MAST30027: Modern Applied Statistics

Assignment 3 Solution 2023

1. **Solution:**

   (a) Let $\theta = (\pi_1, \pi_2, \lambda_1, \lambda_2, \lambda_3)$ and $n = 300$.

$$
\begin{aligned}
p(X_1, ...X_n, Z_1, ..., Z_n|\theta) &= \prod_{i=1}^{n} p(X_i|Z_i, \theta)p(Z_i|\theta) \\
&= \prod_{i=1}^{n}\prod_{k=1}^{3}[p(X_i|Z_i = k, \theta)p(Z_i = k|\theta)]^{I_{(Z_i=k)}}.
\end{aligned}
$$

$$
\log[p(X_1, ...X_n, Z_1, ..., Z_n|\theta)] = \sum_{i=1}^{n}\sum_{k=1}^{3} I_{(Z_i=k)}[\log p(X_i|Z_i = k, \theta) + \log p(Z_i = k|\theta)].
$$

$$
\begin{aligned}
Q(\theta, \theta^0) &= E_{Z|X,\theta^0}[\log p(X_1, ...X_n, Z_1, ..., Z_n|\theta)] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{3} p(Z_i = k|X_i, \theta^0)[\log p(X_i|Z_i = k, \theta) + \log p(Z_i = k|\theta)] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{3} p(Z_i = k|X_i, \theta^0)[X_i \log \lambda_k - \lambda_k - \log X_i! + \log \pi_k],
\end{aligned}
$$

   where $\pi_3 = 1 - \pi_1 - \pi_2$.

   (b) Let $\theta^0 = (\pi_1^0, \pi_2^0, \lambda_1^0, \lambda_2^0, \lambda_3^0)$.
   E-step: For $k \in \{1, 2\}$,

$$
\begin{aligned}
p(Z_i = k|X_i, \theta^0) &= \frac{p(Z_i = k, X_i|\theta^0)}{p(X_i|\theta^0)} \\
&= \frac{p(X_i|Z_i = k, \theta^0)p(Z_i = k|\theta^0)}{\sum_{s=1}^{3} p(X_i|Z_i = s, \theta^0)p(Z_i = s|\theta^0)} \\
p(Z_i = 3|X_i, \theta^0) &= 1 - p(Z_i = 1|X_i, \theta^0) - p(Z_i = 2|X_i, \theta^0),
\end{aligned}
$$

   where $p(X_i|Z_i = k, \theta^0) = \frac{(\lambda_k^0)^{X_i} e^{-\lambda_k^0}}{X_i!}$, $p(Z_i = k|\theta^0) = \pi_k^0$ and $p(Z_i = 3|\theta^0) = 1 - \pi_1^0 - \pi_2^0$.

   (c) M-step:

$$
\begin{aligned}
\frac{\partial Q(\theta, \theta^0)}{\partial \pi_1} &= \sum_{i=1}^{n}\Big[\frac{p(Z_i = 1|X_i, \theta^0)}{\pi_1} - \frac{p(Z_i = 3|X_i, \theta^0)}{1 - \pi_1 - \pi_2}\Big] \\
&= \frac{(1 - \pi_1 - \pi_2)\sum_{i=1}^{n} p(Z_i = 1|X_i, \theta^0) - \pi_1 \sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)}{\pi_1(1 - \pi_1 - \pi_2)} = 0 \quad (1)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial Q(\theta, \theta^0)}{\partial \pi_2} &= \sum_{i=1}^{n}\Big[\frac{p(Z_i = 2|X_i, \theta^0)}{\pi_2} - \frac{p(Z_i = 3|X_i, \theta^0)}{1 - \pi_1 - \pi_2}\Big] \\
&= \frac{(1 - \pi_1 - \pi_2)\sum_{i=1}^{n} p(Z_i = 2|X_i, \theta^0) - \pi_2 \sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)}{\pi_1(1 - \pi_1 - \pi_2)} = 0. \quad (2)
\end{aligned}
$$

Let $\pi_3 = 1 - \pi_1 - \pi_2$. From (1) and (2), we obtain

$$\sum_{i=1}^{n} p(Z_i = 1|X_i, \theta^0)\pi_3 = \sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)\pi_1, \tag{3}$$

$$\sum_{i=1}^{n} p(Z_i = 2|X_i, \theta^0)\pi_3 = \sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)\pi_2. \tag{4}$$

