Homework 1

2019150432 임효진

March 20, 2021

Q1

(a)

```r
x1=c(4, 12, 8, 10, 6, 8)
x2=c(2, 3, 0, 2, 1, 4)
x3=c(5, 9, 6, 8, 6, 2)

tab1=cbind(x1, x2, x3)
x1=c(4, 12, 8, 10, 6, 8)
x2=c(2, 3, 0, 2, 1, 4)
x3=c(5, 9, 6, 8, 6, 2)

# sample mean vector
tab1=cbind(x1, x2, x3)
meanVec1=mean(x1)
meanVec2=mean(x2)
meanVec3=mean(x3)
c(meanVec1, meanVec2, meanVec3)
```

```
## [1] 8 2 6
```

```r
# sample covariance matrix S
cov(tab1)
```

```
##     x1   x2 x3
## x1 8.0  1.2  4
## x2 1.2  2.0 -1
## x3 4.0 -1.0  6
```

(b)

```r
cor(tab1)
```

```
##           x1        x2         x3
## x1 1.0000000 0.3000000  0.5773503
## x2 0.3000000 1.0000000 -0.2886751
```

```
## x3 0.5773503 -0.2886751  1.0000000
```

(c)

```r
z1=-x1+3*x2-2*x3
mean(z1)
```

```
## [1] -14
```

```r
var(z1)
```

```
## [1] 70.8
```

(d)

```r
z2=5*x2-x3
mean(z2)
```

```
## [1] 4
```

```r
var(z2)
```

```
## [1] 66
```

(e)

```r
z3=-x1+x3
mean(z3)
```

```
## [1] -2
```

```r
var(z3)
```

```
## [1] 6
```

(f)

```r
z=data.frame(z1,z2,z3)
z=as.matrix(z)
zt=t(z)
cov(zt)
```

```
##          [,1]  [,2]      [,3]      [,4]      [,5] [,6]
## [1,] 44.33333  91.5  54.66667  77.33333  52.16667   46
## [2,] 91.50000 189.0 111.00000 159.00000 106.50000   99
## [3,] 54.66667 111.0  89.33333 102.66667  78.33333    8
## [4,] 77.33333 159.0 102.66667 137.33333  95.66667   64
## [5,] 52.16667 106.5  78.33333  95.66667  70.33333   23
## [6,] 46.00000  99.0   8.00000  64.00000  23.00000  156
```

Q2

(a)

```
dat1=read.table("./usair.dat", header = T)

colMeans(dat1)
```

```
##        SO2       TEMP      MANUF        POP       WIND     PRECIP       DAYS
##  30.048780  55.763415 463.097561 608.609756   9.443902  36.769024 113.902439
```

```
cov(dat1)
```

```
##                SO2       TEMP       MANUF         POP        WIND      PRECIP
## SO2     550.947561 -73.560671   8527.7201   6711.9945   3.1753049  15.0017988
## TEMP    -73.560671  52.239878   -773.9713   -262.3496  -3.6113537  32.8629884
## MANUF  8527.720122 -773.971341 317502.8902 311718.8140 191.5481098 -215.0199024
## POP    6711.994512 -262.349634 311718.8140 335371.8939 175.9300610 -178.0528902
## WIND      3.175305  -3.611354    191.5481    175.9301   2.0410244  -0.2185311
## PRECIP   15.001799  32.862988   -215.0199   -178.0529  -0.2185311 138.5693840
## DAYS    229.929878 -82.426159   1968.9598    645.9860   6.2143902 154.7929024
##               DAYS
## SO2     229.92988
## TEMP    -82.42616
## MANUF  1968.95976
## POP     645.98598
## WIND      6.21439
## PRECIP  154.79290
## DAYS    702.59024
```

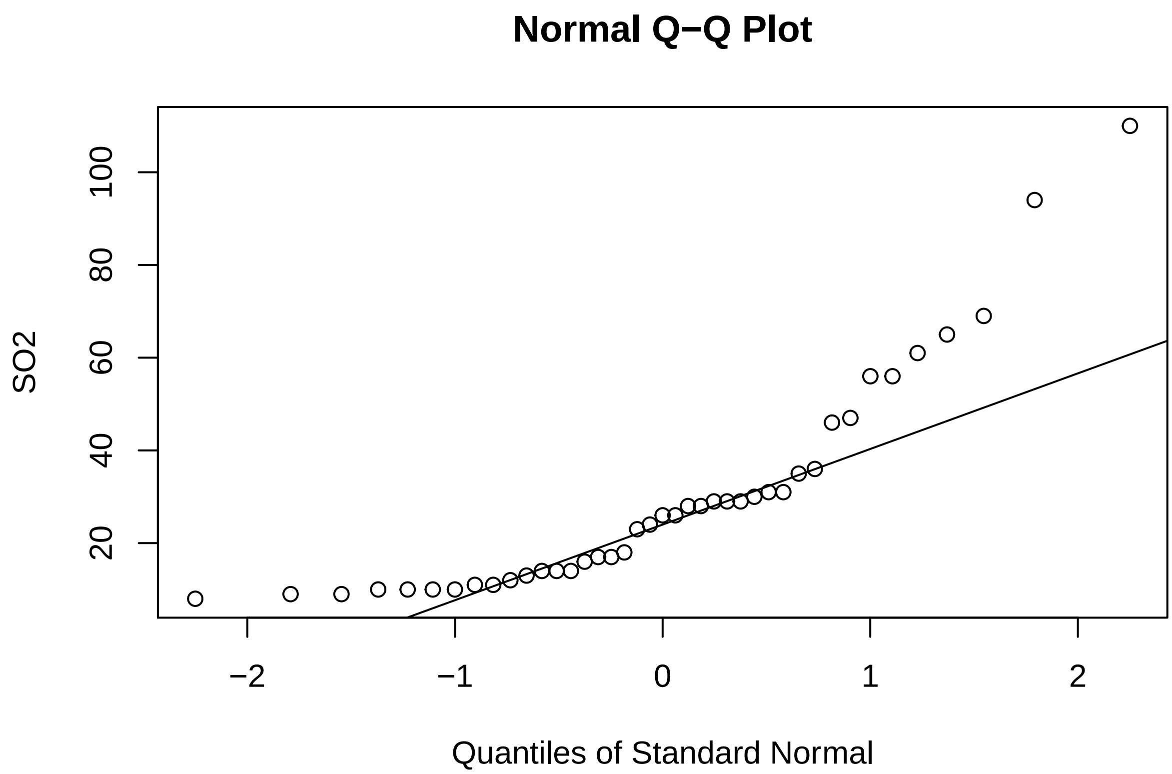
##(b)

