154HW1_Jiyoon_Jeong_Lab2

Jiyoon Clover Jeong

9/10/2017

Problem 1

```
A <- matrix(c(1,4,2,0,-3,1),2,3)
B <- matrix(c(2,-1,3,2,4,0),2,3)
C <- matrix(c(0,4,1,-1,0,-2),2,3)

A + B
```

```
##      [,1] [,2] [,3]
## [1,]    3    5    1
## [2,]    3    2    1
```

```
(A + C) + B
```

```
##      [,1] [,2] [,3]
## [1,]    3    6    1
## [2,]    7    1   -1
```

```
A -(C + B)
```

```
##      [,1] [,2] [,3]
## [1,]   -1   -2   -7
## [2,]    1   -1    3
```

```
-(A + B)
```

```
##      [,1] [,2] [,3]
## [1,]   -3   -5   -1
## [2,]   -3   -2   -1
```

```
(A - B) + C
```

```
##      [,1] [,2] [,3]
## [1,]   -1    0   -7
## [2,]    9   -3   -1
```

Problem 2

```
X <- matrix(c(2, 1, 0, 9, 4, 2, 3, 8, 3, 5, 2, 4, 7, 3, 4, 5, 8, 7, 7, 2, 9, 8, 7, 1), 6, 4, byrow = T)

rownames(X) <- c(letters[1:6])
colnames(X) <- c("Y","X1","X2","X3")
X
```

```
##   Y X1 X2 X3
## a 2  1  0  9
## b 4  2  3  8
## c 3  5  2  4
## d 7  3  4  5
```

```
## e 8  7  7  2
## f 9  8  7  1
```

(a)

```
X[1,1] + X[2,1] + X[3,1] + X[4,1] + X[5,1] + X[6,1]
```

```
## [1] 33
```

```
X[,1] %*% matrix(1, 6, 1)
```

```
##      [,1]
## [1,]   33
```

(b)

```
(X[1,2] + X[2,2] + X[3,2] + X[4,2] + X[5,2] + X[6,2]) /6
```

```
## [1] 4.333333
```

```
(X[,2] %*% matrix(1, 6, 1)) /6
```

```
##          [,1]
## [1,] 4.333333
```

(c)

```
t(X[,1, drop =F]) %*% X[,3, drop = F]
```

```
##    X2
## Y 165
```

(d)

```
(t(X[,4, drop = F]) %*% X[,4, drop = F] ) - (t(X[,4, drop = F]) %*% matrix(1,6,1))^(2)/6
```

```
##          X3
## X3 50.83333
```

(e)

```
meany <- (X[,1] %*% matrix(1, 6, 1)) /6
meanx1 <- (X[,2] %*% matrix(1, 6, 1)) /6
meanx2 <- (X[,3] %*% matrix(1, 6, 1)) /6
meanx3 <- (X[,4] %*% matrix(1, 6, 1)) /6

cbind((X[,1,drop =F] - meany[1]), (X[,2,drop =F] - meanx1[1]),
           (X[,3,drop =F] - meanx2[1]), (X[,4,drop =F] - meanx3[1]))
```

```
##        Y         X1         X2         X3
## a -3.5 -3.3333333 -3.8333333  4.1666667
## b -1.5 -2.3333333 -0.8333333  3.1666667
## c -2.5  0.6666667 -1.8333333 -0.8333333
## d  1.5 -1.3333333  0.1666667  0.1666667
## e  2.5  2.6666667  3.1666667 -2.8333333
## f  3.5  3.6666667  3.1666667 -3.8333333
```

```
X
```

```
##   Y X1 X2 X3
## a 2  1  0  9
## b 4  2  3  8
## c 3  5  2  4
## d 7  3  4  5
## e 8  7  7  2
## f 9  8  7  1
```

```
# check with : apply(X,2, FUN = function(x) {x - mean(x)})
```

(f)

```
p <- ncol(X) #number of variables
n <- nrow(X) #number of objects

#create means for each column
meanmat <- matrix(data=1, nrow=n) %*% cbind(meany[1],meanx1[1],meanx2[1],meanx3[1])

#creates a difference matrix
D <- X - meanmat

#creates the covariance matrix
Cov <- (n-1)^(-1) * (t(D) %*% D)
Cov
```

```
##        Y        X1        X2        X3
## Y   8.3  6.200000  7.700000 -7.500000
## X1  6.2  7.866667  6.666667 -8.733333
## X2  7.7  6.666667  7.766667 -7.633333
## X3 -7.5 -8.733333 -7.633333 10.166667
```

```
# check with cov(X)
cov(X)
```

```
##        Y        X1        X2        X3
## Y   8.3  6.200000  7.700000 -7.500000
## X1  6.2  7.866667  6.666667 -8.733333
## X2  7.7  6.666667  7.766667 -7.633333
## X3 -7.5 -8.733333 -7.633333 10.166667
```

problem 3

(a)

```r
a <- abs(0.5)
b <- abs(4)

a * b * cos(45*pi/180)
```

```
## [1] 1.414214
```

