

Contents

1	Phân tích Tương quan và Heatmap	1
1.1	Chuẩn bị dữ liệu	1
1.2	Tương quan giữa các biến Hóa lý	2
1.3	Tương quan giữa NIR và Biến Hóa lý	4
1.4	Tương quan trong không gian PCA	12
1.5	Tương quan theo Location	14
1.6	Hierarchical Clustering dựa trên Correlation	18
1.7	Tổng hợp và Kết luận	19

1 Phân tích Tương quan và Heatmap

Phân tích tương quan giúp chúng ta hiểu mối quan hệ giữa các biến trong dữ liệu:

1. **Tương quan giữa các biến hóa lý:** Xác định biến nào có quan hệ chặt chẽ
2. **Tương quan giữa NIR và biến hóa lý:** Tìm vùng wavelength quan trọng
3. **Tương quan trong không gian PCA:** Phân tích trên principal components

1.1 Chuẩn bị dữ liệu

```
library(tidyverse)
library(corrplot)
library(pheatmap)
library(RColorBrewer)
library(knitr)
library(kableExtra)
library(reshape2)

# Đọc dữ liệu
coffee_data <- read.csv("coffee_nirs.csv", sep = ";", row.names = 1, stringsAsFactors = FALSE)

# Chuyển đổi sang numeric
convert_to_numeric <- function(x) {
  if(is.character(x)) {
    x <- gsub("\\\\.", "", x)
    x <- gsub(",", ".", x)
    return(as.numeric(x))
  }
  return(x)
}

for(col in names(coffee_data)) {
  if(col != "Localisation") {
    coffee_data[[col]] <- convert_to_numeric(coffee_data[[col]])
  }
}
```

Table 1: Ma trận tương quan Pearson - Biến Hóa lý

	CGA	Cafeine	Fat	Trigonelline	DM
CGA	1.000	0.007	0.089	0.118	-0.035
Cafeine	0.007	1.000	0.006	0.017	0.089
Fat	0.089	0.006	1.000	0.015	-0.024
Trigonelline	0.118	0.017	0.015	1.000	0.067
DM	-0.035	0.089	-0.024	0.067	1.000

```
}

# Định nghĩa các nhóm biến
chemical_vars <- c("CGA", "Cafeine", "Fat", "Trigonelline", "DM")
nir_vars <- grep("^S[0-9]+$", names(coffee_data), value = TRUE)
```

1.2 Tương quan giữa các biến Hóa lý

1.2.1 Ma trận tương quan

```
# Lấy dữ liệu hóa lý hoàn chỉnh
data_chem <- coffee_data[, chemical_vars]
data_chem_complete <- data_chem[complete.cases(data_chem), ]

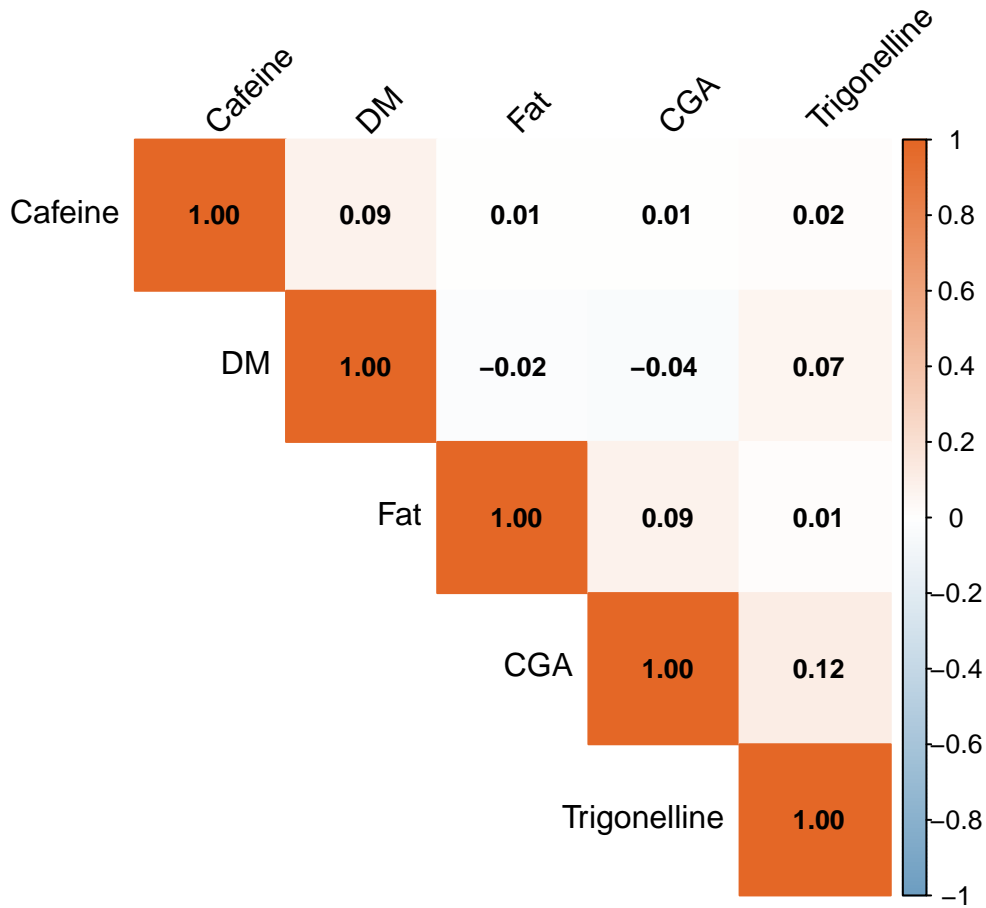
# Tính ma trận tương quan
cor_matrix <- cor(data_chem_complete, method = "pearson")

# Hiển thị bảng
cor_matrix %>%
  round(3) %>%
  kable(caption = "Ma trận tương quan Pearson - Biến Hóa lý") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

1.2.2 Heatmap tương quan - Biến Hóa lý

