Homework 02

Joseph Blubaugh
jblubau1@tamu.edu

STAT 636-720

1.

a)

$$A = \begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix}$$

$$\begin{aligned} A - \lambda I &= \begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \\ &= \begin{bmatrix} 1-\lambda & 2 \\ 2 & -2-\lambda \end{bmatrix} \end{aligned}$$

$$\begin{aligned} |A - \lambda I| =& (1-\lambda)(-2-\lambda) - 4 \\ =& -2 - \lambda + 2\lambda + \lambda^2 - 4 \\ =& -6 + \lambda + \lambda^2 \end{aligned}$$

$$0 = (\lambda - 2)(\lambda + 3)$$

Eigen Values: $\lambda_1 = 2, \lambda_2 = -3$

$$\begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix} = 2 \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix}$$

$$\begin{aligned} e_{11} + 2e_{12} =& 2e_{11} \\ 2e_{11} - 2e_{12} =& 2e_{12} \end{aligned}$$

$$\begin{aligned} -e_{11} + 2e_{12} =& 0 \\ 2e_{11} - 4e_{12} =& 0 \end{aligned}$$

$$\begin{aligned} -e_{11} =& -2e_{12} \\ e_{11} =& 2e_{12} \end{aligned}$$

$$\begin{bmatrix}1 & 2\\ 2 & -2\end{bmatrix}\begin{bmatrix}e_{21}\\ e_{22}\end{bmatrix} = -3\begin{bmatrix}e_{21}\\ e_{22}\end{bmatrix}$$

$$\begin{aligned}e_{21} + 2e_{22} &= -3e_{21}\\ 2e_{21} - 2e_{22} &= -3e_{22}\end{aligned}$$

$$\begin{aligned}4e_{21} + 2e_{22} &= 0\\ 2e_{21} + e_{22} &= 0\end{aligned}$$

$$\begin{aligned}2e_{21} &= -e_{22}\\ 2e_{21} &= -e_{22}\end{aligned}$$

$$e_1 = \begin{bmatrix}2\\ 1\end{bmatrix}\ e_2 = \begin{bmatrix}1\\ -2\end{bmatrix}$$

Normalizing Eigen Vectors $e_1 = \begin{bmatrix}\frac{2}{\sqrt{4}}\\ \frac{1}{\sqrt{4}}\end{bmatrix}\ e_2 = \begin{bmatrix}\frac{1}{\sqrt{5}}\\ \frac{-2}{\sqrt{5}}\end{bmatrix}$

b) Spectral Decomposition

$$\begin{aligned}A &= \lambda_1 e_1 e_1' + \lambda_2 e_2 e_2'\\ &= 2\begin{bmatrix}1 & \frac{1}{2}\\ \frac{1}{2} & \frac{1}{4}\end{bmatrix} - 3\begin{bmatrix}\frac{1}{5} & \frac{2}{5}\\ \frac{2}{5} & \frac{4}{5}\end{bmatrix}\end{aligned}$$

c)

Determinant of A: $(1)(-2) - (2)(2) = -6$

Product of Eigen Values: $(2)(-3) = -6$

d)

Sum of trace of A: $1 - 2 = -1$

Sum of Eigen Values: $2 - 3 = -1$

e) A is symmetrical but it is not orthogonal because it does not satisfy the condition that $A'A = AA' = I$

f) A is not positive definite. If we plug in a small number for a and a big number for b the result will be negative which means A is not positive definite

$$\begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = a^2 - 2b^2 + 4ab$$

g)

$$A^{-1} = \frac{1}{(1)(-2) - (2)(2)} \begin{bmatrix} -2 & -2 \\ -2 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{-1}{6} \end{bmatrix}$$