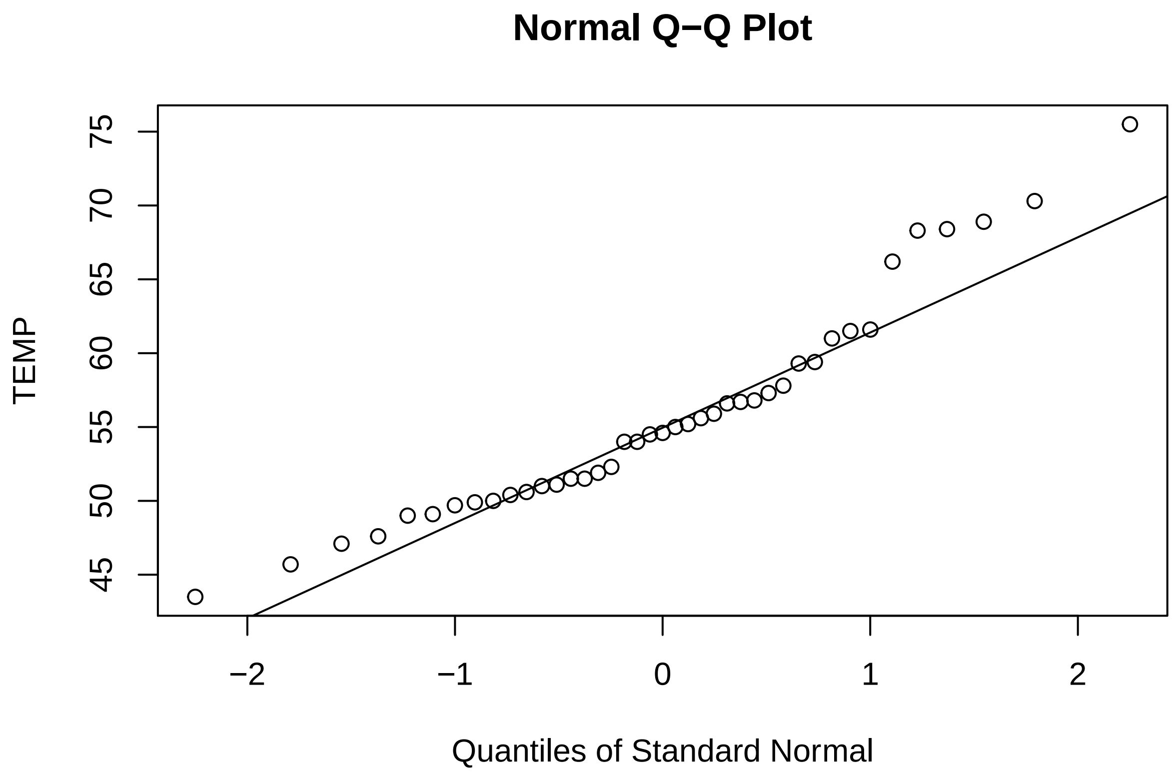
```
cor(dat1)
```

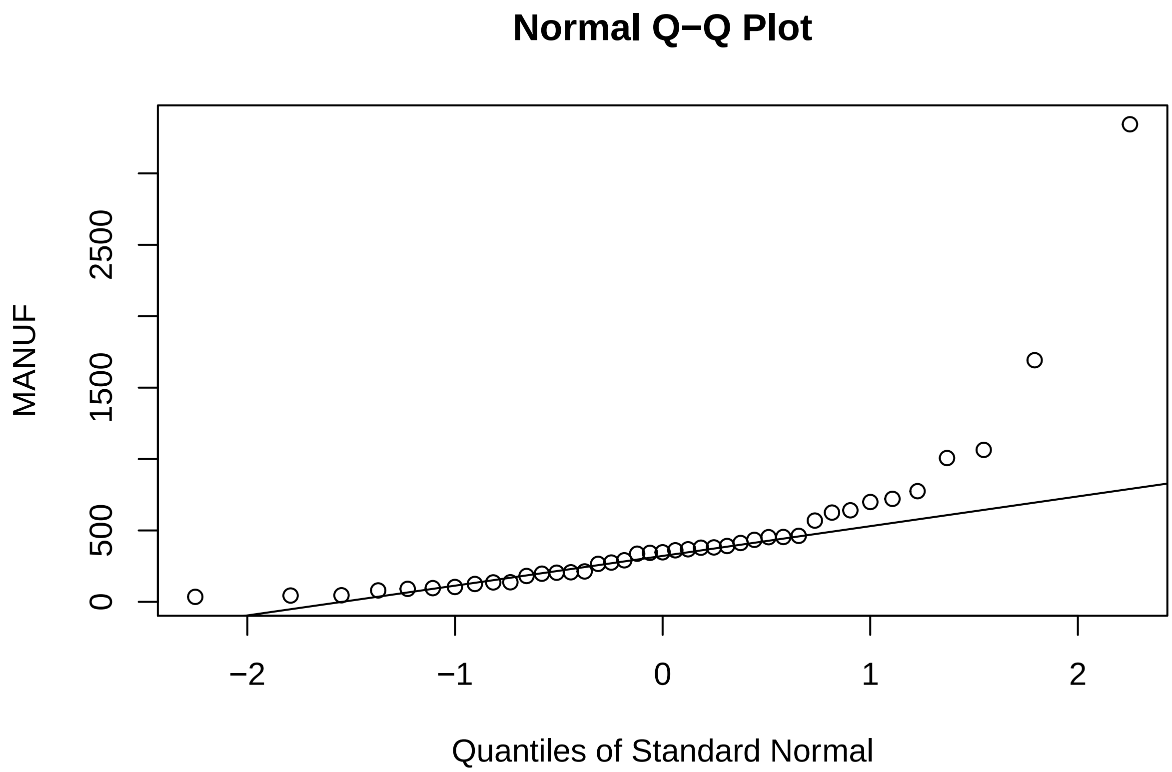
```
##                SO2        TEMP       MANUF         POP        WIND      PRECIP
## SO2     1.00000000 -0.43360020  0.64476873  0.49377958  0.09469045  0.05429434
## TEMP   -0.43360020  1.00000000 -0.19004216 -0.06267813 -0.34973963  0.38625342
## MANUF   0.64476873 -0.19004216  1.00000000  0.95526935  0.23794683 -0.03241688
## POP     0.49377958 -0.06267813  0.95526935  1.00000000  0.21264375 -0.02611873
## WIND    0.09469045 -0.34973963  0.23794683  0.21264375  1.00000000 -0.01299438
## PRECIP  0.05429434  0.38625342 -0.03241688 -0.02611873 -0.01299438  1.00000000
## DAYS    0.36956363 -0.43024212  0.13182930  0.04208319  0.16410559  0.49609671
##               DAYS
## SO2     0.36956363
## TEMP   -0.43024212
## MANUF   0.13182930
## POP     0.04208319
## WIND    0.16410559
## PRECIP  0.49609671
## DAYS    1.00000000
```

