Exercício Geral

Cynthia Tojeiro

2025-02-17

1. **Um breve comentário sobre a diabetes:** A diabetes é uma disfunção do metabolismo, ou seja, o jeito com que o organismo faz a digestão dos alimentos para a produção de energia. A maioria dos alimentos que se ingerem são quebrados em pequenas partículas de glicose, um tipo de açúcar encontrado no sangue, que após a digestão passa para a corrente sanguínea. No entanto, para que a glicose possa adentrar nas células, ela precisa da ajuda da insulina. A insulina é um hormônio produzido no pâncreas, uma glândula localizada por trás do estômago. Quando nos alimentamos, o pâncreas produz automaticamente a quantidade certa de insulina necessária para mover a glicose do sangue para as células do corpo. Em pessoas com diabetes, o pâncreas produz pouca insulina, então as células não respondem de forma esperada à insulina produzida. Assim, a glicose fica no sangue aumentando o que se chama de glicemia (concentração de glicose), ou vai direto para a urina (não sendo aproveitada pelas células).

2. **Dados:**

   O Instituto Nacional de Diabetes e Doenças Digestivas e Renais conduziu um estudo com 768 índias Pima adultas que viviam perto de Phoenix - Arizona. Os dados foram coletados segundo critérios da Organização Mundial da Saúde, em que, as variáveis utilizadas foram as seguintes:

   - Partos: número de vezes que as índias ficaram grávidas; (variável explicativa: contínua)
   - Glicose: concentração plasmática de glicose a 2 horas em um teste oral de tolerância a glicose; (variável resposta: contínua)
   - Diastólica: Pressão arterial diastólica (mm/Hg); ; (variável explicativa: contínua)
   - Triceps: Espessura cutânea tricipal; ; (variável explicativa: contínua)
   - IMC: Índice de massa corporal [peso em kg/(altura em m^2)] (variável explicativa: contínua)
   - Diabetes: Diabetes função da genealogia; (variável explicativa: contínua)
   - Idade: idade em anos; (variável explicativa: contínua)
   - Teste: Teste de sinais de diabetes nos pacientes (0 se negativo, 1 se positivo); (variável explicativa: categórica)

```
 pacotes necessários
library(readr)
library(car)
library(tidyverse)
library(robustbase) #Boxplot robusto das variaveis
library(dplyr)       # Manipulação de dados
library(ggplot2)     # Visualização de dados
library(MASS)
library(alr4)
library(xtable)
library(ggcorrplot)
library(lmtest)
```

```r
# Carregando o dataset
diabetes <- read.csv("C:\\datascience\\diabetes1.csv")
attach(diabetes)

# Inspecionando os dados
summary(diabetes)  # Estatísticas descritivas das variáveis
str(diabetes)      # Estrutura do dataset

Outcome = factor(Outcome)
levels(Outcome) <- c("Não Diabético", "Diabético")

par(mfrow=c(1,1))
adjbox(diabetes$Pregnancies, main = "Partos")
adjbox(diabetes$Glucose, main = "Glicose")
adjbox(diabetes$BloodPressure, main = "Diastólica")
adjbox(diabetes$SkinThickness, main = "Triceps")
adjbox(diabetes$Insulin, main = "Insulina")
adjbox(diabetes$BMI, main = "IMC")
adjbox(diabetes$DiabetesPedigreeFunction, main = "Diabetes")
adjbox(diabetes$Age, main = "Idade")
```

Tratamento de valores ausentes (substituindo zeros por médias nas variáveis contínuas)

```r
diabetes_clean <- diabetes %>%
  mutate(across(c(Glucose, BloodPressure,
SkinThickness, Insulin, BMI), ~
ifelse(. == 0, mean(., na.rm = TRUE), .)))

# Confirmando o tratamento
summary(diabetes_clean)
```

