

Avaliações_ME 2

2024-10-07

Contents

Ajuste modelo Geral	3
QUIZ	6
Perguta 1	6
Perguta 2	7
Perguta 3	7
Perguta 4	8
Perguta 5	8
Perguta 6	9
Perguta 7	10
Perguta 8	10
Perguta 9 - ERRO	11
Perguta 10 - ERRO	13
Perguta 11	14
Perguta 12	14

```
knitr::opts_chunk$set(echo = TRUE)
```

```
require("ISLR")
```

```
## Carregando pacotes exigidos: ISLR
```

```
require("ggplot2")
```

```
## Carregando pacotes exigidos: ggplot2
```

```
require("GGally")
```

```
## Carregando pacotes exigidos: GGally
```

```
## Registered S3 method overwritten by 'GGally':
```

```
##   method from
```

```
##   +.gg    ggplot2
```

```
require("leaps") ## seleção de variáveis
```

```
## Carregando pacotes exigidos: leaps
```

```
require("car")
```

```
## Carregando pacotes exigidos: car
```

```
## Carregando pacotes exigidos: carData
```

```
require(tidyverse)
```

```
## Carregando pacotes exigidos: tidyverse
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v dplyr      1.1.4      v readr      2.1.5
```

```
## v forcats    1.0.0      v stringr    1.5.1
```

```
## v lubridate  1.9.3      v tibble     3.2.1
```

```
## v purrr      1.0.2      v tidyr      1.3.1
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()    masks stats::lag()
```

```
## x dplyr::recode() masks car::recode()
```

```
## x purrr::some()   masks car::some()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
require(caret)
```

```
## Carregando pacotes exigidos: caret
```

```
## Carregando pacotes exigidos: lattice
```

```
##
```

```
## Anexando pacote: 'caret'
```

```
##
```

```
## O seguinte objeto é mascarado por 'package:purrr':
```

```
##
```

```
## lift
```

```
require(MASS)
```

```
## Carregando pacotes exigidos: MASS
```

```
##
```

```
## Anexando pacote: 'MASS'
```

```
##
```

```
## O seguinte objeto é mascarado por 'package:dplyr':
```

```
##
```

```
## select
```

```
require(labeling)
```

```
## Carregando pacotes exigidos: labeling
```

```
require(faraway)
```

```
## Carregando pacotes exigidos: faraway
```

```
##
```

```
## Anexando pacote: 'faraway'
```

```
##
```

```
## O seguinte objeto é mascarado por 'package:lattice':
```

```
##
```

```
##     melanoma
```

```
##
```

```
## Os seguintes objetos são mascarados por 'package:car':
```

```
##
```

```
##     logit, vif
```

```
##
```

```
## O seguinte objeto é mascarado por 'package:GGally':
```

```
##
```

```
##     happy
```

```
require(pscl)
```

```
## Carregando pacotes exigidos: pscl
```

```
## Classes and Methods for R originally developed in the
```

```
## Political Science Computational Laboratory
```

```
## Department of Political Science
```

```
## Stanford University (2002-2015),
```

```
## by and under the direction of Simon Jackman.
```

```
## hurdle and zeroinfl functions by Achim Zeileis.
```

```
require(pROC)
```

```
## Carregando pacotes exigidos: pROC
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
```

```
## Anexando pacote: 'pROC'
```

```
##
```

```
## Os seguintes objetos são mascarados por 'package:stats':
```

```
##
```

```
##     cov, smooth, var
```

