Códigos utilizados no laboratório 6

Camille Menezes e Michel Miler

Maio de 2023

```
library(GGally)
library(tidyverse)
library(patchwork)
library(lmtest)
library(MASS)
library(broom)
library(lindia)
library(dplyr)
library(summarytools)
library(gtsummary)
library(rigr)
library(olsrr)
library(gridExtra)

dados <-  read.csv('Dados2.csv', sep=';', dec=",")

dados <- dados %>%
  mutate(height = height/39.3701,
         weight = weight/2.20462,
         waist = waist/39.3701,
         hip= hip/39.3701)

#RCQ = Relação Cintura Quadril

dados <- dados %>%
  mutate(IMC = weight/(height^2),
         RCQ = waist/hip)

dados <- dados %>%
  mutate(chol = NULL,
         hdl = NULL,
         bp.2s = NULL,
         bp.2d = NULL,
         height = NULL,
         weight = NULL,
         waist = NULL,
         hip = NULL,
         location = NULL,
         frame = NULL,
         gender = NULL,
         time.ppn = NULL
  )
```

```r
nrow(dados)

dados <- dados %>%
  na.omit()

nrow(dados)

# analise descritiva

ggpairs(dados)

summary(dados)

Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

apply(dados, 2, Mode)

boxp <- function(y,name){
  ggplot(dados)+
    geom_boxplot(aes(y=dados[,y]))+
    scale_x_discrete(breaks = NULL) +
    labs(y=name)+
    theme_minimal()
}

boxp(1, "Glicose estabilizada") / boxp(2, "Razão col. total e col. bom") |
  boxp(3,"Hemoglobina glicada") / boxp(4, "Idade") | boxp(5, "Pressão sanguínea sistólica") /
  boxp(6, "Pressão sanguínea diastólica") | boxp(7,"IMC") / boxp(8,"RCQ")

disp <- function(x,name){
  ggplot(dados)+
    aes(y=IMC,x=dados[,x])+
    geom_point()+
    labs(x=name)+
    ggpubr::stat_cor(method = "pearson",
                     p.accuracy = 0.001,
                     r.accuracy = 0.01,
                     label.sep = "  ",
                     col = "blue")+
    theme_minimal()
}

disp(1, "Glicose estabilizada") + disp(2, "Razão col. total e col. bom")+
  disp(3,"Hemoglobina glicada") + disp(4, "Idade") + disp(5, "Pressão sanguínea sistólica") +
  disp(6, "Pressão sanguínea diastólica") + disp(8,"RCQ")


# modelo com todas as covariaveis
model <- lm(IMC~stab.glu+ratio+glyhb+age+bp.1s+bp.1d+RCQ, data=dados)
summary(model)
tab <- augment(model)
```

```r
a <- ggplot(model)+
  geom_point(aes(x=.fitted,y=.resid))+
  labs(x="Valores preditos \n A",y="Resíduos")+
  theme_minimal()

b <- ggplot(model, aes(sample=studres(model)))+
  geom_qq()+
  geom_qq_line(color="red")+
  labs(x="Quantil teórico \n B", y="Resíduos studentizados")+
  theme_minimal()

c <- ggplot(model)+
  geom_histogram(aes(x=.resid),binwidth=2.2,color="black", fill="white")+
  labs(y="Frequência",x="Resíduos \n C")+
  theme_minimal()

a | b | c

shapiro.test(model$residuals)
gqtest(model)
dwtest(model)

# Modelo reduzido
model2<-lm(IMC~ratio+bp.1d+age,data=dados)
summary(model2)
tab <- augment(model2)
lrtest(model2, model)

k = anova(model)
SqRegc = sum(k$`Sum Sq`[-8])
SqResc = k$`Sum Sq`[8]
k=anova(model2)
SqRegr = sum(k$`Sum Sq`[-4])

Fr = (SqRegc - SqRegr)*(nrow(dados)-8)/(SqResc*4)

1-pf(Fr,4,nrow(dados)-8)

model2 %>% tbl_regression(
  pvalue_fun =  ~ style_pvalue ( .x , digits =  3)) %>%
  modify_header(update = list(
    label  ~  "**Variáveis**",
    estimate ~ "**Estimativa**",
    p.value  ~  "**Valor-p**",
    ci ~ "**I.C. 95%**"))  %>%
  bold_p(t = 0.05)

MyAnova <- function(modelo, nd){
  m1 <- modelo
  np <- dim(anova(m1))[1]
  SQReg <- round(sum(anova(m1)$"Sum Sq"[1:(np-1)]), nd)