```
# Heatmap với corrplot
corrplot(cor_matrix,
  method = "color",
  type = "upper",
  order = "hclust",
  addCoef.col = "black",
  tl.col = "black",
  tl.srt = 45,
  number.cex = 0.8,
  col = colorRampPalette(c("#6D9EC1", "white", "#E46726"))(200),
  title = "Correlation Heatmap - Chemical Variables",
  mar = c(0, 0, 2, 0))
```

Correlation Heatmap – Chemical Variables



1.2.3 Phân tích ý nghĩa

```
# Tìm các cặp biến có tương quan cao ( $|r| > 0.5$ )
cor_df <- as.data.frame(as.table(cor_matrix))
names(cor_df) <- c("Var1", "Var2", "Correlation")

high_corr <- cor_df %>%
  filter(Var1 != Var2) %>%
  filter(abs(Correlation) > 0.5) %>%
  arrange(desc(abs(Correlation))) %>%
  distinct(Correlation, .keep_all = TRUE)

if(nrow(high_corr) > 0) {
  cat("Các cặp biến có tương quan mạnh ( $|r| > 0.5$ ):\n\n")
  high_corr %>%
```

```

    mutate(Correlation = round(Correlation, 3)) %>%
    kable() %>%
    kable_styling(bootstrap_options = c("striped", "hover"))
} else {
  cat("Không có cặp biến nào có tương quan mạnh ( $|r| > 0.5$ )\n")
}

```

Không có cặp biến nào có tương quan mạnh ($|r| > 0.5$)

1.3 Tương quan giữa NIR và Biến Hóa lý

1.3.1 Tính toán tương quan

```

# Lấy dữ liệu đầy đủ
data_nir <- coffee_data[, nir_vars]
complete_rows <- complete.cases(cbind(data_chem, data_nir))

data_chem_full <- data_chem[complete_rows, ]
data_nir_full <- data_nir[complete_rows, ]

# Tính tương quan giữa từng biến hóa lý và tất cả NIR wavelengths
cor_nir_chem <- cor(data_nir_full, data_chem_full, method = "pearson")

cat("Kích thước ma trận tương quan NIR-Chemical:", dim(cor_nir_chem), "\n")

```

Kích thước ma trận tương quan NIR-Chemical: 1050 5

```
cat("Số wavelengths:", nrow(cor_nir_chem), "\n")
```

Số wavelengths: 1050

```
cat("Số biến hóa lý:", ncol(cor_nir_chem), "\n")
```

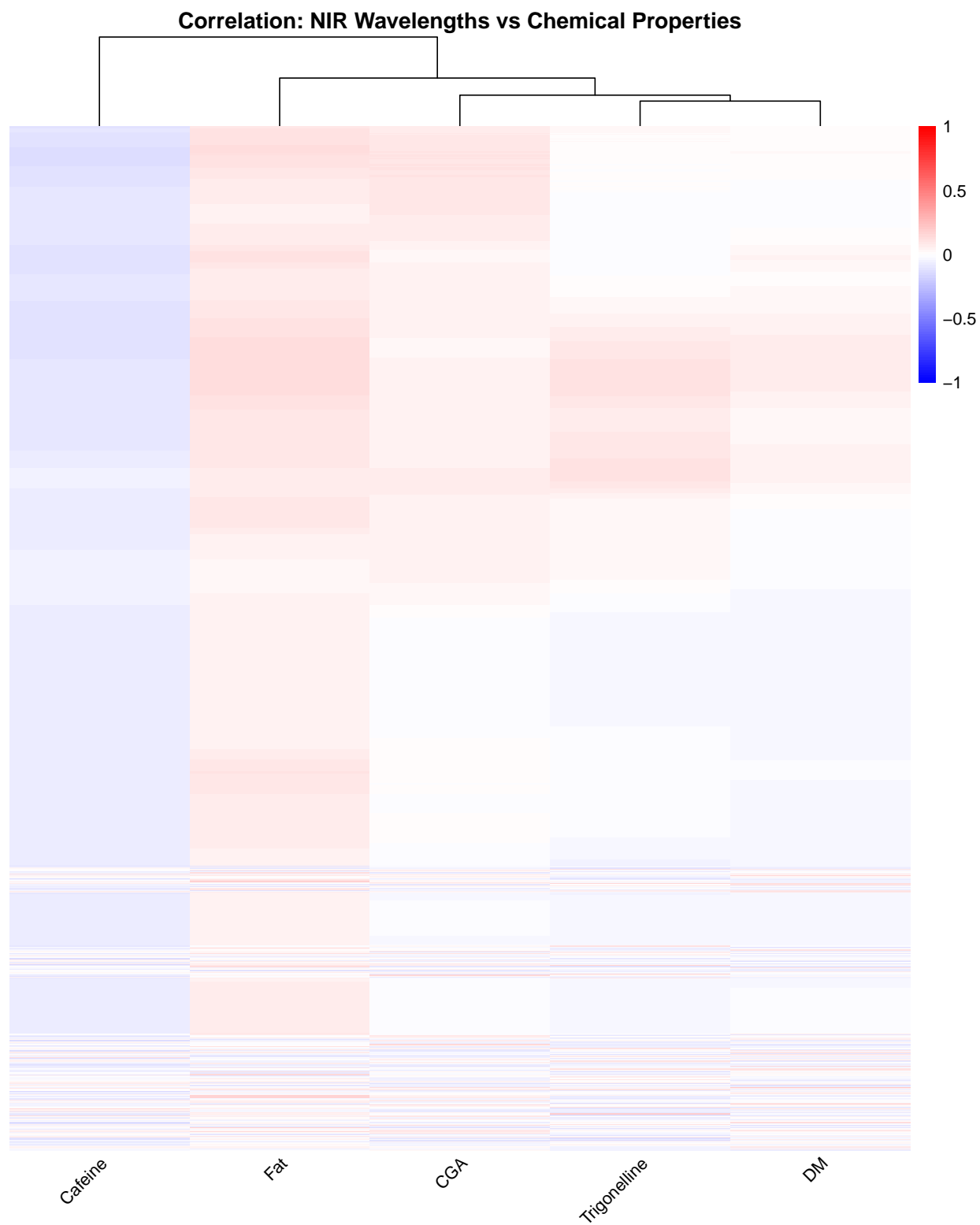
Số biến hóa lý: 5

1.3.2 Heatmap tổng quan NIR-Chemical

