Vetores numéricos** → 09:30 - 11:00 e 11:15 - 12:30
  - Contagens (c/ exercícios)
  - Transformações numéricas (c/ exercícios)
  - Transformações genéricas (c/ exercícios)
  - Estatísticas descritivas (c/ exercícios)
- **Fatores** → 14:00 - 15:00
  - Operações básicas
  - Base de dados gss\_cat (c/ exercícios)
  - Alterar ordem dos fatores (c/exercícios)
  - Alterar os fatores (c/ exercícios)
- **Vetores lógicos** → 15:15 - 17:00
  - Comparações (c/ exercícios)
  - Álgebra booleana (c/ exercícios)
  - Sumarização (c/ exercícios)
  - Transformações condicionais (c/ exercícios)



# Recapitulação



## » Pacote **dplyr**

- `filter()`      Selecionar linhas (i.e. observações);
- `arrange()`      Ordenar linhas (i.e. observações);
- `select()`      Selecionar colunas (i.e. variáveis);
- `mutate()`      Criar novas colunas (i.e. variáveis);
- `summarize()`      Calcular estatísticas descritivas.
- `group_by()`      Criar grupos de observações para manipulação.

<https://dplyr.tidyverse.org/reference/index.html>

<https://rstudio.github.io/cheatsheets/html/data-transformation.html>







## » Pacote dplyr

```
library("tidyverse")

?diamonds

diamonds |>
  select(price, carat, cut) |>
  filter(carat < 3) |>
  mutate(lprice = log10(price)) |>
  group_by(cut) |>
  summarize(
    mean_lprice = mean(lprice)
    mean_carat = mean(carat)
  ) |>
  arrange(desc(mean_price))
```

```
#use data "diamonds"
#select "price", "carat" and "cut"
#filter for smaller diamonds
#create variable "lprice"
#group by "cut"

#calculate mean of "lprice"
#calculate mean of "carat"

#arrange by "mean_lprice"
```



# Recapitulação



## » Pacote dplyr

```
# A tibble: 5 × 3
  cut      mean_lprice mean_carat
<ord>      <dbl>      <dbl>
1 Fair          3.51          1.03
2 Premium        3.45          0.889
3 Good           3.41          0.847
4 Very Good      3.39          0.806
5 Ideal          3.32          0.702
```



INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Dia 2 - Exploração





# Dia 2 - Exploração



- **Variação** → 09:30 - 11:00
  - Valores típicos (c/ exercícios)
  - Valores invulgares
- **Valores em falta** → 11:15 - 12:30
  - Explícitos (c/ exercícios)
  - Implícitos (c/ exercícios)
  - Fatores e grupos vazios
- **Covariação** → 14:00 - 16:00
  - Uma categórica e uma numérica (c/ exercícios)
  - Duas categóricas (c/ exercícios)
  - Duas numéricas (c/ exercícios)
- **Padrões e modelos** → 16:15 - 17:00



# Recapitulação



## » Pacote `ggplot2`

- Data;
- Aesthetic mapping (**aes**);
- Geometric object (**geom**);
- Statistical transformation (**stat**);
- Scale;
- Themes.

<https://ggplot2.tidyverse.org/reference/index.html>

<https://rstudio.github.io/cheatsheets/html/data-visualization.html>



# Recapitulação



## » Pacote ggplot2

```
set.seed(1984)

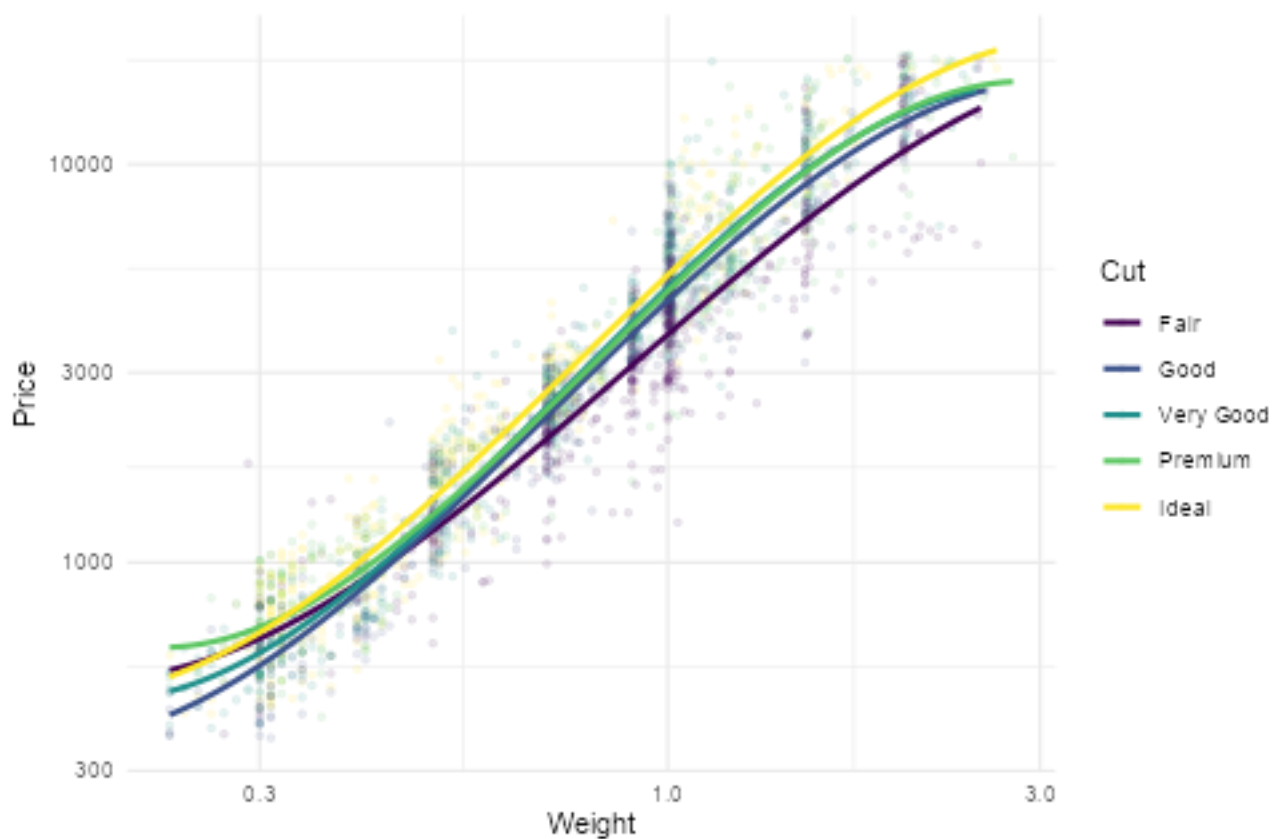
diamonds |>
  filter(carat < 3) |>
  slice_sample(n = 500, by = cut) |>
  ggplot(aes(x = carat, y = price, color = cut)) +
  geom_point(alpha = 0.1, size = 1) +
  stat_smooth(
    method = "lm",
    formula = "y ~ x + I(x^2) + I(x^3)",
    se = FALSE) +
  scale_x_continuous(trans = "log10") +
  scale_y_continuous(trans = "log10") +
  labs(x = "Weight", y = "Price", color = "Cut") +
  theme_minimal()
```

