4630 Assignment 1 R Code

Ravish Kamath: 213893664

06 December, 2022

```
## *** Package RVAideMemoire v 0.9-81-2 ***
```

Question 1

Let

$$A = \begin{pmatrix} 2 & 6 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

(a) For the following questions, you have to clearly show your work.

1. Find the eigenvalues and eigenvectors for A.
2. Find the square root of A using Cholesky decomposition.
3. Find the square root of A using spectral decomposition.

(b). Is A a positive definite matrix? Why or why not?

(c). Use any software to verify your answers in part (a).

Solution

Part A

Please refer to the handwritten solution

Part B

Yes A is a positive definite matrix because its' eigenvalues are strictly positive values. Hence, $\lambda_1 = 2, \lambda_2 = 3 + \sqrt{3}, \lambda_3 = 3 - \sqrt{3} > 0$.

Part C

```
A = matrix(c(2,1,0,1,4,1,0,1,2), nrow = 3,
           ncol = 3, byrow = TRUE)
```

Getting the eigen values and the eigen vectors

```
eigen(A)
```

```
## eigen() decomposition
## $values
## [1] 4.732051 2.000000 1.267949
##
## $vectors
##           [,1]          [,2]       [,3]
## [1,] 0.3250576  7.071068e-01  0.6279630
## [2,] 0.8880738 -3.140185e-16 -0.4597008
## [3,] 0.3250576 -7.071068e-01  0.6279630
```

Here is the Cholesky Decomposition

```
t(chol(A))
```

```
##           [,1]      [,2]     [,3]
## [1,] 1.4142136 0.0000000 0.000000
## [2,] 0.7071068 1.8708287 0.000000
## [3,] 0.0000000 0.5345225 1.309307
```

Finally, here is the Spectral Decomposition

```
ev = eigen(A)
L = ev$values
V = ev$vectors
D = diag(L)
sqrtD = sqrt(D)
sqrtD
```

```
##          [,1]     [,2]     [,3]
## [1,] 2.175328 0.000000 0.000000
## [2,] 0.000000 1.414214 0.000000
## [3,] 0.000000 0.000000 1.126033
```

```
sqrtA = V%*%sqrtD%*%t(V)
sqrtA
```

```
##             [,1]      [,2]        [,3]
## [1,]  1.38099412 0.3029054 -0.03321944
## [2,]  0.30290545 1.9535856  0.30290545
## [3,] -0.03321944 0.3029054  1.38099412
```

```
all.equal(A, zapsmall(sqrtA%*%t(sqrtA)) )
```

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

Question 2

Let A be a $(p \times p)$ matrix and is partitioned into

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

where A_{11} is a $(p_1 \times p_1)$ matrix, A_{22} is a $(p_2 \times p_2)$ matrix, and $p_1 + p_2 = p$. Similarly, let B be a $(p \times p)$ matrix and is partitioned into

$$A = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

where B_{11} is a $(p_1 \times p_1)$ matrix, B_{22} is a $(p_2 \times p_2)$ matrix, and $p_1 + p_2 = p$. Assume A_{11}, A_{22}, B_{11}, and B_{22} are non singular matrices.

(a) Denote I_p be the $(p \times p)$ identity matrix. Let $AB = I_p$. Express B_{ij} in terms of A_{ij} for all $i, j = 1, 2$.

(b) Let $BA = I_p$.Express B_{ij} in terms of A_{ij} for all $i, j = 1, 2$.

(c) Show that

$$|A| = |A_{22}||A_{11} - A_{12}A_{22}^{-1}A_{21}| = |A_{11}||A_{22} - A_{21}A_{11}^{-1}A_{12}|$$

(d) Show that

$$B_{11} = A_{11}^{-1} + A_{11}^{-1}A_{12}(A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1}A_{21}A_{11}^{-1}$$

and

$$B_{22} = A_{22}^{-1} + A_{22}^{-1}A_{21}(A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1}A_{12}A_{22}^{-1}$$

Solution

Part A

Please refer to the handwritten notes

Part B

Please refer to the handwritten notes

Part C

Please refer to the handwritten notes

Part D

Please refer to the handwritten notes

Question 4

Consider the following data set:

x1:	3	3	4	5	6	8
x2:	17.95	15.54	14.00	12.95	8.94	7.49

For the following questions, you have to clearly show your steps. Computer commanda and print out is not accepted.

