HW5

Qi Qi

10/12/2018

1.

E-steps:

$$Q(\Psi|\Psi^{(k)}) = E[\ln L(\Psi|x_i, y_i, z)|x_i, y_i, \Psi^{(k)}]$$

$$= \sum_{z_{ij}} p(z_{ij}|x_i, y_i, \Psi^{(k)})(\sum_{i=1}^{n}\sum_{j=1}^{m} z_{ij} \log \pi_j \phi(y_i - x_i^T\beta_j; 0, \sigma^2))$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{m}[\sum_{z_{ij}} p(z_{ij}|x_i, y_i, \Psi^{(k)})z_{ij}] \log \pi_j \phi(y_i - x_i^T\beta_j; 0, \sigma^2)$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)}\{\log \pi_j + \log \phi(y_i - x_i^T\beta_j; 0, \sigma^2)\}$$

where

$$p_{ij}^{(k+1)} = \sum_{z_{ij}} p(z_{ij}|x_i, y_i, \Psi^{(k)})z_{ij} = E[z_{ij}|x_i, y_i, \Psi^{(k)}]$$

Since $z_{ij} = 1$ if i^{th} observation is from j^{th} component, then

$$p_{ij}^{(k+1)} = E[z_{ij}|x_i, y_i, \Psi^{(k)}] = p(z_{ij} = 1|x_i, y_i, \Psi^{(k)})$$

$$= \frac{p(z_{ij} = 1, x_i, y_i, \Psi^{(k)})}{p(x_i, y_i, \Psi^{(k)})} = \frac{\pi_j^{(k)}\phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^{2^{(k)}})}{\sum_{j=1}^{m}\pi_j^{(k)}\phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^{2^{(k)}})}$$

M-steps:

Since $1 = \int f(y_i|x_i, \Psi)dy_i = \int \sum_{j=1}^{m}\pi_j\phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^2)dy_i = \sum_{j=1}^{m}\pi_j \int \phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^2)dy_i = \sum_{j=1}^{m}\pi_j$, then

$$\frac{\partial Q(\Psi|\Psi^{(k)})}{\partial \pi_1} = \sum_{i=1}^{n} p_{i1}^{(k+1)}/\pi_1 - \sum_{i=1}^{n} p_{im}^{(k+1)}/(1 - \pi_1 - ... - \pi_{m-1}) = 0$$

$$\vdots$$

$$\frac{\partial Q(\Psi|\Psi^{(k)})}{\partial \pi_{m-1}} = \sum_{i=1}^{n} p_{i(m-1)}^{(k+1)}/\pi_1 - \sum_{i=1}^{n} p_{im}^{(k+1)}/(1 - \pi_1 - ... - \pi_{m-1}) = 0$$

$$\Rightarrow \sum_{i=1}^{n} p_{i1}^{(k+1)}/\pi_1^{(k+1)} = ... = \sum_{i=1}^{n} p_{im}^{(k+1)}/\pi_m^{(k+1)} = c$$

Since $\sum_{j=1}^{m}\pi_j^{(k+1)} = 1$, then

$$\sum_{j=1}^{m}\sum_{i=1}^{n} p_{ij}^{(k+1)}/c = 1 \Rightarrow c = \sum_{j=1}^{m}\sum_{i=1}^{n} p_{ij}^{(k+1)} = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} = \sum_{i=1}^{n}\frac{\sum_{j=1}^{m}\pi_j^{(k)}\phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^{2^{(k)}})}{\sum_{j=1}^{m}\pi_j^{(k)}\phi(y_i - x_i^T\beta_j^{(k)}; 0, \sigma^{2^{(k)}})} = n$$

So,

$$\pi_j^{(k+1)} = \frac{\sum_{i=1}^{n} p_{ij}^{(k+1)}}{n},\ j = 1, ..., m$$

$$\frac{\partial Q(\Psi|\Psi^{(k)})}{\partial \beta_j} = \sum_{i=1}^{n} p_{ij}^{(k+1)} \frac{2x_i y_i - 2x_i x_i^T \beta_j}{2\sigma^2} = \frac{1}{\sigma^2}[\sum_{i=1}^{n} x_i p_{ij}^{(k+1)} y_i - \sum_{i=1}^{n} p_{ij}^{(k+1)} x_i x_i^T \beta_j] = 0$$

$$\Rightarrow \beta_j^{(k)} = (\sum_{i=1}^{n} p_{ij}^{(k+1)} x_i x_i^T)^{-1} (\sum_{i=1}^{n} x_i p_{ij}^{(k+1)} y_i),\ j = 1, ..., m$$

