STAT 8121

Assignment 1

Student id: 47541164

Question 1 :-

Ans a) $Y = AX$

$$\text{Mean}(Y) = A\ \text{Mean}(X)$$

$$X = \begin{bmatrix} \underbrace{\begin{matrix}1\\4\end{matrix}}_{\substack{\downarrow\\ \text{mean}\\ (\mu)}} & \underbrace{\begin{matrix}2\\3\end{matrix}}_{\substack{\downarrow\\ \text{variance}\\ (\sigma^2)}} \end{bmatrix}$$

$$\text{Mean}(X) = \begin{bmatrix} 1 \\ 4 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 1 \\ 2 & -1 \end{bmatrix}$$

$$\text{Mean}(Y) = \underset{2\times2}{\begin{bmatrix} 1 & 1 \\ 2 & -1 \end{bmatrix}} \underset{2\times1}{\begin{bmatrix} 1 \\ 4 \end{bmatrix}} \qquad \boxed{2\times2 \quad 2\times1 = 2\times1}$$

$$= \begin{bmatrix} 1+4 \\ 2+(-4) \end{bmatrix}$$

$$= \begin{bmatrix} 5 \\ -2 \end{bmatrix}$$

$$\therefore \text{Mean}(Y) = \underset{2\times1}{\begin{bmatrix} 5 \\ -2 \end{bmatrix}}$$

Question 1 :-

Answer b)

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ \underbrace{4}_{\substack{\downarrow \\ \text{mean} \\ (\mu)}} & \underbrace{3}_{\substack{\downarrow \\ \text{variance} \\ (\sigma^2)}} \end{bmatrix}$$

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

$$\text{var-cov}(Y) = A * \text{var-cov}(X) * A^T$$

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

$$= \begin{bmatrix} 2+1 & 1+3 \\ 4+(-1) & 2+(-3) \end{bmatrix} * \begin{bmatrix} 1 & 2 \\ 1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 4 \\ 3 & -1 \end{bmatrix} * \begin{bmatrix} 1 & 2 \\ 1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 3+4 & 6-4 \\ 3-1 & 6+1 \end{bmatrix}$$

$$= \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix}$$

Question 1 :-

Answer c)

$$C_y = \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix}$$

$$\text{Identity matrix } (I) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$|C_y - \lambda I| = 0$$

For 2×2 matrix

$$\lambda^2 - S_1\lambda + |A| = 0$$

$S_1 \rightarrow$ sum of diagnonals

$$|A| = \begin{vmatrix} 7 & 2 \\ 2 & 7 \end{vmatrix} = 49 - 4 = 45$$

$$\lambda^2 - 14\lambda + 45 = 0$$

$$\lambda^2 - 9\lambda - 5\lambda + 45 = 0$$

$$\lambda(\lambda - 9) - 5(\lambda - 9) = 0$$

$$(\lambda - 5)(\lambda - 9) = 0$$

$$\lambda - 5 = 0 \quad \text{OR} \quad \lambda - 9 = 0$$

45

$-9 \quad -5$

$$\lambda - 9 = 0 \quad \text{OR} \quad \lambda - 5 = 0$$

$$\lambda = 9 \quad \text{OR} \quad \lambda = 5$$

∴ Eigenvalues = ~~5~~, 9

= 9 , 5

For $\lambda = 9$

$$\begin{bmatrix} 7-\lambda & 2 \\ 2 & 7-\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$= \begin{bmatrix} 7-9 & 2 \\ 2 & 7-9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$= \begin{bmatrix} -2 & 2 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

According to Row-Echelon Form

1> Divide Row1 (R_1) by -2

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

~~2> multiply Row2 R_2 by~~
2> Row2 $(R_2) - 2R_1$

$$\begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$1x_1 - 1x_2 = 0$; $1x_1 = 1x_2$

if we take $x_2 = t$, then $x_1 = t$

thus $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ } eigenvector for $\lambda = 9$

for $\lambda = 5$

$$\begin{bmatrix} 7-\lambda & 2 \\ 2 & 7-\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$= \begin{bmatrix} 7-5 & 2 \\ 2 & 7-5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$= \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

