Multivariate Analysis HW1

313657003 周佳萱

2025-03-27

Notes

The homework questions are from the exercises of Johnson and Wichern. *Applied Multivariate Statistical Analysis* (6th ed.). Upper Saddle River: Prentice Hall, 2007.

Data

In this homework, use the data in Table 4.6 (page 207) (`T4-6.csv`) for the analysis. Please refer to chapter 4, exercise 4.39 for the details of the data.

Questions

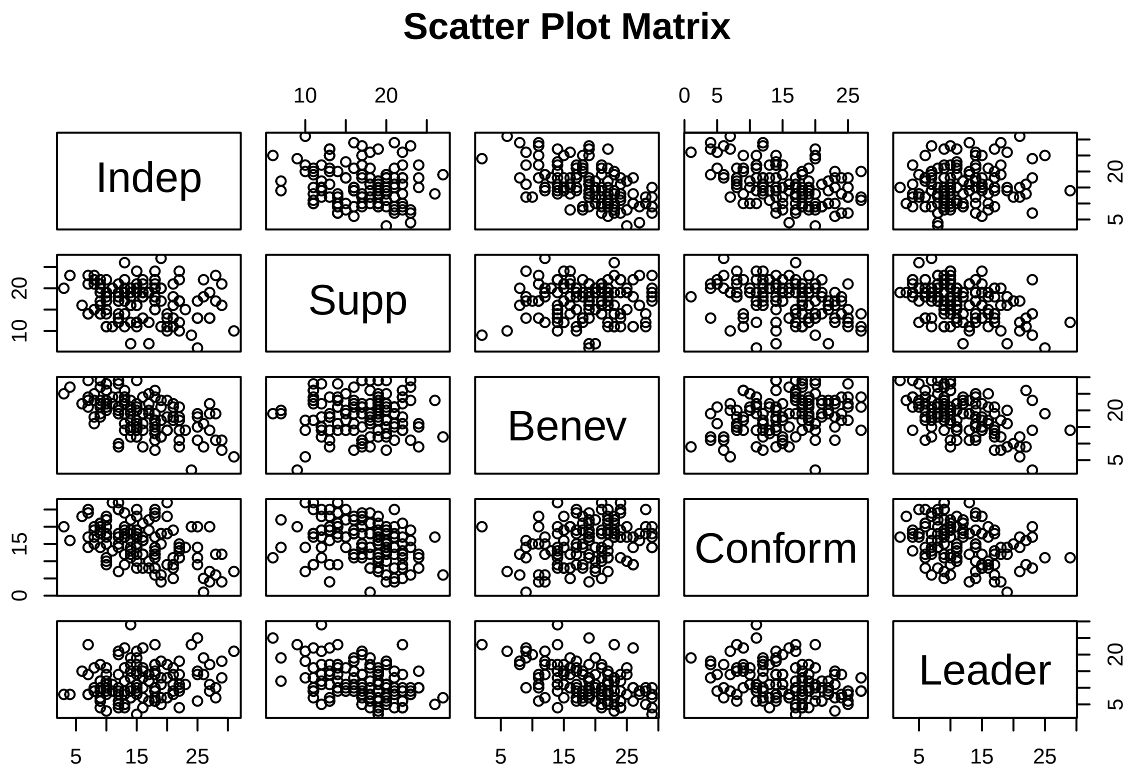
1. Chapter 1, exercise 1.19: use the variables independence, support, benevolence, conformity, and leadership for the analysis.(Create the scatter plot and boxplot.)

```
# Load necessary library
library(readr)

# Read the CSV file
file_path <- "T4-6.csv"
df <- read.csv(file_path, header = FALSE)

# Add column names
colnames(df) <- c("Indep", "Supp", "Benev", "Conform", "Leader", "Gender", "Socio")

# Display the updated dataframe
head(df)
```

```
##   Indep Supp Benev Conform Leader Gender Socio
## 1    27   13    14      20     11      2     1
## 2    12   13    24      25      6      2     1
## 3    14   20    15      16      7      2     1
## 4    18   20    17      12      6      2     1
## 5     9   22    22      21      6      2     1
## 6    18   15    17      25      9      2     1
```