  glReg <- sum(anova(m1)$"Df"[1:(np-1)])
  SQRes <- round(anova(m1)$"Sum Sq"[np], nd)
```

```
  glRes <- anova(m1)$"Df"[np]
  SQTotal <- round(SQReg + SQRes, nd)
  glTotal <- glReg + glRes
  QMReg <- round(SQReg/glReg, nd)
  QMRes <- round(SQRes/glRes, nd)
  MyF <- round(QMReg/QMRes, nd)
  vpF <- ifelse(pf(MyF, glReg, glRes, lower.tail = F) < 0.0001, "<0.001",
                roud(pf(MyF, glReg, glRes, lower.tail = F), nd))
  ncolunas <- c("Fonte de Variação", "SQ", "gl", "F", "valor p")
  Tanova <- data.frame(FV = c("Regressão",
                              "Resíduos",
                              "Total"),
                       gl = c(glReg, glRes, glTotal),
                       SQ = c(SQReg, SQRes, SQTotal),
                       QM = c(QMReg, QMRes, " "),
                       Est.F = c(MyF, " ", " "),
                       valor.p = c(vpF, " ", " ")
  )
  Tanova
}

MyAnova(model2,2)


#Analise dos residuos
shapiro.test(model2$residuals)
gqtest(model2)
dwtest(model2)

m_infl <- influence.measures(model2)
metricas <- as.data.frame(m_infl$infmat)
names(metricas)

# Valores Ajustados e Resíduos Studentizado
a <- ggplot(model2)+
  geom_point(aes(x=.fitted,y=studres(model2)))+
  labs(x="Valores preditos \n A",y="Resíduos studentizados")+
  theme_bw()

# Gráco Quantil-Quantil
b <- ggplot(model2, aes(sample=studres(model2)))+
  geom_qq()+
  geom_qq_line(color="red")+
  labs(x="Quantil teórico \n B", y="Resíduos studentizados")+
  theme_bw()

c <- ggplot(model2)+
  geom_histogram(aes(x=.resid),binwidth=2.2,color="black", fill="white")+
  labs(y="Frequência",x="Resíduos \n C")+
  theme_bw()

a | b | c
```

```r
# Gráfico de Distância de Cook
k <- ols_prep_cdplot_data(model2)
d <- ols_prep_outlier_obs(k)
f <- ols_prep_cdplot_outliers(k)
c <- ggplot(d, aes(x = obs, y = cd, label = txt)) +
  geom_bar(width = 0.5, stat = "identity",
           aes(fill = fct_color)) + scale_fill_manual(values = c("black", "red")) +
  labs(fill = "Observação") + ylim(0, k$maxx) +
  ylab("Distância de Cook") + xlab("Observação")+
  geom_hline(yintercept = 0) +
  geom_hline(yintercept = k$ts, colour = "red") +
  geom_text(hjust = -0.2, nudge_x = 0.05, size = 2, na.rm = TRUE) +
  theme_bw()+
  annotate("text", x = Inf, y = Inf, hjust = 1.2, vjust = 2, family = "serif",
           fontface = "italic", colour = "darkred", label = paste("Limite", round(k$ts, 3)))


# Gráfico dos pontos de Alavanca e Resíduo Studentizado
g <- ols_prep_rstudlev_data(model2)
d <- g$levrstud
d$txt <- ifelse(d$color == "normal", NA, d$obs)
f <- d[d$color == "outlier", c("obs", "leverage", "rstudent")]
colnames(f) <- c("observation", "leverage", "stud_resid")

d <- ggplot(d, aes(leverage, rstudent, label = txt)) +
  geom_point(aes(colour = fct_color)) +
  scale_color_manual(labels = c("normal","ponto de alavanca","outlier",
                                "outlier e ponto de alavanca"),
                                values = c("black","blue", "red", "green")) +
  xlim(g$minx, g$maxx) + ylim(g$miny, g$maxy) +
  labs(colour = "Observação", x = "Pontos de alavanca", y = "Resíduo studentizado") +
  geom_hline(yintercept = c(2,-2), colour = "black") +
  geom_vline(xintercept = g$lev_thrsh, colour = "black") +
  geom_text(vjust = -1, size = 3, family = "serif", fontface = "italic", colour = "black") +
  theme_bw()+
  annotate("text", x = Inf, y = Inf, hjust = 1.2, vjust = 2,
           family = "serif", fontface = "italic", colour = "darkred",