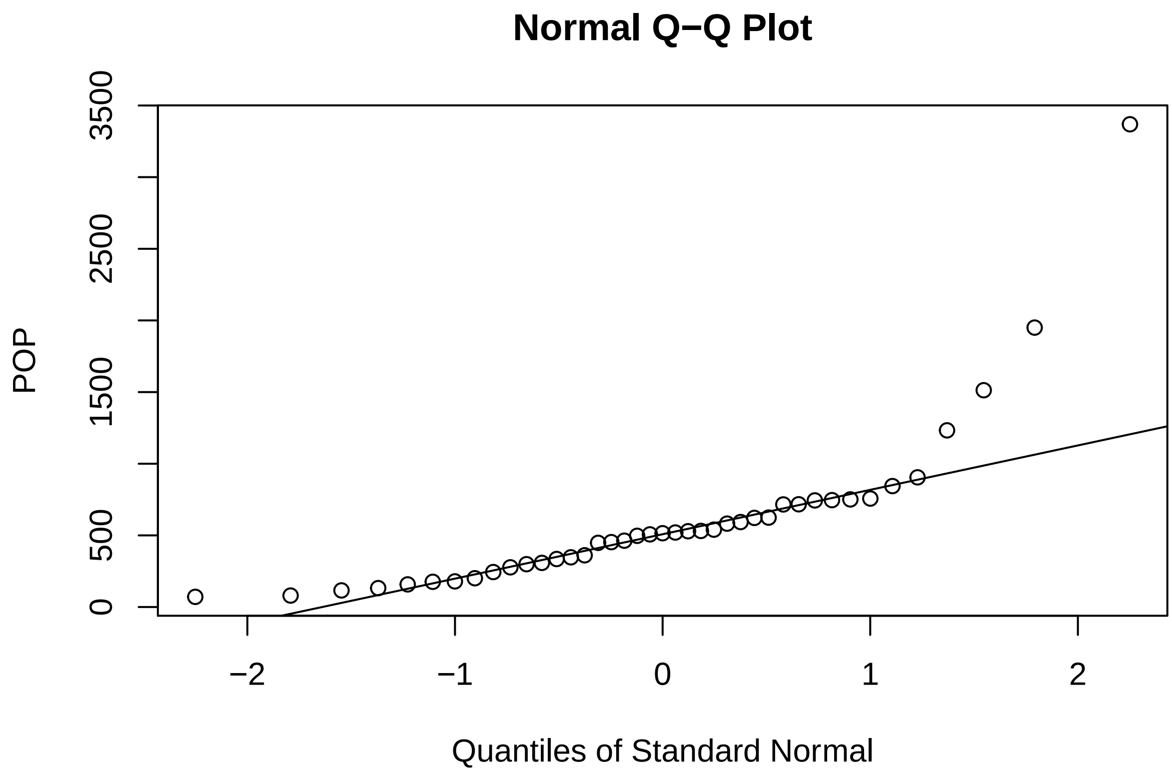
(c)

```
for(i in 1:ncol(dat1)){qqnorm(dat1[,i],
                        xlab="Quantiles of Standard Normal",
                        ylab=colnames(dat1)[i])
                        qqline(dat1[,i])
}
```