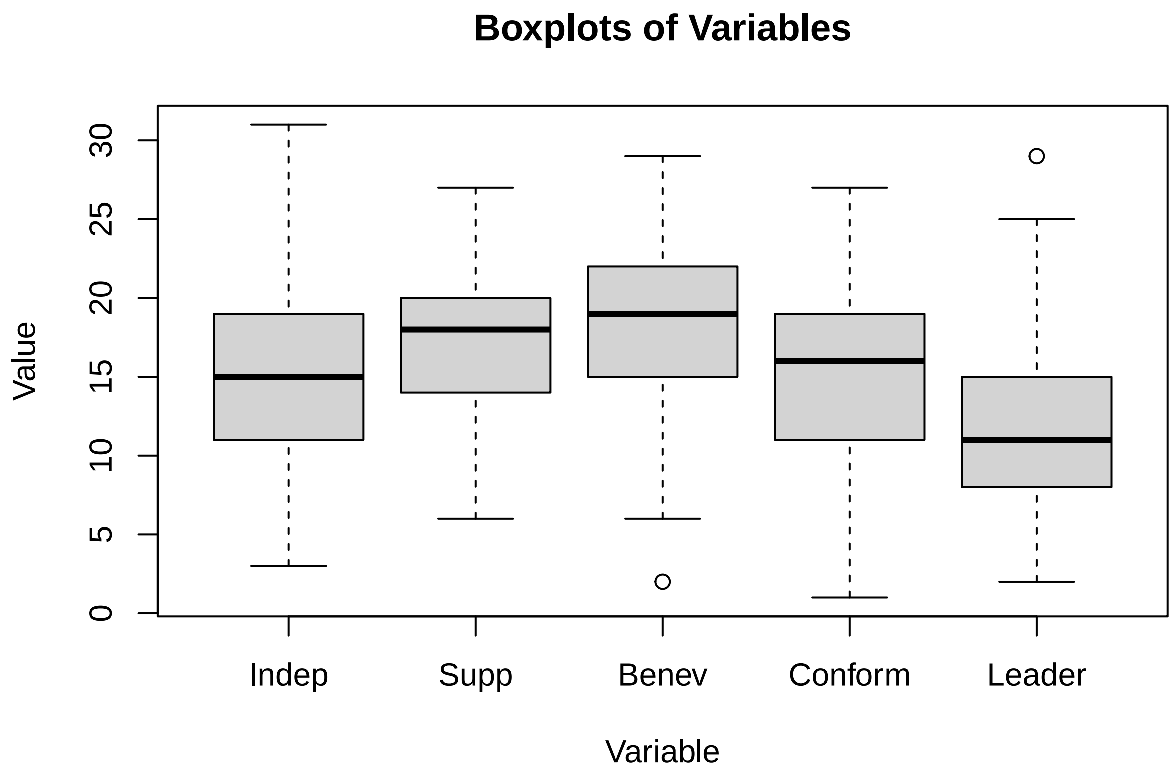
Scatter Plot

```r
# Create scatter plot matrix
pairs(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")], main="Scatter Plot Matrix")
```



Boxplot

```r
# Create boxplots for the selected variables
boxplot(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")],
        main = "Boxplots of Variables",
        xlab = "Variable",
        ylab = "Value",
        col = "lightgray")
```



2. Chapter 1, exercise 1.26 (a): use the variables independence, support, benevolence, conformity, and leadership for the analysis.(Compute the x_bar, Sn, and R arrays. Interpret the pairwise correlations. Do some of these variables appear to distinguish one breed from another?)

Compute Sample Means

```
# Compute means
x_bar <- colMeans(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
print(x_bar)
```

```
##    Indep     Supp    Benev  Conform   Leader
## 15.66923 17.07692 18.78462 15.50000 11.73077
```

Compute Sample Covariance Matrix

```
# Compute covariance matrix
Sn <- cov(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
print(Sn)
```

```
##               Indep        Supp       Benev     Conform      Leader
```

```
## Indep    34.750209 -4.2766846 -18.0717949 -15.972868   5.716458
## Supp     -4.276685 17.5134168   0.4197973  -7.868217  -8.723315
## Benev   -18.071795  0.4197973  29.8447227   9.348837 -13.942159
## Conform -15.972868 -7.8682171   9.3488372  33.042636  -9.941860
## Leader    5.716458 -8.7233154 -13.9421586  -9.941860  26.957961
```

Compute Sample Correlation Matrix

```r
# Compute correlation matrix
R <- cor(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
print(R)
```

```
##              Indep        Supp       Benev    Conform     Leader
## Indep    1.0000000 -0.17335767 -0.56116271 -0.4713753  0.1867690
## Supp    -0.1733577  1.00000000  0.01836202 -0.3270797 -0.4014696
## Benev   -0.5611627  0.01836202  1.00000000  0.2977052 -0.4915331
## Conform -0.4713753 -0.32707967  0.29770524  1.0000000 -0.3331093
## Leader   0.1867690 -0.40146956 -0.49153305 -0.3331093  1.0000000
```

Interpret the pairwise correlations

Independence vs. Benevolence(r=−0.5612): 負相關，Independence 越高，Benevolence 越低。越獨立的個體，通常較不會表現出關懷或善良的行為。

Independence vs. Conformity(r=-0.4714)：負相關，Independence 越高，Conformity 越低。代表越獨立的個體，越不願意遵守規則。

Benevolence vs. Leadership(r=-0.4915): 負相關，Benevolence 越高，Leadership 越低。越善良、關懷他人的人，通常不太具有強勢的領導風格。

Support vs. Benevolence (r=0.0184):Support 和 Benevolence 幾乎無關。Support 與 Benevolence 幾乎沒有關聯。是否願意支持他人，與他是否仁慈沒有明顯的線性關係。

