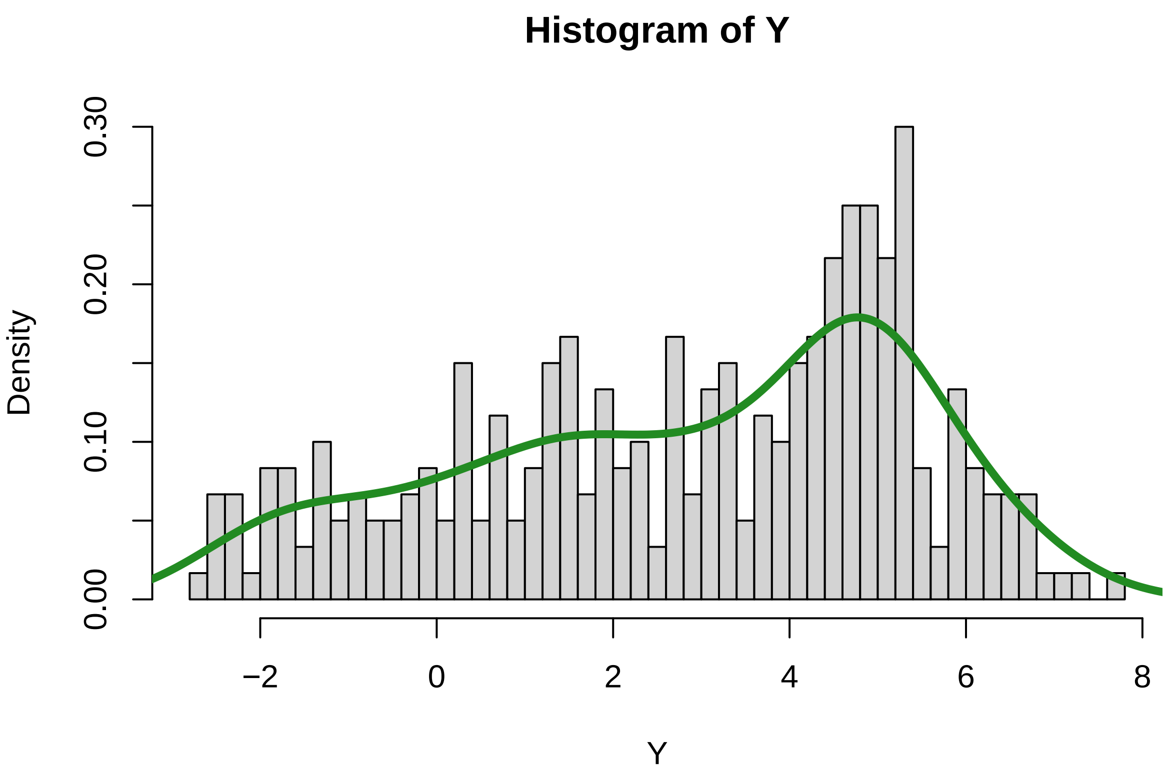
534 Homework 5 p.II

Michael Pena

2024-03-25

Part (a).

```r
Y = as.matrix(read.table('ExJ42.txt',header = T))
hist(Y,breaks = 40, prob = T)
# superimpose the density on it
den = density(Y[,1])
lines(den$x,den$y,col = 'forestgreen', lwd = 4)
```



part (b).

1. initialize iteration number at 1

1.1 begin while loop the closes when iteration is higher than max iteration or MRE is less that tolerr

2. define alpha,beta,μ_1,μ_2,μ_3,σ^2 using theta
3. define 3 density functions with the μ_i's that where just defined
4. (E-step) define posterior distributions for each density mixture

`Post_j = f1*PI[j]/sum(f1*PI[1]+f2*PI[2]+f3*PI[3])` where `PI` = (α, β, 1 - α - β) and `j = {1,2,3}` and f1,f2,f3 are the 3 densities defines in (3) $E[Z_{ij}]$ = Post_j

5. (M-step) find the new parameters of the maximized Q functions using…

$$\alpha = \frac{\sum_{i=0}^{N} E[Z_{i1}]}{N}$$

$$\beta = \frac{\sum_{i=0}^{N} E[Z_{i2}]}{N}$$

$$\mu_j = \frac{\sum_{i=0}^{N} E[Z_{ij}]x_i}{\sum_{i=0}^{N} E[Z_{ij}]}$$

$$\sigma^2 = \frac{\sum_{j=0}^{3} \sum_{i=0}^{N} E[Z_{ij}](x_i - \mu_j)^T(x_i - \mu_j)}{\sum_{j=0}^{3} \sum_{i=0}^{N} E[Z_{ij}]}$$