$$\begin{aligned} A^{-1} - \lambda I &= \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{-1}{6} \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{3} - \lambda & \frac{1}{3} \\ \frac{1}{3} & \frac{-1}{6} - \lambda \end{bmatrix} \\ |A^{-1} - \lambda I| &= (\frac{1}{3} - \lambda)(\frac{-1}{6} - \lambda) - \frac{1}{9} \\ &= \frac{-1}{18} - \frac{1}{3}\lambda + \frac{1}{6}\lambda + \lambda^2 - \frac{1}{9} \\ &= \frac{-1}{6} - \frac{1}{6}\lambda + \lambda^2 \end{aligned}$$

$$0 = (\lambda - \frac{2}{6})(\lambda - \frac{3}{6})$$

Eigen Values:

$$\lambda_1 = \frac{-1}{3}, \lambda_2 = \frac{1}{2}$$

Eigen Vectors:

$$\begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{-1}{6} \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix} = \frac{-1}{3} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix}$$

$$\frac{e_{11}}{3} + \frac{e_{12}}{3} = \frac{-e_{11}}{3}$$
$$\frac{e_{11}}{3} - \frac{e_{12}}{6} = \frac{-e_{12}}{3}$$

$$\frac{2e_{11}}{3} + \frac{e_{12}}{3} = 0$$
$$\frac{e_{11}}{3} + \frac{e_{12}}{6} = 0$$

$$\frac{2e_{11}}{3} = \frac{-e_{12}}{3}$$
$$\frac{e_{11}}{3} = \frac{-e_{12}}{6}$$

$$\begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{-1}{6} \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix}$$

$$\frac{e_{11}}{3} + \frac{e_{12}}{3} = \frac{e_{11}}{2}$$
$$\frac{e_{11}}{3} - \frac{e_{12}}{6} = \frac{e_{12}}{2}$$

$$\frac{-e_{11}}{6} + \frac{e_{12}}{3} = 0$$
$$\frac{e_{11}}{3} - \frac{4e_{12}}{6} = 0$$

$$\frac{-e_{11}}{6} = \frac{-2e_{12}}{6}$$
$$\frac{e_{11}}{3} = \frac{2e_{12}}{3}$$

$$e_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix} e_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

Normalized Eigen Vectors: $e_1 = \begin{bmatrix} \frac{1}{\sqrt{5}} \\ \frac{-2}{\sqrt{5}} \end{bmatrix} e_2 = \begin{bmatrix} \frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{bmatrix}$

2.

```r
(A = matrix(c(4, 4.001, 4.001, 4.002), nrow = 2))
```

```
      [,1]  [,2]
[1,] 4.000 4.001
[2,] 4.001 4.002
```

```r
(B = matrix(c(4, 4.001, 4.001, 4.002001), nrow = 2))
```

```
      [,1]     [,2]
[1,] 4.000 4.001000
[2,] 4.001 4.002001
```

```r
solve(A); solve(B) * -3
```

```
         [,1]     [,2]
[1,] -4002000  4001000
[2,]  4001000 -4000000
```

```
         [,1]     [,2]
[1,] -4002001  4001000
[2,]  4001000 -4000000
```

3.

a) For $X_1 - 2X_2$

$$\begin{aligned} E(X_1 - 2X_2) =& E(X_1) - 2E(X_2) = \mu_1 - 2\mu_2 \end{aligned}$$

$$\begin{aligned} Var(X_1 - 2X_2) =& E((X_1 - 2X_2) - (\mu_1 - 2\mu_2))^2 \\ =& E((X_1 - \mu_1) - 2(X_2 - \mu_2))^2 \\ =& E((X_1 - \mu_1)^2 + 4(X_2 - \mu_2)^2 - 4(X_1 - \mu_1)(X_2 - \mu_2)) \\ =& Var(X_1) + 4Var(X_2) - 4Cov(X_1, X_2) \\ =& \sigma_{11} + 4\sigma_{22} - 4\sigma_{12} \end{aligned}$$

b) For $X_1 + 2X_2 - X_3$

$$\begin{aligned} E(X_1 + 2X_2 - X_3) =& E(X_1) + 2E(X_2) - E(X_3) = \mu_1 + 2\mu_2 - \mu_3 \end{aligned}$$

