```
# Pacotes ----
if (!require("pacman")) install.packages("pacman")
pacman::p_load(
  readxl,
  tidyverse,
  cowplot,
  mdscore,
  AICcmodavg,
  questionr,
  mlpack,
  ResourceSelection,
  lmtest,
  car,
  stats,
  knitr,
  pROC,
  ROCit,
  labelled,
  compareGroups,
  arm,
  performance
)

# Dados de análise e treino dos modelos ----
df <- read_excel("arquivos/Amostra_g06_Bruno_Rafael.xlsx")
colnames(df) <- c("ID", # Identificação do paciente
                  "resultado_radiografia", # 0 = negativo | 1 - positivo
                  "estagio_tumor", # 0 - menos grave | 1 - mais grave
                  "nivel_fosfatase_acida", # x100
                  "envolvimento_nodal" # 0 - não | 1 - sim
                  )


var_label(df) <- list(
  resultado_radiografia = "Resultado da Radiografia",
  estagio_tumor         = "Estágio do Tumor",
  nivel_fosfatase_acida = "Nível da Fosfatase Ácida",
  envolvimento_nodal    = "Envolvimento Nodal (x100)"
)

# Dados de validação do modelo (teste) ----
teste <- read_excel("arquivos/Amostra_VALIDACAO.xlsx")
colnames(teste) <- c("ID", # Identificação do paciente
                     "resultado_radiografia", # 0 = negativo | 1 - positivo
                     "estagio_tumor", # 0 - menos grave | 1 - mais grave
                     "nivel_fosfatase_acida", # x100
                     "envolvimento_nodal" # 0 - não | 1 - sim
)
#####################################################################
# Parte 1) - Análise exploratória ----

dados = df %>%
  mutate(resultado_radiografia = case_when(
    resultado_radiografia == "0" ~ "Negativo",
    resultado_radiografia == "1" ~ "Positivo"
   ), estagio_tumor = case_when(
    estagio_tumor == "0" ~ "- grave",
    estagio_tumor == "1" ~ "+ grave"
    ), envolvimento_nodal = case_when(
      envolvimento_nodal == "0" ~ "Não",
      envolvimento_nodal == "1" ~ "Sim"
      ), nivel_fosfatase_acida = nivel_fosfatase_acida)

dados = dados[,2:5]
dados$envolvimento_nodal = factor(dados$envolvimento_nodal)
dados$estagio_tumor = factor(dados$estagio_tumor)
dados$resultado_radiografia = factor(dados$resultado_radiografia)

compareGroups(envolvimento_nodal ~ . ,
```

```
              data=dados,
              method = c(nivel_fosfatase_acida=NA), alpha= 0.05 # Testando a
normalidade para decidir o teste
              )

tabela = compareGroups(envolvimento_nodal ~ . ,
              data=dados,
              method = c(nivel_fosfatase_acida=NA), alpha= 0.05 # Testando a
normalidade para decidir o teste
              )

createTable(tabela, show.ratio=TRUE)

descrTable(dados)

plot(tabela)
plot(tabela, bivar=TRUE)

# rm(dados,tabela)
#####################################################################
# Parte 2) Regressão logística: nivel_fosfatase_acida EXPLICANDO envolvimento_nodal ---
-


# Somente intercepto
fit0 <- glm(envolvimento_nodal ~ 1,
            family=binomial(link=logit),
            data=df)

plot(
  jitter(envolvimento_nodal, 0.01) ~ nivel_fosfatase_acida,
  xlab = "nivel_fosfatase_acida",
  ylab = "envolvimento_nodal",
  data = df,
  pch = 16
)

curve(
  predict(fit0, data.frame(nivel_fosfatase_acida = x), type = "resp"),
  add = T,