```

# Heatmap cho toàn bộ correlation
pheatmap(cor_nir_chem,
  cluster_rows = FALSE,
  cluster_cols = TRUE,
  show_rownames = FALSE,
  main = "Correlation: NIR Wavelengths vs Chemical Properties",
  color = colorRampPalette(c("blue", "white", "red"))(100),
  breaks = seq(-1, 1, length.out = 101),
  fontsize = 10,
  angle_col = 45)

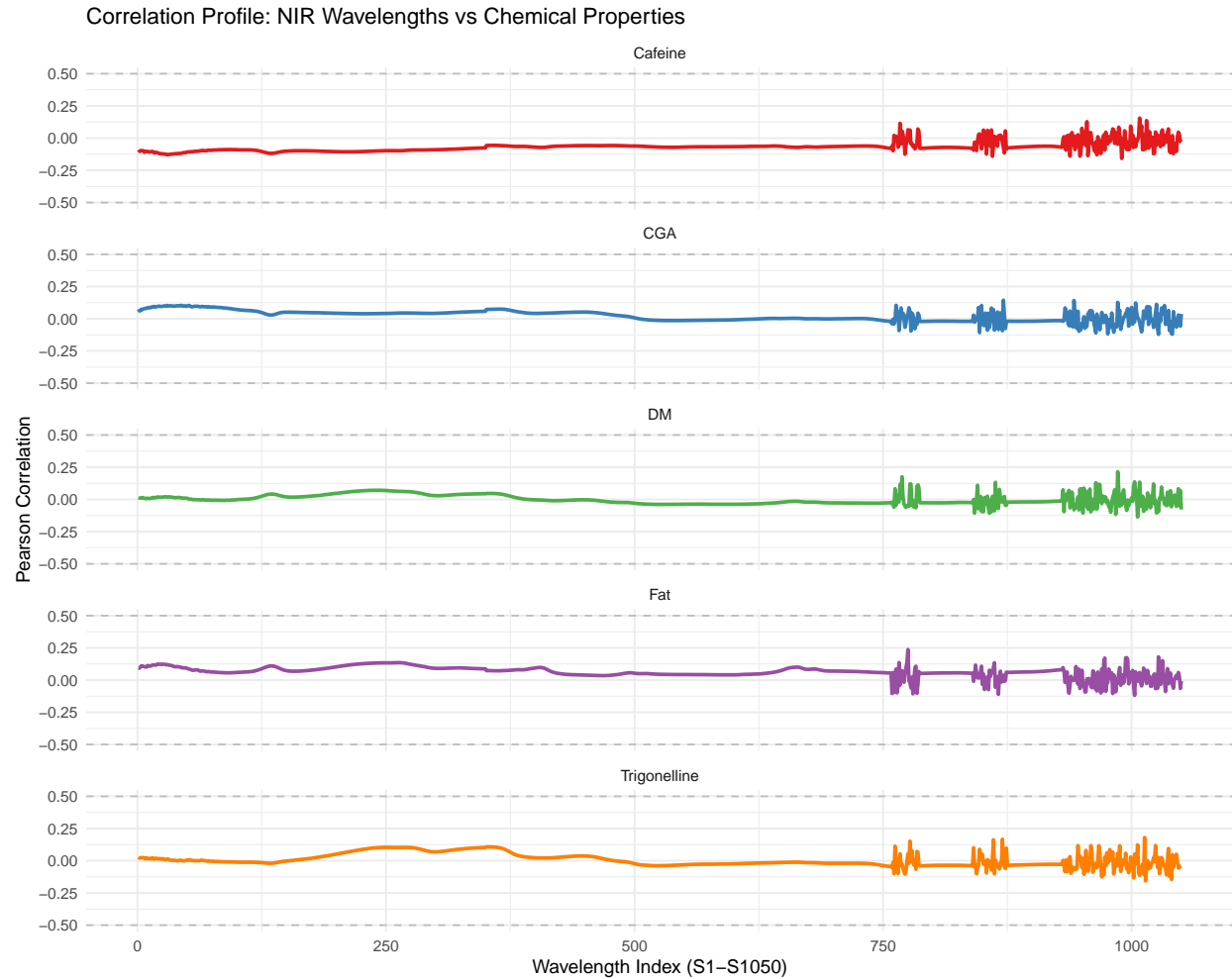
```



1.3.3 Biểu đồ Line Plot - Correlation profile

```
# Chuyển sang long format để vẽ
cor_long <- as.data.frame(cor_nir_chem) %>%
  mutate(Wavelength = 1:nrow(cor_nir_chem)) %>%
  pivot_longer(cols = -Wavelength, names_to = "Chemical", values_to = "Correlation")

# Line plot cho từng biến hóa lý
ggplot(cor_long, aes(x = Wavelength, y = Correlation, color = Chemical)) +
  geom_line(linewidth = 1) +
  geom_hline(yintercept = c(-0.5, 0.5), linetype = "dashed", color = "gray") +
  facet_wrap(~Chemical, ncol = 1, scales = "free_y") +
  scale_color_brewer(palette = "Set1") +
  labs(
    title = "Correlation Profile: NIR Wavelengths vs Chemical Properties",
    x = "Wavelength Index (S1-S1050)",
    y = "Pearson Correlation",
    color = "Chemical Variable"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```



1.3.4 Xác định wavelengths quan trọng

```
# Tìm wavelengths có tương quan cao với mỗi biến hóa lý
important_wavelengths <- list()

for(chem_var in chemical_vars) {
  # Lấy tương quan tuyệt đối
  abs_corr <- abs(cor_nir_chem[, chem_var])

  # Tìm top 10 wavelengths
  top_idx <- order(abs_corr, decreasing = TRUE)[1:10]

  important_wavelengths[[chem_var]] <- data.frame(
    Chemical = chem_var,
    Wavelength = nir_vars[top_idx],
    Wavelength_Index = top_idx,
    Correlation = cor_nir_chem[top_idx, chem_var],
    Abs_Correlation = abs_corr[top_idx]
  )
}
```

```
# Kết hợp tất cả
all_important <- do.call(rbind, important_wavelengths)

cat("Top 10 wavelengths có tương quan mạnh nhất với từng biến hóa lý:\n\n")
```

