

Nonzero Mean ASH

Chaoxing Dai

2014-Sep-10

Normal mixture with normal likelihood

df=NULL,mixcompdist="normal"

Likelihood

By assuming a unimodal distribution, we have the hierarchical model being the following:

$$\beta_j \sim g(\cdot) \sim \sum_{k=1}^K \pi_k f_k(\cdot) \sim \sum_{k=1}^K \pi_k N(\cdot; \mu, \sigma_k^2)$$
$$\hat{\beta}_j \mid \beta_j, s_j^2 \sim N(\cdot; \beta_j, s_j^2)$$

Then we would have the likelihood for π and μ being (Here we assume iid or exchangeable data)

$$\begin{aligned} L(\pi, \mu) &= \prod_{j=1}^n p(D_j \mid \pi, \mu) = \prod_{j=1}^n \int p(\hat{\beta}_j \mid s_j, \pi, \mu, \beta_j) p(\beta_j \mid \pi, \mu) d\beta_j \\ &= \prod_{j=1}^n \int \frac{1}{\sqrt{2\pi s_j^2}} \exp \left\{ -\frac{1}{2s_j^2} (\hat{\beta}_j - \beta_j)^2 \right\} \sum_{k=1}^K \pi_k \frac{1}{\sqrt{2\pi \sigma_k^2}} \exp \left\{ -\frac{1}{2\sigma_k^2} (\beta_j - \mu)^2 \right\} d\beta_j \\ &= \prod_{j=1}^n \sum_{k=1}^K \pi_k \int \frac{1}{\sqrt{2\pi s_j^2 \times 2\pi \sigma_k^2}} \exp \left\{ -\frac{1}{2s_j^2} (\hat{\beta}_j - \beta_j)^2 - \frac{1}{2\sigma_k^2} (\beta_j - \mu)^2 \right\} d\beta_j \\ &= \prod_{j=1}^n \sum_{k=1}^K \pi_k \frac{1}{\sqrt{2\pi (s_j^2 + \sigma_k^2)}} \exp \left[-\frac{(\hat{\beta}_j - \mu)^2}{2(s_j^2 + \sigma_k^2)} \right] \end{aligned}$$

Where the term after π_k is the density of $N(\cdot; \mu, s_j^2 + \sigma_k^2)$ at $\hat{\beta}_j$, i.e.

$$L(\pi, \mu) = \prod_{j=1}^n \sum_{k=1}^K \pi_k N(\hat{\beta}_j; \mu, s_j^2 + \sigma_k^2)$$

EM Algorithm

For convenience we denote $x_j \equiv \hat{\beta}_j$, and introduce a latent variable z_j specifying the mixture component that x_j belongs to. We then have

$$p(x_j, z_j \mid \mu, \pi) = \prod_{k=1}^K [\pi_k N(x_j; \mu, s_j^2 + \sigma_k^2)]^{I_{z_j=k}}$$

$$\log p(x_j, z_j \mid \mu, \pi) = \sum_{k=1}^K [I_{z_j} = k] [\log(\pi_k) - \frac{1}{2} \log(2\pi) - \frac{1}{2} \log(s_j^2 + \sigma_k^2) - \frac{(x_j - \mu)^2}{2(s_j^2 + \sigma_k^2)}]$$

$$\log p(X, Z \mid \mu, \pi) = \sum_{j=1}^n \sum_{k=1}^K [I_{z_j} = k] [\log(\pi_k) - \frac{1}{2} \log(2\pi) - \frac{1}{2} \log(s_j^2 + \sigma_k^2) - \frac{(x_j - \mu)^2}{2(s_j^2 + \sigma_k^2)}]$$

Then we will have the E-step, given the estimate of π and μ , we calculate the responsibility of j -th data point to the k -th mixture component.

$$\omega_{jk} = p(z_j = k \mid x_j, \pi, \mu) = \frac{\pi_k N(x_j; \mu, s_j^2 + \sigma_k^2)}{\sum_{l=1}^K \pi_l N(x_j; \mu, s_j^2 + \sigma_l^2)}$$

For the M-step, we take ω_{jk} in the previous step, and maximize the expected loglikelihood over the latent variable Z .

$$Q = E_{Z \mid X, \mu, \pi} [\log p(X, Z \mid \mu, \pi)] = \sum_{j=1}^n \sum_{k=1}^K [\omega_{jk}] [\log(\pi_k) - \frac{1}{2} \log(2\pi) - \frac{1}{2} \log(s_j^2 + \sigma_k^2) - \frac{(x_j - \mu)^2}{2(s_j^2 + \sigma_k^2)}]$$

We take partial derivatives of Q with respect to π and μ and set them to be 0 to get the next estimates of our parameters, subject to the constraint that $\sum_{k=1}^K \pi_k = 1$. Then we have

$$\hat{\mu} = \frac{\sum_{j=1}^n \sum_{k=1}^K \frac{\omega_{jk} x_j}{s_j^2 + \sigma_k^2}}{\sum_{j=1}^n \sum_{k=1}^K \frac{\omega_{jk}}{s_j^2 + \sigma_k^2}}$$

$$\hat{\pi}_k = \frac{1}{n} \sum_{j=1}^n \omega_{jk}$$

Then we go back to the E-step and repeat with the new estimate, until we get a convergence.

