STAT7017 Homework 3

Rui Qiu

2018-09-20

Question 1

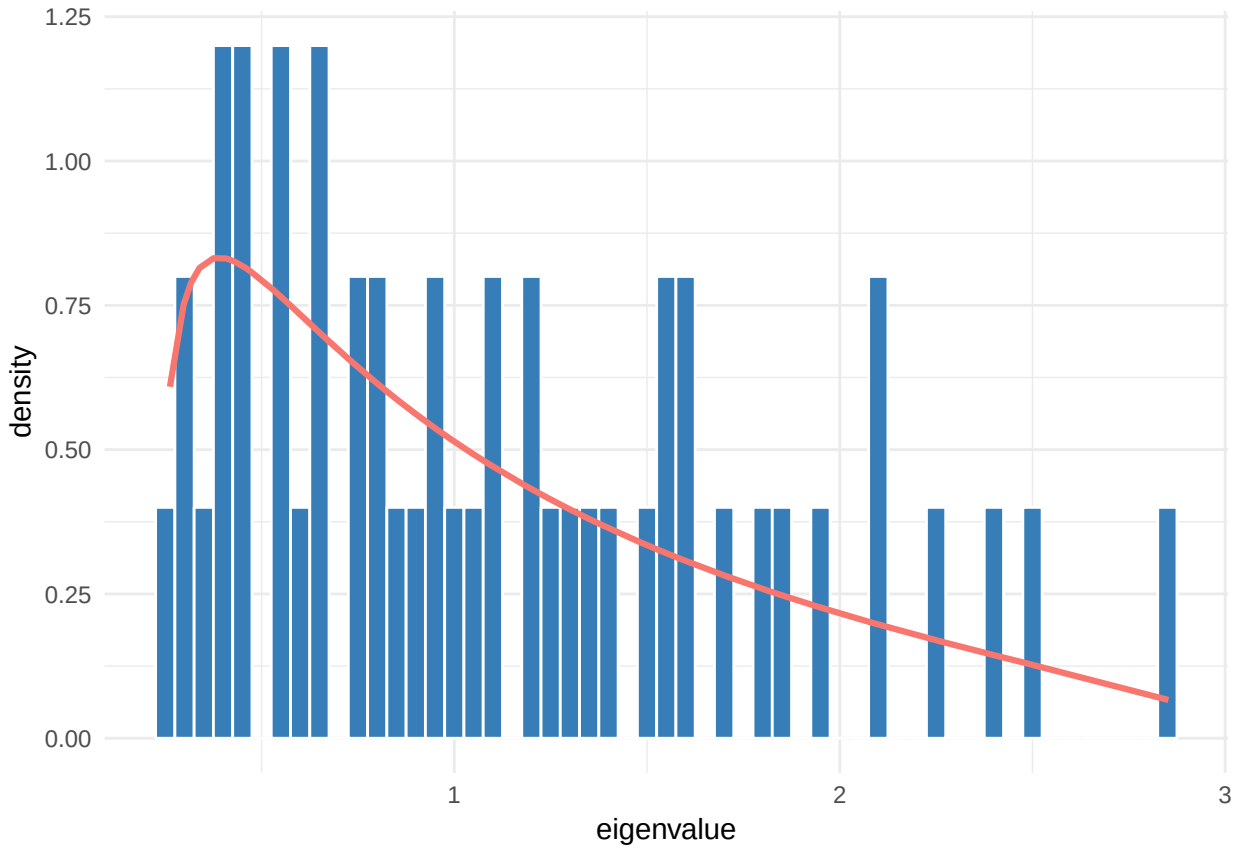
S_1 and S_2 are two sample covariance matrices for p dimensional observations of size n_1 and n_2. With $V_n = S_1 S_2^{-1}$, $y_{n_1} = p/n_1, y_{n_2} = p/n_2$, the parameter here I choose here is