Top 10 wavelengths có tương quan mạnh nhất với từng biến hóa lý:

```
for(chem_var in chemical_vars) {
  cat("\n**", chem_var, "：**\n")
  all_important %>%
    filter(Chemical == chem_var) %>%
    select(Wavelength, Wavelength_Index, Correlation) %>%
    mutate(Correlation = round(Correlation, 3)) %>%
    head(5) %>%
    kable() %>%
    kable_styling(bootstrap_options = c("striped", "hover")) %>%
    print()
}
```

```
##
## ** CGA :**
## \begin{table}
## \centering
## \begin{tabular}{l|l|r|r}
## \hline
##   & Wavelength & Wavelength_Index & Correlation\\
## \hline
## CGA.S871 & S871 & 871 & 0.144\\
## \hline
## CGA.S942 & S942 & 942 & 0.142\\
## \hline
## CGA.S986 & S986 & 986 & 0.128\\
## \hline
## CGA.S1004 & S1004 & 1004 & 0.123\\
## \hline
## CGA.S1027 & S1027 & 1027 & -0.121\\
## \hline
## \end{tabular}
## \end{table}
##
## ** Caffeine :**
## \begin{table}
## \centering
## \begin{tabular}{l|l|r|r}
## \hline
##   & Wavelength & Wavelength_Index & Correlation\\
## \hline
## Caffeine.S990 & S990 & 990 & -0.158\\
## \hline
## Caffeine.S1008 & S1008 & 1008 & 0.156\\
## \hline
## Caffeine.S860 & S860 & 860 & -0.141\\
```



```

## \hline
## Caffeine.S948 & S948 & 948 & -0.140\\
## \hline
## Caffeine.S938 & S938 & 938 & -0.140\\
## \hline
## \end{tabular}
## \end{table}
##
## ** Fat :**
## \begin{table}
## \centering
## \begin{tabular}{l|l|r|r}
## \hline
## & Wavelength & Wavelength\_Index & Correlation\\
## \hline
## Fat.S775 & S775 & 775 & 0.238\\
## \hline
## Fat.S1027 & S1027 & 1027 & 0.180\\
## \hline
## Fat.S995 & S995 & 995 & 0.174\\
## \hline
## Fat.S994 & S994 & 994 & 0.172\\
## \hline
## Fat.S973 & S973 & 973 & 0.171\\
## \hline
## \end{tabular}
## \end{table}
##
## ** Trigonelline :**
## \begin{table}
## \centering
## \begin{tabular}{l|l|r|r}
## \hline
## & Wavelength & Wavelength\_Index & Correlation\\
## \hline
## Trigonelline.S1013 & S1013 & 1013 & 0.181\\
## \hline
## Trigonelline.S870 & S870 & 870 & 0.167\\
## \hline
## Trigonelline.S861 & S861 & 861 & 0.163\\
## \hline
## Trigonelline.S1014 & S1014 & 1014 & -0.157\\
## \hline
## Trigonelline.S777 & S777 & 777 & 0.152\\
## \hline
## \end{tabular}
## \end{table}
##
## ** DM :**
## \begin{table}
## \centering
## \begin{tabular}{l|l|r|r}
## \hline
## & Wavelength & Wavelength\_Index & Correlation\\

```

```
## \hline
## DM.S986 & S986 & 986 & 0.215\\
## \hline
## DM.S769 & S769 & 769 & 0.176\\
## \hline
## DM.S1006 & S1006 & 1006 & -0.137\\
## \hline
## DM.S1003 & S1003 & 1003 & 0.136\\
## \hline
## DM.S1008 & S1008 & 1008 & 0.135\\
## \hline
## \end{tabular}
## \end{table}
```

1.3.5 Heatmap cho wavelengths quan trọng nhất

```
# Lấy top 50 wavelengths có tương quan cao nhất (cho bất kỳ biến nào)
max_abs_corr <- apply(abs(cor_nir_chem), 1, max)
top_50_idx <- order(max_abs_corr, decreasing = TRUE)[1:50]

# Heatmap cho top wavelengths
pheatmap(cor_nir_chem[top_50_idx, ],
          cluster_rows = TRUE,
          cluster_cols = TRUE,
          show_rownames = TRUE,
          labels_row = paste0("S", top_50_idx),
          main = "Top 50 Most Correlated Wavelengths",
          color = colorRampPalette(c("blue", "white", "red"))(100),
          breaks = seq(-1, 1, length.out = 101),
          fontsize = 8,
          angle_col = 45)
```

