Math 760

# Chapter 5 HW

Gabrielle Salamanca

April 14, 2024

## 3.

### (a) Use expression (5-15) to evaluate for the data in Exercise 5.1

The (5-15) expression is as such:

The dimensions of a matrix is n\*p, therefore our matrix is:

## [1] 4 2

Now, our matrix:

## [,1]  
## [1,] 6  
## [2,] 10

Let’s now solve for the numerator. First, the summation with :

## [,1] [,2]  
## [1,] 28 -6  
## [2,] -6 10

Now, the rest of the numerator:

## [1] 732

Now, let’s solve for the denominator.

## [1] 44

Now that we have all the parts, let’s plug it all into (5-15), and our answer is:

## The T-squared value is 13.63636

### 

### (b) Use the data in Exercise 5.1 to evaluate in (5-13). Also, evaluate Wilk’s lambda.

The (5-13) is:

We will need to divide the numerator by 3, but otherwise it’s just a matter of flipping the fraction from (a) and setting it to the power of .

## The value of the Lambda is 0.03251814

Now, the Wilk’s lambda formula is this: . Plugging in all the values, we get this:

## [1] 0.1803279

## 10. Refer to the bear growth data in Example 1.10 (see Table 1.4). Restrict your attention to the measurements of length

### (a) Obtain the 95% simultaneous confidence intervals for the four population means for length.

The simultaneous confidence interval formula is:

Then, the confidence interval for length2 is:

## ( 132.9837 , 153.5877 )

For length3, the confidence interval is:

## ( 132.9074 , 185.6641 )

For length4, the confidence interval is:

## ( 162.6495 , 183.6362 )

For length5, the confidence interval is:

## ( 159.3459 , 194.9399 )

### 

### (b) Refer to Part a. Obtain the 95% simultaneous confidence intervals for the three successive yearly increases in mean length

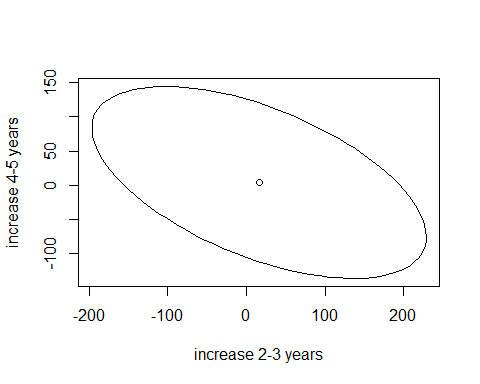
## Length2 - Length3: ( -8.283481 , 40.28348 )

## Length3 - Length4: ( -38.05854 , 10.34425 )

## Length4 - Length5: ( -10.37438 , 18.37438 )

### 

### (c) Obtain the 95% confidence ellipse for the mean increase in length from 2 to 3 years and the mean increase in length from 4 to 5 years.



### (d) Refer to Parts a and b. Construct the 95% Bonferroni confidence intervals for the set consisting of four mean lengths and three successive yearly increases in mean length.

## Length 2: ( 138.0904 , 148.481 )

## Length 3: ( 145.9831 , 172.5883 )

## Length 4: ( 167.8511 , 178.4347 )

## Length 5: ( 168.1678 , 186.1179 )

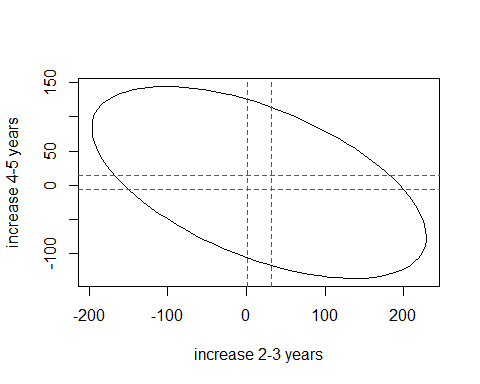
## Length 2-3: ( 3.753818 , 28.24618 )

## Length 3-4: ( -26.06193 , -1.652357 )

## Length 4-5: ( -3.249013 , 11.24901 )

### 

### (e) Refer to Parts c and d. Compare the 95% Bonferroni confidence rectangle for the mean increase in length from 2 to 3 years and the mean increase in length from 4 to 5 years with the confidence ellipse produced by the -procedure.



While I do not believe my 95% Bonferroni confidence rectangle isn’t exactly right, I would still say it’s much smaller and more informative than the 95% confidence ellipse. This is simply because from previous results, we learned that intervals are longer in length compared to Bonferroni ones, therefore the latter would have a more contained area.

## 12. Given the data

**with missing components, use the prediction-estimation algorithm of Section 5.7 to estimate and . Determine the initial estimates, and iterate to find the first revised estimates.**

We’ll first find the initial sample averages.

## The sample averages are: 4 , 6 , and 2

Now, we can find .

## [,1] [,2] [,3]  
## [1,] 0.5 0 0.5  
## [2,] 0.0 2 0.0  
## [3,] 0.5 0 1.5

Now, based on Example 5.13, we will start doing the prediction steps.

## The predicted x31 is 4.333333

## The predicted x31-squared is 19.61111

## The predicted x42 and x43 are 6 3

## The predicted x42 and x43-squared and products are 38 18 18 10

Now, we’ll jump into the predicted data sufficient statistics.

## The T1 matrix is

## [,1]  
## [1,] 16.33333  
## [2,] 24.00000  
## [3,] 9.00000

##   
## The T2 matrix is

## [,1] [,2] [,3]  
## [1,] "68.7777777777778" " " " "   
## [2,] "98.6666666666667" "154" " "   
## [3,] "40" "54" "28"

Now, we can find our first revised estimates of and .

## The first revised estimate of mu is:

## [,1]  
## [1,] 4.083333  
## [2,] 6.000000  
## [3,] 2.250000

## The first revised estimate of Sigma is:

## [,1] [,2] [,3]  
## [1,] 0.5208333 0.1666667 0.8125  
## [2,] 0.1666667 2.5000000 0.0000  
## [3,] 0.8125000 0.0000000 1.9375

## 

## 15. Let and be the ith and kth components, respectively, of .

### (a) Show that and , .

The general expected value formula is: . We also know generally probability is either or ; and if not given specific values for the ’s, we assume it’s either 1 or 0. Thus, .

The general variance formula is: .

And thus:

### 

### (b) Show that . Why must this covariance necessarily be negative?

The general covariance formula is: . Thus, .

, because we are assuming the x’s are independent.

Covariance can be negative, positive, or zero. But in terms of this question, it must be necessarily be negative, because there is an inverse relationship between the variables.

## 18. Use the college test data in Table 5.2 (See Example 5.5)

### (a) Test the null hypothesis vs at the level of significance. Suppose [500,50,30]’ represent average scores for thousands of college students over the last 10 years. Is there reason to believe that the group of students represented by the scores in Table 5.2 is scoring differently? Explain.

vs

According to (5-7), would be rejected if

## The critical region for this alpha is 8.557973

##   
## Hotelling's one sample T2-test  
##   
## data: college  
## T.2 = 72.706, df1 = 3, df2 = 84, p-value < 2.2e-16  
## alternative hypothesis: true location is not equal to c(500,50,30)

We find that the p-value is small and , therefore we reject . This means, the average scores for thousands of college students over the last 10 years do differ significantly from the students in Table 5.2. The scores in Table 5.2 could be scoring differently because of sample size and perhaps because these students are scoring notably higher or lower in the chosen subject.

