# STAT 32950 Assignment 2

Bin Yu

Apr 7, 2025

## Question 1

### (a)

To analyze the data in `ladyrun25.dat` (with the nominal variable "Country" removed) by scaling all numerical variables to have unit variance. This is equivalent to performing PCA on the correlation matrix. The R code used is:

```
data <- ladyrun[, !(names(ladyrun) %in% c("Country"))]
summary(princomp(data, cor = TRUE), loading = TRUE)
```

The output produced is summarized as follows:

```
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4    Comp.5      Comp.6      Comp.7
Standard deviation     2.387740 0.8584325 0.53721402 0.33824062 0.2934077 0.225255770 0.148174712
Proportion of Variance 0.814472 0.1052723 0.04122842 0.01634382 0.0122983 0.007248595 0.003136535
Cumulative Proportion  0.814472 0.9197443 0.96097276 0.97731657 0.9896149 0.996863465 1.000000000

Loadings:
         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7
100m      0.372  0.458  0.149  0.526  0.155  0.568
200m      0.374  0.480         0.111        -0.750 -0.204
400m      0.375  0.331 -0.487 -0.508 -0.456  0.200
800m      0.395 -0.221 -0.148 -0.377  0.769  0.121 -0.156
1500m     0.396 -0.231  0.425 -0.140        -0.143  0.750
3000m     0.383 -0.318  0.477        -0.377  0.140 -0.598
Marathon  0.349 -0.497 -0.553  0.534 -0.137 -0.146
```

Thus, the first two principal components are expressed as the following linear combinations of the *scaled* variables:

$$\begin{aligned}\text{PC1} = {} & 0.372\,(\text{scaled 100m}) + 0.374\,(\text{scaled 200m}) + 0.375\,(\text{scaled 400m}) \\ & + 0.395\,(\text{scaled 800m}) + 0.396\,(\text{scaled 1500m}) \\ & + 0.383\,(\text{scaled 3000m}) + 0.349\,(\text{scaled Marathon}),\end{aligned}$$

$$\begin{aligned}\text{PC2} = {} & 0.458\,(\text{scaled 100m}) + 0.480\,(\text{scaled 200m}) + 0.331\,(\text{scaled 400m}) \\ & - 0.221\,(\text{scaled 800m}) - 0.231\,(\text{scaled 1500m}) \\ & - 0.318\,(\text{scaled 3000m}) - 0.497\,(\text{scaled Marathon}).\end{aligned}$$

**Interpretation:**

- **Scaled Variables (X):** The principal components consist some scaled variables. The original performance measures (record times for 100m, 200m, 400m, 800m, 1500m, 3000m, and Marathon) have been standardized to create the scaled variables, denoted as $X_1, X_2, \ldots, X_7$. Each $X_i$ represents the number of standard deviations by which an observation deviates from the mean performance in that event. In short, the scaled variables represent the performance records (normalized to have mean 0 and variance 1). Higher value means that this observation use less time to finish 100m, 200m, 400m, 800m, 1500m, 3000m, and Marathon, and vice versa.
- **PC1:** All the coefficients in PC1 are positive and are of similar magnitude. This indicates that PC1 captures a general or overall performance factor. In other words, athletes who tend to perform well (or poorly) across all events will have high (or low) PC1 scores.
- **PC2:** The loadings for PC2 exhibit a clear pattern: the sprint events (100m, 200m, 400m) have positive loadings while the longer events (800m, 1500m, 3000m, Marathon) have negative loadings. This suggests that PC2 contrasts short-distance (sprint) performance with long-distance (endurance) performance. A high (more positive) PC2 score would indicate a relatively higher record in short-distance and lower record in long-distance events (or vice versa).
- **Uniqueness Up to a Sign:** Recall that each principal component is unique only up to multiplication by $\pm 1$. Thus, the sign of all loadings could be reversed without changing the underlying interpretation.

## (b)

perform an eigen-decomposition of the sample correlation matrix using

```
eigen(cor(data))$vectors
```

output:

```
     [,1]       [,2]        [,3]        [,4]        [,5]       [,6]        [,7]
[1,] -0.3720342 -0.4575195 -0.14870245  0.52629124 -0.15450205  0.5677425  0.08348107
[2,] -0.3738784 -0.4801563 -0.07423786  0.11131548 -0.09164471 -0.7495258 -0.20389904
[3,] -0.3747904 -0.3314811  0.48724807 -0.50849863  0.45647911  0.1996520  0.07373480
[4,] -0.3949123  0.2210770  0.14789147 -0.37710528 -0.76947015  0.1212119 -0.15592393
[5,] -0.3956582  0.2305757 -0.42485979 -0.13992068  0.08162078 -0.1431547  0.75036695
[6,] -0.3834289  0.3180749 -0.47659266 -0.07501674  0.37659087  0.1401873 -0.59797109
[7,] -0.3490255  0.4970255  0.55267291  0.53351836  0.13707747 -0.1455350  0.03296996
```

The loadings for PC1 and PC2 obtained from are exactly the negatives of $v_1$ and $v_2$, respectively. Since eigenvectors (and thus principal components) are unique only up to multiplication by $\pm 1$, this sign difference does not affect the interpretation—in other words, both methods yield the same directions, so they are equivalent in terms of direction and the variance they explain, to be specific:

Let $R = \text{cor(data)}$ be the sample correlation matrix, and suppose that

$$Rv_i = \lambda_i v_i, \quad i = 1, \ldots, p,$$

with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$. Let $v_1$ and $v_2$ be the eigenvectors corresponding to the largest and second largest eigenvalues, respectively.

When principal component analysis is performed (e.g., via `princomp(mydata, cor = TRUE)`), the loadings of the first two principal components are given (up to sign) by

$$e_1 = \pm v_1 \quad \text{and} \quad e_2 = \pm v_2.$$

Since the correlation (i.e., the inner product) between unit vectors is

$$\text{corr}(e_i, v_i) = e_i^\top v_i \quad \text{and} \quad \|e_i\| = \|v_i\| = 1,$$

so the two estimators yield equivalent directions and explain the same amount of variance.

## (c)

Let $Y_i$ denote $i$th PC, the percentage of total (scaled) variation explained by each principal component is calculated as:

$$\text{Percentage}_{\text{PC}_i} = \frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j} = \frac{Var(Y_i)}{\sum_{j=1}^{p} Var(Y_j)} \times 100\%,$$

where $\lambda_i = Var(Y_i)$ are the eigenvalues of the correlation matrix, and $p = 7$ is the number of variables.

Using the code

```
round(princomp(data, cor = TRUE)$sdev^2 / sum(princomp(data, cor = TRUE)$sdev^2), 4)
```

we obtain:

```
Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7
0.8145 0.1053 0.0412 0.0163 0.0123 0.0072 0.0031
```

That is, PC1 explains approximately 81.45% of the total variation and PC2 explains about 10.53%.

## (d)

### (i)

Use the following R code to construct a two-dimensional scatterplot of the 54 observations in the (PC1, PC2) plane and compare the PC1 scores (which reflect the overall performance) with the ranking of the countries associated with the athletes.

```
full_data <- read.table("/Users/yubin/Desktop/Multivariate Analysis/ladyrun25.dat",
                        header = TRUE)
colnames(full_data)=c("Country","100m","200m","400m","800m","1500m","3000m","Marathon")
country_labels <- full_data$Country

data_numeric <- full_data[, !(names(full_data) %in% c("Country"))]

pca_result <- princomp(data_numeric, cor = TRUE)

par(mfrow = c(1,1))
plot(pca_result$scores[,1:2], type = "n",
     main = "Athletes in PC1 vs. PC2 Space",
     xlab = "PC1", ylab = "PC2")
text(pca_result$scores[,1:2], labels = country_labels, cex = 0.7)

# Rank the observations by PC1 scores (higher PC1 score indicates better overall performance)
ranking <- order(pca_result$scores[,1], decreasing = TRUE)
cat("Ranking of countries by PC1 score:\n")
print(country_labels[ranking])
```

The output:

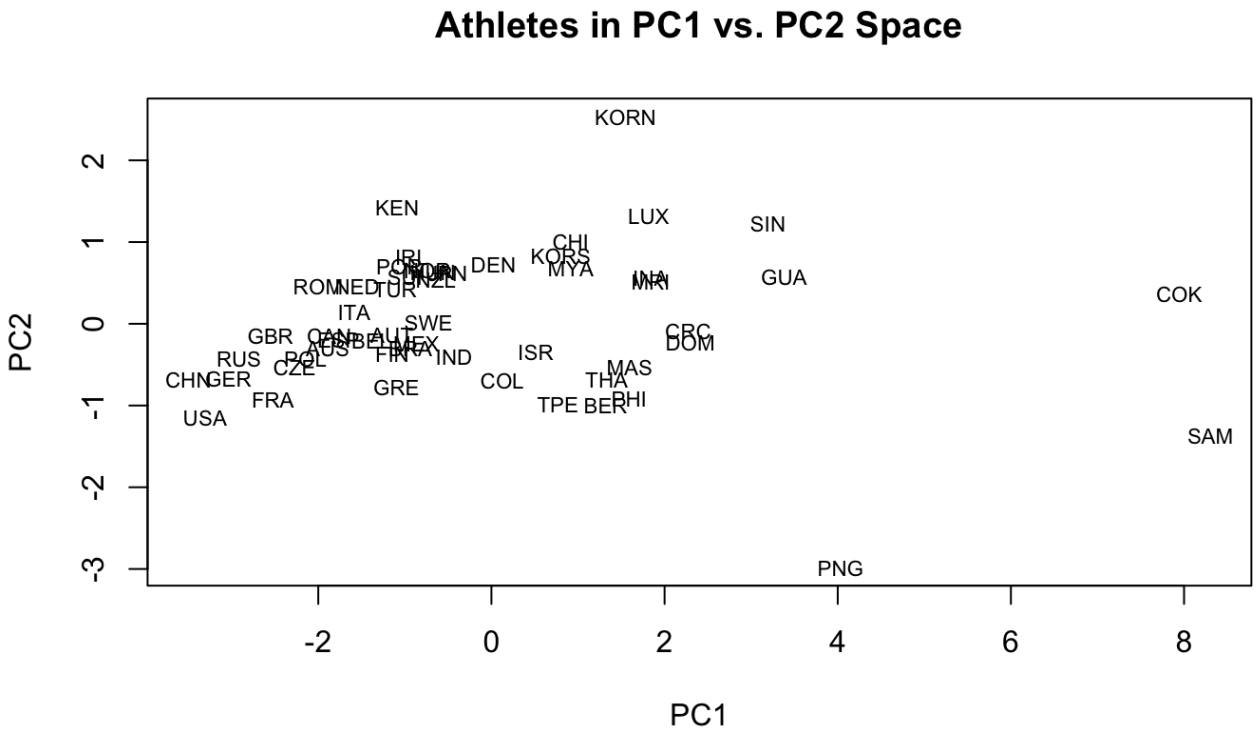


Figure 1: Output

**Explanation:**

```
print(country_labels[ranking])
 [1] "SAM"  "COK"  "PNG"  "GUA"  "SIN"  "DOM"  "CRC"  "MRI"  "INA"  "LUX"  "MAS"  "PHI"  "KORN" "THA"  "BE
[19] "TPE"  "ISR"  "COL"  "DEN"  "IND"  "JPN"  "NZL"  "HUN"  "SWE"  "NOR"  "MEX"  "IRL"  "BRA"  "SUI"  "PO
[37] "FIN"  "AUT"  "BEL"  "NED"  "ITA"  "ESP"  "CAN"  "AUS"  "ROM"  "POL"  "CZE"  "FRA"  "GBR"  "RUS"  "GE
>
```