  col = "blue",
  lwd = 2
)



get_confint <- function(fit, variavel, digitos){

  if(!is.null(dim(confint(fit)))){

    round(exp(confint(fit)[variavel, ]), digitos)

  } else {

    round(confint(fit), digitos) |>
      str_c( collapse = "; ")

  }

}


fit0_pred <- predict(fit0, type="response", se.fit=TRUE)
fit0_fit <- fit0_pred$fit[1]
fit0_se <- fit0_pred$se.fit[1]

  # Intervalo de confiança para a probabilidade
Lb_fit0 <- fit0_fit - qnorm(0.975) * fit0_se
Ub_fit0 <- fit0_fit + qnorm(0.975) * fit0_se
```

```
medidas0 <- as.data.frame(cbind(fit0$deviance,fit0$aic, BIC(fit0),
                                logLik(fit0)[1]))
colnames(medidas0) <- c("Deviance","AIC","BIC","Log Likelihood")
medidas0
# fit1: somente nivel_fosfatase_acida

fit1 <- glm(envolvimento_nodal ~ nivel_fosfatase_acida,
            family=binomial(link=logit),
            data=df)

plot(jitter(envolvimento_nodal,0.01) ~ nivel_fosfatase_acida,
xlab="nivel_fosfatase_acida",
     ylab="envolvimento_nodal",
     data=df, pch=16)

curve(predict(fit1, data.frame(nivel_fosfatase_acida=x),type="resp"),
      add=T, col="blue", lwd=2)

summary(fit1); confint(fit1)

hoslem_fit1 <- ResourceSelection::hoslem.test(fit1$y, fitted(fit1), g = 10)

# DescTools::HosmerLemeshowTest(fitted(fit1), fit1$y)
# H_0) Valores observados e valores esperados são iguais para diferentes níveis de
nivel_fosfatase_acida
# H_1) c.c.
# O teste de H-L rejeita a hipótese nula, portanto indica que o modelo não é adequado.


# Analisando a influência de cada valor no resultado da regressão logística
stats::influence.measures(fit1)

# teste verossimilhanca
t1 <- lr.test(fit0,fit1)
# O teste de razão de verossimilhança não rejeita (por pouco) a hipótese nula, portanto
o modelo fit0 é preferível

# teste de wald

thetahat <- fit1$coefficients
vcov1 <- vcov(fit1)
LL <- rbind(c(0,1))

WaldTest = function(L,thetahat,Vn,h=0) {
  WaldTest = numeric(3)
  names(WaldTest) = c("W","df","p-value")
  r = dim(L)[1]
  W = t(L%*%thetahat-h) %*% solve(L%*%Vn%*%t(L)) %*%
    (L%*%thetahat-h)
  W = as.numeric(W)
  pval = 1-pchisq(W,r)
  WaldTest[1] = W; WaldTest[2] = r; WaldTest[3] = pval
  WaldTest}


# teste score
score1<-anova(fit0,fit1, test="Rao")
resultados <- cbind(t1$LR,1,t1$pvalue)
resultados2 <- matrix(WaldTest(LL,thetahat,vcov1), ncol=3)
resultados3 <- cbind(anova(fit0,fit1, test="Rao")[2,4],
                     anova(fit0,fit1, test="Rao")[2,3],
                     anova(fit0,fit1, test="Rao")[2,6])

testes1 <- rbind(resultados,resultados2,resultados3)
rownames(testes1) <- c("Razao de Verossimilhanca","Wald","Score")
colnames(testes1) <- c("Estatistica","GL","P-valor")
```

```
testes1
# Vemos que nenhum dos 3 testes rejeitaram a hipótese nula a 5%. Isto é, o modelo não é
significativamente melhor que o modelo que contém apenas o intercepto.

coef1 <- summary(fit1)$coefficients

colnames(coef1) <- c("Estimativa", "Erro Padrao", "Valor Z", "Pr(>|z|)")
rownames(coef1) <- c("Intercepto", "Nível de \n fosfatase ácida")

coef1

odds.ratio(fit1)[2,]
# A Odds-ratio mostra que para cada 1 (x100) unidades do nível de fosfatase ácida,
aumenta em 1,41% a chance de não haver envolvimento nodal.