#use data "diamonds"  
#filter for smaller diamonds  
#sample for 500 obs per "cut"  
#aesthetics mapping  
#geometric object  
#statistical transformation  
#scale for x-axis  
#scale for y-axis  
#scale for labels  
#change theme

# Recapitulação



## » Pacote ggplot2





INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Dia 3 - Modelação







# Dia 3 - Modelação



- **Construir modelo simples** → 09:30 - 11:00
  - Explorar dados
  - Ajustar modelo (c/ exercícios)
  - Usar modelo para previsão (c/ exercícios)
- **Construir modelo com *workflow*** → 11:15 - 12:30
  - Explorar dados
  - Dividir dados
  - Criar *workflow* (c/ exercícios)
  - Ajustar modelo (c/ exercícios)
  - Avaliar modelo (c/ exercícios)
- **Construir vários modelos com *workflow*** → 14:00 - 15:00 e 15:15 - 17:00
  - Explorar dados
  - Dividir dados
  - Criar *workflow* e ajustar modelo 1 (c/ exercícios)
  - Criar *workflow* e ajustar modelo 2 (c/ exercícios)
  - Avaliar o melhor modelo (c/ exercícios)



# Recapitulação



## » Pacote `stats`: funções `lm()` e `summary()`

- Data;
- Formula (e.g. `formula()`);
- Fit model (eg. `lm()`, `glm()`, `aov()`, ...);
- Extract parameters (eg. `residuals()`, `predicted()`, `coef()`, ...);
- Testing assumptions (eg. `plot()`, ...);
- Evaluate model (eg. `summary()`, `AIC()`, `logLik()`, ...).

<https://www.datacamp.com/tutorial/linear-regression-R>

<https://rpubs.com/abigailpayne/743827>



# Recapitulação



## » Pacote `stats`: funções `lm()` e `summary()`

```
diamonds2 <- diamonds |>                                #use data "diamonds"
  filter(
    (x > 0) & (y > 0 & y < 20) & (z > 0 & z < 10), #filter out outliers
    carat < 3) |>                                         #filter in smaller diamonds
  slice_sample(n = 1000) |>                             #sample for 1000 observations
  select(price, carat, cut) |>                          #select "price", "carat" and "cut"
  mutate(
    lprice = log10(price),                               #create variable "lprice"
    lcarat = log10(carat),                               #create variable "lcarat"
    fct_cut = factor(cut, ordered = FALSE))             #make variable "cut" into factor

mod1 <- formula("lprice ~ lcarat + fct_cut")             #specify model
res1 <- lm(mod1, data = diamonds2)                     #fit model to data

summary(res1)                                           #get summary
```

# Recapitulação



## » Pacote `stats`: funções `lm()` e `summary()`

### Residuals:

| Min      | 1Q       | Median  | 3Q      | Max     |
|----------|----------|---------|---------|---------|
| -0.33696 | -0.06961 | 0.00255 | 0.07358 | 0.36898 |

### Coefficients:

|                  | Estimate | Std. Error | t value | Pr(> t ) |     |
|------------------|----------|------------|---------|----------|-----|
| (Intercept)      | 3.56450  | 0.02020    | 176.449 | < 2e-16  | *** |
| lcarat           | 1.70134  | 0.01384    | 122.938 | < 2e-16  | *** |
| fct_cutGood      | 0.07818  | 0.02347    | 3.331   | 0.000896 | *** |
| fct_cutVery Good | 0.09259  | 0.02164    | 4.279   | 2.06e-05 | *** |
| fct_cutPremium   | 0.08857  | 0.02144    | 4.131   | 3.91e-05 | *** |
| fct_cutIdeal     | 0.12437  | 0.02111    | 5.891   | 5.24e-09 | *** |

Multiple R-squared: 0.9396, Adjusted R-squared: 0.9393

F-statistic: 3093 on 5 and 994 DF, p-value: < 2.2e-16

# Recapitulação



## » Pacote **performance**

MLR.1 O modelo é linear nos parâmetros;

MLR.2 Amostra aleatória (e.g. não há outliers, valores omissos aleatórios);

MLR.3 Não há multicolinearidade entre preditores;

MLR.4 Erro com valor esperado zero dado qualquer valor dos preditores;

MLR.5 Erro com variância constante dado qualquer valor dos preditores;

MLR.6 Erro é independente dos preditores e tem distribuição normal.

Wooldridge J, Introductory Econometrics: A Modern Approach, 7 ed. Thomson



## » Pacote performance

```
library("performance")

check_model(res1, check = c(
  #MLR.1 The population model is linear in the parameters
  "linearity",
  #MLR.3 No multicollinearity between predictors
  "vif",
  #MLR.5 The error has constant variance given any values of the parameters
  "homogeneity",
  #MLR.6 The error is independent of the predictors and is normally distributed
  "qq"
))
```