Now extract and compare the original data for the top 5 and bottom 5 countries (as ranked by their PC1 scores):

```
rank_order <- order(pca_result$scores[,1], decreasing = TRUE)
cat("Ranking of countries by PC1 score:\n")
print(country_labels[rank_order])

top5_indices <- rank_order[1:5]
bottom5_indices <- rank_order[(length(rank_order) - 4):length(rank_order)]

top5_data <- full_data[top5_indices, ]
bottom5_data <- full_data[bottom5_indices, ]

cat("Top 5 Countries by PC1 Score and Their Original Data:\n")
print(top5_data)
```

```
cat("\nBottom 5 Countries by PC1 Score and Their Original Data:\n")
print(bottom5_data)
```

Output:

```
Country  100m    200m   400m    800m    1500m   3000m   Marathon
45 SAM 12.38 25.45 56.32 2.29 5.42 13.12 191.58
10 COK 12.52 25.91 61.65 2.28 4.82 11.10 212.33
39 PNG 11.29 23.12 55.18 2.24 4.62 10.21 221.14
20 GUA 11.92 24.50 55.64 2.15 4.48 9.71 171.33
46 SIN 12.13 24.54 55.08 2.12 4.52 9.94 154.41
Country  100m    200m   400m    800m    1500m   3000m   Marathon
18 GBR 10.85 22.10 49.43 1.94 3.97 8.37 135.25
44 RUS 10.77 21.87 49.11 1.91 3.87 8.38 141.31
17 GER 10.81 21.71 47.60 1.92 3.96 8.33 139.32
53 USA 10.49 21.34 48.70 1.93 3.92 8.43 139.60
8 CHN 10.79 22.01 45.14 1.93 3.84 8.10 139.65
```

**Explanation:** In summary, the top 5 countries (with the highest PC1 scores) tend to have track records associated with a higher performance record, which means that they use more time to finish 100m, 200m, 400m, 800m, 1500m, 3000m, and Marathon, while the bottom 5 countries (with lowest PC1 scores) tend to have a lower performance record, which means that they use less time to finish 100m, 200m, 400m, 800m, 1500m, 3000m and Marathon. This confirms the trend that PC1 reflect the general levels of performance in the dataset.

**(ii)**

**R Code:**

```
pca_result <- princomp(data_numeric, cor = TRUE)

par(mfrow = c(1,1))
plot(pca_result$loadings[,1:2],
     xlim = c(0, 0.6), ylim = c(-0.7, 0.6),
     type = "n", main = "Original Variables in PC1 vs. PC2 Loadings")
text(pca_result$loadings[,1:2],
     labels = colnames(data_numeric),
     cex = 0.8, col = c("blue"))
```

**Output:**

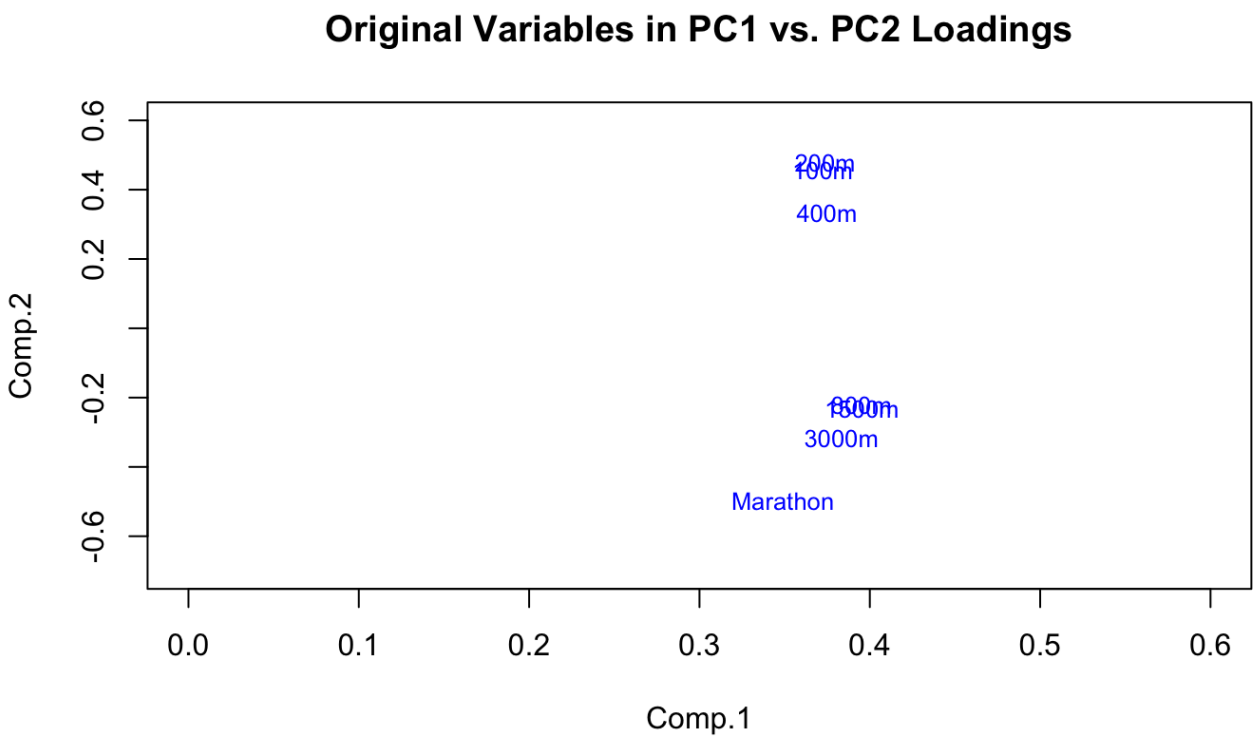


Figure 2: Output

**Explanation:**

The scatterplot uses the loadings for PC1 and PC2 as coordinates for the original variables. Each variable (100m, 200m, 400m, 800m, 1500m, 3000m, and Marathon) is represented by a point whose $x$-coordinate is its loading on PC1 and whose $y$-coordinate is its loading on PC2.

For PC2, however, we observe that the loadings for the sprint events (100m, 200m, and 400m) are positive, whereas the loadings for the longer distance events (800m, 1500m, 3000m, and Marathon) are negative. This suggests that PC2 contrasts short-distance (sprint) performance with long-distance (endurance) performance. A high (more positive) PC2 score would indicate a relatively higher record in short-distance and lower record in long-distance events (or vice versa).

# Question 2

## (a)

### (1) PCA Using Original Data (Covariance Matrix)

```
> pca_orig <- princomp(mydata, cor = FALSE)
> summary(pca_orig)  # Examine the cumulative proportion of variance for original data
Importance of components:
                            Comp.1       Comp.2       Comp.3       Comp.4       Comp.5
Standard deviation     6099.7809499 3488.9515515 2.360509e+02 5.097917e+01 7.210752e-01
Proportion of Variance    0.7525993    0.2462211 1.127059e-03 5.256792e-05 1.051711e-08
```

```
Cumulative Proportion     0.7525993    0.9988204 9.999474e-01 1.000000e+00 1.000000e+00
```

### (2) PCA Using Standardized Data (Correlation Matrix)

```
> pca_scaled <- princomp(mydata, cor = TRUE)
> summary(pca_scaled)  # Examine the cumulative proportion of variance for standardized data
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4      Comp.5
Standard deviation     1.6950851 1.3403955 0.46350500 0.31612348 0.123512644
Proportion of Variance 0.5746627 0.3593320 0.04296738 0.01998681 0.003051075
Cumulative Proportion  0.5746627 0.9339947 0.97696211 0.99694893 1.000000000
```

- **Original Data (Covariance Matrix):** From the summary, we see that:

  $$\text{PC1} \approx 75.26\% \quad \text{(cumulative: 75.26\%)}, \quad \text{PC2} \approx 24.62\% \quad \text{(cumulative: 99.88\%)}.$$

  Thus, the first component alone already captures over 75%, so we would need just **1** component to exceed the 75% threshold.

- **Standardized Data (Correlation Matrix):** From the summary:

  $$\text{PC1} \approx 57.47\% \quad \text{(cumulative: 57.47\%)}, \quad \text{PC2} \approx 35.93\% \quad \text{(cumulative: 93.40\%)}.$$

  Hence, the first principal component alone does not reach 75%, but the first two components together exceed 75% (indeed, they reach 93.40%). Therefore, we would need **2** components to exceed 75% in the scaled data.