According to Row-Echelon Form

1) Divide Row1 (R_1) by 2

$$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

2) Row 2 $(R_2) - 2R_1$

$$\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$1x_1 + 1x_2 = 0$; $1x_1 = -1x_2$

if we take $x_2 = t$, then $x_1 = -t$

thus $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \Big\}$ eigenvector for $\lambda = 5$

Question 1 :-

Answer d)

formula for SVD,

$$A = U \Sigma V^T$$

$$A = \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix}$$

From previous calculations
we know the eigenvalues are 9, 5
the eigenvectors are $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$

$$V = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

$$A A^T = \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix} \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix} = \begin{bmatrix} 53 & 28 \\ 28 & 53 \end{bmatrix}$$

$$|A A^T - \lambda I| = 0$$

$$\begin{bmatrix} 53-\lambda & 28 \\ 28 & 53-\lambda \end{bmatrix} = 0$$

Determinent = $((53-\lambda)(53-\lambda)) - (28 \times 28) = 0$

$$= 2809 - 53\lambda - 53\lambda + \lambda^2 - 784 = 0$$
$$= 2025 - 106\lambda + \lambda^2 = 0$$

$\cancel{\lambda^2}$

$\lambda_1 = 81$ OR $\lambda_2 = 25$

$$\sigma_1 = \sqrt{\lambda_1} = \sqrt{81} = 9$$
$$\sigma_2 = \sqrt{\lambda_2} = \sqrt{25} = 5$$

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} = \begin{bmatrix} 9 & 0 \\ 0 & 5 \end{bmatrix}$$

For $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$, unit vector $= \sqrt{(1)^2 + (1)^2} = \sqrt{2}$

For $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$, unit vector $= \sqrt{(-1)^2 + (1)^2} = \sqrt{2}$

$$\therefore \quad V = \begin{bmatrix} 1 \times \frac{1}{\sqrt{2}} & -1 \times \frac{1}{\sqrt{2}} \\ 1 \times \frac{1}{\sqrt{2}} & 1 \times \frac{1}{\sqrt{2}} \end{bmatrix}$$

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

$$A = U \Sigma V^T$$

$$\therefore \quad U\Sigma = AV$$

$$AV = \begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 9/\sqrt{2} & -5\sqrt{2} \\ 9/\sqrt{2} & 5/\sqrt{2} \end{bmatrix}$$

$$\sigma_1 \vec{u_1} = 9 \begin{bmatrix} 9/\sqrt{2} \\ 9/\sqrt{2} \end{bmatrix}$$

$$= \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

$$\sigma_2 \vec{u_2} = 5 \begin{bmatrix} -5/\sqrt{2} \\ 5/\sqrt{2} \end{bmatrix}$$

$$= \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

$$\therefore \quad U = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

Now, $A = U \Sigma V^T$

$$\begin{bmatrix} 7 & 2 \\ 2 & 7 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 9 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

Question 2 :-

Answer a) $\bar{x} = [\bar{x}_1 , \bar{x}_2 , \bar{x}_3]$

Sample Mean Vector is

$$\bar{x} = \begin{bmatrix} \bar{x}_{.1} \\ \vdots \\ \bar{x}_{.p} \end{bmatrix} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

∴ By getting mean of each calumn we get

$$\bar{x} = [5.66 , 5.16 , 8]^T$$

$$\bar{x} = \begin{bmatrix} 5.66 \\ 5.16 \\ 8 \end{bmatrix}$$

Question 2 :-

Answer b) Calculate $1\bar{x}^T$

$$\bar{x} = \begin{bmatrix} 5.66 \\ 5.16 \\ 8 \end{bmatrix}$$

$$X = \begin{bmatrix} 2 & 3 & 15 \\ 6 & 8 & 9 \\ 5 & 2 & 7 \\ 9 & 4 & 3 \\ 11 & 10 & 2 \\ 1 & 4 & 12 \end{bmatrix}$$

$$1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$1 \cdot x^T = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 5.66 & 5.16 & 8 \end{bmatrix}$$