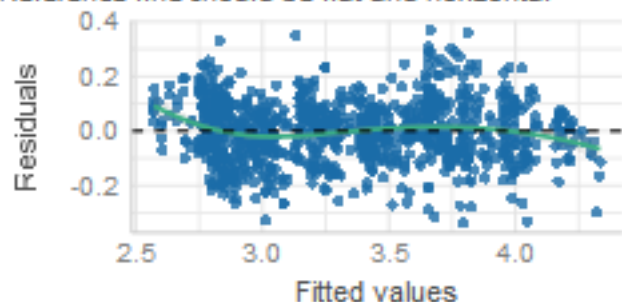
# Recapitulação



## » Pacote performance

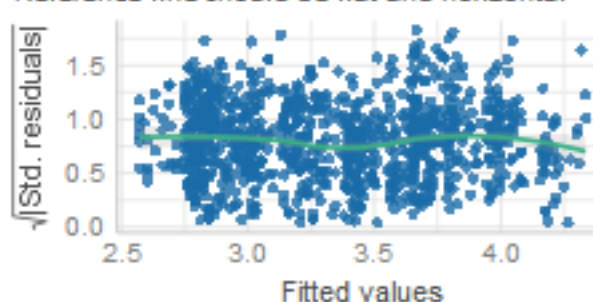
### Linearity

Reference line should be flat and horizontal



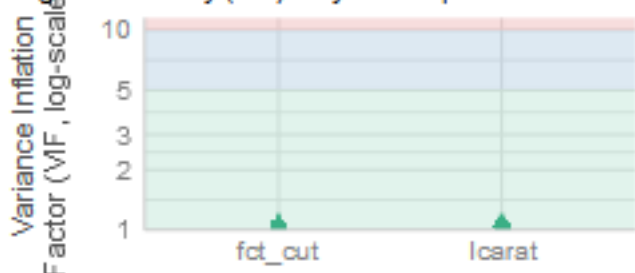
### Homogeneity of Variance

Reference line should be flat and horizontal



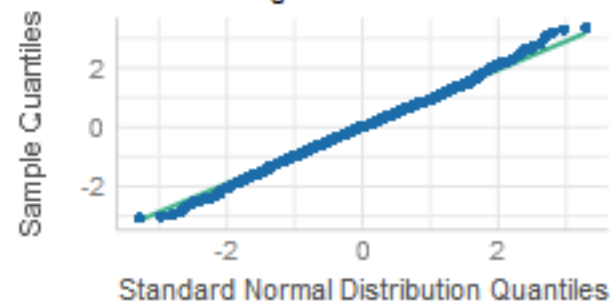
### Collinearity

High collinearity (VIF) may inflate parameter uncertainty



### Normality of Residuals

Dots should fall along the line



Low (< 5)



## » Regressão linear simples

```
beta0 <- -1.6
beta1 <- 0.03
tb_lm <- tibble(
  x = runif(20, min = 18, max = 60),
  y = beta0 + beta1*x + rnorm(20, mean = 0, sd = 0.1)
)

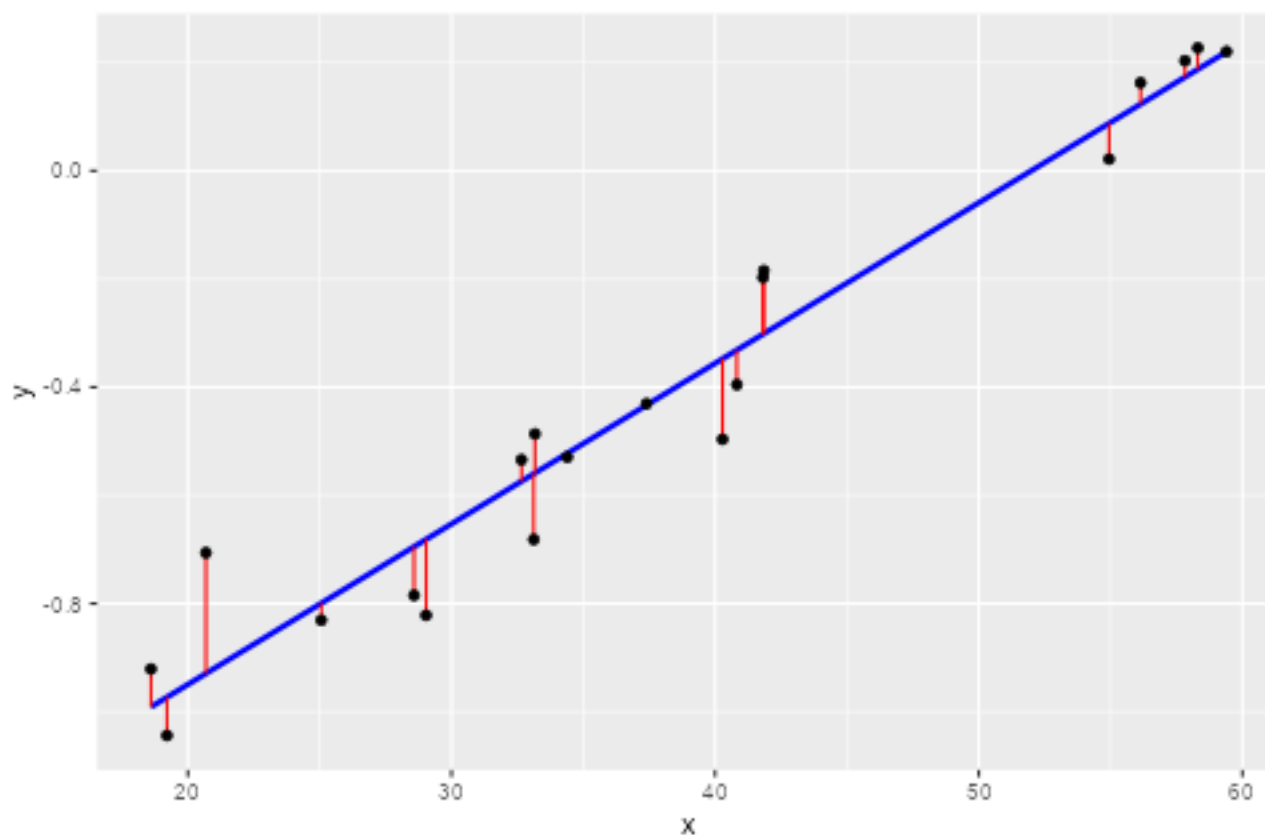
res_lm <- lm(y ~ x, data = tb_lm)

tb_lm |>
  mutate(preds = predict(res_lm), resid = residuals(res_lm)) |>
  ggplot(aes(x = x, y = y)) + geom_point() +
  stat_smooth(method = "lm", formula = "y ~ x", se = FALSE, color = "blue") +
  geom_segment(aes(xend = x, yend = preds), color = "red")
```