## (b)

### R Code and Output

```
par(mfrow = c(1,2))
screeplot(pca_orig, main = "Scree Plot - Original Data")
screeplot(pca_scaled, main = "Scree Plot - Standardized Data")
```

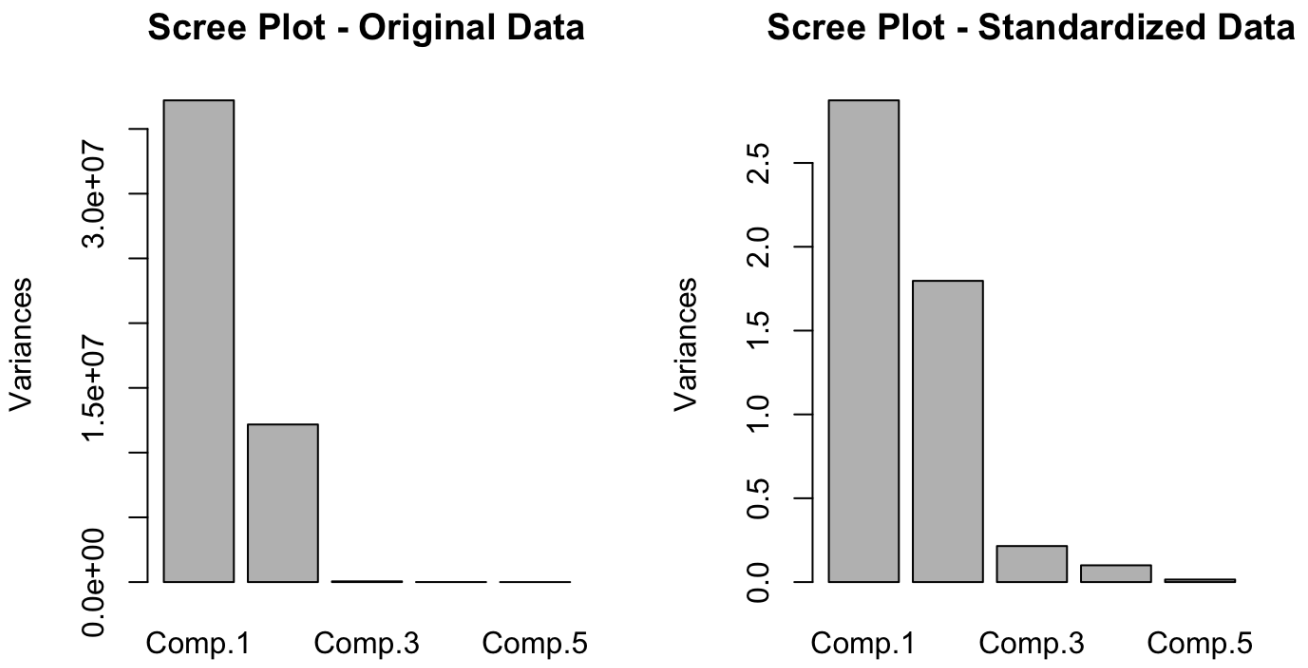


Figure 3: Compare the variance explained by each principal component for the original data vs. the standardized data

We see that for the original data, the drop-off after the first principal component is very large. For the standardized data, there is a more gradual drop after the first component.

## (c)

**R code and Output**

```
> loadings_orig <- pca_orig$loadings
> loadings_scaled <- pca_scaled$loadings

> cat("Loadings for PC1 (Original Data):\n")
Loadings for PC1 (Original Data):
> print(loadings_orig[, 1])
  population    schooling   employment professional   housevalue
  0.0210326211 0.0002420299 0.0269236336 0.0141541619 0.9993159400

> cat("Loadings for PC1 (Standardized Data):\n")
Loadings for PC1 (Standardized Data):
> print(loadings_scaled[, 1])
  population    schooling   employment professional   housevalue
    0.3427304    0.4525067    0.3966948    0.5500565    0.4667384
```

**Comparison of PCA Results**

**(a) Number of Principal Components to Summarize 75% Variability**

- **Original Data:** From the PCA summary using the covariance matrix, PC1 alone explains approximately 75.26% of the total variation. Thus, only one component is needed to reach the 75% threshold. However, note that this high percentage is driven primarily by the large scale of the `housevalue` variable.
- **Standardized Data:** When the data are standardized (using the correlation matrix), PC1 explains only about 57.47% of the variation, and adding PC2 raises the cumulative proportion to approximately 93.40%. Therefore, at least two components are needed to summarize at least 75% of the variability. This indicates that the scaled data have a more balanced structure that is not dominated by a single variable.

**(b) Scree Plots Comparison**

- In the scree plot for the original (unstandardized) data, there is a very steep decline after the first principal component. This reflects the fact that most of the variance is absorbed by PC1, again largely due to the scale of `housevalue`.
- In contrast, the scree plot based on the standardized data (correlation matrix) shows a more gradual decline. The first component accounts for about 57% of the variance, with the second adding about 36%. This more gradual drop-off confirms that, when all variables are standardized, the variance is more evenly distributed across components.

**Comparison of the First Principal Component Loadings**

**Coefficients (Loadings) of PC1:**

$$\text{Original Data: } (0.021,\ 0.00024,\ 0.027,\ 0.014,\ 0.999)$$

$$\text{Standardized Data: } (0.343,\ 0.453,\ 0.397,\ 0.550,\ 0.467)$$

- In the **original-data loadings**, `housevalue` almost completely dominates PC1. This occurs because `housevalue` is measured on a much larger scale and thus has a much higher variance than the other socioeconomic variables. As a result, PC1 in the original-data analysis is primarily reflecting variations in `housevalue`.
- In the **standardized-data loadings**, all five variables contribute more evenly to PC1. Although `professional` has the highest loading, the contributions of the other variables (i.e., population, schooling, employment, and housevalue) are all moderate. This balanced contribution is more informative when the goal is to understand the general socioeconomic structure without a single variable dominating the analysis.

**Which Analysis is Better? Why?**

If the goal is to understand the general socioeconomic structure without a single variable dominating the analysis, and the variables are on different scales that we wish to treat them equally, using the correlation matrix (i.e., standardized data) is preferable. In our case, the original data show that `housevalue` overwhelms PC1 due to its large variance, obscuring the contribution of the other variables. Scaling the data produces a more balanced PC1 of the underlying structure. Therefore, the PCA based on the scaled data is more appropriate.

# Question 3

## (a)

Let

$$X \sim N_p(0, \Sigma),$$

where $\Sigma$ is a $p \times p$ positive definite covariance matrix and the largest eigenvalue $\lambda_1$ of $\Sigma$ is unique and strictly larger than the others. Denote by $v_1$ the corresponding eigenvector, i.e.,

$$\Sigma v_1 = \lambda_1 v_1.$$

Suppose we draw a random sample $X_1, X_2, \ldots, X_n$ from $N_p(0, \Sigma)$ with $n > p$ and perform principal component analysis on the sample data. Let $\hat{e}_1$ denote the sample estimate of the eigenvector corresponding to the largest eigenvalue of the sample covariance matrix.

since the sample covariance matrix

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^T$$

is a consistent estimator of the true covariance matrix $\Sigma$; that is, as $n \to \infty$,

$$\hat{\Sigma} \to \Sigma$$

Since the eigenvalues and eigenvectors are continuous functions of the entries of $\Sigma$ (provided that the eigenvalues are distinct), the eigen-decomposition of $\hat{\Sigma}$ converges to the eigen-decomposition of $\Sigma$.

In particular, let

$$\hat{\Sigma}\hat{e}_1 = \hat{\lambda}_1 \hat{e}_1, \quad \text{and} \quad \Sigma v_1 = \lambda_1 v_1,$$

where $\lambda_1$ is the largest eigenvalue and is assumed to be unique. Then we have

$$\|\hat{e}_1 - v_1\| \to 0 \quad \text{or} \quad \hat{e}_1 \to \pm v_1,$$

where the sign ambiguity arises because if $v_1$ is an eigenvector, so is $-v_1$. Thus, the estimator $\hat{e}_1$ is consistent for $v_1$ up to a sign, meaning that for large $n$, $\hat{e}_1$ is almost parallel to $v_1$ (or $-v_1$).

As $n$ grows large, since $\hat{e}_1$ converges to $\pm v_1$the Pearson correlation coefficient between $\hat{e}_1$ and $v_1$ is given by

$$\rho = \text{corr}(\hat{e}_1, v_1) \to \pm 1$$

Hence,

$$|\rho| \approx 1.$$

Therefore, the first sample principal component $\hat{e}_1$ is nearly parallel to the true eigenvector $v_1$ corresponding to the largest eigenvalue of $\Sigma$. Therefore, the Pearson correlation $\rho = \text{corr}(\hat{e}_1, v_1)$ is approximately $\pm 1$ (and its absolute value is approximately 1).

## (b)

## R Code

```
library(MASS)

Dim = 10       # p = 10
sampN = 50     # n = 50
# Construct a non-trivial covariance matrix that is not simply cI_p
C = replicate(Dim, rnorm(Dim))
myCov = C %*% t(C)   # a p x p covariance matrix

M = 100
Rhos = rep(0, M)
```

```
for (i in 1:M) {
  Data = mvrnorm(n = sampN, mu = rep(0, Dim), Sigma = myCov)
  samS = cov(Data)
  # corr of the 1st sample PC vector with the true 1st PC vector
  Rhos[i] = cor(eigen(samS)$vectors[,1], eigen(myCov)$vectors[,1])
}

hist(abs(Rhos), nclass = 15,
     main = paste("Distribution of PC1 corr's, p =", Dim, ", n =", sampN),
     xlim = c(0, 1))
```