3. Chapter 2, exercises 2.7: use the sample variance-covariance matrix S_n obtained in Question 2 for the analysis.

(a) Compute the Eigenvalues and Eigenvectors

```r
# Compute covariance matrix
Sn <- cov(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")])

# Compute eigenvalues and eigenvectors
eigen_decomp <- eigen(Sn)
eigenvalues <- eigen_decomp$values
eigenvectors <- eigen_decomp$vectors

# Display results
eigenvalues
```

```
## [1] 68.752385 31.508994 23.100973 16.354182  2.392411
```

```r
eigenvectors
```

```
##             [,1]        [,2]       [,3]        [,4]      [,5]
## [1,]  0.57943538  0.07917988  0.6428795 -0.30939267 0.3859629
## [2,] -0.04165689  0.61192825 -0.1399143  0.51462195 0.5825777
## [3,] -0.52428496  0.21883511 -0.1192554 -0.73403767 0.3524249
## [4,] -0.49309245 -0.57215650  0.4221873  0.30427403 0.3983365
## [5,]  0.38013742 -0.49398633 -0.6120997 -0.08970196 0.4782893
```

(b) Spectral Decomposition

```r
# Compute spectral decomposition
spectral_dec <- eigenvectors %*% diag(eigenvalues) %*% solve(eigenvectors)

# Display results
spectral_dec
```

```
##            [,1]       [,2]        [,3]       [,4]       [,5]
## [1,]  34.750209 -4.2766846 -18.0717949 -15.972868   5.716458
## [2,]  -4.276685 17.5134168   0.4197973  -7.868217  -8.723315
## [3,] -18.071795  0.4197973  29.8447227   9.348837 -13.942159
## [4,] -15.972868 -7.8682171   9.3488372  33.042636  -9.941860
## [5,]   5.716458 -8.7233154 -13.9421586  -9.941860  26.957961
```

(c) Compute the Inverse of A^{-1}

```r
# Compute inverse of covariance matrix
Sn_inv <- solve(Sn)

# Display results
Sn_inv
```

```
##              Indep       Supp      Benev    Conform     Leader
## Indep   0.09109292 0.08154330 0.06355526 0.06466208 0.06378665
## Supp    0.08154330 0.17081442 0.06801094 0.09320390 0.10752923
## Benev   0.06355526 0.06801094 0.09099546 0.04262876 0.07131297
## Conform 0.06466208 0.09320390 0.04262876 0.09362590 0.07302338
## Leader  0.06378665 0.10752923 0.07131297 0.07302338 0.12217626
```

(d) Compute Eigenvalues and Eigenvectors of A^{-1}

```r
# Compute eigenvalues and eigenvectors of inverse matrix
eigen_decomp_inv <- eigen(Sn_inv)
eigenvalues_inv <- eigen_decomp_inv$values
eigenvectors_inv <- eigen_decomp_inv$vectors

# Display results
eigenvalues_inv
```

```
## [1] 0.41798836 0.06114644 0.04328822 0.03173697 0.01454495
```

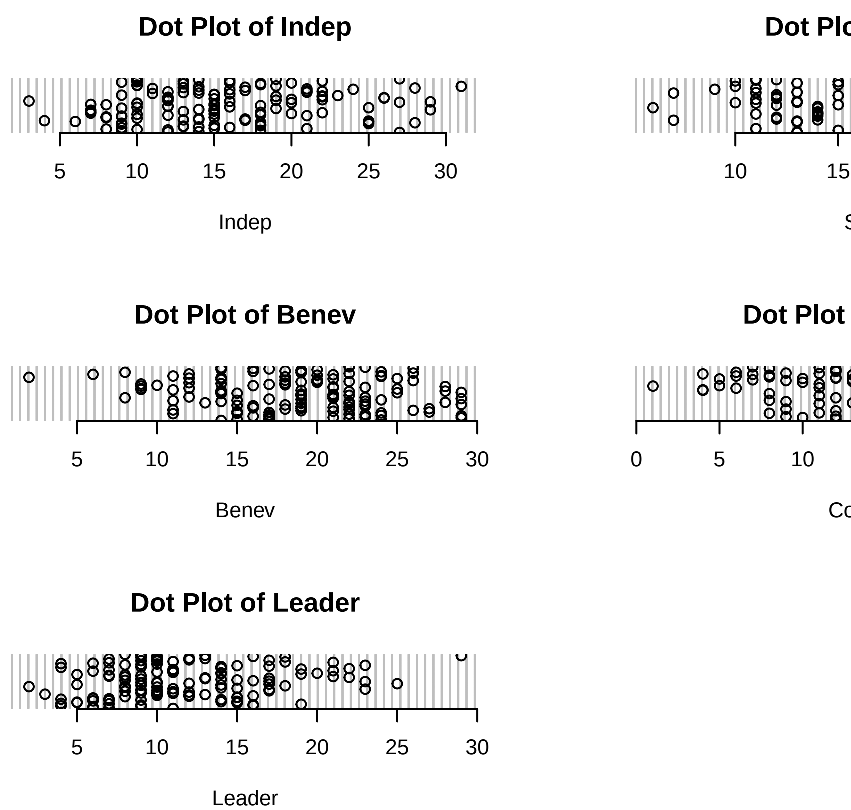
```r
eigenvectors_inv
```

```
##              [,1]        [,2]       [,3]        [,4]        [,5]
## [1,] -0.3859629 -0.30939267  0.6428795 -0.07917988  0.57943538
## [2,] -0.5825777  0.51462195 -0.1399143 -0.61192825 -0.04165689
## [3,] -0.3524249 -0.73403767 -0.1192554 -0.21883511 -0.52428496
## [4,] -0.3983365  0.30427403  0.4221873  0.57215650 -0.49309245
## [5,] -0.4782893 -0.08970196 -0.6120997  0.49398633  0.38013742
```

4. Chapter 4, exercises 4.39. Please first follow the steps in page 189 to identify outliers, and delete these identified outliers, if any, before doing questions in (a)-(c).