(a) Find the sample mean vector.

(b) Find the sample unbiased variance matrix.

(c) Report the squared statistical distances $(x_j - \underset{\sim}{\overline{x}})' S^{-1} (x_j - \underset{\sim}{\overline{x}})$ for $j = 1, ..., 6$.

(d) Assume the data set is from a bi variate normal distribution.

1. Describe how you would estimate the 50% probability contour of the population mean vector.
2. At 5% level of significance, is there significant evidence that the population mean vector is different from $(3, 10)'$.

Solution

```
X = matrix(c(3, 17.95, 3, 15.54, 4, 14, 5, 12.95, 6, 8.94,
             8, 7.49), nrow = 6, ncol = 2, byrow = TRUE)
n = 6
p = 2
```

Part A

Please refer to the handwritten notes, but here is the optional R code as well.

```
vec1 = matrix(1, 6, 1)
xbar = 1/6*t(X)%*%vec1
xbar
```

```
##           [,1]
## [1,]  4.833333
## [2,] 12.811667
```

Part B

Please refer to the handwritten notes, but here is the optional R code as well.

```
M = t(X)%*%X
L = xbar%*%t(xbar)
N = 6*L
S = 1/5*(M-N)
S
```

```
##           [,1]      [,2]
## [1,]  3.766667 -7.351667
## [2,] -7.351667 15.717497
```

Part C

Please refer to the handwritten notes, but here is the optional R code as well.

```
S_inv = solve(S)
S_inv
```

```
##          [,1]      [,2]
## [1,] 3.048645 1.4259664
## [2,] 1.425966 0.7306017
```

```
vec1 = matrix(1, 6, 1)
r = X[,1] - xbar[1,]
t = X[,2] - xbar[2,]
centered_mat = cbind(r,t)
distance = centered_mat%*%S_inv%*%t(centered_mat)
diag(distance)
```

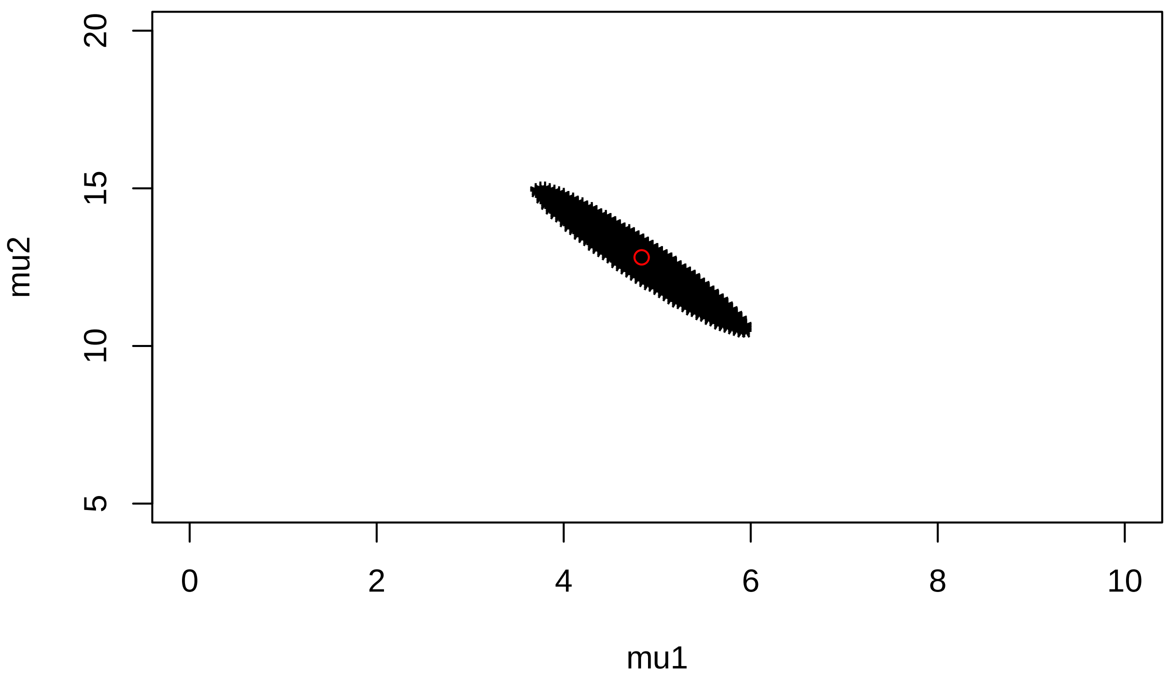
```
## [1] 2.6705244 1.4200802 0.3246180 0.1644184 2.2190979 3.2012612
```