### Result and Discussion

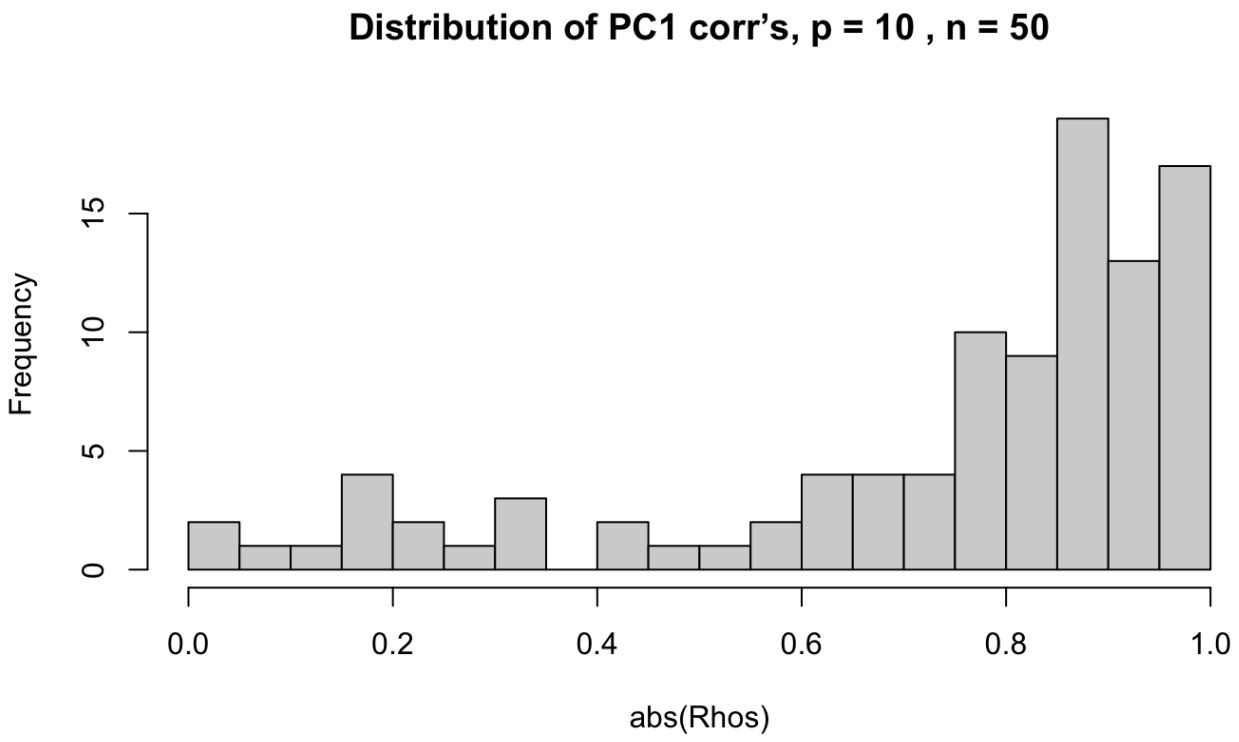


Figure 4: p=10

- **Conclusion for p = 10:** The histogram of $|\rho|$ (see figure in the question) shows that many of the correlation values are clustered toward 1, indicating that the first principal component of the sample is usually close (in direction) to the true first eigenvector for moderate dimension $p = 10$ and sample size $n = 50$.
- **Does it agree with the hypothesis in (a)?** Yes. Part (a) posited that $\hat{e}_1$ should converge to $\pm v_1$ in direction for sufficiently large $n$, implying that $|\rho| \approx 1$.

## (c)

### R Code

```
library(MASS)
```

```
Dim = 100      # p = 100
sampN = 50     # n = 50
C = replicate(Dim, rnorm(Dim))
myCov = C %*% t(C)

M = 100
Rhos = rep(0, M)

for (i in 1:M) {
  Data = mvrnorm(n = sampN, mu = rep(0, Dim), Sigma = myCov)
  samS = cov(Data)
  Rhos[i] = cor(eigen(samS)$vectors[,1], eigen(myCov)$vectors[,1])
}

hist(abs(Rhos), nclass = 15,
     main = paste("Distribution of PC1 corr's, p =", Dim, ", n =", sampN),
     xlim = c(0, 1))
```

## Result and Discussion

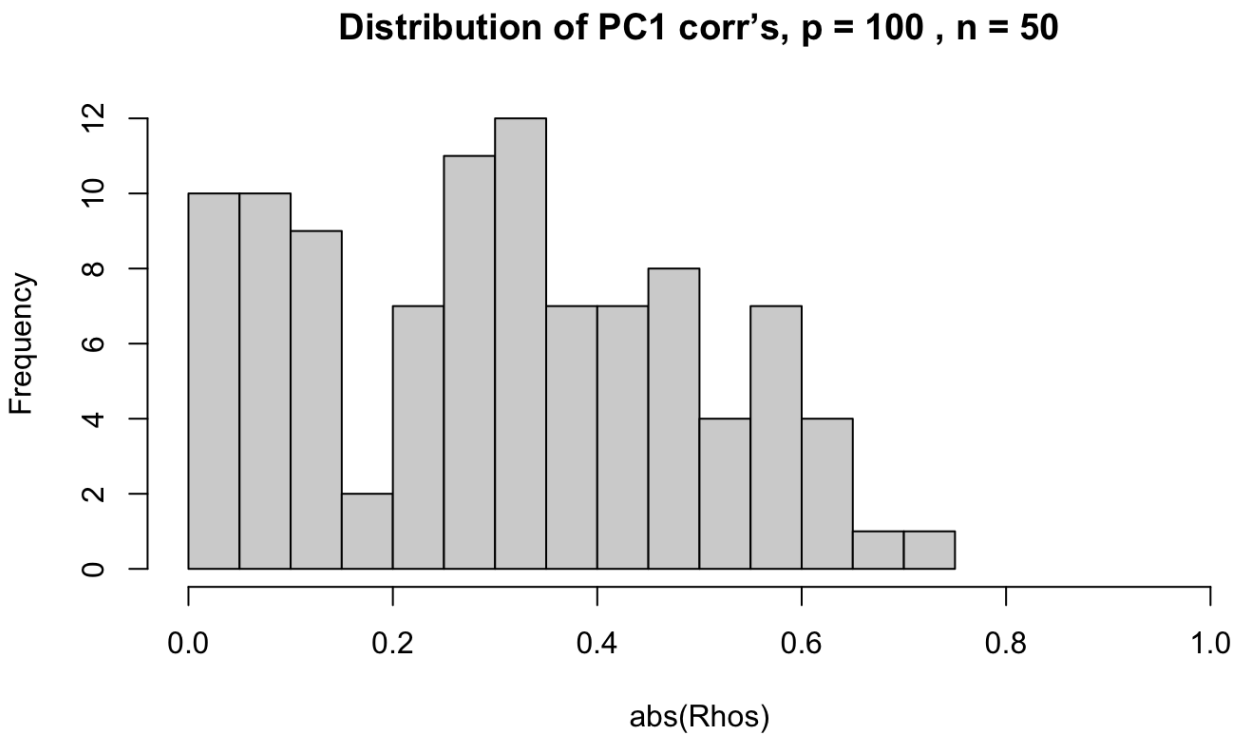


Figure 5: p=100

The histogram in the question shows that $|\rho|$ tends to have a peak around 0.3–0.5, with fewer samples near 1. This indicates that the first sample principal component is much less stable and often deviates from the true eigenvector.

- **Does it agree with the hypothesis in (a)?** In theory, for fixed $p$ and large $n$, $\hat{e}_1 \to \pm v_1$. However, here

$n$ is not large compared to $p = 100$. We see $|\rho|$ is frequently far from 1. That means the high-dimensional nature of the problem (with relatively small $n$) prevents the first principal component from being accurately estimated.

- **Meaning in terms of the first sample principal component:** The result suggests that the first sample PC is not a great estimator of the true eigenvector if $p$ is large relative to $n$. In other words, the sample principal component may not reliably capture the true direction of maximum variance.

## (d)

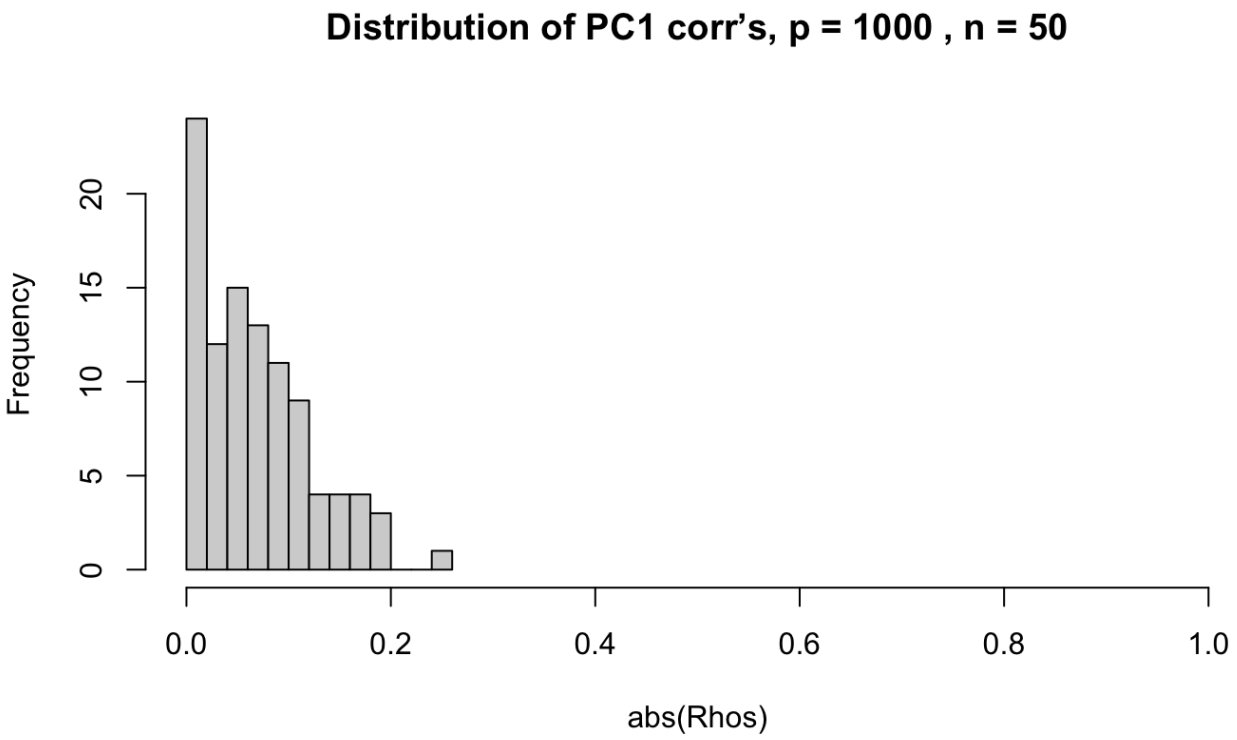


Figure 6: p=1000

We replicate the same simulation procedure for $p = 1000$. The histogram of $|\rho|$ is given in the question. We see that most of the correlations are near 0, with only a few near 0.1 or 0.2, and almost none near 1.

