multivariate_lab2_hw

Na SeungChan

2024-11-01

Exercises

1

Let mu = (0, 1, 2), Sigma_{i,j} = sigma_i * sigma_j * rho^{|i-j|}, rho = 0.9, (sigma_1, sigma_2, sigma_3) = (3,2,1). Randomly sample n = 100 observations following N(mu,Sigma), and compute the sample mean Xbar, covariance matrix S and correlation matrix R.

```
set.seed(42)
mu_q1 <- c(0, 1, 2)
sigma_q1 <- matrix(c(9, 5.4, 2.7,
                     5.4, 4, 1.8,
                     2.7, 1.8, 1), nrow = 3)
cor_q1 <- cov2cor(sigma_q1)
n_q1 <- 100
X_q1 <- rmvnorm(n_q1, mean = mu_q1, sigma = sigma_q1)
```

```
colMeans(X_q1) #sample mean Xbar
```

```
## [1] -0.1396325  0.8669039  1.9687345
```

```
var(X_q1) #covariance matrix S
```

```
##          [,1]     [,2]     [,3]
## [1,] 7.555700 4.315951 2.289927
## [2,] 4.315951 3.263390 1.493857
## [3,] 2.289927 1.493857 0.889572
```

```
cor(X_q1) #correlation matrix R
```

```
##           [,1]      [,2]      [,3]
## [1,] 1.0000000 0.8691699 0.8832706
## [2,] 0.8691699 1.0000000 0.8767664
## [3,] 0.8832706 0.8767664 1.0000000
```

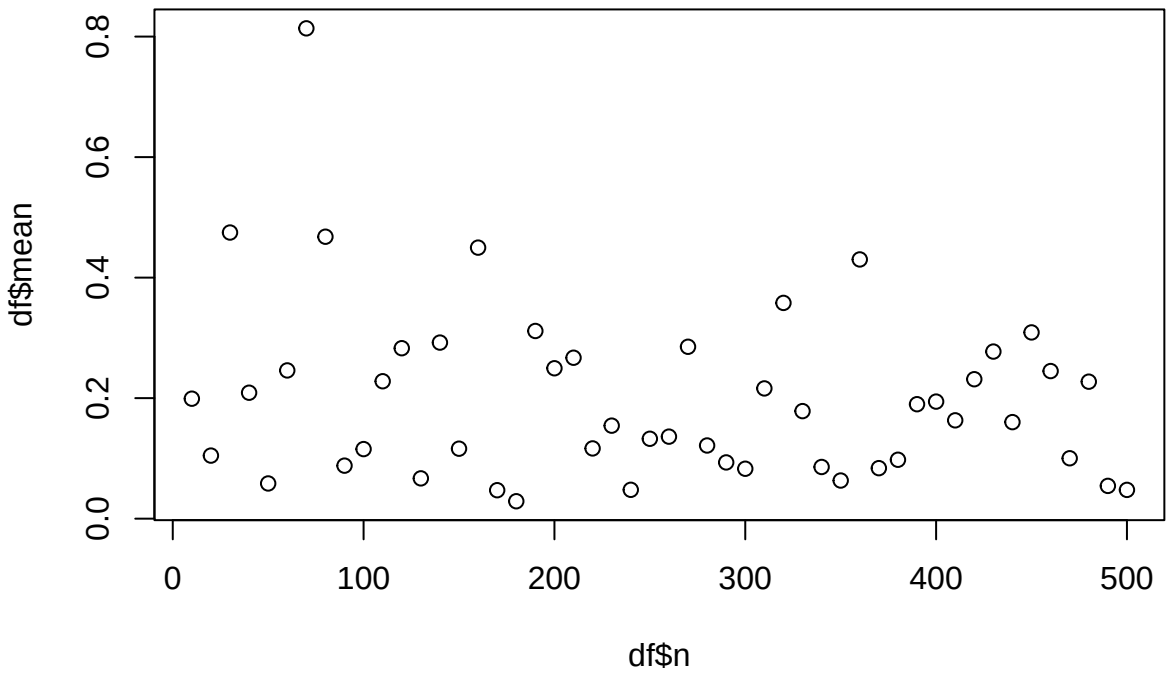
2

Compute the difference between Xbar and mu, S and Sigma, R and the population correlation matrix, using the Frobenius norm. Repeat for n = 10, 20, ⋯ 500, and visually display the result. In what rate does the difference reduce?

