STAT34800 HW6

Sarah Adilijiang

Question 1

(i) simulate data from model

```r
# function for simulating data from model
n=10       # number of individuals i=1,...,n
m=1000     # number of genes j=1,..,m
sigma_j=1  # sd for Dij

simulate_data = function(pi_0, sigma_b){
    beta = rep(0,m)
    D = matrix(0,n,m)
    for (j in 1:m) {
        pi = rbinom(m, 1, 1-pi_0)    # pi=1 with prob=1-pi_0
        if(pi==0){beta[j]=0}
        if(pi==1){beta[j]=rnorm(1,0,sigma_b)}
        D[,j] = rnorm(n,beta[j],sigma_j)
    }
    return(list(D=D, beta=beta))
}
```

(ii) compute p-values

Since $D_{ij}|\beta,\sigma \sim N(\beta_j,\sigma_j^2)$ where $\sigma_j = 1$ is known for all j, so by CLT, under null hypothesis $H_j : \beta_j = 0$, we have the test statistic:

$$T_j = \frac{\bar{D}_j - \beta_j^0}{\sigma_j/\sqrt{n}} = \frac{\bar{D}_j - 0}{1/\sqrt{10}} = \sqrt{10}\ \bar{D}_j \ \sim\ N(0,1)$$

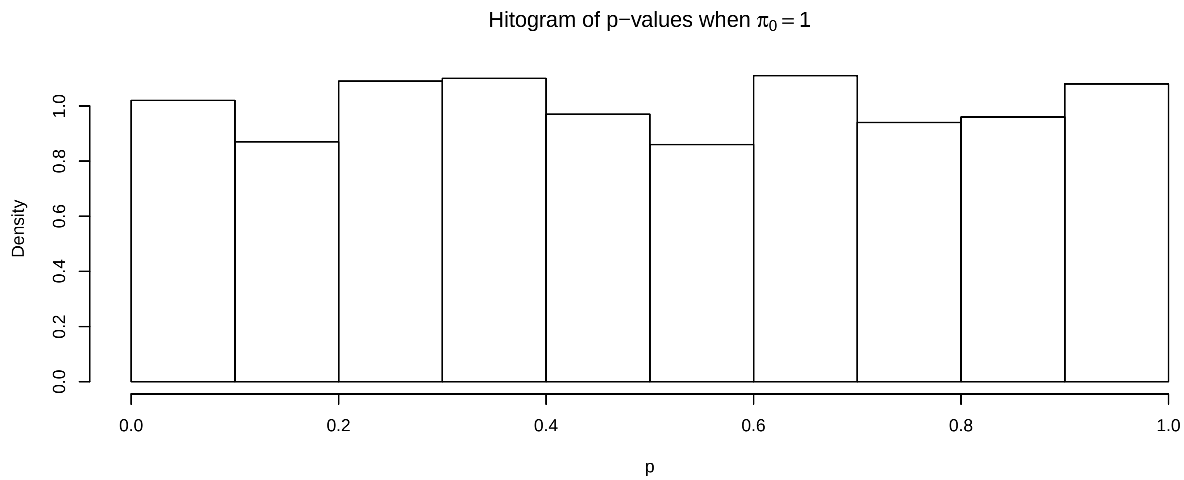
Thus the p-value p_j is:

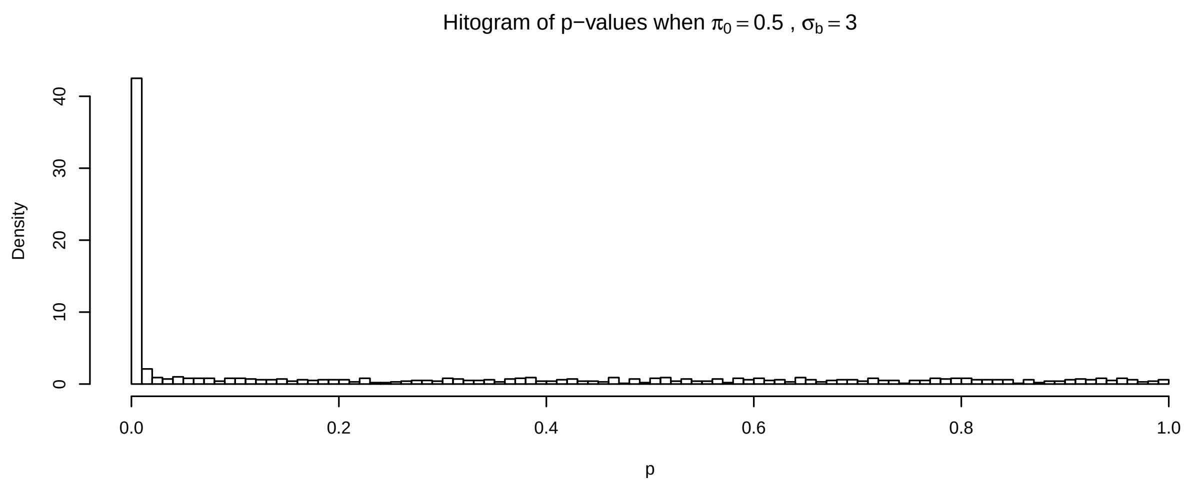
$$p_j = 2 \times Pr(Z > |T_{obsj}|) = 2 \times Pr(Z > \sqrt{10}\ |\bar{D}_j|)$$

