Heart Attack

Natália Freitas

2022-12-23

Contextualização

O infarto agudo do miocárdio, conhecido como ataque cardíaco (no inglês *Heart Attack*), está entre uma das doenças cardiovasculares mais mortais. Geralmente ocorre quando a circulação ou o fluxo sanguíneo para o coração é interrompido, fazendo com que o coração não receba oxigênio. Leva apenas de seis a oito minutos sem oxigênio para que o músculo cardíaco pare de funcionar, levando o indivíduo a morte.

No Brasil, em 2021, estima-se que 230 mil pessoas morreram por doenças cardiovasculares, e destes, 73.035 mil decorrentes de infarto (CNN,2021).

E se pudéssemos predizer quais pessoas estão propensas a sofrer ataques cardíacos?

Neste contexto, este estudo surge com o objetivo de desenvolver um sistema de predição para identificar, com base em características biológicas, os indivíduos com maior propensão a sofrer ataque cardíaco. Além de ajudar no diagnóstico, possibilita melhor compreensão do problema ao mapear as características que mais se associam ao risco de ataque cardíaco.

Código utilizados para a análises pode ser acessado em GitHub nfreitas1990. Assim como demais materiais utilizados, inclusive a base de dados.

Guia da Análise

```
1. Carregamento dos Pacotes
2. Carregamento dos Dados
3. Conhecendo os Dados
    3.1 Presença de Valores faltantes
    3.2 Tipologia
    3.3 Colinearidade
4. Análise Descritiva
    4.1 Criação de Funções
    4.2 Avaliação de cada variável individualmente
5. Modelagem Preditiva
6. Testes do Modelo
```

1. Pacotes

```
library(tidyverse)
library(dplyr)
library(corrplot)
library(ggplot2)
library(skimr)
library(tidymodels)
```

2. Dados

Os dados utilizados para este relatório foram obtidos no Kaggle. A base de dados *Heart Attack* foi carregada com o nome `ha`.

A base de dados (`ha`) é composta por 14 variáveis. Segue abaixo a tabela com as variáveis e o respectivo significado `tab_sig`.

3. Conhecendo os Dados

```
tab_sig |>
knitr::kable()
```

Siglas	Significado
age	**Idade do paciente**
sex	**Sexo do paciente**

Siglas	Significado
cp	**Tipo de dor no peito**: 1|angina tipica; 2|angina atipica; 3|dor não angina; 4|assintomático
trtbps	**Pressão arterial em repouso** (mm/Hg)
chol	**Colesterol** (mg/dl)
fbs	**Glicemia** (jejum > 120 mg/dl): 1|Verdadeiro; 0|Falso
restecg	**Eletrocardiográficos** (repouso): 1|normal; 2|tendo anormalidade da onda ST-T; 3|provável hipertrofia ventricular esquerda
thalachh	**Frequência cardíaca máxima**
exng	**Angina induzida por exercício**: 1|Sim; 0|Não
oldpeak	**Depressão de ST induzida por exercício**
slp	**Inclinação do segmento ST**: 0|sem inclinação; 1|plano; 2|descendo
caa	**Número de grandes vasos**
thall	**Talassemia**: 0|nulo; 1|defeito corrigido; 2|normal; 3|defeito reversível
output	**Diagnóstico de doença cardíaca**: 0|| < 50% estreitamento do diâmetro. Menos chance de doença cardíaca; 1| > 50% de estreitamento do diâmetro. Mais chance de doença cardíaca

3.1 Avaliando a presença de valores faltantes

Não foram encontrados valores `NA` em nenhuma coluna da tabela `ha`.

```
table(map(ha, is.na)) |>
      knitr::kable()
```

age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	out
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FAL

3.2 Tipologia

As variáveis foram avaliadas quanto ao tipo, categórica ou numérica, e foram transformadas para que respeitassem a sua natureza. As variáveis `sex`, `cp`, `fbs`, `restecg`, `exng`, `slp`, `thall` e `output` foram transformadas para categoricas.

Os dados obtidos correspondem aos registros referentes a 303 pacientes.

```
ha <- ha |>
      mutate(
      across(.cols = c(sex, cp, fbs, restecg, exng, slp, thall, output),
             .fns = as.factor))

skimr::skim(ha)
```

Data summary

Name	ha
Number of rows	303
Number of columns	14
_____	
Column type frequency:	
factor	8

numeric	6
_____	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
sex	0	1	FALSE	2	1: 207, 0: 96
cp	0	1	FALSE	4	0: 143, 2: 87, 1: 50, 3: 23
fbs	0	1	FALSE	2	0: 258, 1: 45
restecg	0	1	FALSE	3	1: 152, 0: 147, 2: 4
exng	0	1	FALSE	2	0: 204, 1: 99
slp	0	1	FALSE	3	2: 142, 1: 140, 0: 21
thall	0	1	FALSE	4	2: 166, 3: 117, 1: 18, 0: 2
output	0	1	FALSE	2	1: 165, 0: 138

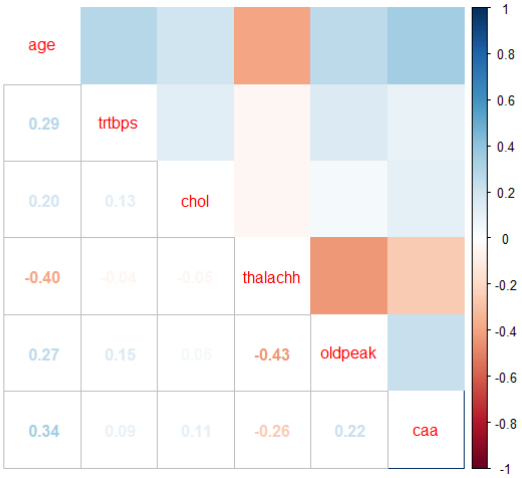
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	54.37	9.08	29	47.5	55.0	61.0	77.0	
trtbps	0	1	131.62	17.54	94	120.0	130.0	140.0	200.0	
chol	0	1	246.26	51.83	126	211.0	240.0	274.5	564.0	
thalachh	0	1	149.65	22.91	71	133.5	153.0	166.0	202.0	
oldpeak	0	1	1.04	1.16	0	0.0	0.8	1.6	6.2	
caa	0	1	0.73	1.02	0	0.0	0.0	1.0	4.0	

3.3 Colinearidade

As variáveis numéricas foram testadas para evitar colinearidade. O método de Spearman foi escolhido em detrimento do método de Pearson para evitar assumir pressuposto com relação a normalidade dos dados.

```
ha_numeric <- ha  |>
                  select(where(is.numeric))
corrplot.mixed(cor(ha_numeric, method = "spearman"),lower = "number", upper = 'color')
```



O teste de Spearman evidenciou baixa correlação entre os pares de variáveis (<0.45). Indicando que podem ser utilizadas simultâneamente nas análises futuras, sem incorrer em risco de colinearidade.

4. Análise Descritiva

4.1 Criando funções que serão usadas nesta seção

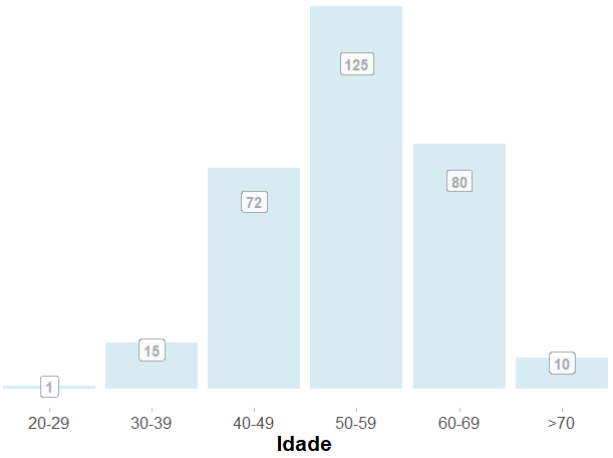
```
# Função grafico_proporcao ( ): Cria Histograma das proporções de pacientes
# com e sem doença cardíaca
  grafico_proporcao <- function(coluna, bins=NULL, breaks = NULL, eixox = NULL){

    ha |>
      ggplot(aes(y = as.numeric(output)-1, x = coluna)) +
      stat_summary_bin(size = 1, alpha = 0.1, colour = "white", bins = bins,breaks = breaks,
                       geom = "bar", fill = "royalblue", fun = function(x) 1,
```

```
                    na.rm = T) +
      stat_summary_bin(size = 1,alpha = 0.3, colour = "white", bins = bins,
                    breaks = breaks,geom = "bar", fill = "orange", na.rm = T) +
      stat_summary_bin(size = 2, alpha = 1, colour = "purple", bins = bins,
                    breaks = breaks, geom = "point", na.rm = T) +
      stat_smooth(method = "glm", method.args = list(family = "binomial"),
                  se = FALSE, na.rm = T) +
      xlab(eixox)+
      ylab("Proporção Doença cardíaca")+
      geom_point() +
      meu_tema}

# Função tabela_proporcao ( ): Cria tabela com as proporções de pacientes
# com e sem doença cardíaca

  tabela_proporcao <- function(coluna, breaks = NULL ){
    ha |>
      mutate(
        coluna_faixa = cut(coluna, breaks = breaks)) |>
      group_by(coluna_faixa) |>
      summarise(
        n = n(),
        coluna = mean(coluna),
        p_output = mean(output == 1),
        logit_chance_output = log(p_output/(1-p_output)))}
```

i. Idade (age)

