hw3

Sina Sanei

November 13, 2018

1)

a)

Let $X = (A_1, ..., A_m)$ be the (unobserved) lifetimes associated with room A, and $S = 254$ the number of litebulbs in the second experiment that are still on at time $\tau = 15$. Thus, the total observed data combined is:

$$Y = (B_1, ..., B_n, E_1, ..., E_m)$$

where $E_i = 1$ if the bulb is still burning, and $E_i = 0$ if the light is out in room A. Hence $S = \sum_{i=1}^{m} E_i = 254$. given $A_i, B_i \overset{iid}{\sim} exp(\theta)$ the complete-data log-likelihood is :

$$l_c(\theta; Y, X) = log(\theta^{-1}e^{-\theta^{-1}B_1}...\theta^{-1}e^{-\theta^{-1}B_n})log(\theta^{-1}e^{-\theta^{-1}A_1}...\theta^{-1}e^{-\theta^{-1}A_m})$$

$$= -n(log\theta + \bar{B}/\theta) - \sum_{i=1}^{m}(log\theta + A_i/\theta)$$

now we should look at conditional expectations of unobserved data , given observed data :

$$E[A_i|Y] = E[A_i|E_i] = \begin{cases} \tau + \theta & \text{if } E_i = 1 \\ \theta - \frac{\tau e^{-\tau/\theta}}{1-e^{-\tau/\theta}} & \text{if } E_i = 0 \end{cases}$$

first equality in above follows from independence assumption and the jth step consist of replacing A_i in log-likelihood by its expected value from above equations, using the current numerical parameter value $\theta^{(j-1)}$. The result is:

$$l^{(j)}(\theta) = -(n+m)log\theta - 1/\theta[n\bar{B} + S(\tau + \theta^{(j-1)}) + (m-S)(\theta^{(j-1)} - \tau p^{(j-1)})] \tag{1}$$

where : $p^{(j)} = \frac{e^{-\tau/\theta^j}}{1-e^{-\tau/\theta^j}}$ The jth M-step maximizes (1), yielding :

$$\theta^{(j)} \equiv \frac{n\bar{B} + S(\tau + \theta^{(j-1)}) + (m-S)(\theta^{(j-1)} - \tau p^{(j-1)})}{n+m} \tag{2}$$

Thus, we can simply iterate Equation (2), starting with an arbitrary positive $\theta^{(0)}$,until convergence.

```
lifes_B = c(6.223391,0.3739535,39.60146,22.44155,75.21525,0.4719523,
            20.49098,6.297307,34.58661,55.29653,10.50263,16.05646,
            11.3004,9.472861,72.04453,33.95477,13.58237,14.3674,
            24.84866,22.79063,49.88125,44.27289,17.09678,162.3117,6.630124)
```

```
m=400
n=25
S=254 # number of survived lamps in A up to tau=15
tau= 15
theta_init= 200#mean(lifes_B)#median(lifes_B)mean(lifes_B)
theta_current= theta_init
```

```r
theta_new=0
p=function(theta){
  exp(-tau/theta)/(1-exp(-tau/theta))
}
max_iter = 100000
iter=0
for (i in 1:max_iter){
  theta_new = (n*mean(lifes_B)+S*(tau+theta_current)+(m-S)*(theta_current-tau*(p(theta_current))))/(n+m
  if (abs(theta_new-theta_current)<0.000001){
    cat("converged:", "theta =", theta_new, "iter_num =", iter)
    break
  }else {theta_current = theta_new
  iter=iter+1}

  }
```

```
## converged: theta = 32.69994 iter_num = 36
```

b)

nonparametric bootstrap pseudocode to obtain an estimated covariance for EM :
1. Calculate $\hat{\theta}_{(EM)}$ using EM approach applied to observed data. Let j = 1 and set $\hat{\theta}_j = \hat{\theta}_{(EM)}$.