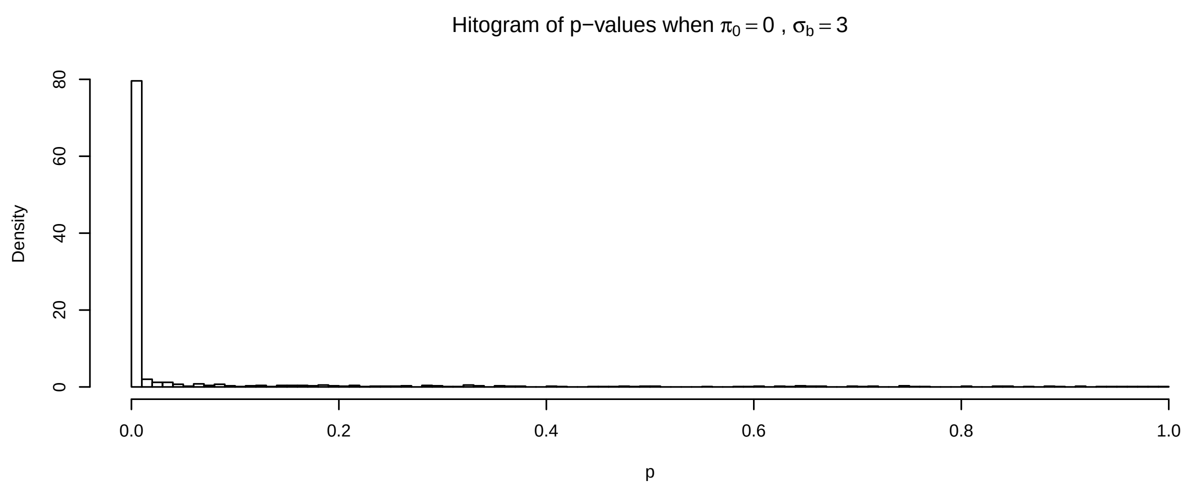
```r
# function for computing p-values
compute_p = function(D){
    z = sqrt(10) * abs(colMeans(D))
    p = 2 * pnorm(z,0,1,lower.tail=FALSE)
    return(p)
}
```

```r
# try different settings
set.seed(123)
par(mfrow=c(3,1))

data = simulate_data(pi_0=1)
p = compute_p(data$D)
hist(p, prob=TRUE, nclass=10, main=expression(paste("Hitogram of p-values when ", pi[0]==1)))

data = simulate_data(pi_0=0.5, sigma_b=3)
p = compute_p(data$D)
hist(p, prob=TRUE, nclass=100, main=expression(paste("Hitogram of p-values when ",pi[0]==0.5," , ",sigma
```

```
data = simulate_data(pi_0=0, sigma_b=3)
p = compute_p(data$D)
hist(p, prob=TRUE, nclass=100, main=expression(paste("Hitogram of p-values when ",pi[0]==0," , ",sigma[
```







Comments:

When $\pi_0 = 1$, true effect $\beta_j = 0$ for all j, so the p-value is uniformly distributed on (0,1).

When $\pi_0 = 0.5, \sigma_b = 3$, true effect $\beta_j \sim 0.5\delta_0 + 0.5N(0, 3^2)$ for all j, we can see that most of the p-values are distributed within the range of (0,0.01).

When $\pi_0 = 0, \sigma_b = 3$, true effect $\beta_j \sim N(0, 3^2)$ for all j, now much more p-values are are distributed within the range of (0,0.01) comparing with the above setting.

