```r
# Load the necessary library
library(here)
```

```
## here() starts at C:/Users/mutse/OneDrive/Desktop/UCT/Courses/Multivariate/Multivariate-Analysis
```

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(reshape2) #for melt()
library(ggplot2)
library(patchwork)
library(tidyr)  #for pivot_longer
```

```
##
## Attaching package: 'tidyr'

## The following object is masked from 'package:reshape2':
##
##     smiths
```

```r
# Read the dataset
data <- read.csv(here("CA\\CA1\\CA1.csv"))
str(data)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ MaxBreadth: int  131 125 131 119 136 138 139 125 131 134 ...
##  $ BasHeight : int  138 131 132 132 143 137 130 136 134 134 ...
##  $ BasLength : int  89 92 99 96 100 89 108 93 102 99 ...
##  $ NasHeight : int  49 48 50 44 54 56 48 48 51 51 ...
##  $ TimePeriod: int  1 1 1 1 1 1 1 1 1 1 ...
```

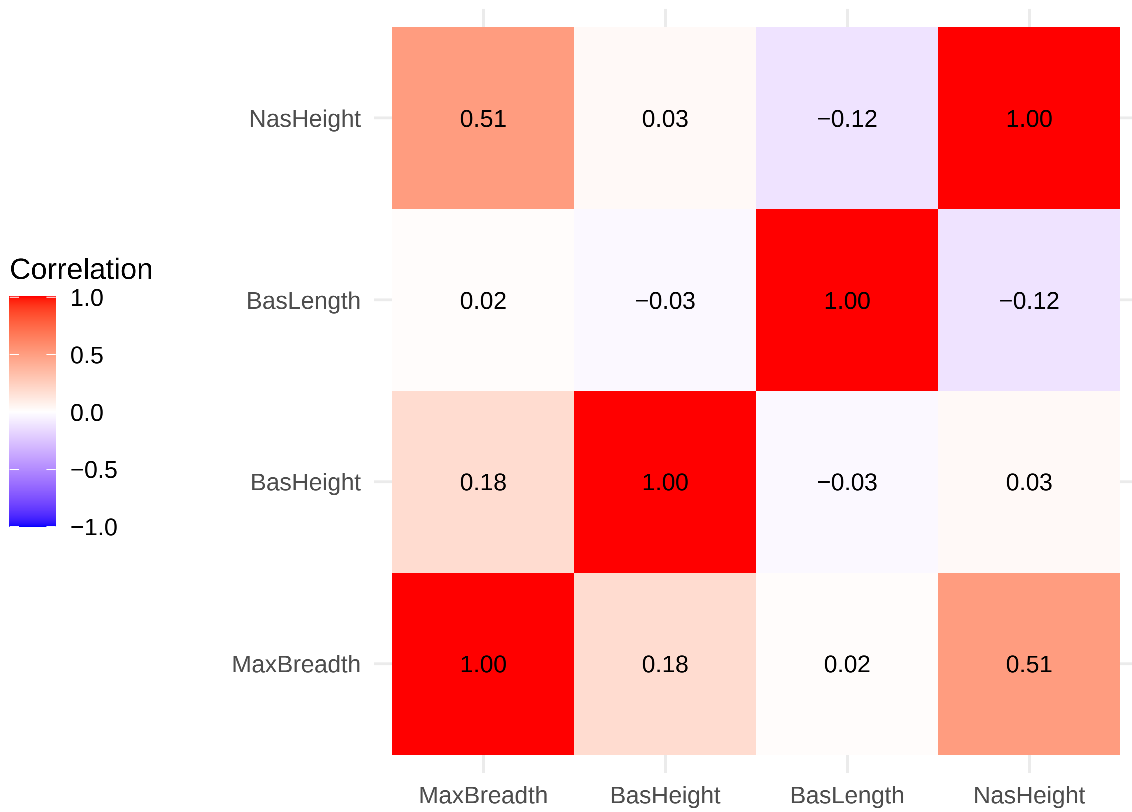
```r
#Qtn 1
# Compute the sample mean vectors for each time period
mean_vectors <- data %>%
  group_by(TimePeriod) %>%
  summarise(
    MaxBreadth = mean(MaxBreadth, na.rm = TRUE),
    BasHeight = mean(BasHeight, na.rm = TRUE),
    BasLength = mean(BasLength, na.rm = TRUE),
    NasHeight = mean(NasHeight, na.rm = TRUE)
  )