# Entretanto, o intervalo contém o valor 1, o que indica que é pouco significante este
resultado, e pode não dizer nada na realidade.

# Medidas de qualidade de ajuste

medidas1 <- as.data.frame(cbind(fit1$deviance,fit1$aic, BIC(fit1),
                                logLik(fit1)[1]))
colnames(medidas1) <- c("Deviance","AIC","BIC","Log Likelihood")
medidas1

# Intervalo de confiança para a odds ratio
fit1_ci <- get_confint(fit      = fit1,
                       variavel = "nivel_fosfatase_acida",
                       digitos  = 5)

# Plotando com I.C. 95%
temp.data <- data.frame(df$nivel_fosfatase_acida)
colnames(temp.data) = "nivel_fosfatase_acida"
predicted.data <- as.data.frame(predict(fit1, newdata = temp.data,
                                        type="link", se=TRUE))

new.data <- cbind(temp.data, predicted.data)

new.data$yci <- fit1$family$linkinv(new.data$fit - qnorm(0.975) * new.data$se)
new.data$ycs <- fit1$family$linkinv(new.data$fit + qnorm(0.975) * new.data$se)
new.data$fit <- fit1$family$linkinv(new.data$fit)

p <- ggplot(df, aes(x=nivel_fosfatase_acida, y=envolvimento_nodal))
p + geom_point(na.rm = T) +
  geom_ribbon(data=new.data, aes(y=fit, ymin=yci, ymax=ycs),
              fill="lightblue", alpha=0.3) +
  geom_line(data=new.data, aes(y=fit)) +
  labs(x="Nível fosfatase ácida (x100)", y="Probabilidade de envolvimento nodal")

#####################################################################
# Parte 3) Adicionando outras variáveis no modelo ----

# Modelo saturado:
fit2 <- glm(envolvimento_nodal ~ resultado_radiografia +
               + estagio_tumor +
              nivel_fosfatase_acida,
            family=binomial(link=logit),
            data=df)

summary(fit2); confint(fit2)
# Com o modelo saturado, todas as variáveis se tornam significativas a 5%, inclusive o
nível de fosfatase ácida.

hoslem_fit2 <- ResourceSelection::hoslem.test(fit2$y,fit2$fitted.values)
# H_0) Valores observados e valores esperados são iguais para diferentes níveis de
nivel_fosfatase_acida e outras variáveis.
# H_1) c.c.
# O teste de H-L rejeita a hipótese nula a 5%, portanto indica que o modelo não é
adequado.
```

```
# Outra opção de teste
# DescTools::HosmerLemeshowTest(fit = fitted(fit2),
#                               obs =  df$envolvimento_nodal)


# Analisando a influência de cada valor no resultado da regressão logística (análise de
resíduos)
stats::influence.measures(fit2)

# teste verossimilhanca
t2 <- lr.test(fit0,fit2)
# O teste de razão de verossimilhança rejeita a hipótese nula, portanto o modelo
saturado é preferível

# teste score
score2<-anova(fit0,fit2, test="Rao")
resultados <- cbind(t2$LR,3,t2$pvalue)
resultados3 <- cbind(anova(fit0,fit2, test="Rao")[2,4],
                     anova(fit0,fit2, test="Rao")[2,3],
                     anova(fit0,fit2, test="Rao")[2,6])

testes2 <- rbind(resultados,resultados3)
rownames(testes2) <- c("Razao de Verossimilhanca","Score")
colnames(testes2) <- c("Estatistica","GL","P-valor")

testes2
# Vemos que ambos os testes rejeitaram a hipótese nula a 5%.
# Isto é, o modelo é significativamente melhor que o modelo que contém apenas o
intercepto.

