Variation in thermal pressures and resource availability drives disease dynamics

Question 2

A)

```
## here() starts at /Users/colebrookson/github/fall-courses-2024
```

First, I_2 is assumed to be the identity matrix:

```r
Omega <- diag(2)
nu <- 3 # deg freedom
n <- 100 # since there are 100 observation vectors

# get sample covariance matrix S
S <- t(y) %*% y

# for simplicity set the post. parameters
post_scale <- Omega + S
post_df <- nu + n
```

Get the posterior mean of the matrix Σ^{-1}:

```r
post_mean <- post_df * solve(post_scale)
```

if we want the posterior variance of Σ_{ij}^{-1} then we can find the diagonal elements Σ_{ii}^{-1} and the off-diagonals Σ_{ij}^{-1}:

```r
m <- 2
post_var_matrix <- matrix(0, nrow = m, ncol = m)

# diagonal elements
for (i in 1:m) {
  post_var_matrix[i, i] <- 2 * post_df * solve(post_scale)[i, i]^2
}

# off-diagonal elements
for (i in 1:(m - 1)) {
  for (j in (i + 1):m) {
    post_var_matrix[i, j] <- post_df *
      (solve(post_scale)[i, i] *
        solve(post_scale)[j, j] + solve(post_scale)[i, j]^2)
    post_var_matrix[j, i] <- post_var_matrix[i, j] # Symmetry
  }
}
```

We can now get the mean and variance for each ij combination:

```r
for (i in 1:2) {
  for (j in 1:2) {
```

```
    cat(
      "For i = ", i, " and j = ", j, " \n",
      "Posterior mean of Sigma^{-1}_{ij} (precision matrix): ",
      round(post_mean[i, j], 3), " \n",
      "Posterior variance of the Sigma^{-1}_{ij}: ",
      round(post_var_matrix[i, j], 3), "\n \n"
    )
  }
}
```

```
## For i =  1  and j =  1
##  Posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.977
##  Posterior variance of the Sigma^{-1}_{ij}:  0.019
##
## For i =  1  and j =  2
##  Posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.408
##  Posterior variance of the Sigma^{-1}_{ij}:  0.011
##
## For i =  2  and j =  1
##  Posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.408
##  Posterior variance of the Sigma^{-1}_{ij}:  0.011
##
## For i =  2  and j =  2
##  Posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.991
##  Posterior variance of the Sigma^{-1}_{ij}:  0.019
##
```

B)

i. Get 100K MC samples from the Wishert:

```
m <- 2
nu <- 3
Omega <- diag(2)

# generate 100,000 samples from the Wishart distribution
set.seed(123)
num_samples <- 100000
samples <- rWishart(num_samples, post_df, solve(post_scale))

# now approximate the values:
mc_mean_estimate <- matrix(0, nrow = m, ncol = m)
mc_var_estimate <- matrix(0, nrow = m, ncol = m)
# calculate  MC estimates of posterior mean and variance
# Compute the mean and variance of each element Sigma^{-1}_{ij} across the samples
for (i in 1:m) {
  for (j in 1:m) {
    mc_samples_ij <- samples[i, j, ]
    mc_mean_estimate[i, j] <- mean(mc_samples_ij)
    mc_var_estimate[i, j] <- var(mc_samples_ij)
  }
}
for (i in 1:2) {
  for (j in 1:2) {
    cat(
```

```r
      "For i = ", i, " and j = ", j, " \n",
      "Monte Carlos estimate of the posterior mean of Sigma^{-1}_{ij} (precision matrix): ",
      round(mc_mean_estimate[i, j], 3), " \n",
      "Monte Carlos estimate of the pPosterior variance of the Sigma^{-1}_{ij}: ",
      round(mc_var_estimate[i, j], 3), "\n \n"
    )
  }
}
```

```
## For i =  1  and j =  1
##  Monte Carlos estimate of the posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.977
##  Monte Carlos estimate of the pPosterior variance of the Sigma^{-1}_{ij}:  0.019
##
## For i =  1  and j =  2
##  Monte Carlos estimate of the posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.408
##  Monte Carlos estimate of the pPosterior variance of the Sigma^{-1}_{ij}:  0.011
##
## For i =  2  and j =  1
##  Monte Carlos estimate of the posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.408
##  Monte Carlos estimate of the pPosterior variance of the Sigma^{-1}_{ij}:  0.011
##
## For i =  2  and j =  2
##  Monte Carlos estimate of the posterior mean of Sigma^{-1}_{ij} (precision matrix):  0.991
##  Monte Carlos estimate of the pPosterior variance of the Sigma^{-1}_{ij}:  0.019
##
```