Ajuste modelo Geral

```
#Carseats
```

```
data("wbca")
```

```
#help("wbca")
```

```
head(wbca, 10) ### Visualizando as dez primeiras linhas
```

```
##      Class Adhes BNucl Chrom Epith Mitos NNucl Thick UShap USize
## 1      1      1      1      3      2      1      1      5      1      1
## 2      1      5     10      3      7      1      2      5      4      4
## 3      1      1      2      3      2      1      1      3      1      1
## 4      1      1      4      3      3      1      7      6      8      8
## 5      1      3      1      3      2      1      1      4      1      1
## 6      0      8     10      9      7      1      7      8     10     10
## 7      1      1     10      3      2      1      1      1      1      1
## 8      1      1      1      3      2      1      1      2      2      1
## 9      1      1      1      1      2      5      1      2      1      1
## 10     1      1      1      2      2      1      1      4      1      2
```

```
str(wbca)
```

```
## 'data.frame':    681 obs. of  10 variables:
## $ Class: int  1 1 1 1 1 0 1 1 1 1 ...
## $ Adhes: int  1 5 1 1 3 8 1 1 1 1 ...
## $ BNucl: int  1 10 2 4 1 10 10 1 1 1 ...
## $ Chrom: int  3 3 3 3 3 9 3 3 1 2 ...
## $ Epith: int  2 7 2 3 2 7 2 2 2 2 ...
## $ Mitos: int  1 1 1 1 1 1 1 1 5 1 ...
## $ NNucl: int  1 2 1 7 1 7 1 1 1 1 ...
## $ Thick: int  5 5 3 6 4 8 1 2 2 4 ...
## $ UShap: int  1 4 1 8 1 10 1 2 1 1 ...
## $ USize: int  1 4 1 8 1 10 1 1 1 2 ...
```

```
dim(wbca) ### Acessando a dimensão da base
```

```
## [1] 681 10
```

```
summary(wbca) ### Resumo das variáveis
```

```
##      Class      Adhes      BNucl      Chrom
## Min.   :0.0000  Min.   : 1.000  Min.   : 1.000  Min.   : 1.000
## 1st Qu.:0.0000  1st Qu.: 1.000  1st Qu.: 1.000  1st Qu.: 2.000
## Median :1.0000  Median : 1.000  Median : 1.000  Median : 3.000
## Mean   :0.6505  Mean   : 2.816  Mean   : 3.542  Mean   : 3.433
## 3rd Qu.:1.0000  3rd Qu.: 4.000  3rd Qu.: 6.000  3rd Qu.: 5.000
## Max.   :1.0000  Max.   :10.000  Max.   :10.000  Max.   :10.000
##      Epith      Mitos      NNucl      Thick
## Min.   : 1.000  Min.   : 1.000  Min.   : 1.000  Min.   : 1.000
## 1st Qu.: 2.000  1st Qu.: 1.000  1st Qu.: 1.000  1st Qu.: 2.000
## Median : 2.000  Median : 1.000  Median : 1.000  Median : 4.000
## Mean   : 3.231  Mean   : 1.604  Mean   : 2.859  Mean   : 4.436
## 3rd Qu.: 4.000  3rd Qu.: 1.000  3rd Qu.: 4.000  3rd Qu.: 6.000
## Max.   :10.000  Max.   :10.000  Max.   :10.000  Max.   :10.000
##      UShap      USize
## Min.   : 1.000  Min.   : 1.00
## 1st Qu.: 1.000  1st Qu.: 1.00
## Median : 1.000  Median : 1.00
## Mean   : 3.204  Mean   : 3.14
## 3rd Qu.: 5.000  3rd Qu.: 5.00
## Max.   :10.000  Max.   :10.00
```

```

# Redefinir o nível de referência da variável Class (1 = maligno, 0 = benigno)
wbca$Class <- factor(wbca$Class, levels = c('1', '0'))

# Dividir a base de dados em base de ajuste (500 primeiras linhas) e base de validação (resto)
base_ajuste <- wbca[1:500, ]
base_validacao <- wbca[501:681, ]

# Ajustar o modelo de regressão logística
modelo_logistico <- glm(Class ~ Adhes + BNucl + Thick, data = base_ajuste, family = binomial)

# Ver o resumo do modelo ajustado
summary(modelo_logistico)