Identify outliers, and delete these identified outliers

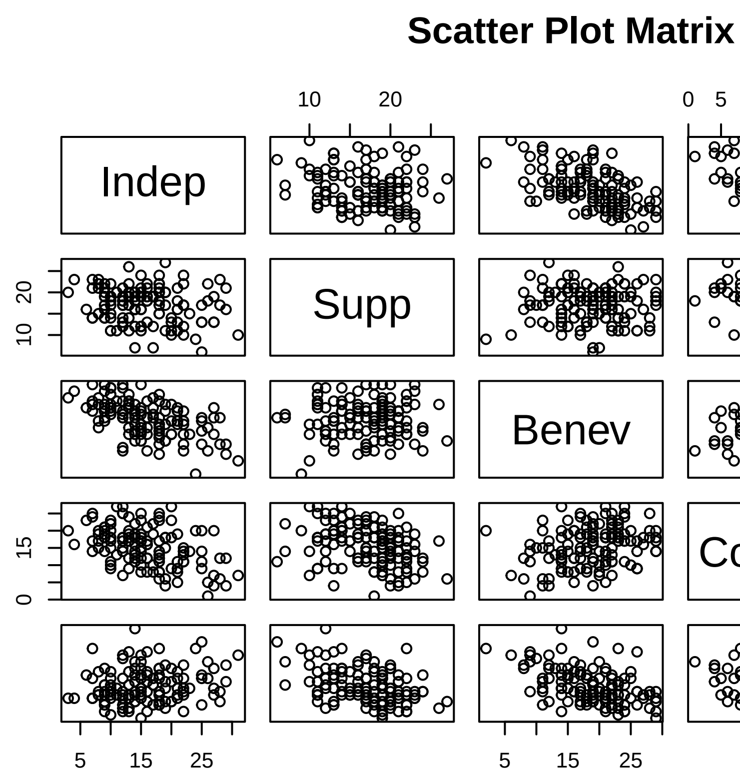
```r
# Create dot plots
par(mfrow=c(3,2))
for (var in c("Indep", "Supp", "Benev", "Conform", "Leader")) {
  dotchart(df[[var]], main = paste("Dot Plot of", var), xlab = var, frame.plot = FALSE)
}
```



1. Make a dot plot for each variable.

根據 Dot plot，在 Benev 中小於 3 和在 Leader 中大於 27 的點可能為離群值，要再做更進一步的檢查。

```r
# Create scatter plot matrix
pairs(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")], main="Scatter Plot Matrix")
```



2. Make a scatter plot for each pair of variables.

根據 scatter plot，在 Indep vs. Benev、Benev vs. Conform 有幾個點偏離群體，要做更進一步的檢查。

```r
# standardized values
standardized_df <- as.data.frame(scale(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")]))

# Pick the standardized values greater than 3 or less than -3
outliers <- which(standardized_df > 3 | standardized_df < -3, arr.ind = TRUE)
print(outliers)
```

3. Calculate the standardized values

```
##      row col
## [1,] 104   3
## [2,] 127   5
```

計算標準化值若大於 3 和小於-3 判定為離群值，發現沒有資料為離群值。

```r
# Compute square distance matrix

diff <- df[, c("Indep", "Supp", "Benev", "Conform", "Leader")] - rowMeans(df[, c("Indep", "Supp", "Bene
#diff
S <- cov(df[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
Sn_inv <- solve(S)

# Compute d square
d_squared <- apply(diff, 1, function(x) t(x) %*% Sn_inv %*% x)

# Find chi-square critical value at alpha = 0.05 with 5 degrees of freedom
chi_square_threshold <- qchisq(0.95, df = 5)

# Identify extreme outliers based on chi-square threshold
outliers_1 <- which(d_squared > chi_square_threshold)
print(outliers_1)
```

4. Calculate the generalized squared distances. Examine these distances for unusually large values. In a chi-square plot, these would be the points farthest from the origin.

```
## [1]  60  78  96 129
```

```r
# Delete the outliers from steps3 and step4
outlier_indices <- unique(c(outliers[, 1], outliers_1))
df_final <- df[-outlier_indices, ]

# Display the cleaned data
head(df_final)
```