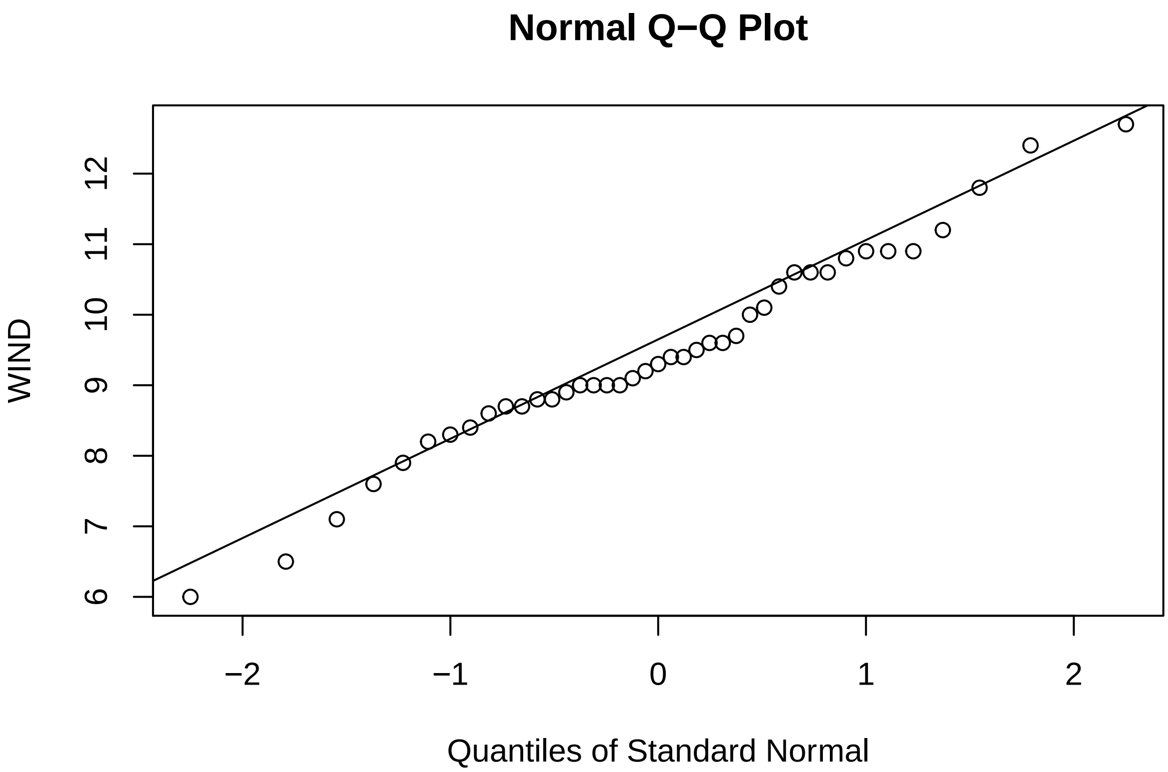


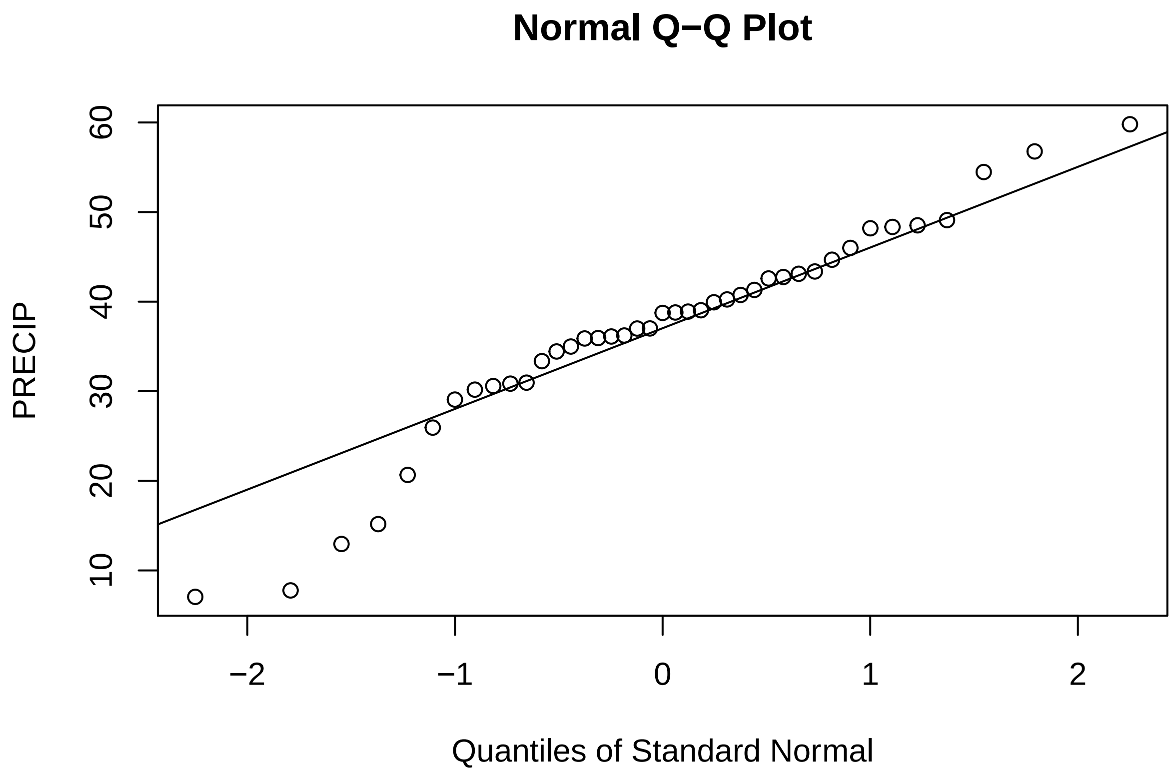

Normal Q−Q Plot
TEMP
75
70
65
60
55
50
45
−2
−1
0
1
2
Quantiles of Standard Normal