Taking sum of (3) and (4), we have

$$\pi_3 \sum_{i=1}^{n} \Big[p(Z_i = 1|X_i, \theta^0) + p(Z_i = 2|X_i, \theta^0)\Big] = (1 - \pi_3) \sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0),$$

$$\hat{\pi}_3 = \frac{\sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)}{n}.$$

From (3), we have

$$\hat{\pi}_1 = \frac{1}{\sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)} \hat{\pi}_3 \sum_{i}^{n} p(Z_i = 1|X_i, \theta^0) = \frac{\sum_{i=1}^{n} p(Z_i = 1|X_i, \theta^0)}{n}.$$

From (4), we have

$$\hat{\pi}_2 = \frac{1}{\sum_{i=1}^{n} p(Z_i = 3|X_i, \theta^0)} \hat{\pi}_3 \sum_{i}^{n} p(Z_i = 2|X_i, \theta^0) = \frac{\sum_{i=1}^{n} p(Z_i = 2|X_i, \theta^0)}{n}.$$

For $k = 1, 2, 3$,

$$\begin{aligned} \frac{\partial Q(\theta, \theta^0)}{\partial \lambda_k} &= \sum_{i=1}^{n} p(Z_i = k|X_i, \theta^0)\Big[\frac{X_i}{\lambda_k} - 1\Big] \\ &= \sum_{i=1}^{n} p(Z_i = k|X_i, \theta^0)\frac{X_i}{\lambda_k} - \sum_{i=1}^{n} p(Z_i = k|X_i, \theta^0) = 0. \\ \hat{\lambda}_k &= \frac{\sum_{i=1}^{n} p(Z_i = k|X_i, \theta^0)X_i}{\sum_{i=1}^{n} p(Z_i = k|X_i, \theta^0)}. \end{aligned}$$

(d) Implement the EM algorithm.

```
> # w.init : initial value for pi
> # lambda.init : initial value for lambda
> # epsilon : stop if the change of the incomplete log-likelihood is less than epsilon
> # max.iter : maximum number of EM-iterations
> mixture.EM <- function(X, w.init, lambda.init, epsilon=1e-5, max.iter=100) {
+
+   w.curr = w.init
+   lambda.curr = lambda.init
+
+   # store incomplete log-likehoods for each iteration
+   log_liks = c()
+
+   # compute incomplete log-likehoods using initial values of parameters.
+   log_liks = c(log_liks, compute.log.lik(X, w.curr, lambda.curr)$ill)
+
+   # set the change in incomplete log-likelihood with 1
+   delta.ll = 1
```

```
+
+   # number of iteration
+   n.iter = 1
+
+   # If the log-likelihood has changed by less than epsilon, EM will stop.
+   while((delta.ll > epsilon) & (n.iter <= max.iter)){
+
+     # run EM step
+     EM.out = EM.iter(X, w.curr, lambda.curr)
+
+     # replace the current value with the new parameter estimate
+     w.curr = EM.out$w.new
+     lambda.curr = EM.out$lambda.new
+
+     # incomplete log-likehoods with new parameter estimate
+     log_liks = c(log_liks, compute.log.lik(X, w.curr, lambda.curr)$ill)
+
+     # compute the change in incomplete log-likelihood
+     delta.ll = log_liks[length(log_liks)]  - log_liks[length(log_liks)-1]
+
+     # increase the number of iteration
+     n.iter = n.iter + 1
+
+   }
+
+   return(list(w.curr=w.curr, lambda.curr=lambda.curr, log_liks=log_liks))
+
+ }
> EM.iter <- function(X, w.curr, lambda.curr) {
+
+   # E-step: compute E_{Z|X,\theta_0}[I(Z_i = k)]
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$
+   prob.x.z = compute.prob.x.z(X, w.curr, lambda.curr)$prob.x.z
+
+   # compute P(Z_i=k | X_i)
+   P_ik = prob.x.z / rowSums(prob.x.z)
+
+   # M-step
+   w.new = colSums(P_ik)/sum(P_ik)  # sum(P_ik) is equivalent to sample size
+   lambda.new = colSums(P_ik*X)/colSums(P_ik)
+
+   return(list(w.new=w.new, lambda.new=lambda.new))
+ }
> # for each sample $X_i$, compute $P(X_i, Z_i=k)$
> compute.prob.x.z <- function(X, w.curr, lambda.curr) {
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$.
+   # store these values in the columns of L:
+   L = matrix(NA, nrow=length(X), ncol= length(w.curr))
+   for(k in seq_len(ncol(L))) {
+     L[, k] = dpois(X, lambda.curr[k])*w.curr[k]
+   }
+
+   return(list(prob.x.z=L))
+ }
> # Compute incomplete log-likehoods
> compute.log.lik <- function(X, w.curr, lambda.curr) {
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$
+   prob.x.z = compute.prob.x.z(X, w.curr, lambda.curr)$prob.x.z
+
+   # incomplete log-likehoods
+   ill = sum(log(rowSums(prob.x.z)))
+
+   return(list(ill=ill))
+ }
```