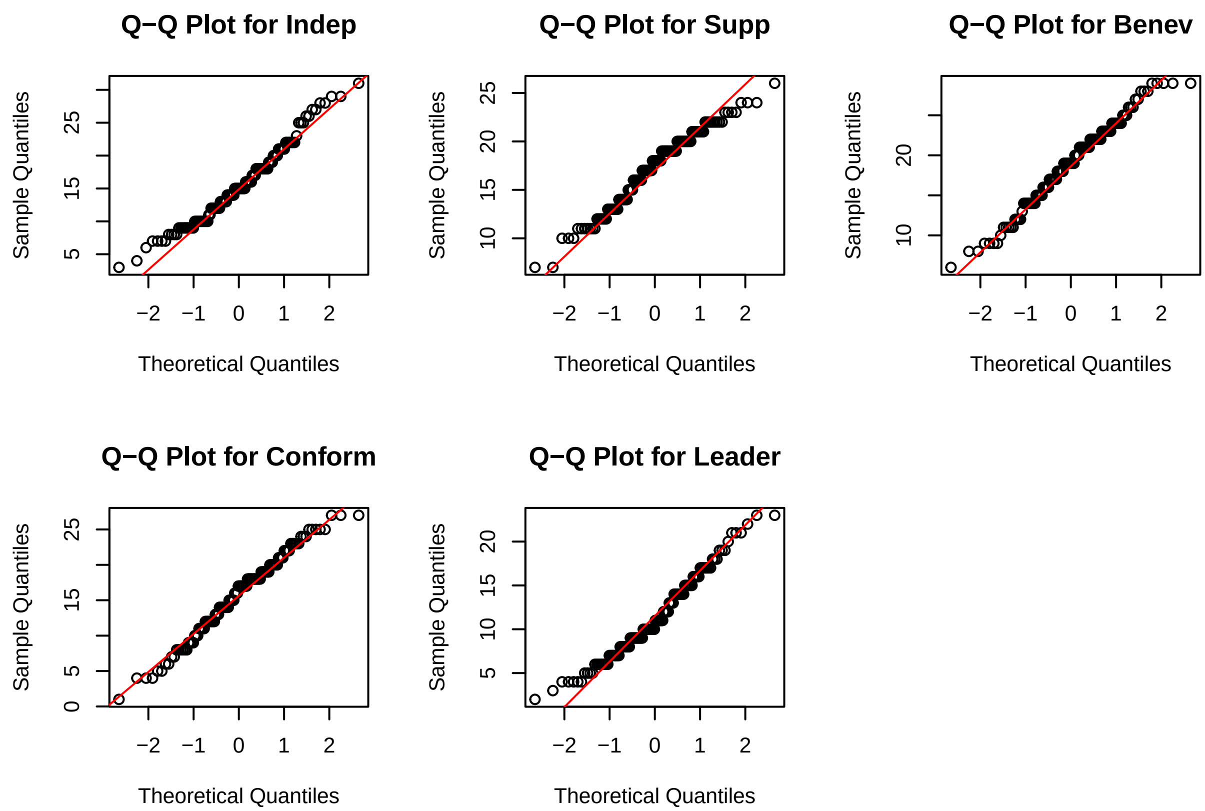
```
##   Indep Supp Benev Conform Leader Gender Socio
## 1    27   13    14      20     11      2     1
## 2    12   13    24      25      6      2     1
## 3    14   20    15      16      7      2     1
## 4    18   20    17      12      6      2     1
## 5     9   22    22      21      6      2     1
## 6    18   15    17      25      9      2     1
```

計算資料的 d^2，若大於 $\chi^2_{0.05,5}$ 則判定為離群值並刪除。最後，將步驟 3 和步驟 4 的離群值刪除。

(a) Examine each of the variables independence, support, benevolence, conformity and leadership for marginal normality.

```r
par(mfrow = c(2, 3))

vars <- c("Indep", "Supp", "Benev", "Conform", "Leader")

for (var in vars) {
 qqnorm(df_final[[var]], main = paste("Q-Q Plot for", var))
  qqline(df_final[[var]], col = "red")
```

```r
}

par(mfrow = c(1, 1))
```




```r
library(stats)

shapiro_results <- data.frame()

# Do Shapiro-Wilk Test
for (var in c("Indep", "Supp", "Benev", "Conform", "Leader")) {
  test <- shapiro.test(df_final[[var]])

  shapiro_results <- rbind(shapiro_results, data.frame(
    Variable = var,
    Statistic = test$statistic,
    P_value = test$p.value
  ))
}


print(shapiro_results)
```

```
##    Variable Statistic    P_value
## W     Indep 0.9739716 0.01688103
## W1     Supp 0.9738217 0.01634822
```

```
## W2    Benev 0.9831122 0.12465777
## W3  Conform 0.9845469 0.17035807
## W4   Leader 0.9736605 0.01579465
```