## » Regressão linear simples



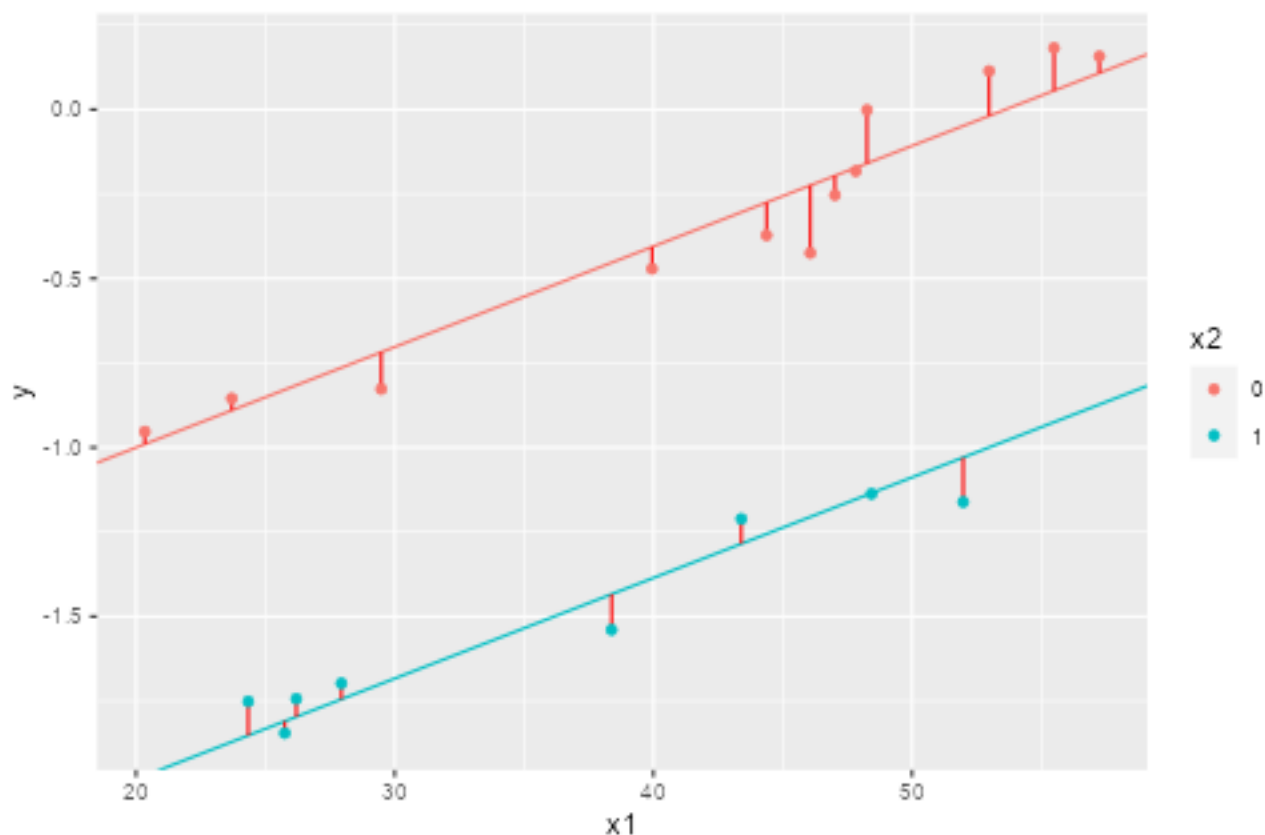


## » Regressão linear múltipla

```
beta0 <- -1.6; beta1 <- 0.03; beta2 <- 0.6
tb_lm2 <- tibble(
  x1 = runif(20, min = 18, max = 60),
  x2 = rbinom(20, size = 1, prob = 0.5),
  y = beta0 + beta1*x1 + (beta0 + beta2)*x2 + rnorm(20, mean = 0, sd = 0.1)
) |>
  mutate(x2 = factor(x2))
res_lm2 <- lm(y ~ x1 + x2, data = tb_lm2)
v1 <- res_lm2$coef
tb_lm2 |>
  mutate(preds = predict(res_lm2), resid = residuals(res_lm2)) |>
  ggplot(aes(x = x1, y = y, color = x2)) + geom_point() +
  geom_abline(slope = v1[2], intercept = v1[1], color = "#F8766D") +
  geom_abline(slope = v1[2], intercept = sum(v1[c(1, 3)]), color = "#00BFC4") +
  geom_segment(aes(xend = x1, yend = preds), color = "red")
```