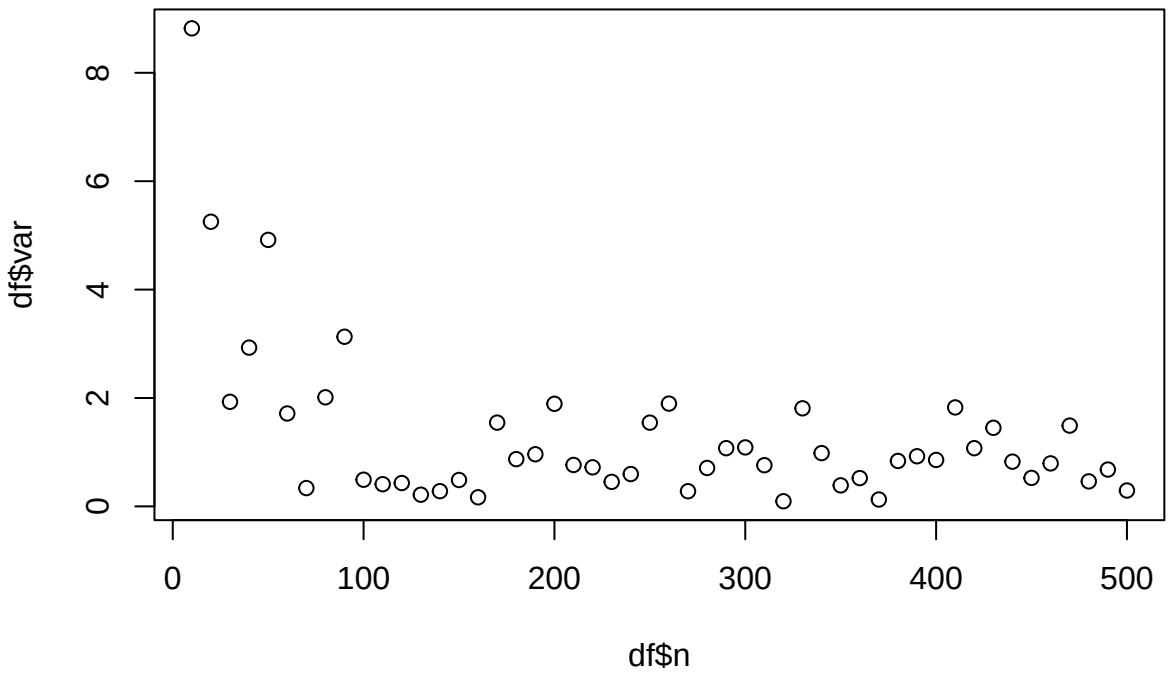
```
set.seed(42)
cal <- function(n){
  mu <- c(0, 1, 2)
  sig <- matrix(c(9, 5.4, 2.7,
                  5.4, 4, 1.8,
                  2.7, 1.8, 1), nrow = 3)
  corr <- cov2cor(sig)
  rv.n <- rmvnorm(n, mean = mu, sigma = sig) #sampling n times

  xb <- colMeans(rv.n)
  s <- var(rv.n)
  r <- cor(rv.n) #calculate x_bar, S, R

  a <- norm(matrix(mu - xb), 'f')
  b <- norm(matrix(sig - s), 'f')
  c <- norm(matrix(corr - r), 'f') #calculate F-norm

  return (c(a, b, c))
}

df <- data.frame()
for (i in 1:50) {
  temp <- cal(i*10)
  temp[4] <- 10*i
  df <- rbind(df, temp)
}
colnames(df) <- c('mean', 'var', 'corr', 'n')
head(df)
```

```
##         mean      var       corr  n
## 1 0.19909850 8.820304 0.08328150 10
## 2 0.10466751 5.252304 0.06429084 20
## 3 0.47484737 1.929682 0.04324363 30
## 4 0.20899007 2.929503 0.09917909 40
## 5 0.05839519 4.917120 0.08241003 50
## 6 0.24600226 1.714125 0.02562513 60
```