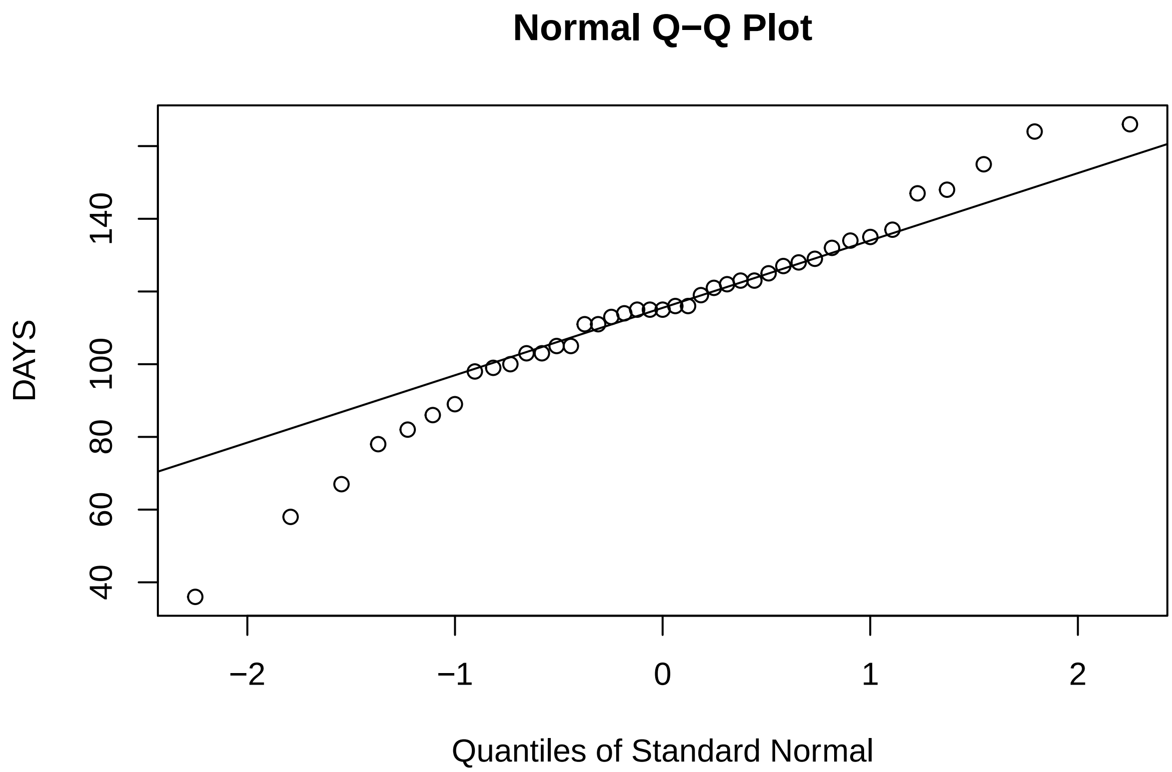

Normal Q−Q Plot
MANUF
0
500
1500
2500
−2
−1
0
1
2
Quantiles of Standard Normal



Normal Q−Q Plot
POP
0
500
1500
2500
3500
−2
−1
0
1
2
Quantiles of Standard Normal


Normal Q−Q Plot
WIND
12
11
10
9
8
7
6
−2
−1
0
1
2
Quantiles of Standard Normal


Normal Q−Q Plot
PRECIP
10
20
30
40
50
60
−2
−1
0
1
2
Quantiles of Standard Normal




```r
apply(dat1, 2, shapiro.test)
```

```
## $SO2
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.81165, p-value = 9.723e-06
##
##
## $TEMP
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.93554, p-value = 0.02215
##
##
## $MANUF
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
```

```
## W = 0.60548, p-value = 2.781e-09
##
##
## $POP
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.68049, p-value = 3.623e-08
##
##
## $WIND
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.98057, p-value = 0.6973
##
##
## $PRECIP
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.94214, p-value = 0.03725
##
##
## $DAYS
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.9654, p-value = 0.2419
```

When applying Shapiro-Wilk normality test, we can observe that except for variable Wind and Days other variables don't hold the normal distribution assumption.

(d)

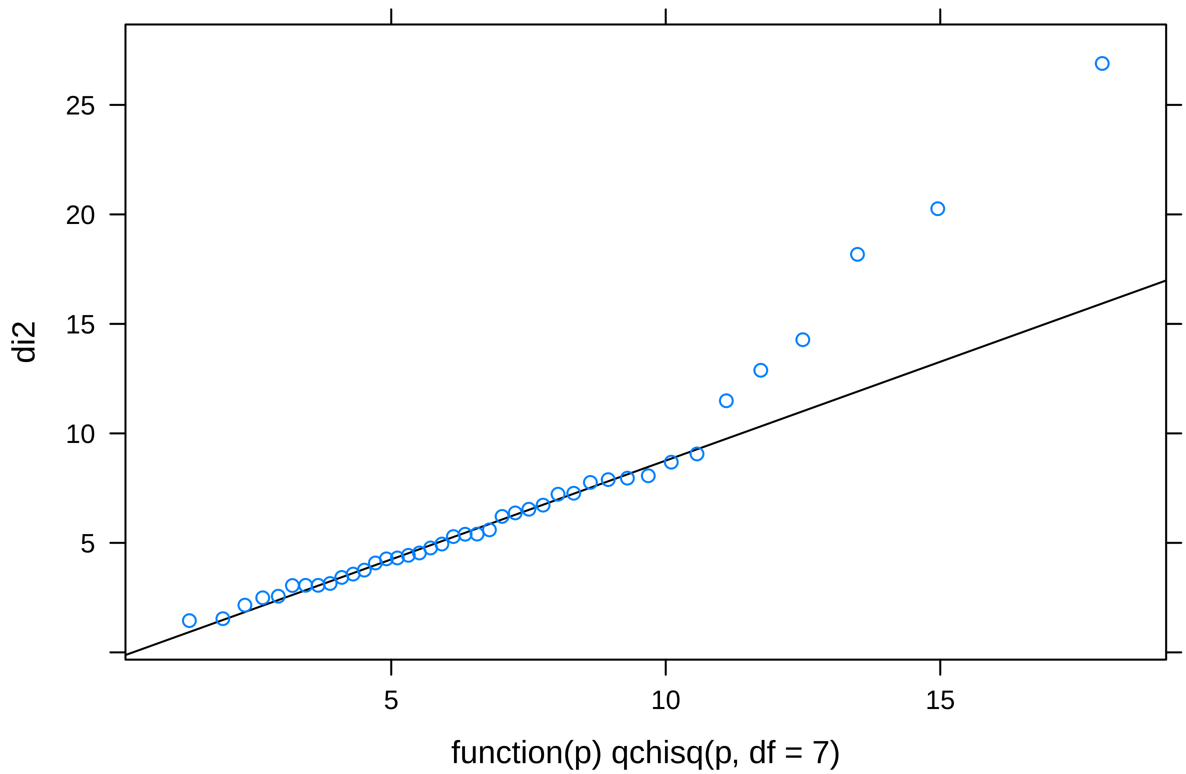
```
Sx = cov(dat1)
di2=mahalanobis(dat1, colMeans(dat1), Sx)
di2
```

```
##   [1] 20.258912  6.207429  4.539210  5.400965  7.760199  2.156830  1.538300
##   [8]  5.286583 14.277037  1.448871 26.891450  4.270100  4.310154  9.060638
##  [15]  3.060861  5.394046  3.421276  7.222633  4.945830  3.060650  4.767204
##  [22]  3.052545  8.063093  4.081128 12.880983  7.265585 11.489013  3.145760
##  [29]  6.722708  7.955753 18.176040  3.573384  2.564959  6.368290  8.684843
```

```
## [36]  4.431380  3.754730  2.492672  6.535345  7.888784  5.593824
```