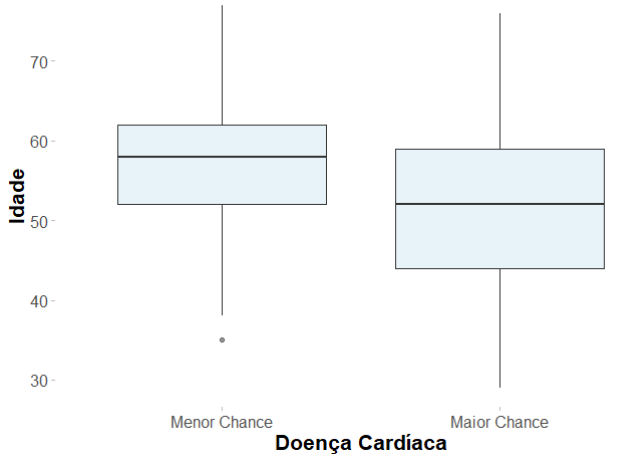
Os dados analisados representam, em sua maioria, pacientes que estão na faixa de 50-59 anos de idade. Não existe paciente com menos de 30 anos de idade, com exceção de 1 paciente na faixa de 20-29 anos.

```
ha |>
    mutate(
      idade = cut(age, breaks = c(20, 29, 39, 49, 59, 69, 79, 89))) |>
    group_by(idade) |>
    summarise(
      n = n()) |>
    ggplot(aes(y = n, x = idade, label = n))+
    geom_bar(stat = "identity", alpha = 1/2, fill= "lightblue") +
    geom_label(position = position_stack (vjust = 0.85),alpha = 0.8,
               colour = "darkgray", fontface = "bold", show_guide  = F)+
    scale_y_continuous(breaks=NULL)+
    ylab(" ")+
    xlab("Idade")+
    scale_x_discrete(labels = c("20-29","30-39","40-49","50-59","60-69",">70"))+
    meu_tema
```



Com base na análise visual do Box Plot entre a idade (`age`) e `output` , variável que representa a chance de doença cardíaca, observamos que os pacientes com **menor chance** doença cardíaca tem média de idade maior do que o grupo com **maior chance** doença cardíaca.

```
ha |>
      ggplot(aes(x = output, y = age))+
      geom_boxplot(fill = "lightblue", alpha = 0.3)+
      scale_x_discrete(labels = c("Menor Chance", "Maior Chance"))+
      xlab ("Doença Cardíaca")+
      ylab("Idade")+
      meu_tema
```



Para confirmar a análise visual, foi avaliado: a Homogeneidade das variâncias e a normalidade dos dados. Estes testes de pressupostos indica que o Teste de Welch é preferível em detrimento do teste T-Student para avaliar a diferença nas médias, devido a **não** homogeneidade das variâncias.

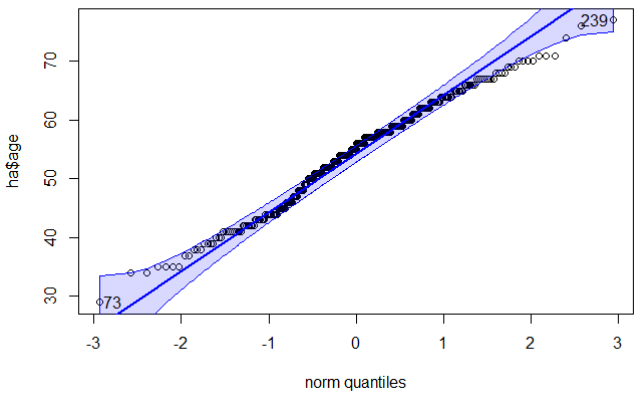
O teste confirmou haver diferença significativa nas médias entre os grupos. O grupo com **menor chance** doença cardíaca apresenta a média de idade superior (~ 56 anos) ao grupo com **maior chance** doença cardíaca (~52 anos).

***Obs: Nas demais variáveis utilizarei apenas avaliação visual.

```
## Testando pressupostos da Análise
# Pressuposto 1. Homogeneidade de variancias
    # Ho: variâncias iguais - Rejeitada
    car::leveneTest(ha$age ~ ha$output)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value   Pr(>F)
## group   1  7.9854 0.005031 **
##       301
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Presuposto 2. Teste de normalidade
  # Inspeção visual - Aceita
    car::qqPlot(ha$age, grid = FALSE)
```




```
## [1]  73 239
```

```
# Teste de Weltch - se diferencia do T-Student pelo argumento "var.equal = F"
t.test(ha$age ~ ha$output, var.equal=F)
```

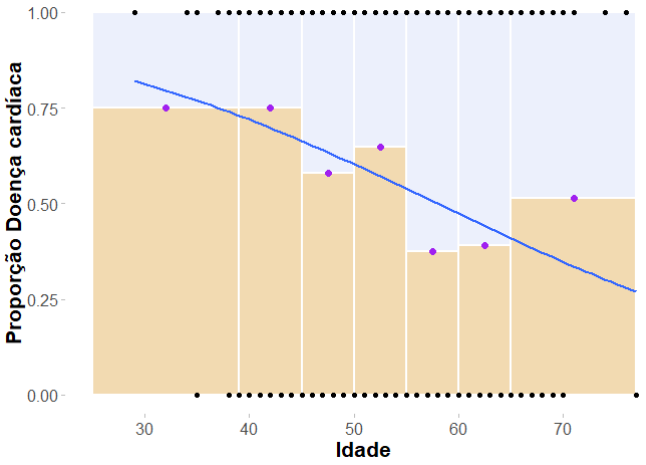
```
##
##  Welch Two Sample t-test
##
## data:  ha$age by ha$output
## t = 4.0797, df = 301, p-value = 5.781e-05
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  2.124635 6.084324
## sample estimates:
```

```
## mean in group 0 mean in group 1
##        56.60145        52.49697
```

Chance de Doença cardíaca, com base apenas na idade (age)

Com base no histograma e na tabela de proporções, podemos observar que 75% dos pacientes que estão na faixa de idade entre 25-45 anos possui doença cardíaca. A partir dos 45 anos a proporção de pacientes com doença cardíaca diminui atingido o proporção mínima de 37% e 39% (na faixa de 55 - 60 anos e 60 - 65 anos, respectivamente)

```
# Funções Criadas no início da seção
    grafico_proporcao(coluna = ha$age, breaks = c(25,39,45,50,55,60,65,77), eixox = "Idade")
```




```
    tabela_proporcao(coluna = ha$age, breaks = c(25,39,45,50,55,60,65,77)) |>
      knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(25,39]	16	54.36634	0.7500000	1.0986123
(39,45]	48	54.36634	0.7500000	1.0986123
(45,50]	31	54.36634	0.5806452	0.3254224
(50,55]	57	54.36634	0.6491228	0.6151856
(55,60]	72	54.36634	0.3750000	-0.5108256
(60,65]	46	54.36634	0.3913043	-0.4418328
(65,77]	33	54.36634	0.5151515	0.0606246

Com base no modelo de regressão logistico individual, podemos inferir que a `age` está associada com a chance de ter doença cardíaca (p<0.05). E que **o aumento em uma unidade de `age` implica na redução em 5% da chance de ter doença cardíaca.** (OR: 1-0.95 = 0.05).

```
model_age <- glm(output ~age, data=ha, family = "binomial")
gtsummary::tbl_regression(model_age, exponentiate = TRUE)
```

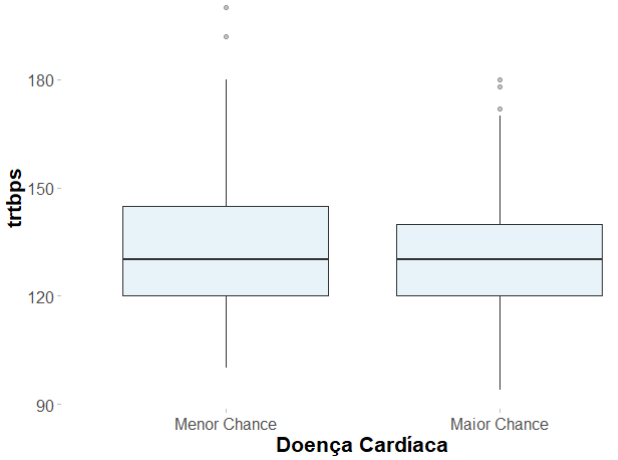
Characteristic	OR[1]	95% CI[1]	p-value
age	0.95	0.92, 0.97	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

ii. Pressão Arterial (`trtbps`)

A análise visual não indica diferença entre grupos com **menor** e **maior** chance de doença cardíaca. Como consequência, inicialmente esta variável não parece interferir nas chances do paciente ter ou não doença cardíaca.