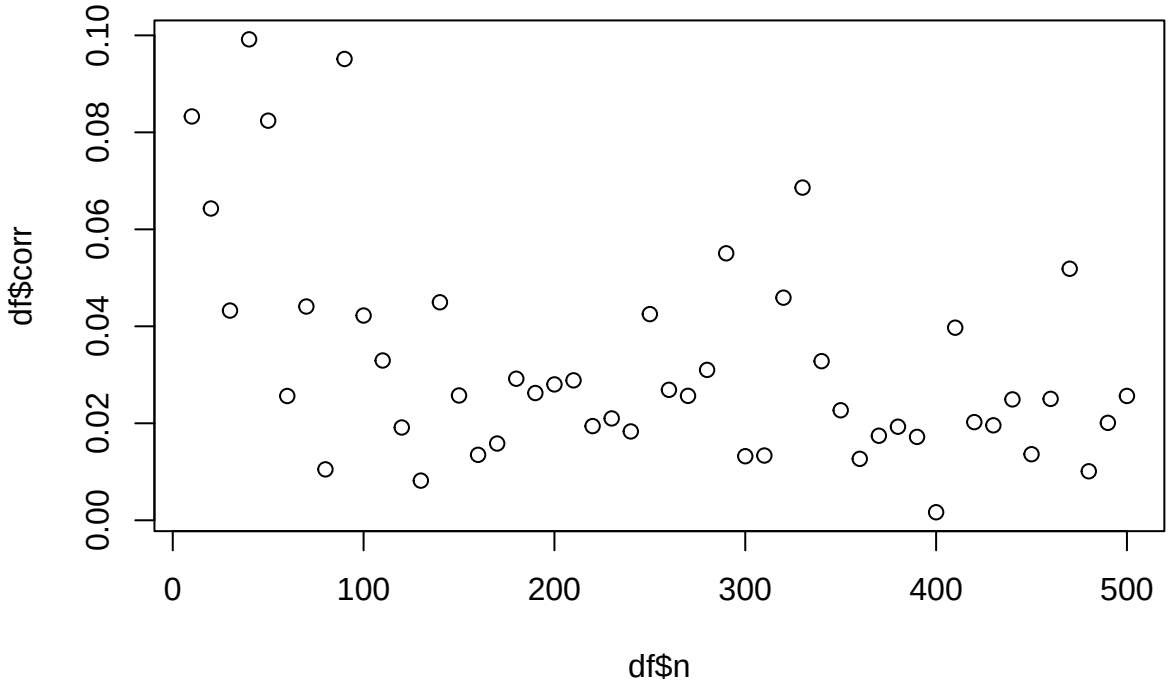
```
plot(df$n, df$mean)
```




```
plot(df$n, df$var)
```




```
plot(df$n, df$corr)
```



세 수치 모두, 프로베니우스 노름이 전반적으로 n이 늘어날수록 감소하는 경향을 보인다는 정도는 관측할 수 있다. 그러나, n이 커짐에 따라 노름이 수렴하는 경향을 관찰할 정도로 n이 충분히 큰 것은 아니어서인지, seed를 변화시켜 보면 그래프상 수렴 속도는 계속 변화한다.

3

Suggest how would you generate a non-normal 3-variate distribution. The three variables should be (linearly) correlated with each other. Randomly sample n = 100 observations from the non-normal distribution. Empirically confirm that the three variables are indeed correlated with each other by

다음과 같이 확률변수를 정의한다. unif(0,1)을 따르는 독립적인 확률변수 3개 (X1, X2, X3)를 생성하고, 이로부터 확률변수의 차를 통해 새로운 확률변수 (Y1 = X1, Y2 = X2 = X1, Y3 = X3 - X2)를 생성한다. 변수 변환을 통해 이와 같이 생성된 Y1, Y2, Y3의 확률밀도함수를 구하면 분포의 토대가 분리되지 않으므로 당연히 세 확률변수는 mutual independent가 아니다.

```
set.seed(42)
X1_q3 <- runif(100, 0, 1)
X2_q3 <- runif(100, 0, 1)
X3_q3 <- runif(100, 0, 1)

Y1_q3 <- X1_q3
Y2_q3 <- X2_q3 - X1_q3
Y3_q3 <- X3_q3 - X2_q3

data_q3 <- tibble(Y1_q3, Y2_q3, Y3_q3)
```