(iii) Benjamini-Hochberg rule

```
# function for applying Benjamini-Hochberg rule
BH_rule = function(p, alpha){
    index = order(p)
    order_p = p[index]  # increasing
    gamma = rep(0,m)    # m=1000
    if( min(order_p-(1:m)*alpha/m) > 0){return(gamma)}
    else{
        k = max(which(order_p-(1:m)*alpha/m <= 0))
        gamma[index[1:k]] = 1       # gamma==1: reject null
        return(gamma)
    }
}
```

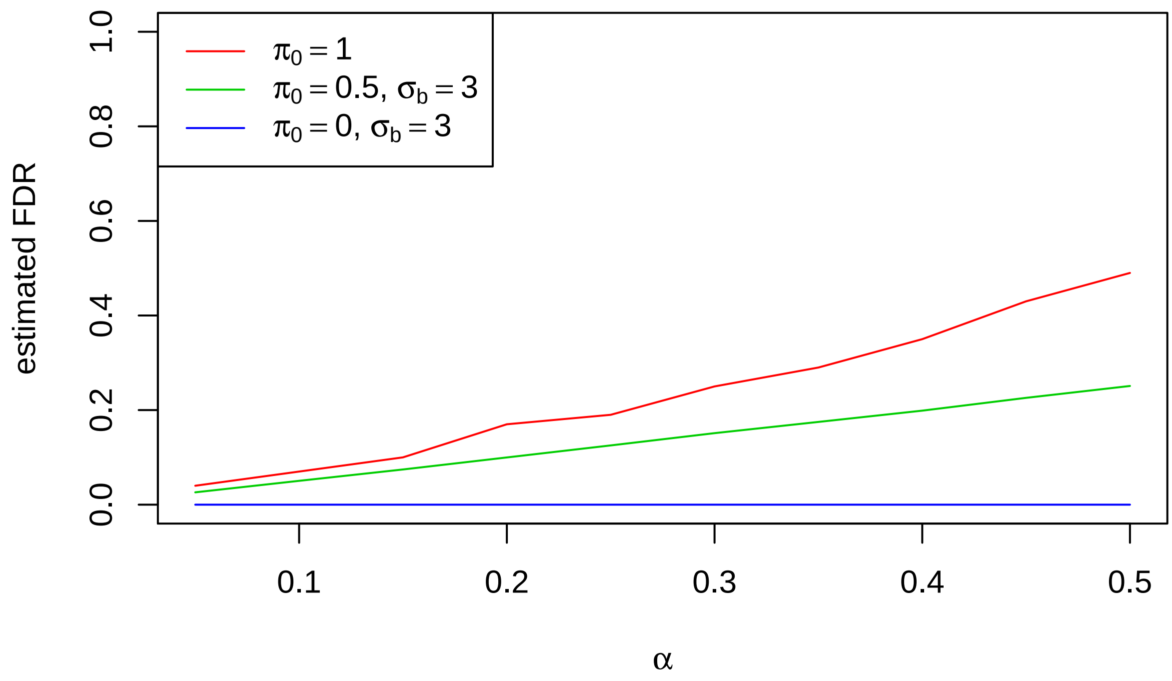
(iv) compute empirical FDR

```
# function for computing empirical FDR
FDR_empirical = function(beta, gamma){
    V = sum(beta==0 & gamma==1)
    R = sum(gamma==1)
    if(R==0){return(0)}
    else{return(V/R)}
}
```

(v) estimate FDR via esimating E(V/R)

```
set.seed(123)
alpha = seq(0.05,0.5,by=0.05)

est_FDR = function(pi_0, sigma_b){
    FDR = matrix(NA,100,length(alpha))
    for (i in 1:100) {
        data = simulate_data(pi_0, sigma_b)
        p = compute_p(data$D)
        for (j in 1:length(alpha)) {
            gamma = BH_rule(p, alpha[j])
            FDR[i,j] = FDR_empirical(data$beta, gamma)
        }
    }
    return(colMeans(FDR))
}

fdr1 = est_FDR(pi_0=1, sigma_b=NA)
```

```
fdr2 = est_FDR(pi_0=0.5, sigma_b=3)
fdr3 = est_FDR(pi_0=0, sigma_b=3)
plot(alpha, fdr1, col=2, type="l", ylim=c(0,1), xlab=expression(alpha), ylab="estimated FDR")
lines(alpha,fdr2, col=3, lty=1)
lines(alpha,fdr3, col=4, lty=1)
legend("topleft", legend=c(expression(pi[0]==1), expression(paste(pi[0]==0.5,", ",sigma[b]==3)), expres
```



Comments:

By definition, $E(V/R|R=0)=0$, so:

$$FDR = E(V/R) = E(V/R|R>0)\ Pr(R>0)$$

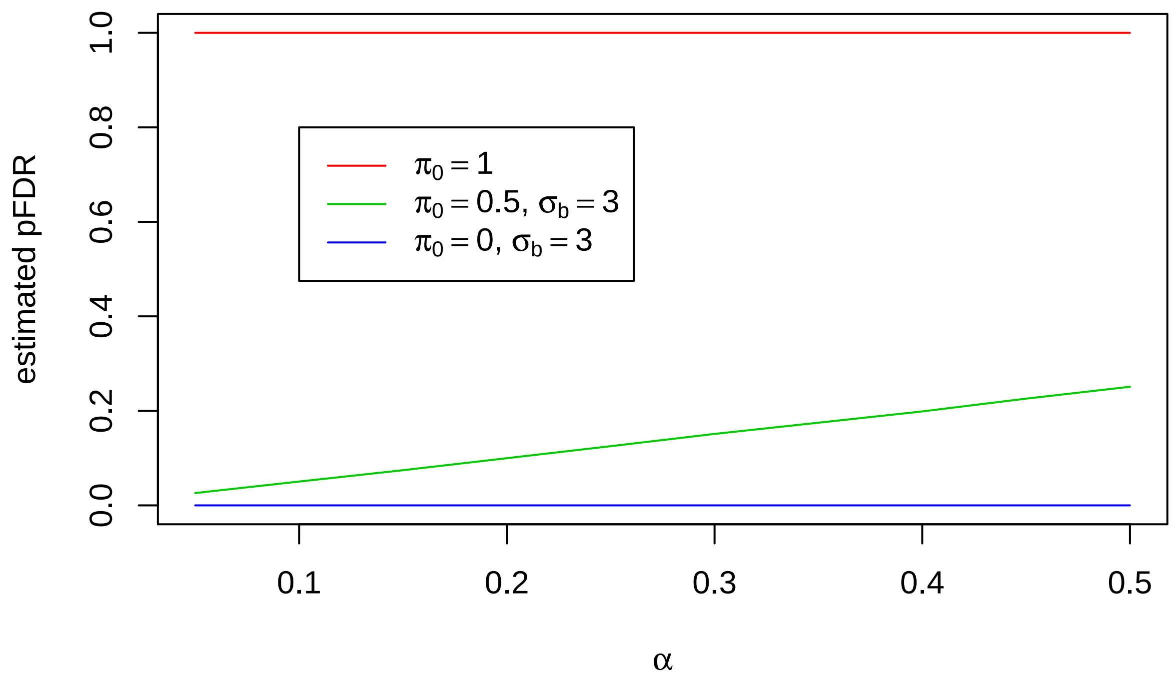
When $\pi_0 = 1$, true effect $\beta_j = 0$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are true, thus $V = R$ and $E(V/R|R>0) = 1$. Therefore, in this case, the estimated $FDR = Pr(R>0)$. We can see that as α increases, the value of FDR increases.

When $\pi_0 = 0, \sigma_b = 3$, true effect $\beta_j \sim N(0, 3^2)$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are false, thus $V = 0$ and $E(V/R|R>0) = 0$. Therefore, in this case, all the estimated $FDR = 0$.

When $\pi_0 = 0.5, \sigma_b = 3$, true effect $\beta_j \sim 0.5\delta_0 + 0.5N(0, 3^2)$ for all j. We can see that as α increases, the value of FDR increases, but their values are smaller than the case when $\pi_0 = 1$ and larger than the case when $\pi_0 = 0$.

(vi) estimate pFDR via esimating E(V/R|R>0)

```r
set.seed(123)
alpha = seq(0.05,0.5,by=0.05)

est_pFDR = function(pi_0, sigma_b){
    pFDR = matrix(NA,100,length(alpha))
    for (i in 1:100) {
        data = simulate_data(pi_0, sigma_b)
        p = compute_p(data$D)
        for (j in 1:length(alpha)) {
            gamma = BH_rule(p, alpha[j])
            R = sum(gamma==1)
            if(R>0){
                pFDR[i,j] = FDR_empirical(data$beta, gamma)
            }
        }
    }
    return(colMeans(pFDR,na.rm=TRUE))
}

fdr1 = est_pFDR(pi_0=1, sigma_b=NA)
fdr2 = est_pFDR(pi_0=0.5, sigma_b=3)
fdr3 = est_pFDR(pi_0=0, sigma_b=3)
plot(alpha, fdr1, col=2, type="l", ylim=c(0,1), xlab=expression(alpha), ylab="estimated pFDR")
lines(alpha,fdr2, col=3, lty=1)
lines(alpha,fdr3, col=4, lty=1)
legend(0.1,0.8, legend=c(expression(pi[0]==1), expression(paste(pi[0]==0.5,", ",sigma[b]==3)), expressi
```



Comments:

$$pFDR = E(V/R|R > 0)$$

When $\pi_0 = 1$, true effect $\beta_j = 0$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are true, thus $V = R$ and $E(V/R|R > 0) = 1$. Therefore, in this case, all the estimated $pFDR = 1$.