Part D.1

```
plot(xbar[1], xbar[2], type="p", xlim=c(0, 10),
     ylim=c(5, 20), xlab="mu1", ylab="mu2")

mu1 = matrix(seq(-5, 20, 0.05), ncol=1, byrow=T)
nmu1 = nrow(mu1)
mu2 = matrix(seq(5, 20, 0.05), ncol=1, byrow=T)
nmu2 = nrow(mu2)


for (i in 1:nmu1) {
  for (j in 1:nmu2) {
    mu = matrix(c(mu1[i, 1], mu2[j, 1]), ncol=1, byrow=T)
    Fcomp = c((n-p)/((n-1)*2)*(n*t(xbar-mu)%*%solve(S)%*%(xbar-mu)))
    Fcrit = qf(0.50, p, n-p)
    if (Fcomp < Fcrit) points(mu1[i, 1], mu2[j, 1], pch="*")
  }
}
points(xbar[1], xbar[2], col='red')
```



Part D.2

Let

$$H_0 : \mu = \begin{pmatrix} 3 \\ 10 \end{pmatrix} \qquad H_a : \mu \neq \begin{pmatrix} 3 \\ 10 \end{pmatrix}$$

```
mu0 = matrix(c(3, 10), ncol=1, byrow=T)
Tobs = n*t(xbar-mu0)%*%S_inv%*%(xbar-mu0)
Tobs
```

```
##          [,1]
## [1,] 184.341
```

```
Fcriticalvalue = (n-1)*p/(n-p)*qf(p = 0.05, df1 = p, df2 = n-p, lower.tail = FALSE)
Fcriticalvalue
```

```
## [1] 17.36068
```

```
pvalue = 1-pf((n-p)/((n-1)*2)*Tobs, p, n-p)
pvalue
```

```
##              [,1]
## [1,] 0.0006973496
```

As we can see that since our observed Hotelling squared statistic is larger than the critical value, we can say that we will reject H_0 and say that there is evidence that the population mean is different from $\mu_0 = (3, 10)^{'}$.

Question 5

Data are given in the excel file.

(a) Using a graphical method to check if the data of East is a sample from the normal distribution. How about data of South, West, and North?

(b) Regardless of your result in part (a), obtain the 95% confidence interval for the mean of

$$(1) North \qquad (2) South \qquad (3) East \qquad (4) West$$

Clearly state the necessary assumptions needed for your

(c) Considering the data set as a multivariate data set. Use a software and report the sample mean vector, sample covariance matrix and sample correlation matrix.

(d) Use a graphical method to check if the data set is a sample from a multivariate normal distribution.

(e) Obtain the equation for obtaining the 95% confidence region for the population mean vector, $\underset{\sim}{\mu} = (\mu_N, \mu_S, \mu_E, \mu_W)'$. *(No calculations needed. Just the equations.)* Clearly state the necessary assumptions needed for your answer.

(f) At 5% level of significance, test

$$H_0 : \underset{\sim}{\mu} = (1450, 1900, 1700, 1700)' \quad \text{vs} \quad H_a : \underset{\sim}{\mu} \neq (1450, 1900, 1700, 1700)'.$$

(g) Based on the your answer in part (f), is $\underset{\sim}{\mu} = (1450, 1900, 1700, 1700)'$ falls within the 95% confidence region of $\underset{\sim}{\mu}$ obtained in part (e)? Why or why not?