### 

### (b) Determine the lengths and directions for the axes 95% confidence ellipsoid for .

The formula for finding the length and direction of the axes of the 95% confidence ellipsoid is: , where and are the eigenvalues and eigenvectors from the sample covariance matrix.

We will first find the direction of the 3 axes.

The direction of the social science & history axis is:

## [ 0.9939054 0.1034434 0.03809906 ]

The direction of the verbal axis is:

## [ 0.1037315 -0.9945892 -0.005660238 ]

The direction of the science axis is:

## [ -0.0373074 -0.009577815 0.9992579 ]

Now, we find the axes’ lengths.

## The axis length of social science & history is 23.73

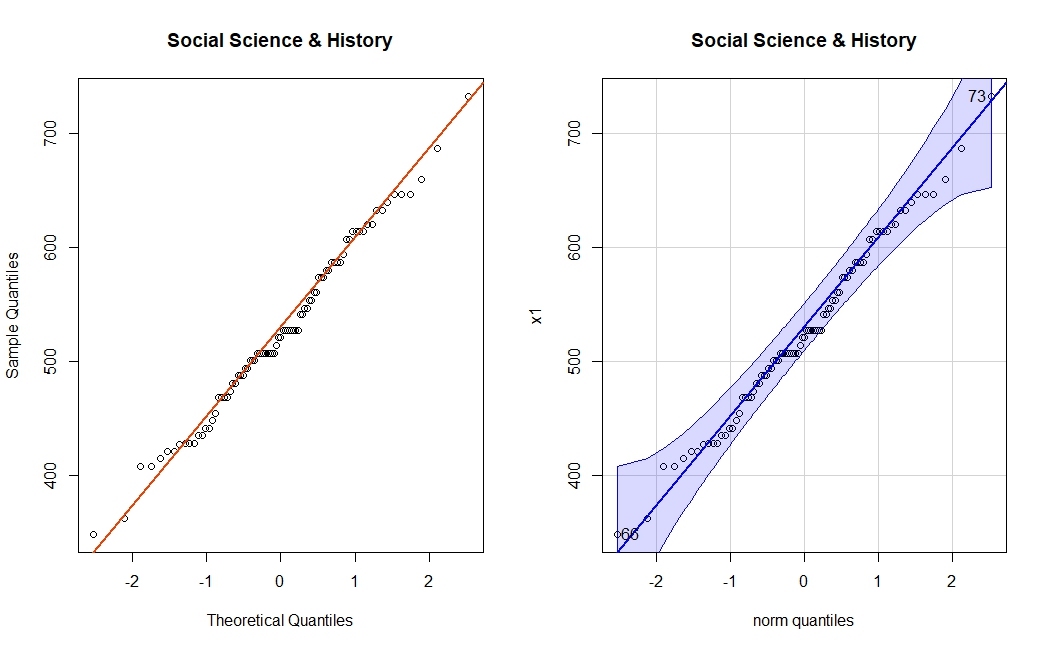
## The axis length of verbal is 2.472768

## The axis length of science is 1.1825

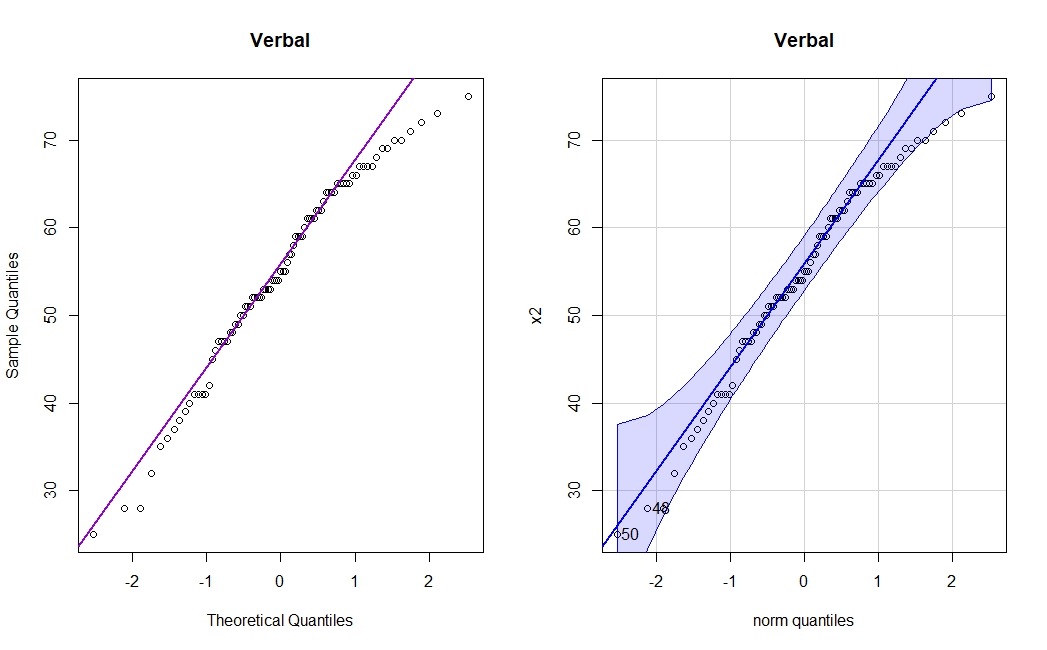
### 

### (c) Contstruct Q-Q plots from the marginal distributions of social science and history, verbal, and science scores. Also, construct the three possible scatter diagrams from the pairs of observations on different variables. Do these data appear to be normally distributed? Discuss.

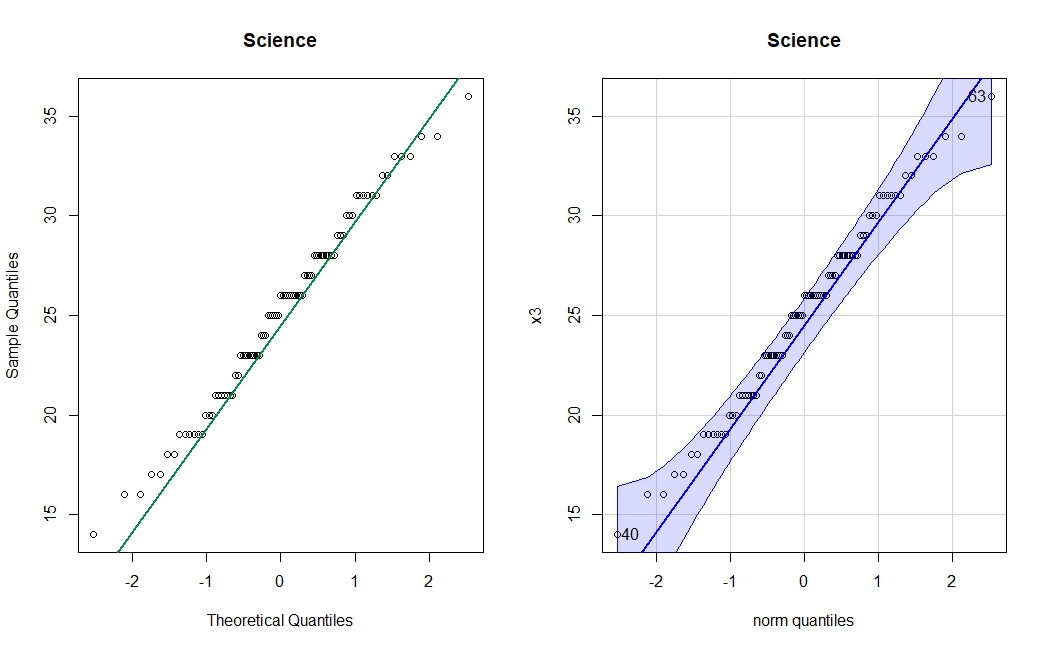
Let’s first look at the Q-Q plots of each variable.



