

INSTITUTE OF ACTUARIES OF INDIA
EXAMINATIONS

February 2025

CS2B - Risk Modelling and Survival Analysis

Time allowed: 1 Hour 45 Minutes

Total Marks: 100

INDICATIVE SOLUTION

Introduction:

The indicative solution has been written by the Examiners with the aim of helping candidates. The solutions are only indicative. It is realized that there could be other points as valid answers and examiners have given credit for any alternative approach or interpretation which they consider to be reasonable.

Sol.1)

```
# Load necessary libraries
install.packages("ggplot2")
install.packages("forecast")
install.packages("tseries")
```

```
library(ggplot2)
library(forecast)
library(tseries)
```

i)

```
# Load the dataset
city_data <- read.csv("city_data.csv")
```

```
# Convert the 'Date' column to Date format
```

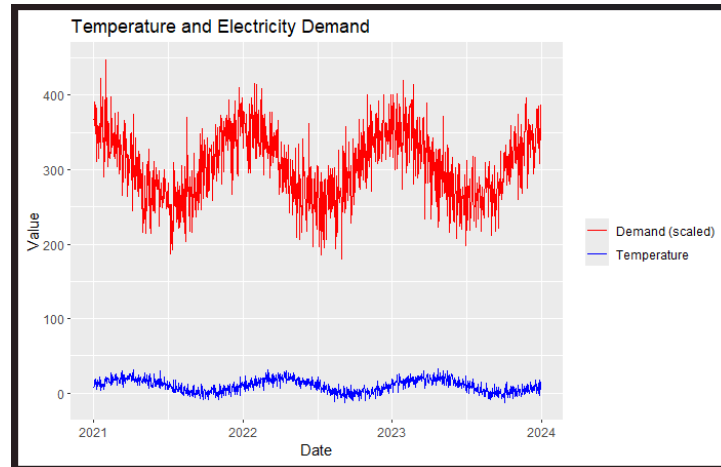
```
city_data$Date <- as.Date(city_data$Date, format="%Y-%m-%d")
```

(2)

ii)

```
# Plot temperature and electricity demand over time
```

```
ggplot(city_data, aes(x = Date)) +
  geom_line(aes(y = Temperature, color = "Temperature")) +
  geom_line(aes(y = Demand, color = "Demand (scaled)")) +
  labs(title = "Temperature and Electricity Demand", x = "Date", y = "Value") +
  scale_color_manual("", values = c("Temperature" = "blue", "Demand (scaled)" = "red"))
```



The demand and temperature is showing a seasonal trend from the above graph.

(4)

iii)

```
# Correlation between temperature and electricity demand
```

```
correlation <- cor(city_data$Temperature, city_data$Demand)
```

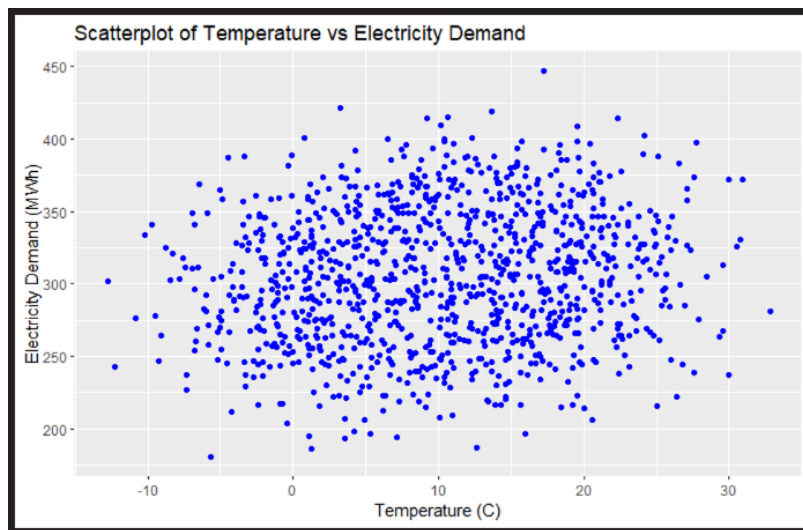
```
print(correlation)
```

```
0.1293143
```

```
# Plot scatterplot to visualize relationship
```

```
ggplot(city_data, aes(x = Temperature, y = Demand)) +
```

```
geom_point(color = "blue") +
labs(title = "Scatterplot of Temperature vs Electricity Demand", x = "Temperature (C)",
y = "Electricity Demand (MWh)")
```



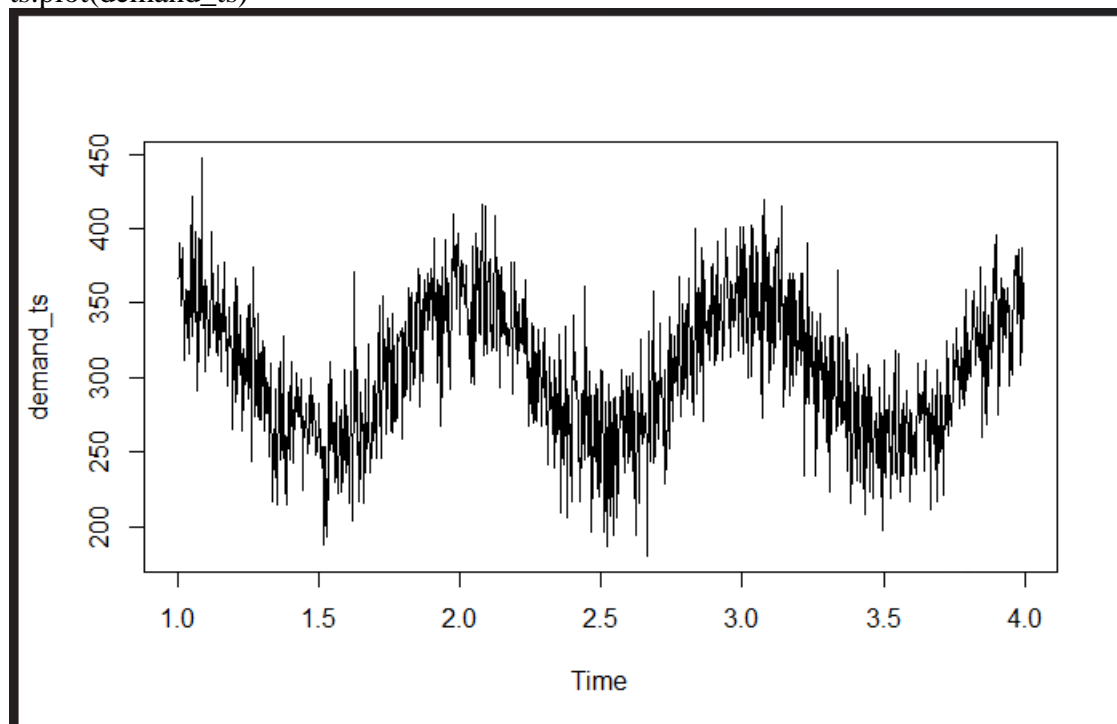
There seems to be a positive correlation between electricity demand and temperature, however the correlation is not very strong. The scatterplot diagram is also showing that the demand and temperature is scattered and doesn't have any linear trend.