When $\pi_0 = 0, \sigma_b = 3$, true effect $\beta_j \sim N(0, 3^2)$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are false, thus $V = 0$ and $E(V/R|R > 0) = 0$. Therefore, in this case, all the estimated $pFDR = 0$.

When $\pi_0 = 0.5, \sigma_b = 3$, true effect $\beta_j \sim 0.5\delta_0 + 0.5N(0, 3^2)$ for all j. We can see that as α increases, the value of pFDR increases, and their values are between 0 and 1.

Question 3

(i)

Since $D_{ij}|\beta, \sigma \sim N(\beta_j, \sigma_j^2)$ where $\sigma_j = 1$ is known for all j, so $\bar{D}_j \sim N(\beta_j, \sigma_j^2/n)$.

Thus:

$$p(D|\beta) = \prod_{i=1}^{n}\prod_{j=1}^{m} \frac{1}{\sqrt{2\pi\sigma_j^2}}\ \exp\{-\frac{(D_{ij}-\beta_j)^2}{2\sigma_j^2}\}\ =\ \prod_{j=1}^{m}\left(\frac{1}{\sqrt{2\pi\sigma_j^2}}\right)^n\ \exp\{-\frac{\sum_{i=1}^{n}(D_{ij}-\beta_j)^2}{2\sigma_j^2}\}$$

$$= \prod_{j=1}^{m} \left(\frac{1}{\sqrt{2\pi\sigma_j^2}} \right)^n \exp\{-\frac{\sum_{i=1}^{n} D_{ij}^2 - 2n\bar{D}_j\beta_j + n\beta_j^2}{2\sigma_j^2}\} = \prod_{j=1}^{m} \left(\frac{1}{\sqrt{2\pi\sigma_j^2}} \right)^n \exp\{-\frac{\sum_{i=1}^{n} D_{ij}^2/n - 2\bar{D}_j\beta_j + \beta_j^2}{2\sigma_j^2/n}\}$$

$$= \prod_{j=1}^{m} \left(\frac{1}{\sqrt{2\pi\sigma_j^2}} \right)^n \exp\{\frac{\bar{D}_j^2 - \sum_{i=1}^{n} D_{ij}^2/n}{2\sigma_j^2/n}\} \exp\{-\frac{(\bar{D}_j - \beta_j)^2}{2\sigma_j^2/n}\}$$

and:

$$p(\bar{D}|\beta) = \prod_{j=1}^{m} \frac{1}{\sqrt{2\pi\sigma_j^2/n}} \exp\{-\frac{(\bar{D}_j - \beta_j)^2}{2\sigma_j^2/n}\}$$

Therefore, we have that: $p(D|\beta) \propto p(\bar{D}|\beta)$, where the constant of proportionality does not depend on β.

(ii) log-likelihood

Since $\bar{D}_j \sim N(\beta_j, \sigma_j^2/n)$ where $\sigma_j = 1$ is known for all j, and the true effects β_j are independent and identically distributed, with $\beta_j \sim \pi_0\delta_0 + (1-\pi_0)N(0, \sigma_b^2)$, thus we have:

$$\bar{D}_j|\pi_0, \sigma_b \sim \pi_0 N(0, \sigma_j^2/n) + (1-\pi_0)N(0, \sigma_j^2/n + \sigma_b^2)$$

Therefore:

$$l(\pi_0, \sigma_b) = \log(p(\bar{D}|\pi_0, \sigma_b)) = \log(\prod_{j=1}^{m} p(\bar{D}_j|\pi_0, \sigma_b)) = \sum_{j=1}^{m} \log(p(\bar{D}_j|\pi_0, \sigma_b))$$

$$= \sum_{j=1}^{m} \log \left(\frac{\pi_0}{\sqrt{2\pi\sigma_j^2/n}} \exp\{-\frac{\bar{D}_j^2}{2\sigma_j^2/n}\} + \frac{1-\pi_0}{\sqrt{2\pi(\sigma_j^2/n + \sigma_b^2)}} \exp\{-\frac{\bar{D}_j^2}{2(\sigma_j^2/n + \sigma_b^2)}\} \right)$$