```

```

##
## Call:
## glm(formula = Class ~ Adhes + BNucl + Thick, family = binomial,
##      data = base_ajuste)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.66162    0.96280  -8.996 < 2e-16 ***
## Adhes        0.50432    0.11748   4.293 1.76e-05 ***
## BNucl        0.63393    0.09394   6.748 1.50e-11 ***
## Thick       0.89154    0.13201   6.753 1.44e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 671.36  on 499  degrees of freedom
## Residual deviance: 116.62  on 496  degrees of freedom
## AIC: 124.62
##
## Number of Fisher Scoring iterations: 7

```

```

# Instalar pacote para métricas de ajuste, se necessário -> install.packages("pscl")
# Obter o pseudo-R^2 de McFadden
pR2(modelo_logistico)

```

```

## fitting null model for pseudo-r2

```

```

##           llh      llhNull          G2      McFadden      r2ML      r2CU
## -58.3106800 -335.6782179  554.7350757   0.8262899   0.6702664   0.9071583

```

```

# Fazer previsões na base de validação
previsoes <- predict(modelo_logistico, newdata = base_validacao, type = "response")

# Classificar como maligno (1) ou benigno (0) com um limiar de 0.5
previsao_class <- ifelse(previsoes > 0.5, '1', '0')

# Comparar com a variável Class na base de validação

```

```
tabela_confusao <- table(previsao_class, base_validacao$Class)
```

```
# Ver a matriz de confusão  
print(tabela_confusao)
```

```
##  
## previsao_class    1    0  
##                0 141    4  
##                1    0  36
```

QUIZ

Pergunta 1

Qual a chance estimada de tumor maligno para uma observação com as seguintes características:

- Adhes = 4
- BNucl = 7
- Thick = 5

Obs minha:

$$Chance = \frac{prob.est}{(1 - prob.est)}$$

```
# Criar uma nova observação com os valores fornecidos  
nova_observacao <- data.frame(Adhes = 4, BNucl = 7, Thick = 5)  
  
# Prever a probabilidade de tumor maligno para a nova observação  
probabilidade_maligno <- predict(modelo_logistico, newdata = nova_observacao, type = "response")  
  
# Exibir a probabilidade  
probabilidade_maligno
```

```
##          1  
## 0.9047231
```

```
# Calcular a chance (odds)  
chance_maligno <- probabilidade_maligno / (1 - probabilidade_maligno)  
chance_maligno
```

```
##          1  
## 9.495722
```

Perguta 2

Qual a probabilidade estimada de tumor maligno para uma observação com as seguintes características:

- Adhes = 4
- BNucl = 7
- Thick = 5

```
# Criar uma nova observação com os valores fornecidos
nova_observacao <- data.frame(Adhes = 4, BNucl = 7, Thick = 5)

# Prever a probabilidade de tumor maligno
probabilidade_maligno <- predict(modelo_logistico, newdata = nova_observacao, type = "response")

# Exibir a probabilidade estimada
probabilidade_maligno

##          1
## 0.9047231
```

Perguta 3

Considere a seguinte conjectura:

Estima-se um que a chance de tumor maligno aumente K% para um aumento unitário em BNucl mantendo fixos Adhes e Thick.

Qual das alternativas abaixo apresenta o valor de K?

```
# Ver o resumo do modelo para obter os coeficientes
summary(modelo_logistico)

##
## Call:
## glm(formula = Class ~ Adhes + BNucl + Thick, family = binomial,
##      data = base_ajuste)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.66162    0.96280  -8.996 < 2e-16 ***
## Adhes        0.50432    0.11748   4.293 1.76e-05 ***
## BNucl        0.63393    0.09394   6.748 1.50e-11 ***
## Thick        0.89154    0.13201   6.753 1.44e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 671.36  on 499  degrees of freedom
## Residual deviance: 116.62  on 496  degrees of freedom
## AIC: 124.62
##
## Number of Fisher Scoring iterations: 7
```

```
# Extrair o coeficiente de BNucl
beta_BNucl <- coef(modelo_logistico)["BNucl"]

# Calcular o aumento percentual nas chances (odds)
K <- (exp(beta_BNucl) - 1) * 100
K
```

```
##      BNucl
## 88.49985
```

Pergunta 4

Considere a seguinte conjectura:

Estima-se um que a chance de tumor maligno aumente K% para um aumento de duas unidades em Adhes mantendo fixos BNucl e Thick.