Run the EM algorithm two times with the two initial values provided in the problem.

```
> # read the data
> X = scan(file="assignment3_prob1_2023.txt", what=double())
> EM1 <- mixture.EM(X, w.init=c(0.3,0.3,0.4), lambda.init=c(3, 20, 35),
+                   epsilon=1e-5, max.iter=100)
> EM2 <- mixture.EM(X, w.init=c(0.1,0.2,0.7), lambda.init=c(5, 25, 40),
+                   epsilon=1e-5, max.iter=100)
> check = rbind(c(EM1$w.curr, EM1$lambda.curr),
+               c(EM2$w.curr, EM2$lambda.curr))
> colnames(check) = c('pi_1', 'pi_2', 'pi_3', 'lambda_1', 'lambda_2', 'lambda_3')
> rownames(check) = c('EM1', 'EM2')
> check
```

```
         pi_1      pi_2      pi_3 lambda_1 lambda_2 lambda_3
EM1 0.2491314 0.2497808 0.5010877 5.167479 18.09393 36.93918
EM2 0.2491307 0.2497786 0.5010907 5.167466 18.09381 36.93911
```

```
> print(EM1$log_liks[length(EM1$log_liks)], digits=16)
```

```
[1] -1151.01487128212
```

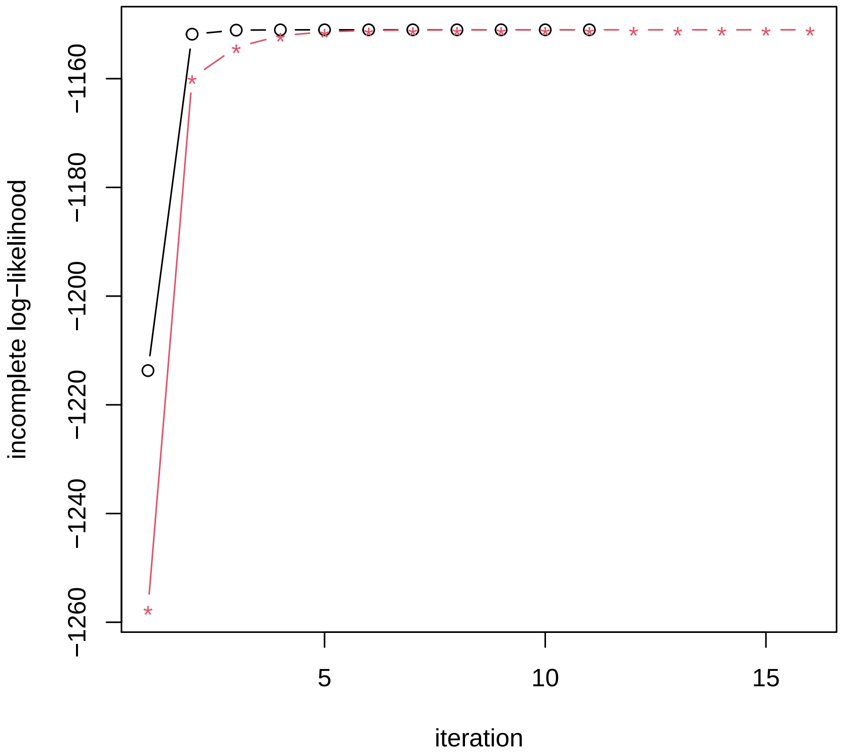
```
> print(EM2$log_liks[length(EM2$log_liks)], digits=16)
```

```
[1] -1151.014870863243
```