(4)

iv)

```
# Convert 'Demand' to a time series object with daily frequency
demand_ts <- ts(city_data$Demand, frequency = 365)
```

```
ts.plot(demand_ts)
```

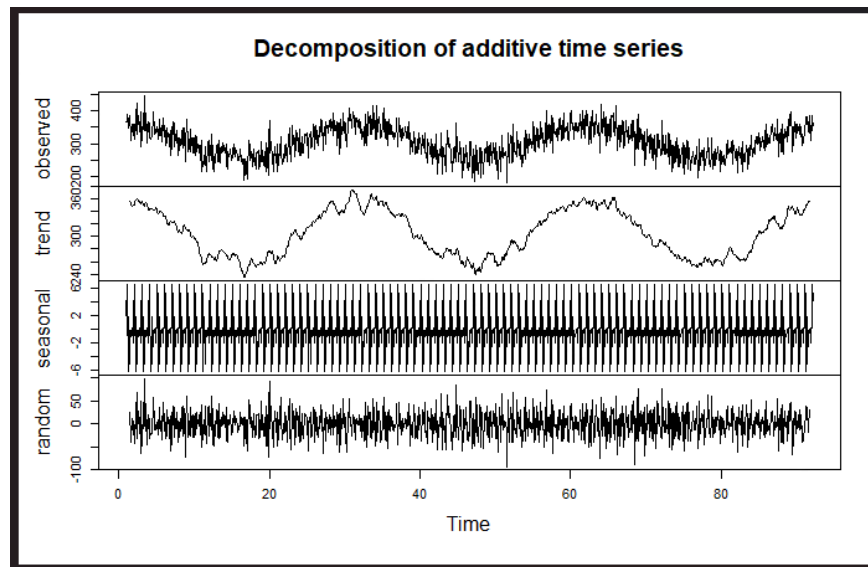


(2)

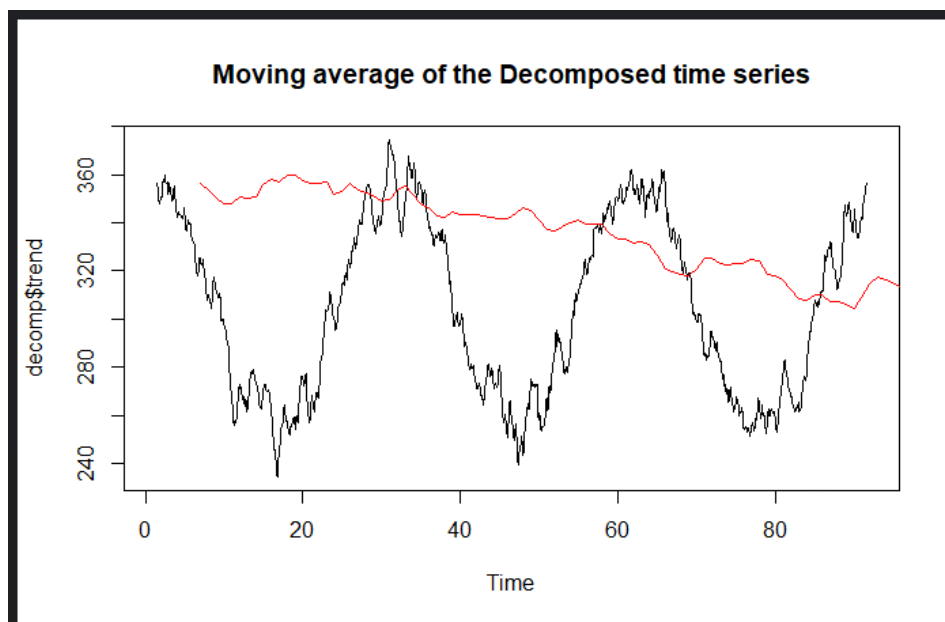
v)

```
# Decompose the time series
decomp <- decompose(ts(demand_ts, frequency = 12))
```

```
plot(decomp)
```



