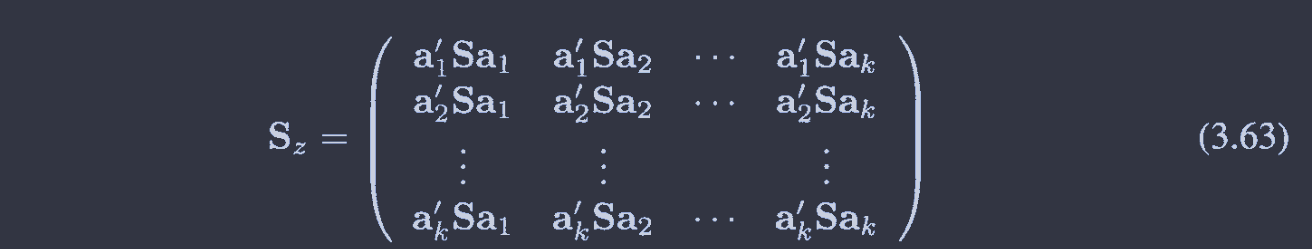
多元统计分析



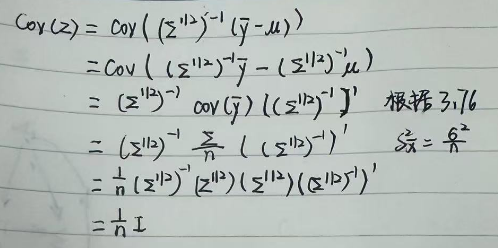


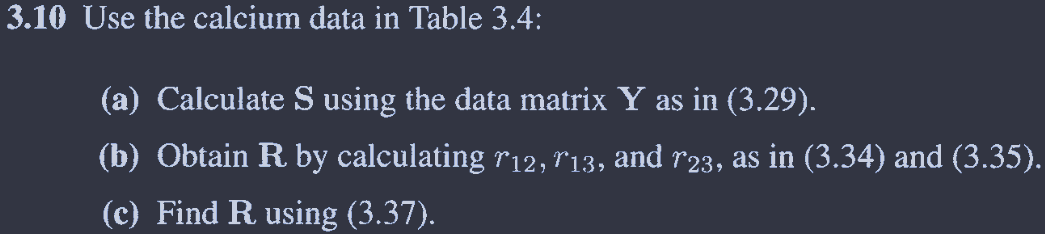




求trace(31)对角线求和 基于3.6.3 得证。







a

> y1 <- c(35,35,40,10,6,20,35,35,35,30)

> y2 <- c(3.5,4.9,30,2.8,2.7,2.8,4.6,10.9,8.0,1.6)

> y3 <- c(2.8,2.7,4.38,3.21,2.73,2.81,2.88,2.90,3.28,3.20)

>

> # 组合成矩阵

> Y <- cbind(y1, y2, y3)

> n <- nrow(Y)

>

> # 计算协方差矩阵 S

> J <- matrix(1, n, n) # 全 1 矩阵

> I <- diag(n) # 单位矩阵

> H <- I - (1/n) \* J

> S <- (1 / (n - 1)) \* t(Y) %\*% H %\*% Y

> S

y1 y2 y3

y1 140.544444 49.680000 1.9412222

y2 49.680000 72.248444 3.6760889

y3 1.941222 3.676089 0.2501211

>

> cov(Y)

y1 y2 y3

y1 140.544444 49.680000 1.9412222

y2 49.680000 72.248444 3.6760889

y3 1.941222 3.676089 0.2501211

b

> cor(Y)

y1 y2 y3

y1 1.0000000 0.4930154 0.327411

y2 0.4930154 1.0000000 0.864762

y3 0.3274110 0.8647620 1.000000

> S <- cov(Y)

>

> # 计算标准差

> s\_j <- sqrt(diag(S)) # 提取 S 的对角线元素并开方

>

> # 计算相关性矩阵 R

> R\_manual <- S / (s\_j %\*% t(s\_j)) # 按公式计算

> R\_manual

y1 y2 y3

y1 1.0000000 0.4930154 0.327411

y2 0.4930154 1.0000000 0.864762

y3 0.3274110 0.8647620 1.000000

c

> Ds <- diag(sqrt(diag(S))) # 对角线元素取平方根，形成对角矩阵

>

> # 计算 D\_s^(-1)