$$p = 50, n_1 = 200, n_2 = 500.$$

```
set.seed(7017)
library(ggplot2)

p <- 50
n1 <- 200
n2 <- 500
X1 <- matrix(rnorm(p*n1),p,n1)
X2 <- matrix(rnorm(p*n2),p,n2)
S1 <- X1 %*% t(X1) / n1
S2 <- X2 %*% t(X2) / n2
V <- S1 %*% solve(S2)
e <- eigen(V)
eigenvalue <- e$values
dat <- as.data.frame(eigenvalue)

# LSD
y1 <- p/n1
y2 <- p/n2
h <- sqrt(y1+y2-y1*y2)
a <- ((1-h)/(1-y2))**2
b <- ((1+h)/(1-y2))**2

l <- function(x) {
    if (x < a || x > b) {
        return(0)
    } else {
        return((1-y2)*sqrt((b-x)*(x-a))/(2*pi*x*(y1+y2*x)))
    }
}
dat$lx <- sapply(dat$eigenvalue,l)
```

The histogram of simulation is plotted below. I also include a red curve indicating the density. (Guess it works better than histogram in our case.)

```
ggplot(dat, aes(x=eigenvalue,y=..density..)) +
    geom_histogram(fill='#377EB8',color='white',binwidth = 0.05) +
    geom_line(aes(x=eigenvalue, y=lx, color='#C83E45'), lwd=1.1) +
    guides(fill=FALSE, color=FALSE) +
    theme_minimal()
```



Question 2

One of the precondition of the conclusion in paper is that,

$$p, q_1, n - q, \longrightarrow \infty.$$

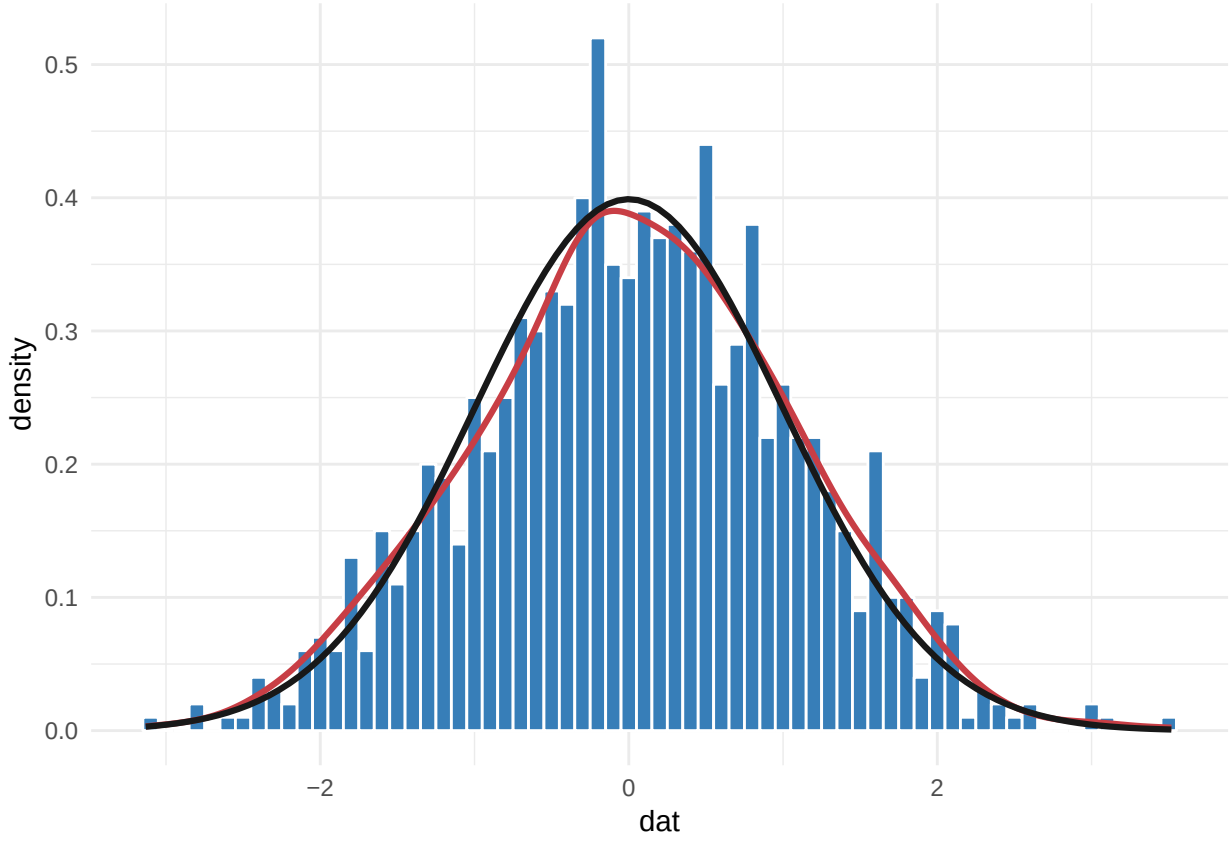
Spent a ton of time trialing the parameters since due to the limitation of computing power, we know they are "infinitely large" but we could only use some relatively small numbers to imitate.

```
# n >= p+q
p <- 100
q <- 1000
q1 <- 500
n.diff.q <- 500

yn1 <- p/q1
yn2 <- p/n.diff.q

hn <- sqrt(yn1+yn2-yn1*yn2)

an <- ((1-hn)/(1-yn2))**2
bn <- ((1+hn)/(1-yn2))**2
cn <- 0.5*(sqrt(1+yn2/yn1*bn)+sqrt(1+yn2/yn1*an))
dn <- 0.5*(sqrt(1+yn2/yn1*bn)-sqrt(1+yn2/yn1*an))

mf <- 0.5*log((cn**2-dn**2)*hn**2/(cn*hn-yn2*dn)**2)
```

```
nuf <- 2*log(cn**2/(cn**2-dn**2))
Ff <- (yn2-1)/yn2*log(cn) + (yn1-1)/yn1*log(cn-dn*hn) +
    (yn1+yn2)/(yn1*yn2)*log((cn*hn-dn*yn2)/hn)

Sigma <- diag(p)

dat <- c()
rep <- 1000
S1 <- rWishart(rep,n.diff.q,Sigma)
S2 <- rWishart(rep,q1,Sigma)
for (i in 1:rep) {
   FF <- n.diff.q/q1 * solve(S1[,,i]) %*% S2[,,i]
   Lambda <- solve(det(diag(p)+q1/n.diff.q*FF))
   Tn <- nuf**(-0.5)*(-log(Lambda[1,1])-p*Ff-mf)
   dat <- c(dat, Tn)
}
dat <- as.data.frame(dat)
```