Análise Exploratória: Distribuição da variável dependente (Glucose)

```r
ggplot(diabetes_clean, aes(x = Glucose)) +
  geom_bar(fill = "skyblue") +
  labs(x = "Glucose", y = "Frequência", title = "Distribuição de taxas de glicose") +
  theme_minimal()

#Gráficos de dispersão
par(mfrow = c(3,3))

plot(Glucose~Pregnancies, data = diabetes_clean, xlab = 'partos', ylab = 'glicose')
abline(lm(Glucose~Pregnancies, data = diabetes_clean), col=2, lwd = 2)

ggplot(diabetes_clean) +
  geom_point(aes(x = Pregnancies, y = Glucose, color = factor(Outcome)),
             size = 3) +
  scale_color_manual("Diabetes",
                     values = c("red", "blue"),
```

```
                    labels = c("Não", "Sim")) +
  labs(title = 'Relação entre Glicose, partos e diabéticos',
       y = 'Glicose',
       x = 'Partos')



plot(Glucose~BloodPressure, data = diabetes_clean, xlab = 'diastolica', ylab = 'glicose')
abline(lm(Glucose~BloodPressure, data = diabetes_clean), col=2, lwd = 2)

plot(Glucose~SkinThickness, data = diabetes_clean, xlab = 'triceps', ylab = 'glicose')
abline(lm(Glucose~SkinThickness, data = diabetes_clean), col=2, lwd = 2)

plot(Glucose~Insulin, data = diabetes_clean, xlab = 'insulina', ylab = 'glicose')
abline(lm(Glucose~Insulin, data = diabetes_clean), col=2, lwd = 2)


plot(Glucose~BMI, data = diabetes_clean, xlab = 'imc', ylab = 'glicose')
abline(lm(Glucose~BMI, data = diabetes_clean), col=2, lwd = 2)

plot(Glucose~DiabetesPedigreeFunction, data = diabetes_clean, xlab = 'diabetes', ylab = 'glicose')
abline(lm(Glucose~DiabetesPedigreeFunction, data = diabetes_clean), col=2, lwd = 2)

plot(Glucose~Age, data = diabetes_clean, xlab = 'idade', ylab = 'glicose')
abline(lm(Glucose~Age, data = diabetes_clean), col=2, lwd = 2)

Boxplot(diabetes_clean$Glucose,Outcome,id=FALSE)
```

```
#Matriz de Correlações
cor_matrix <- cor(diabetes_clean, method = "pearson")
corrplot(cor(dados[,-9],method = "pearson"), method = "number")
```

diabetes=diabetes_clean # Regressão linear múltipla

```
#Ajuste do modelo com todas as variáveis
fit.model<-ajuste1<-lm(Glucose~., data=diabetes)
summary(fit.model)

#Verificando multicolinearidade
vif(ajuste1)

#Análise de Resíduos
source("C:\\datascience\\Programas\\Diag2.norm.r")
source("C:\\datascience\\Programas\\Envel_norm.r")
source("C:\\datascience\\Programas\\anainflu_norm.r")

par(mfrow=c(1,1))
envelnorm(fit.model)
diag2norm(fit.model)
anainflu_norm(fit.model)

#Testes para normalidade e homocedasticidade
shapiro.test(fit.model$residuals)
bptest(fit.model)
```

```r
#Seleção de covariáveis

step(ajuste1, direction = "both")

fit.model<-ajuste2<-lm(Glucose~Pregnancies+BloodPressure+
                         Insulin+Age+Outcome, data=diabetes)
summary(ajuste2)
```

Teste F para comparar a qualidade dos modelos com e sem a variável DiabetesPedigreeFunction.

```r
ajuste2 <- update(ajuste1,~Pregnancies+BloodPressure+
                    Insulin+Age+Outcome )

summary(ajuste2)

anova(ajuste2, ajuste1)
```

- A comparação dos modelos indica que não existem indícios para rejeitar a hipótese nula de igualdade de qualidade dos modelos.
- Os modelos são semelhantes escolhendo-se, portanto, o modelo mais simples, pelo princípio da parcimónia.

```r
par(mfrow=c(1,1))
envelnorm(fit.model)
diag2norm(fit.model)
anainflu_norm(fit.model)

#Testes para normalidade e homocedasticidade
shapiro.test(fit.model$residuals)
bptest(fit.model)
```