The values are essentially the same.

ii. Assuming that the S.E. of each Monte Carlo estimate of the posterior mean for Σ_{ij}^{-1} is

$$\mathrm{SE}(\hat{u}_{ij}) = \frac{\mathrm{SD}(X_{ij})}{\sqrt{n}}$$

```r
# get standard error of each element Sigma^{-1}_{ij} across the samples
mc_se_matrix <- matrix(0, nrow = m, ncol = m)

for (i in 1:m) {
  for (j in 1:m) {
    mc_samples_ij <- samples[i, j, ]
    mc_se_matrix[i, j] <- sd(mc_samples_ij) / sqrt(num_samples)
  }
}

cat("Monte Carlo standard error matrix for posterior means of Sigma^{-1}_{ij}:\n")
```

```
## Monte Carlo standard error matrix for posterior means of Sigma^{-1}_{ij}:
```

```r
print(mc_se_matrix)
```

```
##              [,1]         [,2]
## [1,] 0.0004321176 0.0003328119
## [2,] 0.0003328119 0.0004366300
```

```r
# store correlations
correlation_samples <- numeric(num_samples)

# calculate correlation for each sample
for (k in 1:num_samples) {
  # extract the current precision matrix sample
  precision_matrix <- samples[, , k]

  # invert the precision matrix to get the covariance matrix Sigma
  covariance_matrix <- solve(precision_matrix)

  # compute the correlation Sigma12 / sqrt(Sigma11 * Sigma22)
  sigma_12 <- covariance_matrix[1, 2]
  sigma_11 <- covariance_matrix[1, 1]
  sigma_22 <- covariance_matrix[2, 2]

  correlation <- sigma_12 / sqrt(sigma_11 * sigma_22)

  correlation_samples[k] <- correlation
}

# mean and variance of the correlation
posterior_mean_correlation <- mean(correlation_samples)
posterior_variance_correlation <- var(correlation_samples)

# 95% CI:
credible_interval_correlation <- quantile(correlation_samples,
  probs = c(0.025, 0.975)
)

# Step 10: Print the results
cat(
  paste0("Posterior mean of correlation Sigma12 / sqrt(Sigma11 Sigma22): ", posterior_mean_correlation)
)
```

iii.

```
## Posterior mean of correlation Sigma12 / sqrt(Sigma11 Sigma22): -0.413128482147256
```

```r
cat(
  paste0("Posterior variance of correlation Sigma12 / sqrt(Sigma11 Sigma22): ", posterior_variance_corre
)
```