```
ha |>
   ggplot(aes(x = output, y = trtbps))+
   geom_boxplot(fill = "lightblue", alpha = 0.3)+
   scale_x_discrete(labels = c("Menor Chance", "Maior Chance"))+
   xlab ("Doença Cardíaca")+
   meu_tema
```

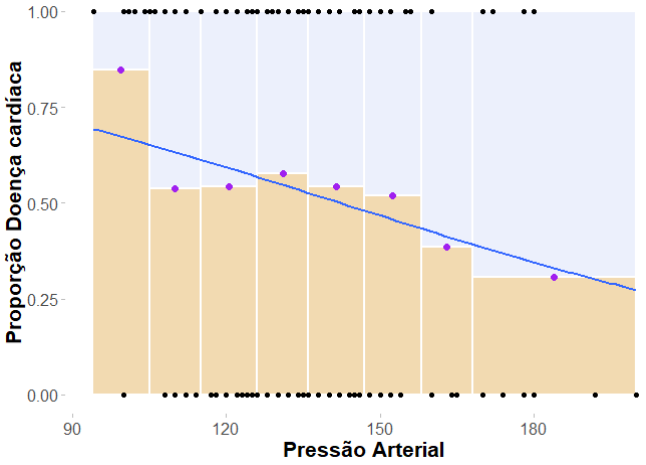


A pressão arterial na faixa entre 93.9 - 105 possui maior proporção de pacientes com doença cardíaca (85%). Cabe ressaltar que esta faixa apresenta, comparativamente com as demais, um número baixo de observações, o que prejudica um pouco a interpretação. Após esta faixa (93.9 - 105) a proporção de pacientes com doença cardíaca se mantém constante (em torno de 50%), diminuindo somente acima de 158 de pressão arterial.

```
grafico_proporcao(coluna = ha$trtbps, breaks = c(93.9,105,
                                                 115,126,136,147,
                                                 158,168,200),
                  eixox = "Pressão Arterial")
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
tabela_proporcao(coluna = ha$trtbps, breaks = c(93.9,105,
                                                115,126,136,147,
                                                158,168,200,330)) |>
    knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(93.9,105]	13	131.6238	0.8461538	1.7047481
(105,115]	39	131.6238	0.5384615	0.1541507
(115,126]	70	131.6238	0.5428571	0.1718503
(126,136]	71	131.6238	0.5774648	0.3123747
(136,147]	57	131.6238	0.5438596	0.1758907
(147,158]	27	131.6238	0.5185185	0.0741080
(158,168]	13	131.6238	0.3846154	-0.4700036
(168,200]	13	131.6238	0.3076923	-0.8109302

Após regredir a variável `output` em função `trtbps`, notamos que a pressão arterial em repouso é uma variável importante (p<0.05).Embora **o aumento da pressão reduza a chance de doença cardíaca em apenas 2% a cada unidade**.

```
model_trtbps <- glm(output ~trtbps, data=ha, family = "binomial")
  summary(model_trtbps)
```

```
##
## Call:
## glm(formula = output ~ trtbps, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4928  -1.2451   0.9426   1.0902   1.4575
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.409327   0.904136   2.665   0.0077 **
## trtbps      -0.016929   0.006802  -2.489   0.0128 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 411.22  on 301  degrees of freedom
## AIC: 415.22
##
## Number of Fisher Scoring iterations: 4
```

```
  gtsummary::tbl_regression(model_trtbps, exponentiate = TRUE)
```

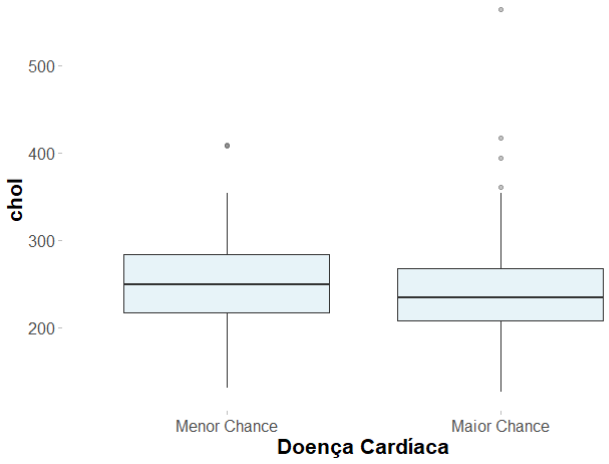
Characteristic	OR[1]	95% CI[1]	p-value
trtbps	0.98	0.97, 1.00	0.013

[1] OR = Odds Ratio, CI = Confidence Interval

iii. Colesterol (`chol`)

O colesterol, embora sabidamente importante de acordo com a literatura, não apresentou diferença visual entre os grupos com **menor** e **maior** chance de doença cardíaca. De acordo com o modelo de regressão, essa variável não altera a chance de doenças cardíacas (p>0.05; OR= 1).

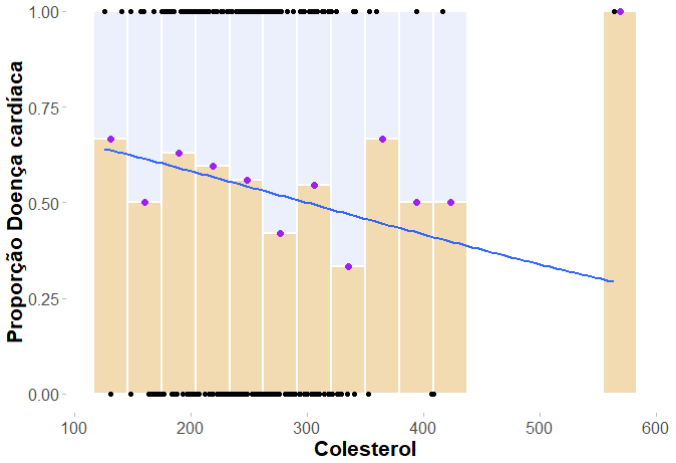
```
ha |>
    ggplot(aes(x = output, y = chol))+
    geom_boxplot(fill = "lightblue", alpha = 0.3)+
    scale_x_discrete(labels = c("Menor Chance", "Maior Chance"))+
    xlab ("Doença Cardíaca")+
    meu_tema
```




```
grafico_proporcao(coluna = ha$chol, bins = 15, eixox = "Colesterol")
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
tabela_proporcao(coluna = ha$chol, breaks = 15) |>
  knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(126,155]	5	246.264	0.6000000	0.4054651
(155,184]	22	246.264	0.5454545	0.1823216
(184,214]	58	246.264	0.6379310	0.5663955
(214,243]	69	246.264	0.6086957	0.4418328
(243,272]	68	246.264	0.5441176	0.1769307
(272,301]	38	246.264	0.2894737	-0.8979416
(301,330]	30	246.264	0.5333333	0.1335314
(330,360]	7	246.264	0.4285714	-0.2876821
(360,389]	1	246.264	1.0000000	Inf
(389,418]	4	246.264	0.5000000	0.0000000
(535,564]	1	246.264	1.0000000	Inf

```
model_chol <- glm(output ~ chol, data=ha, family = "binomial")
summary(model_chol)
```

```
##
## Call:
## glm(formula = output ~ chol, family = "binomial", data = ha)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.425  -1.241   1.015   1.093   1.567
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.001617   0.571467   1.753   0.0797 .
## chol        -0.003338   0.002269  -1.471   0.1412
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 415.43  on 301  degrees of freedom
## AIC: 419.43
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_chol, exponentiate = TRUE)
```

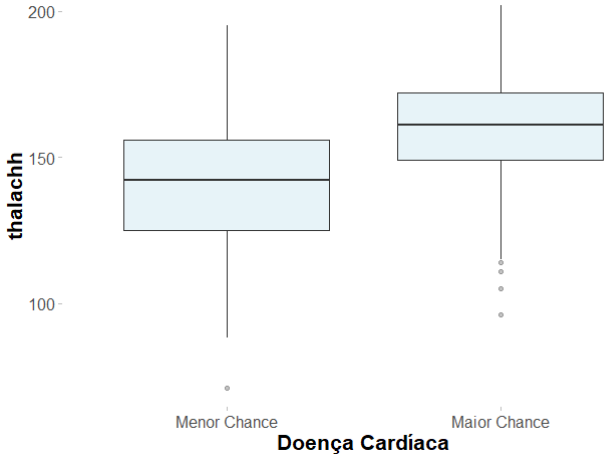
Characteristic	OR[1]	95% CI[1]	p-value
chol	1.00	0.99, 1.00	0.14

[1] OR = Odds Ratio, CI = Confidence Interval

iv. Frequencia cardíaca Máxima (`thalachh`)

A análise visual sugere que valores de frequência cardíaca máxima maior do que 150 tem **maior** chance de doença cardíaca. Enquanto valores entre 125-150 tem **menor** chance de doença cardíaca.