mean_vectors
```

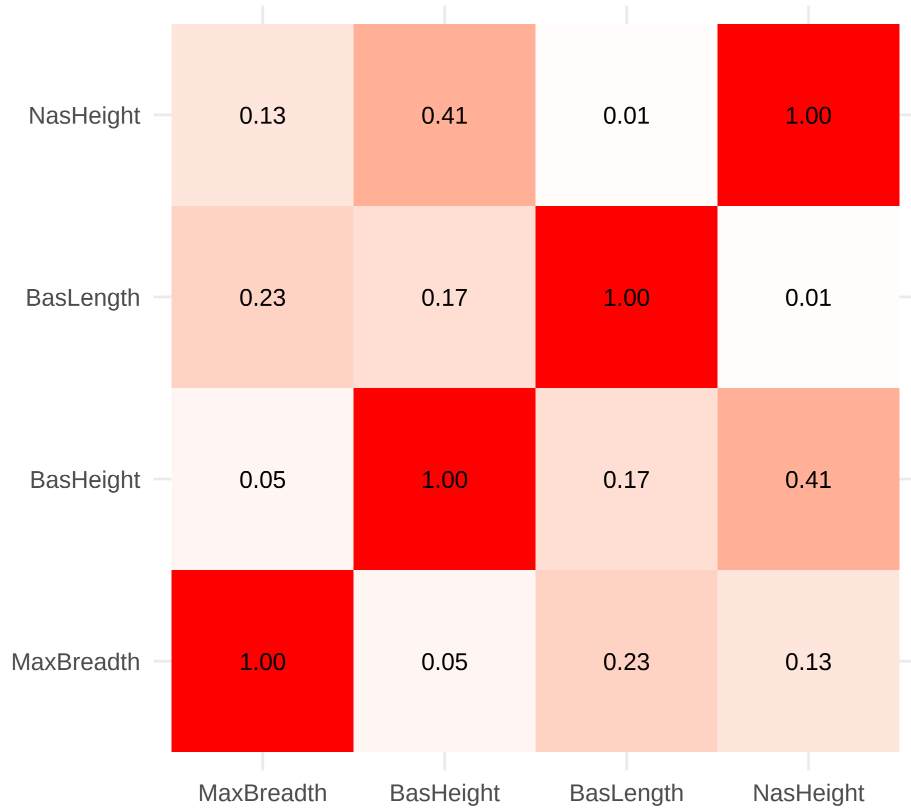
```
## # A tibble: 5 x 5
##   TimePeriod MaxBreadth BasHeight BasLength NasHeight
##        <int>      <dbl>     <dbl>     <dbl>     <dbl>
## 1          1       131.      134.      99.2      50.5
## 2          2       132.      133.      99.1      50.2
## 3          3       134.      134.      96.0      50.6
## 4          4       136.      132.      94.5      52.0
## 5          5       136.      130.      93.5      51.4
```

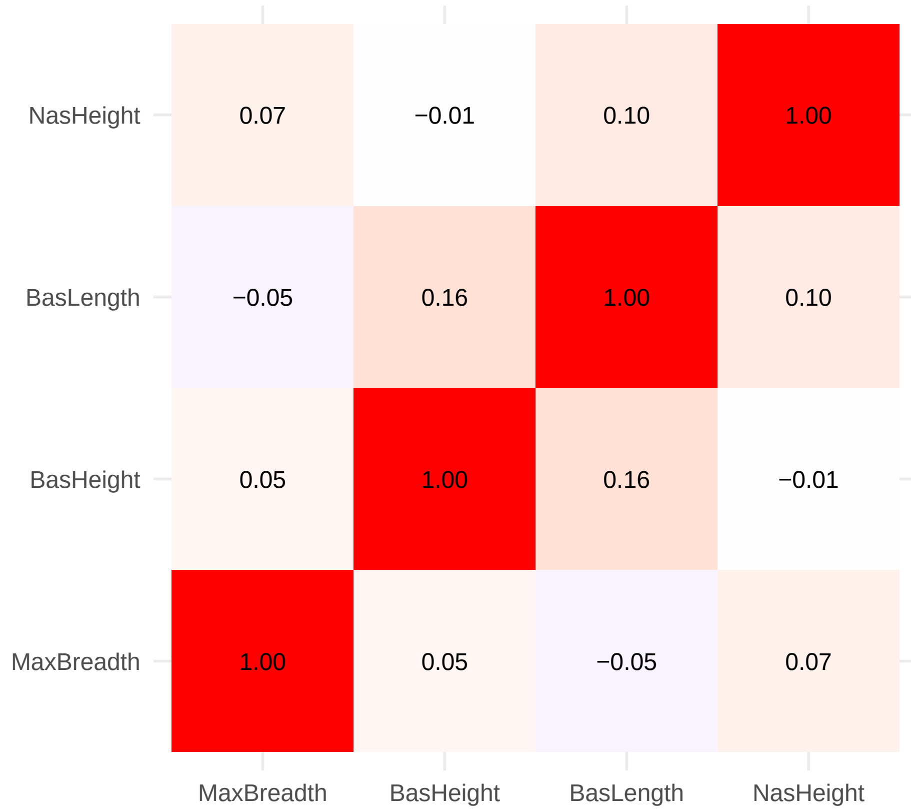
```
#Qtn 2 function
# Function to generate a heat map for a given time period
generate_heat_map <- function(time_period, show_y_axis, show_legend){
  filtered_data <- data[data$TimePeriod == time_period,]
  df_corr_data <- select(filtered_data, -TimePeriod)
  # Assuming df_corr_data is your dataframe
  corr_matrix <- cor(df_corr_data)