(iii) maximize log-likelihood

```
# function for computing log-likelihood
minus_loglik = function(theta, D_mean){
    pi_0 = exp(theta[1])/(1+exp(theta[1]))   # theta1 = log(pi_0/(1-pi_0))
    sigma_b = exp(theta[2])                  # theta2 = log(sigma_b)
    - sum(log( pi_0 * dnorm(D_mean,0,sqrt(1/n)) +
            (1-pi_0) * dnorm(D_mean,0,sqrt(1/n+sigma_b^2)) ))
}

# function for computing MLEs using R function optim()
MLEs = function(D_mean){
    MLEs = optim(par=c(0,0), minus_loglik, D_mean=D_mean)
    theta1 = MLEs$par[1]
    theta2 = MLEs$par[2]
    pi_0_hat = exp(theta1)/(1+exp(theta1))
    sigma_b_hat = exp(theta2)
    return(list(pi_0.hat=pi_0_hat, sigma_b.hat=sigma_b_hat))
}
```

```
# test estimation results
set.seed(123)
test_MLEs = function(pi_0, sigma_b){
```

```r
    pi_0_hat = rep(NA,10)
    sigma_b_hat = rep(NA,10)
    for (i in 1:10) {
        data = simulate_data(pi_0, sigma_b)
        D_mean = colMeans(data$D)
        estimates = MLEs(D_mean)
        pi_0_hat[i] = estimates$pi_0.hat
        sigma_b_hat[i] = estimates$sigma_b.hat
    }
    results = list(pi_0.hat=pi_0_hat, sigma_b.hat=sigma_b_hat)
    print(results)
}

test_MLEs(pi_0=1)
```

```
## $pi_0.hat
##  [1] 0.9999906933 0.9999998514 0.0002743292 0.9999906933 0.9999906933
##  [6] 0.8852612823 0.9999906933 0.0022612972 0.9350493750 0.9999998514
##
## $sigma_b.hat
##  [1] 9.890089e-07 6.235739e-09 6.232886e-02 9.890089e-07 9.890089e-07
##  [6] 2.192216e-01 9.890089e-07 3.244699e-02 4.577741e-02 6.235739e-09
```

```r
test_MLEs(pi_0=0.5, sigma_b=3)
```

```
## $pi_0.hat
##  [1] 0.5060753 0.4937525 0.4757541 0.4880177 0.5122689 0.5081414 0.4965769
##  [8] 0.5139837 0.4669746 0.5133851
##
## $sigma_b.hat
##  [1] 3.161660 3.165853 2.867964 3.010504 3.113176 2.942425 2.979895
##  [8] 2.979675 3.100713 3.043975
```

```r
test_MLEs(pi_0=0, sigma_b=3)
```

```
## $pi_0.hat
##  [1] 5.625909e-05 1.231452e-06 8.933304e-08 1.904119e-02 1.180050e-07
##  [6] 8.438980e-07 5.749481e-08 9.244348e-03 9.834911e-07 4.354320e-03
##
## $sigma_b.hat
##  [1] 3.091793 3.029110 3.040785 3.045447 3.060135 3.020971 3.008804
##  [8] 3.074572 2.990196 3.015738
```

Comments:

We can see that the optimization and MLE estimation works roughly well.

(iv) posterior distribution

Prior distribution of β_j is:

$$\beta_j|\pi_0, \sigma_b \ \sim \ \pi_0\delta_0 + (1-\pi_0)N(0, \sigma_b^2)$$

From question (i), we have: $p(D|\beta) \propto p(\bar{D}|\beta)$, where the constant of proportionality does not depend on β. So we can treat $\bar{D}$ as our data instead of D and use the distribution of $\bar{D}_j$ as the likelihood:

$$\bar{D}_j|\beta_j \ \sim \ N(\beta_j, \sigma_j^2/n)$$

where $\sigma_j = 1$ is known for all j.

Also, from question (ii), we have derived the distibution:

$$\bar{D}_j|\pi_0, \sigma_b \ \sim \ \pi_0 N(0, \sigma_j^2/n) + (1-\pi_0) N(0, \sigma_j^2/n + \sigma_b^2)$$

Now we denote the density of normal distribution at $\bar{D}_j$ is:

$$f(\bar{D}_j; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \ \exp\{-\frac{(\bar{D}_j - \mu)^2}{2\sigma^2}\}$$