coef2 <- summary(fit2)$coefficients

colnames(coef2) <- c("Estimativa","Erro Padrao","Valor Z","Pr(>|z|)")
rownames(coef2) <- c("Intercepto",
                     "Resultado da Radiografia",
                     "Estágio do Tumor",
                     "Nível de Fosfatase Ácida"
                     )
coef2

odds.ratio(fit2)[2,]
# A odds ratio mostra que o resultado positivo na radiografia aumenta de 4,246 a 82,785
a chance de haver envolvimento nodal.


odds.ratio(fit2)[3,]
# A odds ratio mostra que o fato do tumor ser mais grave aumenta de 6,654 a 137,73 a
chance de haver envolvimento nodal.


odds.ratio(fit2)[4,]
# A Odds-ratio mostra que para cada 1 (x100) unidades do nível de fosfatase ácida,
aumenta em 2,82% a chance de não haver envolvimento nodal.
# Diferente do modelo simples, o intervalo não contém o valor 1, indicano que é
positiva a relação dessas variáveis a >95% de significância


fit2_odds <-
odds.ratio(fit2) |>
  as_tibble() |>
  slice(-1) |>
  mutate(Variável = c("Resultado da Radiografia",
                      "Estágio do Tumor",
                      "Nível de Fosfatase Ácida")) |>
  dplyr::select(Variável, everything()) |>
  rename("Estimativa Pontual" = OR, "P-valor" = p)
```

```
# Essas duas variáveis são extremamente significativas, e contribuem fortemente para a
explicação de haver ou não envolvimento nodal.

# Medidas de qualidade de ajuste
medidas2 <- as.data.frame(cbind(fit2$deviance,fit2$aic, BIC(fit2),
                                logLik(fit2)[1]))
colnames(medidas2) <- c("Deviance","AIC","BIC","Log Likelihood")
medidas2

#####################################################################
# Outros modelos possíveis:
fit3 <- glm(envolvimento_nodal ~ resultado_radiografia + estagio_tumor,
            family=binomial(link=logit),
            data=df)

medidas3 <- as.data.frame(cbind(fit3$deviance,fit3$aic, BIC(fit3),
                                logLik(fit3)[1]))
colnames(medidas3) <- c("Deviance","AIC","BIC","Log Likelihood")

fit4 <- glm(envolvimento_nodal ~ resultado_radiografia + nivel_fosfatase_acida,
            family=binomial(link=logit),
            data=df)

medidas4 <- as.data.frame(cbind(fit4$deviance,fit4$aic, BIC(fit4),
                                logLik(fit4)[1]))
colnames(medidas4) <- c("Deviance","AIC","BIC","Log Likelihood")

fit5 <- glm(envolvimento_nodal ~ nivel_fosfatase_acida + estagio_tumor,
            family=binomial(link=logit),
            data=df)

medidas5 <- as.data.frame(cbind(fit5$deviance,fit5$aic, BIC(fit5),
                                logLik(fit5)[1]))
colnames(medidas5) <- c("Deviance","AIC","BIC","Log Likelihood")

fit6 <- glm(envolvimento_nodal ~ estagio_tumor,
            family=binomial(link=logit),
            data=df)

medidas6 <- as.data.frame(cbind(fit6$deviance,fit6$aic, BIC(fit6),
                                logLik(fit6)[1]))
colnames(medidas6) <- c("Deviance","AIC","BIC","Log Likelihood")

fit7 <- glm(envolvimento_nodal ~ resultado_radiografia,
            family=binomial(link=logit),
            data=df)

medidas7 <- as.data.frame(cbind(fit7$deviance,fit7$aic, BIC(fit7),
                                logLik(fit7)[1]))
colnames(medidas7) <- c("Deviance","AIC","BIC","Log Likelihood")

#####################################################################
# Comparando os modelos

fit = glm(
  envolvimento_nodal ~ nivel_fosfatase_acida + resultado_radiografia + estagio_tumor,
  family = binomial(link = logit),
  data = df
)

car::Anova(fit) # A iteração não é significativa.