$$\begin{aligned} Var(X1 + 2X_2 - X_3) =& E((X1 + 2X_2 - X_3) - (\mu_1 + 2\mu_2 - \mu_3))^2 \\ =& E((X_1 - \mu_1) + 2(X_2 - \mu_2) - (X_3 - \mu_3))^2 \\ =& E((X_1 - \mu_1)^2 + 4(X_2 - \mu_2)^2 + (X_3 = \mu_3)^2) + \\ & 4(X_1 - \mu_1)(X_2 - \mu_2) - 2(X_1 - \mu_1)(X_3 - \mu_3) - 4(X_2 - \mu_2)(X_3 - \mu_3) \\ =& Var(X_1) + 4Var(X_2) + Var(X_3) + 4Cov(X_1, X_2) - 2Cov(X_1, X_3) \\ & - 4Cov(X_2, X_3) \\ =& \sigma_{11} + 4\sigma_{22} + \sigma_{33} + 4\sigma_{12} - 2\sigma_{13} - 4\sigma_{23} \end{aligned}$$

c) For $3X_1 - 4X_2$, where $\sigma_{12} = 0$

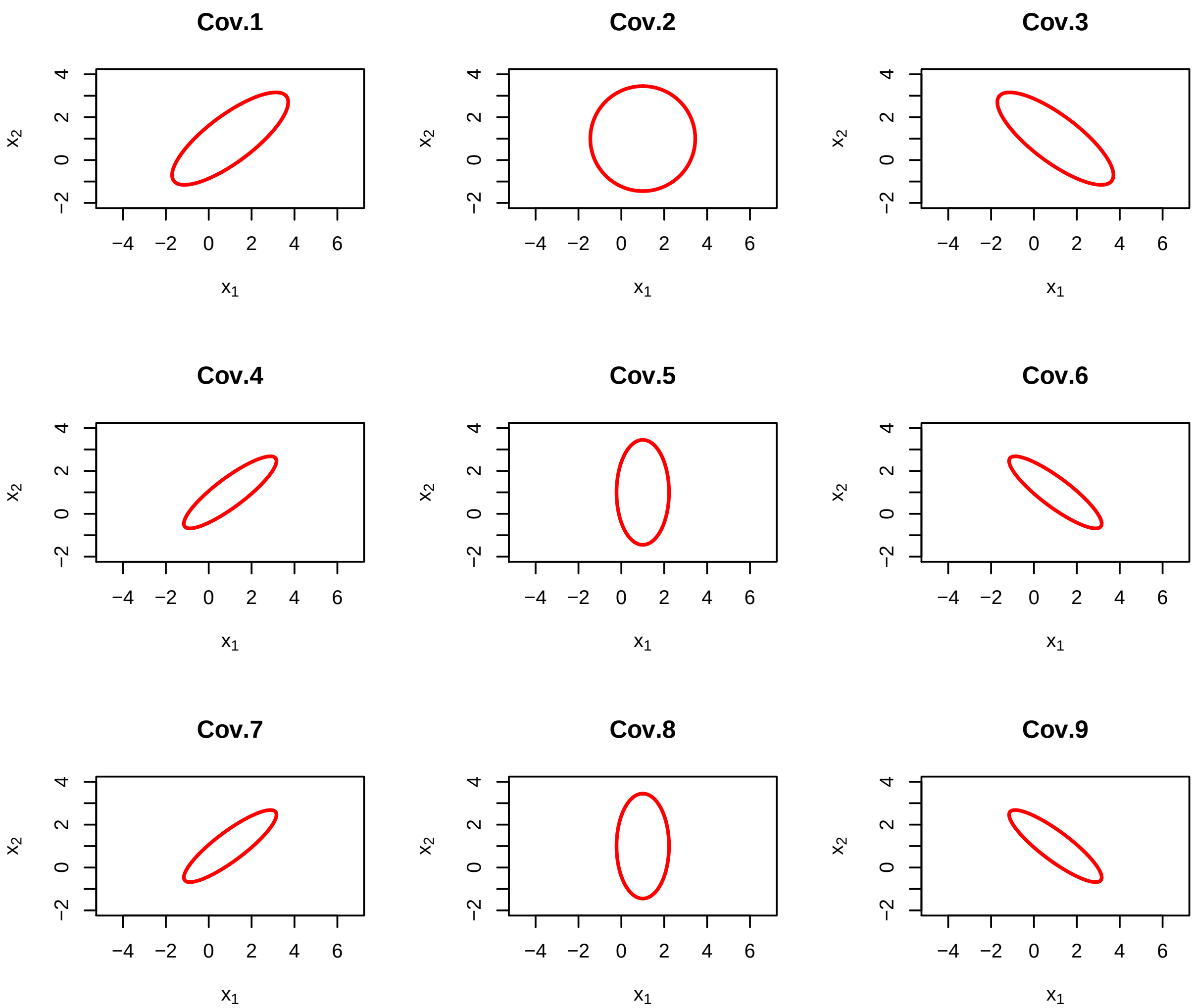
$$\begin{aligned} E(3X_1 - 4X_2) =& 3E(X_1) - 4E(X_2) = 3\mu_1 - 4\mu_2 \end{aligned}$$