(e)

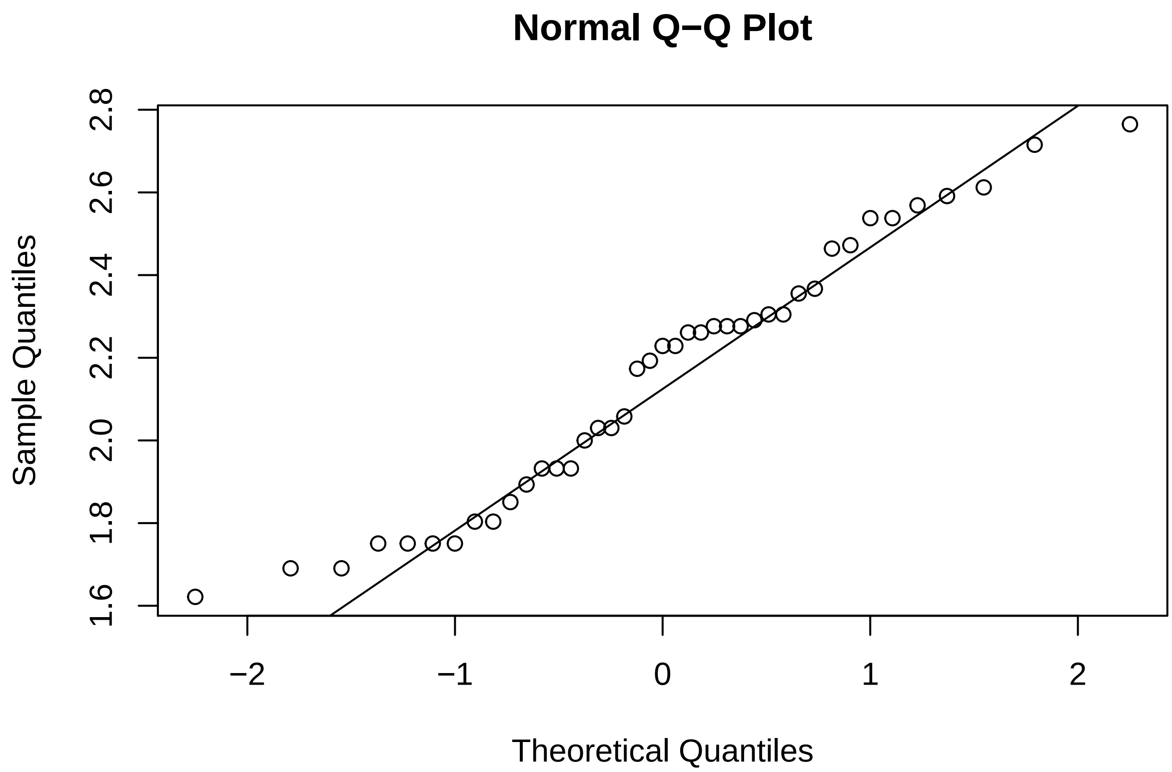
```
library(lattice)
qqmath(di2, distribution = function(p) qchisq(p,df=7),
     panel = function(x, ...) {
         panel.qqmathline(x, ...)
         panel.qqmath(x, ...)
       })
```



As we can observe from the chi-square plot of the generalized distance, the Mahalanobis distances don't resemble a straight and have outliers. This implies that the data don't arise from multivariate normal distribution.

(f)