6. calculate log-likehood

6.1 calculate MRE

6.2 print iteration,loglikelihood,mre

7. add 1 to iteration number; set new theta back into old theta

7.1 close loop

7.2 return theta

7.3 print final parameters

part (c).

```
# lets build the algorirh in this chunk
EM_alg <- function(y,theta,maxit,tolerr){
  # initials
  N = length(y)
  it = 1
  theta1 <- theta
  mre = 1

  #print header
  header = paste0("iteration","        log-likelihood", "    MRE")
  print(header)

  # loop part
  while(it <= maxit && mre > tolerr){
    # initialize things again
    PI = c(theta[1],theta[2], 1-theta[2]-theta[1])
    mu1 = theta[3]
    mu2 = theta[4]
    mu3 = theta[5]
    var = theta[6]
    sig = sqrt(var)
    f1 = dnorm(y,mean = mu1, sd = sig)
    f2 = dnorm(y,mean = mu2, sd = sig)
    f3 = dnorm(y,mean = mu3, sd = sig)
    N1 = PI[1] * f1
```

```r
    N2 = PI[2] * f2
    N3 = PI[3] * f3
    D = N1+ N2 + N3
    Post1 = N1/D
    Post2 = N2/D
    Post3 = N3/D
    # find the alpha and beta
    theta1[1] = sum(Post1)/N
    theta1[2] = sum(Post2)/N
    # find the new mus
    theta1[3] = sum(Post1*y)/sum(Post1)
    theta1[4] = sum(Post2*y)/sum(Post2)
    theta1[5] = sum(Post3*y)/sum(Post3)
    # get the new variance
    nom  = 0

#    for(j in 1:3){
#      nom =  nom + sum(POST[,j] * (t(y - theta[j+2])%*%(y - theta[j+2]))[1])
#    }
    var = sum(Post1*(y-mu1)^2 + Post2*(y-mu2)^2 + Post3*(y - mu3)^2)/sum(Post1 + Post2 + Post3)
    theta1[6] = var
    # calculate likelihood
    ell = sum(Post1 * (log(f1) + log(PI[1])) + Post2*(log(f2) + log(PI[2])) + Post3*(log(f3)+ log(PI[3]

    # calculate MRE
    mre = max(abs(theta1 - theta) / abs(max(1,abs(theta1))))
    # print line
    print(sprintf('%2.0f           %12.5f       %.2e', it, ell, mre))

    # loop factors
    it = it + 1
    theta <- theta1
  }
  header2 = paste0("Alpha","       Beta", "        Mu_1", "       Mu_2", "       Mu_3", "    Variance")
  print(header2)
  print(theta)
  return(theta)
}

# run the function
data <-  Y[,1]
theta_i <- c(.3,.3,0,2,5,1)
EM_alg(data,theta_i,200,1e-06) -> theta_f
```