- **Interpretation:** As $p$ grows much larger than $n$, the estimation error in the sample covariance matrix $\hat{\Sigma}$ becomes severe. The first sample principal component is essentially dominated by noise and does not align with the true eigenvector anymore.

## (e)

- **Why is this problematic?** When $p \gg n$, classical PCA loadings can be highly unstable, as evidenced by the simulations above. The first principal components may not be consistent with the true eigenvectors and can not reliably reflect any true underlying "direction" in the data. Large loadings might simply be noise. Therefore, reporting genes with "large PC loadings" in such a high-dimensional, small-sample setting can lead to wrong discoveries. Many of those "important genes" might just be due to random fluctuations in the data rather than true direction of maximum variance.

### Summary of (a)–(e)

- As long as $p$ is small relative to $n$, the sample first principal component correlates strongly with the true one, in line with the hypothesis that $\hat{e}_1 \to \pm v_1$.
- As $p$ grows, but $n$ remains fixed, the distribution of $|\rho|$ skews toward smaller values. The sample first PC is no longer a good estimator of the true direction of largest variance.
- In extremely high-dimensional settings, such as $p = 1000$ or $p = 4000$ with only tens or hundreds of observations, the classical PCA loadings can become highly unreliable. One should be cautious in interpreting large loadings as meaningful.

## Question 4

### (a)

**(i)** We first obtain the PC solution for the factor model using the PC method, for Original Data, using the covariance matrix:

```
pca_orig <- princomp(mydata, cor = FALSE)
summary(pca_orig)

rtev <- pca_orig$sdev

# Construct the loading matrix for m = 2 factors (PC method)
L_PC_orig <- cbind(
  rtev[1]*pca_orig$loadings[, 1],
  rtev[2]*pca_orig$loadings[, 2]
)

cat("\nFactor loadings (PC method, original data):\n")
print(round(L_PC_orig, 4))

# Estimate LL^T
LLT <- L_PC_orig %*% t(L_PC_orig)
cat("\nEstimate of L L^T:\n")
print(round(LLT, 4))

# Estimate Psi = diag(Sigma) - diag(LL^T)
# (Here Sigma is the sample covariance, from cov(mydata))
Sigma_sample <- cov(mydata)
Psi <- diag(Sigma_sample) - diag(LLT)

cat("\nEstimate of Psi (specific variances):\n")
print(round(Psi, 4))

cat("\nEstimate of communality h_i^2")
round(diag(LLT),4)
```

Output:

```
Importance of components:
                          Comp.1       Comp.2        Comp.3         Comp.4           Comp.5
```

```
Standard deviation     6099.7809499 3488.9515515 236.050943693 50.97916903390 0.72107515451751
Proportion of Variance    0.7525993    0.2462211   0.001127059  0.00005256792 0.00000001051711
Cumulative Proportion     0.7525993    0.9988204   0.999947422  0.99999998948 1.00000000000000

Factor loadings (PC method, original data):
                  [,1]      [,2]
population    128.2944 3290.1105
schooling       1.4763   -0.0296
employment    164.2283 1155.7320
professional   86.3373   45.0940
housevalue   6095.6083 -101.0235

Estimate of L L^T:
               population schooling   employment professional   housevalue
population   10841286.7921   92.0577 3823555.4579   159440.994   449653.755
schooling          92.0577    2.1804     208.2595      126.128     9002.114
employment    3823555.4579  208.2595 1362687.2834    66295.656   884315.085
professional   159440.9943  126.1280   66295.6565     9487.600   521722.728
housevalue     449653.7546 9002.1139  884315.0849   521722.728 37166646.709

Estimate of Psi (specific variances):
  population    schooling   employment professional   housevalue
 992273.8140       1.0113  177918.7772    3720.7329 3378807.8363

Estimate of communality h_i^2   population     schooling    employment  professional    housevalue
10841286.7921        2.1804  1362687.2834     9487.6004 37166646.7091
```

**(ii)** We then obtain the PC solution for the factor model using the PC method, for normalized Data (each variable is standardized to variance 1; equivalent to using the correlation matrix).

**R code**

```
pca_scaled <- princomp(mydata, cor = TRUE)
summary(pca_scaled)

rtev_scaled <- pca_scaled$sdev

L_PC_scaled <- cbind(
  rtev_scaled[1] * pca_scaled$loadings[, 1],
  rtev_scaled[2] * pca_scaled$loadings[, 2]
)

cat("\nFactor loadings (PC method, normalized data):\n")
print(round(L_PC_scaled, 4))

# Estimate LL^T
LLT_scaled <- L_PC_scaled %*% t(L_PC_scaled)
cat("\nEstimate of L L^T:\n")
print(round(LLT_scaled, 4))

# Estimate Psi = diag(Sigma) - diag(LL^T)
# (Here Sigma is the sample correlation matrix, given that data are normalized)
Sigma_scaled <- cor(mydata)
Psi_scaled <- diag(Sigma_scaled) - diag(LLT_scaled)

cat("\nEstimate of Psi (specific variances):\n")
```

```
print(round(Psi_scaled, 4))

cat("\nEstimate of communality h_i^2:\n")
print(round(diag(LLT_scaled), 4))
```

**Result**

```
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4      Comp.5
Standard deviation     1.6950851 1.3403955 0.46350500 0.31612348 0.123512644
Proportion of Variance 0.5746627 0.3593320 0.04296738 0.01998681 0.003051075
Cumulative Proportion  0.5746627 0.9339947 0.97696211 0.99694893 1.000000000

Factor loadings (PC method, normalized data):
               [,1]    [,2]
population   0.5810  0.8064
schooling    0.7670 -0.5448
employment   0.6724  0.7260
professional 0.9324 -0.1043
housevalue   0.7912 -0.5582

Estimate of L L^T:
             population schooling employment professional housevalue
population       0.9878    0.0063     0.9762       0.4576     0.0095
schooling        0.0063    0.8851     0.1203       0.7720     0.9109
employment       0.9762    0.1203     0.9793       0.5512     0.1267
professional     0.4576    0.7720     0.5512       0.8802     0.7959
housevalue       0.0095    0.9109     0.1267       0.7959     0.9375

Estimate of Psi (specific variances):
  population    schooling   employment professional   housevalue
      0.0122       0.1149       0.0207       0.1198       0.0625

Estimate of communality h_i^2:
  population    schooling   employment professional   housevalue
      0.9878       0.8851       0.9793       0.8802       0.9375
```

## (b)

The following R code uses the `factanal` function to compute the maximum likelihood (ML) estimates of the factor loading matrix $L$ and the specific variances $\Psi$ for $m = 2$ factors. Then attempt to run the model with $m = 3$ factors to see what happens.

**R Code:**

```
# ML factor analysis with m = 2 factors (no rotation)
Lm <- factanal(mydata, 2, rotation = "none")$loading[, 1:2]
Psim <- factanal(mydata, 2, rotation = "none")$uniq

cat("\nLoadings (m=2):\n")
print(Lm, digits = 4)

cat("\nUniquenesses (diagonal of Psi), m=2:\n")
print(Psim, digits = 4)
```

**R Output:**

```
Loadings (m=2):
               Factor1    Factor2
population    -0.02594   0.99717
schooling      0.89738   0.03767
employment     0.09129   0.97752
professional   0.77697   0.45959
housevalue     0.96122   0.04612

Uniquenesses (diagonal of Psi), m=2:
  population    schooling   employment professional   housevalue
     0.00500      0.19329      0.03613      0.18509      0.07393

Attempting ML factor analysis with m = 3 factors:
Error in factanal(mydata, 3, rotation = "none") :
  3 factors are too many for 5 variables
```

**Explanation:**

- For $m = 2$: The ML factor analysis provides the following estimates for the loading matrix and the specific variances:

$$L = \begin{pmatrix} -0.02594 & 0.99717 \\ 0.89738 & 0.03767 \\ 0.09129 & 0.97752 \\ 0.77697 & 0.45959 \\ 0.96122 & 0.04612 \end{pmatrix}, \quad \Psi = \text{diag}(0.00500,\ 0.19329,\ 0.03613,\ 0.18509,\ 0.07393).$$

  These estimates are obtained on the original data.

- When trying to fit a model with $m = 3$ factors, it returns an error stating that "3 factors are too many for 5 variables." This occurs because with only 5 observed variables there is insufficient information to estimate a 3-factor model.

## (c)

**Residual Matrix Comparison: ML vs. PC Method**

We compare the residuals from the maximum likelihood (ML) factor analysis and the principal component (PC) factor analysis (both with $m = 2$ factors) using scaled data:

$$\text{Residual} = \Sigma - LL^\top - \Psi,$$

where $\Sigma$ is the correlation matrix, $L$ is the loading matrix, and $\Psi$ is the diagonal matrix of uniquenesses (for ML). In the PC approach, we replace $\Psi$ by $\text{diag}\big(\mathbf{1} - \text{diag}(L_n L_n^\top)\big)$.

**(a) ML Residual:**

```
Lm = factanal(mydata,2, rotation="none")$loading[,1: 2]
Psim = factanal(mydata,2, rotation="none")$uniq
round(cor(mydata) - Lm%*%t(Lm) - diag(Psim),4)
```

```
            population schooling employment professional housevalue
population       0.0000   -0.0045     0.0001       0.0007     0.0014
schooling       -0.0045    0.0000     0.0355      -0.0231    -0.0012
employment       0.0001    0.0355     0.0000      -0.0055    -0.0109
professional     0.0007   -0.0231    -0.0055       0.0000     0.0096
housevalue       0.0014   -0.0012    -0.0109       0.0096     0.0000
```

**(b) Scaled PC Residual:**

```
normrtev = princomp(mydata,cor=T)$sdev
Ln = cbind(normrtev[1]*princomp(mydata,cor=T)$loading[,1], normrtev[2]*princomp(mydata,cor=T)$loading[,2])
round(cor(mydata) - Ln%*%t(Ln) - diag(rep(1,5) - diag(Ln%*%t(Ln))),4)
```

```
            population schooling employment professional housevalue
population       0.0000    0.0034    -0.0037      -0.0187     0.0129
schooling        0.0034    0.0000     0.0340      -0.0806    -0.0479
employment      -0.0037    0.0340     0.0000      -0.0365    -0.0048
professional    -0.0187   -0.0806    -0.0365       0.0000    -0.0182
housevalue       0.0129   -0.0479    -0.0048      -0.0182     0.0000
```

**Comparison:**

Hence, we see that the PC residual is typically larger in magnitude. In particular, the ML method yields residuals that are very close to zero, indicating that the model-implied correlation matrix matches the observed correlation matrix well. In contrast, the PC method (which does not optimize the overall fit of the correlation matrix) produces larger residual errors.