## » Regressão linear múltipla



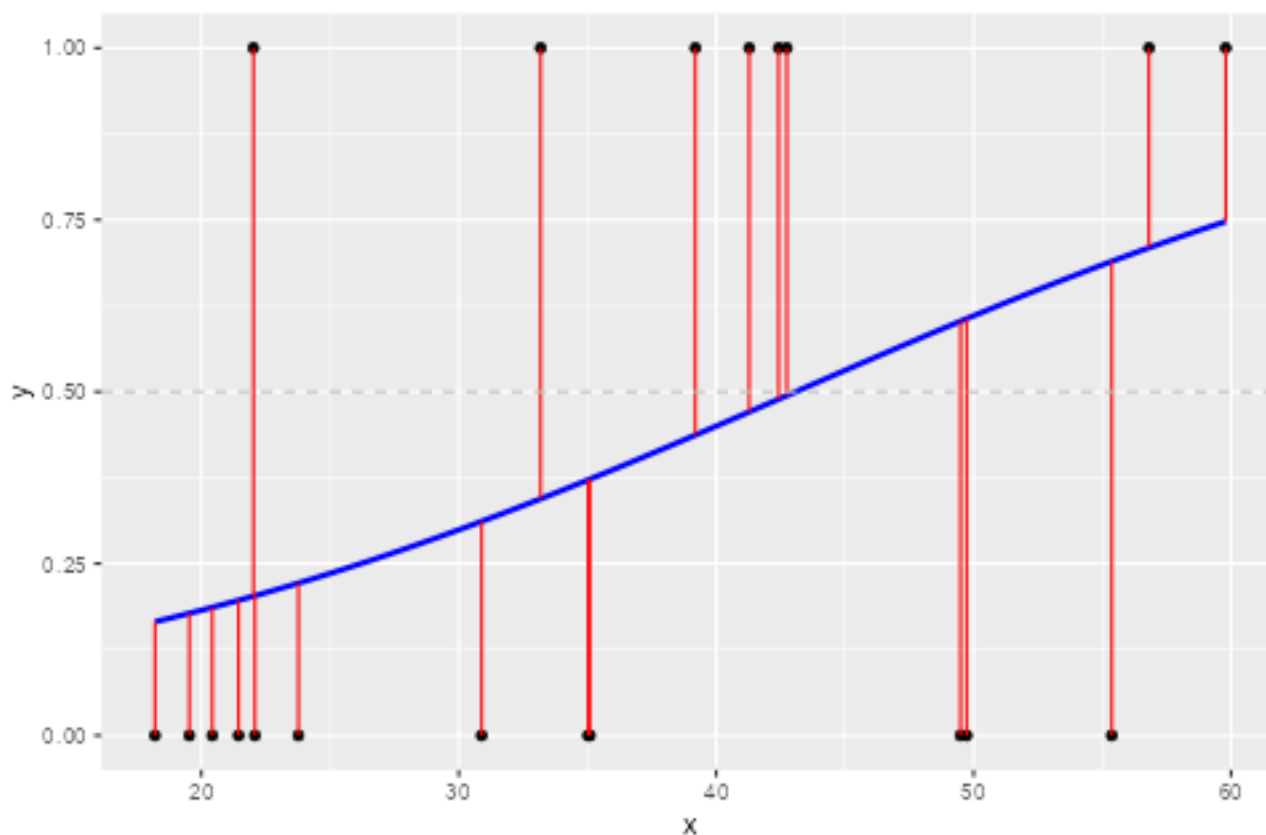


## » Regressão logística

```
beta0 <- -1.6
beta1 <- 0.03
tb_glm <- tibble(
  x = runif(20, min = 18, max = 60),
  pi_x = exp(beta0 + beta1*x) / (1 + exp(beta0 + beta1*x)),
  y = rbinom(20, size = 1, prob = pi_x)
)
res_glm <- glm(y ~ x, binomial(link="logit"), data = tb_glm)
tb_glm |>
  mutate(preds = predict(res_glm, type="resp"), resid = residuals(res_glm)) |>
  ggplot(aes(x = x, y = y)) + geom_point() +
  stat_smooth(method = "glm", formula = "y ~ x", se = FALSE, color = "blue",
    method.args = list(family = "binomial")) +
  geom_segment(aes(xend = x, yend = preds), color = "red") +
  geom_hline(yintercept = 0.5, linetype = "dashed", color = "gray")
```



## » Regressão logística





## » Árvores de decisão

```
Library("rpart")

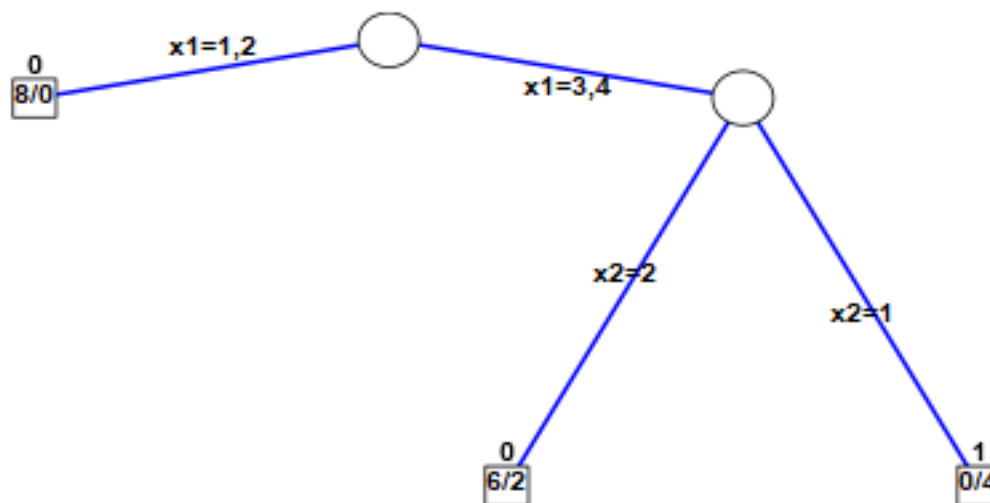
tb_rf <- tibble(
  x1 = ceiling(runif(20, min = 0, max = 4)),
  x2 = ceiling(runif(20, min = 0, max = 3)),
  x3 = ceiling(runif(20, min = 0, max = 3)),
  x = x1 - x2 - x3,
  pi_x = exp(x) / (1 + exp(x)),
  y = rbinom(20, size = 1, prob = pi_x)
) |> mutate(across(c(x1:x3, y), as.factor))

res_rd <- rpart(y ~ x1 + x2 + x3, data = tb_rf, method = "class",
  control = rpart.control(minsplit = 5))

plot(res_rd, branch = 0, margin = 0.02, branch.lwd = 2, branch.col = "blue")
text(res_rd, minlength = 2, use.n = TRUE, fancy = TRUE, fwidth = 1.5,
  fheight = 1, cex = 0.8, font = 2)
```