Hence the posterior distribution:

$$p(\beta_j = 0|D, \pi_0, \sigma_b) = \frac{p(\bar{D}_j|\beta_j = 0) \ p(\beta_j = 0|\pi_0, \sigma_b)}{p(\bar{D}_j|\pi_0, \sigma_b)} = \frac{\pi_0 \ f(\bar{D}_j; 0, \sigma_j^2/n)}{\pi_0 \ f(\bar{D}_j; 0, \sigma_j^2/n) + (1-\pi_0) \ f(\bar{D}_j; 0, \sigma_j^2/n + \sigma_b^2)}$$

and:

$$p(\beta_j|D, \pi_0, \sigma_b, \beta_j \neq 0) \ \propto \ p(\bar{D}_j|\beta_j) \, p(\beta_j|\pi_0, \sigma_b, \beta_j \neq 0) \ \propto \exp\{-\frac{(\bar{D}_j - \beta_j)^2}{2\sigma_j^2/n}\} \ \exp\{-\frac{\beta_j^2}{2\sigma_b^2}\} \ \propto \ \exp\{-\frac{\left(\beta_j - \frac{\sigma_b^2 \bar{D}_j}{\sigma_b^2 + \sigma_j^2/n}\right)^2}{2\frac{\sigma_b^2 \sigma_j^2/n}{\sigma_b^2 + \sigma_j^2/n}}\}$$

$$\Rightarrow \qquad \beta_j|D, \pi_0, \sigma_b, \beta_j \neq 0 \ \sim \ N(\frac{\sigma_b^2 \bar{D}_j}{\sigma_b^2 + \sigma_j^2/n}, \frac{\sigma_b^2 \sigma_j^2/n}{\sigma_b^2 + \sigma_j^2/n})$$

Therefore, the posterior distribution of β_j is:

$$\beta_j|D, \pi_0, \sigma_b \ \sim \ w_0 \delta_0 + (1-w_0) \ N(\frac{\sigma_b^2 \bar{D}_j}{\sigma_b^2 + \sigma_j^2/n}, \frac{\sigma_b^2 \sigma_j^2/n}{\sigma_b^2 + \sigma_j^2/n})$$

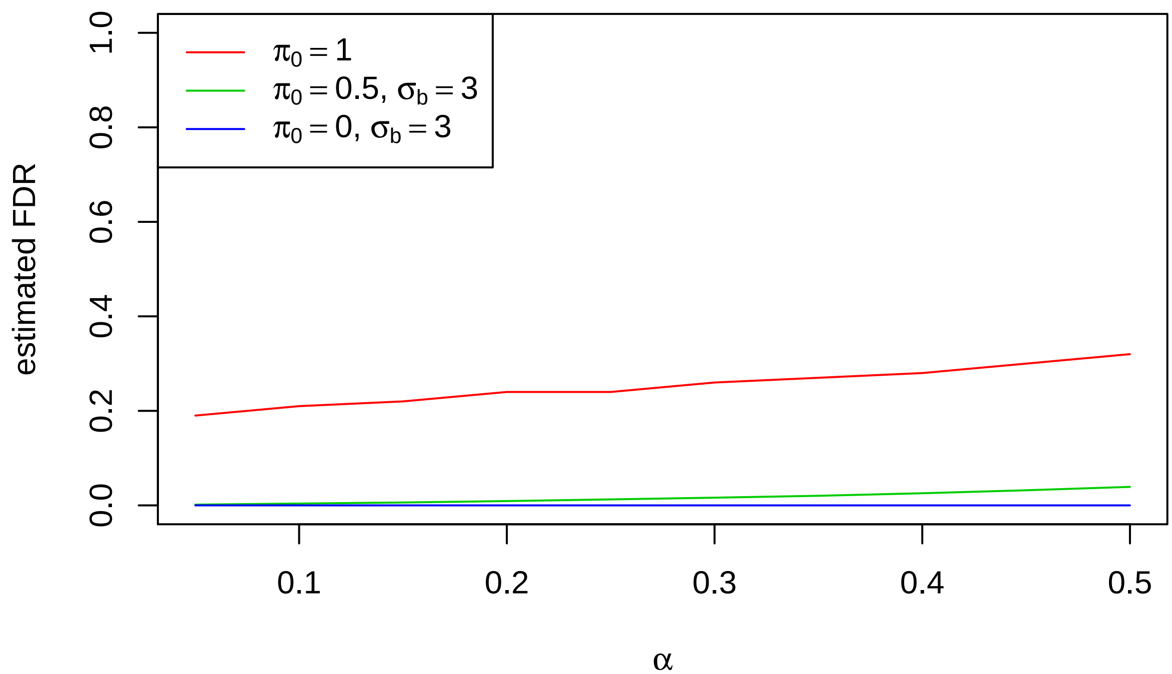
where

$$w_0 \ = \ p(\beta_j = 0|D, \pi_0, \sigma_b) \ = \ \frac{\pi_0 \ f(\bar{D}_j; 0, \sigma_j^2/n)}{\pi_0 \ f(\bar{D}_j; 0, \sigma_j^2/n) + (1-\pi_0) \ f(\bar{D}_j; 0, \sigma_j^2/n + \sigma_b^2)}$$