(b)

```r
a <- abs(4)
b <- abs(1)

a * b * cos(90*pi/180)
```

```
## [1] 2.449294e-16
```

(c)

```r
a <- abs(1)
b <- abs(1)

a * b * cos(120*pi/180)
```

```
## [1] -0.5
```

problem 4

```r
u = c(1, 3, 5)
v = c(2, 4, 6)

# v onto u
proj <- function(v,u){

  c <- (t(u) %*% v) / (t(u) %*% u)
  return(c*u)

}
proj(u,v)
```

```
## [1] 1.571429 3.142857 4.714286
```

problem 5

(a)

```
x = c(1, 2, 3)
y = c(3, 0, 2)
z = c(3, 1, 1)

norm <- function(x){
  return(sum(x^2)^(1/2))
}

u1 <- x
e1 <- u1/ norm(u1)
cat("u1 : ", u1, "\n")
```

```
## u1 :  1 2 3
```

```
cat("e1 : ", e1, "\n")
```

```
## e1 :  0.2672612 0.5345225 0.8017837
```

```
u2 <- y - proj(y,u1)
e2 <- u2 / norm(u2)
cat("u2 : ", u2, "\n")
```

```
## u2 :  2.357143 -1.285714 0.07142857
```

```
cat("e2 : ", e2, "\n")
```

```
## e2 :  0.8775851 -0.4786828 0.02659349
```

```
u3 <- z - proj(z,u1) - proj(z,u2)
e3 <- u3 / norm(u3)
cat("u3 : ", u3, "\n")
```

```
## u3 :  0.5148515 0.9009901 -0.7722772
```

```
cat("e3 : ", e3, "\n")
```

```
## e3 :  0.3980149 0.696526 -0.5970223
```

(b)

```
x = c(2, 1)
y = c(1, 2)
z = c(1, 1)

u1 <- x
e1 <- u1/ norm(u1)
cat("u1 : ", u1, "\n")
```

```
## u1 :  2 1
```

```
cat("e1 : ", e1, "\n")
```

```
## e1 :  0.8944272 0.4472136
```

```
u2 <- y - proj(y,u1)
e2 <- u2 / norm(u2)
cat("u2 : ", u2, "\n")
```

```
## u2 :  -0.6 1.2
```

```
cat("e2 : ", e2, "\n")
```

```
## e2 :  -0.4472136 0.8944272
```

```
u3 <- z - proj(z,u1) - proj(z,u2)
u3 <- round(u3,10)
e3 <- u3 / norm(u3)
cat("u3 : ", u3, "\n")
```

```
## u3 :  0 0
```

```
cat("Since u3 is (0, 0), we can omit u3 and conclude that e1 and e2 forms orthonormal basis of R^2. As w
```

```
## Since u3 is (0, 0), we can omit u3 and conclude that e1 and e2 forms orthonormal basis of R^2. As we
```

problem 6

```
lp_norm <- function(x, p=1){
  if(p =="max"){
    return(max(abs(x)))
  }
  else{
    lp <- sum((abs(x)^p))^(1/p)
  }
  return(lp)
}



zero <- rep(0, 10)
p = 1
lp_norm(zero,p)
```

```
## [1] 0
```

```
ones <- rep(1, 5)
p = 3
lp_norm(ones,p)
```

```
## [1] 1.709976
```

```
u <- rep(0.4472136, 5)
p = 2
lp_norm(u,p)
```

```
## [1] 1
```

```
u <- c(-40:0)
p = 100
lp_norm(u,p)
```

```
## [1] 40.03297
```

```r
u <- c(1:1000)
p = "max"
lp_norm(u, p)
```

```
## [1] 1000
```

problem 7

```r
u1 <- 1/sqrt(11) * c(3, 1, 1)
u2 <- 1/sqrt(6) * c(-1, 2, 1)
u3 <- 1/sqrt(66) * c(-1,-4, 7)

#norms of u1, u2, and u3 are zero.
lp_norm(u1,2)
```

```
## [1] 1
```

```r
lp_norm(u2,2)
```

```
## [1] 1
```

```r
lp_norm(u3,2)
```

```
## [1] 1
```

```r
#<u1,u2> = <u1,u3> = <u2,u3> = 0
t(u1) %*% u2
```

```
##      [,1]
## [1,]    0
```

```r
t(u1) %*% u3
```

```
##              [,1]
## [1,] 5.551115e-17
```

```r
t(u2) %*% u3
```

```
##      [,1]
## [1,]    0
```

