###########################################################################

### IBI5086 2023 Introdução aos Métodos Estatísticos para a Bioinformática ###

### Profa. Júlia Maria Pavan Soler ###

### Análise Descritiva Multivariada ###

###########################################################################

##Considere os seguintes dados:

##Caracterização nutricional de 27 produtos alimentícios (Everitt, 2007)

energia<- c(340,245,420,375,180,115,170,160,265,300,340,340,355,205,185,135,70,45,90,135,200,155,195,120,180,170,110)

proteina<-c(20,21,15,19,22,20,25,26,20,18,20,19,19,18,23,22,11,7,14,16,19,16,16,17,22,25,23)

gordura<-c(28,17,39,32,10,3,7,5,20,25,28,29,30,14,9,4,1,1,2,5,13,9,11,5,9,7,1)

calcio<-c(9,9,7,9,17,8,12,14,9,9,9,9,9,7,9,25,82,74,38,15,5,157,14,159,367,7,98)

ferro<-c(2.6,2.7,2,2.5,3.7,1.4,1.5,5.9,2.6,2.3,2.5,2.5,2.4,2.5,2.7,0.6,6,5.4,0.8,0.5,1,1.8,1.3,0.7,2.5,1.2,2.6)

dat<-cbind(energia,proteina,gordura,calcio,ferro)

dat

##Estatísticas descritivas multivariadas

##Dados originais

mi<-colMeans(dat) #centróide dos dados

mi

s<-cov(dat) #matriz de covariância

round(s,2)

r<-cor(dat) # matriz de correlação de Y = matriz de covariância de Y\*

round(r,2)

#Medidas de variabilidade multivariada

vartotal <- sum(diag(s))

vartotal

vargen <- det(s)

vargen

#interpretação deve ser comparativa entre grupos, por ex.

##Padronização das variáveis (subtrai a média e divide pelo desvio padrão)

datz<-scale(dat,center=TRUE,scale=TRUE)

datz

round(cbind(dat,datz),2)

colMeans(datz)

sz<-cov(datz) #cov(datz) = cor(dat)

round(sz,2)

rz<-cor(datz) #cov(datz) = cor(datz)

round(rz,2)

vartotalz <- sum(diag(rz))

vartotalz # =p

vargen <- det(rz)

vargen

##Medidas de Distância

##Distância Euclidiana (ordinária) entre observações

de<-dist(dat)

de<-round(de,2)

de

de <- as.matrix(de)

dim(de)

de[1:6,1:6]

de14<-de[1,4]

de14

dat[1,]

dat[4,]

de36<-de[3,6]

de46

dat[6,]

dat[3,]

##Distância Euclidiana Ponderada (dist. Pearson) entre observações

datp<-scale(dat, center = FALSE, scale = apply(dat, 2, sd, na.rm = TRUE))

dep<-dist(datp)

dep

dep<-as.matrix(dep)

round(dep[1:6,1:6],2)

dep14<-dep[1,4]

dep14

datp[1,]

datp[4,]

dep36<-dep[3,6]

dep36

datp[6,]

datp[3,]

## Distância de Mahalanobis das observações AO CENTRÓIDE

##Critério de observações atípicas

dm<-mahalanobis(dat,mi,s)

dm

round(dm,2)

qchisq(c(0.9), df = 5, lower.tail = TRUE)

dat.o<-cbind(seq(1:27),dm)

plot(dat.o, main="Observações Atípicas")

abline(h=qchisq(c(0.9), df = 5, lower.tail = TRUE))

text(dat.o, labels=c("1","2","3","4","5","6","7","8","9","10","11","12",

"13","14","15","16","17","18","19","20","21","22","23","24","25","26","27"), lwd=3)

##Considerando os dados originais:

##Há algum alimento com composição nutricional atípica?