```
ha |>
   ggplot(aes(x = output, y = thalachh))+
   geom_boxplot(fill = "lightblue", alpha = 0.3)+
   scale_x_discrete(labels = c("Menor Chance", "Maior Chance"))+
   xlab ("Doença Cardíaca")+
 meu_tema
```

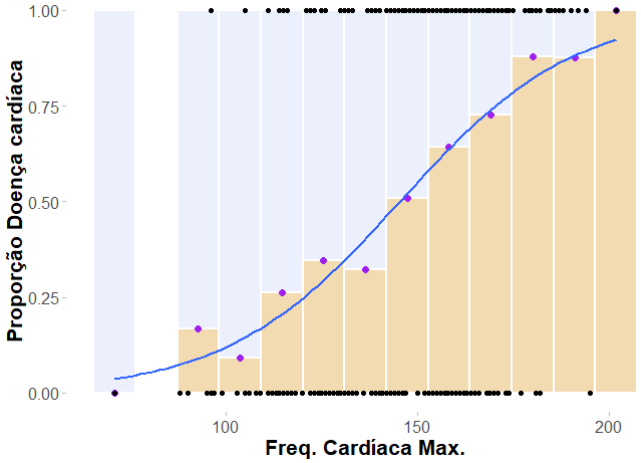


A proporção de pacientes com doenças cardíacas aumenta conforme o aumento nos valores de frequência cardíaca máxima, atingindo proporções acima de 60% a partir da faixa 147-158.

```
grafico_proporcao(coluna = ha$thalachh, bins = 12, eixox = "Freq. Cardíaca Max.")
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
tabela_proporcao(coluna = ha$thalachh, breaks = 12) |>
  knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(70.9,81.9]	1	149.6469	0.0000000	-Inf
(81.9,92.8]	2	149.6469	0.0000000	-Inf
(92.8,104]	7	149.6469	0.1428571	-1.7917595
(104,115]	17	149.6469	0.1764706	-1.5404450
(115,126]	25	149.6469	0.4000000	-0.4054651
(126,136]	27	149.6469	0.2222222	-1.2527630
(136,147]	48	149.6469	0.3750000	-0.5108256
(147,158]	53	149.6469	0.6415094	0.5819215
(158,169]	60	149.6469	0.6666667	0.6931472

coluna_faixa	n	coluna	p_output	logit_chance_output
(169,180]	45	149.6469	0.8444444	1.6916760
(180,191]	14	149.6469	0.8571429	1.7917595
(191,202]	4	149.6469	0.7500000	1.0986123

De acordo com o modelo logístico individual, a frequência cardíaca máxima (`thalachh`) influencia na chance de doenças cardiacas (p< 0.05). **O aumento na frequência cardiaca máxima, aumenta a chance do paciente ter doença cardíaca em 4% a cada unidade**.

```
model_thalachh <- glm(output ~ thalachh, data=ha, family = "binomial")
  summary(model_thalachh)
```

```
##
## Call:
## glm(formula = output ~ thalachh, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1383  -1.0780   0.6043   0.9200   2.1354
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.391452   0.987133  -6.475 9.50e-11 ***
## thalachh     0.043951   0.006531   6.729 1.71e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 359.26  on 301  degrees of freedom
## AIC: 363.26
##
## Number of Fisher Scoring iterations: 4
```

```
  gtsummary::tbl_regression(model_thalachh, exponentiate = TRUE)
```

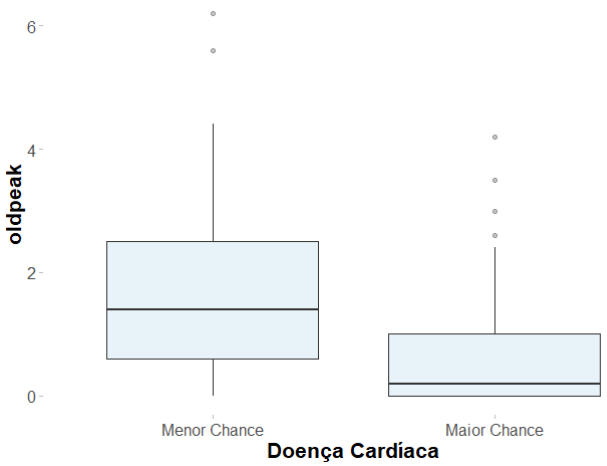
Characteristic	OR[1]	95% CI[1]	p-value
thalachh	1.04	1.03, 1.06	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

v. Depressão de ST induzida por exercício (`oldpeak`)

A inspeção visual indica diferença entre os grupos com **menor** e **maior** chance de doença cardíaca. O grupo com maior chance de doença possuem valores de `oldpeack` entre 1-0. Enquanto o grupo com menor chance apresenta a média dos valores acima de 1.

```
ha |>
    ggplot(aes(x = output, y = oldpeak))+
    geom_boxplot(fill = "lightblue", alpha = 0.3)+
    scale_x_discrete(labels = c("Menor Chance", "Maior Chance"))+
    xlab ("Doença Cardíaca")+
    meu_tema
```



Com valores menores de depressão de st induzida por exercício, observamos alta proporção de pacientes com doença cardíaca (74%). A partir de valores de `oldpeak` maiores do que 2.07, a proporção de pacientes doentes reduz para menos de 30% em cada faixa.

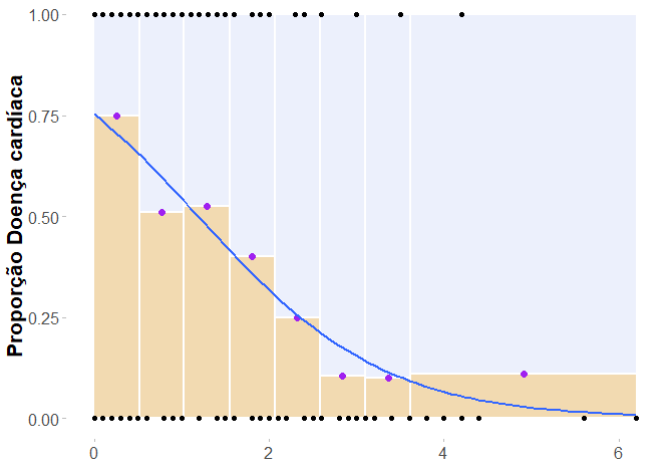
```
tabela_proporcao(coluna = ha$oldpeak, breaks = c(-0.0061, 0.517,1.03,1.55,
                                                 2.07,2.58,3.1,3.62,6.21)) |>
  knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(-0.0061,0.517]	135	1.039604	0.7481481	1.0887600
(0.517,1.03]	45	1.039604	0.5111111	0.0444518
(1.03,1.55]	38	1.039604	0.5263158	0.1053605
(1.55,2.07]	35	1.039604	0.4000000	-0.4054651
(2.07,2.58]	12	1.039604	0.2500000	-1.0986123
(2.58,3.1]	19	1.039604	0.1052632	-2.1400662
(3.1,3.62]	10	1.039604	0.1000000	-2.1972246
(3.62,6.21]	9	1.039604	0.1111111	-2.0794415

```
grafico_proporcao(coluna = ha$oldpeak, breaks = c(-0.0061, 0.517,1.03,1.55,
                                                  2.07,2.58,3.1,3.62,6.21))
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



A Depressão de ST tem forte influência na chance de doença cardíaca. **Ao aumentarmos os valores de `oldpeak` em 1 unidade, há uma redução de 61% na chance de ocorrência de doença cardiaca** (OR:0.39).

```
# Modelo individual
model_oldpeak <- glm(output ~ oldpeak, data=ha, family = "binomial")
summary(model_oldpeak)
```

```
##
## Call:
## glm(formula = output ~ oldpeak, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6737  -1.0186   0.7522   0.8656   2.4025
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.1177     0.1810   6.177 6.55e-10 ***
## oldpeak      -0.9396     0.1386  -6.779 1.21e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 355.00  on 301  degrees of freedom
## AIC: 359
```

```
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_oldpeak, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
oldpeak	0.39	0.29, 0.51	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

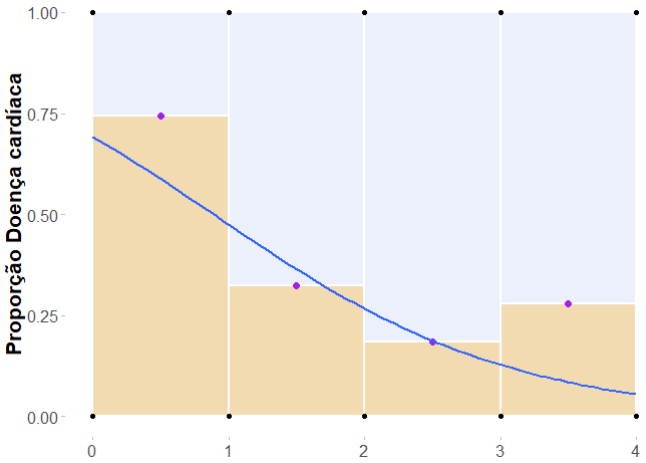
vi. Número de grandes vasos (caa)

O aumento no número de grande vasos diminui a chance de doenças cardíacas. Com apenas 1 vaso temos proporção maior de pacientes com doença (0.62%) quando comparadas aos demais grupos que possuem mais de 1 vaso. Importante ressaltar que apesar da classe "4 vasos" ter o maior percentual de pacientes com doença (80%; Tabela de proporções), esta classe foi desconsiderada para essa interpretação devido a baixa quantidade de registros.