> Ds\_inv <- diag(1 / sqrt(diag(S))) # 直接计算 D\_s 的逆矩阵

>

> # 计算相关矩阵 R

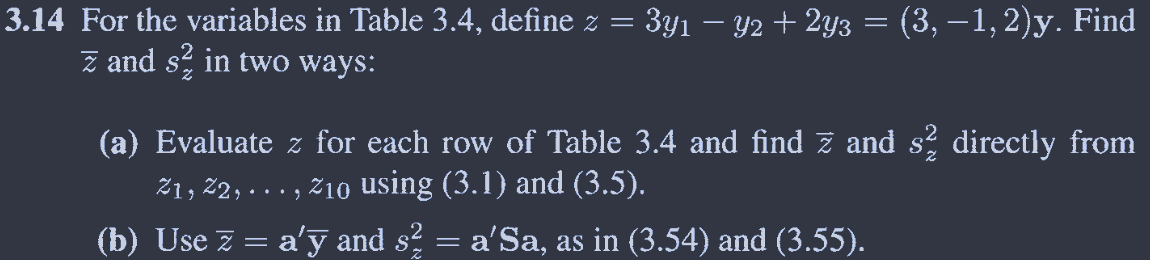
> Ds\_inv %\*% S %\*% Ds\_inv

[,1] [,2] [,3]

[1,] 1.0000000 0.4930154 0.327411

[2,] 0.4930154 1.0000000 0.864762

[3,] 0.3274110 0.8647620 1.000000



> library(data.table)

> dat <- read.table("C:/Users/hcy/Zotero/storage/MM8WF4GT/Software-Files/T3\_4\_CALCIUM.DAT") |>

+ setDT()

>

> setnames(dat,names(dat),c("num",'y1','y2','y3'))

> dat[,z := 3\*y1-y2+2\*y3][]

num y1 y2 y3 z

<int> <int> <num> <num> <num>

1: 1 35 3.5 2.80 107.10

2: 2 35 4.9 2.70 105.50

3: 3 40 30.0 4.38 98.76

4: 4 10 2.8 3.21 33.62

5: 5 6 2.7 2.73 20.76

6: 6 20 2.8 2.81 62.82

7: 7 35 4.6 2.88 106.16

8: 8 35 10.9 2.90 99.90

9: 9 35 8.0 3.28 103.56

10: 10 30 1.6 3.20 94.80

>

> # 第一种计算

> z\_bar <- sum(dat[,z])/nrow(dat)

> z\_bar

[1] 83.298

> s\_2 <- sum((dat[,z]-z\_bar)^2) /(nrow(dat)-1)

> s\_2

[1] 1048.659

>

> # 第二种计算

> a <- c(3,-1,2)

> z\_bar <- a%\*%apply(dat[,-c("num","z")],2,mean)

> z\_bar

[,1]

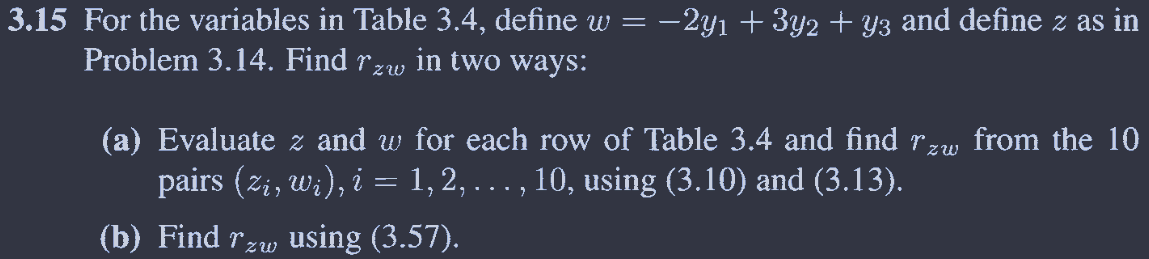
[1,] 83.298

> s\_2 <- t(a)%\*%cov(dat[,-c("num","z")])%\*%a

> s\_2

[,1]

[1,] 1048.659



> dat <- read.table("C:/Users/hcy/Zotero/storage/MM8WF4GT/Software-Files/T3\_4\_CALCIUM.DAT") |>

+ setDT()

>

> setnames(dat,names(dat),c("num",'y1','y2','y3'))

