Stat-Computing-2 Assignment-5

Anjan Ghosh, MS1803

Here we will consider a hierarchical Bayes approach for normal mixture model.

Data Generation

200 observations are simulated from the mixture distribution:

$$f = 0.35N(10, 5^2) + 0.65N(25, 5^2)$$

This is done in two steps.

- First, a bernoulli observation is drawn $\sim Ber(p = 0.35)$.
- If the simulated value is less than 0.35, we draw one sample from $N(10, 5^2)$ distribution, else from $N(25, 5^2)$ distribution.

We denote this data as $y = (y_1, y_2, \ldots, y_{200})$.

Derivation of the MCMC

We use the simulated dataset and estimate the underlying model parameters using a hierarchichal Bayes algorithm. We assume that data comes from the model of the form:

$$f = pN(\mu_0, \sigma^2) + (1 - p)N(\mu_1, \sigma^2)$$

This implies that with probabilty p, the data is coming from $N(\mu_0, \sigma^2)$ distribution and from $N(\mu_1, \sigma^2)$ with probability $1 - p$.

The hierarchical structure that we develop is:

$$\begin{aligned}
p &\sim Uniform(0, 1), \\
K_i|p &\sim Bernoulli(p),\ i = 1, 2, \ldots, 200 \\
y_i|K_i = k &\sim N(\mu_{k+1\ mod\ 2}, \sigma^2),\ k = 0, 1 \\
\mu_k &\sim N(0, 100),\ k = 0, 1 \\
\sigma^2 &\sim Inverse\ Gamma(0.01, 0.01)
\end{aligned}$$

This fits the undertaken mixture model since, when $K_i = 1$ with probability p, we take sample from $N(\mu_0, \sigma^2)$ and when $K_i = 0$ with probability $1 - p$, we take sample from $N(\mu_1, \sigma^2)$, for a certain i.

This helps in writing the joint likelihood and derive the posteriors from there.

Let $K = \sum_{i=1}^{200} K_i$.

$$
\begin{aligned}
\pi(y, K_1, K_2, \ldots, K_{200}, p, \mu_0, \mu_1, \sigma^2) &\propto \prod_{i=1}^{200} f(y_i|K_i, \mu_0, \mu_1, \sigma^2)\pi(K_i|p)\pi(p)\pi(\mu_0)\pi(\mu_1)\pi(\sigma^2) \\
&\propto p^K(1-p)^{200-K} \prod_{i:K_i=1} f(y_i|\mu_0, \sigma^2) \prod_{i:K_i=0} f(y_i|\mu_1, \sigma^2) \times 1 \times \pi(\mu_0)\pi(\mu_1)\pi(\sigma^2) \\
\therefore \pi(p|K_1, K_2, \ldots, K_{200}) &\propto p^K(1-p)^{200-K} \\
\sigma^2 &\propto (\sigma^2)^{0.01+100+1} \exp -\frac{1}{\sigma^2}(0.01 + \frac{1}{2}(\sum_{i:K_i=1}(y_i-\mu_0)^2 + \sum_{i:K_i=0}(y_i-\mu_1)^2))
\end{aligned}
$$

Therefore it can be said that,

$$
\begin{aligned}
p|K_1, K_2, \ldots, K_{200} &\sim Beta(K+1, 200-K+1) \\
\sigma^2|y, K_1, K_2, \ldots, K_{200}, \mu_0, \mu_1 &\sim Inverse\ Gamma(0.01+100, 0.01 + \frac{1}{2}(\sum_{i:K_i=1}(y_i-\mu_0)^2 + \sum_{i:K_i=0}(y_i-\mu_1)^2)) \\
\mu_0|y, K_1, K_2, \ldots, K_{200}, \sigma^2 &\sim N(\frac{\frac{\sum_{i:K_i=1} y_i}{\sigma^2}}{\frac{K}{\sigma^2} + \frac{1}{10^2}}, \frac{1}{\frac{K}{\sigma^2} + \frac{1}{10^2}}) \\
\mu_1|y, K_1, K_2, \ldots, K_{200}, \sigma^2 &\sim N(\frac{\frac{\sum_{i:K_i=0} y_i}{\sigma^2}}{\frac{200-K}{\sigma^2} + \frac{1}{10^2}}, \frac{1}{\frac{200-K}{\sigma^2} + \frac{1}{10^2}})
\end{aligned}
$$