H_0: 樣本來自常態分佈

H_1: 樣本不來自常態分佈

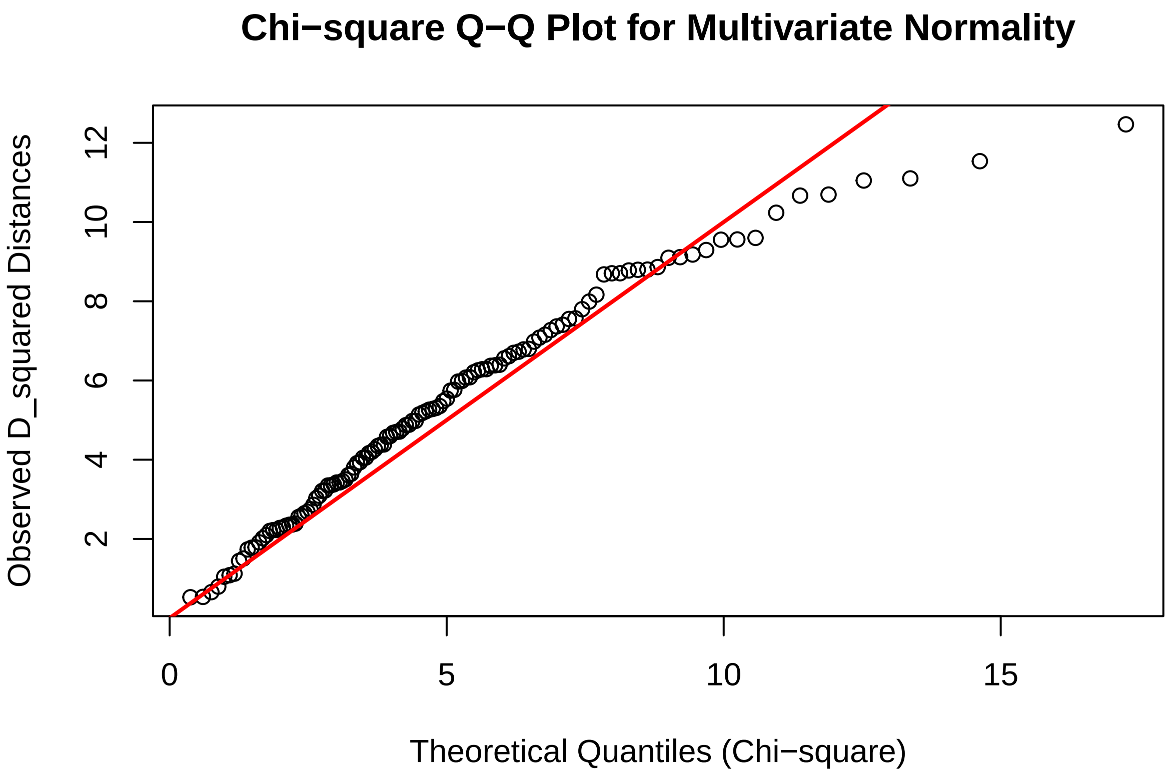
alpha = 0.05

假如 p-value<0.05，拒絕 H_0 Indep、Supp、Leader 拒絕 H_0，有強烈的證據說樣本不來自常態分佈 (非常態)

Benev、Conform 不拒絕 H_0，沒有有強烈的證據說樣本不來自常態分佈 (常態)

(b) Using all five variables, check for multivariate normality.

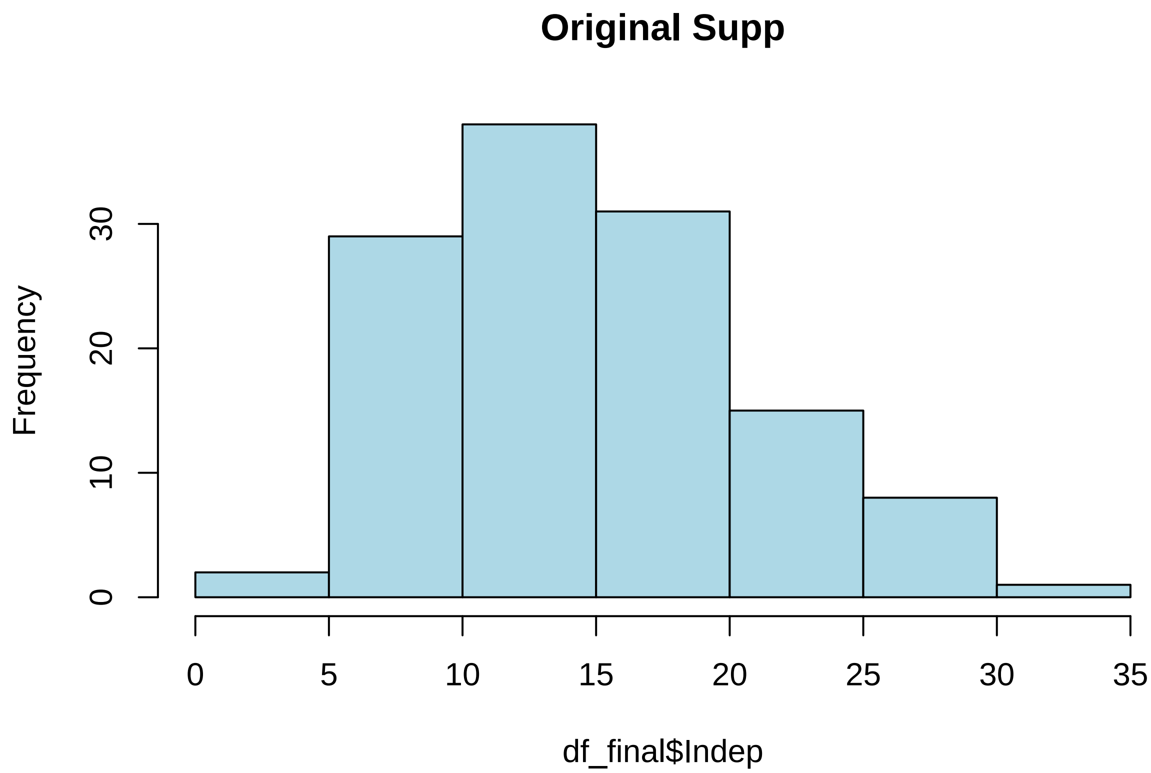
```r
x_bar <- rowMeans(df_final[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
S_inv <- solve(cov(df_final[, c("Indep", "Supp", "Benev", "Conform", "Leader")]))
diff <- as.matrix(df_final[, c("Indep", "Supp", "Benev", "Conform", "Leader")]) - x_bar
d_squared <- rowSums((diff %*% S_inv) * diff)

sort_d = sort(d_squared)
n <- nrow(df_final)
p <- ncol(df_final[, c("Indep", "Supp", "Benev", "Conform", "Leader")])
chi_theoretical <- qchisq((1:n - 0.5) / n, df = p)

# Draw Chi-square Q-Q Plot
qqplot(chi_theoretical, sort_d,
       main = "Chi-square Q-Q Plot for Multivariate Normality",
       xlab = "Theoretical Quantiles (Chi-square)",
       ylab = "Observed D_squared Distances")
abline(0, 1, col = "red", lwd = 2)
```