Transformação de variáveis

```r
fit.model<-ajuste3<-lm(log(Glucose)~Pregnancies+
                       BloodPressure+Insulin+Age+
                         Outcome,data=diabetes_clean)

summary(ajuste3)

#Análise de Resíduos
par(mfrow=c(1,1))
envelnorm(fit.model)
diag2norm(fit.model)
anainflu_norm(fit.model)

#Testes para normalidade e homocedasticidade
shapiro.test(ajuste3$residuals)
bptest(ajuste3)
ncvTest(ajuste3)
```

```r
#Verificando observações influentes
plot(ajuste3, which =1:4)
plot(ajuste3, which = 4, cook.levels = distance.cook)

diabetes.1 = diabetes_clean[-538,]
fit.model<-ajuste3.1<-lm(log(Glucose)~Pregnancies+BloodPressure+
                           Insulin+Age+Outcome,data=diabetes.1)
#Verificando se realmente a observação
#538 não é influente para podermos
#retirá-la do modelo.

require(stargazer)
stargazer(ajuste3, ajuste3.1, type = "text")

shapiro.test(ajuste3.1$residuals)
bptest(ajuste3.1)

diabetes.2 = diabetes_clean[-c(538,14),]
fit.model<-ajuste4<-lm(log(Glucose)~Pregnancies+
                         BloodPressure+Insulin+Age+Outcome,
                       data=diabetes.2)

sumres<-summary(ajuste4)
m.X<-(model.matrix(ajuste4))
v.beta <- ajuste4$coefficients
ep.beta<- sqrt(diag(vcov(ajuste4)))
quantilt<-qt(0.975,df=768-9)
xtable(cbind(sumres$coefficients[,1:3],v.beta-quantilt*ep.beta,
v.beta+quantilt*ep.beta,sumres$coefficients[,4]),digits=4)


stargazer(ajuste3, ajuste4, type = "text")

shapiro.test(ajuste4$residuals)
bptest(ajuste4)
#Verificando se realmente as observações
#580 e 538 não são influentes para podermos
#retirá-las do modelo.

require(stargazer)
stargazer(ajuste3, ajuste4, type = "text")
```

	Estimativa	Erro padrão	valor t	LI(IC)	LS(IC)	p-valor
(Intercept)	4.3297	0.0512	84.5923	4.2292	4.4302	0.0000
Pregnancies	-0.0049	0.0026	-1.9000	-0.0100	0.0002	0.0578
BloodPressure	0.0023	0.0007	3.5010	0.0010	0.0036	0.0005
SkinThickness	-0.0010	0.0009	-1.1003	-0.0029	0.0008	0.2716
Insulin	0.0008	0.0001	10.4023	0.0007	0.0010	0.0000
BMI	0.0012	0.0013	0.9251	-0.0014	0.0039	0.3552
Age	0.0034	0.0008	4.3733	0.0019	0.0049	0.0000
Outcome	0.1963	0.0166	11.8416	0.1637	0.2288	0.0000

Alternativas para estimação em modelos com heterocedasticidade:

1) Modelagem dupla

```
require(dglm)
fit.model=ajuste5 = dglm(log(Glucose)~Pregnancies+BloodPressure+
                    Insulin+Age+Outcome,Pregnancies+BloodPressure+
                    Insulin+Age+Outcome)
summary(ajuste5)
```

2) Minimos quadrados ponderados e obter os erros padrões robusto.

```
library(sandwich)
ajuste6=coeftest(ajuste3, vcov = vcovHC(ajuste3, type="HC0"))
```

3) Mínimos Quadrados Generalizados (MQGF)

O estimador de Mínimos Quadrados Generalizados Factíveis (MQGF) tem por sua vez a função de realizar o ajuste da variância através de uma matriz de ponderação sobre a heterocedasticidade:

```
resid2 = (ajuste3$residuals)^2
r <- log(resid2)
varreg <- lm(r ~Pregnancies+BloodPressure+Insulin+Age+
               Outcome,data=diabetes_clean)

#ponderação

w <- 1/exp(fitted(varreg))

#MQP

mqgf <- lm(ajuste3, weights = w, data =diabetes_clean )
```