Estimates from the two EM runs are very similar and they have very similar incomplete log-likelihoods. So I will report one from the second run as it has slightly higher incomplete log-likelihoods. MLEs for the parameters are $\hat{\pi}_1 = 0.2491307, \hat{\pi}_2 = 0.2497786, \hat{\lambda}_1 = 5.167466, \hat{\lambda}_2 = 18.09381, \hat{\lambda}_3 = 36.93911$.

Check that the incomplete log-likelihoods increases at each step by plotting them

```
> plot(EM1$log_liks, type='b',
+      xlim=c(1,max(length(EM1$log_liks), length(EM2$log_liks))),
+      ylim=range(c(EM1$log_liks, EM2$log_liks)),
+      ylab='incomplete log-likelihood', xlab='iteration')
> points(EM2$log_liks, type='b', pch='*', col=2)
```



2. **Solution:**

   (a) Let $\theta = (\pi_1, \pi_2, \lambda_1, \lambda_2, \lambda_3)$, $n = 300$ and $n' = 100$.

$$
\begin{aligned}
& p(X_1, ...X_n, Z_1, ..., Z_n, X_{n+1}, ..., X_{n+n'}|\theta) \\
= & \prod_{i=1}^{n} p(X_i|Z_i, \theta)p(Z_i|\theta) \prod_{i=n+1}^{n+n'} p(X_i|\theta) \\
= & \prod_{i=1}^{n} \prod_{k=1}^{3} [p(X_i|Z_i = k, \theta)p(Z_i = k|\theta)]^{I_{(Z_i=k)}} \prod_{i=n+1}^{n+n'} p(X_i|\theta).
\end{aligned}
$$

$$
\begin{aligned}
& \log[p(X_1, ...X_n, Z_1, ..., Z_n, X_{n+1}, ..., X_{n+n'}|\theta)] \\
= & \sum_{i=1}^{n} \sum_{k=1}^{3} I_{(Z_i=k)}[\log p(X_i|Z_i = k, \theta) + \log p(Z_i = k|\theta)] + \sum_{i=n+1}^{n+n'} \log p(X_i|\theta)
\end{aligned}
$$

$$
\begin{aligned}
Q(\theta,\theta^0) &= E_{Z|X,\theta^0}[\log p(X_1,...X_n,Z_1,...,Z_n,X_{n+1},...,X_{n+n'}|\theta)]\\
&= \sum_{i=1}^{n}\sum_{k=1}^{3} p(Z_i=k|X_i,\theta^0)[\log p(X_i|Z_i=k,\theta)+\log p(Z_i=k|\theta)] + \sum_{i=n+1}^{n+n'} \log p(X_i|\theta)\\
&= \sum_{i=1}^{n}\sum_{k=1}^{3} p(Z_i=k|X_i,\theta^0)[X_i\log\lambda_k-\lambda_k-\log X_i!+\log\pi_k] + \sum_{i=n+1}^{n+n'} [X_i\log\lambda_2-\lambda_2-\log X_i!],
\end{aligned}
$$

where $\pi_3 = 1-\pi_1-\pi_2$.

(b) E-step: same as the E-step in the solution for the problem 1 (b).
M-step: same as the M-step in the solution for the problem 1 (c) except for the following:

$$
\begin{aligned}
\frac{\partial Q(\theta,\theta^0)}{\partial\lambda_2} &= \sum_{i=1}^{n} p(Z_i=2|X_i,\theta^0)\Big[\frac{X_i}{\lambda_2}-1\Big] + \sum_{i=n+1}^{n+n'}\Big[\frac{X_i}{\lambda_2}-1\Big]\\
&= \sum_{i=1}^{n} p(Z_i=2|X_i,\theta^0)\frac{X_i}{\lambda_2} - \sum_{i=1}^{n} p(Z_i=2|X_i,\theta^0) + \sum_{i=n+1}^{n+n'}\frac{X_i}{\lambda_2} - n' = 0.
\end{aligned}
$$