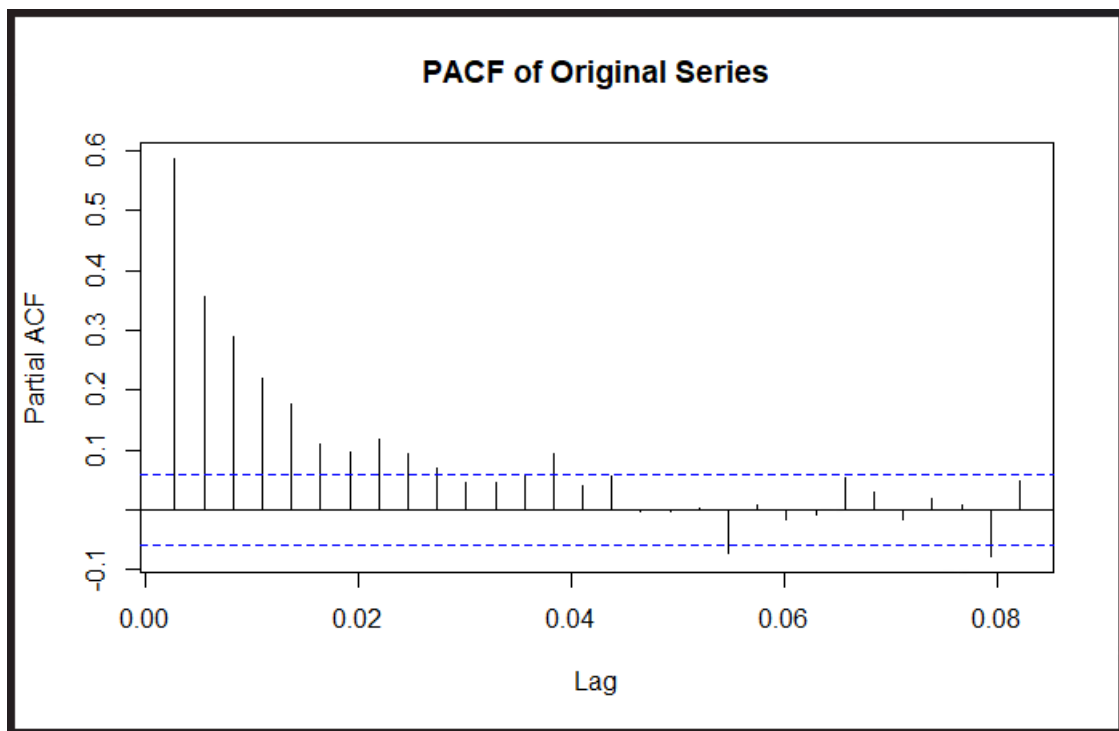
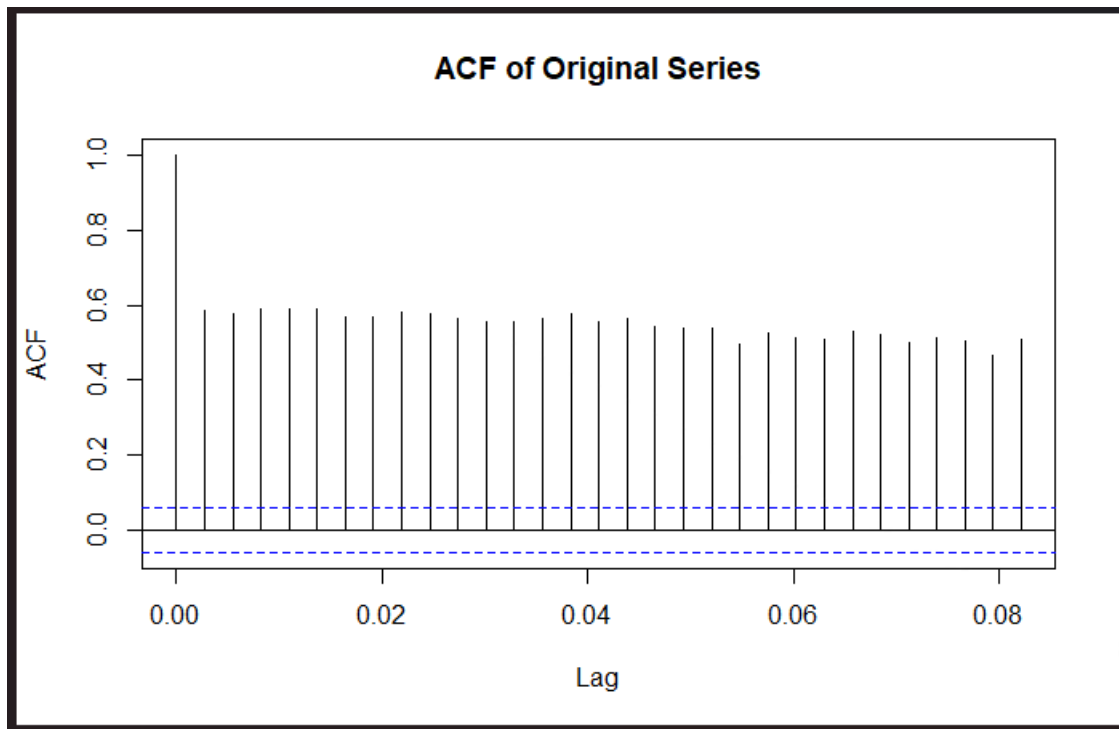
```
plot(decomp$trend)
lines(as.vector(decomp$trend), col = "red")
```



(4)

vi)

```
acf(demand_ts, main = "ACF of Original Series")
pacf(demand_ts, main = "PACF of Original Series")
```



The ACF is decaying very slowly and in fact has become steady hence it is non-stationary. Therefore, the data needs differencing before fitting a model.

(4)

vii)

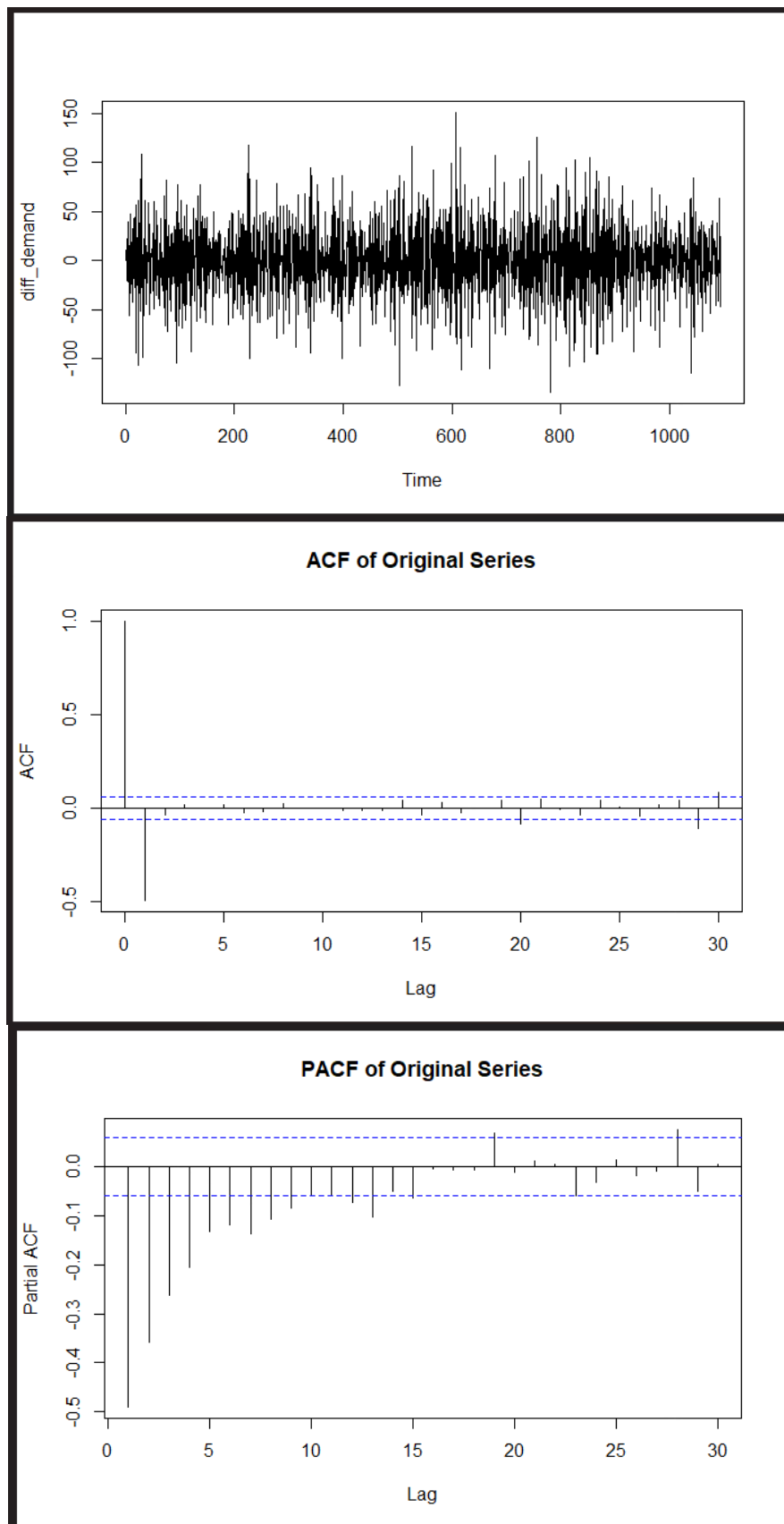
```
diff_demand <- diff(city_data$Demand)
diff_temp <- diff(city_data$Temperature)
```