  # # Melt the correlation matrix for ggplot
  melted_corr_matrix <- melt(corr_matrix)
  # Exclude the lower triangle and diagonal
  # melted_corr_matrix$value[melted_corr_matrix$Var1 >= melted_corr_matrix$Var2] <- NA

  # Plot
  p <- ggplot(melted_corr_matrix, aes(x = Var1, y = Var2, fill = value)) +
    geom_tile() +
    geom_text(aes(label = sprintf("%.2f", value)), color = "black", size = 3) + # Adds text labels
    scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                         midpoint = 0, limit = c(-1,1), space = "Lab",
                         name="Correlation") +
    theme_minimal() +
    labs(x = '', y = '') +
    coord_fixed()


  if (!show_legend) {
    p <- p + theme(legend.position = "none")
  } else {
    # Position the legend to the left of the chart
    p <- p + theme(legend.position = "left")
  }
  return(p)
}
```

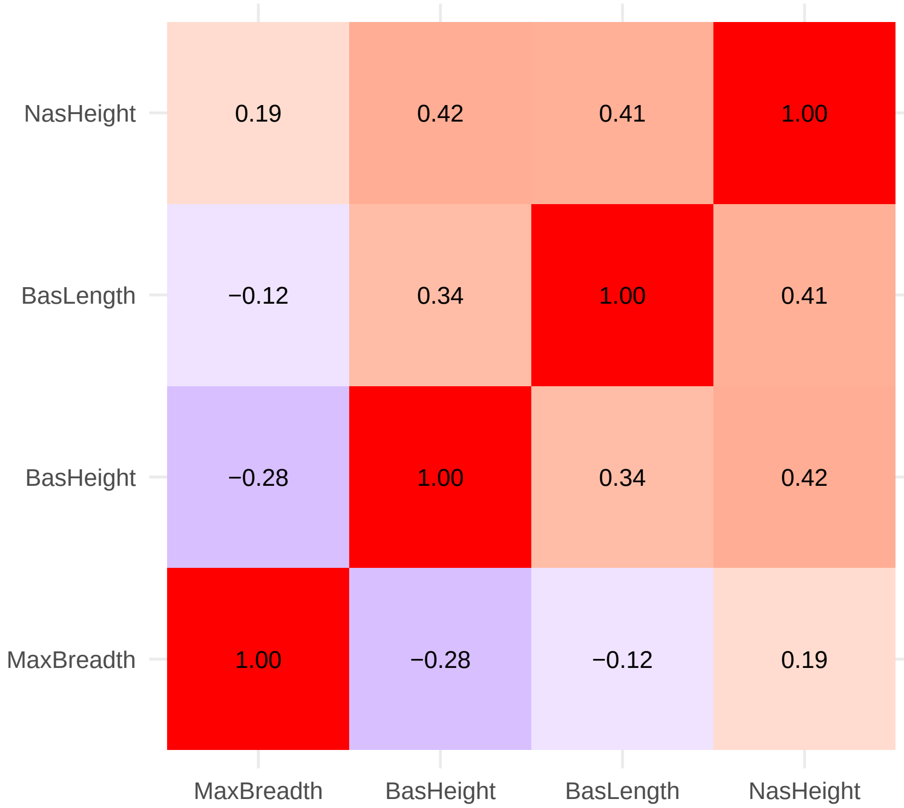
```
# #Q2 cntd
# Generate heat map for time periods
time_periods <- unique(data$TimePeriod)
plot_list <- list()