1)

computing the sample correlation coefficients and

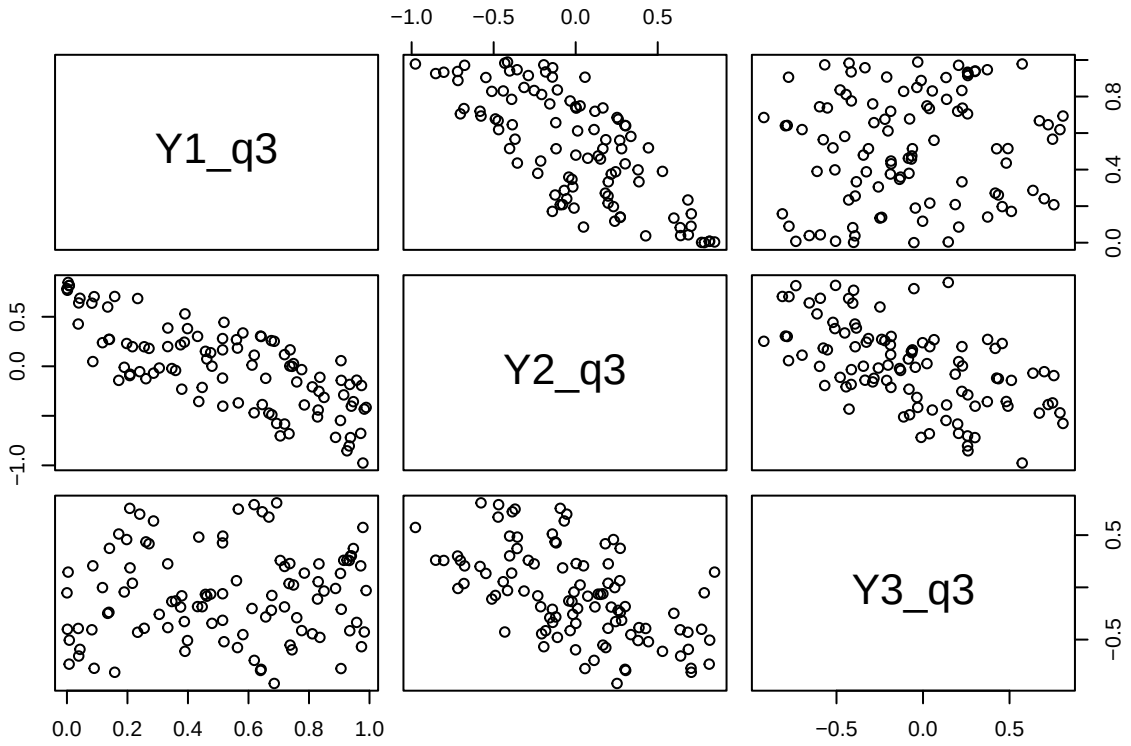
```
cor(data_q3)
```

```
##             Y1_q3      Y2_q3       Y3_q3
## Y1_q3  1.00000000 -0.7402261  0.08678681
## Y2_q3 -0.74022614  1.0000000 -0.56507587
## Y3_q3  0.08678681 -0.5650759  1.00000000
```

2)

displaying the scatter. (The scatterplot should also exhibit the non-normality to some degree.)

```
plot(data_q3)
```



산점도 모양을 살펴보면, 정규분포의 모양에서는 보이지 말아야 할 두꺼운 꼬리가 보인다. 즉, 분포의 두께가 0 부근과 0 부근에서 서로 유사하므로 해당 분포는 정규성을 따른다고 볼 수 없을 것이다.