(v) control FDR via Empirical Bayesian approach

```
# function for computing posterior probability: w0 = p(beta_j=0|D,pi_0,sigma_b)
posterior_w0 = function(D_mean, pi_0, sigma_b){
    w0 =  pi_0 * dnorm(D_mean,0,sqrt(1/n)) /
        ( pi_0 * dnorm(D_mean,0,sqrt(1/n)) +
         (1-pi_0) * dnorm(D_mean,0,sqrt(1/n+sigma_b^2)) )
    return(w0)
}


# function for applying Empirical Bayesian rejection rule
EB_rule = function(w0, alpha){
    gamma = ifelse(w0<alpha,1,0)  # gamma==1: reject null
    return(gamma)
}
```

```r
# below use the same functions built in question 1:
# simulate_data(pi_0, sigma_b)
# FDR_empirical(beta, gamma)


# estimate FDR via esimating E(V/R)
set.seed(123)
alpha = seq(0.05,0.5,by=0.05)

est_FDR = function(pi_0, sigma_b){
    FDR = matrix(NA,100,length(alpha))
    for (i in 1:100) {
        data = simulate_data(pi_0, sigma_b)
        D_mean = colMeans(data$D)
        estimates = MLEs(D_mean)
        pi_0_hat = estimates$pi_0.hat
        sigma_b_hat = estimates$sigma_b.hat
        w0 = posterior_w0(D_mean, pi_0_hat, sigma_b_hat)
        for (j in 1:length(alpha)) {
            gamma = EB_rule(w0, alpha[j])
            FDR[i,j] = FDR_empirical(data$beta, gamma)
        }
    }
    return(colMeans(FDR))
}

fdr1 = est_FDR(pi_0=1, sigma_b=3)
fdr2 = est_FDR(pi_0=0.5, sigma_b=3)
fdr3 = est_FDR(pi_0=0, sigma_b=3)
plot(alpha,  fdr1, col=2, type="l", ylim=c(0,1), xlab=expression(alpha), ylab="estimated FDR")
lines(alpha, fdr2, col=3, lty=1)
lines(alpha, fdr3, col=4, lty=1)
legend("topleft", legend=c(expression(pi[0]==1), expression(paste(pi[0]==0.5,", ",sigma[b]==3)), expres
```



Comments:

By definition, $E(V/R|R=0)=0$, so:

$$FDR = E(V/R) = E(V/R|R>0)\ Pr(R>0)$$

When $\pi_0 = 1$, true effect $\beta_j = 0$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are true, thus $V = R$ and $E(V/R|R>0) = 1$. Therefore, in this case, the estimated $FDR = Pr(R>0)$. However, if using the true value of π_0, posterior probability $w_0 = p(\beta_j = 0|D, \pi_0, \sigma_b) = 1$, so all the null hypothesis will be rejected, thus $R = 0$ and all the $FDR = 0$. But here we are using the estimated $\hat{\pi}_0$, so $w_0 = p(\beta_j = 0|D, \hat{\pi}_0, \hat{\sigma}_b)$ will not be always equal to 1. As a result, we see that the estimated FDR are not equal to 0. And as α increases, the value of FDR increases.

When $\pi_0 = 0, \sigma_b = 3$, true effect $\beta_j \sim N(0, 3^2)$ for all j, so all the null hypothesis $H_j : \beta_j = 0$ are false, thus $V = 0$ and $E(V/R|R>0) = 0$. Therefore, in this case, all the estimated $FDR = 0$, no matter whether we are using the true value of π_0 or not.

When $\pi_0 = 0.5, \sigma_b = 3$, true effect $\beta_j \sim 0.5\delta_0 + 0.5N(0, 3^2)$ for all j. We can see that as α increases, the value of FDR increases, but their values are smaller than the case when $\pi_0 = 1$ and larger than the case when $\pi_0 = 0$.

Comparing with the previous method using Benjamini-Hochberg rule, here the Empirical Bayesian method provides much better results for the case $\pi_0 = 0.5$. EB method also works slight better for the case $\pi_0 = 1$ when the α value are larger, but worse when α value are smaller. However, it provides the same results for the case $\pi_0 = 0$.