Again, instead of letting the histogram stand there alone, I add a red curve indicating the density. While another black, familiar-shaped curve is our old friend stand normal density curve. Generally speaking, these two are matched. Hence we are confident enough to reconfirm that $T_n \sim N(0,1)$.

```
ggplot(dat,aes(x=dat)) +
    geom_histogram(aes(y=..density..),fill='#377EB8',color='white',binwidth = 0.1) +
    stat_density(geom="line",color='#C83E45',lwd=1.1) +
    stat_function(
        fun = dnorm,
        args = list(mean=0, sd=1),
        color = '#181818',
        lwd=1.1
    ) +
    theme_minimal()
```



Question 3

In the question, I basically want to replicate the table on page 3834 in reference [C] (Bai, Jiang, Yao and Zheng, 2009). That is to say, with two setups: one group with $y_1 = y_2 = 0.05$ and one group with $y_1 = 0.05, y_2 = 0.1$.

I also printed out the calculated $\mu_2, \sigma_2, \mu 1, \sigma_1$ from each set of (p, n_1, n_2). The number of simulations for each set of parameter is 5000. Moreover, for each calculation of α (size) and $1 - \beta$ (power), I implement the simulation study with the corrected likelihood reatio test and traditional likelihood ratio test (Box's M) with 1000 simulations each.