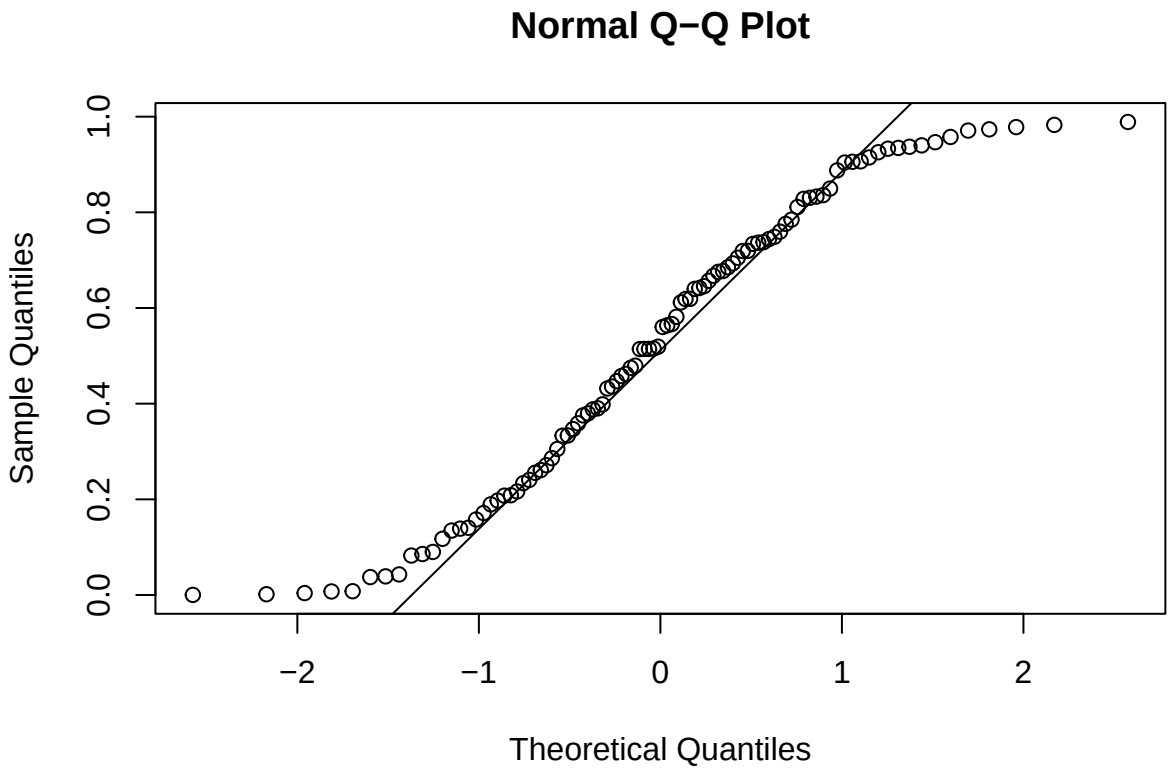
4

To check normality (rather, non-normality) of the data you have generated,

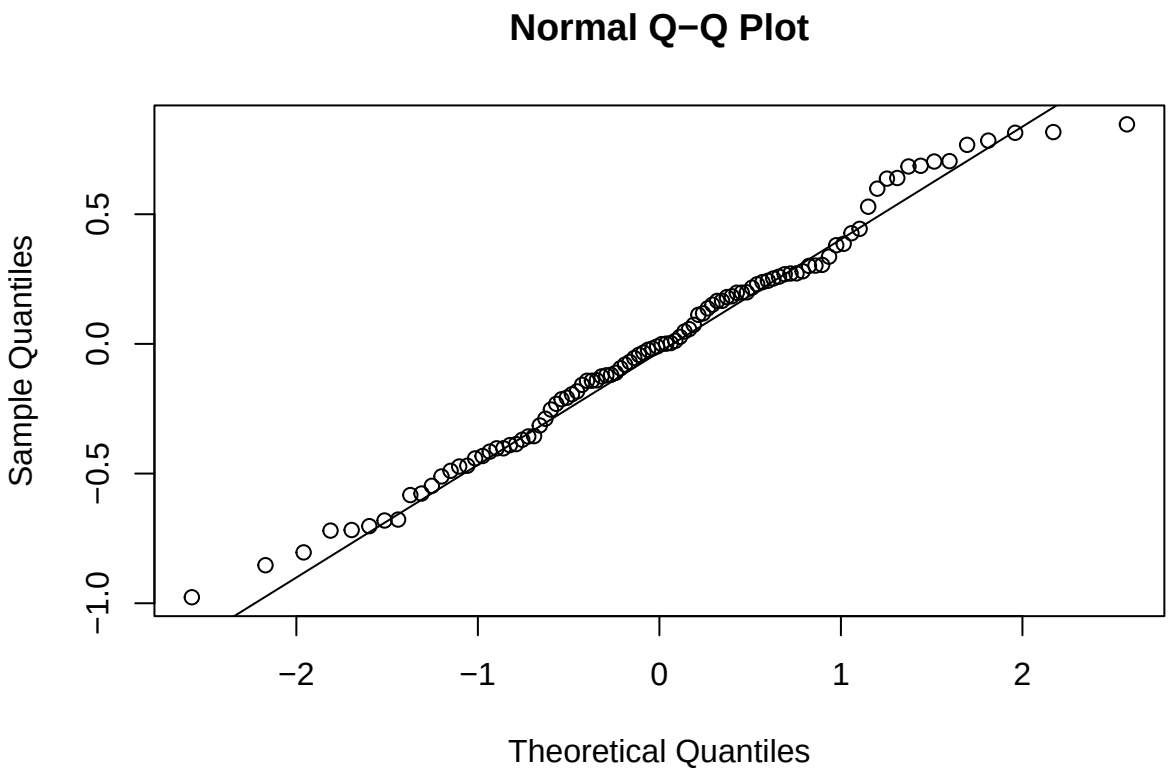
a)