Solution

Let it be known that the excel dataset is called df.

```
df = data.frame(df)
X = data.matrix(df)
n = dim(df)[1]
p = dim(df)[2]
```

Part A

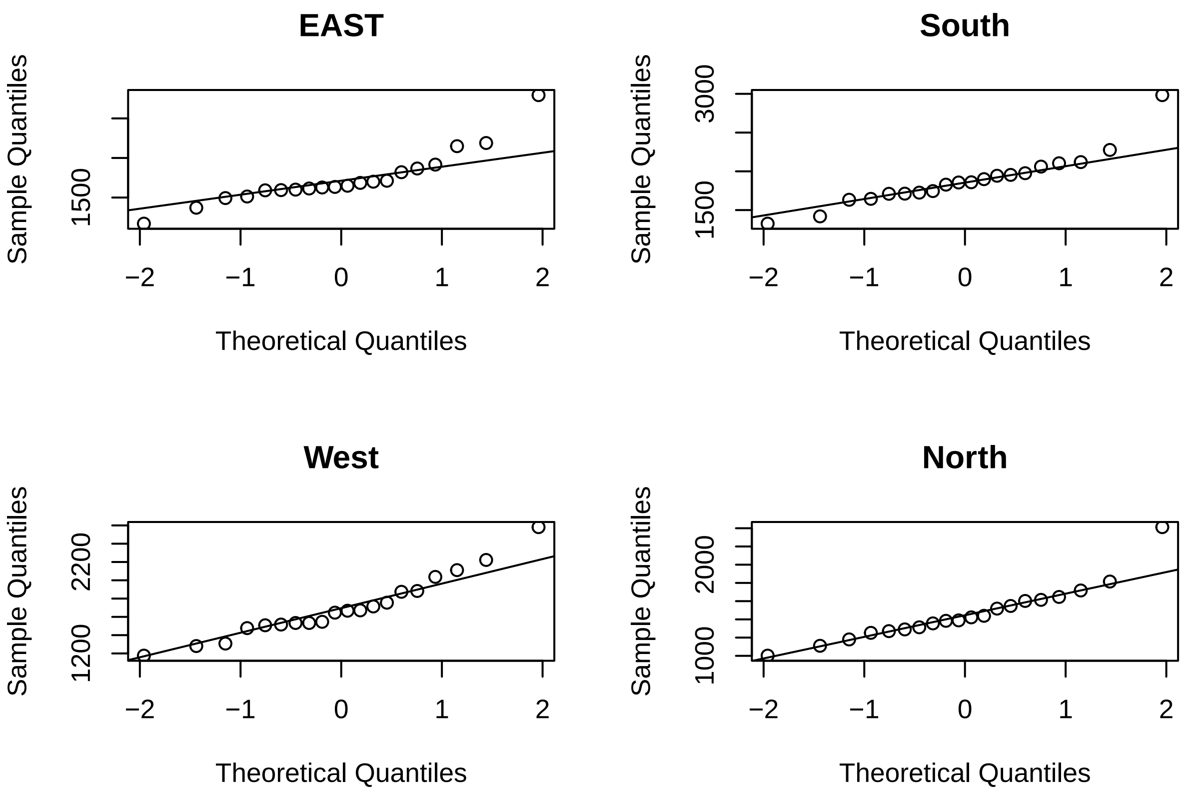
```
par(mfrow = c(2,2))

qqnorm(df$East, main = 'EAST')
qqline(df$East)

qqnorm(df$South, main = 'South')
qqline(df$South)

qqnorm(df$West, main = 'West')
qqline(df$West)

qqnorm(df$North, main = 'North')
qqline(df$North)
```



I would advise that the sample from East is not from a normal distribution, however the rest of the direction variables does appear to be normally distributed based off the above QQ-plots.