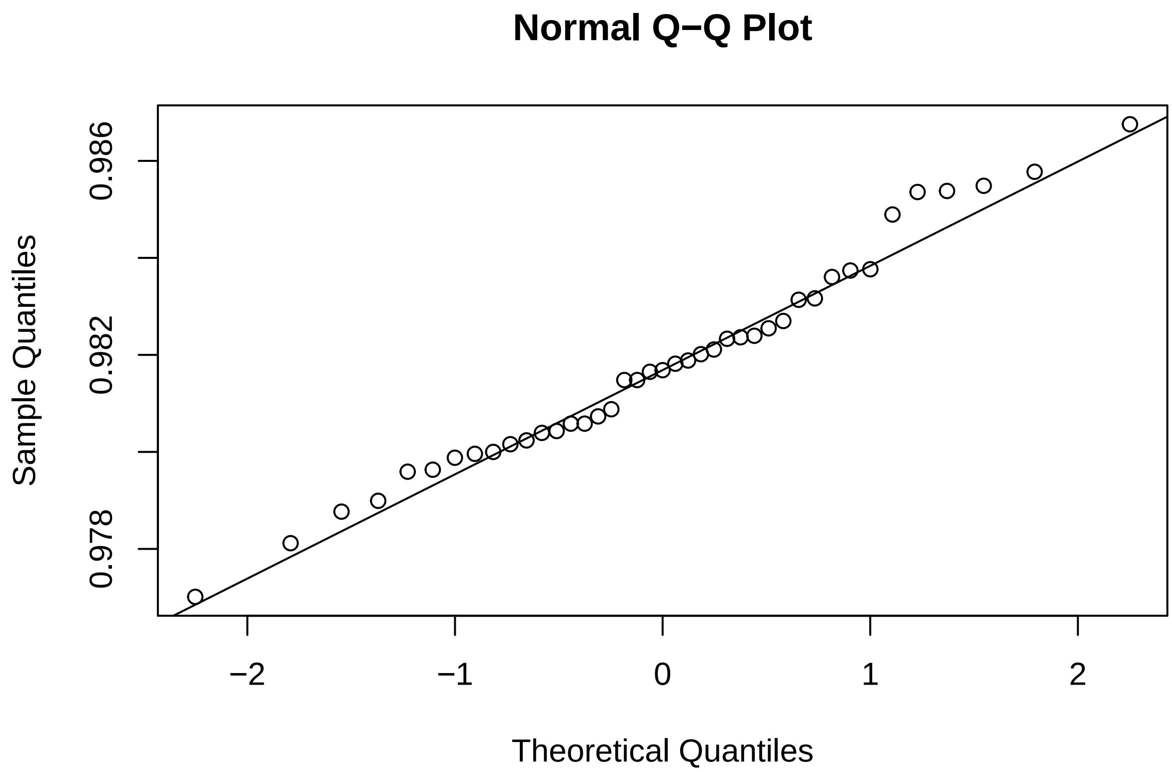
```
library(forecast)
lambda=BoxCox.lambda(dat1$SO2, method = "loglik")
new_var=BoxCox(dat1$SO2, lambda)
qqnorm(new_var)
qqline(new_var)
```




```r
shapiro.test(new_var)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_var
## W = 0.95782, p-value = 0.1319
```

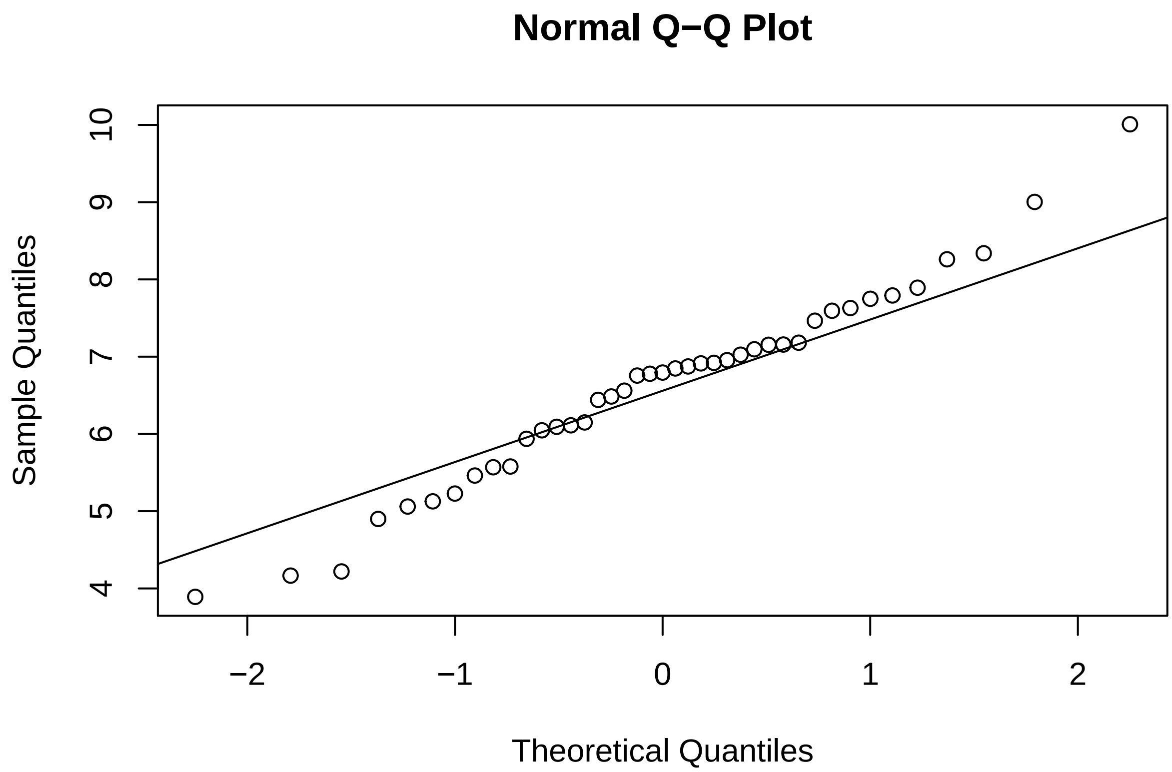
```r
lambda=BoxCox.lambda(dat1$TEMP, method = "loglik")
new_var=BoxCox(dat1$TEMP, lambda)
qqnorm(new_var)
qqline(new_var)
```




```r
shapiro.test(new_var)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_var
## W = 0.97975, p-value = 0.6659
```