$$\therefore \quad 1 \cdot x^T = \begin{bmatrix} 5.66 & 5.16 & 8 \\ 5.66 & 5.16 & 8 \\ 5.66 & 5.16 & 8 \\ 5.66 & 5.16 & 8 \\ 5.66 & 5.16 & 8 \\ 5.66 & 5.16 & 8 \end{bmatrix}$$

$$x - 1 \cdot x^T = \begin{bmatrix} -3.66 & -2.16 & 7 \\ 0.34 & 2.84 & 1 \\ -0.66 & -3.16 & -1 \\ 3.34 & -1.16 & -5 \\ 5.34 & 4.84 & -6 \\ -4.66 & -1.16 & 4 \end{bmatrix}$$

Question 2 :-

Answer d)

$$SSCP = (X - 1\bar{x}^T)^T \cdot (X - 1\bar{x}^T)$$

$$SSCP = \begin{bmatrix} -3.66 & 0.34 & -0.66 & 3.34 & 5.34 & -4.66 \\ -2.16 & 2.84 & -3.16 & -1.16 & 4.84 & -1.16 \\ 7 & 1 & -1 & -5 & -6 & 4 \end{bmatrix} \cdot \begin{bmatrix} -3.66 & -2.16 & 7 \\ 0.34 & 2.84 & 1 \\ -0.66 & -3.16 & -1 \\ 3.34 & -1.16 & -5 \\ 5.34 & 4.84 & -6 \\ -4.66 & -1.16 & 4 \end{bmatrix}$$

$$= \begin{bmatrix} 75.3336 & 38.3336 & -92 \\ 38.3336 & 48.8336 & -37 \\ -92 & -37 & 128 \end{bmatrix}$$

Question 2:-

Answer e)

S, Sample covariance matrix $= \frac{1}{n-1}$ (SSCP)

Here, $n = 6$

SSCP is calculated in Answer 2.d

$$\therefore S = \frac{1}{6-1} \begin{bmatrix} 75.3336 & 38.3336 & -92 \\ 38.3336 & 48.8336 & -37 \\ -92 & -37 & 128 \end{bmatrix}$$

$$S = \begin{bmatrix} \frac{75.3336}{5} & \frac{38.3336}{5} & \frac{-92}{5} \\ \frac{38.3336}{5} & \frac{48.8336}{5} & \frac{-37}{5} \\ \frac{-92}{5} & \frac{-37}{5} & \frac{128}{5} \end{bmatrix}$$

$$= \begin{bmatrix} 15.066 & 7.660 & -18.4 \\ 7.66 & 9.76 & -7.4 \\ -18.4 & -7.4 & 25.6 \end{bmatrix}$$

Question 2 :-

Answer f)

The eigenvalues from previous answer 2.e the eigenvalues of S (sample covariance matrix) are :

42.825 , 0.745 , 6.854

For the matrix to be positive definite, all of its eigenvalues must be greater than zero.

In this case all three eigenvalues are greater than zero.

Hence, the sample covariance matrix S is positive definite.

Question 2 :-

Answer g)

To calculate R, sample correlation matrix we follow the following formula

$$r_{jk} = \frac{\sum_{i=1}^{n} (x_{ij} - \bar{x}_{\cdot j})(x_{ik} - \bar{x}_{\cdot k})}{\sqrt{\sum_{i=1}^{n} (x_{ij} - \bar{x}_j)^2} \cdot \sqrt{\sum_{i=1}^{n} (x_{ik} - \bar{x}_{\cdot k})^2}}$$

$$= \frac{S_{jk}}{\sqrt{S_{jj}} \sqrt{S_{kk}}}$$

where,

$S_{jk} \rightarrow$ sample covariance between x_j & x_k

$S_{jj} \Rightarrow$ sample variance of x_j.

Also,

$r_{jj} = 1$.