Part B

Our assumptions are that the data from each direction is normally distributed and the variance is unknown.

```
onemat = matrix(1, n, 1)
xbar =1/n*t(X)%*%onemat
xbar
```

```
##          [,1]
## North 1463.95
## South 1888.60
## East  1734.40
## West  1701.95
```

```
alpha = 0.05
degrees.freedom = n - 1
t.score= qt(p = alpha/2, df = degrees.freedom, lower.tail = F)
t.score
```

```
## [1] 2.093024
```

North C.I.

```
sample.sd = sd(df$North)
sample.se = sample.sd/sqrt(n)
sample.se
```

```
## [1] 67.8844
```

```
lower_bound = xbar[1] - t.score*sample.se
upper_bound = xbar[1] + t.score*sample.se
c(lower_bound, upper_bound)
```

```
## [1] 1321.866 1606.034
```

Therefore the **C.I.for the mean of North** would be **(1321.866, 1606.034)**.

South C.I.

```
sample.sd = sd(df$South)
sample.se = sample.sd/sqrt(n)
sample.se
```

```
## [1] 77.30495
```

```
lower_bound = xbar[2] - t.score*sample.se
upper_bound = xbar[2] + t.score*sample.se
c(lower_bound, upper_bound)
```

```
## [1] 1726.799 2050.401
```

Therefore the **C.I. for the mean of South** would be **(1726.799, 2050.401)**.

East C.I.

```
sample.sd = sd(df$East)
sample.se = sample.sd/sqrt(n)
sample.se
```

```
## [1] 76.48465
```

```
lower_bound = xbar[3] - t.score*sample.se
upper_bound = xbar[3] + t.score*sample.se
c(lower_bound, upper_bound)
```

```
## [1] 1574.316 1894.484
```

Therefore the **C.I. for the mean of East** would be **(1574.315, 1894.484)**.

West C.I.

```
sample.sd = sd(df$West)
sample.se = sample.sd/sqrt(n)
sample.se
```

```
## [1] 76.20324
```

```
lower_bound = xbar[4] - t.score*sample.se
upper_bound = xbar[4] + t.score*sample.se
c(lower_bound, upper_bound)
```

```
## [1] 1542.455 1861.445
```

Therefore the **C.I. for the mean of West** would be **(1542.455, 1861.445)**.

Part C

Sample Mean Vector

```
xbar =1/n*t(X)%*%onemat
xbar
```

```
##          [,1]
## North 1463.95
## South 1888.60
## East  1734.40
## West  1701.95
```

Sample Variance-Covariance Matrix

```
M = t(X)%*%X
L = xbar%*%t(xbar)
N = n*L
S = 1/(n - 1)*(M-N)
S
```

```
##          North     South      East      West
## North 92165.84  91525.08  76724.18  93988.10
## South 91525.08 119521.09 108840.91 103275.98
## East  76724.18 108840.91 116998.04  85358.18
## West  93988.10 103275.98  85358.18 116138.68
```