use the normal-probability plot for each variable to check normality

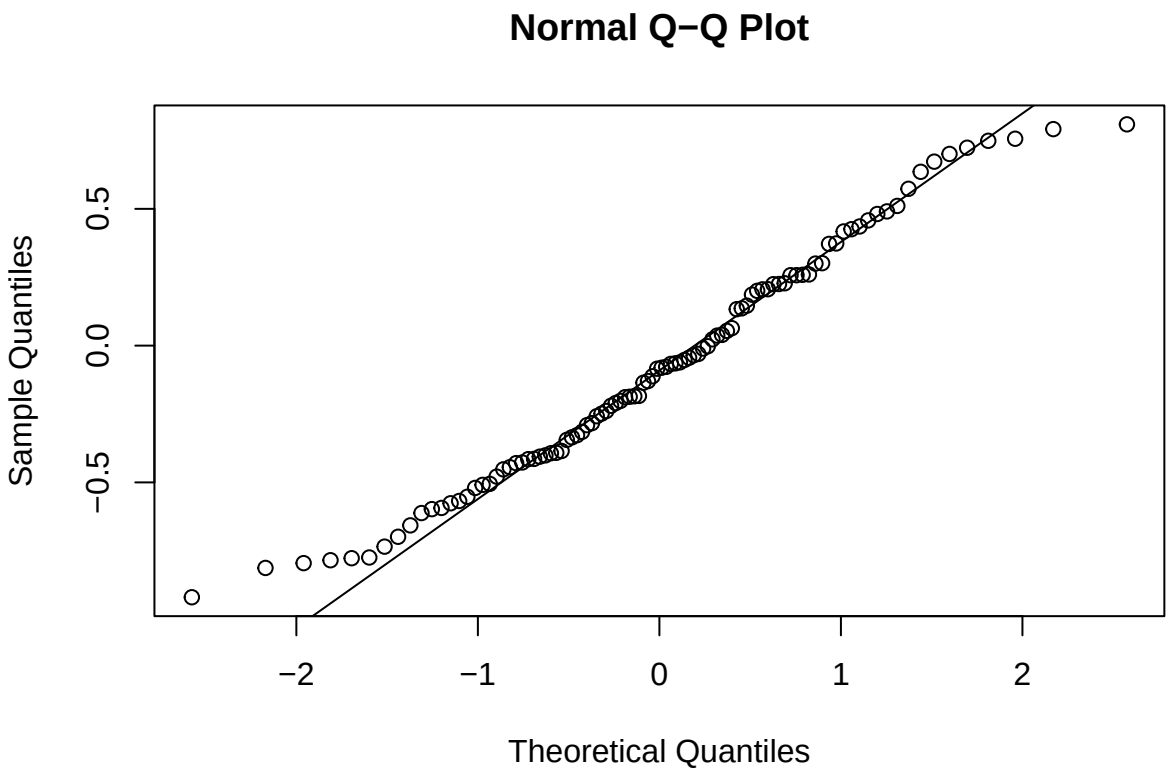
```
qqnorm(Y1_q3); qqline(Y1_q3)
```




```
qqnorm(Y2_q3); qqline(Y2_q3)
```




```
qqnorm(Y3_q3); qqline(Y3_q3)
```



Y1은 직선에서 벗어나는 포인트들이 확연히 많은 경향을 보인다. Y2, Y3은 Q-Q plot만으로 정규성을 판단하기 어려워 보인다. 꼬리 부분이 좀 벗어나지만 데이터만으로 판단하면 정규분포 가정을 하고 분석해도 무방해 보인다.

b)

Use the Shapiro-Wilk test (cf `?shapiro.test`). Be sure to specify the null hypothesis of the test

```
shapiro.test(Y1_q3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Y1_q3
## W = 0.94644, p-value = 0.0004872
```

```
shapiro.test(Y2_q3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Y2_q3
## W = 0.98551, p-value = 0.3455
```

```r
shapiro.test(Y3_q3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Y3_q3
## W = 0.97878, p-value = 0.1067
```