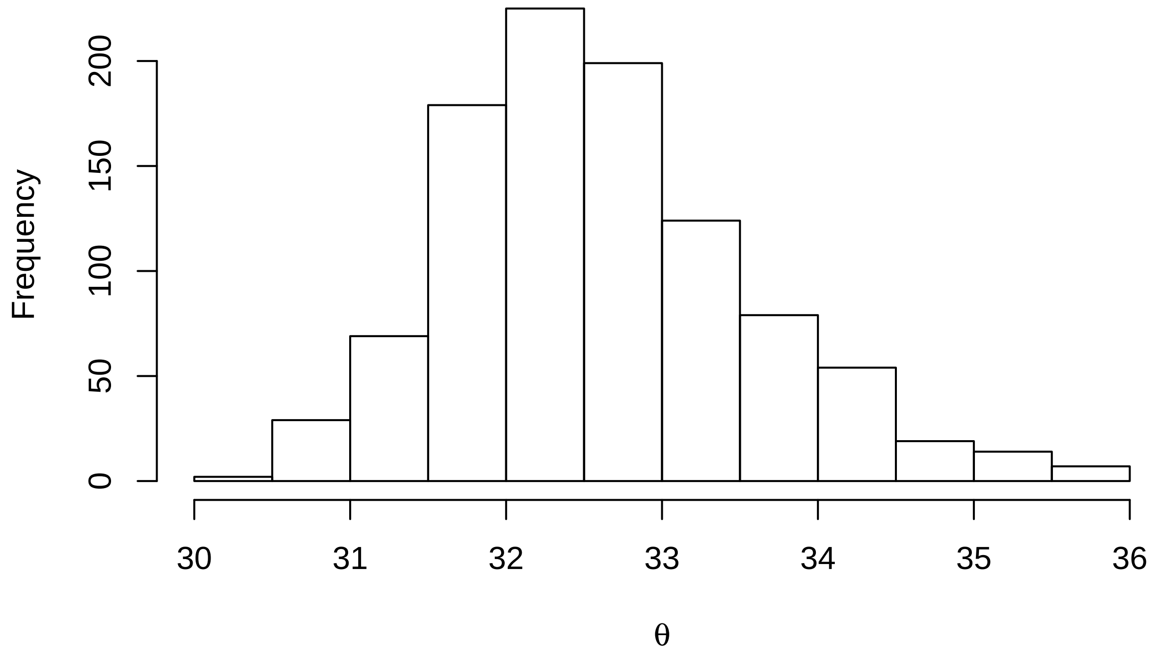
2. Increment j. Sample pseudo-data $(B_1^*, ..., B_n^*, E_1^*, ..., E_m^*)$ completely at random from $(B_1, ..., B_n, E_1, ..., E_m)$ with replacement.

3. Calculate $\hat{\theta}_j$ by applying the same EM approach to the pseudo-data $(B_1^*, ..., B_n^*, E_1^*, ..., E_m^*)$.

4. Stop if j is large enough; otherwise return to step 2

You can also embed plots, for example:

```r
set.seed(501)
theta_em=32.69994
theta_seq=NULL
for(j in 1:1000){
  bs_sample=sample(lifes_B, length(lifes_B), replace = TRUE)
for (i in 1:max_iter){
  theta_new = (n*mean(bs_sample)+S*(tau+theta_current)+
                (m-S)*(theta_current-tau*(p(theta_current))))/(n+m)
  if (abs(theta_new-theta_current)<0.000001){
    theta_seq[j]=theta_new
    break
  }else {theta_current = theta_new
  iter=iter+1}
}
}
hist(theta_seq,breaks = "scott",xlab = bquote(theta),main="")
```




```
cat("standard error of theta =" , sd(theta_seq))
```

```
## standard error of theta = 0.9750778
```

```
cat("95% CI = " ,"(",32.69994-qnorm(0.975)*sd(theta_seq), ",",
    32.69994+qnorm(0.975)*sd(theta_seq), ")")
```

```
## 95% CI =  ( 30.78882 , 34.61106 )
```