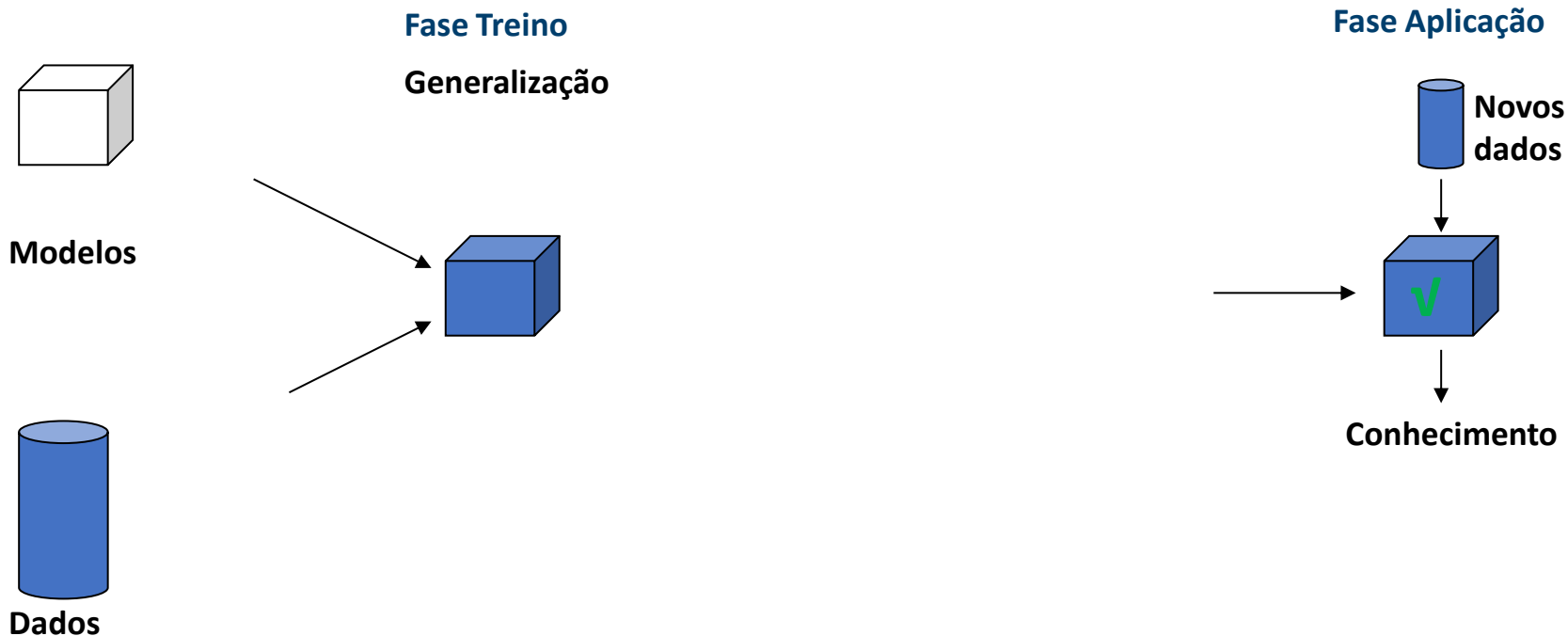
## » Árvores de decisão



# Avaliação de modelos



## » Visão Geral

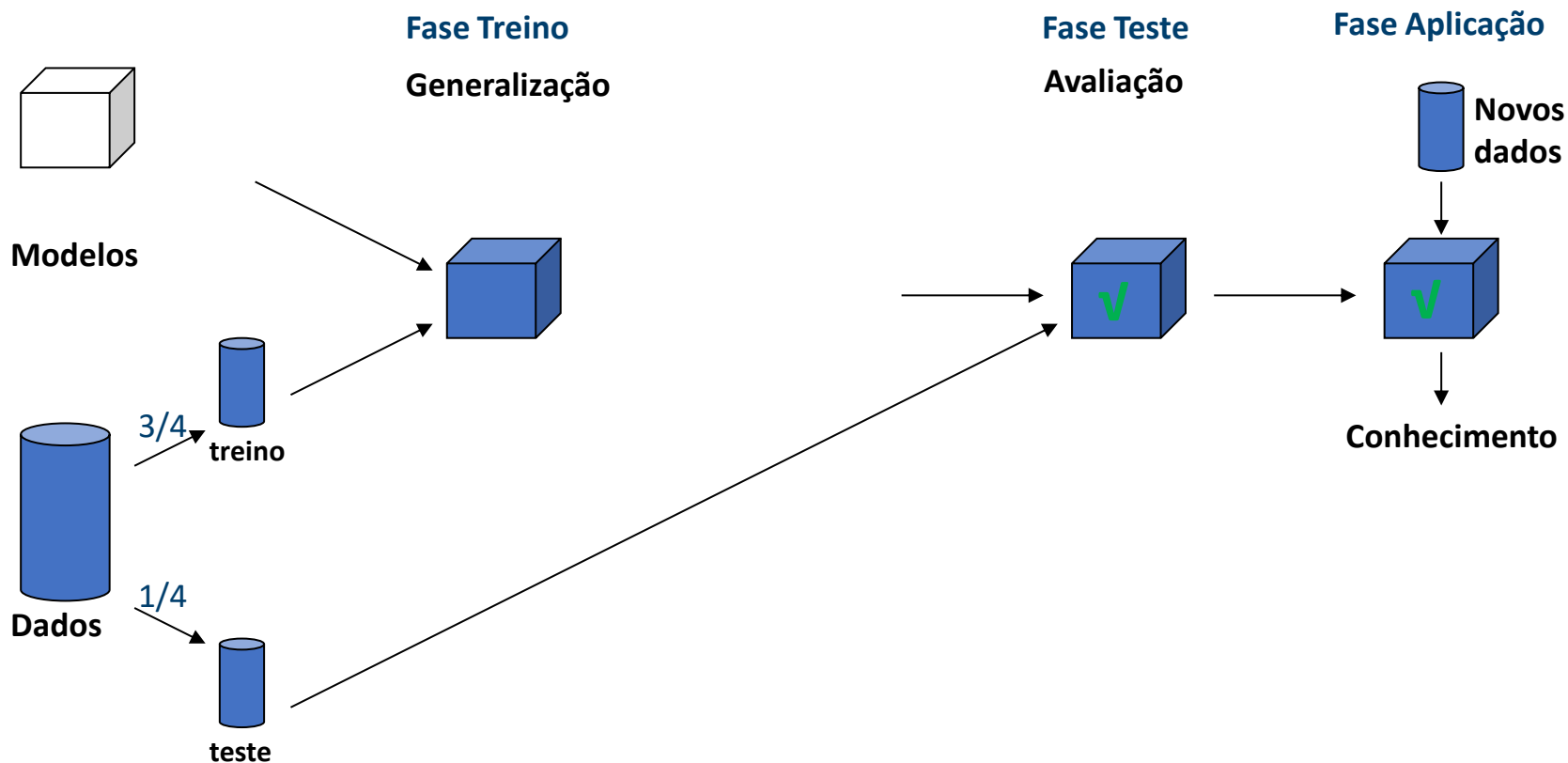




# Avaliação de modelos



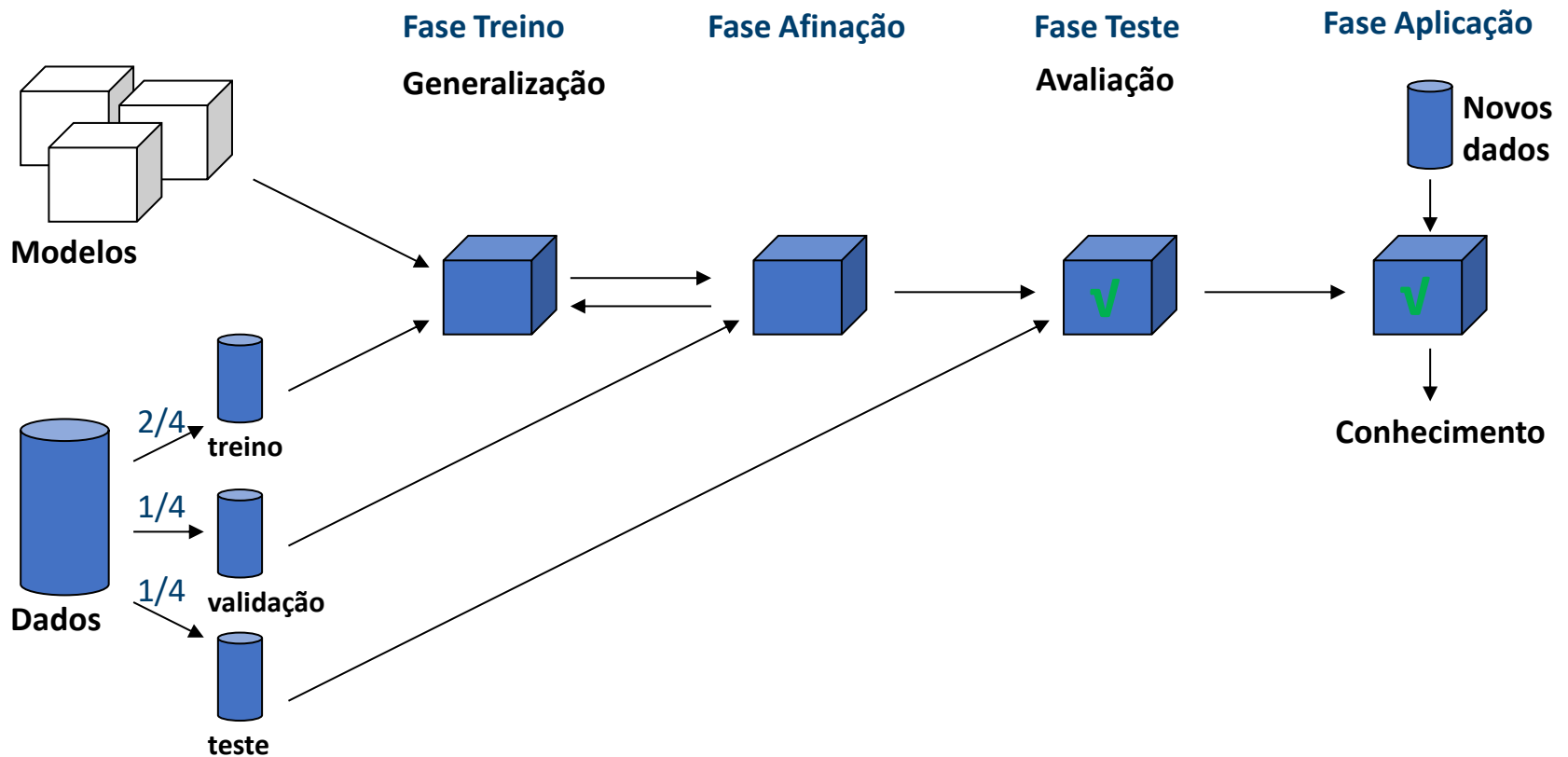
## » Visão Geral



# Avaliação de modelos



## » Visão Geral



# Avaliação de modelos



## » Métricas: variável contínua

Root Mean Square Error (RMSE)

$$\text{RMSE} = \sqrt{\sum_{i=1}^n (\hat{y} - y)^2 / n}$$

Mean Absolute Error (MAE)

$$\text{MAE} = \sum_{i=1}^n |\hat{y} - y| / n$$

## » Métricas: variável categórica (binária)

True Positive Rate (Sensitivity)