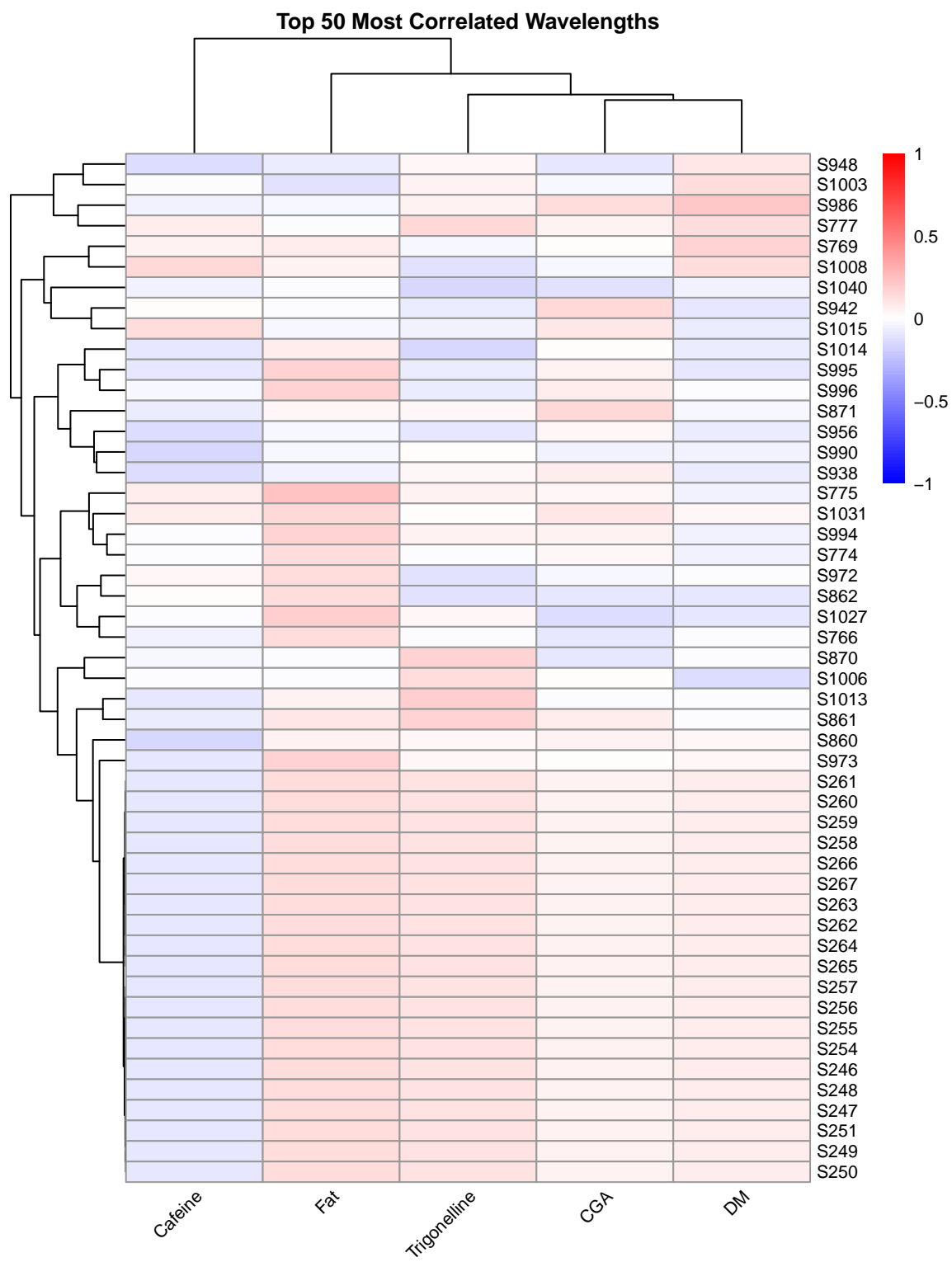


Table 2: Correlation: PC Scores vs Chemical Properties

	CGA	Cafeine	Fat	Trigonelline	DM
Dim.1	0.029	-0.087	0.080	0.007	-0.002
Dim.2	0.063	-0.033	0.063	0.139	0.085
Dim.3	-0.092	0.078	-0.059	-0.012	-0.062
Dim.4	-0.207	-0.038	0.057	-0.013	0.033
Dim.5	0.105	-0.077	0.177	0.061	0.034

1.4 Tương quan trong không gian PCA

```
# Thực hiện PCA trên NIR
library(FactoMineR)
pca_nir <- PCA(data_nir_full, scale.unit = TRUE, graph = FALSE)

# Lấy PC scores (10 PC đầu tiên)
n_pcs <- min(10, ncol(pca_nir$ind$coord))
pc_scores <- pca_nir$ind$coord[, 1:n_pcs]

# Tính tương quan giữa PC scores và biến hóa lý
cor_pc_chem <- cor(pc_scores, data_chem_full, method = "pearson")

cat("Ma trận tương quan PC-Chemical:\n")
```

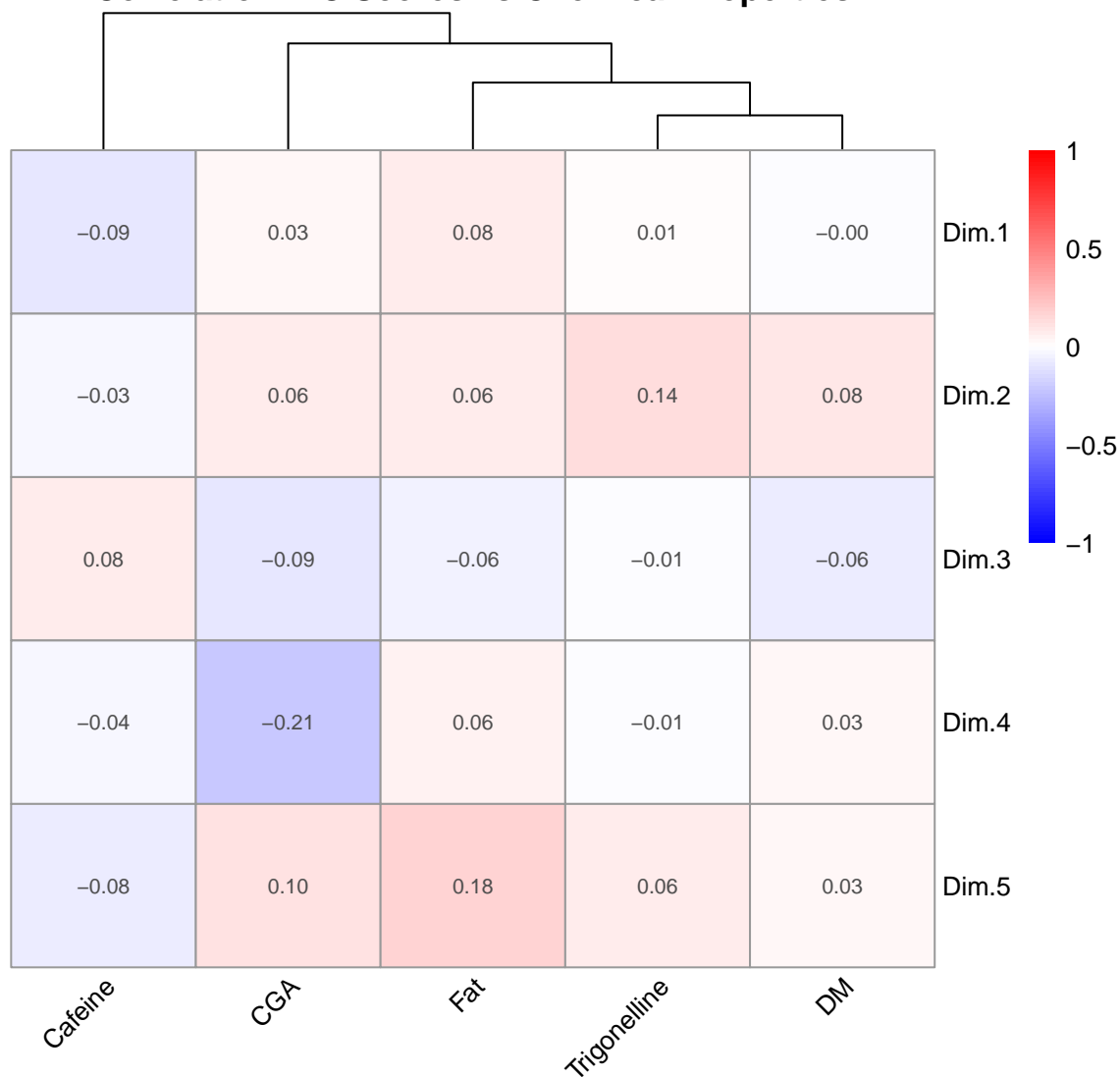
Ma trận tương quan PC-Chemical:

```
cor_pc_chem %>%
  round(3) %>%
  kable(caption = "Correlation: PC Scores vs Chemical Properties") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

1.4.1 Heatmap PC-Chemical