## [1] 73 66



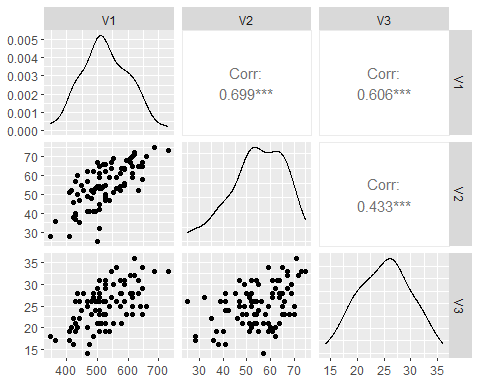
## [1] 50 48



## [1] 40 63

Overall, each variable the college data seems to follow a normal distribution, if straying just a bit from their respective lines. I would definitely want to investigate the verbal column, because it does seem the points stray farther from their line compared to the other two, and a few do seem to partially step out of the normality region.

Now, let’s look over the paired scatter diagrams.

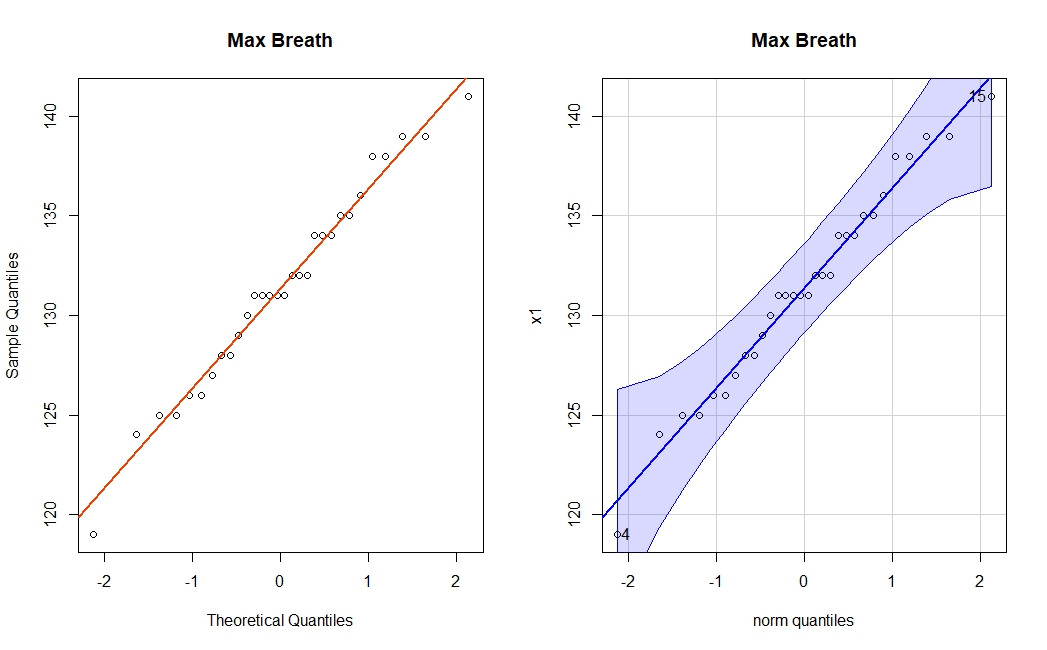


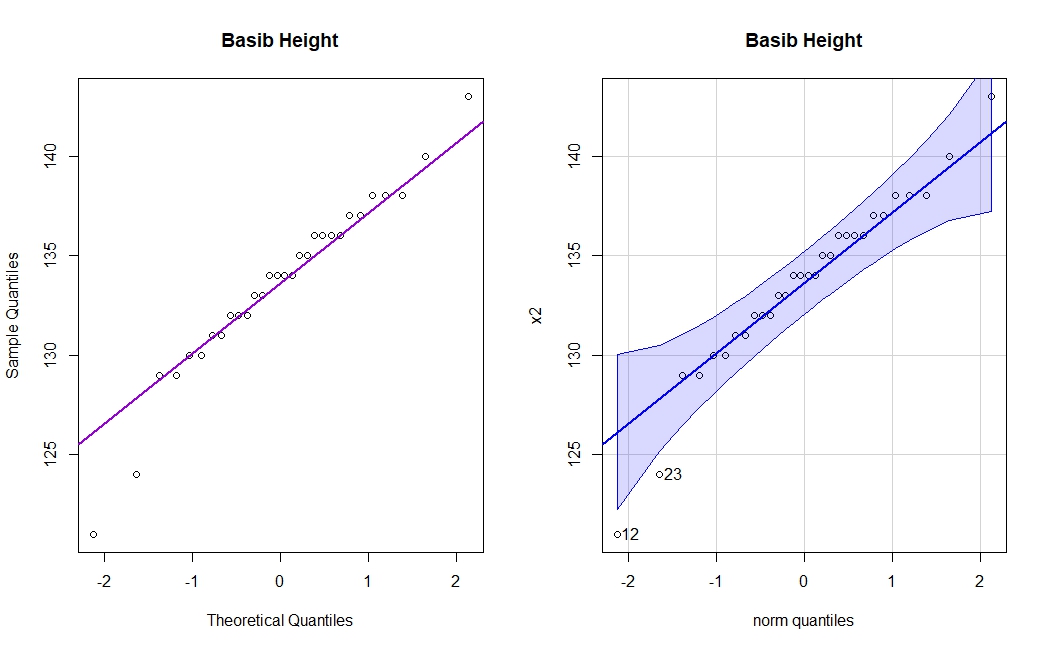
If we were to draw an ellipse over each paired scatter plot, it would at least seem that a majority of the points fall within it. And we know from Chapter 4, if roughly 50% of the points fall within the ellipse, we may assume normality.