(1)

viii)

```
ts.plot(diff_demand)
acf(diff_demand, main = "ACF of differenced Series")
```

```
pacf(diff_demand, main = "PACF of Differenced Series")
```



From the above graph it can be seen that the ACF has decreased rapidly and thus has become stationary.

(4)

ix)

```
arima_model <- auto.arima(diff_demand)
```

```
summary(arima_model)
```

Series: diff_demand

ARIMA(1,0,2) with zero mean

Coefficients:

```
      ar1    ma1    ma2  
0.9876 -1.9337 0.9371  
s.e. 0.0055 0.0111 0.0112
```

$\sigma^2 = 949.6$: log likelihood = -5302.4

AIC=10612.79 AICc=10612.83 BIC=10632.78

Training set error measures:

```
      ME    RMSE    MAE    MPE    MAPE    MASE    ACF1  
Training set 0.04410371 30.77317 24.53037 38.95104 302.0277 0.4208159 0.008621595
```

The parameters are 1,0,2 indicating the time series depend on the past 1 term of the series and can be written as weighted average of the past 2 white noise term plus a new white noise process.

(3)

x)

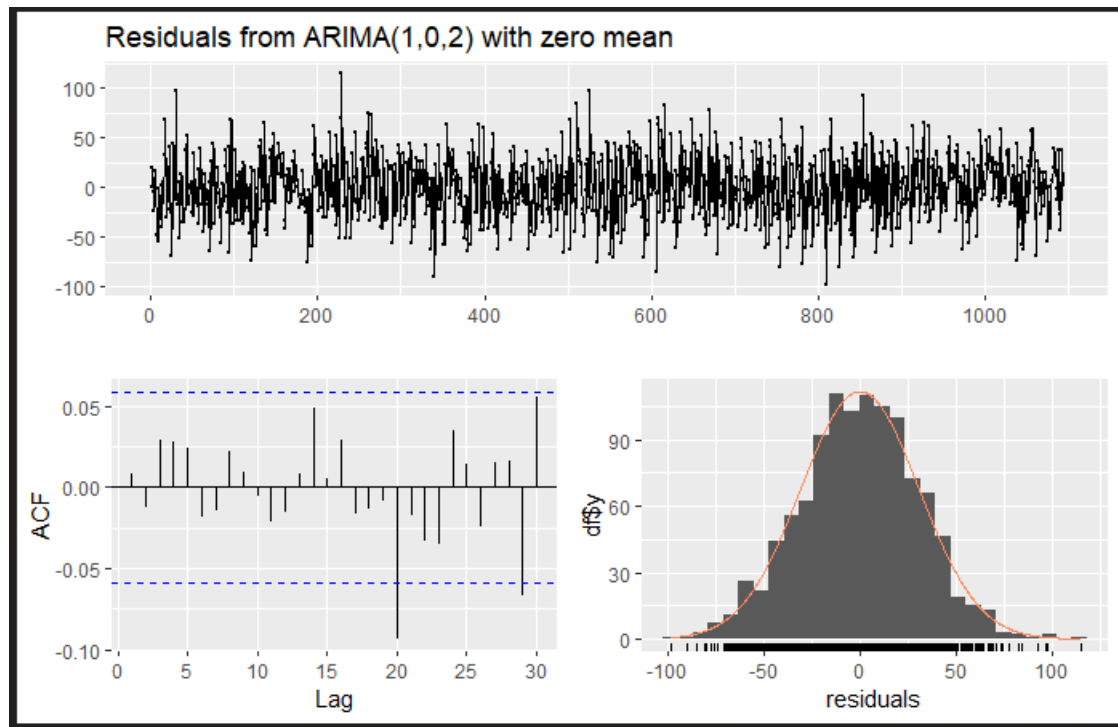
```
checkresiduals(arima_model)
```

Ljung-Box test

data: Residuals from ARIMA(1,0,2) with zero mean

$Q^* = 4.0121$, $df = 7$, $p\text{-value} = 0.7784$

Model df: 3. Total lags used: 10



The fitted model residuals don't seem to have any trend. The ACF has decayed rapidly and the residuals have 0 mean. This indicates the model is a good fit. (3)

xi)

```
predict(arima_model, n.ahead = 30)$pred
```

Time Series:

Start = 1095

End = 1124

Frequency = 1

```
[1] -11.8338153  0.3362773  0.3321017  0.3279780  0.3239054  0.3198834  0.3159114
[8]  0.3119886  0.3081146  0.3042887  0.3005103  0.2967788  0.2930937  0.2894543
[15] 0.2858601  0.2823105  0.2788050  0.2753430  0.2719240  0.2685475  0.2652129
[22] 0.2619197  0.2586674  0.2554555  0.2522835  0.2491508  0.2460571  0.2430017
[29] 0.2399843  0.2370044
```

(2)

[33]

Sol.2)

i)

```
Mall_Customers <- read.csv("<path>/Mall_customers.csv")
```

```
summary(Mall_Customers)
```

CustomerID	Gender	Age	Income	spend_score
Min. : 1.00	Length:200	Min. :18.00	Min. : 15.00	Min. : 1.00
1st Qu.: 50.75	Class :character	1st Qu.:28.75	1st Qu.: 41.50	1st Qu.:34.75
Median :100.50	Mode :character	Median :36.00	Median : 61.50	Median :50.00
Mean :100.50		Mean :38.85	Mean : 60.56	Mean :50.20
3rd Qu.:150.25		3rd Qu.:49.00	3rd Qu.: 78.00	3rd Qu.:73.00
Max. :200.00		Max. :70.00	Max. :137.00	Max. :99.00