##Note que a dm é invariante por padronização dos dados

##Boxplot bivariado

##Copiar a função bivbox proposta por Everitt:

par(mfrow=c(2,5))

bivbox(dat[,c(1,2)], method ="O"); title(main="Y1xY2")

bivbox(dat[,c(1,3)], method ="O"); title(main="Y1xY3")

bivbox(dat[,c(1,4)], method ="O"); title(main="Y1xY4")

bivbox(dat[,c(1,5)], method ="O"); title(main="Y1xY5")

bivbox(dat[,c(2,3)], method ="O"); title(main="Y2xY3")

bivbox(dat[,c(2,4)], method ="O"); title(main="Y2xY4")

bivbox(dat[,c(2,5)], method ="O"); title(main="Y2xY5")

bivbox(dat[,c(3,4)], method ="O"); title(main="Y3xY4")

bivbox(dat[,c(3,5)], method ="O"); title(main="Y3xY5")

bivbox(dat[,c(4,5)], method ="O"); title(main="Y4xY5")

##Boxplot bivariado

##Copiar a função bivbox proposta por Everitt:

bivbox<-function(a, d = 7, mtitle = "Bivariate Boxplot",

method = "robust",xlab="X",ylab="Y")

{

#a is data matrix

#d is constant(usually 7)

p <- length(a[1, ])

if(method == "robust") {

param <- biweight(a[, 1:2]); m1 <- param[1]; m2 <- param[2]

s1 <- param[3]; s2 <- param[4]; r <- param[5]

}

else {

m1 <- mean(a[, 1]); m2 <- mean(a[, 2]);

s1 <- sqrt(var(a[, 1])); s2 <- sqrt(var(a[, 2])); r <- cor(a[, 1:2])[1, 2]

}

x <- (a[, 1] - m1)/s1; y <- (a[, 2] - m2)/s2

e <- sqrt((x \* x + y \* y - 2 \* r \* x \* y)/(1 - r \* r))

e2 <- e \* e; em <- median(e); emax <- max(e[e2 < d \* em \* em])

r1 <- em \* sqrt((1 + r)/2); r2 <- em \* sqrt((1 - r)/2); theta <- ((2 \* pi)/360) \* seq(0, 360, 3)

xp <- m1 + (r1 \* cos(theta) + r2 \* sin(theta)) \* s1; yp <- m2 + (r1 \* cos(theta) - r2 \* sin(theta)) \* s2

r1 <- emax \* sqrt((1 + r)/2); r2 <- emax \* sqrt((1 - r)/2); theta <- ((2 \* pi)/360) \* seq(0, 360, 3)

xpp <- m1 + (r1 \* cos(theta) + r2 \* sin(theta)) \* s1; ypp <- m2 + (r1 \* cos(theta) - r2 \* sin(theta)) \* s2

maxxl <- max(xpp); minxl <- min(xpp); maxyl <- max(ypp); minyl <- min(ypp)

b1 <- (r \* s2)/s1; a1 <- m2 - b1 \* m1; y1 <- a1 + b1 \* minxl; y2 <- a1 + b1 \* maxxl

b2 <- (r \* s1)/s2; a2 <- m1 - b2 \* m2; x1 <- a2 + b2 \* minyl; x2 <- a2 + b2 \* maxyl

maxx <- max(c(a[, 1], xp, xpp, x1, x2)); minx <- min(c(a[, 1], xp, xpp, x1, x2))

maxy <- max(c(a[, 2], yp, ypp, y1, y2)); miny <- min(c(a[, 2], yp, ypp, y1, y2))

plot(a[, 1], a[, 2], xlim = c(minx, maxx), ylim = c(miny, maxy), xlab =xlab, ylab =ylab,

lwd = 2, pch = 1)

lines(xp, yp, lwd = 2); lines(xpp, ypp, lty = 2, lwd = 2)

segments(minxl, y1, maxxl, y2, lty = 3, lwd = 2); segments(x1, minyl, x2, maxyl, lty = 4, lwd = 2)

}

## Faces de Chernoff

library(TeachingDemos)

faces( as.matrix(dat), fill=T) ##unidades amostrais

faces( as.matrix(t(dat)), fill=T) ##variáveis

#Exercício

# Gerar dados da Normal Multivariada

#Realizar uma análise descritiva dos dados gerados

#Exemplo: Caso Bivariado

library(MASS)

##set.seed(1298)

mu<-c(0,0)

sigma<-matrix(c(2,1,1,2),ncol=2)

n<-10

yr<-mvrnorm(n,mu,sigma)

yr

mi<-colMeans(yr)

mi

s<-cov(yr)

s

plot(yr)

#Outra alternativa para gerar dados da normal multivariada

library(clusterGeneration)

library(mvtnorm)

## Definir “n” e “p”

mu<-rnorm(p)

R<-rcorrmatrix(p,alphad=1)

yr<-rmvnorm(n,mu,R)

yr