**Frobenius Norm Comparison:**

To quantify the fit, we can compute the norm of each residual matrix. For example, in R we might use:

```
# ML residual:
c(sum(abs(cor(mydata) - Lm%*%t(Lm) - diag(Psim))), sum((cor(mydata) - Lm%*%t(Lm) - diag(Psim))^2))

# PC residual:
c(sum(abs(cor(mydata) - Ln%*%t(Ln) - diag(rep(1,5)- diag(Ln%*%t(Ln))))), sum((cor(mydata) -Ln%*%t(Ln) - di

[1] 0.185238727 0.004129386
[1] 0.52156892 0.02434754
```

In our case, the computed sums (sum of absolute residuals and sum of squared residuals) indicate that the ML residuals are lower than those for the PC method.

Based on both the residual matrices and the Frobenius norm comparisons, the ML method is better in estimating the correlation matrix, as it provides a closer fit to the observed correlation structure.

Note that a direct comparison with the PC method applied to the covariance matrix is not meaningful here because the ML method is automatically performed on normalized (scaled) data, ensuring that each variable has unit variance.

```
# PC without scaling the data
rtev = princomp(mydata,cor=F)$sdev
```

```
Ln_n = cbind(rtev[1]*princomp(mydata,cor=F)$loading[,1], rtev[2]*princomp(mydata,cor=F)$loading[,2])

round(cor(mydata) - Ln_n%*%t(Ln_n) - diag(rep(1,5) - diag(Ln_n%*%t(Ln_n))),4)

           population  schooling    employment professional  housevalue
population          0.0000   -92.0479 -3823554.4854 -159440.5554 -449653.732
schooling         -92.0479     0.0000     -208.1052    -125.4366   -9001.251
employment   -3823554.4854  -208.1052        0.0000  -66295.1418 -884314.963
professional  -159440.5554  -125.4366   -66295.1418       0.0000 -521721.950
housevalue    -449653.7321 -9001.2508  -884314.9630 -521721.9501       0.000
```

In contrast, if one uses the PC method on the original (unscaled) covariance matrix, the differences in variable scales may cause the dominant variable(s) to heavily influence the factor solution, which typically leads to a poorer fit to the observed correlation structure.

# Question 5

## (a)

Since the covariance matrix

$$\Sigma = \begin{pmatrix} 5 & 2 & 3 \\ 2 & 6 & 6 \\ 3 & 6 & 10 \end{pmatrix}$$

is a $3 \times 3$ matrix, there are $p = 3$ observed variables in the study, denoted by

$$\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}_{3\times 1}.$$

We assume a one-factor model with $m = 1$. In detailed vector-matrix form, the population factor model can be written as:

$$\underbrace{\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}_{3\times 1}}_{\mathbf{X}} = \underbrace{\begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}_{3\times 1}}_{\boldsymbol{\mu}} + \underbrace{\begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}_{3\times 1}}_{\mathbf{l}} \underbrace{F_{1\times 1}}_{\text{common factor}} + \underbrace{\begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix}_{3\times 1}}_{\boldsymbol{\epsilon}},$$

where

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix}, \quad \text{Var}(\boldsymbol{\epsilon}) = \Psi = \begin{pmatrix} \psi_1 & 0 & 0 \\ 0 & \psi_2 & 0 \\ 0 & 0 & \psi_3 \end{pmatrix}.$$

## (b)

By equating the elements of the decomposed form to $\Sigma$, we obtain equations for both the variances (diagonal elements) and covariances (off-diagonal elements).

Equating the Diagonal Elements:

$$\begin{cases} l_1^2 + \psi_1 = 5, \\ l_2^2 + \psi_2 = 6, \\ l_3^2 + \psi_3 = 10. \end{cases}$$

Equating the Off-Diagonal Elements:

$$\begin{cases} l_1\, l_2 = 2, \\ l_1\, l_3 = 3, \\ l_2\, l_3 = 6. \end{cases}$$

Solve for $l_i$:

From the off-diagonal equations, we have:

$$l_1 l_2 = 2, \quad l_1 l_3 = 3, \quad l_2 l_3 = 6.$$

$$l_2 = \frac{2}{l_1} \quad \text{and} \quad l_3 = \frac{3}{l_1}.$$

Substitute these into $l_2 l_3 = 6$:

$$\frac{2}{l_1} \cdot \frac{3}{l_1} = \frac{6}{l_1^2} = 6 \quad \Longrightarrow \quad l_1^2 = 1.$$

Assume $l_1 > 0$, so

$$l_1 = 1, \quad l_2 = 2, \quad l_3 = 3.$$

Solve for $\psi_i$:

Substitute these into the variance equations:

$$\begin{aligned} l_1^2 + \psi_1 = 1 + \psi_1 = 5 \quad &\Longrightarrow \quad \psi_1 = 4, \\ l_2^2 + \psi_2 = 4 + \psi_2 = 6 \quad &\Longrightarrow \quad \psi_2 = 2, \\ l_3^2 + \psi_3 = 9 + \psi_3 = 10 \quad &\Longrightarrow \quad \psi_3 = 1. \end{aligned}$$

Thus, the solution is:

$$l_1 = 1, \quad l_2 = 2, \quad l_3 = 3, \quad \psi_1 = 4, \quad \psi_2 = 2, \quad \psi_3 = 1.$$

## (c)

For each variable $X_i$, the variance explained by the common factor is given by $l_i^2$ and the total variance is given by

$$\text{Var}(X_i) = l_i^2 + \psi_i.$$

Therefore, the percentage of variance explained by the common factor is:

$$\text{Percentage}_{X_i} = \frac{l_i^2}{l_i^2 + \psi_i} \times 100\%.$$

- For $X_1$:

$$l_1^2 = 1, \quad \text{Var}(X_1) = 5, \quad \text{Percentage} = \frac{1}{5} \times 100\% = 20\%.$$

- For $X_2$:

$$l_2^2 = 4, \quad \text{Var}(X_2) = 6, \quad \text{Percentage} = \frac{4}{6} \times 100\% \approx 66.67\%.$$

- For $X_3$:

$$l_3^2 = 9, \quad \text{Var}(X_3) = 10, \quad \text{Percentage} = \frac{9}{10} \times 100\% = 90\%.$$

## (d)

Since the covariance matrix

$$\Sigma = \begin{pmatrix} 5 & 2 & 3 \\ 2 & 6 & 6 \\ 3 & 6 & 8 \end{pmatrix}$$

is a $3 \times 3$ matrix, there are $p = 3$ observed variables in the study, denoted by

$$\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}_{3\times 1}.$$

We assume a one-factor model with $m = 1$. In detailed vector-matrix form, the population factor model can be written as:

$$\underbrace{\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}_{3\times 1}}_{\mathbf{X}} = \underbrace{\begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}_{3\times 1}}_{\boldsymbol{\mu}} + \underbrace{\begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}_{3\times 1}}_{\mathbf{l}} \underbrace{F_{1\times 1}}_{\text{common factor}} + \underbrace{\begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix}_{3\times 1}}_{\boldsymbol{\epsilon}},$$

where

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{pmatrix}, \quad \text{Var}(\boldsymbol{\epsilon}) = \Psi = \begin{pmatrix} \psi_1 & 0 & 0 \\ 0 & \psi_2 & 0 \\ 0 & 0 & \psi_3 \end{pmatrix}.$$

To solve $L$:

**(i) Diagonal Elements:**

$$\begin{cases} l_1^2 + \psi_1 = 5, \\ l_2^2 + \psi_2 = 6, \\ l_3^2 + \psi_3 = 8. \end{cases}$$

**(ii) Off-Diagonal Elements:**

$$\begin{cases} l_1\, l_2 = 2, \\ l_1\, l_3 = 3, \\ l_2\, l_3 = 6. \end{cases}$$

From the off-diagonal equations, we have:

$$l_1\, l_2 = 2, \quad l_1\, l_3 = 3.$$

Expressing $l_2$ and $l_3$ in terms of $l_1$, we obtain:

$$l_2 = \frac{2}{l_1} \quad \text{and} \quad l_3 = \frac{3}{l_1}.$$

Substitute these into the equation $l_2\, l_3 = 6$:

$$\frac{2}{l_1} \cdot \frac{3}{l_1} = \frac{6}{l_1^2} = 6 \quad \Longrightarrow \quad l_1^2 = 1.$$

Choosing the positive solution, we have:

$$l_1 = 1, \quad l_2 = 2, \quad l_3 = 3.$$

Substitute $l_1 = 1$, $l_2 = 2$, and $l_3 = 3$ into the diagonal equations:

$$\begin{aligned} l_1^2 + \psi_1 = 1 + \psi_1 = 5, &\quad \Longrightarrow \psi_1 = 4, \\ l_2^2 + \psi_2 = 4 + \psi_2 = 6, &\quad \Longrightarrow \psi_2 = 2, \\ l_3^2 + \psi_3 = 9 + \psi_3 = 8, &\quad \Longrightarrow \psi_3 = -1. \end{aligned}$$

Recall that in part (c) we already get percentage explained by the factor is equal to

$$\frac{l_i^2}{l_i^2 + \psi_i} \times 100\%.$$

For the covariance matrix

$$\Sigma = \begin{pmatrix} 5 & 2 & 3 \\ 2 & 6 & 6 \\ 3 & 6 & 8 \end{pmatrix},$$

we solved the factor model equations and obtained

$$l_1 = 1, \quad l_2 = 2, \quad l_3 = 3, \qquad \psi_1 = 4, \quad \psi_2 = 2, \quad \psi_3 = -1.$$

Now, we calculate the percentage for each variable:

- For $X_1$:
$$\text{Percentage}_{X_1} = \frac{l_1^2}{l_1^2 + \psi_1} \times 100\% = \frac{1^2}{1+4} \times 100\% = \frac{1}{5} \times 100\% = 20\%.$$
- For $X_2$:
$$\text{Percentage}_{X_2} = \frac{l_2^2}{l_2^2 + \psi_2} \times 100\% = \frac{2^2}{4+2} \times 100\% = \frac{4}{6} \times 100\% \approx 66.67\%.$$
- For $X_3$:
$$\text{Percentage}_{X_3} = \frac{l_3^2}{l_3^2 + \psi_3} \times 100\% = \frac{3^2}{9+(-1)} \times 100\% = \frac{9}{8} \times 100\% = 112.5\%.$$

**Comment:**

For $X_3$, the common factor is computed to explain 112.5% of the total variance, which is clearly not reasonable because the percentage of variance explained cannot exceed 100%. A proportion greater than 100% would imply that the common factor is accounting for more variance than is present in $X_3$.