```
grafico_proporcao(coluna = ha$caa, bins = 4)
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
tabela_proporcao(coluna = ha$caa, breaks = 4) |>
  knitr::kable()
```

coluna_faixa	n	coluna	p_output	logit_chance_output
(-0.004,1]	240	0.7293729	0.6291667	0.5286435
(1,2]	38	0.7293729	0.1842105	-1.4880771
(2,3]	20	0.7293729	0.1500000	-1.7346011
(3,4]	5	0.7293729	0.8000000	1.3862944

A partir do modelo de regressão logístico, podemos concluir que o número de grandes vasos influencia fortemente na chance de doença cardíaca. **O aumento no número de vasos reduz a chance de doença cardíaca em 60% a cada unidade**.

```
model_caa <- glm(output ~ caa, data=ha, family = "binomial")
  summary(model_caa)
```

```
##
## Call:
## glm(formula = output ~ caa, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5349  -1.1355   0.8579   0.8579   2.4020
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.8099     0.1544   5.247 1.54e-07 ***
## caa          -0.9093     0.1466  -6.201 5.60e-10 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 367.63  on 301  degrees of freedom
## AIC: 371.63
##
## Number of Fisher Scoring iterations: 4
```

```
  gtsummary::tbl_regression(model_caa, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
caa	0.40	0.30, 0.53	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

vii. Sexo (sex)

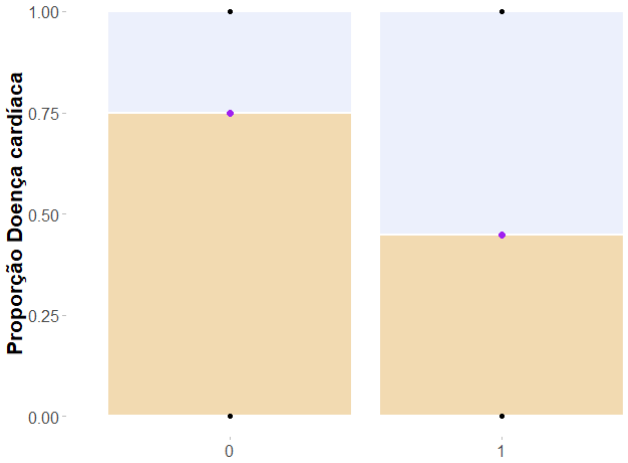
Observamos maior proporção de mulheres (0) com doença cardíaca do que homens (1) conforme pode ser observado a seguir.

```
graf_proporcao_categoricos <- ha |>
    select(is.factor) |>
    map(grafico_proporcao)
```

```
graf_proporcao_categoricos$sex
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



A chance de doença cardíaca no sexo masculino é 73% menor do que no sexo feminino

```
model_sex <- glm(output ~ sex, data=ha, family = "binomial")
  summary(model_sex)
```

```
##
## Call:
## glm(formula = output ~ sex, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6651  -1.0923   0.7585   1.2650   1.2650
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0986     0.2357   4.661 3.15e-06 ***
## sex1         -1.3022     0.2740  -4.752 2.01e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 392.80  on 301  degrees of freedom
```

```
## AIC: 396.8
##
## Number of Fisher Scoring iterations: 4
```

```
  gtsummary::tbl_regression(model_sex, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
sex			
0	—	—	
1	0.27	0.16, 0.46	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

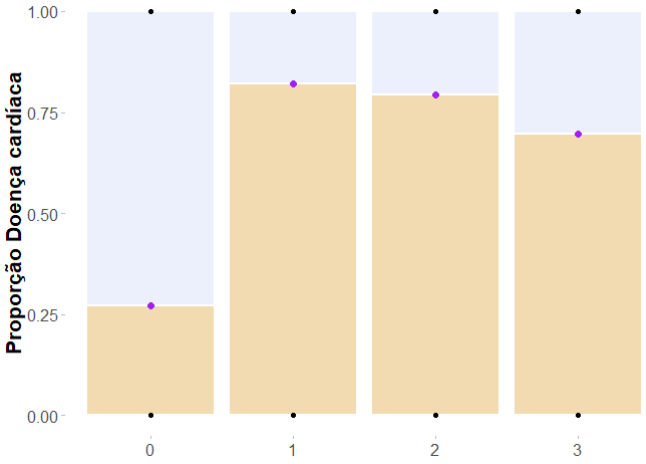
viii Tipo de dor no peito (cp)

Observamos menor proporção de doença cardíaca em pacientes com tipo de dor no peito *angina típica* (0), quando comparado com os demais.

```
graf_proporcao_categoricos$cp
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Ter Dor no peito do tipo *angina atipica* (1) aumenta em 12 vezes a chance de doença cardíaca quando comparada *angina tipica* (0); aumenta em 10 vezes quando a dor é do tipo *não angina*; e em 6 vezes quando *assintomático*.

```
model_cp <- glm(output ~ cp, data=ha, family = "binomial")
  summary(model_cp)
```

```
##
## Call:
## glm(formula = output ~ cp, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8519  -0.7981   0.6300   0.6809   1.6120
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.9808     0.1878  -5.224 1.75e-07 ***
## cp1           2.4972     0.4132   6.043 1.51e-09 ***
## cp2           2.3246     0.3245   7.163 7.87e-13 ***
## cp3           1.8075     0.4905   3.685 0.000229 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
```

```
## Residual deviance: 331.70  on 299  degrees of freedom
## AIC: 339.7
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_cp, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
cp			
0	—	—	
1	12.1	5.62, 28.8	<0.001
2	10.2	5.51, 19.8	<0.001
3	6.10	2.41, 16.9	<0.001

[1] OR = Odds Ratio, CI = Confidence Interval

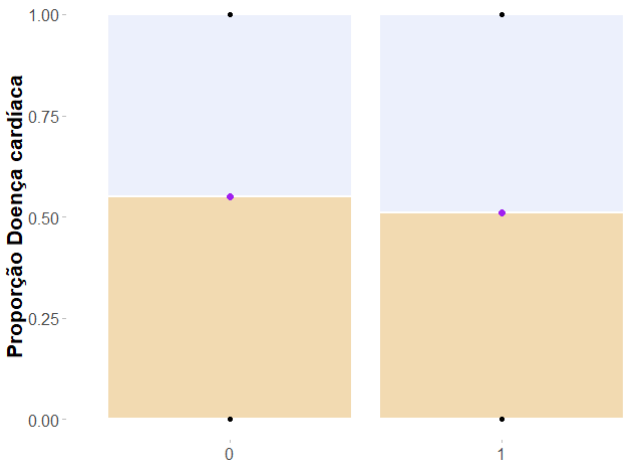
ix. Glicemia (fbs)

Proporção de pacientes com **menor** e **maior** chance de doença cardíaca não parece ser modificada pela glicemia.

```
graf_proporcao_categoricos$fbs
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



De acordo com o modelo de regressão logístico, **a glicemia não parece influenciar a chance de doenças cardíacas** (p>0.05).

```
model_fbs <- glm(output ~ fbs, data=ha, family = "binomial")
  summary(model_fbs)
```

```
##
## Call:
## glm(formula = output ~ fbs, family = "binomial", data = ha)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.264  -1.264   1.093   1.093   1.159
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.2022     0.1252   1.616    0.106
## fbs1         -0.1578     0.3234  -0.488    0.626
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 417.40  on 301  degrees of freedom
## AIC: 421.4
##
## Number of Fisher Scoring iterations: 3
```

```
gtsummary::tbl_regression(model_fbs, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
fbs			
0	—	—	
1	0.85	0.45, 1.62	0.6

[1] OR = Odds Ratio, CI = Confidence Interval

x. Eletrocardiográficos (`restecg`)

Pacientes com Eletrocardiograma *normal* (0) possuem menor proporção de doença cardíaca do que pacientes com *anormalidade da onda ST-T* (1). A categoria de *hipertrofia ventricular* (2) não será considerada na interpretação por apresentar poucas observações.