Qual das alternativas abaixo apresenta o valor de K?

```
# Extrair o coeficiente de Adhes do modelo
beta_Adhes <- coef(modelo_logistico)["Adhes"]

# Calcular o aumento percentual nas chances (odds) para um aumento de duas unidades
K <- (exp(2 * beta_Adhes) - 1) * 100
K
```

```
##      Adhes
## 174.1883
```

Pergunta 5

O número de variáveis explicativas que estão significativamente associadas ao tipo de tumor, ao nível de significância de 5%, é:

```
# Exibir o resumo do modelo ajustado
summary(modelo_logistico)

##
## Call:
## glm(formula = Class ~ Adhes + BNucl + Thick, family = binomial,
##      data = base_ajuste)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.66162    0.96280  -8.996 < 2e-16 ***
## Adhes        0.50432    0.11748   4.293 1.76e-05 ***
## BNucl        0.63393    0.09394   6.748 1.50e-11 ***
## Thick       0.89154    0.13201   6.753 1.44e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```



```
##
## Null deviance: 671.36 on 499 degrees of freedom
## Residual deviance: 116.62 on 496 degrees of freedom
## AIC: 124.62
##
## Number of Fisher Scoring iterations: 7

# Extrair os valores de p dos coeficientes
valores_p <- summary(modelo_logistico)$coefficients[, "Pr(>|z|)"]

# Contar o número de variáveis significativas ao nível de 5%
variaveis_significativas <- sum(valores_p < 0.05)

# Exibir o número de variáveis significativas
variaveis_significativas

## [1] 4
```

Perguta 6

Qual das alternativas a seguir apresenta um intervalo de confiança (95%) para a chance estimada de tumor maligno para uma observação com as seguintes características:

- Adhes = 4
- BNucl = 7
- Thick = 5

```
# Criar a nova observação
nova_observacao <- data.frame(Adhes = 4, BNucl = 7, Thick = 5)

# Prever as log-odds e o intervalo de confiança
previsao <- predict(modelo_logistico, newdata = nova_observacao, type = "link", se.fit = TRUE)

# Calcular o intervalo de confiança para as log-odds (95%)
z_value <- qnorm(0.975) # Valor z para 95% de confiança
IC_logodds_lower <- previsao$fit - z_value * previsao$se.fit
IC_logodds_upper <- previsao$fit + z_value * previsao$se.fit

# Converter as log-odds para chances (odds)
odds_lower <- exp(IC_logodds_lower)
odds_upper <- exp(IC_logodds_upper)

# Exibir o intervalo de confiança para a chance
odds_lower

## 1
## 3.9861

odds_upper

## 1
## 22.62079
```

Perguta 7

Qual das alternativas a seguir apresenta um intervalo de confiança (95%) para a probabilidade estimada de tumor maligno para uma observação com as seguintes características:

- Adhes = 4
- BNucl = 7
- Thick = 5

```
# Criar a nova observação
nova_observacao <- data.frame(Adhes = 4, BNucl = 7, Thick = 5)

# Prever as log-odds e o intervalo de confiança
previsao <- predict(modelo_logistico, newdata = nova_observacao, type = "link", se.fit = TRUE)

# Valor z para o intervalo de confiança de 95%
z_value <- qnorm(0.975)

# Calcular o intervalo de confiança para as log-odds (fit = previsão das log-odds, se.fit = erro padrão)
IC_logodds_lower <- previsao$fit - z_value * previsao$se.fit
IC_logodds_upper <- previsao$fit + z_value * previsao$se.fit