Therefore, this over-100% result suggests that the one-factor model is not a suitable representation for this covariance matrix. In other words, the model is over-explaining the variance in $X_3$, indicating that additional factors (i.e., $m > 1$) are needed.

# Question 6

Assume that

$$\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \sim N_3\Big(\boldsymbol{\mu}, \Sigma\Big),$$

with

$$\boldsymbol{\mu} = \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix} \quad \text{and} \quad \Sigma = \begin{pmatrix} 2 & 0 & -1 \\ 0 & 7 & 0 \\ -1 & 0 & 4 \end{pmatrix}.$$

## (a)

(i) Define

$$W_1 = -2X_1 + 3X_2 + X_3.$$

To check whether $X_3$ and $W_1$ are independent (for a multivariate normal, independence is equivalent to zero covariance), by linearity of covariance:

$$\begin{aligned}\mathrm{Cov}(X_3, W_1) &= -2\ \mathrm{Cov}(X_3, X_1) + 3\ \mathrm{Cov}(X_3, X_2) + \mathrm{Var}(X_3) \\ &= -2\big(-1\big) + 3 \cdot 0 + 4 \\ &= 2 + 0 + 4 = 6.\end{aligned}$$

Since $\mathrm{Cov}(X_3, W_1) \neq 0$, $X_3$ and $W_1$ are not independent.

(ii) Define

$$W_2 = 4X_1 - X_2 + X_3.$$

Then,

$$\begin{aligned}\mathrm{Cov}(X_3, W_2) &= 4\ \mathrm{Cov}(X_3, X_1) - \mathrm{Cov}(X_3, X_2) + \mathrm{Var}(X_3) \\ &= 4(-1) - 0 + 4 \\ &= -4 + 4 = 0.\end{aligned}$$

Thus, $X_3$ and $W_2$ are independent.

## (b)

Consider a general case:

Let

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\Big(\begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{pmatrix}\Big).$$

That is, $(X, Y)$ is jointly normal with mean vector $\mu = (\mu_x, \mu_y)$ and covariance matrix $\Sigma$ as above.

**Lemma:** From the formula of block matrix, if we have a partitioned matrix

$$\Sigma \ = \ \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{pmatrix}.$$

A standard result for its inverse (assuming the necessary inverses exist) is

$$\Sigma^{-1} \ = \ \begin{pmatrix} \Sigma_{xx.y}^{-1} & -\,\Sigma_{xx.y}^{-1}\,\Sigma_{xy}\,\Sigma_{yy}^{-1} \\ -\,\Sigma_{yy}^{-1}\,\Sigma_{yx}\,\Sigma_{xx.y}^{-1} & \Sigma_{yy}^{-1} + \Sigma_{yy}^{-1}\,\Sigma_{yx}\,\Sigma_{xx.y}^{-1}\,\Sigma_{xy}\,\Sigma_{yy}^{-1} \end{pmatrix},$$

where

$$\Sigma_{xx.y} \ = \ \Sigma_{xx} \ - \ \Sigma_{xy}\,\Sigma_{yy}^{-1}\,\Sigma_{yx}.$$

Marginally,

$$X \sim \mathcal{N}(\mu_x,\ \Sigma_{xx}), \quad Y \sim \mathcal{N}(\mu_y,\ \Sigma_{yy}).$$

The joint density is

$$f_{X,Y}(x, y) \ = \ \frac{1}{(2\pi)^{\frac{n}{2}}\,|\Sigma|^{\frac{1}{2}}}\ \exp\Big(-\tfrac{1}{2}\begin{pmatrix} x - \mu_x \\ y - \mu_y \end{pmatrix}^T \Sigma^{-1} \begin{pmatrix} x - \mu_x \\ y - \mu_y \end{pmatrix}\Big).$$

The conditional density of $X$ given $Y = y$ is

$$f_{X|Y}(x \mid y) \;=\; \frac{f_{X,Y}(x,y)}{f_Y(y)}.$$

and the marginal density of $Y$ as

$$f_Y(y) \;\propto\; \exp\Big(-\tfrac{1}{2}\,(y-\mu_y)^T\,\Sigma_{yy}^{-1}\,(y-\mu_y)\Big).$$

Hence,

$$f_{X|Y}(x \mid y) \;=\; \frac{f_{X,Y}(x,y)}{f_Y(y)} \;\propto\; \exp\Big(-\tfrac{1}{2}\,\big[(x-\mu_x),(y-\mu_y)\big]\,\Sigma^{-1}\begin{pmatrix}x-\mu_x\\ y-\mu_y\end{pmatrix} \;+\; \tfrac{1}{2}\,(y-\mu_y)^T\,\Sigma_{yy}^{-1}\,(y-\mu_y)\Big).$$

Calculate the exponential term and factor out $-\frac{1}{2}$:

$$\Big(\big[(x-\mu_x),(y-\mu_y)\big]\,\Sigma^{-1}\begin{pmatrix}x-\mu_x\\ y-\mu_y\end{pmatrix} \;-\; (y-\mu_y)^T\,\Sigma_{yy}^{-1}\,(y-\mu_y)\Big)$$

$$= \;(x-\mu_x)^T\,\Sigma_{(x,x)}^{-1}\,(x-\mu_x) + (x-\mu_x)^T\,\Sigma_{(x,y)}^{-1}\,(y-\mu_y) + (y-\mu_y)^T\,\Sigma_{(y,x)}^{-1}\,(x-\mu_x) + (y-\mu_y)^T\big[\Sigma_{(y,y)}^{-1} - \Sigma_{yy}^{-1}\big]\,(y-\mu_y).$$

$$\begin{aligned}= (x-\mu_x)^T\Sigma xx.y^{-1}(x-\mu_x) + (x-\mu_x)^T(-\Sigma_{xx.y}^{-1}\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y) + (y-\mu_y)^T(-\Sigma_{yy}^{-1}\Sigma_{yx}\Sigma_{xx.y}^{-1})(x-\mu_x)\\ +(y-\mu_y)^T(\Sigma_{yy}^{-1}+\Sigma_{yy}^{-1}\Sigma_{yx}\Sigma_{xx.y}^{-1}\Sigma_{xy}\Sigma_{yy}^{-1}-\Sigma_{yy}^{-1})(y-\mu_y)\end{aligned}$$

$$\begin{aligned}= (x-\mu_x)^T\Sigma_{xx.y}^{-1}(x-\mu_x) + (x-\mu_x)^T(-\Sigma_{xx.y}^{-1}\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y) + (y-\mu_y)^T(-\Sigma_{yy}^{-1}\Sigma_{yx}\Sigma_{xx.y}^{-1})(x-\mu_x)\\ +(y-\mu_y)^T(\Sigma_{yy}^{-1}\Sigma_{yx}\Sigma_{xx.y}^{-1}\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y)\end{aligned}$$

$$= (x-\mu_x)^T\Sigma_{xx.y}^{-1}[(x-\mu_x)-(\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y)] + (y-\mu_y)^T(\Sigma_{yy}^{-1}\Sigma_{yx}\Sigma_{xx.y}^{-1})[(\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y)-(x-\mu_x)]$$

$$= [(x-\mu_x)^T-(y-\mu_y)^T(\Sigma_{yy}^{-1}\Sigma_{yx})]\Sigma_{xx.y}^{-1}[(x-\mu_x)-(\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y)]$$

Since

$$\Sigma_{yx}^T \;=\; \Sigma_{xy}, \quad (\Sigma_{yy}^{-1})^T \;=\; \Sigma_{yy}^{-1}, \quad \text{and} \quad (AB)^T = B^T\,A^T.$$

Specifically,

$$\begin{aligned}(y-\mu_y)^T\,(\Sigma_{yy}^{-1}\,\Sigma_{yx}) &= \big[(\Sigma_{yy}^{-1}\,\Sigma_{yx})^T\,(y-\mu_y)\big]^T && \text{(since it is a scalar, equals its transpose)}\\ &= \big[\Sigma_{yx}^T\,(\Sigma_{yy}^{-1})^T\,(y-\mu_y)\big]^T && \text{(transpose of a product reverses the order)}\\ &= \big[\Sigma_{xy}\,\Sigma_{yy}^{-1}\,(y-\mu_y)\big]^T && \text{(using } \Sigma_{yx}^T=\Sigma_{xy} \text{ and } \Sigma_{yy}^{-1} \text{ is symmetric)}\\ &= \big[\Sigma_{xy}\,\Sigma_{yy}^{-1}\,(y-\mu_y)\big]^T.\end{aligned}$$

Therefore,

$$(x-\mu_x)^T \;-\; (y-\mu_y)^T\,(\Sigma_{yy}^{-1}\,\Sigma_{yx}) \;=\; (x-\mu_x)^T \;-\; \big[\Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big]^T.$$

For vectors $a$ and $b$, we know $a^T - b^T = (a-b)^T$. Hence the above difference can be written as

$$\big[(x-\mu_x) \;-\; \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big]^T.$$