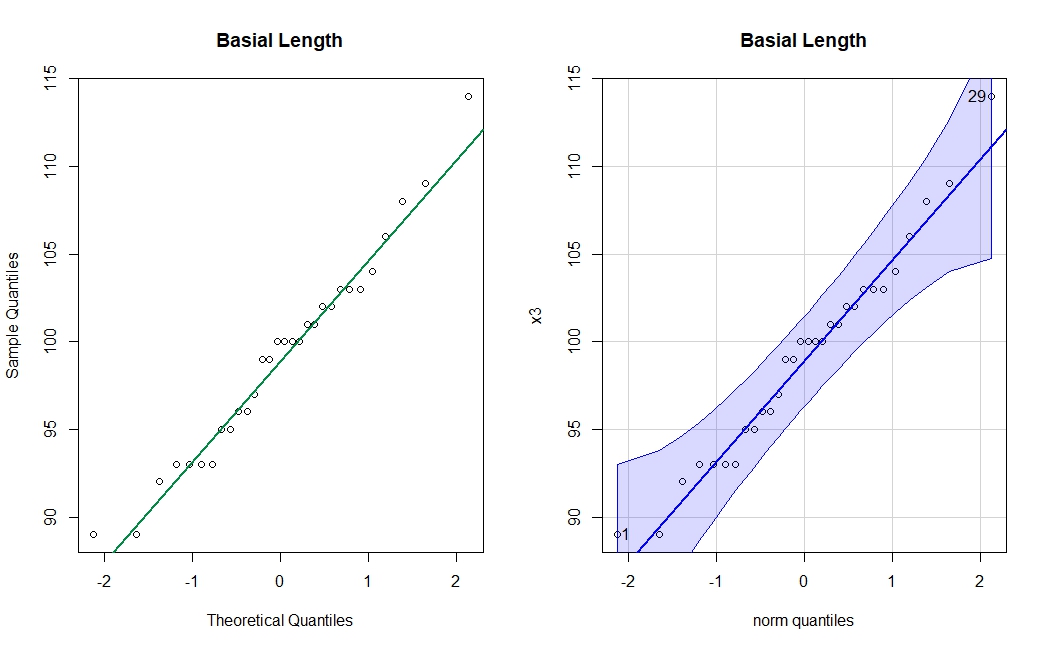
## 23. Consider the 30 observations on male Egyptian skulls for the first time period given in Table 6.13 on page 349.

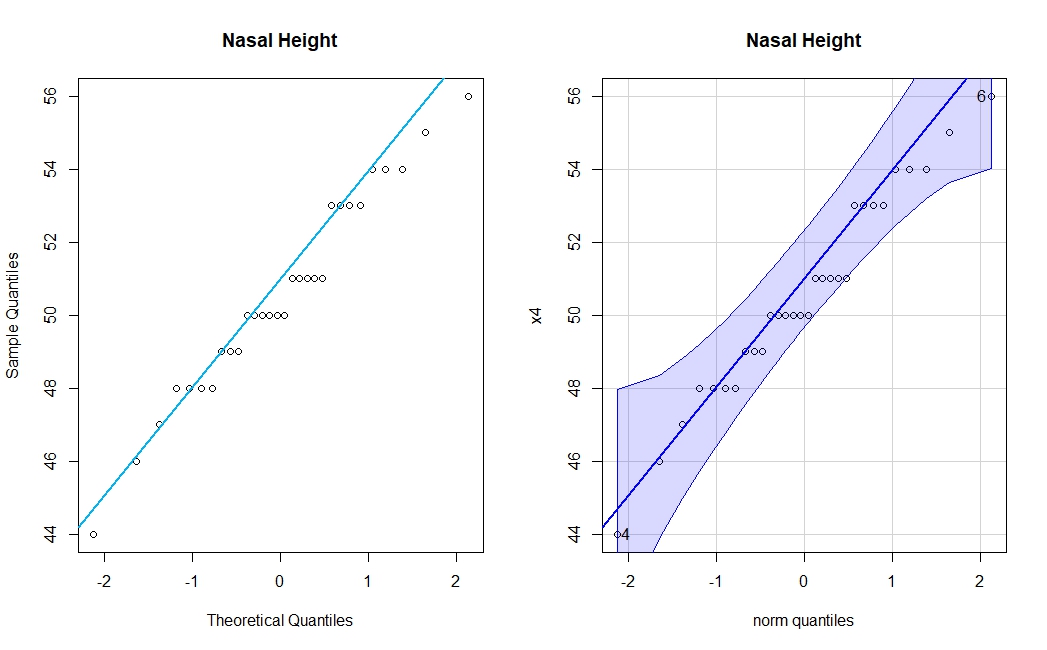
### (a) Construct Q-Q plots of the marginal distributions of the maxbreath, basheight, baslength, and nasheight variables. Also, construct a chi-square plot of the multivariate observations. Do these data appear to be normally distributed? Explain.

Let’s first look at the Q-Q plots of each variable.

## [1] 4 15

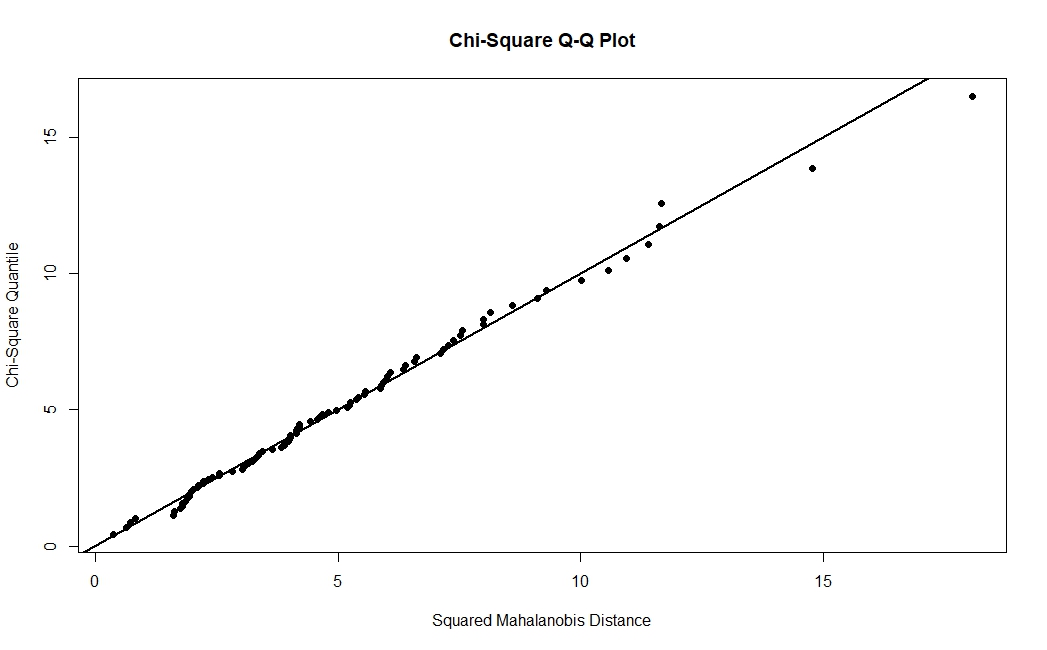
## [1] 12 23

## [1] 29 1

## [1] 4 6

Overall, each variable looks normal when looking at both versions of their plot.

Let’s look at the Q-Q plot.



With this plot, I would say the skull data follows a multivariate normal.

### (b) Construct 95% Bonferroni intervals for the individual skull dimension variables. Also, find the 95% -intervals. Compare the two sets of intervals.