> dat[,z := 3\*y1-y2+2\*y3][]

num y1 y2 y3 z

<int> <int> <num> <num> <num>

1: 1 35 3.5 2.80 107.10

2: 2 35 4.9 2.70 105.50

3: 3 40 30.0 4.38 98.76

4: 4 10 2.8 3.21 33.62

5: 5 6 2.7 2.73 20.76

6: 6 20 2.8 2.81 62.82

7: 7 35 4.6 2.88 106.16

8: 8 35 10.9 2.90 99.90

9: 9 35 8.0 3.28 103.56

10: 10 30 1.6 3.20 94.80

> dat[,w:=-2\*y1+3\*y2+y3][]

num y1 y2 y3 z w

<int> <int> <num> <num> <num> <num>

1: 1 35 3.5 2.80 107.10 -56.70

2: 2 35 4.9 2.70 105.50 -52.60

3: 3 40 30.0 4.38 98.76 14.38

4: 4 10 2.8 3.21 33.62 -8.39

5: 5 6 2.7 2.73 20.76 -1.17

6: 6 20 2.8 2.81 62.82 -28.79

7: 7 35 4.6 2.88 106.16 -53.32

8: 8 35 10.9 2.90 99.90 -34.40

9: 9 35 8.0 3.28 103.56 -42.72

10: 10 30 1.6 3.20 94.80 -52.00

> # 第一种计算

>

> s\_zw <- (sum(dat[,z]\*dat[,w])-nrow(dat)\*mean(dat[,z])\*mean(dat[,w]))/(nrow(dat)-1)

>

> S <- function(x){

+ sum((x-mean(x))^2) /(length(x)-1)

+ }

>

> s\_z <- S(dat[,z])

> s\_w <- S(dat[,w])

>

> r\_zx <- s\_zw/sqrt(s\_z)/sqrt(s\_w)

> r\_zx

[1] -0.6105745

>

> # 第二种计算

> a <- c(3,-1,2)

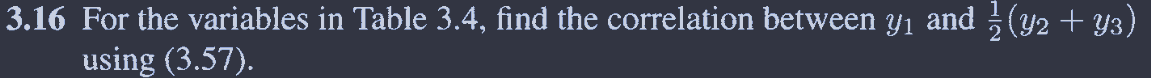
> b <- c(-2,3,1)

>

> (t(a) %\*% cov(dat[, -c("num","z", "w")]) %\*% b) / sqrt((t(a) %\*%cov(dat[, -c("num","z", "w")]) %\*% a) \* (t(b) %\*% cov(dat[, -c("num","z", "w")]) %\*%b))

[,1]

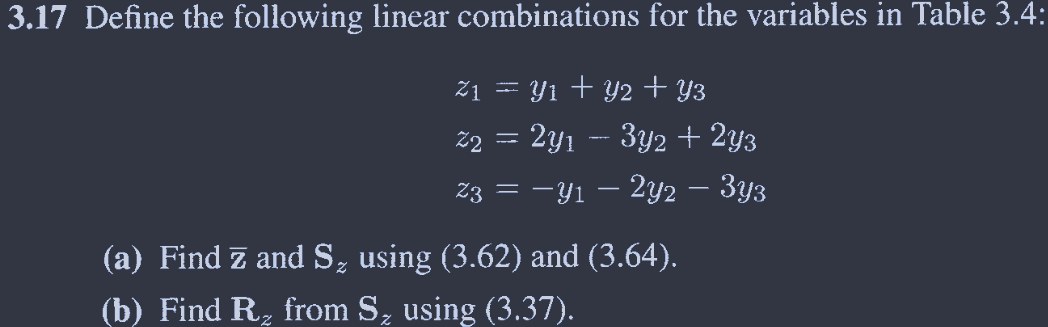
[1,] -0.6105745



> (t(a) %\*% cov(dat[, -c("num","z", "w")]) %\*% b) / sqrt((t(a) %\*%cov(dat[, -c("num","z", "w")]) %\*% a) \* (t(b) %\*% cov(dat[, -c("num","z", "w")]) %\*%b))

[,1]

[1,] 0.8178321



> dat <- read.table("C:/Users/hcy/Zotero/storage/MM8WF4GT/Software-Files/T3\_4\_CALCIUM.DAT") |>

+ setDT()