Therefore,

$$\Big\{(x-\mu_x)^T \;-\; (y-\mu_y)^T(\Sigma_{yy}^{-1}\,\Sigma_{yx})\Big\}\,\Sigma_{xx.y}^{-1}\,\Big\{(x-\mu_x) \;-\; \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\Big\}$$
$$=\; \big[x-\mu_x-\Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big]^T\,\Sigma_{xx.y}^{-1}\,\big[x-\mu_x-\Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big].$$

Therefore,

$$f_{X|Y}(x \mid y) \propto\; \exp\Big(-\tfrac{1}{2}\;\big[x-\mu_x-\Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big]^T\,\Sigma_{xx.y}^{-1}\,\big[x-\mu_x-\Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big],$$

we see that the exponent is the usual quadratic form

$$-\tfrac{1}{2}\,\big[x-\mu_x \;-\; \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big]^T\,\Sigma_{xx.y}^{-1}\,\big[x-\mu_x \;-\; \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y)\big].$$

This indicates that $f(x \mid y)$ has the kernel of a multivariate normal density in $x$ with shifted mean

$$\mu_x + \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y) \quad \text{and covariance} \quad \Sigma_{xx.y} \;=\; \Sigma_{xx} \;-\; \Sigma_{xy}\,\Sigma_{yy}^{-1}\,\Sigma_{yx}.$$

Hence, putting the normalizing constant back in, we conclude

$$(X \mid Y=y) \;\sim\; \mathcal{N}\Big(\mu_x + \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y),\; \Sigma_{xx} - \Sigma_{xy}\,\Sigma_{yy}^{-1}\,\Sigma_{yx}\Big).$$

In other words, the conditional distribution $X \mid Y = y$ is still Gaussian.

Thus, we have:

## (i) Conditional Expectation

$$E(X \mid Y=y) = \mu_x + \Sigma_{xy}\,\Sigma_{yy}^{-1}(y-\mu_y).$$

## (ii) Conditional Variance

$$\mathrm{Var}(X \mid Y=y) = \Sigma_{xx} - \Sigma_{xy}\,\Sigma_{yy}^{-1}\,\Sigma_{yx}.$$

In this specific setting, we have a joint normal random vector

$$\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \sim N_3\Big(\mu,\, \Sigma\Big)$$

has

$$\mu = \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix} \quad \text{and} \quad \Sigma = \begin{pmatrix} 2 & 0 & -1 \\ 0 & 7 & 0 \\ -1 & 0 & 4 \end{pmatrix}.$$

We partition the vector by letting

$$X = X_1 \quad \text{and} \quad Y = X_3.$$

Thus, we identify:

$$\mu_x = 2, \quad \mu_y = 1,$$
$$\Sigma_{xx} = \mathrm{Var}(X_1) = 2, \quad \Sigma_{yy} = \mathrm{Var}(X_3) = 4,$$
$$\Sigma_{xy} = \mathrm{Cov}(X_1, X_3) = \mathrm{Cov}(X_3, X_1) = \Sigma_{yx} = -1.$$

Applying the general formulas, we have:

$$E(X_1 \mid X_3 = x_3) = \mu_1 + \frac{\mathrm{Cov}(X_1, X_3)}{\mathrm{Var}(X_3)}\,(x_3 - \mu_3),$$
$$\mathrm{Var}(X_1 \mid X_3 = x_3) = \mathrm{Var}(X_1) - \frac{\mathrm{Cov}(X_1, X_3)^2}{\mathrm{Var}(X_3)}.$$

With $\mu_1 = 2$, $\mu_3 = 1$, $\mathrm{Cov}(X_1, X_3) = -1$ and $\mathrm{Var}(X_3) = 4$, it follows that

$$\begin{aligned} E(X_1 \mid X_3 = x_3) &= 2 + \left(-\frac{1}{4}\right)(x_3 - 1) \\ &= 2 - \frac{1}{4}(x_3 - 1). \end{aligned}$$

Also, since $\mathrm{Var}(X_1) = 2$,

$$\mathrm{Var}(X_1 \mid X_3 = x_3) = 2 - \frac{(-1)^2}{4} = 2 - \frac{1}{4} = \frac{7}{4}.$$

**(c)**

To derive the conditional distribution of $X_3$ given $X_1 = x_1$, we use the result from (b) that if

$$\begin{pmatrix} X_1 \\ X_3 \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_3 \end{pmatrix}, \begin{pmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1, X_3) \\ \mathrm{Cov}(X_3, X_1) & \mathrm{Var}(X_3) \end{pmatrix}\right),$$

then

$$X_3 \mid X_1 = x_1 \sim N\left(\mu_3 + \frac{\mathrm{Cov}(X_3, X_1)}{\mathrm{Var}(X_1)}(x_1 - \mu_1),\ \mathrm{Var}(X_3) - \frac{\mathrm{Cov}(X_3, X_1)^2}{\mathrm{Var}(X_1)}\right).$$

Substitute $\mu_3 = 1$, $\mu_1 = 2$, $\mathrm{Cov}(X_3, X_1) = -1$, $\mathrm{Var}(X_1) = 2$, and $\mathrm{Var}(X_3) = 4$:

$$\mu_{3|1} = 1 - \frac{1}{2}(x_1 - 2),$$

$$\sigma^2_{3|1} = 4 - \frac{1}{2} = \frac{7}{2}.$$

Hence, the density function of $X_3 \mid X_1 = x_1$ is

$$f_{X_3|X_1=x_1}(x_3) = \frac{1}{\sqrt{2\pi\left(\frac{7}{2}\right)}} \exp\left\{-\frac{\left[x_3 - \left(1 - \frac{1}{2}(x_1 - 2)\right)\right]^2}{2\left(\frac{7}{2}\right)}\right\} = \frac{1}{\sqrt{7\pi}} \exp\left\{-\frac{\left[x_3 - \left(1 - \frac{1}{2}(x_1 - 2)\right)\right]^2}{7}\right\}.$$

**(d)**

We begin by partitioning the random vector as

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_a \\ X_3 \end{pmatrix} \quad \text{with} \quad \mathbf{X}_a = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}.$$

The joint distribution is

$$\mathbf{X} \sim \mathcal{N}_3\left(\begin{pmatrix}\mu_1\\ \mu_2\\ \mu_3\end{pmatrix}, \begin{pmatrix}\Sigma_{1:2} & \Sigma_{a,3}\\ \Sigma_{3,a} & \sigma_{33}\end{pmatrix}\right),$$

Thus, we identify:

$$\Sigma_{1:2} = \begin{pmatrix}2 & 0\\ 0 & 7\end{pmatrix}, \quad \Sigma_{3,a} = \begin{pmatrix}-1 & 0\end{pmatrix} \quad (\text{and } \Sigma_{a,3} = \Sigma_{3,a}^T),$$

$$\sigma_{33} = 4.$$

For a partitioned multivariate normal, the conditional distribution of $X_3$ given $\mathbf{X}_a = (x_1, x_2)^T$ is

$$X_3 \mid (X_1, X_2) = (x_1, x_2) \sim \mathcal{N}\left(\mu_3 + \Sigma_{3,a}\,\Sigma_{1:2}^{-1}\left(\begin{pmatrix}x_1\\ x_2\end{pmatrix} - \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}\right),\ \sigma_{33} - \Sigma_{3,a}\,\Sigma_{1:2}^{-1}\,\Sigma_{3,a}^T\right).$$

Since

$$\Sigma_{1:2} = \begin{pmatrix}2 & 0\\ 0 & 7\end{pmatrix},$$

$$\Sigma_{1:2}^{-1} = \begin{pmatrix}\frac{1}{2} & 0\\ 0 & \frac{1}{7}\end{pmatrix}.$$

The conditional mean is given by

$$\mu_{3|(1,2)} = \mu_3 + \Sigma_{3,a}\,\Sigma_{1:2}^{-1}\left(\begin{pmatrix}x_1\\ x_2\end{pmatrix} - \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}\right).$$

Substitute the values:

$$\mu_{3|(1,2)} = 1 + \begin{pmatrix}-1 & 0\end{pmatrix}\begin{pmatrix}\frac{1}{2} & 0\\ 0 & \frac{1}{7}\end{pmatrix}\begin{pmatrix}x_1 - 2\\ x_2 - (-3)\end{pmatrix}.$$

$$\mu_{3|(1,2)} = 1 + \begin{pmatrix}-1 & 0\end{pmatrix}\begin{pmatrix}\frac{1}{2}(x_1 - 2)\\ \frac{1}{7}(x_2 + 3)\end{pmatrix} = 1 - \frac{1}{2}(x_1 - 2).$$

The conditional variance is given by

$$\sigma_{3|(1,2)}^2 = \sigma_{33} - \Sigma_{3,a}\,\Sigma_{1:2}^{-1}\,\Sigma_{3,a}^T.$$

$$\sigma_{3|(1,2)}^2 = 4 - \begin{pmatrix}-1 & 0\end{pmatrix}\begin{pmatrix}\frac{1}{2} & 0\\ 0 & \frac{1}{7}\end{pmatrix}\begin{pmatrix}-1\\ 0\end{pmatrix}.$$

$$\sigma_{3|(1,2)}^2 = 4 - \frac{1}{2} = \frac{7}{2}.$$

The density function for a normal random variable with mean $\mu_{3|(1,2)}$ and variance $\sigma_{3|(1,2)}^2$ is

$$f_{X_3|(X_1,X_2)=(x_1,x_2)}(x_3) = \frac{1}{\sqrt{2\pi\,\sigma_{3|(1,2)}^2}}\,\exp\left\{-\frac{1}{2}\,\frac{\left[x_3 - \mu_{3|(1,2)}\right]^2}{\sigma_{3|(1,2)}^2}\right\}.$$

Thus,

$$f_{X_3|(X_1,X_2)=(x_1,x_2)}(x_3) = \frac{1}{\sqrt{7\pi}}\,\exp\left\{-\frac{\left[x_3 - \left(1 - \frac{1}{2}(x_1 - 2)\right)\right]^2}{7}\right\}.$$

Also we can directly observe that $X_3$ and $X_2$ are independent (since $\mathrm{Cov}(X_3, X_2) = 0$ and $X_3$ and $X_2$ are multivariate normal), so the answer is identical to that in part (c).