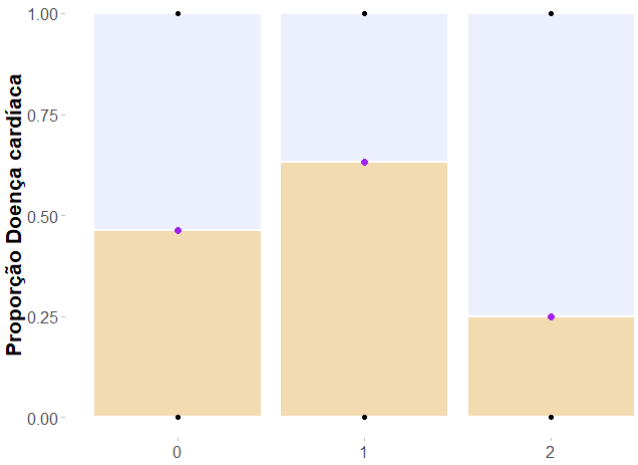
```
table(ha$restecg) |>
  knitr::kable(col.names = c("Categorias", "Quantidade de Observações"), align = "l")
```

Categorias	Quantidade de Observações
0	147
1	152
2	4

```
graf_proporcao_categoricos$restecg
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Os Eletrocardiográficos com *anormalidade da onda ST-T* (1) aumenta em 99% a chance de doença quando comparado com eletrocardiografico *normal* (0). A categoria de *hipertrofia ventricular* (2) não será considerada na interpretação por apresentar poucas observações.

```
model_restecg <- glm(output ~ restecg, data=ha, family = "binomial")
  summary(model_restecg)
```

```
##
## Call:
## glm(formula = output ~ restecg, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4132  -1.1144   0.9587   0.9587   1.6651
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.1499     0.1654  -0.906  0.36472
## restecg1      0.6889     0.2359   2.921  0.00349 **
## restecg2     -0.9487     1.1665  -0.813  0.41606
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 407.53  on 300  degrees of freedom
## AIC: 413.53
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_restecg, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
restecg			
0	—	—	
1	1.99	1.26, 3.17	0.003
2	0.39	0.02, 3.10	0.4

[1] OR = Odds Ratio, CI = Confidence Interval

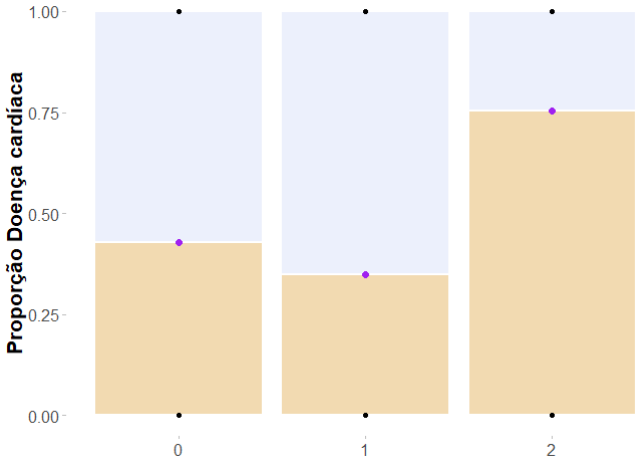
xi. Inclinação do segmento ST (`slp`)

A Inclinação *descendo* (2) apresenta maior proporção de pacientes com doença cardíaca do que as demais.

```
graf_proporcao_categoricos$slp
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



A inclinação do segmento *descendo* (2) **aumenta 4 vezes** a chance de doença cardíaca quando comparado com pacientes *sem inclinação* (0). A inclinação do segmento na categoria *plano* (1), **reduz a chance de doença em 28%** quando comparada aos pacientes *sem inclinação* (0)

```
model_slp <- glm(output ~ slp, data=ha, family = "binomial")
  summary(model_slp)
```

```
##
## Call:
## glm(formula = output ~ slp, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6736  -0.9282   0.7523   0.7523   1.4490
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.2877     0.4410  -0.652  0.51414
## slp1         -0.3314     0.4752  -0.697  0.48564
## slp2          1.4052     0.4820   2.915  0.00356 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 417.64  on 302  degrees of freedom
## Residual deviance: 368.56  on 300  degrees of freedom
## AIC: 374.56
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_slp, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
slp			
0	—	—	
1	0.72	0.28, 1.87	0.5
2	4.08	1.60, 10.8	0.004

[1] OR = Odds Ratio, CI = Confidence Interval

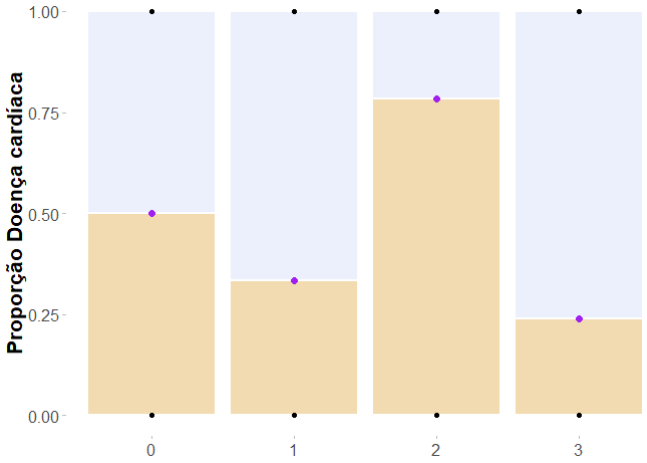
xii. Talassemia (`thall`)

A categoria *Talassemia Normal* (2) é a que apresenta maior proporção de pacientes com doença cardíaca.

```
graf_proporcao_categoricos$thall
```

```
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
## No summary function supplied, defaulting to `mean_se()`
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Antes da análise de regressão, vamos retirar a categoria *nulo* (0), que possui poucas observações e está sendo usada como categoria de referência na análise.

```
# Retirando classe 0
 ha <-  ha |>
            filter(thall != 0)
```

A Talassemia *normal* (2), **aumenta em 7 vezes** a chance de doença cardíaca quando comparada com a categoria *defeito corrigido* (1). Já a categoria *defeito reversível* (3) **reduz em 37%** a chance de doença cardíaca quando comparada a categoria de referência *defeito corrigido* (1)

```
model_thall <- glm(output ~ thall, data=ha, family = "binomial")
  summary(model_thall)
```

```
##
## Call:
## glm(formula = output ~ thall, family = "binomial", data = ha)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7484  -0.7397   0.6992   0.6992   1.6911
```

```
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.6931     0.5000  -1.386 0.165657
## thall2        1.9772     0.5343   3.701 0.000215 ***
## thall3       -0.4633     0.5449  -0.850 0.395230
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 414.85  on 300  degrees of freedom
## Residual deviance: 325.29  on 298  degrees of freedom
## AIC: 331.29
##
## Number of Fisher Scoring iterations: 4
```

```
gtsummary::tbl_regression(model_thall, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
thall			
1	—	—	
1	—	—	
2	7.22	2.62, 22.0	<0.001
3	0.63	0.22, 1.95	0.4

[1] OR = Odds Ratio, CI = Confidence Interval

Conclusões sobre a **Avaliação Individual** da influência das variáveis na chance de doenças cardíacas

Importante lembrar que ao serem avaliadas dentro de um mesmo modelo a resposta aqui apresentada possívelmente sofrerar modificações. Esta análise individual foi realizada como uma tentativa de selecionar variáveis com maior importância para compor o modelo múltiplo apresentado posteriormente.

- Idade (`age`) **reduz em 5%** a chance de doença cardíaca a cada unidade;
- Pressão Arterial (`trtbps`) **reduz em 2%** a chance de doença cardíacaa cada unidade;
- Frequência Cardíaca Máxima (`thalachh`) **aumenta em 4%** a chance de doença cardíaca a cada unidade;
- Depressão de ST induzida por exercício (`oldpeak`) **reduz em 61%** a chance de doença cardíaca a cada unidade;
- Número de grandes vasos (`caa`) **reduz em 60%** a chance de doença cardíaca a cada unidade;
- Sexo (`sex`): A chance de doença cardíaca no sexo masculino é 73% menor do que no sexo feminino;
- Tipo de dor no peito (`cp`): *Angina atipica* **aumenta em 12 vezes** a chance de doença cardíaca quando comparada *Angina tipica*; **aumenta em 10 vezes** quando a dor é do tipo *não angina*; e **aumenta em 6 vezes** quando *assintomático*; <p align="justify"
- Eletrocardiográficos (`restecg`): Eletrocardiográficos com *anormalidade da onda ST-T* **aumenta em 99%** a chance de doença quando comparado com eletrocardiografico *normal*;

  <p align="justify"
- Inclinação do segmento ST (`slp`): A inclinação do segmento *descendo* **aumenta 4 vezes** a chance de doença cardíaca quando comparado com pacientes *sem inclinação*. A inclinação do segmento na categoria *plano*, **reduz em 28%** a chance de doença quando comparada aos pacientes *sem inclinação*;

  <p align="justify"
- Talassemia (`thall`) *normal* **aumenta em 7 vezes** a chance de doença cardíaca quando comparada com a categoria *defeito corrigido*. Já a categoria defeito *reversível* **reduz em 37%** a chance de doença cardíaca quando comparada a categoria de referência *defeito corrigido*;
- Colesterol (`chol`) e Glicemia (`fbs`) não influenciou na chance de doenças cardíacas (p>0.05).