For calculations, result & code see following page

Question 2 :-

Answer g)

To calculate R, sample correlation matrix

we follow the following formula

$$r_{jk} = \frac{\sum_{i=1}^{n} (x_{ij} - \bar{x}_{\cdot j})(x_{ik} - \bar{x}_{\cdot k})}{\sqrt{\sum_{i=1}^{n} (x_{ij} - \bar{x}_{j})^2} \cdot \sqrt{\sum_{i=1}^{n} (x_{ik} - \bar{x}_{\cdot k})^2}}$$

$$= \frac{s_{jk}}{\sqrt{s_{jj}} \sqrt{s_{kk}}}$$

where,

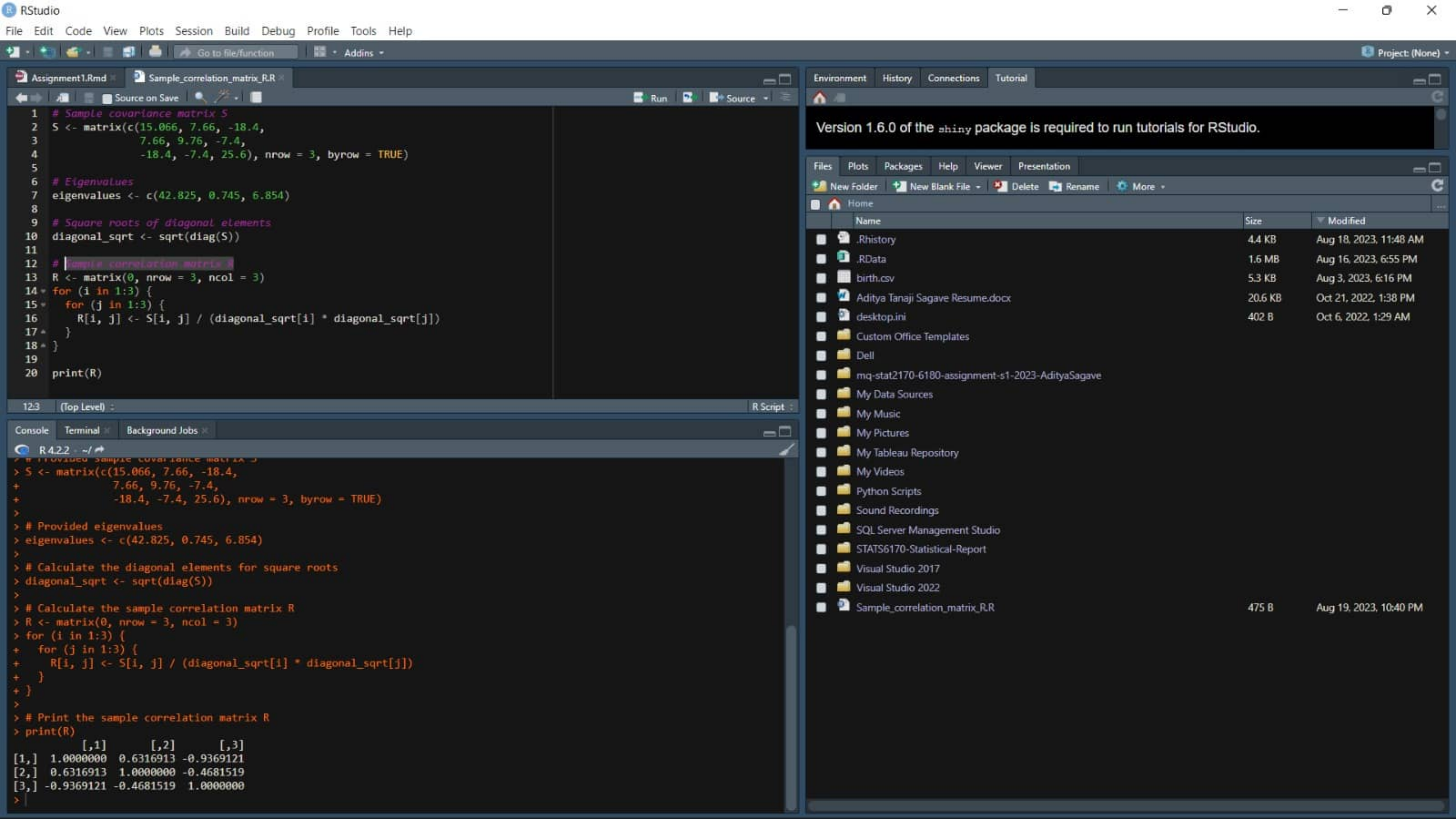
$s_{jk} \rightarrow$ sample covariance between x_j & x_k

$s_{jj} \Rightarrow$ sample variance of x_j.