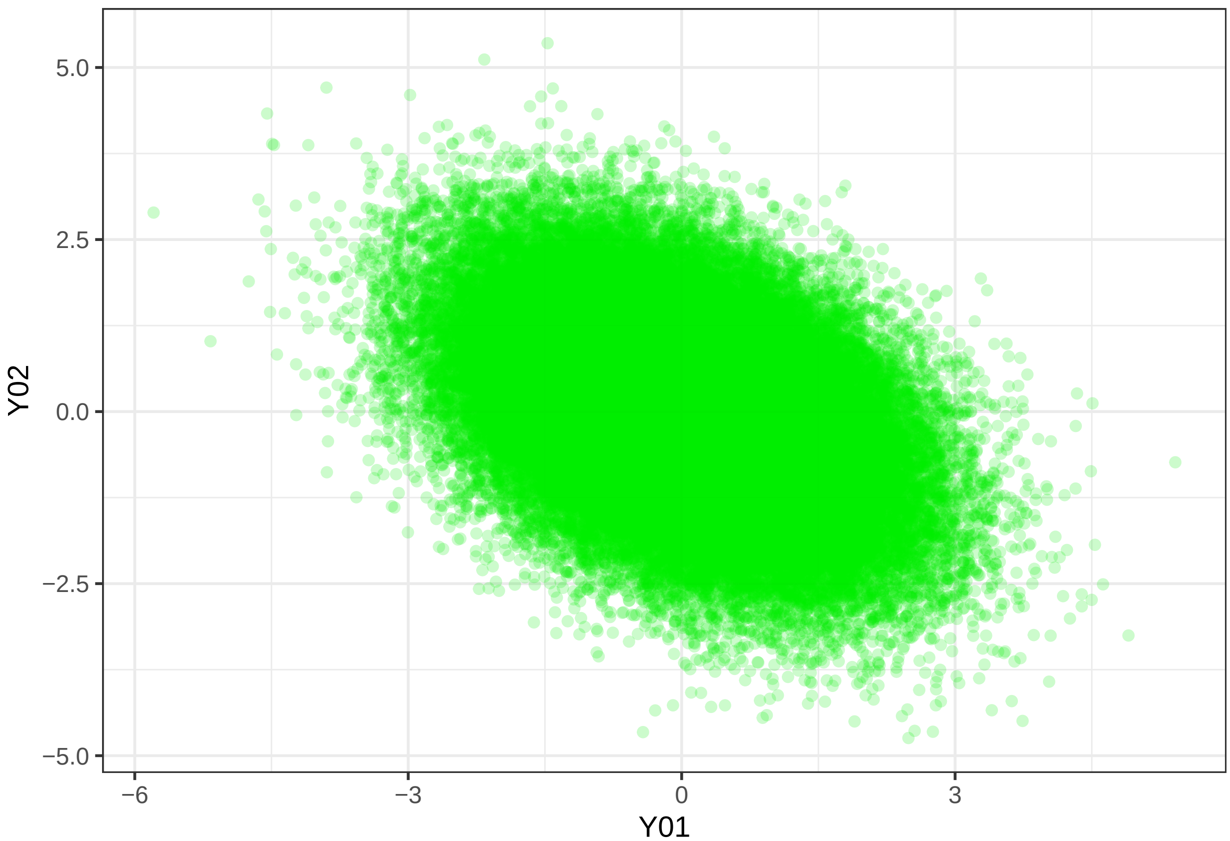
```
## Posterior variance of correlation Sigma12 / sqrt(Sigma11 Sigma22): 0.00674950303363959
```

```r
cat(
  paste0("95% credible interval for correlation Sigma12 / sqrt(Sigma11 Sigma22): ", credible_interval_c
)
```

```
## 95% credible interval for correlation Sigma12 / sqrt(Sigma11 Sigma22): -0.563992028818764 95% credib
```

```r
# store predictive samples
Y0_samples <- matrix(0, nrow = num_samples, ncol = 2)

# generate posterior predictive samples from the multivariate normal distribution
```

```r
for (k in 1:num_samples) {
  # extract the current precision matrix sample
  precision_matrix <- samples[, , k]

  # invert the precision matrix to get the covariance matrix Sigma
  covariance_matrix <- solve(precision_matrix)

  # sample Y_0 ~ MVN(0, Sigma) using rmnorm from the mnormt package
  Y0_samples[k, ] <- rmnorm(1, mean = rep(0, 2), varcov = covariance_matrix)
}

ggplot(data = data.frame(Y0_samples)) +
  geom_point(aes(x = X1, y = X2), colour = "green2", alpha = 0.2) +
  theme_bw() +
  labs(x = "Y01", y = "Y02")
```



iv.