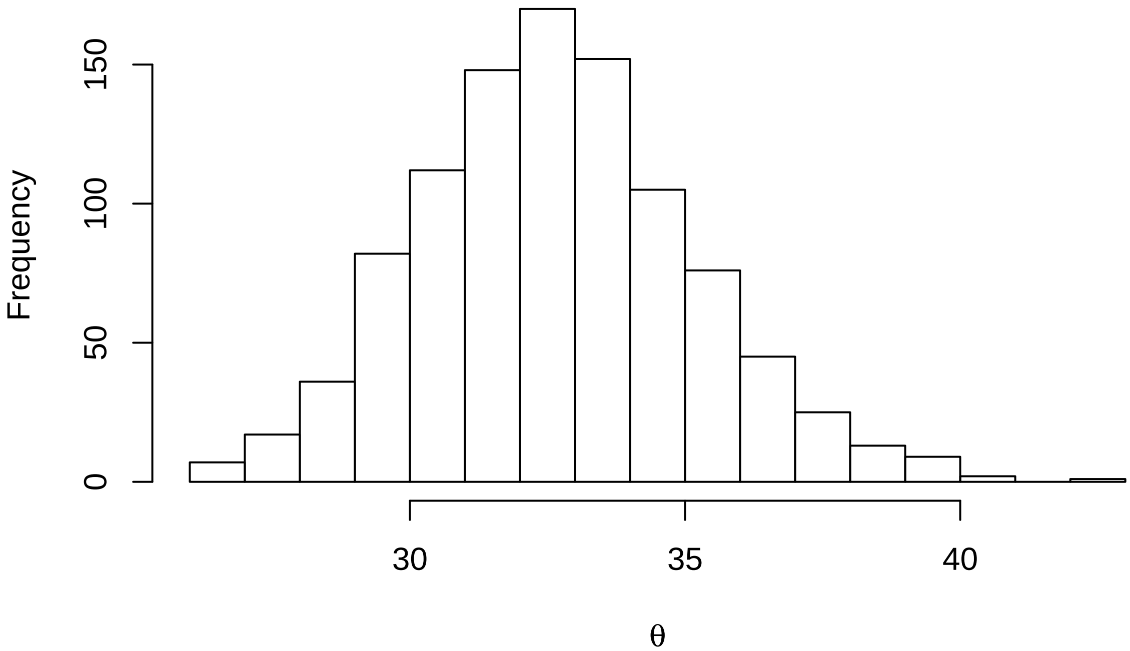
The algorithm converges to the same number well with diefferent starting values I have tried , so in this case starting value is not an issue.

c)

parametric bootstrap pseudocode to obtain an estimated covariance is similar to above, only in step (2) instead of sampling from observed data we will sample from exponential distribution with parameter estimated by E-M.

```
set.seed(501)
theta_em=32.69994
theta_seq_2=NULL
for(j in 1:1000){
  bs2_sample=rexp(n,rate =  1/theta_em)
  S2 = length(which(rexp(m,rate =  1/theta_em) >= tau))
for (i in 1:max_iter){
  theta_new = (n*mean(bs2_sample)+S2*(tau+theta_current)+
                 (m-S2)*(theta_current-tau*(p(theta_current))))/(n+m)
  if (abs(theta_new-theta_current)<0.000001){
    theta_seq_2[j]=theta_new
    break
  }else {theta_current = theta_new
  iter=iter+1}
}
```

```r
}
hist(theta_seq_2,breaks = "scott",xlab = bquote(theta),main="")
```




```r
cat("standard error of theta =" , sd(theta_seq_2))
```

```
## standard error of theta = 2.495772
```

```r
cat("95% CI = " ,"(",32.69994-qnorm(0.975)*sd(theta_seq_2), ",",
    32.69994+qnorm(0.975)*sd(theta_seq_2), ")")
```

```
## 95% CI =  ( 27.80832 , 37.59156 )
```

d)

to contruct a M-H algorithm we first obtain the posterior by assuming the $X = (A_1, ..., A_m)$ be auxilary variables :

$$\pi(\theta, A_1, .., A_m|B_1, .., B_n, S) \propto L_0(\theta, A_1, ..A_m, B_1, ..., B_n, S) \times p(\theta) \times p(A_1)...p(A_m)$$

by assuming $p(\theta) \sim Unif(0, 100)$full conditionals are obtained by:

$$\pi(\theta|\tilde{A}, \tilde{B}, S) \propto 1/\theta^{n+m} exp(-(\sum A_i + \sum B_j)/\theta)p(\theta)$$

$$\pi(B_j|\tilde{A}, \tilde{B_{-j}}, S, \theta) \propto 1/\theta exp(-B_j/\theta), j = 1, .., m$$

$$\pi(A_i|\tilde{A_{-i}}, \tilde{B}, S, \theta) \propto \begin{cases} \frac{1/\theta exp(\frac{-1}{\theta}A)}{exp(-\tau/\theta)} & \text{if } E_i = 1 \\ \frac{1/\theta exp(\frac{-1}{\theta}A)}{1-exp(-\tau/\theta)} & \text{if } E_i = 0 \end{cases}$$

where the last marginalization is obtained based on truncated exponentioal distribution : now the M-H algorithm :

1- pick a starting value say (θ_0, A, B) , A, B are vectors.

2-update θ, A, B in each step to aquire a markov chain.