# Converter as log-odds para probabilidades (usando a função logística)
prob_lower <- exp(IC_logodds_lower) / (1 + exp(IC_logodds_lower))
prob_upper <- exp(IC_logodds_upper) / (1 + exp(IC_logodds_upper))

# Exibir o intervalo de confiança para a probabilidade
prob_lower
```

```
##          1
## 0.7994425
```

```
prob_upper
```

```
##          1
## 0.9576644
```

Perguta 8

Com base na amostra de validação, a acurácia estimada do modelo, ao classificar como tumor maligno as observações com probabilidade estimada maior que 0.50, e como tumor benigno as observações com probabilidade estimada menor que 0.50, é igual a:

```
prob_validacao <- predict(modelo_logistico, newdata = base_validacao, type = "response")

# Classificar com base na probabilidade de corte 0.50
predicoes <- ifelse(prob_validacao > 0.50, "1", "0")

# Comparar as previsões com as classes reais
tabela_confusao <- table(Predito = predicoes, Real = base_validacao$Class)
```

```
# Calcular a acurácia
acuracia <- sum(diag(tabela_confusao)) / sum(tabela_confusao)
acuracia
```

```
## [1] 0.9779006
```

Perguta 9 - ERRO

Com base na amostra de validação, a sensibilidade estimada do modelo, ao classificar como tumor maligno as observações com probabilidade estimada maior que 0.50, e como tumor benigno as observações com probabilidade estimada menor que 0.50, é igual a:

```
predicoes <- predict(modelo_logistico, newdata = base_validacao, type = 'response')

tab_pred <- table(ifelse(predicoes < 0.5, 'Pred_No', 'Pred_Yes'), base_validacao$Class)
tab_pred
```

```
##
##           1    0
## Pred_No 141    4
## Pred_Yes  0   36
```

```
# Cálculo da acurácia
TP <- tab_pred["Pred_Yes", '1'] # Verdadeiros Positivos
TN <- tab_pred["Pred_No", '0']  # Verdadeiros Negativos
FP <- tab_pred["Pred_Yes", '0'] # Falsos Positivos
FN <- tab_pred["Pred_No", '1']  # Falsos Negativos
total <- sum(tab_pred)          # Total de observações
```

```
# Acurácia
acuracia <- (TP + TN) / total
acuracia
```

```
## [1] 0.02209945
```

```
sensibilidade <- (TP)/(TP + FN)
sensibilidade
```

```
## [1] 0
```

```
especificidade <- (TN)/(TN + FP)
especificidade
```

```
## [1] 0.1
```

```
# Ajustar o modelo de regressão logística com base na amostra de ajuste
modelo_logistico <- glm(Class ~ Adhes + BNucl + Thick, data = base_ajuste, family = binomial)

# Prever probabilidades para a amostra de validação
```

```
#prob_validacao <- predict(modelo_logistico, newdata = base_validacao, type = "response")
#print(prob_validacao)

# Classificar com base na probabilidade de corte 0.50
#predicoes <- ifelse(prob_validacao >= 0.50, "1", "0")

# Criar uma matriz de confusão
tabela_confusao2 <- table(Predito = previsao_class, Real = base_validacao$Class)
#print(tabela_confusao2)
print(tabela_confusao2)
```

```
##          Real
## Predito   1   0
##          0 141   4
##          1   0  36
```

```
# Calcular a sensibilidade
verdadeiros_positivos <- tabela_confusao2["1", "1"]
falsos_negativos <- tabela_confusao2["0", "1"]
sensibilidade <- verdadeiros_positivos / (verdadeiros_positivos + falsos_negativos)

# Exibir a sensibilidade
sensibilidade
```

```
## [1] 0
```

```
# Supondo que o modelo foi ajustado e as predições foram feitas
# Prever as probabilidades para a amostra de validação
prob_validacao <- predict(modelo_logistico, newdata = base_validacao, type = "response")