stepAIC(fit, direction = c("both")) # Voltamos ao modelo saturado.
rm(fit)

stepwise <- stepAIC(fit2, direction = c("both"))
# O método stepwise indica ficar com o modelo saturado.
```

```
Modelo = c("X4 ~ 1","X4 ~ X3","X4 ~ X1 + X2 + X3","X4 ~ X1 + X2","X4 ~ X1 + X3","X4 ~
X2 + X3","X4 ~ X2","X4 ~ X1")
medidas =
rbind(medidas0,medidas1,medidas2,medidas3,medidas4,medidas5,medidas6,medidas7)
medidas$Modelo = Modelo
medidas = medidas[c(1,3,4:6,8,7,2),c(5,1:4)]

kable(medidas)
# Pela tabela, devemos optar pelo modelo saturado, pois:
# 1) tem menor deviance 2) tem menor AIC 3) tem maior BIC 4) tem maior log-
verossimilhança.

# rm(fit0,fit1,fit3,fit4,fit5,fit6,fit7,
#    medidas0,medidas1,medidas3,medidas4,medidas5,medidas6,medidas7,
#    resultados,resultados2,resultados3,p,predicted.data,vcov1,Modelo,thetahat,
#    WaldTest,LL,new.data,score1,t1,testes1,temp.data,coef1,t2)
##################################################################
# Métricas do modelo escolhido ----
kable(medidas2)

kable(coef2)
# Curva ROC do modelo fit2
ROC <- roc(response = df$envolvimento_nodal, predictor = predict(fit2, type =
"response"))

roc_data <- data.frame(
  Spec_comp = 1 - ROC$specificities,
  Sensit = ROC$sensitivities
)



plot(ROC, main = "Curva ROC - Modelo completo")

# Alternativo:
ROC2 <- rocit(score=predict(fit2, type = "response"),class=df$envolvimento_nodal)
plot(ROC2)
ksplot(ROC2)

performance_hosmer(fit2, n_bins = 10)

# Aplicando o modelo aos dados de teste:
teste$predict <- predict(fit2, teste,
                         type = "response")

# Usando um threshould = 0.5...
teste = teste %>%
  mutate(env_pred = ifelse(predict > .5,1,0))

# Matriz de confusão:
table_confusao <- table(teste$envolvimento_nodal, teste$env_pred)
kable(table_confusao)


acertos <- sum(diag(table_confusao)) / sum(table_confusao)


matriz_conf_alternativa <- table(teste$envolvimento_nodal, teste$env_pred) |>
  as.data.frame() |>
  pivot_wider(values_from = Freq,
              names_from = Var2) |>
  rename("Observado\\Predito" = Var1)

# Para os dados de validação, o modelo teve 54+28/102 = 80,4% de acerto, o que é
bastante razoável.

# Análise de resíduos
plot(residuals(fit2, "pearson"))
```

```
# Outra visualização: Binned residual plot
binnedplot(fitted(fit2),
           residuals(fit2, type = "response"),
           nclass = NULL,
           xlab = "Valores esperados",
           ylab = "Resíduo médio",
           main = "Gráfico residual armazenado",
           cex.pts = 0.8,
           col.pts = 1,
           col.int = "gray")

#Deveríamos observar 95% das observações sobre o intervalo produzido pelo ASE (linhas
cinzas)

y <- rbinom(102, 1, 0.5)
f1 <- fitted(fit2)
plot( residuals(fit2, "pearson"), (y-f1)/sqrt(f1*(1-f1)))
abline(0,1)

# Worm plot (está se popularizando para análise de MLGs)
p_load(gamlss)
wp(resid=resid(fit2),  xvar=df$envolvimento_nodal)

# Pelo worm plot, o ajuste do modelo parece muito ruim.
# Mas isso aparenta ter mais a ver com a técnica (regressão logistica) do que quanto ao
modelo
# específico ajustado.
```