5. Modelagem preditiva

_. Funções Criadas para serem usadas nesta seção

```
# Função analise_predicao ( ): para predizer valores de acordo com modelo da regressao
  analise_predicao <- function (modelo){
    fitted_results <- dismo::predict(modelo, newdata = data_test, type = "response")
    fitted_results_cat <- ifelse(fitted_results > 0.7,1,0) #threshold

    data_test_pred <- data_test |>
```

```
      add_column(fitted =  fitted_results,
                 fitted_cat = fitted_results_cat ) |>
      mutate(fitted_exp = exp(fitted_results))
    print(data_test_pred)}


#Função desempenho ( ): para calcular as metricas de desempenho dos modelos

  desempenho <-  function(modelo){
    fitted_results <- dismo::predict(modelo, newdata = data_test, type = "response")
    fitted_results_cat <- ifelse(fitted_results > 0.7,1,0) #threshold

    data_test_pred <- data_test |>
      add_column(fitted =  fitted_results,
                 fitted_cat = fitted_results_cat ) |>
      mutate(fitted_exp = exp(fitted_results))

    data_test_pred <- data_test_pred |>
      mutate(fitted_cat = as_factor(fitted_cat))

    metricas <- caret::confusionMatrix(data = data_test_pred1$output,
                                       reference = data_test_pred1$fitted_cat)
    print(metricas)}
```

i. Separação dos dados para Calibração e Validação do Modelo

Os dados referentes a 301 pacientes foram separados aleatoriamente, 80% dos dados foram utilizados para calibrar e 20% para validar o modelo na próxima etapa.

```
set.seed(1)
  grupo = dismo::kfold(ha, 5)
  data_train <- ha[grupo !=1,] #80% (4/5 dos registros)
  data_test <- ha[grupo ==1,] #20% (1/5 dos registros)
```

ii. Modelo Logístico Completo

O modelo contendo todas as variáveis foi utilizado como ponto de partida. Neste modelo, algumas variáveis foram significativas (p<0.05):

`sex cp oldpeak caa exng`

```
modelo_completo <- glm(output ~ ., data = data_train, family = binomial)
 summary(modelo_completo)
```

```
##
## Call:
## glm(formula = output ~ ., family = binomial, data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8618  -0.3596   0.1539   0.4861   2.8390
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.650555   3.078394   0.536  0.59184
## age         -0.001628   0.027099  -0.060  0.95209
## sex1        -1.334676   0.587511  -2.272  0.02310 *
## cp1          0.793844   0.686481   1.156  0.24752
## cp2          1.867791   0.567922   3.289  0.00101 **
## cp3          1.819685   0.720542   2.525  0.01156 *
## trtbps      -0.008233   0.013002  -0.633  0.52658
## chol        -0.001925   0.004328  -0.445  0.65654
## fbs1         0.203766   0.653365   0.312  0.75514
## restecg1     0.680371   0.449971   1.512  0.13053
## restecg2     0.043869   2.300808   0.019  0.98479
## thalachh     0.016302   0.011753   1.387  0.16545
## exng1       -0.882818   0.520361  -1.697  0.08978 .
## oldpeak     -0.559507   0.270505  -2.068  0.03860 *
## slp1        -0.803210   0.925137  -0.868  0.38528
## slp2         0.496624   1.005718   0.494  0.62145
## caa         -0.967001   0.235506  -4.106 4.02e-05 ***
## thall2      -0.327074   0.852243  -0.384  0.70114
## thall3      -1.707645   0.822314  -2.077  0.03784 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 331.07  on 240  degrees of freedom
## Residual deviance: 155.05  on 222  degrees of freedom
## AIC: 193.05
##
## Number of Fisher Scoring iterations: 6
```

```
 gtsummary::tbl_regression(modelo_completo, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
age	1.00	0.95, 1.05	>0.9
sex			
0	—	—	
1	0.26	0.08, 0.81	0.023
cp			
0	—	—	
1	2.21	0.60, 9.18	0.2
2	6.47	2.21, 20.9	0.001
3	6.17	1.57, 27.1	0.012
trtbps	0.99	0.97, 1.02	0.5
chol	1.00	0.99, 1.01	0.7
fbs			
0	—	—	
1	1.23	0.35, 4.54	0.8
restecg			
0	—	—	
1	1.97	0.82, 4.87	0.13
2	1.04	0.01, 52.9	>0.9
thalachh	1.02	0.99, 1.04	0.2
exng			
0	—	—	
1	0.41	0.15, 1.15	0.090
oldpeak	0.57	0.33, 0.95	0.039
slp			
0	—	—	
1	0.45	0.07, 2.58	0.4
2	1.64	0.21, 11.0	0.6
caa	0.38	0.23, 0.59	<0.001
thall			
1	—	—	
1	—	—	
2	0.72	0.13, 3.84	0.7
3	0.18	0.03, 0.91	0.038

[1] OR = Odds Ratio, CI = Confidence Interval

iii. Verificar o VIF do modelo

O fator VIF indica baixa colinearidade entre as variáveis.

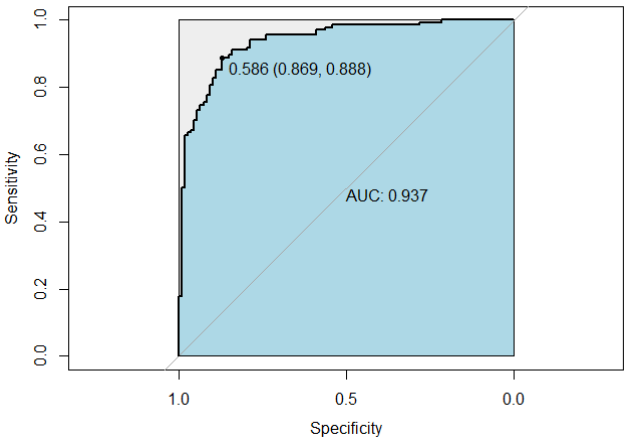
```
car::vif(modelo_completo) |>
knitr::kable(align = "l")
```

	GVIF	Df	GVIF^(1/(2*Df))
age	1.412540	1	1.188503
sex	1.540644	1	1.241227
cp	1.896229	3	1.112539
trtbps	1.285002	1	1.133579
chol	1.307061	1	1.143268
fbs	1.139655	1	1.067546
restecg	1.302335	2	1.068269
thalachh	1.451954	1	1.204970

	GVIF	Df	GVIF^(1/(2*Df))
exng	1.253402	1	1.119555
oldpeak	1.563801	1	1.250520
slp	1.791515	2	1.156925
caa	1.354388	1	1.163782
thall	1.646676	2	1.132797

iv. Testar o Modelo com a Curva ROC

```
plot(roc_completo,
     print.auc = TRUE,
     auc.polygon = TRUE,
     grud = c(0.1,0.2),
     grid.col = c("green", "red"),
     max.auc.polygon = TRUE,
     auc.polygon.col = "lightblue",
     print.thres = TRUE)
```



Conclusão:

- A chance de doença cardíaca no sexo masculino é 74% menor do que no sexo feminino;
- Dores no peito do tipo *não angina* e *assintomático* aumenta em 6 vezes a chance de doença cardíaca quando comparado ao tipo *angina típica*.
- Depressão de ST induzida por exercício reduz em 43% a chance de doença cardíaca a cada unidade de incremento;
- O aumento no Número de grandes vasos reduz a chance de doença cardíaca em 62% a cada unidade de incremento.
- A Talassemia com *defeito reversível* reduz em 82% a chance de doença cardíaca quando comparada a Talassemia *nula*;

Validação do Modelo Completo

Validação da capacidade preditiva do modelo em dados externos, ou seja, dados que não foram utilizados para a calibração do modelo. Construção da matriz de confusão para o cálculo das métricas de desemepnho.

```
# Predicao com dados teste
      fitted_results <- dismo::predict(modelo_completo, newdata = data_test, type = "response")
      fitted_results_cat <- ifelse(fitted_results > 0.7,1,0) #threshold


# Unir: Observados vs Preditos
      data_test_pred <- data_test |>
                        add_column(fitted =  fitted_results,
                                   fitted_cat = fitted_results_cat ) |>
                        mutate(fitted_exp = exp(fitted_results))

      data_test_pred <- data_test_pred |>
        mutate(fitted_cat = as_factor(fitted_cat))

# Metricas Desempenho: Tabela de confusão
      caret::confusionMatrix(data = data_test_pred$output, reference = data_test_pred$fitted_cat)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 27  3
##          1  7 23
##
```

```
##               Accuracy : 0.8333
##                 95% CI : (0.7148, 0.9171)
##    No Information Rate : 0.5667
##    P-Value [Acc > NIR] : 1.084e-05
##
##                  Kappa : 0.6667
##
##  Mcnemar's Test P-Value : 0.3428
##
##             Sensitivity : 0.7941
##             Specificity : 0.8846
##          Pos Pred Value : 0.9000
##          Neg Pred Value : 0.7667
##              Prevalence : 0.5667
##          Detection Rate : 0.4500
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.8394
##
##        'Positive' Class : 0
##
```

Conclusão:

O modelo Completo teve acurácia de 83%, indicando bom desempenho preditivo. As demais métricas, especificidade e sensibilidade, também apresentaram bons desempenhos (acima de 75%).

Modelos Alternativos

Modelo 1:

Retendo somente as variáveis que foram significativas no modelo completo.