The simultaneous Bonferroni intervals for each skull variable are:

## Max Breadth: ( 128.8727 , 133.8607 )

## Basibregmatic Height: ( 131.427 , 135.773 )

## Basialveolar Length: ( 96.30548 , 102.0279 )

## Nasal Height: ( 49.18965 , 51.87702 )

The 95% intervals for each skull variable are:

## Max Breadth: ( 126.6252 , 136.1081 )

## Basibregmatic Height: ( 129.4688 , 137.7312 )

## Basialveolar Length: ( 93.72711 , 104.6062 )

## Nasal Height: ( 47.97878 , 53.08789 )

If we subtracted each of these intervals,

## Bonferroni T2  
## [1,] 4.987998 9.482954  
## [2,] 4.345980 8.262379  
## [3,] 5.722376 10.879120  
## [4,] 2.687371 5.109107

We find that the Bonferroni inetervals are roughly half the length of the intervals.

## 

## 30. Refer to the data on energy consumption in Exercise 3.18.

### (a) Obtain the large sample 95% Bonferroni confidence intervals for the mean consumption of each of the four types, the total of the four, and the differnce, petroleum minus natural gas

The Bonferroni confidence interval formula is: .

Therefore, each of the 95% Bonferroni confidence intervals for each type is:

## Petroleum: ( 0.4520686 , 1.079931 )

## Natural Gas: ( 0.2996904 , 0.7163096 )

## Coal: ( 0.3752871 , 0.5007129 )

## Nuclear: ( 0.1452301 , 0.1767699 )

## Total: ( 1.272276 , 2.473724 )

## Petroleum - Gas: ( 0.1523782 , 0.3636218 )

### 

### (b) Obtain the large sample 95% simultaneous intervals for the mean consumption of each of the four tyopes, the total of the four, and the difference, petroleum minus natural gas. Compare your results for Part a.

The confidence interval formula is: .

Therefore, each of the 95% confidence intervals for each type is:

## Petroleum: ( 0.3931193 , 1.138881 )

## Natural Gas: ( 0.2605745 , 0.7554255 )

## Coal: ( 0.363511 , 0.512489 )

## Nuclear: ( 0.1422688 , 0.1797312 )

## Total: ( 1.159474 , 2.586526 )

## Petroleum - Gas: ( 0.1325448 , 0.3834552 )

We know from Problem 23, intervals are roughly twice the length of Bonferroni intervals. Also, its critical region (shown below) is larger than Bonferroni’s, so intervals will be wider.