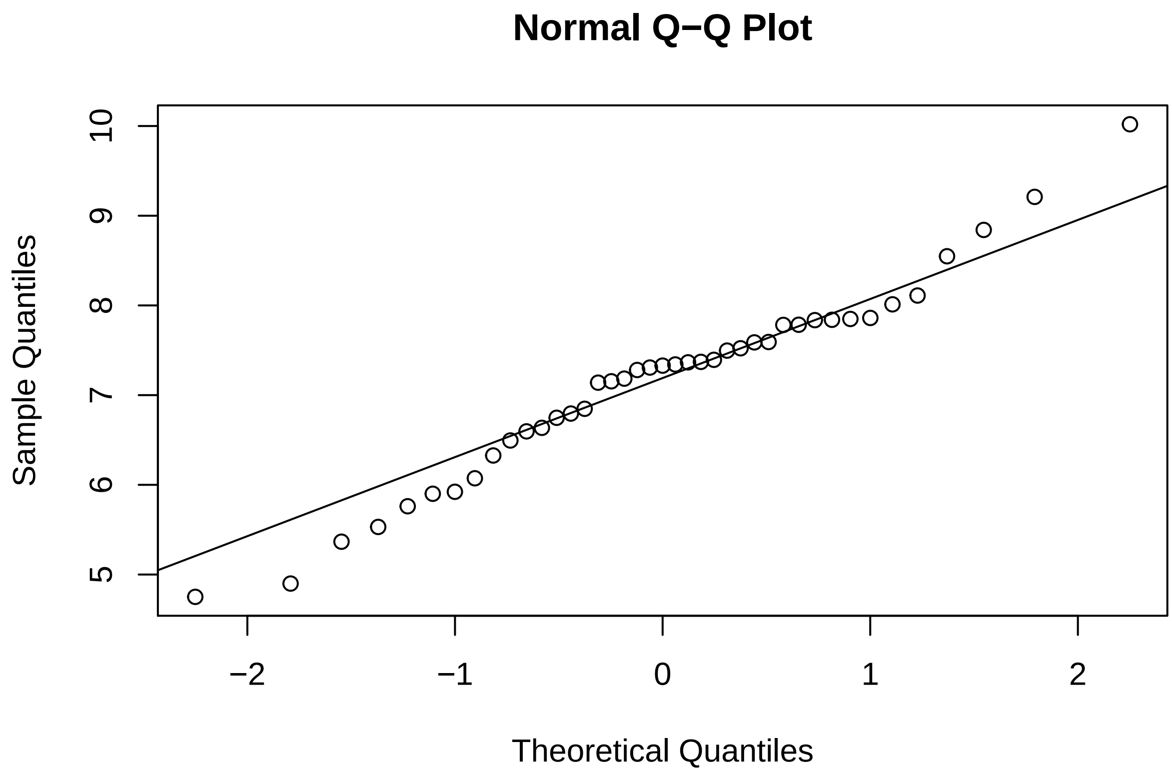
```r
lambda=BoxCox.lambda(dat1$MANUF, method = "loglik")
new_var=BoxCox(dat1$MANUF, lambda)
qqnorm(new_var)
qqline(new_var)
```




```r
shapiro.test(new_var)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_var
## W = 0.98089, p-value = 0.7094
```

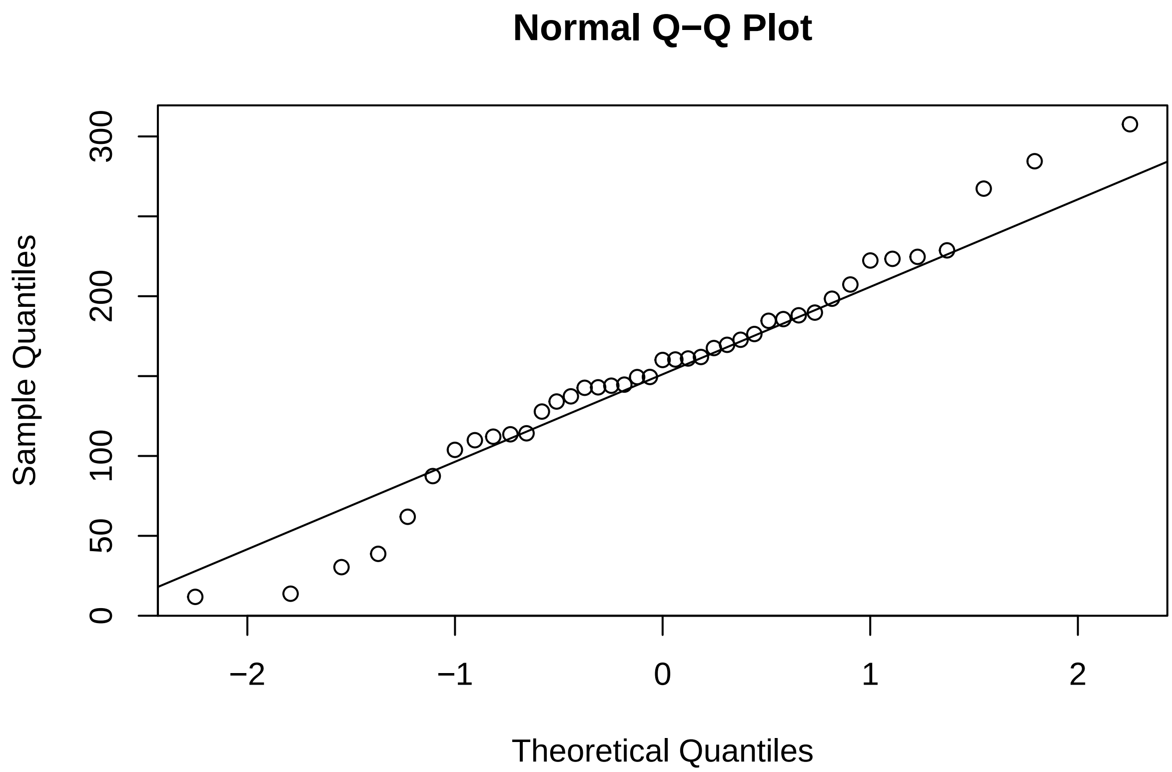
```r
lambda=BoxCox.lambda(dat1$POP, method = "loglik")
new_var=BoxCox(dat1$POP, lambda)
qqnorm(new_var)
qqline(new_var)
```




```r
shapiro.test(new_var)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_var
## W = 0.97206, p-value = 0.4014
```

```r
lambda=BoxCox.lambda(dat1$PRECIP, method = "loglik")
new_var=BoxCox(dat1$PRECIP, lambda)
qqnorm(new_var)
qqline(new_var)
```




```r
shapiro.test(new_var)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_var
## W = 0.97291, p-value = 0.4269
```

By transforming each variable with Box-Cox transformation with the optimal lambda, we could derive a p-value bigger than 0.05 when implementing Shapiro-Wilk normality test. This implies the null hypotheis which indicates that the noramlization assumption holds cannot be rejected.