$$\begin{aligned} Var(3X_1 - 4X_2) =& E((3X_1 - 4X_2) - (3\mu_1 - 4\mu_2))^2 \\ =& E(3(X_1 - \mu_1) - 4(X_2 - \mu_2))^2 \\ =& E(9(X_1 - \mu_1) + 16(X_2 - \mu_2)^2 + 288(X_1 - \mu_1)(X_2 - \mu_2)) \\ =& 9Var(X_1) + 16Var(X_2) + 288Cov(X_1, X_2) \\ =& 9\sigma_{11} + 16\sigma_{22} \end{aligned}$$

4.

```
library(plotrix)

## Create the Covariance Matrices
Cov.1 = matrix(c(1, .8, .8, 1), nrow = 2)
Cov.2 = matrix(c(1, 0, 0, 1), nrow = 2)
Cov.3 = matrix(c(1, -.8, -.8, 1), nrow = 2)
Cov.4 = matrix(c(1, .4, .4, .25), nrow = 2)
Cov.5 = matrix(c(1, 0, 0, .25), nrow = 2)
Cov.6 = matrix(c(1, -.4, -.4, .25), nrow = 2)
Cov.7 = matrix(c(.25, .4, .4, 1), nrow = 2)
Cov.8 = matrix(c(.25, 0, 0, 1), nrow = 2)
Cov.9 = matrix(c(.25, -.4, -.4, 1), nrow = 2)

## Calculate the correlation for each matrix
Cor.1 = Cov.1[1, 2] / (sqrt(Cov.1[1, 1]) * sqrt(Cov.1[2, 2]))
Cor.2 = Cov.2[1, 2] / (sqrt(Cov.2[1, 1]) * sqrt(Cov.2[2, 2]))
Cor.3 = Cov.3[1, 2] / (sqrt(Cov.3[1, 1]) * sqrt(Cov.3[2, 2]))
Cor.4 = Cov.4[1, 2] / (sqrt(Cov.4[1, 1]) * sqrt(Cov.4[2, 2]))
Cor.5 = Cov.5[1, 2] / (sqrt(Cov.5[1, 1]) * sqrt(Cov.5[2, 2]))
Cor.6 = Cov.6[1, 2] / (sqrt(Cov.6[1, 1]) * sqrt(Cov.6[2, 2]))
Cor.7 = Cov.7[1, 2] / (sqrt(Cov.7[1, 1]) * sqrt(Cov.7[2, 2]))
Cor.8 = Cov.8[1, 2] / (sqrt(Cov.8[1, 1]) * sqrt(Cov.8[2, 2]))
Cor.9 = Cov.9[1, 2] / (sqrt(Cov.9[1, 1]) * sqrt(Cov.9[2, 2]))

## Create angle for the ellipse
Theta.1 = acos(Cor.1)
Theta.2 = acos(Cor.2)
Theta.3 = acos(Cor.3)
Theta.4 = acos(Cor.4)
Theta.5 = acos(Cor.5)
Theta.6 = acos(Cor.6)
Theta.7 = acos(Cor.7)
Theta.8 = acos(Cor.8)
Theta.9 = acos(Cor.9)

c2 = qchisq(0.95, 2)

ellipse.plot = function(cov, theta, name) {
  plot(c(-2,4), c(-2,4), main = name, type = "n",
       xlab = expression(x[1]), ylab = expression(x[2]), asp = 1)
  draw.ellipse(x = 1, y = 1,
               a = sqrt(c2 * eigen(cov)[[1]][1]),
               b = sqrt(c2 * eigen(cov)[[1]][2]),
```

```
                  angle = theta * (360 / (2*pi)), deg = TRUE,
                  border = "red", lwd = 2)
}

par(mfrow = c(3, 3))
ellipse.plot(cov = Cov.1, theta = Theta.1, name = 'Cov.1')
ellipse.plot(cov = Cov.2, theta = Theta.2, name = 'Cov.2')
ellipse.plot(cov = Cov.3, theta = Theta.3, name = 'Cov.3')
ellipse.plot(cov = Cov.4, theta = Theta.4, name = 'Cov.4')