## The Bonferroni crit region is 2.59326

## The T2 crit region is 3.080216

**Appendix**

knitr::opts\_chunk$set(echo = FALSE)  
library(car)  
library(DescTools)  
library(ellipse)  
library(GGally)  
library(ggplot2)  
library(graphics)  
library(gridExtra)  
library(investr)  
library(matlib)  
library(MVN)  
library(robustbase)  
library(SIBER)  
xMat <- c(2,12,8,9,6,9,8,10)  
x <- matrix(xMat, nrow = 4, ncol = 2, byrow = TRUE)  
muMat <- c(7,11)  
mu <- matrix(muMat, nrow = 2, ncol = 1, byrow = TRUE)  
dim(x)  
n <- dim(x)[1]  
one <- as.matrix(rep(1, dim(x)[1]))  
xbar <- 1/n\*t(x)%\*%one  
xbar  
sum1 <- 0  
# mu for loop  
for (i in 1:nrow(x)) {  
 subtract <- x[i,] - mu  
 multi <- subtract %\*% t(subtract)  
 sum1 <- sum1 + multi  
}  
print(sum1)  
mutrix <- (28\*10) - (-6)\*(-6)  
num <- (n-1)\*mutrix  
num  
sum2 <- 0  
# xbar for loop  
for (i in 1:nrow(x)) {  
 subtract <- x[i,] - xbar  
 multi <- subtract %\*% t(subtract)  
 sum2 <- sum2 + multi  
}  
denom <- (24\*6) - (-10)\*(-10)  
print(denom)  
t2 <- (num/denom) - (n-1)  
cat("The T-squared value is", t2)  
lambda <- (denom/(num/3))^(n/2)  
cat("The value of the Lambda is", lambda)  
lambda^(2/n)  
bear <- read.table("D:/Coding/R Storage/T1-4.dat", header = FALSE, sep = " ")  
# new data  
length <- bear[,5:8]  
n <- dim(length)[1]  
p <- dim(length)[2]  
x2 <- length$V5 #length2  
x3 <- length$V6 #length3  
x4 <- length$V7 #length4  
x5 <- length$V8 #length5  
length <- as.matrix(length)  
# xbar  
one <- as.matrix(rep(1,dim(length)[1]))  
n <- dim(length)[1]  
xbar <- 1/n \* t(length)%\*%one  
# covariance  
mean\_matrix <- matrix(data = 1, nrow = n)%\*%cbind(xbar[[1]], xbar[[2]],   
 xbar[[3]], xbar[[4]])  
xstar <- length - mean\_matrix  
covar <- 1/(n-1)\*t(xstar)%\*%xstar  
# vars  
crit <- qf(0.05, df1 = n, df2 = p, lower.tail = FALSE)  
frac <- (p\*(n-1))/(n-p)  
right <- crit\*frac  
neg2 <- xbar[1] - sqrt(frac\*crit\*(covar[1,1]/n))  
pos2 <- xbar[1] + sqrt(frac\*crit\*(covar[1,1]/n))  
cat("(", neg2, ",", pos2, ")")  
neg3 <- xbar[2] - sqrt(frac\*crit\*(covar[2,2]/n))  
pos3 <- xbar[2] + sqrt(frac\*crit\*(covar[2,2]/n))  
cat("(", neg3, ",", pos3, ")")  
neg4 <- xbar[3] - sqrt(frac\*crit\*(covar[3,3]/n))  
pos4 <- xbar[3] + sqrt(frac\*crit\*(covar[3,3]/n))  
cat("(", neg4, ",", pos4, ")")  
neg5 <- xbar[4] - sqrt(frac\*crit\*(covar[4,4]/n))  
pos5 <- xbar[4] + sqrt(frac\*crit\*(covar[4,4]/n))  
cat("(", neg5, ",", pos5, ")")  
neg23 <- (xbar[2]-xbar[1]) - sqrt(frac\*crit\*((covar[2,2]-covar[1,1])/n))  
pos23 <- (xbar[2]-xbar[1]) + sqrt(frac\*crit\*((covar[2,2]-covar[1,1])/n))  
cat("Length2 - Length3: (", neg23, ",", pos23, ") \n")  
# 3-4  
neg34 <- (xbar[2]-xbar[3]) - sqrt(frac\*crit\*((covar[2,2]-covar[3,3])/n))  
pos34 <- (xbar[2]-xbar[3]) + sqrt(frac\*crit\*((covar[2,2]-covar[3,3])/n))  
cat("Length3 - Length4: (", neg34, ",", pos34, ") \n")  
# 4-5  
neg45 <- (xbar[4]-xbar[3]) - sqrt(frac\*crit\*((covar[4,4]-covar[3,3])/n))  
pos45 <- (xbar[4]-xbar[3]) + sqrt(frac\*crit\*((covar[4,4]-covar[3,3])/n))  
cat("Length4 - Length5: (", neg45, ",", pos45, ")")  
a <- matrix(c(-1,1,0,0), ncol = 1)  
b <- matrix(c(0,0,-1,1), ncol = 1)  
# T2  
meandiff <- c(t(a)%\*%xbar, t(b)%\*%xbar)  
Sdiff <- matrix(c(t(a)%\*%covar%\*%a, t(a)%\*%covar%\*%b, t(b)%\*%covar%\*%a,  
 t(b)%\*%covar%\*%b), 2, 2)  
plot(ellipse(Sdiff, centre = meandiff, t = right/sqrt(n)), type = "l",  
 xlab = "increase 2-3 years", yla = "increase 4-5 years")  
points(meandiff[1], meandiff[2])  
crit1 <- qt(0.05/(2\*p), df = n-1, lower.tail = FALSE)  
# L2  
bneg1 <- xbar[1] - crit1\*sqrt((covar[1,1]/n))  
bpos1 <- xbar[1] + crit1\*sqrt((covar[1,1]/n))  
cat("Length 2: (", bneg1, ",", bpos1, ") \n")  
# L3 height  
bneg2 <- xbar[2] - crit1\*sqrt((covar[2,2]/n))  
bpos2 <- xbar[2] + crit1\*sqrt((covar[2,2]/n))  
cat("Length 3: (", bneg2, ",", bpos2, ") \n")  
# L4  
bneg3 <- xbar[3] - crit1\*sqrt((covar[3,3]/n))  
bpos3 <- xbar[3] + crit1\*sqrt((covar[3,3]/n))  
cat("Length 4: (", bneg3, ",", bpos3, ") \n")  
# L5  
bneg4 <- xbar[4] - crit1\*sqrt((covar[4,4]/n))  
bpos4 <- xbar[4] + crit1\*sqrt((covar[4,4]/n))  
cat("Length 5: (", bneg4, ",", bpos4, ") \n")  
# L2-3  
bneg23 <- (xbar[2]-xbar[1]) - crit1\*sqrt((covar[2,2]-covar[1,1])/n)  
bpos23 <- (xbar[2]-xbar[1]) + crit1\*sqrt((covar[2,2]-covar[1,1])/n)  
cat("Length 2-3: (", bneg23, ",", bpos23, ") \n")  
# L3-4  
bneg34 <- (xbar[2]-xbar[3]) - crit1\*sqrt((covar[2,2]-covar[3,3])/n)  
bpos34 <- (xbar[2]-xbar[3]) + crit1\*sqrt((covar[2,2]-covar[3,3])/n)  
cat("Length 3-4: (", bneg34, ",", bpos34, ") \n")  
# L4-5  
bneg45 <- (xbar[4]-xbar[3]) - crit1\*sqrt((covar[4,4]-covar[3,3])/n)  
bpos45 <- (xbar[4]-xbar[3]) + crit1\*sqrt((covar[4,4]-covar[3,3])/n)  
cat("Length 4-5: (", bneg45, ",", bpos45, ") \n")  
a <- matrix(c(-1,1,0,0), ncol = 1)  
b <- matrix(c(0,0,-1,1), ncol = 1)  
# T2  
meandiff <- c(t(a)%\*%xbar, t(b)%\*%xbar)  
Sdiff <- matrix(c(t(a)%\*%covar%\*%a, t(a)%\*%covar%\*%b, t(b)%\*%covar%\*%a,  
 t(b)%\*%covar%\*%b), 2, 2)  
plot(ellipse(Sdiff, centre = meandiff, t = right/sqrt(n)), type = "l",  
 xlab = "increase 2-3 years", yla = "increase 4-5 years")  
#points(meandiff[1], meandiff[2])  
# Bonferroni  
amu.L=t(a)%\*%xbar-crit1\*sqrt(t(a)%\*%covar%\*%a/n)  