The data pertains to people aged 18 to 70 The minimum income is 15000/- per month and maximum income is 1,37,000/- per month. The data seems to be evenly spread with regards to spend score as values ranges from 1 to 99 and mean and median being around 50. (2)

ii)

```
Mall_Customers1 <- Mall_Customers[,3:5]
head(Mall_Customers1)
```

```
  Age Income spend_score
1  19   15      39
2  21   15      81
3  20   16       6
4  23   16      77
5  31   17      40
6  22   17      76
```

(1)

iii)

```
#Assign Clusters
```

```
Mall_Customers1$cluster_name1 <- c(rep("A", length=50), rep("B", length=50),rep("C",
length=50),rep("D", length=50))
```

```
rows_to_print <- c(1:5, 51:55,101:105, 151:155)
```

```
print(Mall_Customers1[rows_to_print,])
```

```
  Age Income spend_score cluster_name1
1  19   15      39 A
2  21   15      81 A
3  20   16       6 A
4  23   16      77 A
5  31   17      40 A
6  49   42      52 B
7  33   42      60 B
8  31   43      54 B
9  59   43      60 B
10 50   43      45 B
11 23   62      41 C
12 49   62      48 C
13 67   62      59 C
14 26   62      55 C
15 49   62      56 C
16 43   78      17 D
17 39   78      88 D
18 44   78      20 D
19 38   78      76 D
20 47   78      16 D
```

(2)

iv)

```
> #Determining coordinates
```

```
> Age_A <- mean(Mall_Customers1$Age[Mall_Customers1$cluster_name1 == "A"])
```

```
> Age_A
```

```
[1] 35.28
```

```
>
> Income_A <- mean(Mall_Customers1$Income[Mall_Customers1$cluster_name1 ==
"A"])
> Income_A
[1] 27.4
>
> spend_score_A <-
mean(Mall_Customers1$spend_score[Mall_Customers1$cluster_name1 == "A"])
> spend_score_A
[1] 49.48
>
> Age_B <- mean(Mall_Customers1$Age[Mall_Customers1$cluster_name1 == "B"])
> Age_B
[1] 44.22
>
> Income_B <- mean(Mall_Customers1$Income[Mall_Customers1$cluster_name1 ==
"B"])
> Income_B
[1] 51.72
>
> spend_score_B <-
mean(Mall_Customers1$spend_score[Mall_Customers1$cluster_name1 == "B"])
> spend_score_B
[1] 50.38
>
> Age_C <- mean(Mall_Customers1$Age[Mall_Customers1$cluster_name1 == "C"])
> Age_C
[1] 38.58
>
> Income_C <- mean(Mall_Customers1$Income[Mall_Customers1$cluster_name1 ==
"C"])
> Income_C
[1] 69.2
>
> spend_score_C <-
mean(Mall_Customers1$spend_score[Mall_Customers1$cluster_name1 == "C"])
> spend_score_C
[1] 50.98
>
>
> Age_D <- mean(Mall_Customers1$Age[Mall_Customers1$cluster_name1 == "D"])
> Age_D
[1] 37.32
>
> Income_D <- mean(Mall_Customers1$Income[Mall_Customers1$cluster_name1 ==
"D"])
> Income_D
[1] 93.92
>
> spend_score_D <-
mean(Mall_Customers1$spend_score[Mall_Customers1$cluster_name1 == "D"])
> spend_score_D
[1] 49.96
```

(6)

v)

#Euclidean Distance

```
> Mall_Customers1$dist_A <- sqrt((Mall_Customers1$Age - Age_A)^2 +
(Mall_Customers1$Income - Income_A)^2+(Mall_Customers1$spend_score - spend_score_A)^2)
```

```
> Mall_Customers1$dist_B <- sqrt((Mall_Customers1$Age - Age_B)^2 +
(Mall_Customers1$Income - Income_B)^2+(Mall_Customers1$spend_score - spend_score_B)^2)
```

```
> Mall_Customers1$dist_C <- sqrt((Mall_Customers1$Age - Age_C)^2 +
(Mall_Customers1$Income - Income_C)^2+(Mall_Customers1$spend_score - spend_score_C)^2)
```

```
> Mall_Customers1$dist_D <- sqrt((Mall_Customers1$Age - Age_D)^2 +
(Mall_Customers1$Income - Income_D)^2+(Mall_Customers1$spend_score - spend_score_D)^2)
```

```
rows_to_print <- c(1:5, 51:55,101:105, 151:155)
```

```
print(Mall_Customers1[rows_to_print,])
```

A tibble: 20 × 9

	Age	Income	spend_score	cluster_name	dist_A	dist_B	dist_C	dist_D
1	19	15	39	A	23.0	46.0	58.9	81.8
2	21	15	81	A	36.8	53.2	64.4	86.4
3	20	16	6	A	47.5	61.9	72.1	91.1
4	23	16	77	A	32.2	49.3	61.2	83.7
5	31	17	40	A	14.7	38.6	53.9	77.8
6	49	42	52	B	20.2	11.0	29.1	53.3
7	33	42	60	B	18.1	17.7	29.2	53.1
8	31	43	54	B	16.8	16.2	27.4	51.5
9	59	43	60	B	30.3	19.7	34.4	56.2
10	50	43	45	B	21.9	11.8	29.2	52.7
11	23	62	41	C	37.7	25.4	19.9	36.1
12	49	62	48	C	37.3	11.6	13.0	34.0
13	67	62	59	C	47.9	26.4	30.4	44.5
14	26	62	55	C	36.2	21.4	15.0	34.2
15	49	62	56	C	37.8	12.7	13.6	34.5
16	43	78	17	D	60.6	42.5	35.4	37.0
17	39	78	88	D	63.7	46.2	38.1	41.3
18	44	78	20	D	59.2	40.2	32.7	34.6

```

19 38 78 76 D 57.2 37.2 26.5 30.5
20 47 78 16 D 61.8 43.4 37.0 38.7