ellipse.plot(cov = Cov.5, theta = Theta.5, name = 'Cov.5')
ellipse.plot(cov = Cov.6, theta = Theta.6, name = 'Cov.6')
ellipse.plot(cov = Cov.7, theta = Theta.7, name = 'Cov.7')
ellipse.plot(cov = Cov.8, theta = Theta.8, name = 'Cov.8')
ellipse.plot(cov = Cov.9, theta = Theta.9, name = 'Cov.9')
```



```r
library(mvtnorm)

## Generate random variables using variance from each covariance matrix
Y.1 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.1)
Y.2 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.2)
Y.3 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.3)
Y.4 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.4)
Y.5 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.5)
Y.6 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.6)
Y.7 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.7)
Y.8 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.8)
Y.9 = rmvnorm(5000, mean = c(1, 1), sigma = Cov.9)

Distance = function(X, Cov, mu = c(1,1)) {
  x = t(X - mu) %*% solve(Cov) %*% (X - mu)
  return(x)
  }

## Vectors of distance for each of the random samples
D.1 = as.numeric(); D.2 = as.numeric(); D.3 = as.numeric()
D.4 = as.numeric(); D.5 = as.numeric(); D.6 = as.numeric()
D.7 = as.numeric(); D.8 = as.numeric(); D.9 = as.numeric()

for(i in 1:5000) {
  D.1 = c(D.1, Distance(Y.1[i,], Cov.1))
  D.2 = c(D.2, Distance(Y.2[i,], Cov.2))
  D.3 = c(D.3, Distance(Y.3[i,], Cov.3))
  D.4 = c(D.4, Distance(Y.4[i,], Cov.4))
  D.5 = c(D.5, Distance(Y.5[i,], Cov.5))
  D.6 = c(D.6, Distance(Y.6[i,], Cov.6))
  D.7 = c(D.7, Distance(Y.7[i,], Cov.7))
  D.8 = c(D.8, Distance(Y.8[i,], Cov.8))
  D.9 = c(D.9, Distance(Y.9[i,], Cov.9))
}

## Proportion of Random Samples less than qchisq(0.95, 2)
(Results =
  data.frame(
  Y.1 = length(which(D.1 < c2)) / 5000,
  Y.2 = length(which(D.2 < c2)) / 5000,
  Y.3 = length(which(D.3 < c2)) / 5000,
  Y.4 = length(which(D.4 < c2)) / 5000,
  Y.5 = length(which(D.5 < c2)) / 5000,
  Y.6 = length(which(D.6 < c2)) / 5000,
  Y.7 = length(which(D.7 < c2)) / 5000,
```

```
  Y.8 = length(which(D.8 < c2)) / 5000,
  Y.9 = length(which(D.9 < c2)) / 5000
  )
)
```

```
     Y.1    Y.2  Y.3    Y.4    Y.5  Y.6   Y.7    Y.8    Y.9
1 0.9458 0.9474 0.95 0.9514 0.9454 0.95 0.952 0.9492 0.9536
```