amu.U=t(a)%\*%xbar+crit1\*sqrt(t(a)%\*%covar%\*%a/n)  
bmu.L=t(b)%\*%xbar-crit1\*sqrt(t(b)%\*%covar%\*%b/n)  
bmu.U=t(b)%\*%xbar+crit1\*sqrt(t(b)%\*%covar%\*%b/n)  
abline(v = amu.L, lty = 2, col = "firebrick3")  
abline(v = amu.U, lty = 2, col = "firebrick3")  
abline(h = bmu.L, lty = 2, col = "firebrick3")  
abline(h = bmu.U, lty = 2, col = "firebrick3")  
xMat <- c(3,6,0,4,4,3,"-",8,3,5,"-","-")  
x <- matrix(xMat, nrow = 4, ncol = 3, byrow = TRUE)  
m1 <- (3+4+5)/3  
m2 <- (6+4+8)/3  
m3 <- (0+3+3)/3  
muMat <- c(m1,m2,m3)  
mu <- matrix(muMat, nrow = 3, ncol = 1, byrow = TRUE)  
cat("The sample averages are:", m1,",", m2,", and", m3)  
# row 1  
s11 <- ((3-m1)^2 + (4-m1)^2 + (m1-m1)^2 + (5-m1)^2)/4  
s12 <- ((3-m1)\*(6-m2) + (4-m1)\*(4-m2) + (m1-m1)\*(8-m2) + (5-m1)\*(m2-m2))/4  
s13 <- ((3-m1)\*(0-m3) + (4-m1)\*(3-m3) + (m1-m1)\*(3-m3) + (5-m1)\*(m3-m3))/4  
# row 2  
s21 <- ((6-m2)\*(3-m1) + (4-m2)\*(4-m1) + (8-m2)\*(m1-m1) + (m2-m2)\*(5-m1))/4  
s22 <- ((6-m2)^2 + (4-m2)^2 + (8-m2)^2 + (m2-m2)^2)/4  
s23 <- ((6-m2)\*(0-m3) + (4-m2)\*(3-m3) + (8-m2)\*(3-m3) + (m2-m2)\*(m3-m3))/4  
# row 3  
s31 <- ((0-m3)\*(3-m1) + (3-m3)\*(4-m1) + (3-m3)\*(m1-m1) + (m3-m3)\*(5-m1))/4  
s32 <- ((0-m3)\*(6-m2) + (3-m3)\*(4-m2) + (3-m3)\*(8-m2) + (m3-m3)\*(m2-m2))/4  
s33 <- ((0-m3)^2 + (3-m3)^2 + (3-m3)^2 + (m3-m3)^2)/4  
# matrix  
sigMat <- c(s11, s12, s13, s21, s22, s23, s31, s32, s33)  
sigma <- matrix(sigMat, nrow = 3, ncol = 3, byrow = TRUE)  
sigma  
# predict 31  
sig22Mat <- c(0,0.5)  
sig22.1 <- matrix(sig22Mat, nrow = 1, ncol = 2, byrow = TRUE)  
sig12Mat <- c(2,0,0,1.5)  
sig12.1 <- matrix(sig12Mat, nrow = 2, ncol = 2, byrow = TRUE)  
muxMat <- c(8 - m2, 3 - m3)  
mux <- matrix(muxMat, nrow = 2, ncol = 1, byrow = TRUE)  
x31 <- 4 + sig22.1 %\*% solve(sig12.1) %\*% mux   
sig31 <- 0.5 + sig22.1 %\*% solve(sig12.1) %\*% mux + x31^2  
x3Mat <- c(8,3)  
x3 <- matrix(x3Mat, nrow = 1, ncol = 2, byrow = TRUE)  
row3 <- x31%\*%x3  
# predict 42-43  
mat4 <- matrix(c(m2,m3), nrow = 2, ncol = 1, byrow = TRUE)  
sig12 <- matrix(c(0,0.5), nrow = 2, ncol = 1, byrow = TRUE)  
sig22 <- sigma[3,1]  
xmu <- matrix(5-m1, nrow = 1, ncol = , byrow = TRUE)  
x42.3 <- mat4 + sig12 %\*% solve(sig22) %\*% xmu  
sig4 <- sigma[2:3, 2:3] - (sig12 %\*% solve(sig22) %\*% sig22.1) + (x42.3 %\*% t(x42.3))  
row4 <- x42.3 %\*% 5  
# cat  
cat("The predicted x31 is", x31, "\n")  
cat("The predicted x31-squared is", sig31, "\n")  
cat("The predicted x42 and x43 are", x42.3, "\n")  
cat("The predicted x42 and x43-squared and products are", sig4)  
T1mat <- c(3+4+x31+5, 6+4+8+6, 0+3+3+3)  
T1 <- matrix(T1mat, nrow = 3, ncol = 1, byrow = TRUE)  
T2mat <- c(3^2+4^2+x31^2+5^2, " ", " ",   
 (3\*6)+(4\*4)+(x31\*8)+(5\*6), 6^2+4^2+8^2+38, " ",  
 (3\*0)+(4\*3)+(x31\*3)+(5\*3),(6\*0)+(4\*3)+(8\*3)+(6\*3),0^2+3^2+3^2+10)  
T2 <- matrix(T2mat, nrow = 3, ncol = 3, byrow = TRUE)  
# print  
cat("The T1 matrix is \n")  
T1  
cat("\n The T2 matrix is \n")  
T2  
n <- dim(x)[1]  
newMu <- 0.25 %\*% t(T1)  
cat("The first revised estimate of mu is: \n")  
t(newMu)  
# sigma  
newT2 <- c(68.7777777777778, 98.6666666666667, 40,  
 98.666666666666, 154, 54,  
 40, 54, 28)  
T2 <- matrix(newT2, nrow = 3, ncol = 3, byrow = TRUE)  
newSig <- (0.25\*T2) - t(newMu)%\*%newMu  
cat("\n The first revised estimate of Sigma is: \n")  
newSig  
college <- read.table("D:/Coding/R Storage/T5-2.dat", header = FALSE)  
# vars  
x1 <- college$V1 #social science & history  
x2 <- college$V2 #verbal  
x3 <- college$V3 #science  
# vars  
n <- 87  
p <- 3  
muMat <- c(500,50,30)  
mu <- matrix(muMat, nrow = 3, ncol = 1, byrow = TRUE)  
# testing  
crit <- qf(0.05, df1 = n, df2 = p, lower.tail = FALSE)  
cat("The critical region for this alpha is", crit, "\n")  
HotellingsT2Test(college, mu = mu)  
score <- as.matrix(college)  
# xbar  
one <- as.matrix(rep(1,dim(score)[1]))  
n <- dim(score)[1]  
xbar <- 1/n \* t(score)%\*%one  
# covariance  
mean\_matrix <- matrix(data = 1, nrow = n)%\*%cbind(xbar[[1]], xbar[[2]], xbar[[3]])  
xstar <- score - mean\_matrix  
covar <- 1/(n-1)\*t(xstar)%\*%xstar  
# inverse  
inv <- solve(covar)  
# eigen  
eigen <- eigen(covar)  
eigenval <- eigen$values  
eigenvec <- eigen$vectors  
cat("[",eigenvec[,1],"]")  
cat("[",eigenvec[,2],"]")  
cat("[",eigenvec[,3],"]")  
# function  
axis\_length <- function(lambda, n, p, alpha = .95) {  
return(sqrt(lambda\*(p\*(n-1))/(n\*(n-p))\*qf(alpha, p, n-p)))  
}  
# axis length  
ax1 <- axis\_length(eigenval[1], n, p)  
ax2 <- axis\_length(eigenval[2], n, p)  
ax3 <- axis\_length(eigenval[3], n, p)  
# cat  
cat("The axis length of social science & history is", ax1, "\n")  
cat("The axis length of verbal is", ax2, "\n")  
cat("The axis length of science is", ax3)  
par(mfrow = c(1,2))  
# x1  
qqnorm(x1, main = "Social Science & History")  
qqline(x1, col = "orangered2", lwd = 2)  
qqPlot(x1, main = "Social Science & History")  
# x2  
qqnorm(x2, main = "Verbal")  
qqline(x2, col = "darkviolet", lwd = 2)  
qqPlot(x2, main = "Verbal")  
# x3  
qqnorm(x3, main = "Science")  
qqline(x3, col = "springgreen4", lwd = 2)  
qqPlot(x3, main = "Science")  
ggpairs(college)  
skull <- read.table("D:/Coding/R Storage/T6-13.dat", header = FALSE)  
skullDim <- skull[1:30,1:4]  
# vars  
x1 <- skullDim$V1 #max breadth  