for (time_period in 1:length(time_periods)) {
  if (time_period == 1) {
    plot <- generate_heat_map(time_period, show_legend = TRUE)
    print(plot)
    #save plot
    plot_type <- "correlation_map_timeperiod_"
```

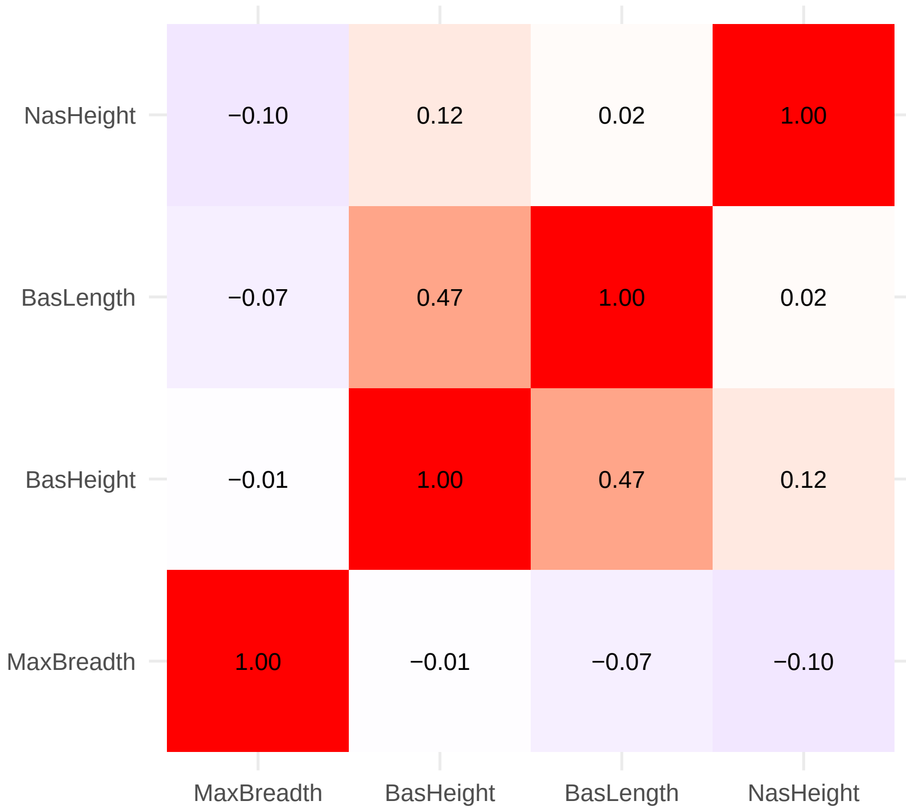
```
    file_ext <- ".png"
    plot_name <- paste(plot_type, time_period, file_ext)
    ggsave(plot_name, plot = plot, width = 10, height = 6, units = "in")
    first_plot <- FALSE
  } else {
    plot <- generate_heat_map(time_period, show_legend = FALSE)
    plot(plot)
    #save plot
    plot_type <- "correlation_map_timeperiod_"
    file_ext <- ".png"
    plot_name <- paste(plot_type, time_period, file_ext)
    ggsave(plot_name, plot = plot, width = 10, height = 6, units = "in")
    first_plot <- FALSE
  }
}
```




NasHeight
BasLength
BasHeight
MaxBreadth
0.13
0.41
0.01
1.00
0.23
0.17
1.00
0.01
0.05
1.00
0.17
0.41
1.00
0.05
0.23
0.13
MaxBreadth
BasHeight
BasLength
NasHeight


NasHeight
BasLength
BasHeight
MaxBreadth
0.07
−0.01
0.10
1.00
−0.05
0.16
1.00
0.10
0.05
1.00
0.16
−0.01
1.00
0.05
−0.05
0.07
MaxBreadth
BasHeight
BasLength
NasHeight


NasHeight
BasLength
BasHeight
MaxBreadth
0.19
0.42
0.41
1.00
−0.12
0.34
1.00
0.41
−0.28
1.00
0.34
0.42
1.00
−0.28
−0.12
0.19
MaxBreadth
BasHeight
BasLength
NasHeight