The detailed results are stored in a table in the end.

```
sim <- 5000
sim2 <- 1000
plist <- c(5,10,20,40,5,10,20,40)
n1list <- c(100,200,400,800,100,200,400,800)
n2list <- c(100,200,400,800,50,100,200,400)

table <- data.frame()
for (iter in 1:length(plist)) {
    p <- plist[iter]
    n1 <- n1list[iter]
    n2 <- n2list[iter]
    N1 <- n1-1
    N2 <- n2-1
    N <- n1+n2
    c1 <- N1/N
```

```
c2 <- N2/N
y1 <- p/N1
y2 <- p/N2

cat("p =", p, "n1 =", n1, "n2 =", n2, "\n")

dat <- c()
logs <- c()

# type1 <- vector(length=sim)
# type2 <- vector(length=sim)

for (i in 1:sim) {

    X1 <- matrix(rnorm(p*N1),p,N1)
    X2 <- matrix(rnorm(p*N2),p,N2)
    A1 <- X1 %*% t(X1) / n1
    A2 <- X2 %*% t(X2) / n2
    N <- N1 + N2
    S1 <- A1 / N1 # TODO: still not sure if denominator is Ng or (N1+N2)
    S2 <- A2 / N2 # TODO: same here

    logV1star <- (N1/2) *
        unlist(determinant(S1%*%solve(S2), logarithm = T))[[1]] -
        (N/2) *
        unlist(determinant(c1*S1%*%solve(S2)+c2*diag(p), logarithm = T))[[1]]
    Fab <- (y1+y2-y1*y2)/(y1*y2)*log((y1+y2)/(y1+y2-y1*y2))+
        y1*(1-y2)/(y2*(y1+y2))*log(1-y2)+y2*(1-y1)/(y1*(y1+y2))*log(1-y1)
    value <- -2/N*logV1star-p*Fab
    dat <- c(dat,value)

    logV1 <- (n1/2) * unlist(determinant(A1, logarithm = T))[[1]] +
        (n2/2) * unlist(determinant(A2, logarithm = T))[[1]] -
        (N/2) * unlist(determinant(n1/N*A1+n2/N*A2, logarithm = T))[[1]]
    # logV1 <- (N1/2) * unlist(determinant(A1, logarithm = T))[[1]] +
    #     (N2/2) * unlist(determinant(A2, logarithm = T))[[1]] -
    #     ((N1+N2)/2) * unlist(determinant(A1+A2, logarithm = T))[[1]]
    logs <- c(logs,logV1)
}

mu2 <- mean(dat)
sigma2 <- sd(dat)
cat("mu2 =", mu2, "sigma2 =", sigma2, "\n")

########################
#         CLRT         #
########################

TN <- sigma2^(-1)*(dat-mu2)

# type1 <- vector(length=sim2)
power <- vector(length=sim2)
for (j in 1:sim2) {
```

```
        # type1[j] <- t.test(TN,rnorm(sim,0,1))$p.value
        power[j] <- t.test(TN,rnorm(sim,0.5,1))$p.value
    }

    CLRT.size <- mean(TN>qnorm(0.95))
    # CLRT.t1e <- mean(type1<.05)
    CLRT.power <- mean(power<.05)

    cat("Type I error is", CLRT.size, "\n")
    cat("Power is", CLRT.power, "\n")

    ########################
    #         LRT          #
    ########################

    TN <- -2*logs

    # type1 <- vector(length=sim2)
    power <- vector(length=sim2)
    for (j in 1:sim2) {
        # type1[j] <- t.test(TN,rchisq(sim,0.5*p*(p+1)))$p.value
        power[j] <- t.test(TN,rchisq(sim,0.5*p*(p+1)+10))$p.value
    }

    LRT.size <- mean(TN>qchisq(0.95,0.5*p*(p+1)))
    # LRT.t1e <- mean(type1<.05)
    LRT.power <- mean(power<.05)

    cat("Type I error is", LRT.size, "\n")
    cat("Power is", LRT.power, "\n")

    # meanwhile by (4.8) and (4.9)

    mu1 <- 0.5*(log((y1+y2-y1*y2)/(y1+y2))-y1/(y1+y2)*log(1-y2)-
                    y2/(y1+y2)*log(1-y1)+6*y1**2*y2/(y1+y2)**2+6*y1*y2**2/(y1+y2)**2)