```

(5)

vi)

```
install.packages("stringr")
```

```
library(stringr)
```

```
# Updated Cluster
```

```
Mall_Customers1$cluster_name2 <- rep("A", 200)
```

```
Mall_Customers1$cluster_name2 <- apply(Mall_Customers1[,5:8], 1, function(row)
{names(Mall_Customers1[,5:8])[which.min(row)]})
```

```
Mall_Customers1$cluster_name2 <- str_sub(Mall_Customers1$cluster_name2, -1, -1)
```

```
rows_to_print <- c(1:5, 51:55, 101:105, 151:155)
```

```
> print(Mall_Customers1[rows_to_print,])
```

```
# A tibble: 20 × 9
```

```

  Age Income spend_score cluster_name1 dist_A dist_B dist_C dist_D cluster_name2
  <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <chr>
1 19 15 39 A 23.0 46.0 58.9 81.8 A
2 21 15 81 A 36.8 53.2 64.4 86.4 A
3 20 16 6 A 47.5 61.9 72.1 91.1 A
4 23 16 77 A 32.2 49.3 61.2 83.7 A
5 31 17 40 A 14.7 38.6 53.9 77.8 A
6 49 42 52 B 20.2 11.0 29.1 53.3 B
7 33 42 60 B 18.1 17.7 29.2 53.1 B
8 31 43 54 B 16.8 16.2 27.4 51.5 B
9 59 43 60 B 30.3 19.7 34.4 56.2 B
10 50 43 45 B 21.9 11.8 29.2 52.7 B
11 23 62 41 C 37.7 25.4 19.9 36.1 C
12 49 62 48 C 37.3 11.6 13.0 34.0 B
13 67 62 59 C 47.9 26.4 30.4 44.5 B
14 26 62 55 C 36.2 21.4 15.0 34.2 C
15 49 62 56 C 37.8 12.7 13.6 34.5 B
16 43 78 17 D 60.6 42.5 35.4 37.0 C
17 39 78 88 D 63.7 46.2 38.1 41.3 C
18 44 78 20 D 59.2 40.2 32.7 34.6 C
19 38 78 76 D 57.2 37.2 26.5 30.5 C
20 47 78 16 D 61.8 43.4 37.0 38.7 C

```

(4)

vii)

```
#Cluster comparison
```

```
table(Mall_Customers1$cluster_name1, Mall_Customers1$cluster_name2)
```

```

  A B C D
A 46 4 0 0
B 1 41 8 0
C 0 9 41 0
D 0 0 13 37

```

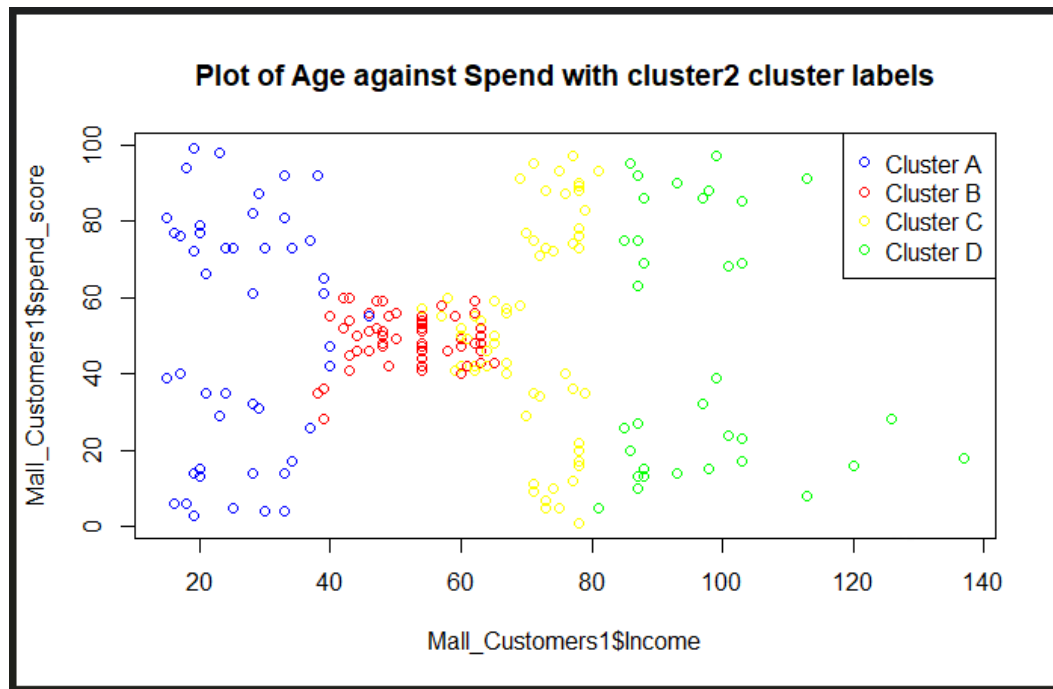
4 customers originally assigned cluster A moved to cluster B, 1 customer originally assigned Cluster B moved to cluster A and 8 customers moved to Cluster C. Similarly, 9 customers

updated cluster is B who were originally assigned cluster C and 13 customers updated clusters are cluster C originally assigned as cluster D. (3)

viii)

```
col_vec <- rep("blue", 200)
col_vec[Mall_Customers1$cluster_name2=="B"] <- "red"
col_vec[Mall_Customers1$cluster_name2=="C"] <- "yellow"
col_vec[Mall_Customers1$cluster_name2=="D"] <- "green"

plot( Mall_Customers1$Income,
      Mall_Customers1$spend_score,
      main="Plot of Age against Spend with cluster2 cluster labels",
      col=col_vec)
legend("topright", legend=c("Cluster A", "Cluster B", "Cluster C", "Cluster D"),
      col=c("blue", "red", "yellow", "green"), pch=1)
```



We could reasonably identify 4 sets of clusters, however, there could be more clusters and the same should be checked by determining the total within sum of squares.

The clustering should be checked for convergence whether adding more clusters leads to better results. The full kmeans algorithm could be implemented to ensure convergence.

Also, there seems to be some overlap in cluster B and cluster C. This could be due to the third variable Age which is not considered in the above graph. (6)

ix)

```
#kmeans
set.seed(123)
stats::kmeans(Mall_Customers1[,1:3], centers = 4, nstart = 10)
```

K-means clustering with 4 clusters of sizes 28, 39, 38, 95

When comparing the Income vs spend_score graph from above, this clustering using 10 iterations shows different results as compared to the earlier clustering done using 1 iteration. The data points seem to be closely following its centroid. However, the model should be checked for convergence, although on visual appearance, 4 cluster seems to be appropriate. (3)