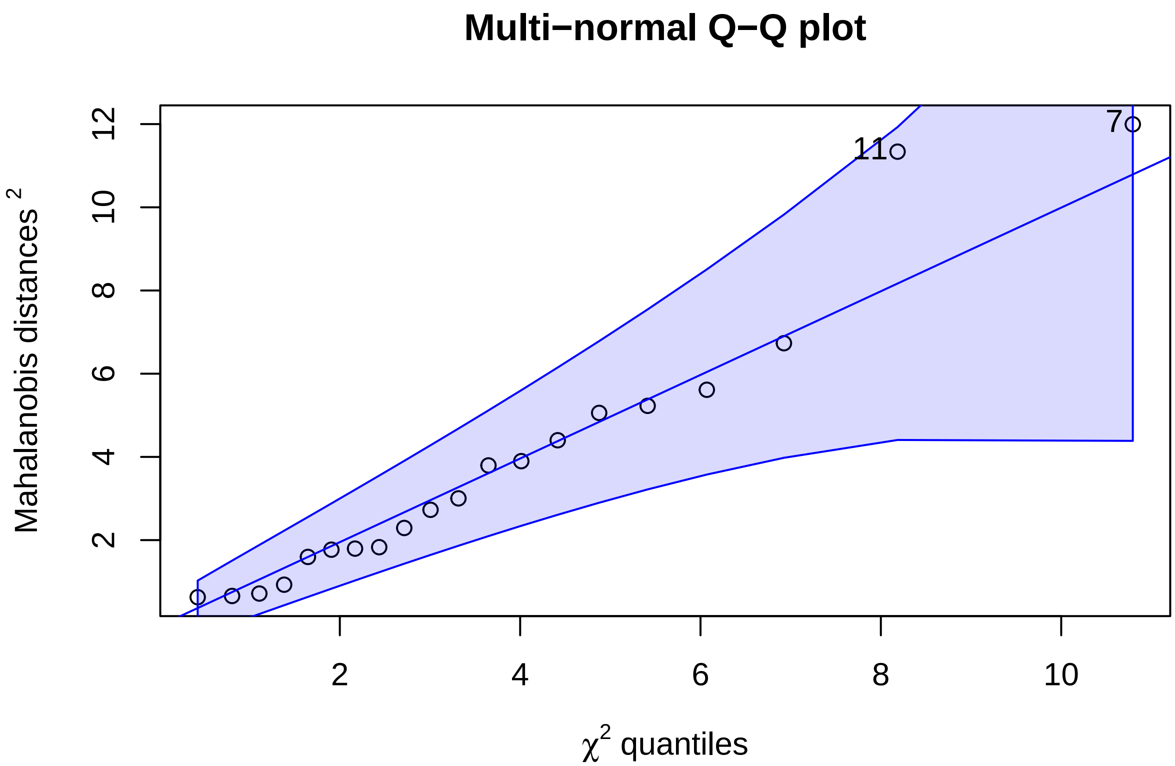
Sample Correlation Matrix

```
variances = diag(S)
D = matrix(diag(variances),ncol=4)
D_sqrt = sqrt(D)
D_sqrt_inv = solve(D_sqrt)
samp_cor = D_sqrt_inv%*%S%*%D_sqrt_inv
samp_cor
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] 1.0000000 0.8720329 0.7388529 0.9084467
## [2,] 0.8720329 1.0000000 0.9204084 0.8765740
## [3,] 0.7388529 0.9204084 1.0000000 0.7322635
## [4,] 0.9084467 0.8765740 0.7322635 1.0000000
```

Part D

```
library(RVAideMemoire)
mqqnorm(X, main = 'Multi-normal Q-Q plot')
```




```
## [1]  7 11
```

Part E

Please check the handwritten notes. Here is the R code for retrieving the S^{-1}. Our assumptions are that the data is multivariate normally distributed and the variance-covariance matrix is unknown.

```
S_inv = solve(S)
S_inv
```

```
##               North         South          East          West
## North  7.349858e-05 -3.158687e-05  8.816218e-06 -3.787165e-05
## South -3.158687e-05  1.435897e-04 -8.270559e-05 -4.133832e-05
## East   8.816218e-06 -8.270559e-05  6.738748e-05  1.688334e-05
## West  -3.787165e-05 -4.133832e-05  1.688334e-05  6.361024e-05
```

Part F

```
mu0 = matrix(c(1450,1900,1700,1700), ncol=1, byrow=T)
Tobs = n*t(xbar-mu0)%*%S_inv%*%(xbar-mu0)
Tobs
```

```
##          [,1]
## [1,] 3.967353
```

```
Fcriticalvalue = (n-1)*p/(n-p)*qf(p = 0.05, df1 = p, df2 = n-p, lower.tail = FALSE)
Fcriticalvalue
```

```
## [1] 14.28286
```

```
pvalue = pf((n-p)/((n-1)*p)*Tobs, p, n-p,lower.tail = FALSE)
pvalue
```

```
##           [,1]
## [1,] 0.5224764
```

Part G

Based of the R code for Part E, since the p-value is greater than 0.05, we would say that the vector $\underset{\sim}{\mu} = (1450, 1900, 1700, 1700)^{'}$ would fall within the 95% confidence region. Furthermore, we can say that since the Hotelling T^2 observed statistic is not greater than the F critical value, we cannot reject H_0 and we shall say that there is no evidence to show that population mean vector is different from $\underset{\sim}{\mu} = (1450, 1900, 1700, 1700)^{'}$.