    sigma1 <- sqrt(-2*y2**2/(y1+y2)**2*log(1-y1)-2*y1**2/(y1+y2)**2*log(1-y2)-
        2*log((y1+y2)/(y1+y2-y1*y2)))
    cat("mu1 =", mu1, "sigma1 =", sigma1, "\n")

    print("--------------------------------")
    table <- rbind(table, c(p,n1,n2,CLRT.size,CLRT.power,LRT.size,LRT.power))
}
```

```
## p = 5 n1 = 100 n2 = 100
## mu2 = -0.03789426 sigma2 = 0.02818591
## Type I error is 0.0676
## Power is 1
## Type I error is 0.4938
## Power is 1
## mu1 = 0.08888169 sigma1 = 0.02591108
## [1] "--------------------------------"
## p = 10 n1 = 200 n2 = 200
## mu2 = -0.03692998 sigma2 = 0.02680084
## Type I error is 0.061
```

```
## Power is 1
## Type I error is 0.6164
## Power is 1
## mu1 = 0.08843246 sigma1 = 0.02577748
## [1] "--------------------------------"
## p = 20 n1 = 400 n2 = 400
## mu2 = -0.03727281 sigma2 = 0.02646104
## Type I error is 0.0564
## Power is 1
## Type I error is 0.6988
## Power is 1
## mu1 = 0.08820954 sigma1 = 0.02571119
## [1] "--------------------------------"
## p = 40 n1 = 800 n2 = 800
## mu2 = -0.03729025 sigma2 = 0.02601299
## Type I error is 0.0508
## Power is 1
## Type I error is 0.8008
## Power is 1
## mu1 = 0.0880985 sigma1 = 0.02567817
## [1] "--------------------------------"
## p = 5 n1 = 100 n2 = 50
## mu2 = 0.2203841 sigma2 = 0.09430989
## Type I error is 0.0618
## Power is 1
## Type I error is 0.5468
## Power is 1
## mu1 = 0.1193182 sigma1 = 0.03519985
## [1] "--------------------------------"
## p = 10 n1 = 200 n2 = 100
## mu2 = 0.4937692 sigma2 = 0.09548587
## Type I error is 0.0518
## Power is 1
## Type I error is 0.659
## Power is 1
## mu1 = 0.1185071 sigma1 = 0.03495024
## [1] "--------------------------------"
## p = 20 n1 = 400 n2 = 200
## mu2 = 1.037941 sigma2 = 0.09525702
## Type I error is 0.0498
## Power is 1
## Type I error is 0.772
## Power is 1
## mu1 = 0.1181057 sigma1 = 0.0348268
## [1] "--------------------------------"
## p = 40 n1 = 800 n2 = 400
## mu2 = 2.129806 sigma2 = 0.09558942
## Type I error is 0.052
## Power is 1
## Type I error is 0.876
## Power is 1
## mu1 = 0.1179061 sigma1 = 0.03476541
## [1] "--------------------------------"
```

```r
names(table) <- c("p","n1","n2","CLRT.size", "CLRT.power", "LRT.size", "LRT.power")
table
```

```
##    p  n1  n2 CLRT.size CLRT.power LRT.size LRT.power
## 1  5 100 100    0.0676          1   0.4938         1
## 2 10 200 200    0.0610          1   0.6164         1
## 3 20 400 400    0.0564          1   0.6988         1
## 4 40 800 800    0.0508          1   0.8008         1
## 5  5 100  50    0.0618          1   0.5468         1
## 6 10 200 100    0.0518          1   0.6590         1
## 7 20 400 200    0.0498          1   0.7720         1
## 8 40 800 400    0.0520          1   0.8760         1
```

As we can see, the power of two tests are generally the same, but the size of CLRT is stable around $\alpha = 0.05$ which is the significant level. On the contrary, the size of LRT is relatively large in our case.