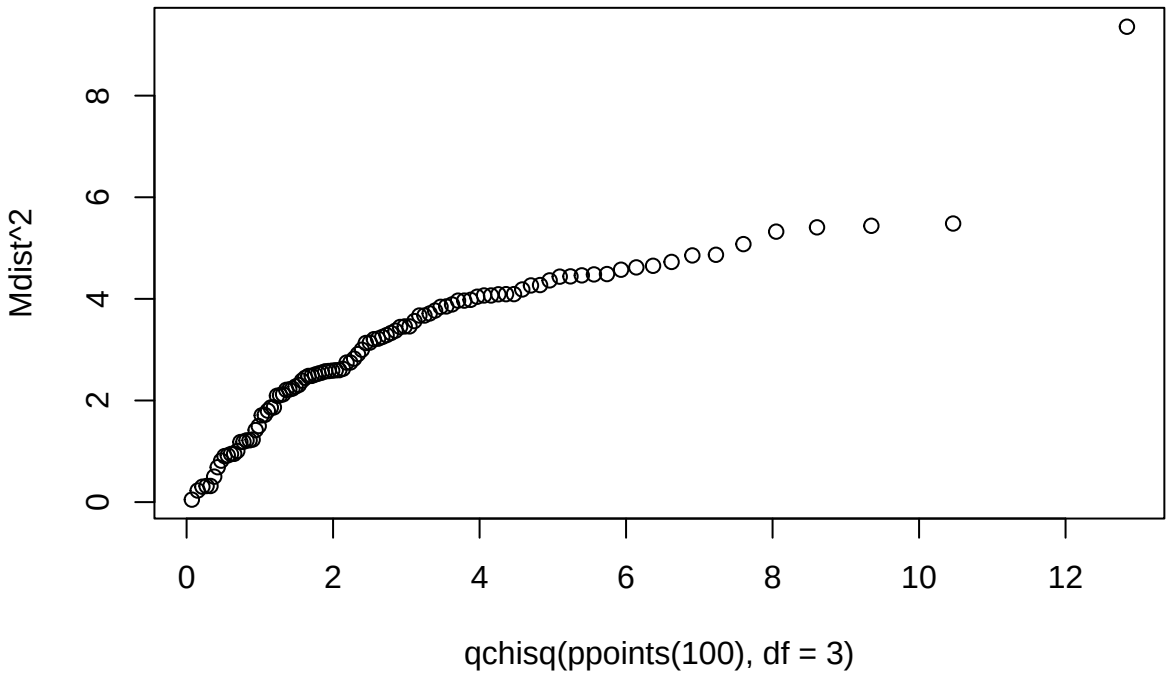
샤피로-윌크 테스트의 귀무가설은 데이터가 정규분포를 따른다는 것이고, 대립가설은 데이터가 정규분포를 따르지 않는다는 것이다.

각 데이터 벡터의 p-value를 확인하면, Y1은 정규분포를 따르지 않는다고 결론을 내릴 수 있다. 그러나 Y2, Y3은 정규분포를 따르지 않는다는 결론을 내리기 어렵다. (물론, n을 늘려서 샘플링한 샘플에 대해 다시 검정을 시행해 보면 p-value가 감소하여 정규분포를 따르지 않는다는 결론이 내려진다.)

c)

Use the chi-squared plot to check normality.

```r
Xc <- t(t(data_q3) - colMeans(data_q3))
S <- cov(data_q3)
Mdist <- sqrt( diag( Xc %*% solve(S) %*% t(Xc) ) )
qqplot( qchisq(ppoints(100), df = 3), Mdist^2)
```



마할라노비스 거리가 이론적 분포와 일직선상에 있지 않다. 정규분포를 따른다고 보기 어렵다.

5

Suppose (X,Y,Z) follow the 3-variate normal distribution defined in #1.

```
colnames(X_q1) <- c('X', 'Y', 'Z')
W <- as_tibble(X_q1)
X_q1 <- as_tibble(X_q1)

mu <- colMeans(W)
Sigma <- as.matrix(var(W))
Sigma
```

```
##          X        Y        Z
## X 7.555700 4.315951 2.289927
## Y 4.315951 3.263390 1.493857
## Z 2.289927 1.493857 0.889572
```

a) What is the conditional distribution of (X,Y) given Z = z?

```
i1 <- c(T, T, F) #predictor
i2 <- !i1 #response

mu1 <- mu[i1]
mu2 <- mu[i2]
Sigma11 <- Sigma[i1,i1]
Sigma12 <- Sigma[i1,i2]
Sigma21 <- Sigma[i2,i1]
Sigma22 <- Sigma[i2,i2]

mu1
```

```
##          X          Y
## -0.1396325  0.8669039
```

```
mu2
```

```
##        Z
## 1.968735
```

```
Sigma12 %*% solve(Sigma22)
```

```
##          [,1]
## [1,] 2.574190
## [2,] 1.679298
```

```
Sigma11 - Sigma12 %*% solve(Sigma22) %*% Sigma21
```

```
##           X         Y
## X 1.6609925 0.4704803
## Y 0.4704803 0.7547589
```