Hence,

$$
\hat{\lambda}_2 = \frac{\sum_{i=1}^{n} p(Z_i=2|X_i,\theta^0)X_i + \sum_{i=n+1}^{n+n'} X_i}{\sum_{i=1}^{n} p(Z_i=2|X_i,\theta^0)+n'}.
$$

(c) Implement the EM algorithm.

```
> # X  : X_1, ..., X_300  which follow a mixture of Poisson distribution
> # X0 : X_301, .., X_400 which follow a Poisson distribution
> mixture.EM <- function(X, X0, w.init, lambda.init, epsilon=1e-5, max.iter=100) {
+
+   w.curr = w.init
+   lambda.curr = lambda.init
+
+   # store incomplete log-likehoods for each iteration
+   log_liks = c()
+
+   # compute incomplete log-likehoods using initial values of parameters.
+   log_liks = c(log_liks, compute.log.lik(X, X0, w.curr, lambda.curr)$ill)
+
+   # set the change in incomplete log-likelihood with 1
+   delta.ll = 1
+
+   # number of iteration
+   n.iter = 1
+
+   # If the log-likelihood has changed by less than epsilon, EM will stop.
+   while((delta.ll > epsilon) & (n.iter <= max.iter)){
+
+     # run EM step
+     EM.out = EM.iter(X, X0, w.curr, lambda.curr)
+
+     # replace the current value with the new parameter estimate
+     w.curr = EM.out$w.new
+     lambda.curr = EM.out$lambda.new
+
+     # incomplete log-likehoods with new parameter estimate
+     log_liks = c(log_liks, compute.log.lik(X, X0, w.curr, lambda.curr)$ill)
+
+     # compute the change in incomplete log-likelihood
+     delta.ll = log_liks[length(log_liks)]  - log_liks[length(log_liks)-1]
+
+     # increase the number of iteration
+     n.iter = n.iter + 1
+   }
+
+   return(list(w.curr=w.curr, lambda.curr=lambda.curr, log_liks=log_liks))
+
+ }
> EM.iter <- function(X, X0, w.curr, lambda.curr) {
+
+   # E-step: compute E_{Z|X,\theta_0}[I(Z_i = k)]
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$
+   prob.x.z = compute.prob.x.z(X, w.curr, lambda.curr)$prob.x.z
+
+   # compute P(Z_i=k | X_i)
+   P_ik = prob.x.z / rowSums(prob.x.z)
+
+   # M-step
+   w.new = colSums(P_ik)/sum(P_ik)  # sum(P_ik) is equivalent to sample size
+   ### Change!!!
+   lambda.new = rep(NA, length(w.new))
+   lambda.new[c(1,3)] = (colSums(P_ik*X)/(colSums(P_ik)))[c(1,3)]
+   lambda.new[2] = (colSums(P_ik*X)[2] + sum(X0))/( colSums(P_ik)[2] + length(X0))
+
+   return(list(w.new=w.new, lambda.new=lambda.new))
+ }
> # for each sample $X_i$, compute $P(X_i, Z_i=k)$
> compute.prob.x.z <- function(X, w.curr, lambda.curr) {
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$.
+   # Store these values in the columns of L:
+   L = matrix(NA, nrow=length(X), ncol= length(w.curr))
+   for(k in seq_len(ncol(L))) {
+     L[, k] = dpois(X, lambda.curr[k])*w.curr[k]
+   }
+
+   return(list(prob.x.z=L))
+ }
> # Compute incomplete log-likehoods
> compute.log.lik <- function(X, X0, w.curr, lambda.curr) {
+
+   # for each sample $X_i$, compute $P(X_i, Z_i=k)$
+   prob.x.z = compute.prob.x.z(X, w.curr, lambda.curr)$prob.x.z
+
+   # incomplete log-likehoods
+   ill = sum(log(rowSums(prob.x.z))) + sum(log(dpois(X0, lambda.curr[2])))
+
+   return(list(ill=ill))
+ }
```