```
# combined_plot
```

```
# Q3

# Filter data for period 1
data_period_1 <- data[data$TimePeriod == 1,]

# Extract vectors for X1 and X3
x1 <- data_period_1$MaxBreadth
x3 <- data_period_1$BasLength

# Compute deviation vectors from their means
x1_dev <- x1 - mean(x1)
x3_dev <- x3 - mean(x3)

# Calculate the cosine of the angle using the dot product
cos_angle <- sum(x1_dev * x3_dev) / (sqrt(sum(x1_dev^2)) * sqrt(sum(x3_dev^2)))
cos_angle
```

```
## [1] 0.0150425
```

```
# Calculate the angle in radians
angle_radians <- acos(cos_angle)

# Convert the angle to degrees
```

```r
angle_degrees <- angle_radians * (180 / pi)

angle_degrees
```

```
## [1] 89.1381
```

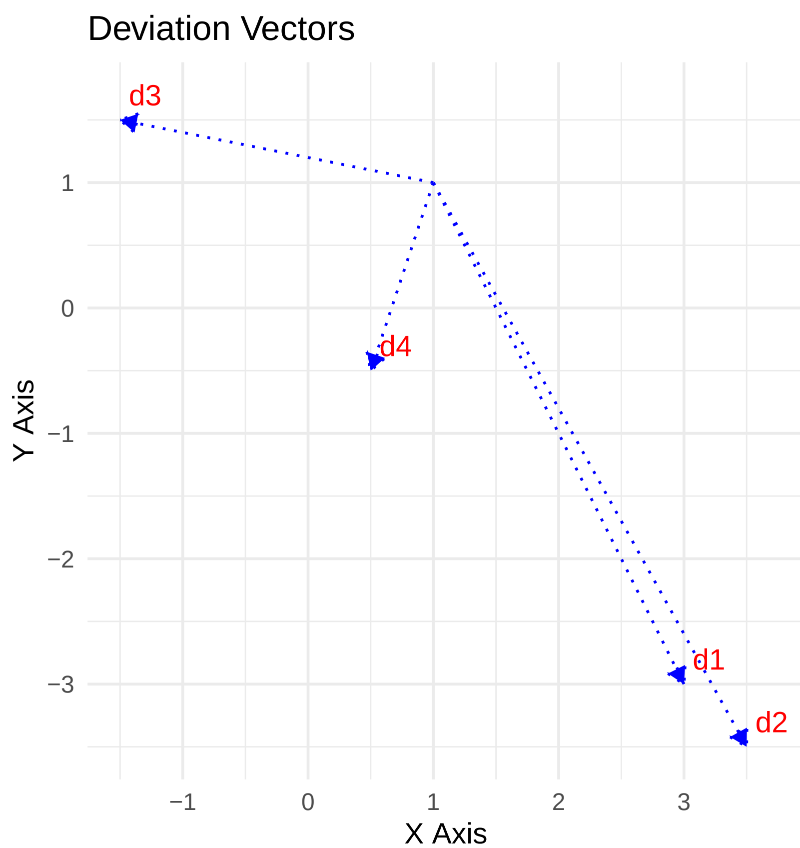
```r
#Qtn 3 Bonus Qtn
period1_obs2 <- data_period_1[1:2,]