```
# Heatmap
pheatmap(cor_pc_chem,
  cluster_rows = FALSE,
  cluster_cols = TRUE,
  display_numbers = TRUE,
  number_format = "%.2f",
  main = "Correlation: PC Scores vs Chemical Properties",
  color = colorRampPalette(c("blue", "white", "red"))(100),
  breaks = seq(-1, 1, length.out = 101),
  fontsize = 10,
  angle_col = 45)
```

Correlation: PC Scores vs Chemical Properties



1.4.2 Phân tích ý nghĩa

```
# Tìm PC có tương quan mạnh nhất với mỗi biến hóa lý
for(chem_var in chemical_vars) {
  pc_corrs <- cor_pc_chem[, chem_var]
  max_pc <- which.max(abs(pc_corrs))
  max_corr <- pc_corrs[max_pc]

  cat(sprintf("**%s** : Tương quan mạnh nhất với PC%d (r = %.3f)\n",
              chem_var, max_pc, max_corr))
}
```

```
## **CGA** : Tương quan mạnh nhất với PC4 (r = -0.207)
## **Cafeine** : Tương quan mạnh nhất với PC1 (r = -0.087)
## **Fat** : Tương quan mạnh nhất với PC5 (r = 0.177)
```

```
## **Trigonelline**:
```

 Tương quan mạnh nhất với PC2 ($r = 0.139$)
DM: Tương quan mạnh nhất với PC2 ($r = 0.085$)

```
cat("\n**Ý nghĩa:**\n")
```

```
##  
## **Ý nghĩa:**
```

```
cat("- PC có tương quan cao với biến hóa lý cho thấy thành phần chính đó\n")
```

```
## - PC có tương quan cao với biến hóa lý cho thấy thành phần chính đó
```

```
cat("  chứa thông tin quan trọng để dự đoán biến đó\n")
```

```
##  chứa thông tin quan trọng để dự đoán biến đó
```

```
cat("- Có thể sử dụng PC scores làm biến độc lập trong mô hình hồi quy\n")
```

```
## - Có thể sử dụng PC scores làm biến độc lập trong mô hình hồi quy
```

1.5 Tương quan theo Location

```
# Tính tương quan riêng cho từng location  
location_info <- coffee_data$Localisation[complete_rows]  
unique_locations <- unique(location_info)
```

```
cat("Phân tích tương quan theo từng Location:\n\n")
```

```
## Phân tích tương quan theo từng Location:
```

```
for(loc in unique_locations) {  
  loc_idx <- which(location_info == loc)  
  
  if(length(loc_idx) > 5) { # Chỉ phân tích nếu có đủ mẫu  
    data_chem_loc <- data_chem_full[loc_idx, ]  
    cor_loc <- cor(data_chem_loc, method = "pearson")  
  
    cat(sprintf("\n### Location %s (n = %d samples):\n", loc, length(loc_idx)))  
  
    # Hiển thị ma trận  
    cor_loc %>%  
      round(3) %>%  
      kable() %>%  
      kable_styling(bootstrap_options = c("striped", "hover")) %>%  
      print()  
  }  
}
```

```

##
## ### Location 1 (n = 50 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
##   & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & 0.029 & -0.002 & 0.340 & -0.104\\
## \hline
## Caffeine & 0.029 & 1.000 & 0.189 & -0.033 & 0.424\\
## \hline
## Fat & -0.002 & 0.189 & 1.000 & -0.064 & -0.051\\
## \hline
## Trigonelline & 0.340 & -0.033 & -0.064 & 1.000 & -0.072\\
## \hline
## DM & -0.104 & 0.424 & -0.051 & -0.072 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 6 (n = 84 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
##   & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & -0.061 & 0.083 & 0.065 & -0.051\\
## \hline
## Caffeine & -0.061 & 1.000 & 0.089 & -0.014 & -0.013\\
## \hline
## Fat & 0.083 & 0.089 & 1.000 & 0.014 & -0.041\\
## \hline
## Trigonelline & 0.065 & -0.014 & 0.014 & 1.000 & 0.091\\
## \hline
## DM & -0.051 & -0.013 & -0.041 & 0.091 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 4 (n = 13 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
##   & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & 0.092 & -0.228 & 0.345 & -0.347\\
## \hline
## Caffeine & 0.092 & 1.000 & 0.022 & -0.136 & -0.019\\
## \hline
## Fat & -0.228 & 0.022 & 1.000 & -0.110 & -0.080\\
## \hline

```