# Classificar com base na probabilidade de corte 0.50
predicoes <- ifelse(prob_validacao > 0.50, "1", "0")

# Criar uma matriz de confusão
tabela_confusao <- table(Predito = predicoes, Real = base_validacao$Class)

# Calcular a sensibilidade
verdadeiros_positivos <- tabela_confusao["1", "1"]
falsos_negativos <- tabela_confusao["0", "1"]

# Se não houver verdadeiros positivos, a sensibilidade será 0
if (is.na(verdadeiros_positivos) || (verdadeiros_positivos + falsos_negativos) == 0) {
  sensibilidade <- 0
} else {
  sensibilidade <- verdadeiros_positivos / (verdadeiros_positivos + falsos_negativos)
}

# Exibir a sensibilidade
sensibilidade
```

```
## [1] 0
```

```

# Carregar o pacote necessário
library(faraway)

# Redefinir a variável Class para modelar a probabilidade de tumor maligno
wbca$Class <- factor(wbca$Class, levels = c('1', '0'))

# Dividir os dados em amostra de ajuste e amostra de validação
ajuste <- wbca[1:500, ]
validacao <- wbca[501:681, ]

# Ajustar o modelo de regressão logística com base na amostra de ajuste
modelo_logistico <- glm(Class ~ Adhes + BNucl + Thick, data = ajuste, family = binomial)

# Prever probabilidades para a amostra de validação
prob_validacao <- predict(modelo_logistico, newdata = base_validacao, type = "response")

# Classificar com base na probabilidade de corte 0.50
predicoes <- ifelse(prob_validacao > 0.50, "1", "0")

# Criar uma matriz de confusão
tabela_confusao <- table(Predito = predicoes, Real = validacao$Class)

# Exibir a matriz de confusão
print(tabela_confusao)

```

```

##           Real
## Predito   1    0
##           0 141   4
##           1    0  36

```

```

# Calcular a sensibilidade
verdadeiros_positivos <- tabela_confusao["1", "1"]
falsos_negativos <- tabela_confusao["0", "1"]
sensibilidade <- verdadeiros_positivos / (verdadeiros_positivos + falsos_negativos)

# Exibir a sensibilidade
sensibilidade

```

```
## [1] 0
```

Perguta 10 - ERRO

Com base na amostra de validação, a especificidade estimada do modelo, ao classificar como tumor maligno as observações com probabilidade estimada maior que 0.50, e como tumor benigno as observações com probabilidade estimada menor que 0.50, é igual a:

```

# Criar a matriz de confusão (caso ainda não tenha feito)
tabela_confusao <- table(Predito = predicoes, Real = validacao$Class)

# Exibir a matriz de confusão
print(tabela_confusao)

```

```
##           Real
## Predito   1   0
##           0 141  4
##           1   0 36
```

```
# Calcular a especificidade
verdadeiros_negativos <- tabela_confusao["0", "0"] # Tumores benignos corretamente classificados
falsos_positivos <- tabela_confusao["1", "0"] # Tumores benignos que foram classificados como mal
especificidade <- verdadeiros_negativos / (verdadeiros_negativos + falsos_positivos)

# Exibir a especificidade
especificidade
```

```
## [1] 0.1
```

Perguta 11

Com base na amostra de validação, a área sob a curva ROC produzida pelo modelo é igual a

```
# Calcular a curva ROC e a AUC
roc_curve <- roc(validacao$Class, prob_valdacao)
```

```
## Setting levels: control = 1, case = 0
```

```
## Setting direction: controls < cases
```

```
# Exibir a AUC
area_auc <- auc(roc_curve)
print(area_auc)
```

```
## Area under the curve: 0.9928
```

Perguta 12

Com base na amostra de validação, e usando o método de Youden, a regra de classificação ótima ao se considerar as informações fornecidas na sequência consiste em classificar um tumor como maligno caso a probabilidade estimada seja superior a K.

- Prevalência de tumor maligno igual a 0.20;
- Custo de classificar um tumor positivo como negativo é três vezes o custo de classificar um tumor negativo como positivo.

Qual o valor de K?