           label = paste("Limite:", round(g$lev_thrsh, 3)))

c | d


# Gráco de DfBeta
ols_dfbetas <- function (model, print_plot = TRUE)
{  obs <- NULL
  txt <- NULL
  dfb <- dfbetas(model)
  n <- nrow(dfb)
  np <- ncol(dfb)
  threshold <- 2/sqrt(n)
  myplots <- list()
  outliers <- list()
  colnames(dfb) <- c("Intercepto","Razão entre colesterois","Pressão diastólica","Idade")
  for (i in seq_len(np)) {
```

```
    dbetas <- dfb[, i]
    df_data <- data.frame(obs = seq_len(n), dbetas = dbetas)
    d <- ols_prep_dfbeta_data(df_data, threshold)
    f <- ols_prep_dfbeta_outliers(d)
    p <- ggplot(d, aes(x = obs, y = dbetas,
                                        label = txt, ymin = 0, ymax = dbetas)) +
                            geom_linerange(colour = "black") +
                            geom_hline(yintercept = c(0, threshold, -threshold),
                                       colour = "red") +
                            ylab("") + xlab("") +
                            ggtitle(paste(colnames(dfb)[i])) +
                            theme_bw()+
                            geom_text(hjust = -0.2, nudge_x = 0.15, size = 2,
                                      family = "serif", fontface = "italic", colour = "darkred",
                                      na.rm = TRUE)
    myplots[[i]] <- p
    outliers[[i]] <- f
  }
  if (print_plot) {
    marrangeGrob(myplots, nrow = 2, ncol = 2, top = quote(paste("")),
                 left="DFBETAS",bottom="Observações")
  }
}

ols_dfbetas(model2)


# Gráfico de DfFit
dbetas <- NULL
obs <- NULL
txt <- NULL
dffitsm <- unlist(dffits(model2))
k <- length(coef(model2))
n <- nrow(dados)
dffits_t <- sqrt(k/n) * 2
title <- names(model.frame(model2))[1]
dfits_data <- data.frame(obs = seq_len(n), dbetas = dffitsm)
d <- ols_prep_dfbeta_data(dfits_data, dffits_t)
f <- ols_prep_dfbeta_outliers(d)
e <- ggplot(d, aes(x = obs, y = dbetas, label = txt, ymin = 0,
                   ymax = dffitsm)) + geom_linerange(colour = "black") +
  geom_hline(yintercept = c(0, dffits_t, -dffits_t), colour = "red") +
  xlab("Observação") +
  theme_minimal()+
  ylab("DFFITS")
  + geom_text(hjust = -0.2, nudge_x = 0.15, size = 3,
              family = "serif", fontface = "italic", colour = "darkred",
              na.rm = TRUE) + annotate("text", x = Inf, y = Inf, hjust = 1.5,
              vjust = 2, family = "serif", fontface = "italic", colour = "darkred",
              label = paste("Limite:", round(dffits_t, 2)))


# Gráfico do COVRatio
fct_color <- NULL
```

```r
color <- NULL
pred <- NULL
dsr <- NULL
txt <- NULL
obs <- NULL
ds <- NULL
k <- ols_prep_dsrvf_data(model2)
d <- k$ds
d$txt <- ifelse(d$color == "outlier", d$obs, NA)
f <- d[color == "outlier", c("obs", "pred", "dsr")]
colnames(f) <- c("Observação", "fitted_values", "del_stud_resid")
h <- ggplot(d, aes(x = pred, y = dsr, label = txt)) +
  geom_point(aes(colour = fct_color)) +
  scale_color_manual(values = c("black", "red")) +
  ylim(k$cminx, k$cmaxx) + xlab("Valores preditos") +
  ylab("COVARATIO") +
  labs(color = "Observação") +
  theme_minimal()+
  geom_hline(yintercept = c(-2, 2), colour = "red") +
  geom_text(hjust = -0.2,  nudge_x = 0.15, size = 3,
            family = "serif", fontface = "italic", colour = "darkred", na.rm = TRUE) +
  annotate("text",
           x = Inf, y = Inf, hjust = 1.5, vjust = 2, family = "serif",
           fontface = "italic", colour = "darkred", label = paste0("Limite: abs(",2, ")"))

e | h
```