```
modelo1 <- glm(output ~ sex + cp+ oldpeak + caa + exng, data = data_train, family = binomial)
summary(modelo1)
```

```
##
## Call:
## glm(formula = output ~ sex + cp + oldpeak + caa + exng, family = binomial,
##     data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4050  -0.5048   0.2419   0.5371   2.6012
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.0252     0.4870   4.159 3.20e-05 ***
## sex1         -1.2616     0.4366  -2.890  0.00386 **
## cp1           1.4825     0.5819   2.547  0.01085 *
## cp2           2.0713     0.4845   4.275 1.91e-05 ***
## cp3           1.9068     0.6585   2.895  0.00379 **
## oldpeak      -0.9660     0.2266  -4.263 2.02e-05 ***
## caa          -0.7841     0.1841  -4.258 2.06e-05 ***
## exng1        -1.2143     0.4341  -2.797  0.00515 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 331.07  on 240  degrees of freedom
## Residual deviance: 184.54  on 233  degrees of freedom
## AIC: 200.54
##
## Number of Fisher Scoring iterations: 5
```

```
gtsummary::tbl_regression(modelo1, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
sex			
0	—	—	
1	0.28	0.12, 0.65	0.004
cp			
0	—	—	
1	4.40	1.48, 14.9	0.011
2	7.94	3.18, 21.5	<0.001
3	6.73	1.94, 26.2	0.004

[1] OR = Odds Ratio, CI = Confidence Interval

Characteristic	OR[1]	95% CI[1]	p-value
oldpeak	0.38	0.24, 0.58	<0.001
caa	0.46	0.31, 0.65	<0.001
exng			
0	—	—	
1	0.30	0.12, 0.69	0.005

[1] OR = Odds Ratio, CI = Confidence Interval

```
# Predicao com dados teste
      fitted_results1 <- dismo::predict(modelo1, newdata = data_test, type = "response")
      fitted_results_cat1 <- ifelse(fitted_results1 > 0.7,1,0) #threshold


# Unir: Observados vs Preditos
      data_test_pred1 <- data_test |>
                      add_column(fitted =  fitted_results1,
                                 fitted_cat = fitted_results_cat1 ) |>
                      mutate(fitted_exp = exp(fitted_results1))

      data_test_pred1 <- data_test_pred1 |>
        mutate(fitted_cat = as_factor(fitted_cat))

# Metricas Desempenho: Tabela de confusão
      caret::confusionMatrix(data = data_test_pred1$output, reference = data_test_pred1$fitted_cat)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 25  5
##          1 11 19
##
##                Accuracy : 0.7333
##                  95% CI : (0.6034, 0.8393)
##     No Information Rate : 0.6
##     P-Value [Acc > NIR] : 0.02208
##
##                   Kappa : 0.4667
##
##  Mcnemar's Test P-Value : 0.21130
##
##             Sensitivity : 0.6944
##             Specificity : 0.7917
##          Pos Pred Value : 0.8333
##          Neg Pred Value : 0.6333
##              Prevalence : 0.6000
##          Detection Rate : 0.4167
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.7431
##
##        'Positive' Class : 0
##
```

Modelo 2:

Retendo somente as variáveis consideradas importantes na Análise individual realizada na etapa de Análise Descritiva. As variáveis não significativas ou com baixa associação com chance de doença cardíaca (menos de 10% na redução ou no aumento) não foram consideradas.

```
modelo2 <- glm(output ~ sex+ cp+ thall+ oldpeak + caa + exng+ restecg + slp, data = data_train, fami
summary(modelo2)
```

```
##
## Call:
## glm(formula = output ~ sex + cp + thall + oldpeak + caa + exng +
##     restecg + slp, family = binomial, data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9271  -0.3978   0.1565   0.4505   2.7971
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.2403     1.3193   1.698 0.089488 .
## sex1         -1.1378     0.5468  -2.081 0.037470 *
## cp1           0.9587     0.6615   1.449 0.147260
## cp2           1.9701     0.5479   3.595 0.000324 ***
## cp3           1.8855     0.6798   2.773 0.005547 **
## thall2       -0.1264     0.8034  -0.157 0.875013
```

```
## thall3       -1.6046     0.7799  -2.057 0.039654 *
## oldpeak      -0.5990     0.2682  -2.233 0.025536 *
## caa          -0.9841     0.2286  -4.305 1.67e-05 ***
## exng1        -1.0068     0.4997  -2.015 0.043950 *
## restecg1      0.7762     0.4248   1.827 0.067661 .
## restecg2     -0.4068     2.2982  -0.177 0.859490
## slp1         -0.9484     0.8840  -1.073 0.283308
## slp2          0.5475     0.9663   0.567 0.570947
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 331.07  on 240  degrees of freedom
## Residual deviance: 157.86  on 227  degrees of freedom
## AIC: 185.86
##
## Number of Fisher Scoring iterations: 6
```

```
gtsummary::tbl_regression(modelo2, exponentiate = TRUE)
```

Characteristic	OR[1]	95% CI[1]	p-value
sex			
0	—	—	
1	0.32	0.11, 0.91	0.037
cp			
0	—	—	
1	2.61	0.74, 10.3	0.15
2	7.17	2.55, 22.2	<0.001
3	6.59	1.82, 26.8	0.006
thall			
1	—	—	
1	—	—	
2	0.88	0.18, 4.32	0.9
3	0.20	0.04, 0.93	0.040
oldpeak	0.55	0.32, 0.91	0.026
caa	0.37	0.23, 0.57	<0.001
exng			
0	—	—	
1	0.37	0.14, 0.97	0.044
restecg			
0	—	—	
1	2.17	0.95, 5.10	0.068
2	0.67	0.01, 34.9	0.9
slp			
0	—	—	
1	0.39	0.06, 2.08	0.3
2	1.73	0.24, 10.9	0.6

[1] OR = Odds Ratio, CI = Confidence Interval

```
# Predicao com dados teste
      fitted_results2 <- dismo::predict(modelo2, newdata = data_test, type = "response")
      fitted_results_cat2 <- ifelse(fitted_results2 > 0.7,1,0) #threshold


# Unir: Observados vs Preditos
      data_test_pred2 <- data_test |>
                      add_column(fitted =  fitted_results2,
                                 fitted_cat = fitted_results_cat2 ) |>
                      mutate(fitted_exp = exp(fitted_results2))

      data_test_pred2 <- data_test_pred2 |>
        mutate(fitted_cat = as_factor(fitted_cat))
```

```
# Metricas Desempenho: Tabela de confusão
      caret::confusionMatrix(data = data_test_pred2$output, reference = data_test_pred2$fitted_cat)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 26  4
##          1  8 22
##
##                Accuracy : 0.8
##                  95% CI : (0.6767, 0.8922)
##     No Information Rate : 0.5667
##     P-Value [Acc > NIR] : 0.0001278
##
##                   Kappa : 0.6
##
##  Mcnemar's Test P-Value : 0.3864762
##
##             Sensitivity : 0.7647
##             Specificity : 0.8462
##          Pos Pred Value : 0.8667
##          Neg Pred Value : 0.7333
##              Prevalence : 0.5667
##          Detection Rate : 0.4333
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.8054
##
##        'Positive' Class : 0
##
```

Comparação entre os três modelos

Ao analisar comparativamente os modelos, podemos notar que o **Modelo Completo** possui a maior acurácia. Entretanto, o **Modelo 2** possui o menor AIC e uma acurácia muito próxima da obtida com o Modelo Completo. Ao olharmos para o intervalo de confiança, notamos que o Modelo 2 apresenta o limite inferior menor do que 70% de desempenho (0.6767, 0.8922), enquanto no Modelo Completo olimite inferior do intervalo permace acima dos 70% (0.7148, 0.9171). Portanto, podemos escolher entre um Modelo mais simples, que seria o Modelo 2, ou o Modelo Completo, que possui o desempenho preditivo ligeiramente melhor.

```
bind_rows(Modelo = c("Modelo_Completo", "Modelo 1", "Modelo 2"),
          AIC = as.numeric(c(modelo_completo$aic,
                     modelo1$aic,
                     modelo2$aic))) |>
  mutate(Acuracia = c (acuracia_completo,acuracia_1, acuracia_2)) |>
  knitr::kable()
```

Modelo	AIC	Acuracia
Modelo_Completo	193.0536	0.833
Modelo 1	200.5419	0.733
Modelo 2	185.8640	0.800