x2 <- skullDim$V2 #basibregmatic height  
x3 <- skullDim$V3 #basialveolar length  
x4 <- skullDim$V4 #nasal height  
par(mfrow = c(1,2))  
# x1  
qqnorm(x1, main = "Max Breath")  
qqline(x1, col = "orangered2", lwd = 2)  
qqPlot(x1, main = "Max Breath")  
# x2  
qqnorm(x2, main = "Basib Height")  
qqline(x2, col = "darkviolet", lwd = 2)  
qqPlot(x2, main = "Basib Height")  
# x3  
qqnorm(x3, main = "Basial Length")  
qqline(x3, col = "springgreen4", lwd = 2)  
qqPlot(x3, main = "Basial Length")  
# x4  
qqnorm(x4, main = "Nasal Height")  
qqline(x4, col = "deepskyblue2", lwd = 2)  
qqPlot(x4, main = "Nasal Height")  
mvn(skull, multivariatePlot = "qq")  
skullDim <- as.matrix(skullDim)  
n <- 30  
p <- 4  
# xbar  
one <- as.matrix(rep(1,dim(skullDim)[1]))  
xbar <- 1/n \* t(skullDim)%\*%one  
# covariance  
mean\_matrix <- matrix(data = 1, nrow = n)%\*%cbind(xbar[[1]], xbar[[2]], xbar[[3]], xbar[[4]])  
xstar <- skullDim - mean\_matrix  
S <- 1/(n-1)\*t(xstar)%\*%xstar  
# vars  
crit1 <- -qt(0.05/8, df = n-1)  
# max breadth  
bneg1 <- xbar[1] - crit1\*sqrt((S[1,1]/n))  
bpos1 <- xbar[1] + crit1\*sqrt((S[1,1]/n))  
cat("Max Breadth: (", bneg1, ",", bpos1, ") \n")  
# basibregmatic height  
bneg2 <- xbar[2] - crit1\*sqrt((S[2,2]/n))  
bpos2 <- xbar[2] + crit1\*sqrt((S[2,2]/n))  
cat("Basibregmatic Height: (", bneg2, ",", bpos2, ") \n")  
# basialveolar length  
bneg3 <- xbar[3] - crit1\*sqrt((S[3,3]/n))  
bpos3 <- xbar[3] + crit1\*sqrt((S[3,3]/n))  
cat("Basialveolar Length: (", bneg3, ",", bpos3, ") \n")  
# nasal height  
bneg4 <- xbar[4] - crit1\*sqrt((S[4,4]/n))  
bpos4 <- xbar[4] + crit1\*sqrt((S[4,4]/n))  
cat("Nasal Height: (", bneg4, ",", bpos4, ") \n")  
# crit  
crit2 <- qf(0.05, df1 = n, df2 = p, lower.tail = FALSE)  
frac <- (p\*(n-1))/(n-p)  
# max breadth  
tneg1 <- xbar[1] - sqrt(frac\*crit2\*(S[1,1]/n))  
tpos1 <- xbar[1] + sqrt(frac\*crit2\*(S[1,1]/n))  
cat("Max Breadth: (", tneg1, ",", tpos1, ") \n")  
# basibregmatic height  
tneg2 <- xbar[2] - sqrt(frac\*crit2\*(S[2,2]/n))  
tpos2 <- xbar[2] + sqrt(frac\*crit2\*(S[2,2]/n))  
cat("Basibregmatic Height: (", tneg2, ",", tpos2, ") \n")  
# basialveolar length  
tneg3 <- xbar[3] - sqrt(frac\*crit2\*(S[3,3]/n))  
tpos3 <- xbar[3] + sqrt(frac\*crit2\*(S[3,3]/n))  
cat("Basialveolar Length: (", tneg3, ",", tpos3, ") \n")  
# nasal height  
tneg4 <- xbar[4] - sqrt(frac\*crit2\*(S[4,4]/n))  
tpos4 <- xbar[4] + sqrt(frac\*crit2\*(S[4,4]/n))  
cat("Nasal Height: (", tneg4, ",", tpos4, ") \n")  
Bonferroni <- c(bpos1-bneg1, bpos2-bneg2, bpos3-bneg3, bpos4-bneg4)  
T2 <- c(tpos1-tneg1, tpos2-tneg2, tpos3-tneg3, tpos4-tneg4)  
cbind(Bonferroni, T2)  
# xbar  
xbarMat <- c(0.766, 0.508, 0.438, 0.161)  
xbar <- matrix(xbarMat, nrow = 4, ncol = 1, byrow = TRUE)  
# sample variance  
sMat <- c(0.856, 0.635, 0.173, 0.096,   
 0.635, 0.568, 0.128, 0.067,   
 0.173, 0.127, 0.171, 0.039,  
 0.096, 0.067, 0.039, 0.043)  
S <- matrix(sMat, nrow = 4, ncol = 4, byrow = TRUE)  
# sample size  
n <- 50  
p <- 4  
crit1 <- -qt(0.05/(2\*p), df = n-1)  
#petrol  
neg1 <- xbar[1] - crit1\*(S[1,1]/sqrt(n))  
pos1 <- xbar[1] + crit1\*(S[1,1]/sqrt(n))  
cat("Petroleum: (", neg1, ",", pos1, ") \n")  
# gas  
neg2 <- xbar[2] - crit1\*(S[2,2]/sqrt(n))  
pos2 <- xbar[2] + crit1\*(S[2,2]/sqrt(n))  
cat("Natural Gas: (", neg2, ",", pos2, ") \n")  
# coal  
neg3 <- xbar[3] - crit1\*(S[3,3]/sqrt(n))  
pos3 <- xbar[3] + crit1\*(S[3,3]/sqrt(n))  
cat("Coal: (", neg3, ",", pos3, ") \n")  
# nuclear  
neg4 <- xbar[4] - crit1\*(S[4,4]/sqrt(n))  
pos4 <- xbar[4] + crit1\*(S[4,4]/sqrt(n))  
cat("Nuclear: (", neg4, ",", pos4, ") \n")  
# total  
neg5 <- (xbar[1] + xbar[2] + xbar[3] + + xbar[4]) - crit1\*((S[1,1]+S[2,2]+S[3,3]+S[4,4])/sqrt(n))  
pos5 <- (xbar[1] + xbar[2] + xbar[3] + + xbar[4]) + crit1\*((S[1,1]+S[2,2]+S[3,3]+S[4,4])/sqrt(n))  
cat("Total: (", neg5, ",", pos5, ") \n")  
# petrol - gas  
neg6 <- (xbar[1] - xbar[2]) - crit1\*((S[1,1]-S[2,2])/sqrt(n))  
pos6 <- (xbar[1] - xbar[2]) + crit1\*((S[1,1]-S[2,2])/sqrt(n))  
cat("Petroleum - Gas: (", neg6, ",", pos6, ")")  
crit2 <- sqrt(qchisq(0.05, df = p, lower.tail = FALSE))  
# petrol  
neg1 <- xbar[1] - (crit2\*(S[1,1]/sqrt(n)))  
pos1 <- xbar[1] + (crit2\*(S[1,1]/sqrt(n)))  
cat("Petroleum: (", neg1, ",", pos1, ") \n")  
# gas  
neg2 <- xbar[2] - (crit2\*(S[2,2]/sqrt(n)))  
pos2 <- xbar[2] + (crit2\*(S[2,2]/sqrt(n)))  
cat("Natural Gas: (", neg2, ",", pos2, ") \n")  
# coal  
neg3 <- xbar[3] - (crit2\*(S[3,3]/sqrt(n)))  
pos3 <- xbar[3] + (crit2\*(S[3,3]/sqrt(n)))  
cat("Coal: (", neg3, ",", pos3, ") \n")  
# nuclear  
neg4 <- xbar[4] - (crit2\*(S[4,4]/sqrt(n)))  
pos4 <- xbar[4] + (crit2\*(S[4,4]/sqrt(n)))  
cat("Nuclear: (", neg4, ",", pos4, ") \n")  
# total  
neg5 <- (xbar[1] + xbar[2] + xbar[3] + + xbar[4]) - (crit2\*((S[1,1]+S[2,2]+S[3,3]+S[4,4])/sqrt(n)))  
pos5 <- (xbar[1] + xbar[2] + xbar[3] + + xbar[4]) + (crit2\*((S[1,1]+S[2,2]+S[3,3]+S[4,4])/sqrt(n)))  
cat("Total: (", neg5, ",", pos5, ") \n")  
# petrol - gas  
neg6 <- (xbar[1] - xbar[2]) - (crit2\*((S[1,1]-S[2,2])/sqrt(n)))  
pos6 <- (xbar[1] - xbar[2]) + (crit2\*((S[1,1]-S[2,2])/sqrt(n)))  
cat("Petroleum - Gas: (", neg6, ",", pos6, ")")  
cat("The Bonferroni crit region is", crit1, "\n")  
cat("The T2 crit region is", crit2)