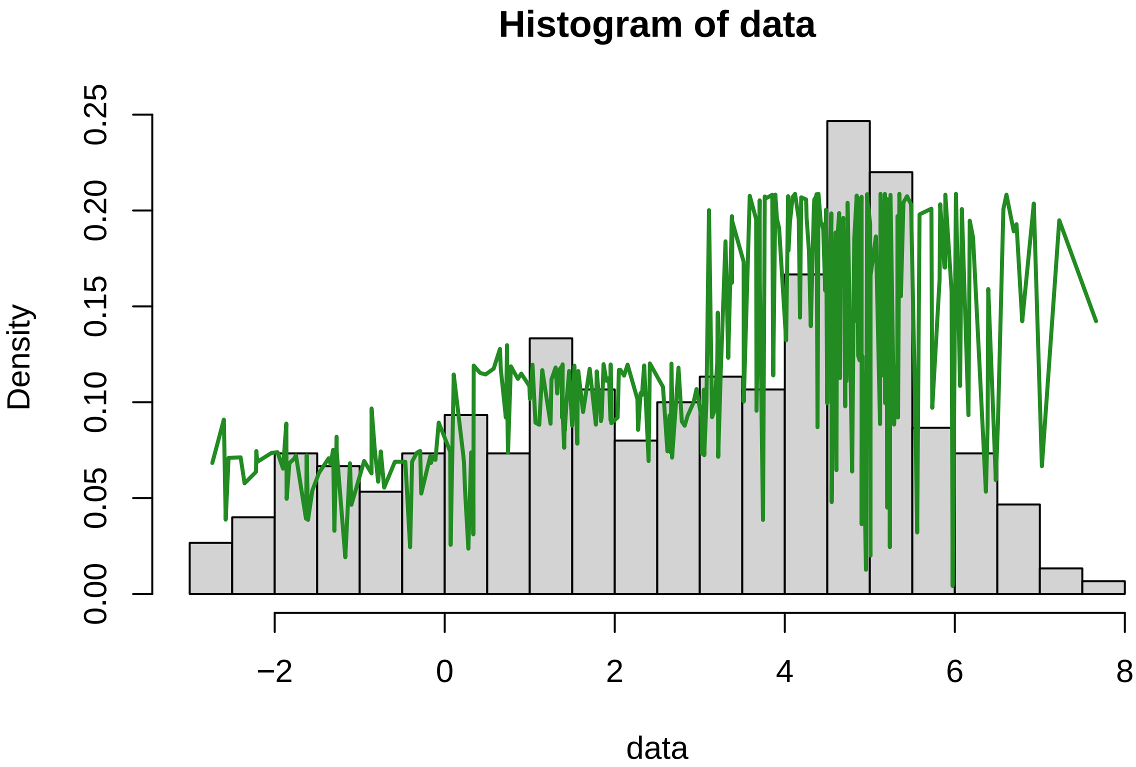
```
## [1] "iteration        log-likelihood    MRE"
## [1] " 1             -773.94765       9.93e-02"
## [1] " 2             -761.21555       2.36e-02"
## [1] " 3             -756.16474       1.54e-02"
## [1] " 4             -751.28046       1.32e-02"
## [1] " 5             -746.68105       1.18e-02"
## [1] " 6             -742.62491       1.04e-02"
## [1] " 7             -739.25281       9.09e-03"
## [1] " 8             -736.57989       7.74e-03"
## [1] " 9             -734.53691       6.45e-03"
```

```
## [1] "10              -733.01546       5.28e-03"
## [1] "11              -731.90151       4.27e-03"
## [1] "12              -731.09387       3.44e-03"
## [1] "13              -730.51082       2.80e-03"
## [1] "14              -730.09003       2.33e-03"
## [1] "15              -729.78556       1.93e-03"
## [1] "16              -729.56423       1.60e-03"
## [1] "17              -729.40236       1.33e-03"
## [1] "18              -729.28311       1.10e-03"
## [1] "19              -729.19458       9.15e-04"
## [1] "20              -729.12830       7.59e-04"
## [1] "21              -729.07825       6.29e-04"
## [1] "22              -729.04012       5.21e-04"
## [1] "23              -729.01083       4.32e-04"
## [1] "24              -728.98814       3.58e-04"
## [1] "25              -728.97043       2.96e-04"
## [1] "26              -728.95650       2.45e-04"
## [1] "27              -728.94548       2.03e-04"
## [1] "28              -728.93670       1.68e-04"
## [1] "29              -728.92966       1.39e-04"
## [1] "30              -728.92400       1.15e-04"
## [1] "31              -728.91943       9.54e-05"
## [1] "32              -728.91572       7.89e-05"
## [1] "33              -728.91270       6.54e-05"
## [1] "34              -728.91024       5.41e-05"
## [1] "35              -728.90823       4.48e-05"
## [1] "36              -728.90658       3.71e-05"
## [1] "37              -728.90522       3.07e-05"
## [1] "38              -728.90411       2.54e-05"
## [1] "39              -728.90319       2.10e-05"
## [1] "40              -728.90244       1.74e-05"
## [1] "41              -728.90182       1.44e-05"
## [1] "42              -728.90131       1.19e-05"
## [1] "43              -728.90088       9.86e-06"
## [1] "44              -728.90053       8.16e-06"
## [1] "45              -728.90024       6.75e-06"
## [1] "46              -728.90001       5.59e-06"
## [1] "47              -728.89981       4.63e-06"
## [1] "48              -728.89965       3.83e-06"
## [1] "49              -728.89951       3.17e-06"
## [1] "50              -728.89940       2.62e-06"
## [1] "51              -728.89931       2.17e-06"
## [1] "52              -728.89923       1.80e-06"
## [1] "53              -728.89917       1.49e-06"
## [1] "54              -728.89912       1.23e-06"
## [1] "55              -728.89907       1.02e-06"
## [1] "56              -728.89904       8.43e-07"
## [1] "Alpha       Beta        Mu_1       Mu_2       Mu_3    Variance"
## [1]  0.1808116  0.2954491 -1.0990768  1.6808201  4.8491651  1.0050523
```