From the calculation above, u1, u2, and u3 are orthonormal vectors

problem 8

(a)

```r
X <- as.matrix(USArrests)
class(X)
```

```
## [1] "matrix"
```

(b)

```
n= nrow(X)
p = ncol(X)

cat("# rows: ", n, "\n")
```

```
## # rows:  50
```

```
cat("# columns: ", p, "\n")
```

```
## # columns:  4
```

(c)

```
D <- diag(1/n,n)
sum(diag(D))
```

```
## [1] 1
```

(d) column means

```
one <- rep(1,n)
g <- t(X) %*% D %*% one
g
```

```
##             [,1]
## Murder     7.788
## Assault  170.760
## UrbanPop  65.540
## Rape      21.232
```

(e) mean-centered matrix

```
one <- matrix(1,n,1)
Xc <- X - one %*% t(g)
colMeans(Xc)
```

```
##        Murder       Assault      UrbanPop          Rape
##  2.469136e-15 -7.617018e-14 -6.252776e-15 -2.700062e-15
```

(f) the (population) variance-covariance matrix

```
V <- t(X) %*% D %*% X - g %*% t(g)
V
```

```
##              Murder   Assault  UrbanPop      Rape
## Murder     18.59106  285.2411   4.29848  22.53158
## Assault   285.24112 6806.2624 306.02960 508.88368
## UrbanPop    4.29848  306.0296 205.32840  54.65272
## Rape       22.53158  508.8837  54.65272  85.97458
```

(g) D__s be a p×p diagonal matrix with elements on the diagonal equal to 1/Sj , where Sj is the standard deviation for the j-th variable

```
D_s <-diag(c(1/sd(X[,1]),1/sd(X[,2]),1/sd(X[,3]), 1/sd(X[,4])))

D_s <-diag(diag(1/sqrt(V)))
D_s
```

```
##           [,1]      [,2]       [,3]      [,4]
## [1,] 0.2319252 0.0000000 0.00000000 0.0000000
## [2,] 0.0000000 0.0121212 0.00000000 0.0000000
## [3,] 0.0000000 0.0000000 0.06978715 0.0000000
## [4,] 0.0000000 0.0000000 0.00000000 0.1078487
```

```
diag(D_s)
```

```
## [1] 0.23192522 0.01212120 0.06978715 0.10784872
```

(h) matrix of standardized data

```
Z <- Xc %*% D_s

colMeans(Z)
```

```
## [1]  5.630219e-16 -9.207912e-16 -4.440892e-16 -2.925438e-16
```

```
apply(Z, 2, sd)
```

```
## [1] 1.010153 1.010153 1.010153 1.010153
```

(i) the (population) correlation matrix

```
R <- D_s %*% V %*% D_s
R
```

```
##            [,1]      [,2]       [,3]      [,4]
## [1,] 1.00000000 0.8018733 0.06957262 0.5635788
## [2,] 0.80187331 1.0000000 0.25887170 0.6652412
## [3,] 0.06957262 0.2588717 1.00000000 0.4113412
## [4,] 0.56357883 0.6652412 0.41134124 1.0000000
```

(j) R can also be obtained as R = t(Z) D Z

```
R <- t(Z) %*% D %*% Z
R
```

```
##            [,1]      [,2]       [,3]      [,4]
## [1,] 1.00000000 0.8018733 0.06957262 0.5635788
## [2,] 0.80187331 1.0000000 0.25887170 0.6652412
## [3,] 0.06957262 0.2588717 1.00000000 0.4113412
## [4,] 0.56357883 0.6652412 0.41134124 1.0000000
```

```r
cor(X)
```

```
##              Murder   Assault   UrbanPop      Rape
## Murder   1.00000000 0.8018733 0.06957262 0.5635788
## Assault  0.80187331 1.0000000 0.25887170 0.6652412
## UrbanPop 0.06957262 0.2588717 1.00000000 0.4113412
## Rape     0.56357883 0.6652412 0.41134124 1.0000000
```