Uniform mixture with Normal likelihood

`df=NULL, mixcompdist="uniform"`

Likelihood

We know that the convolution of a uniform distribution and a normal would be of the following form

$$f(x) \sim N(x; \mu, \sigma^2)$$

$$g(y) \sim U(y; a, b)$$

$$(f * g)(z) = \frac{1}{b-a} [\Psi_0(\frac{z-a-\mu}{\sigma}) - \Psi_0(\frac{z-b-\mu}{\sigma})]$$

Where $\Psi_0(\cdot)$ is the cdf for standard normal distribution.

By assuming a unimodal distribution, we have the hierarchical model being the following:

$$\beta_j \sim g(\cdot) \sim \sum_{k=1}^K \pi_k f_k(\cdot) \sim \sum_{k=1}^K \pi_k U(\cdot; -\sigma_k + \mu, \sigma_k + \mu)$$

$$\hat{\beta}_j \mid \beta_j, s_j^2 \sim N(\cdot; \beta_j, s_j^2)$$

Then we would have the likelihood for π and μ being (Here we assume iid or exchangeable data)

$$\begin{aligned} L(\pi, \mu) &= \prod_{j=1}^n p(D_j \mid \pi, \mu) = \prod_{j=1}^n \int p(\hat{\beta}_j \mid s_j, \pi, \mu, \beta_j) p(\beta_j \mid \pi, \mu) d\beta_j \\ &= \prod_{j=1}^n \sum_{k=1}^K \pi_k \frac{1}{2\sigma_k} [\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})] \end{aligned}$$

EM Algorithm

For convenience we denote $x_j \equiv \hat{\beta}_j$, and introduce a latent variable z_j specifying the mixture component that x_j belongs to. We then have

$$\begin{aligned} p(x_j, z_j \mid \mu, \pi) &= \prod_{k=1}^K [\pi_k \frac{1}{2\sigma_k} [\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})]]^{I_{z_j=k}} \\ \log p(x_j, z_j \mid \mu, \pi) &= \sum_{k=1}^K [I_{z_j=k} \log[\pi_k \frac{1}{2\sigma_k} [\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})]] \\ &= \sum_{k=1}^K [I_{z_j=k} [\log \pi_k - \log(2\sigma_k) + \log[\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})]]] \end{aligned}$$

Then we will have the E-step, given the estimate of π and μ , we calculate the responsibility of j-th data point to the k-th mixture component.

$$\omega_{jk} = p(z_j = k \mid x_j, \pi, \mu) = \frac{\frac{1}{2\sigma_k} [\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})]}{\sum_{l=1}^K \frac{1}{2\sigma_l} [\Psi_0(\frac{\hat{\beta}_j + \sigma_l - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_l - \mu}{s_j})]}$$

For the M-step, we take ω_{jk} in the previous step, and maximize the expected loglikelihood over the latent variable Z .

$$Q = E_{Z \mid X, \mu, \pi} [\log p(X, Z \mid \mu, \pi)] = \sum_{j=1}^n \sum_{k=1}^K [\omega_{jk} [\log \pi_k - \log(2\sigma_k) + \log[\Psi_0(\frac{\hat{\beta}_j + \sigma_k - \mu}{s_j}) - \Psi_0(\frac{\hat{\beta}_j - \sigma_k - \mu}{s_j})]]]$$

The maximization over μ no longer has a closed form, and we will use the one-dimensional R optimization function `optim()` to perform the M step.

For π , we have the same expression as before.

$$\hat{\pi}_k = \frac{1}{n} \sum_{j=1}^n \omega_{jk}$$

Then we go back to the E-step and repeat with the new estimate, until we get a convergence.

Other cases

```
df=NULL,mixcompdist="halfuniform"  
df=scalar,mixcompdist="normal"  
df=scalar,mixcompdist="uniform"  
df=scalar,mixcompdist="halfuniform"
```

Similar to the second one, as we do not have closed form solution for μ in the M step of EM algorithm, would use optimization function with SQUAREM, and a lot of functions in EMest will be re-used.