xi)

```
set.seed(123)
```

```
ks <- 1:10
```

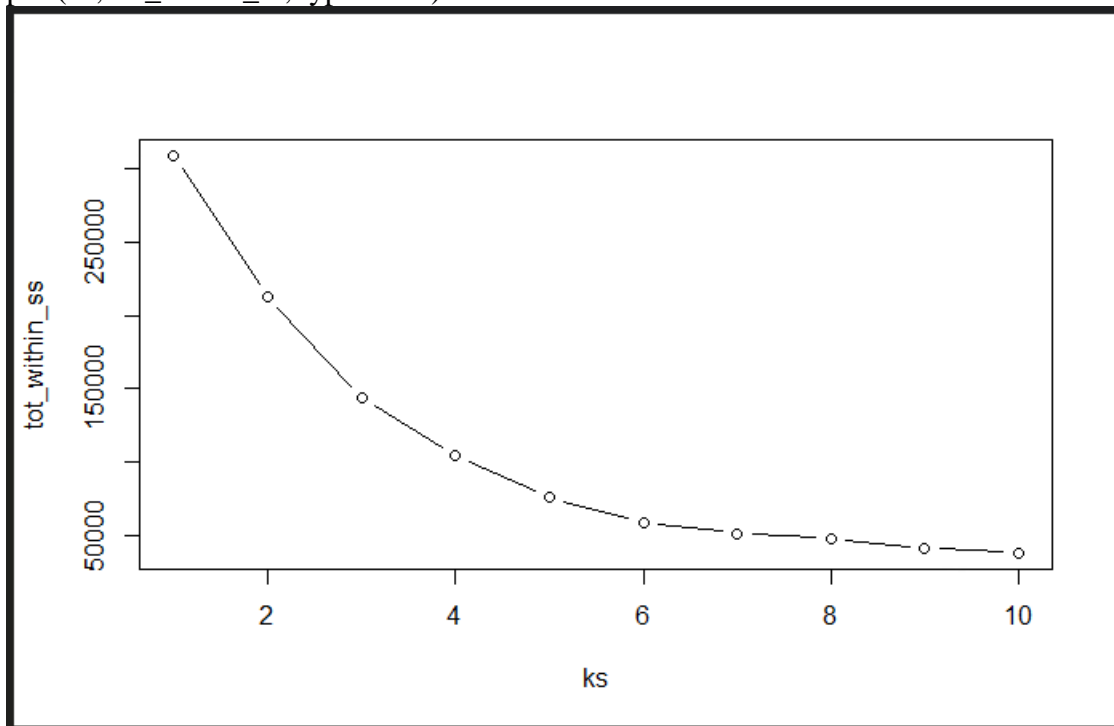
```
tot_within_ss <- sapply(ks, function(k) {
  c2 <- kmeans(Mall_Customers1[,1:3], k, nstart = 10)
  c2$tot.withinss
})
```

```
tot_within_ss
```

```
[1] 308812.78 212840.17 143342.75 104366.15 75350.78 58300.44 51082.54 47340.00
40764.24
```

```
[10] 37804.29
```

```
plot(ks, tot_within_ss, type = "b")
```



The appropriate numbers of clusters based on the above plot is 5 as total within sum of squares starts decreasing slowly beyond 5 clusters. (5)

xii)

```
set.seed(123)
```

```
stats::kmeans(Mall_Customers1[,1:3], centers = 5, nstart = 10)
```

K-means clustering with 5 clusters of sizes 23, 23, 79, 36, 39

Cluster means:

```
Age Income spend_score
1 25.52174 26.30435 78.56522
2 45.21739 26.30435 20.91304
```



```
library(copula)
library(MASS)

set.seed(456)

# Define Gaussian copula
rho <- 0.6
gaussian_cop <- normalCopula(param = rho)

# Simulate from Gaussian copula
u <- rCopula(20000, gaussian_cop)

# Define marginal distributions
# Asset A: Normal distribution
asset_a <- qnorm(u[, 1], mean = 0.05, sd = 0.1)

head(asset_a)
[1] -0.057795363 0.082057007 -0.028017552 0.123443168 0.163693100 0.004579127

# Asset B: t-distribution
asset_b <- qt(u[, 2], df = 4)*0.15 +0.03

head(asset_b)
[1] 0.05638131 -0.15402184 -0.05680725 0.10384026 0.17498915 0.19235551

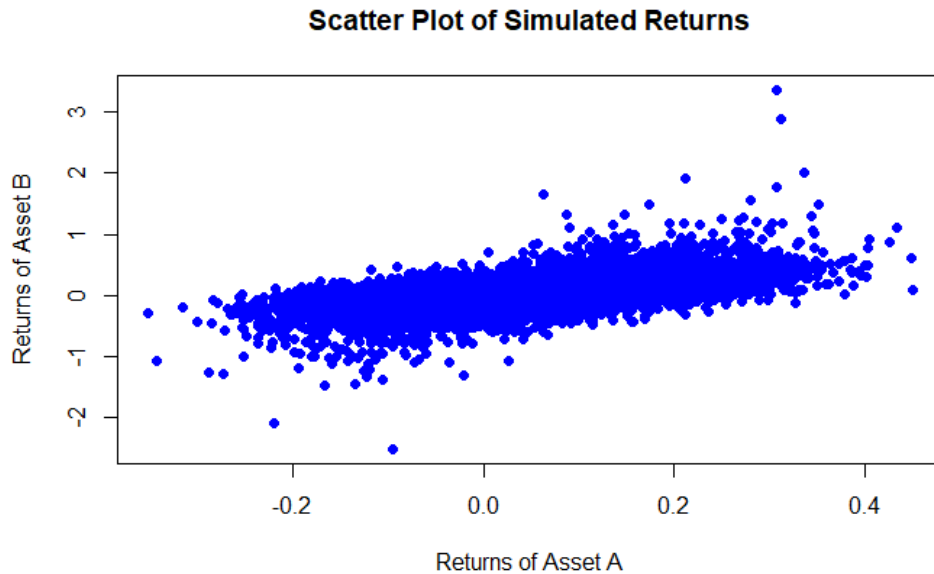
# Combine results into a dataframe
returns <- data.frame(Asset_A = asset_a, Asset_B = asset_b)

# Display first few rows
head(returns)
  Asset_A  Asset_B
1 -0.057795363 0.05638131
2 0.082057007 -0.15402184
3 -0.028017552 -0.05680725
4 0.123443168 0.10384026
5 0.163693100 0.17498915
6 0.004579127 0.19235551
```