x1_dev <- period1_obs2$MaxBreadth - mean(period1_obs2$MaxBreadth)
x2_dev <- period1_obs2$BasHeight - mean(period1_obs2$BasHeight)
x3_dev <- period1_obs2$BasLength - mean(period1_obs2$BasLength)
x4_dev <- period1_obs2$NasHeight - mean(period1_obs2$NasHeight)

dev_vectors <- rbind(c(x1_dev), c(x2_dev), c(x3_dev), c(x4_dev))
dev_vectors
```

```
##      [,1] [,2]
## [1,]  3.0 -3.0
## [2,]  3.5 -3.5
## [3,] -1.5  1.5
## [4,]  0.5 -0.5
```

```r
dev_vectors_df <- data.frame(
        "x" = dev_vectors[,1],
        'y' = dev_vectors[,2],
        'vector' = c('d1', 'd2', 'd3', 'd4'))
dev_vectors_df
```

```
##      x    y vector
## 1  3.0 -3.0     d1
## 2  3.5 -3.5     d2
## 3 -1.5  1.5     d3
## 4  0.5 -0.5     d4
```

```r
#Qtn 3 Bonus Qtn ctnd
# Plot vectors
ggplot(dev_vectors_df, aes(xend = x, yend = y)) +
  geom_segment(aes(x = 1, y = 1, xend = x, yend = y),
               arrow = arrow(type = "closed", length = unit(0.1, "inches")),
               color = "blue",
               linetype = "dotted") +
  geom_text(aes(x = x, y = y, label = vector), nudge_x = 0.2, nudge_y = 0.2, color = "red") +
  coord_fixed() +
  theme_minimal() +
  labs(title = "Deviation Vectors", x = "X Axis", y = "Y Axis")
```




```
#Qtn 4
b <- c(-1,0,0,3)
means_matrix <- as.matrix(mean_vectors[, 2:ncol(mean_vectors)])
y_means <- means_matrix%*%b
#means for y1 to y5
y_means
```

```
##          [,1]
## [1,] 20.23333
## [2,] 18.33333
## [3,] 17.23333
## [4,] 20.40000
## [5,] 17.93333
```

```
#now calculating the covariance matrix
data = data %>% mutate(Y = 3*NasHeight - MaxBreadth) #the y value for each data data point
y_data <- data %>% select(TimePeriod, Y)

#create an index to match data points ie 30 data points for 5 periods
index <- rep(seq(1,30), times = 5)
y_data$index <- index
y_data_wide <- pivot_wider(y_data, names_from = TimePeriod, values_from = Y) #pivot data to use cov fun

head(y_data_wide)
```

```
## # A tibble: 6 x 6
```

```
##   index   `1`   `2`   `3`   `4`   `5`
##   <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1    16    20    19    25    13
## 2     2    19    11    12    18    11
## 3     3    19    -3    12     6    43
## 4     4    13     5    20    18    26
## 5     5    26     9     1     5     3
## 6     6    30    21    10    19    30
```

```
y_data_wide <- y_data_wide %>% select(-index) #remove the index from cov calculation
y_covariances <- cov(y_data_wide)
y_covariances
```

```
##            1         2         3          4           5
## 1 51.5643678  1.402299  23.32299  -7.924138   0.2229885
## 2  1.4022989 90.712644  -6.08046  29.862069   2.8850575
## 3 23.3229885 -6.080460 120.11609 -10.131034 -33.4321839
## 4 -7.9241379 29.862069 -10.13103  74.731034  14.6482759
## 5  0.2229885  2.885057 -33.43218  14.648276 165.0298851
```