```

## Trigonelline & 0.345 & -0.136 & -0.110 & 1.000 & -0.186\\
## \hline
## DM & -0.347 & -0.019 & -0.080 & -0.186 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 7 (n = 19 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
## & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & 0.061 & 0.938 & -0.090 & -0.218\\
## \hline
## Caffeine & 0.061 & 1.000 & -0.125 & -0.055 & -0.226\\
## \hline
## Fat & 0.938 & -0.125 & 1.000 & 0.021 & 0.014\\
## \hline
## Trigonelline & -0.090 & -0.055 & 0.021 & 1.000 & 0.162\\
## \hline
## DM & -0.218 & -0.226 & 0.014 & 0.162 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 2 (n = 26 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
## & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & -0.157 & -0.112 & -0.072 & -0.051\\
## \hline
## Caffeine & -0.157 & 1.000 & -0.211 & 0.449 & -0.123\\
## \hline
## Fat & -0.112 & -0.211 & 1.000 & 0.218 & -0.168\\
## \hline
## Trigonelline & -0.072 & 0.449 & 0.218 & 1.000 & -0.001\\
## \hline
## DM & -0.051 & -0.123 & -0.168 & -0.001 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 3 (n = 26 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
## & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline

```



```

## CGA & 1.000 & 0.361 & 0.224 & 0.069 & 0.311\\
## \hline
## Caffeine & 0.361 & 1.000 & -0.223 & 0.564 & 0.277\\
## \hline
## Fat & 0.224 & -0.223 & 1.000 & -0.073 & 0.281\\
## \hline
## Trigonelline & 0.069 & 0.564 & -0.073 & 1.000 & -0.033\\
## \hline
## DM & 0.311 & 0.277 & 0.281 & -0.033 & 1.000\\
## \hline
## \end{tabular}
## \end{table}
##
## ### Location 5 (n = 22 samples):
## \begin{table}
## \centering
## \begin{tabular}{l|r|r|r|r|r}
## \hline
## & CGA & Caffeine & Fat & Trigonelline & DM\\
## \hline
## CGA & 1.000 & 0.029 & 0.049 & -0.153 & 0.217\\
## \hline
## Caffeine & 0.029 & 1.000 & -0.206 & -0.171 & 0.133\\
## \hline
## Fat & 0.049 & -0.206 & 1.000 & 0.058 & -0.050\\
## \hline
## Trigonelline & -0.153 & -0.171 & 0.058 & 1.000 & 0.503\\
## \hline
## DM & 0.217 & 0.133 & -0.050 & 0.503 & 1.000\\
## \hline
## \end{tabular}
## \end{table}

```

1.5.1 So sánh pattern tương quan giữa các Location

```

# Tạo heatmap cho mỗi location
par(mfrow = c(2, ceiling(length(unique_locations) / 2)))

for(loc in unique_locations) {
  loc_idx <- which(location_info == loc)

  if(length(loc_idx) > 5) {
    data_chem_loc <- data_chem_full[loc_idx, ]
    cor_loc <- cor(data_chem_loc, method = "pearson")

    corrplot(cor_loc,
              method = "color",
              type = "upper",
              addCoef.col = "black",
              tl.col = "black",
              tl.srt = 45,
              number.cex = 0.6,

```

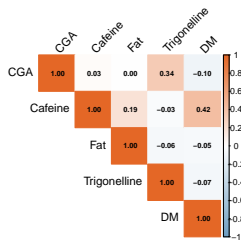
```

col = colorRampPalette(c("#6D9EC1", "white", "#E46726"))(200),
title = paste("Location", loc, "-", length(loc_idx), "samples"),
mar = c(0, 0, 2, 0))
}
}

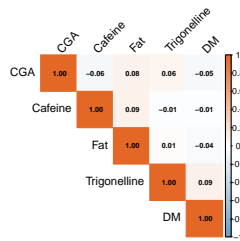
par(mfrow = c(1, 1))

```

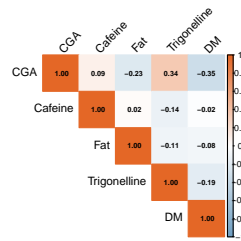
Location 1 – 50 samples



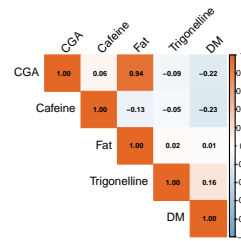
Location 6 – 84 samples



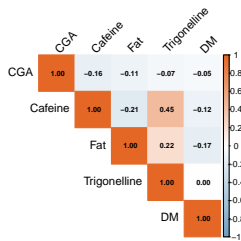
Location 4 – 13 samples



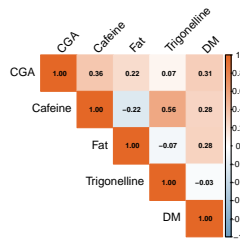
Location 7 – 19 samples



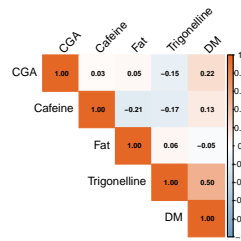
Location 2 – 26 samples



Location 3 – 26 samples



Location 5 – 22 samples



1.6 Hierarchical Clustering dựa trên Correlation

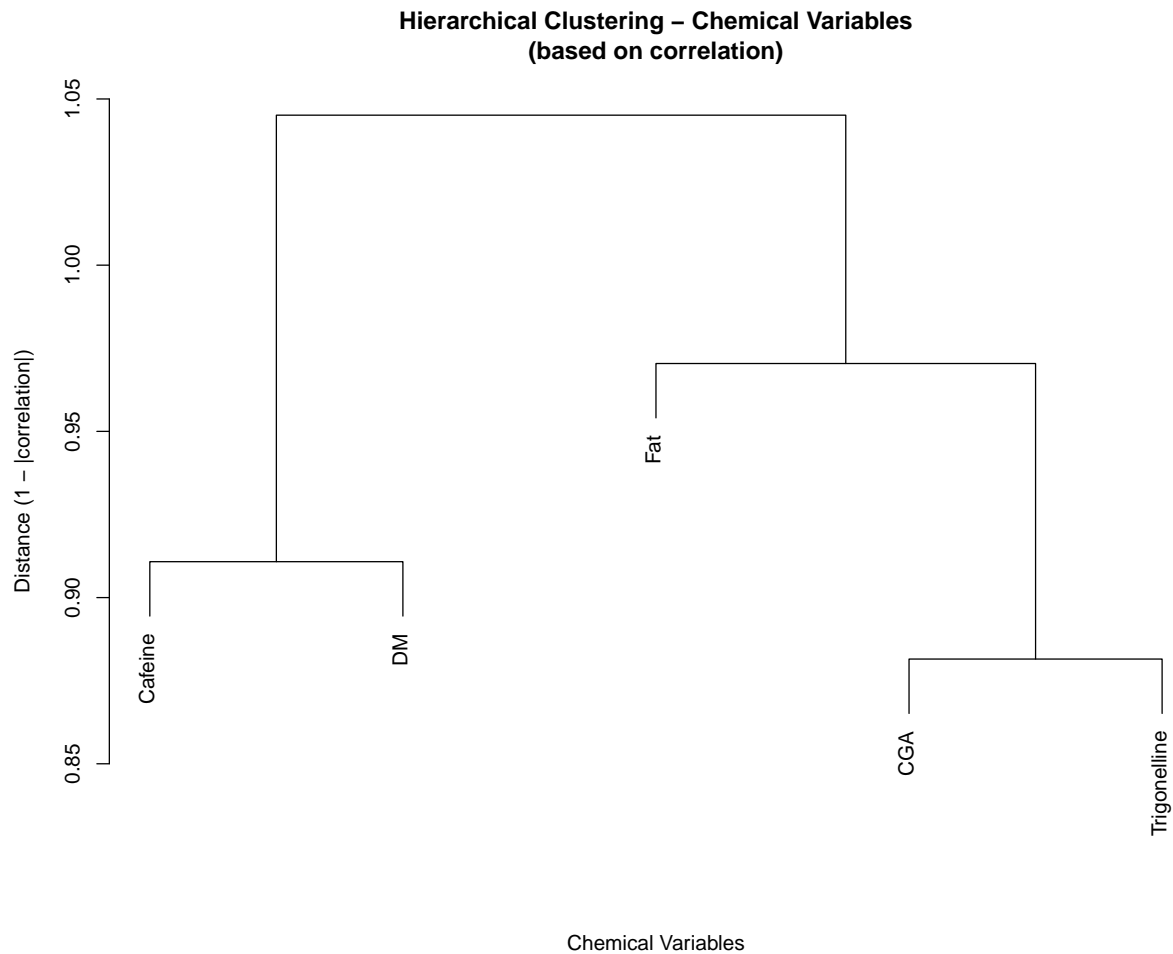
```