$$\frac{\partial Q(\Psi|\Psi^{(k)})}{\partial \sigma^2} = \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k+1)} [\frac{(y_i - x_i^T \beta_j)^2}{2\sigma^4} - \frac{1}{2\sigma^2}]$$

$$= \frac{1}{2\sigma^4}[\sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k+1)} (y_i - x_i^T \beta_j)^2 - \sigma^2 \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k+1)}] = \frac{1}{2\sigma^4}[\sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k+1)} (y_i - x_i^T \beta_j)^2 - \sigma^2 n] = 0$$

$$\Rightarrow \sigma^{2^{(k+1)}} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k+1)} (y_i - x_i^T \beta_j)^2}{n}$$

2.

```r
regmix_em <- function(y, xmat, pi.init, beta.init, sigma.init, control){
  xmat <- as.matrix(xmat)
  y <- as.matrix(y)
  pi0 <- pi.init
  bet0 <- beta.init
  sig0 <- sigma.init ^ 2
  n <- nrow(xmat)
  p <- ncol(xmat)
  m <- ncol(beta.init)
  for (l in 1:control$maxit){

    ### pij(k+1)
    p1 <- matrix(NA, n, m)
    for (i in 1:n){
      for (j in 1:m){
        p1[i, j] <- pi0[j] * dnorm(y[i] - t(xmat[i,]) %*% bet0[, j], 0, sig0^2)
      }
    }
    pj <- rowSums(p1)
  for (i in 1:n){
    p1[i, ] <- p1[i, ]/pj[i]
  }

    ### pi(k+1)
    pi1 <- colMeans(p1)

    ### beta(k+1)
    bet1 <- matrix(NA, p, m)
    for (j in 1:m){
      A <- matrix(0, p, p)
      B <- matrix(0, p, 1)
      for (i in 1:n){
      A <- A + p1[i, j] * xmat[i, ] %*% t(xmat[i, ])
      B <- B + p1[i, j] * y[i] * xmat[i, ]
      }
```

```
      bet1[,j] <- solve(A) %*% B
    }

    ### sigma^2 (k+1)
    C <- 0
    for (j in 1:m){
      for (i in 1:n){
        C <- C + p1[i,j] * (y[i] - t(xmat[i,]) %*% bet1[, j]) ^ 2
      }
    }
    sig1 <- C / n
  if (max(sqrt(sum((pi1 - pi0)^2)), norm(bet1 - bet0),  abs(sig1 - sig0)) < control$tol){
      result <- list(pi1, bet1, sig1)
        return(result)
  }
    pi0 <- pi1
    bet0 <- bet1
    sig0 <- sig1
  }
  return(NA)
}
```

3.

```
regmix_sim <- function(n, pi, beta, sigma) {
    K <- ncol(beta)
    p <- NROW(beta)
    xmat <- matrix(rnorm(n * p), n, p) # normal covaraites
    error <- matrix(rnorm(n * K, sd = sigma), n, K)
    ymat <- xmat %*% beta + error # n by K matrix
    ind <- t(rmultinom(n, size = 1, prob = pi))
    y <- rowSums(ymat * ind)
    data.frame(y, xmat)
}
```

```
n <- 400
pi <- c(.3, .4, .3)
bet <- matrix(c( 1,  1,  1,
                -1, -1, -1), 2, 3)
sig <- 1
set.seed(1205)
dat <- regmix_sim(n, pi, bet, sig)
regmix_em(y = dat[,1], xmat = dat[,-1],
          pi.init = pi / pi / length(pi),
          beta.init = bet * 0,
          sigma.init = sig / sig,
          control = list(maxit = 500, tol = 1e-5))
```

```
## [[1]]
## [1] 0.3333333 0.3333333 0.3333333
##
## [[2]]
```

```
##              [,1]       [,2]       [,3]
## [1,]  0.3335660  0.3335660  0.3335660
## [2,] -0.4754645 -0.4754645 -0.4754645
##
## [[3]]
##          [,1]
## [1,] 3.001528
```

Thus,

$$\hat{\pi} = (0.3333333, 0.3333333, 0.3333333)^T$$

$$\hat{\beta} = \begin{pmatrix} 0.3335660 & 0.3335660 & 0.3335660 \\ -0.4754645 & -0.4754645 & -0.4754645 \end{pmatrix}$$

$$\hat{\sigma}^2 = 3.001528$$