(6)

ii)

```
plot(returns$Asset_A, returns$Asset_B, pch = 19, col = "blue",
     xlab = "Returns of Asset A", ylab = "Returns of Asset B",
     main = "Scatter Plot of Simulated Returns")
```



iii)

```
set.seed(456)
t_cop <- tCopula(param = rho, df = 4)

# Simulate from t copula
v <- rCopula(20000, t_cop)

# Define marginal distributions
# Asset A: Normal distribution
asset_a1 <- qnorm(v[, 1], mean = 0.05, sd = 0.1)

head(asset_a1)
[1] -0.10033719 0.08643757 -0.04279233 0.14475873 0.16680008 0.01466407

# Asset B: t-distribution
asset_b1 <- qt(v[, 2], df = 4)*0.15 + 0.03

head(asset_b1)
[1] 0.07324794 -0.16484159 -0.07754710 0.13003440 0.18577196 0.14949059

# Combine results into a dataframe
returns <- data.frame(Asset_A1 = asset_a1, Asset_B1 = asset_b1)

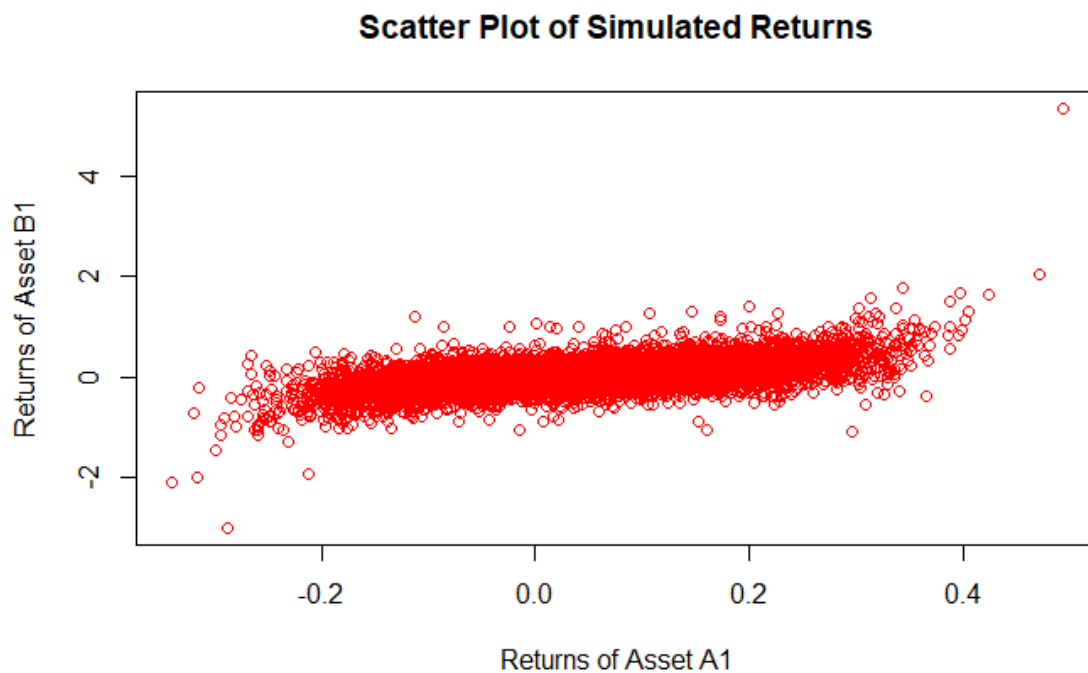
# Display first few rows
head(returns)
  Asset_A1  Asset_B1
1 -0.10033719 0.07324794
2 0.08643757 -0.16484159
3 -0.04279233 -0.07754710
4 0.14475873 0.13003440
5 0.16680008 0.18577196
6 0.01466407 0.14949059
```

(4)

iv)

```
plot(returns$Asset_A1, returns$Asset_B1, col = "red",
     xlab = "Returns of Asset A1", ylab = "Returns of Asset B1",
```

```
main = "Scatter Plot of Simulated Returns")
```

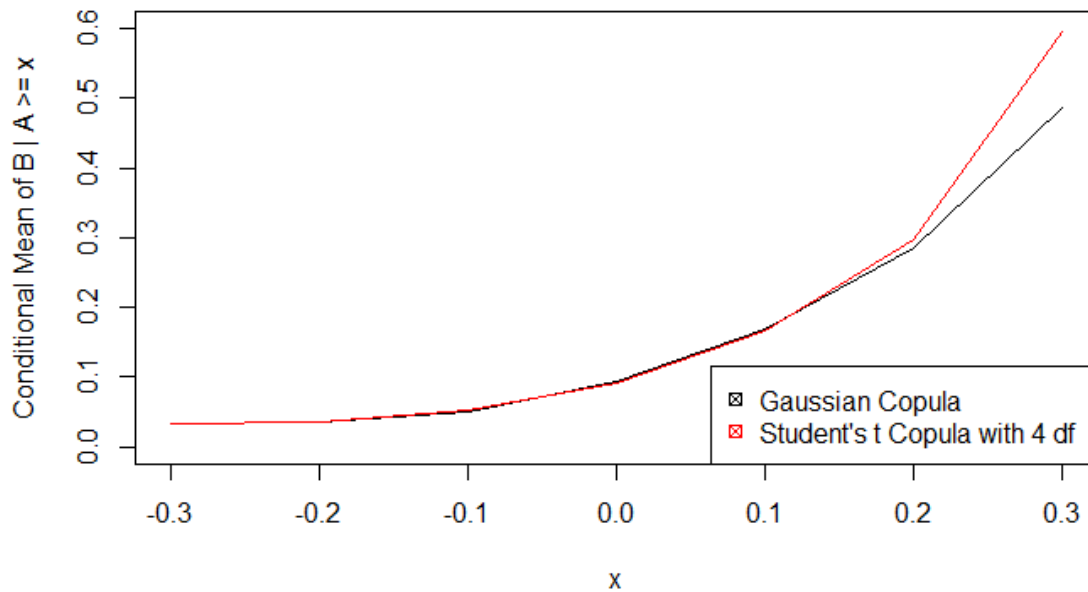


(2)

v)

```
x = seq(from = -0.3, to = 0.3, by = 0.1)
y1 = vector(length = 7)
y2 = vector(length = 7)
for (i in 1:7) {
  y1[i]= mean(asset_b[asset_a >= x[i]])
  y2[i]= mean(asset_b1[asset_a1 >= x[i]])
}
plot(
  x,
  y1,
  ylim = c(0, 0.6),
  type = "l",
  ylab = "Conditional Mean of B | A >= x",
  main = "Conditional Mean of B | A >= x for Two Copulas with rho = 0.6")
lines(
  x,
  y2,
  col = "red")
legend("topright",
  legend = c("Gaussian Copula", "Student's t Copula with 4 df"),
  col = c("black", "red"),
  pch=7)
```

Conditional Mean of $B | A \geq x$ for Two Copulas with $\rho = 0.6$



(6)

vi)

For the lower values of x , the conditional means for both copulas are similar and close to unconditional mean of 0.03.

The conditional means for both copulas increase with increasing x because of the positive value of ρ .

Since the t copula exhibits positive tail dependence and the Gaussian copula has zero tail dependence the graph for the t copula slopes upwards more steeply than for the Gaussian copula.

The order of the Gaussian and t copulas is not consistent for all values of x (i.e. the red line is slightly above the black line and at few places black line is above the red line).

(3)

[23]