$$= \text{TP} / \text{P}$$

False Positive Rate (1 - Specificity)

$$= \text{FP} / \text{N} = 1 - \text{TN} / \text{N}$$

|      |          | Predicted (p = 0.50) |          |
|------|----------|----------------------|----------|
|      |          | Positive             | Negative |
| Real | Positive | True P.              | False N. |
|      | Negative | False P.             | True N.  |

# Avaliação de modelos



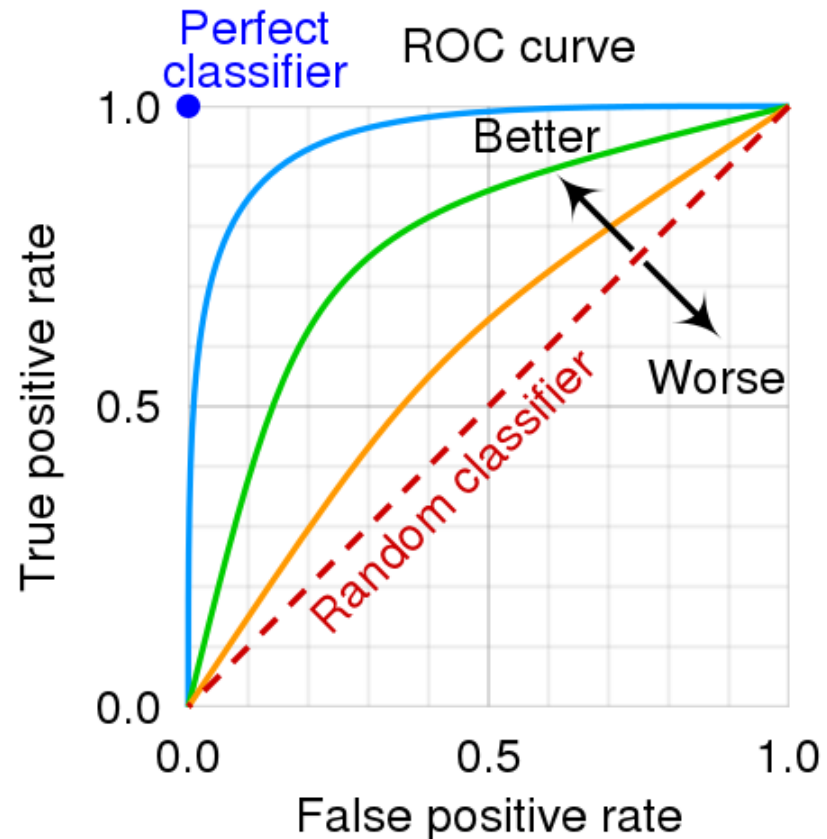
## » Curva de ROC (Receiver Operating Characteristics)

Trade-off:

Sensitivity vs. Specificity

True Positive Rate (Sensitivity)  
=  $TP/P$

False Positive Rate (1 - Specificity)  
=  $FP/N = 1 - TN/N$



M. Thoma (2018)



INSTITUTO NACIONAL DE ESTATÍSTICA  
STATISTICS PORTUGAL

# Informações finais



# Informações finais



## » Gestão de projectos

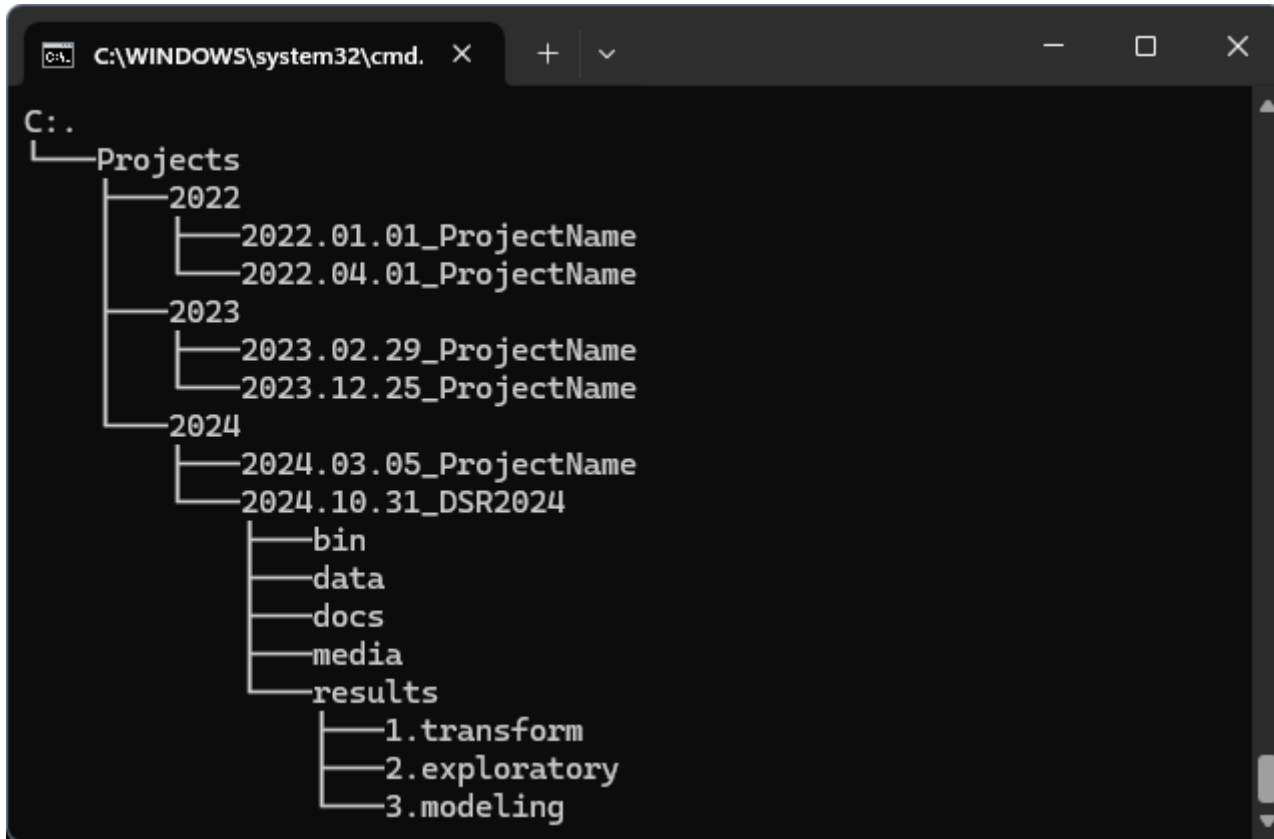
- Estrutura de pastas;
- Ficheiro README (e nomeação de ficheiros);
- Controlo de versões (e.g. github.com, git.ine.pt).



# Informações finais



## » Estrutura de pastas



```
C:\WINDOWS\system32\cmd. X + v - □ X  
C:.  
├── Projects  
│   ├── 2022  
│   │   ├── 2022.01.01_ProjectName  
│   │   └── 2022.04.01_ProjectName  
│   ├── 2023  
│   │   ├── 2023.02.29_ProjectName  
│   │   └── 2023.12.25_ProjectName  
│   └── 2024  
│       ├── 2024.03.05_ProjectName  
│       └── 2024.10.31_DSR2024  
│           ├── bin  
│           ├── data  
│           ├── docs  
│           ├── media  
│           ├── results  
│           │   ├── 1.transform  
│           │   ├── 2.exploratory  
│           │   └── 3.modeling
```

# Informações finais



## » Ficheiro README

```
#autor:      Joao Sollari Lopes
#local:      INE, Lisboa
#criado:     30.10.2023
#modificado: 06.05.2024

+bin
  |0.examples.r           #exemplos para recapitulacao
  |0.install_packages.r  #instalar pacotes necessarios
  |1.transform.r          #transformacao de dados
  |2.exploration.r        #exploração de dados
  |3.modelling.r          #modelacao de dados

+docs
  |DSR-II2024_program.pdf #programa
  |DSR-II2024_slides.pdf  #slides [versao final]
  |DSR-II2024_slides_20230807.pptx #slides [v2023-08-07]
  |DSR-II2024_slides_20240430.pptx #slides [v2024-04-30]

+results
  +1.tranform             #resultados de "1.transform.r"
  +2.exploration          #resultados de "2.exploration.r"
  +3.modelling            #resultados de "3.modelling.r"
README.txt               #Este ficheiro
```



# Informações finais



## » Controlo de versões

| bin/3.modelling.r             |  |             |  |     |  |             |  |
|-------------------------------|--|-------------|--|-----|--|-------------|--|
| @@ -2,7 +2,7 @@               |  |             |  |     |  |             |  |
| 2                             | #local:  | INE, Lisboa |  | 2   | #local:  | INE, Lisboa |  |
| 3                             | #Rversion:                                     | 4.3.1       |  | 3   | #Rversion:                                     | 4.3.1       |  |
| 4                             | #criado:                                       | 17.07.2023  |  | 4   | #criado:                                       | 17.07.2023  |  |
| 5                             | - #modificado:                                 | 18.04.2024  |  | 5   | + #modificado:                                 | 16.09.2024  |  |
| 6                             |  |             |  | 6   |  |             |  |
| 7                             | # 0. INDEX                                     |             |  | 7   | # 0. INDEX                                     |             |  |
| 8                             | {  |             |  | 8   | {  |             |  |
| @@ -344,7 +344,7 @@ lr_res  > |  |             |  |     |  |             |  |
| 344                           | scale_x_log10(labels = scales::label_number()) |             |  | 344 | scale_x_log10(labels = scales::label_number()) |             |  |
| 345                           |  |             |  | 345 |  |             |  |
| 346                           | lr_res  >                                      |             |  | 346 | lr_res  >                                      |             |  |
| 347                           | - show_best("roc_auc", n = 5)  >               |             |  | 347 | + show_best(metric = "roc_auc", n = 5)  >      |             |  |
| 348                           | arrange(penalty)                               |             |  | 348 | arrange(penalty)                               |             |  |
| 349                           |  |             |  | 349 |  |             |  |
| 350                           | lr_best <-                                     |             |  | 350 | lr_best <-                                     |             |  |

# Informações finais



## » Comunidade R

- <https://www.r-project.org/>
- <https://www.tidyverse.org/>
- <https://www.tidymodels.org/>
- <https://education.rstudio.com/learn/>
- <https://www.r-project.org/help.html>
- <https://hour.ine.pt/>



# Informações finais



## » Bibliografia

» Wickham H & Grolemund G (2017) R for Data Science. O'Reilly Media Inc., Sebastopol.

URL: <https://r4ds.had.co.nz/>

» Wickham H, Çetinkaya-Rundel M & Grolemund G (2023) R for Data Science. O'Reilly Media Inc., Sebastopol. O'Reilly Media Inc., Sebastopol.

URL: <https://r4ds.hadley.nz/>