Run the EM algorithm two times with the two initial values provided in the problem.

```
> X0 = scan(file="assignment3_prob2_2023.txt", what=double())
> EM1 <- mixture.EM(X, X0, w.init=c(0.3,0.3,0.4), lambda.init=c(3, 20, 35),
+                    epsilon=1e-5, max.iter=100)
> EM2 <- mixture.EM(X, X0, w.init=c(0.1,0.2,0.7), lambda.init=c(5, 25, 40),
+                    epsilon=1e-5, max.iter=100)
> check = rbind(c(EM1$w.curr, EM1$lambda.curr),
+               c(EM2$w.curr, EM2$lambda.curr))
> colnames(check) = c('pi_1', 'pi_2', 'pi_3', 'lambda_1', 'lambda_2', 'lambda_3')
> rownames(check) = c('EM1', 'EM2')
> check
```

```
         pi_1      pi_2      pi_3 lambda_1 lambda_2 lambda_3
EM1 0.2461850 0.2447837 0.5090313 5.118187 17.36453 36.76509
EM2 0.2461848 0.2447753 0.5090399 5.118186 17.36442 36.76488
```

```
> print(EM1$log_liks[length(EM1$log_liks)], digits=16)
```

```
[1] -1437.052845923898
```

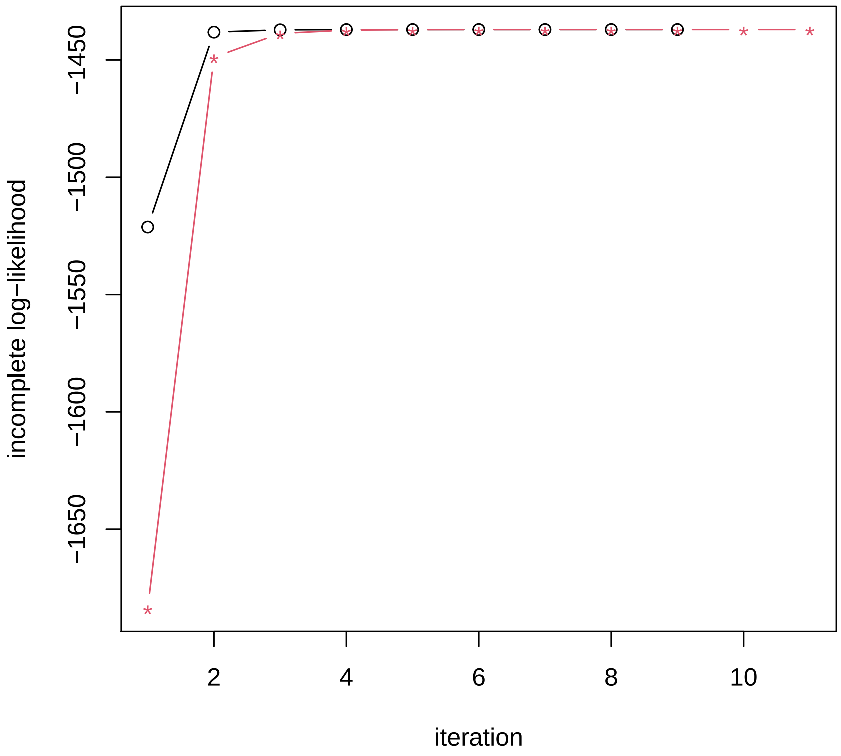
```
> print(EM2$log_liks[length(EM2$log_liks)], digits=16)
```

```
[1] -1437.052845374368
```

Estimates from the two EM runs are very similar and they have very similar incomplete log-likelihoods. So I will report one from the second run as it has slightly higher incomplete log-likelihoods. MLEs for the parameters are $\hat{\pi}_1 = 0.2461848, \hat{\pi}_2 = 0.2447753, \hat{\lambda}_1 = 5.118186, \hat{\lambda}_2 = 17.36442, \hat{\lambda}_3 = 36.76488$.

Check that the incomplete log-likelihoods increases at each step by plotting them

```
> plot(EM1$log_liks, type='b',
+      xlim=c(1,max(length(EM1$log_liks), length(EM2$log_liks))),
+      ylim=range(c(EM1$log_liks, EM2$log_liks)),
+      ylab='incomplete log-likelihood', xlab='iteration')
> points(EM2$log_liks, type='b', pch='*',  col=2)
```


−1450
−1500
−1550
−1600
−1650
incomplete log−likelihood
2
4
6
8
10
iteration