part (d).

```r
# sort data
data_sort <- sort(data)
```

```r
# make variables
a <- theta_f[1]
b <- theta_f[2]
g <- 1 - a - b
mu1 <- theta_f[3]
mu2 <- theta_f[4]
mu3 <- theta_f[5]
sig <- theta_f[6]
f1 <- dnorm(data,mu1,sig)
f2 <- dnorm(data,mu2,sig)
f3 <- dnorm(data,mu3,sig)
# make mixture density
mix_den <- a*f1 + b*f2 + g*f3
# plot
hist(data,breaks = 30,prob = T)
lines(data_sort,mix_den,col = "forestgreen",lwd = 2)
```

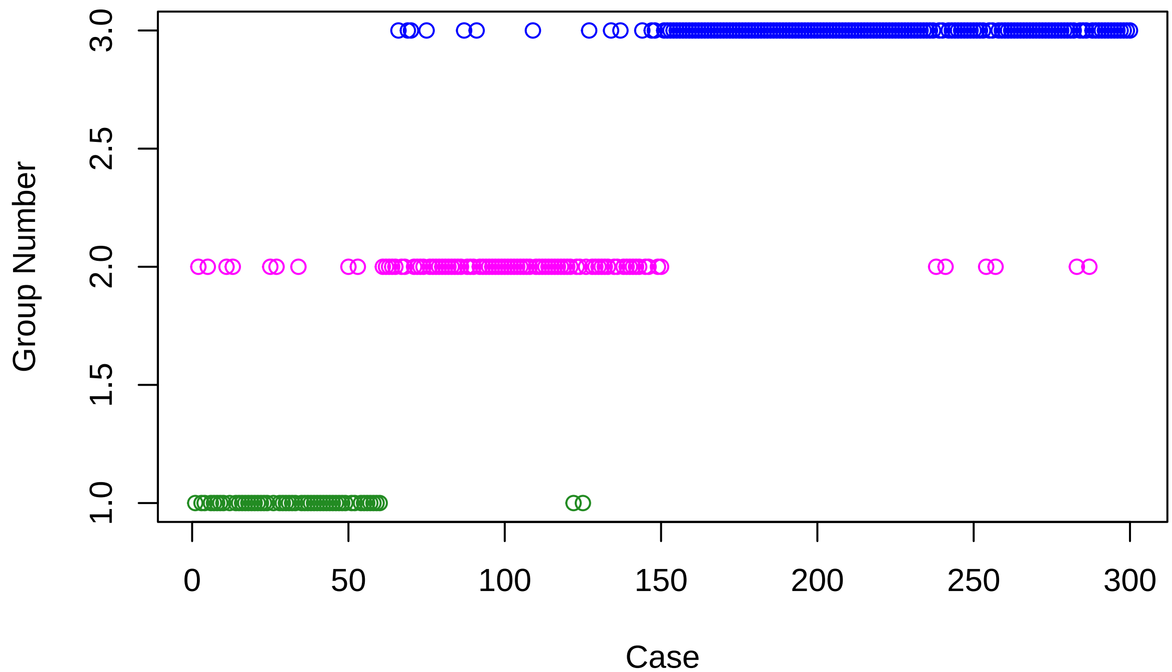


part (e).

```r
N = length(data)
cases = seq(1:N)
group = rep(0,length(cases))

for(i in 1:N){
  row <- data[i]
  f1 <- dnorm(row,mu1,sig)
  f2 <- dnorm(row,mu2,sig)
  f3 <- dnorm(row,mu3,sig)
  Post1 <- a*f1
```

```r
  Post2 <- b*f2
  Post3 <- g*f3
  PostSum <- Post1 + Post2 + Post3
  POST <- c(Post1,Post2,Post3) / PostSum
  N_group <- which.max(POST)
  group[i] <- N_group
}
color <- rep('charmander', N)
for(i in 1:N){
  N_group = group[i]
  if(N_group == 1){
    color[i] = "forestgreen"
  } else if(N_group == 2){
    color[i] = "magenta"
  } else {
    color[i] = 'blue'
  }
}
plot(cases, group, col = color, xlab = "Case", ylab = "Group Number")
```



As cases ascend, the group it belongs to will also. There are some outliers with group 2, but it's negligible.group 3 seems to be dominating more of the cases from points 150 to the end; group one seems to have the least amount of cases belonging to it ranging from 0 to around 60.