로부터 조건부분포 공식에 의해 조건부분포를 구하면

$$(X,Y)|Z=z \sim N_2\left(\begin{pmatrix}-0.1396325+2.574190(z-1.968735)\\ 0.8669039+1.679298(z-1.968735)\end{pmatrix}, \begin{pmatrix}1.6609925 & 0.4704803\\ 0.4704803 & 0.7547589\end{pmatrix}\right)$$

이다.

b) What is the best linear prediction of (X,Y) as a function of Z, i.e. BLP(X,Y | Z)?

(X, Y) : predictor -> X1 Z : response -> X2

BLP(X2 | X1) = AX_1 + b, where A = Sigma_{21}Sigma_{11}^{-1} b = mu2 - Sigma_{21} Sigma_{11}^{-1} mu1

```
A <- Sigma21 %*% solve(Sigma11)
b <- mu2 - A %*% mu1
c(b,A)
```

```
## [1] 1.7906398 0.1700752 0.2328317
```

c) What is multiple correlation coefficient between Z and (X,Y)?

m.corr(X2, X1) = corr(X2, BLP(X2|X1)) = sqrt(Sigma21 * Sigma11^{-1} * Sigma12 / Sigma22)

```
m.corr_squared <- Sigma21 %*% solve(Sigma11) %*% Sigma12 / Sigma22
m.corr <- sqrt(m.corr_squared)
m.corr_squared
```

```
##           [,1]
## [1,] 0.8287996
```

```
m.corr
```

```
##           [,1]
## [1,] 0.9103843
```

d) Randomly sample n = 1000 observations as in #1. Using the data, perform a linear regression analysis of regressing Z onto (X,Y). Compare the R^2 from the regression with your answer in the subproblem c).

```
mu_q5 <- c(0, 1, 2)
sigma_q5 <- matrix(c(9, 5.4, 2.7,
                     5.4, 4, 1.8,
                     2.7, 1.8, 1), nrow = 3)
n_q5 <- 1000
set.seed(42)
```

```r
X_q5 <- rmvnorm(n_q5, mean = mu_q5, sigma = sigma_q5)
colnames(X_q5) <- c('X', 'Y', 'Z')
W <- X_q5
X_q5 <- as_tibble(X_q5)

mu <- colMeans(W)
Sigma <- as.matrix(var(W))
```

```r
i1 <- c(T,T,F)
i2 <- !i1

lm_q5 <- lm(Z ~ X + Y, data = X_q5)
summary(lm_q5)
```

```
##
## Call:
## lm(formula = Z ~ X + Y, data = X_q5)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.22546 -0.26171 -0.01726  0.26580  1.31462
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.776455   0.018371   96.70   <2e-16 ***
## X           0.164224   0.009392   17.49   <2e-16 ***
## Y           0.229540   0.014056   16.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3916 on 997 degrees of freedom
## Multiple R-squared:  0.8456, Adjusted R-squared:  0.8453
## F-statistic:  2730 on 2 and 997 DF,  p-value: < 2.2e-16
```

```r
summary(lm_q5)$r.squared
```

```
## [1] 0.8455731
```

두 값은 이론적으로 비슷해야 하고, 실제로 같은 샘플에서 두 값을 계산했다면 같았을 것이다. 지금은 다른 샘플에서 계산했으니 당연히 다르다.

```r
lm_q1 <- lm(Z ~ X + Y, data = X_q1)
summary(lm_q1)
```

```
##
## Call:
## lm(formula = Z ~ X + Y, data = X_q1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.02965 -0.18990 -0.03008  0.21928  0.81407
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.79064    0.05763  31.070  < 2e-16 ***
## X            0.17008    0.02915   5.834 7.12e-08 ***
## Y            0.23283    0.04436   5.249 9.00e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3943 on 97 degrees of freedom
## Multiple R-squared:  0.8288, Adjusted R-squared:  0.8253
## F-statistic: 234.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

```r
summary(lm_q1)$r.squared
```

```
## [1] 0.8287996
```

원래의 데이터에서 값을 확인하면 당연히 같은 것을 알 수 있다.