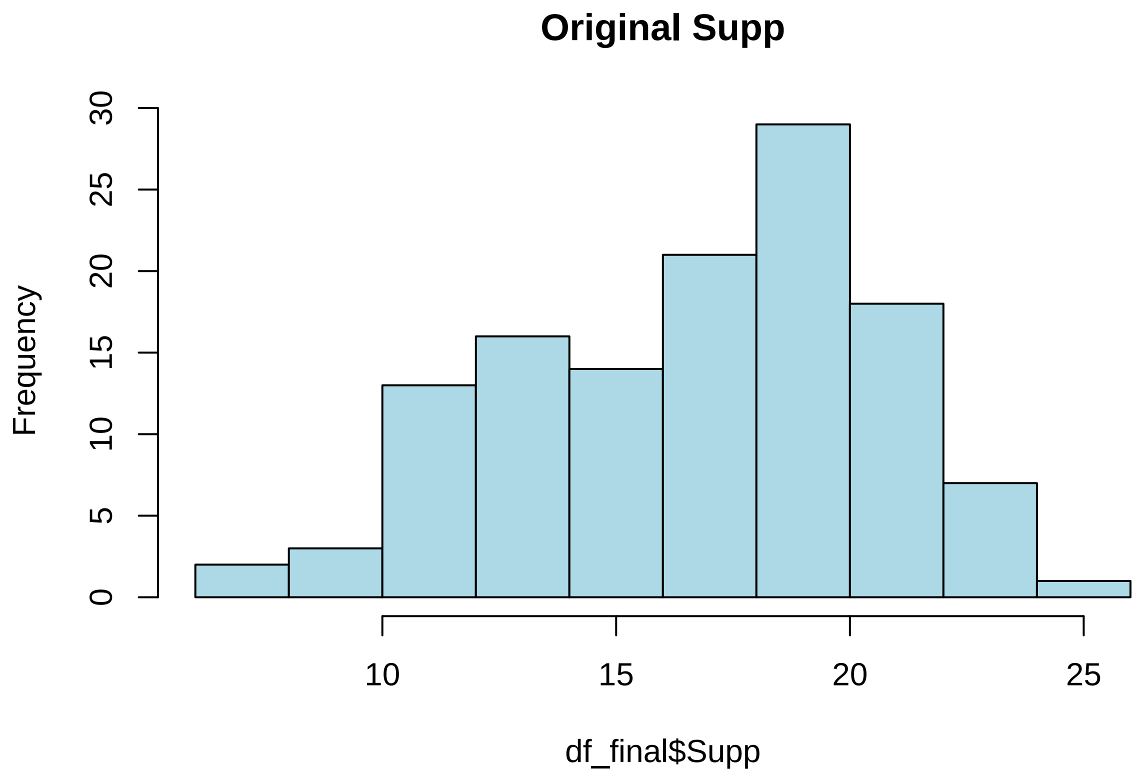
根據 Chi-square Q-Q Plot 可以發現值偏離斜率為 1 的線，推測這個數據沒有服從多重常態。

(c) Refer to part (a). For those variables that are nonnormal, determine the transformation that makes them more nearly normal.