```
# Suponha que você tenha a matriz de confusão com base em um modelo existente
# A prevalência é 0.20
prevalencia <- 0.20
custo_positivo <- 3
custo_negativo <- 1
```

```

# Cálculo do custo total para cada ponto de corte possível
pontos_de_corte <- seq(0, 1, by = 0.01)
custos_totais <- numeric(length(pontos_de_corte))

for (i in seq_along(pontos_de_corte)) {
  k <- pontos_de_corte[i]

  # Calcule as previsões de acordo com o ponto de corte
  predicoes <- ifelse(prob_validacao > k, "1", "0")

  # Crie a matriz de confusão
  tabela_confusao <- table(Predito = predicoes, Real = validacao$Class)

  # Calcule a sensibilidade e a especificidade
  if ("1" %in% rownames(tabela_confusao) && "0" %in% rownames(tabela_confusao)) {
    verdadeiros_positivos <- tabela_confusao["1", "1"]
    falsos_negativos <- tabela_confusao["0", "1"]
    verdadeiros_negativos <- tabela_confusao["0", "0"]
    falsos_positivos <- tabela_confusao["1", "0"]

    sensibilidade <- verdadeiros_positivos / (verdadeiros_positivos + falsos_negativos)
    especificidade <- verdadeiros_negativos / (verdadeiros_negativos + falsos_positivos)
  } else {
    sensibilidade <- 0
    especificidade <- 0
  }

  # Calcule o custo total para o ponto de corte
  custo_total <- custo_positivo * (falsos_negativos / (falsos_negativos + verdadeiros_positivos)) +
    custo_negativo * (falsos_positivos / (falsos_positivos + verdadeiros_negativos))
  custos_totais[i] <- custo_total
}

# Encontre o ponto de corte que minimiza o custo total
indice_otimo <- which.min(custos_totais)
k_otimo <- pontos_de_corte[indice_otimo]

# Exibir o valor de K
print(k_otimo)

## [1] 0.01

# Supondo que r1 é um objeto de curva ROC criado anteriormente, e 'validacao$Class' e 'prob_validacao'
# Exemplo de ajuste do modelo e cálculo da curva ROC

best_coords_equal_cost <- coords(roc_curve, x = "best", best.method = "youden")
print(best_coords_equal_cost)

```

```

## threshold specificity sensitivity
## 1 0.1194988 0.964539 0.975

```

```
# Custos do falso negativo são 3 vezes o do falso positivo
```

```
r2 <- roc(validacao$Class, prob_validacao)
```

```
## Setting levels: control = 1, case = 0
```

```
## Setting direction: controls < cases
```

```
# Encontrar as coordenadas para a classificação ótima usando o método de Youden
```

```
best_coords_equal_cost <- coords(r2, x = "best", best.method = "youden", best.weights = c(1, (1/3)))  
print(best_coords_equal_cost)
```

```
## threshold specificity sensitivity
```

```
## 1 0.2533834 0.9787234 0.95
```

```
# Custos do falso negativo são 3 vezes o do falso positivo
```

```
# Cálculo dos escores de crédito
```

```
cred_escores <- 100 * (1 - prob_validacao) # Assumindo que prob_validacao contém as probabilidades de  
head(cred_escores, 20) # Exibir os primeiros 20 escores de crédito
```

```
##      501      502      503      504      505      506  
## 99.86839922 99.86839922 98.71875089 5.97058237 99.86839922 98.12424805  
##      507      508      509      510      511      512  
## 0.43178442 0.01282205 99.22228811 98.71875089 98.12424805 98.12424805  
##      513      514      515      516      517      518  
## 84.15529815 98.12424805 0.43580025 98.12424805 99.86839922 99.22228811  
##      519      520  
## 99.67964874 99.78227505
```

```
# Histograma dos escores de crédito
```

```
hist(cred_escores, main = 'Distribuição dos Escores de Crédito', xlab = 'Escore de Crédito', ylab = 'Fr
```


Distribuição dos Escores de Crédito