Also,

$$
\begin{aligned}
K_i|y_i, p, \mu_0, \mu_1, \sigma^2 &\sim \frac{f(y_i, K_i|p, \mu_0, \mu_1, \sigma^2)}{f(y_i|p, \mu_0, \mu_1, \sigma^2)},\ i = 1, 2, \ldots, 200\ independently \\
&= \frac{f(y_i|K_i, \mu_0, \mu_1, \sigma^2)\pi(K_i|p)}{f(y_i|p, \mu_0, \mu_1, \sigma^2)} \\
&= \frac{p^{K_i}(1-p)^{1-K_i} \times f(y_i|K_i, \mu_0, \mu_1, \sigma^2)}{p \times f(y_i|K_i, \mu_0, \sigma^2) + (1-p) \times f(y_i|K_i, \mu_1, \sigma^2)}
\end{aligned}
$$

$\therefore K_i|y_i, p, \mu_0, \mu_1, \sigma^2 \sim Bernoulli(\frac{p \times f(y_i|K_i, \mu_0, \sigma^2)}{p \times f(y_i|K_i, \mu_0, \sigma^2) + (1-p) \times f(y_i|K_i, \mu_1, \sigma^2)}),\ i = 1, 2, \ldots, 200\ independently.$

These gives the required posterior distributions. Following steps are followed in sequence for 20001 times. First 2000 samples are discarded as burn-in and remaining ones are thinned by saving every 10th iteration.

1. $p, \mu_0, \mu_1, \sigma^2$ are initialised with the help of respective prior distributions.

Inside a loop,

2. $K_i,\ i = 1, 2, \ldots, 200$ are simulated independetly from its posterior.
3. p is simulated from its posterior.
4. σ^2 is simulated from its posterior.
5. μ_0, μ_1 are simulated from their posteriors.

R code

```
### DATA GENERATION
n<-200
```

```r
mu<-c(10,25)
sig<-c(5,5)
p<-c(.35,.65)
z<-sample(1:length(p),prob = p,replace = T,size = 200)
x<-rnorm(n,mu[z],sig[z])

### PRIOR INITIALISATIONS AND VECTOR/MATRIX INITIALISATIONS
p_init<-runif(1)
mu_init<-rnorm(2,0,10)
sig_init<-sqrt(1/rgamma(1,0.01,0.01))

sig<-numeric(20001)
p<-numeric(20001)
mu<-matrix(,20001,2)

### ITERATIONS
for(i in 1:20001)
{
  K<-rbinom(n,1,p_init*dnorm(x,mu_init[1],sig_init)/
              (p_init*dnorm(x,mu_init[1],sig_init)+
                 (1-p_init)*dnorm(x,mu_init[2],sig_init)))
  p[i]<-rbeta(1,sum(K==1)+1,sum(K==0)+1)
  sig[i]<-sqrt(1/rgamma(1,0.01+n/2,0.01+(sum((x[K==1]-mu_init[1])^2)+
                                          sum((x[K==0]-mu_init[2])^2))/2))
  mu[i,1]<-rnorm(1,sum(K==1)*mean(x[K==1])*100/(100*sum(K==1)+sig[i]^2),
                 100*sig[i]^2/(100*sum(K==1)+sig[i]^2))
  mu[i,2]<-rnorm(1,sum(K==0)*mean(x[K==0])*100/(100*sum(K==0)+sig[i]^2),
                 100*sig[i]^2/(100*sum(K==0)+sig[i]^2))
  p_init<-p[i]
  sig_init<-sig[i]
  mu_init<-mu[i,]
}
### BURN-IN AND THINNING, ESTIMATES AND 95% CONFIDENCE INTERVALS
index<-seq(from=2001,to=20001,by=10)

mean(p[index])
apply(mu[index,], 2, mean)
mean(sig[index])
quantile(p[index],c(0.025,0.5,0.975))
apply(mu[index,],2,quantile,c(0.025,0.5,0.975))
quantile(sig[index],c(0.025,0.5,0.975))
```

Results

Table containing posterior estimates and 95% credible interval of the parameters

Parameter	Posterior Mean	Posterior Median	95% Credible Interval
p	0.3863311	0.3860024	(0.3076422, 0.4682178)
μ_1	10.9103679	10.9203482	(9.8611319, 12.0471979)
μ_2	25.3811611	25.3949254	(24.5405844, 26.1011435)
$sigma$	4.8886941	4.8601455	(4.3251507, 5.6227762)

Conclusion

- Final estimates are more or less close to the original values of parameters that were used in generating samples.
- Posterior medians are close to posterior means.
- 2.5% and 97.5% quantiles of posterior distributions lie closely to each other for all the parameters except for p. Increase in sample size might have a positive effect on this.
- Estimates are comparable with that obtained through EM algorithm previously.