> setnames(dat,names(dat),c("num",'y1','y2','y3'))

> # dat[,z1 := 1\*y1+1\*y2+1\*y3][]

> # dat[,z2:=2\*y1 -3\*y2+2\*y3][]

> # dat[,z3:=-1\*y1 -2\*y2 -3\*y3][]

>

> #a

> a1 <- c(1,1,1)

> a2 <- c(2,-3,2)

> a3 <- c(-1,-2,-3)

>

> A <- matrix(NA,3,3)

> A[1,] <- a1

> A[2,] <- a2

> A[3,] <- a3

>

> y\_bar <- apply(dat[, c("y1","y2","y3")],2,mean) |> as.matrix()

>

> z\_bar <- A%\*%y\_bar

> z\_bar

[,1]

[1,] 38.369

[2,] 40.838

[3,] -51.727

> s\_z <- A%\*%cov(dat[, c("y1","y2","y3")])%\*%t(A)

> s\_z

[,1] [,2] [,3]

[1,] 323.6376 19.2526 -460.9770

[2,] 19.2526 588.6710 104.0717

[3,] -460.9770 104.0717 686.2697

>

> # b

> D <- diag(sqrt(diag(s\_z)))

> D\_inv <- diag(1 / diag(D))

>

> R\_z <-D\_inv %\*% s\_z %\*%D\_inv

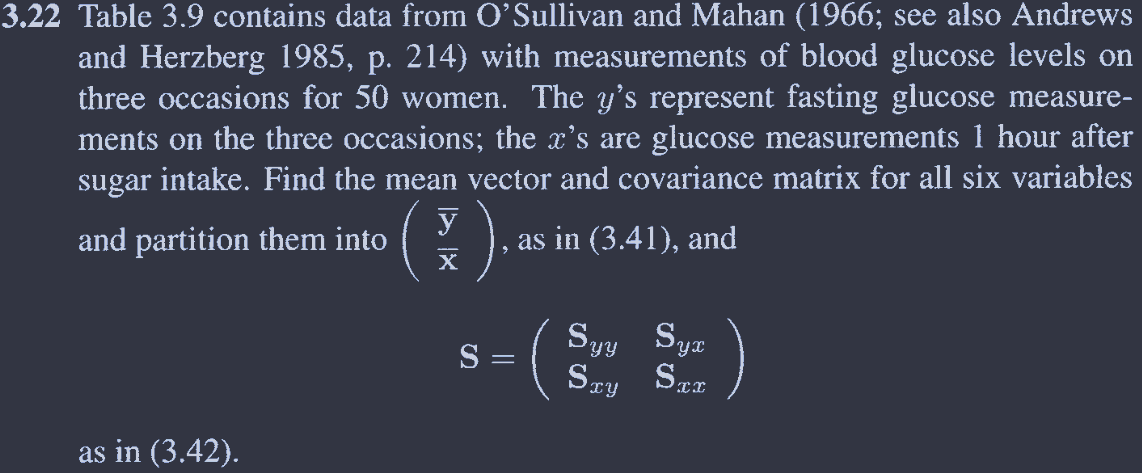
> R\_z

[,1] [,2] [,3]

[1,] 1.00000000 0.04410862 -0.9781430

[2,] 0.04410862 1.00000000 0.1637378

[3,] -0.97814302 0.16373782 1.0000000



> dat <- read.table("C:/Users/hcy/Zotero/storage/MM8WF4GT/Software-Files/T3\_9\_GLUCOSE.DAT")

> setnames(dat,names(dat),c("y1","y2","y3","x1","x2","x3"))

>

> m <- apply(dat,2,mean)

> m

y1 y2 y3 x1 x2 x3

70.08 73.54 75.10 109.68 104.24 109.98

> s <- cov(dat)

> s

y1 y2 y3 x1 x2 x3

y1 95.54449 17.608980 12.17551 60.515918 23.00082 62.838367

y2 17.60898 73.192245 14.25102 5.727347 61.27592 -1.662449

y3 12.17551 14.251020 76.17347 46.746939 32.77143 69.838776

x1 60.51592 5.727347 46.74694 808.630204 320.58857 227.360816

x2 23.00082 61.275918 32.77143 320.588571 505.85959 167.351837

x3 62.83837 -1.662449 69.83878 227.360816 167.35184 508.713878