Also,

$r_{jj} = 1$.

For calculations, result & code see following page



```r
# Sample covariance matrix S
S <- matrix(c(15.066, 7.66, -18.4,
              7.66, 9.76, -7.4,
              -18.4, -7.4, 25.6), nrow = 3, byrow = TRUE)

# Eigenvalues
eigenvalues <- c(42.825, 0.745, 6.854)

# Square roots of diagonal elements
diagonal_sqrt <- sqrt(diag(S))

# Sample correlation matrix R
R <- matrix(0, nrow = 3, ncol = 3)
for (i in 1:3) {
  for (j in 1:3) {
    R[i, j] <- S[i, j] / (diagonal_sqrt[i] * diagonal_sqrt[j])
  }
}

print(R)
```

```
> S <- matrix(c(15.066, 7.66, -18.4,
+               7.66, 9.76, -7.4,
+               -18.4, -7.4, 25.6), nrow = 3, byrow = TRUE)
>
> # Provided eigenvalues
> eigenvalues <- c(42.825, 0.745, 6.854)
>
> # Calculate the diagonal elements for square roots
> diagonal_sqrt <- sqrt(diag(S))
>
> # Calculate the sample correlation matrix R
> R <- matrix(0, nrow = 3, ncol = 3)
> for (i in 1:3) {
+   for (j in 1:3) {
+     R[i, j] <- S[i, j] / (diagonal_sqrt[i] * diagonal_sqrt[j])
+   }
+ }
>
> # Print the sample correlation matrix R
> print(R)
           [,1]       [,2]       [,3]
[1,]  1.0000000  0.6316913 -0.9369121
[2,]  0.6316913  1.0000000 -0.4681519
[3,] -0.9369121 -0.4681519  1.0000000
>
```

Version 1.6.0 of the `shiny` package is required to run tutorials for RStudio.

Name	Size	Modified
.Rhistory	4.4 KB	Aug 18, 2023, 11:48 AM
.RData	1.6 MB	Aug 16, 2023, 6:55 PM
birth.csv	5.3 KB	Aug 3, 2023, 6:16 PM
Aditya Tanaji Sagave Resume.docx	20.6 KB	Oct 21, 2022, 1:38 PM
desktop.ini	402 B	Oct 6, 2022, 1:29 AM
Custom Office Templates		
Dell		
mq-stat2170-6180-assignment-s1-2023-AdityaSagave		
My Data Sources		
My Music		
My Pictures		
My Tableau Repository		
My Videos		
Python Scripts		
Sound Recordings		
SQL Server Management Studio		
STATS6170-Statistical-Report		
Visual Studio 2017		
Visual Studio 2022		
Sample_correlation_matrix_R.R	475 B	Aug 19, 2023, 10:40 PM

Question 3

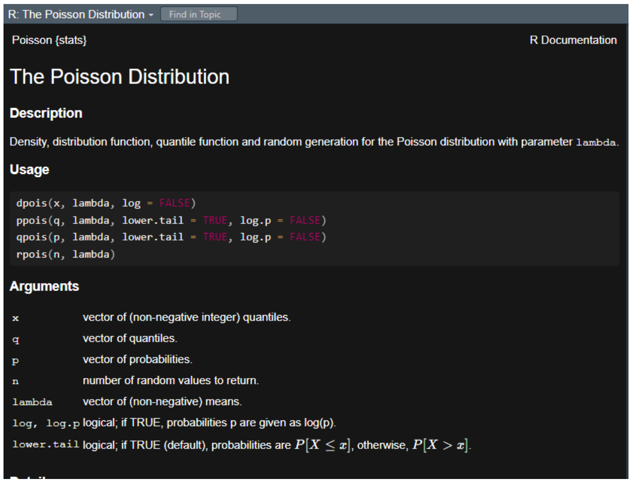
Answer 3.a:

Given code for Matrix A:

```
rm(list=ls())
A <- matrix(rpois(64,5), 8, 8)
A
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    8    3    3   11    7    4    5    3
## [2,]    6    6    8    6    7    5    8    1
## [3,]    5    6    3    6    3    8    4    4
## [4,]    5    6    4    3    6    5    4    3
## [5,]    6    6    7    5    6    8    6    6
## [6,]    2    0    8    3    7    5    4    2
## [7,]    9    4    9    6    2    7    5    5
## [8,]    5    6    3    3    4    5    6    2
```

After looking for rpois() help using ?rpois we see the following



Poisson {stats} R Documentation

The Poisson Distribution

Description

Density, distribution function, quantile function and random generation for the Poisson distribution with parameter `lambda`.

Usage

```
dpois(x, lambda, log = FALSE)
ppois(q, lambda, lower.tail = TRUE, log.p = FALSE)
qpois(p, lambda, lower.tail = TRUE, log.p = FALSE)
rpois(n, lambda)
```

Arguments

`x`	vector of (non-negative integer) quantiles.
`q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of random values to return.
`lambda`	vector of (non-negative) means.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X \le x]$, otherwise, $P[X > x]$.

Figure 1: ?rpois

Here are things happening in the give code:

1. rm(list=ls()): This line clears the workspace, removing any existing variables.
2. A <- matrix(rpois(64, 5), 8, 8): This line generates a matrix A with dimensions 8 x 8.
3. It uses the rpois() function, which generates random values from the Poisson distribution. The function takes two arguments: the number of values to generate (in this case, 64) and the mean parameter (in this case, 5). So, each element of matrix A is obtained by generating a random Poisson-distributed value with a mean of 5.

Question 3

Answer 3.b:

Here is the given code for matric C:

```
C= A %*% t(A)
C
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]  302  268  218  194  264  168  259  184
## [2,]  268  311  223  218  294  202  280  211
## [3,]  218  223  211  177  247  137  234  172
## [4,]  194  218  177  172  227  140  205  161
## [5,]  264  294  247  227  318  201  299  214
## [6,]  168  202  137  140  201  171  187  124
## [7,]  259  280  234  205  299  187  317  197
## [8,]  184  211  172  161  214  124  197  160
```

To generate eigen values and eigenvectors of matrix:

```
# eigenvalues and eigenvectors
eigen_result <- eigen(C)
```

Eigenvalues:

```
print(eigen_result$values)
```

```
## [1] 1764.498447   66.799644   57.736922   42.401697   21.153744    6.431771
## [7]    1.876005    1.101770
```

Eigenvectors:

```
print(eigen_result$vectors)
```

```
##            [,1]        [,2]       [,3]         [,4]       [,5]        [,6]
## [1,] -0.3776107  0.59472801  0.6682629  0.115693133 -0.1170024  0.12939585
## [2,] -0.4067870 -0.22374582  0.2383490 -0.328605293  0.5724334 -0.31311857
## [3,] -0.3286263  0.32079381 -0.3626590 -0.017435116 -0.3891652 -0.63092871
## [4,] -0.3024482 -0.02492948 -0.1408745 -0.373456584 -0.1346622  0.62246438
## [5,] -0.4183882 -0.13606119 -0.2719944  0.002198289 -0.3455004  0.22660064
## [6,] -0.2686620 -0.68260834  0.3800344  0.121894933 -0.3914089 -0.16171263
## [7,] -0.4025903 -0.03724359 -0.2722184  0.738012593  0.3989418  0.14346655
## [8,] -0.2883703  0.08277910 -0.2296459 -0.423478526  0.2394180 -0.03998827
##             [,7]        [,8]
## [1,]  0.11306766  0.02305091
## [2,] -0.22643514 -0.37772254
## [3,] -0.31956370  0.07502656
## [4,] -0.55440285  0.18875863
## [5,]  0.43175276 -0.61263698
## [6,]  0.03113688  0.34965796
## [7,] -0.08229062  0.17680382
## [8,]  0.57646133  0.53543551
```

Question 3

Answer 3.c:

Since all of the eigenvalues of matrix C are positive, we can conclude that matrix C is positive definite.