5.

a) $(4+3)/2 = 3.5$

b)

$$E(BX^{(2)}) = \begin{bmatrix} 1 & -2 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} X_1 - 2X_2 \\ 2X_1 - X_2 \end{bmatrix} = \begin{bmatrix} 2-2(1) \\ 2(2)-1 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

c)

$$\begin{aligned} \begin{bmatrix} 1 & 2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} =& X_1 + 2X_2 \\ =& 3 + 2(1) \\ =& 5 \end{aligned}$$

d)

$$\begin{aligned} Cov(X^{(1)}, X^{(2)}) =& Cov(X_1 + X_2, X_3 + X_4) \\ =& Cov(X_1, X_3) + Cov(X_1, X_4) + Cov(X_2, X_3) + Cov(X_2, X_4) \\ =& \sigma_{13} + \sigma_{14} + \sigma_{23} + \sigma_{24} \\ =& 2 + 2 + 1 + 0 = 5 \end{aligned}$$

e)

$$\begin{aligned} Cov(AX^{(1)}, BX^{(2)}) =& Cov(X_1 + 2X_2, X_3 - 2X_4) \\ =& Cov(X_1, X_3) - 2Cov(X_1, X_4) + 2Cov(X_2, X_3) - 4Cov(X_2, X_4) \\ =& \sigma_{13} - 2\sigma_{14} + 2\sigma_{23} - 4\sigma_{24} \\ =& 2 - 2(2) + 2(1) - 4(0) = 0 \end{aligned}$$

6.

```
## mu and sigma
mu = matrix(c(1, -1))
sigma = matrix(c(1, .8, .8, 1), nrow = 2)

## generating random data
set.seed(101)
x = rmvnorm(n = 100, mean = mu, sigma = sigma)

## sample variances
s.11 = sum((x[, 1] - mu[1])^2)/100
s.22 = sum((x[, 2] - mu[2])^2)/100
s.12 = sum((x[, 1] - mu[1])*(x[, 2] - mu[2]))/100

S.n = matrix(c(s.11, s.12, s.12, s.22), nrow = 2)

r.12 = s.12 / (sqrt(s.11) * sqrt(s.22))

(t(x[, 1]) %*% x[, 1])/100
```

```
         [,1]
[1,] 1.843454
```

```
D = data.frame(
  d.1 = x[, 1] - mean(x[, 1]),
  d.2 = x[, 2] - mean(x[, 2])
)

## a)
s.11; (t(D$d.1) %*% D$d.1) / 100
```

```
[1] 0.8684806

         [,1]
[1,] 0.868324
```

```
## b)
s.22; (t(D$d.2) %*% D$d.2) / 100
```

```
[1] 1.033203

         [,1]
[1,] 1.024416
```

```
## c)
s.12; (t(D$d.1) %*% D$d.2) / 100

[1] 0.7688845

          [,1]
[1,] 0.7677115

## d)
S.n; cov(x)

          [,1]      [,2]
[1,] 0.8684806 0.7688845
[2,] 0.7688845 1.0332029

          [,1]      [,2]
[1,] 0.8770949 0.7754662
[2,] 0.7754662 1.0347641

D = as.matrix(D)

## e)
S.n; (t(D) %*% D) / 100

          [,1]      [,2]
[1,] 0.8684806 0.7688845
[2,] 0.7688845 1.0332029

          d.1       d.2
d.1 0.8683240 0.7677115
d.2 0.7677115 1.0244165

## f)
r.12; (t(D[, 1]) %*% D[, 2]) /
  (sqrt(t(D[, 1]) %*% D[, 1]) *
     sqrt(t(D[, 2]) %*% D[, 2]))

[1] 0.8116863

          [,1]
[1,] 0.8139896
```