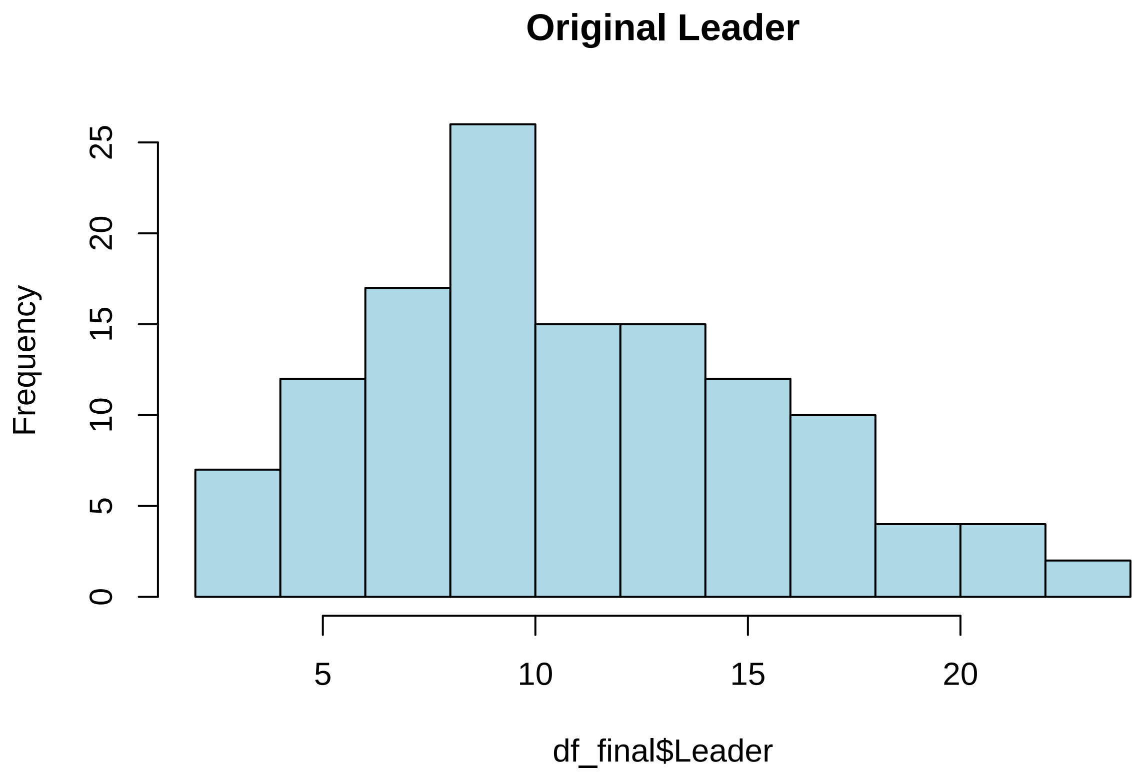
```
# Check the Indep,Supp,Leader  skewnwss
hist(df_final$Indep, main = "Original Supp", col = "lightblue")   # right skewness
```



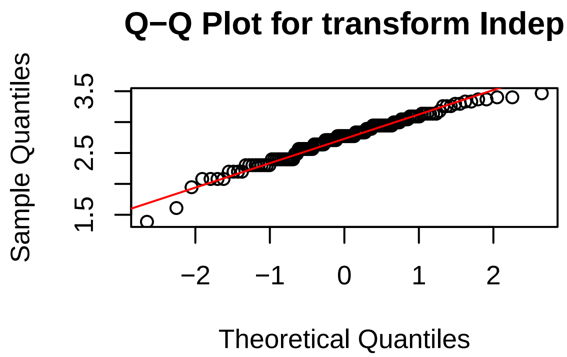

```r
hist(df_final$Supp, main = "Original Supp", col = "lightblue")   # left skewness
```

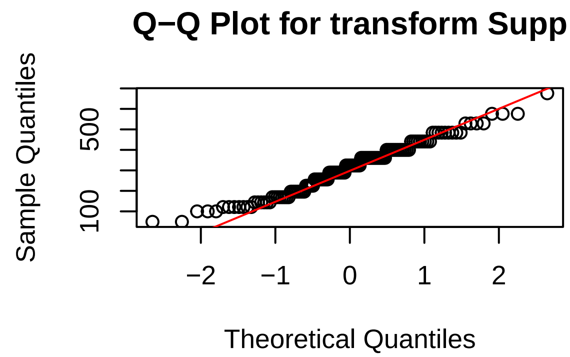


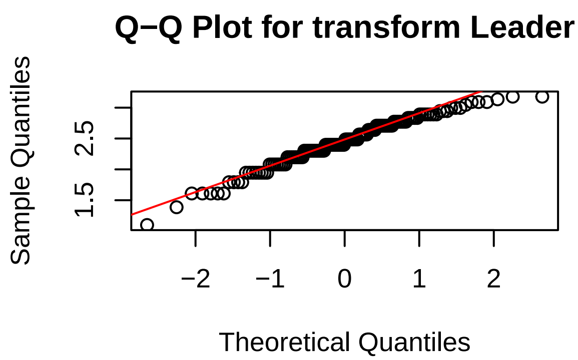

```r
hist(df_final$Leader, main = "Original Leader", col = "lightblue") # right skewness
```




```
# Transform data
# log transform of data
df_final$Indep_log <- log(df_final$Indep + 1)
# square of data
df_final$Supp_squ <- (df_final$Supp) ** 2
# log transform of data
df_final$Leader_log <- log(df_final$Leader + 1)

# Check the normality after transforming
par(mfrow = c(2, 2))
qqnorm(df_final$Indep_log, main = paste("Q-Q Plot for transform Indep"))
qqline(df_final$Indep_log, col = "red")
qqnorm(df_final$Supp_squ, main = paste("Q-Q Plot for transform Supp"))
qqline(df_final$Supp_squ, col = "red")
qqnorm(df_final$Leader_log, main = paste("Q-Q Plot for transform Leader"))
qqline(df_final$Leader_log, col = "red")
```







由 (a) 得知，Indep、Supp、Leader 不來自常態分佈，根據直方圖，可以發現 Supp 左偏尾，Indep、Leader 右偏尾，所以對 Supp 做平方轉換，對 Indep、Leader 做 log 轉換，使資料較為常態。