# Clustering các biến hóa lý dựa trên correlation
dist_chem <- as.dist(1 - abs(cor_matrix))
hc_chem <- hclust(dist_chem, method = "ward.D2")

# Dendrogram
plot(hc_chem,
     main = "Hierarchical Clustering - Chemical Variables\n(based on correlation)",

```

```
xlab = "Chemical Variables",
ylab = "Distance (1 - |correlation|)",
sub = "")
```



1.7 Tổng hợp và Kết luận

```
cat("### Tóm tắt phân tích tương quan:\n\n")
```

```
## ### Tóm tắt phân tích tương quan:
```

```
cat("**1. Tương quan giữa biến hóa lý:**\n")
```

```
## **1. Tương quan giữa biến hóa lý:**
```

```
# Tìm tương quan cao nhất
```

```
max_corr_idx <- which(abs(cor_matrix) == max(abs(cor_matrix[upper.tri(cor_matrix)])), arr.ind = TRUE)[1]
cat(sprintf("- Tương quan mạnh nhất: %s vs %s (r = %.3f)\n",
```

```
rownames(cor_matrix)[max_corr_idx[1]],
colnames(cor_matrix)[max_corr_idx[2]],
cor_matrix[max_corr_idx[1], max_corr_idx[2]])
```

```
## - Tương quan mạnh nhất: Trigonelline vs CGA (r = 0.118)
```

```
cat("\n**2. NIR wavelengths quan trọng:**\n")
```

```
##
```

```
## **2. NIR wavelengths quan trọng:**
```

```
for(chem_var in chemical_vars) {
  top_wave <- all_important %>%
    filter(Chemical == chem_var) %>%
    slice_max(Abs_Correlation, n = 1)

  cat(sprintf("- %s: Wavelength %s (r = %.3f)\n",
              chem_var, top_wave$Wavelength, top_wave$Correlation))
}
```

```
## - CGA: Wavelength S871 (r = 0.144)
## - Caffeine: Wavelength S990 (r = -0.158)
## - Fat: Wavelength S775 (r = 0.238)
## - Trigonelline: Wavelength S1013 (r = 0.181)
## - DM: Wavelength S986 (r = 0.215)
```

```
cat("\n**3. PC scores hiệu quả:**\n")
```

```
##
```

```
## **3. PC scores hiệu quả:**
```

```
for(chem_var in chemical_vars) {
  pc_corrs <- cor_pc_chem[, chem_var]
  max_pc <- which.max(abs(pc_corrs))
  cat(sprintf("- %s: PC%d (r = %.3f)\n",
              chem_var, max_pc, pc_corrs[max_pc]))
}
```

```
## - CGA: PC4 (r = -0.207)
## - Caffeine: PC1 (r = -0.087)
## - Fat: PC5 (r = 0.177)
## - Trigonelline: PC2 (r = 0.139)
## - DM: PC2 (r = 0.085)
```

```
cat("\n**4. Khuyến nghị:**\n")
```

```
##
```

```
## **4. Khuyến nghị:**
```

```
cat("- Sử dụng wavelengths có tương quan cao để xây dựng mô hình đơn giản\n")
```

```
## - Sử dụng wavelengths có tương quan cao để xây dựng mô hình đơn giản
```

```
cat("- Cân nhắc sử dụng PC scores thay vì toàn bộ NIR để giảm chiều dữ liệu\n")
```

```
## - Cân nhắc sử dụng PC scores thay vì toàn bộ NIR để giảm chiều dữ liệu
```

```
cat("- Lưu ý sự khác biệt tương quan giữa các Location khi xây dựng mô hình\n")
```

```
## - Lưu ý sự khác biệt tương quan giữa các Location khi xây dựng mô hình
```

```
cat("- Các biến hóa lý có tương quan cao với nhau có thể gây multicollinearity\n")
```

```
## - Các biến hóa lý có tương quan cao với nhau có thể gây multicollinearity
```