Question 3

Answer 3.d:

Transferring the values of eigenvectors in a variable U

```
U <- eigen_result$vectors
```

```
# Creating U transpose
U_transpose <- t(U)

#U*U_transpose product
U_product <- U %*% U_transpose

# Identity matrix of the same size as U_product
I <- diag(nrow(U_product))

# View the product
U_product
```

```
##               [,1]          [,2]          [,3]          [,4]          [,5]
## [1,]  1.000000e+00  8.326673e-17 -1.322727e-17  5.039372e-16  1.037365e-15
## [2,]  8.326673e-17  1.000000e+00 -5.620504e-16 -2.498002e-16  8.326673e-17
## [3,] -1.322727e-17 -5.620504e-16  1.000000e+00  4.666406e-16  9.714451e-17
## [4,]  5.039372e-16 -2.498002e-16  4.666406e-16  1.000000e+00  2.775558e-17
## [5,]  1.037365e-15  8.326673e-17  9.714451e-17  2.775558e-17  1.000000e+00
## [6,]  8.812395e-16  9.159340e-16 -1.346145e-15 -2.081668e-16 -5.551115e-16
## [7,]  8.361367e-16 -1.804112e-16  9.194034e-17 -3.191891e-16 -1.387779e-16
## [8,]  3.295975e-16 -1.110223e-16  5.134781e-16 -1.526557e-16 -1.665335e-16
##               [,6]          [,7]          [,8]
## [1,]  8.812395e-16  8.361367e-16  3.295975e-16
## [2,]  9.159340e-16 -1.804112e-16 -1.110223e-16
## [3,] -1.346145e-15  9.194034e-17  5.134781e-16
## [4,] -2.081668e-16 -3.191891e-16 -1.526557e-16
## [5,] -5.551115e-16 -1.387779e-16 -1.665335e-16
## [6,]  1.000000e+00 -8.187895e-16 -8.326673e-16
## [7,] -8.187895e-16  1.000000e+00  1.387779e-16
## [8,] -8.326673e-16  1.387779e-16  1.000000e+00
```

By looking at the product matrix all the diagonal elements are 1 and non-diagonal elements are close to zero (the values are significantly low). Hence we have verified that $U^T * U = I$ Where I is the unit matrix.

Question 3

Answer 3.e:

Since it isn't clearly mentioned on which matrix should we apply SVD on I am applying SVD on both matrix A and Matrix C

For matrix A:

```
# Calculate the SVD
svd_result <- svd(A)

# Extract the matrices U, D, and V from svd_result
U <- svd_result$u
D <- diag(svd_result$d)  # Create a diagonal matrix from singular values
V <- svd_result$v

# Verify that A = U * D * t(V)
reconstructed_A <- U %*% D %*% t(V)

# Check if the reconstructed_C is close to the original C (within a tolerance)
is_equal <- all.equal(A, reconstructed_A)
```

```
# Print the result of verification
if (is_equal) {
  cat("SVD verification: Succeeded\n")
} else {
  cat("SVD verification: Failed\n")
}
```

```
## SVD verification: Succeeded
```

For Matrix C:

```
# Calculate the SVD
svd_result <- svd(C)

# Extract the matrices U, D, and V from svd_result
U <- svd_result$u
D <- diag(svd_result$d)  # Create a diagonal matrix from singular values
V <- svd_result$v

# Verify that C = U * D * t(V)
reconstructed_C <- U %*% D %*% t(V)

# Check if the reconstructed_C is close to the original C (within a tolerance)
is_equal <- all.equal(C, reconstructed_C)

# Print the result of verification
if (is_equal) {
  cat("SVD verification: Succeeded\n")
} else {
  cat("SVD verification: Failed\n")
}
```

```
## SVD verification: Succeeded
```