STAT GR5241 HW5_Q3_mjs2364

Mathieu Sauterey - UNI: MJS2364

April 22nd, 2018

Problem 3 - Multinomial Clustering (25 points)

```
# Reads histograms data from the file
H <- matrix(readBin("histograms.bin", "double", 640000), 40000, 16)
dim(H) # Expected dim = 40000x16
```

```
## [1] 40000    16
```

```
head(rowSums(H)) # Expected sums = 121
```

```
## [1] 121 121 121 121 121 121
```

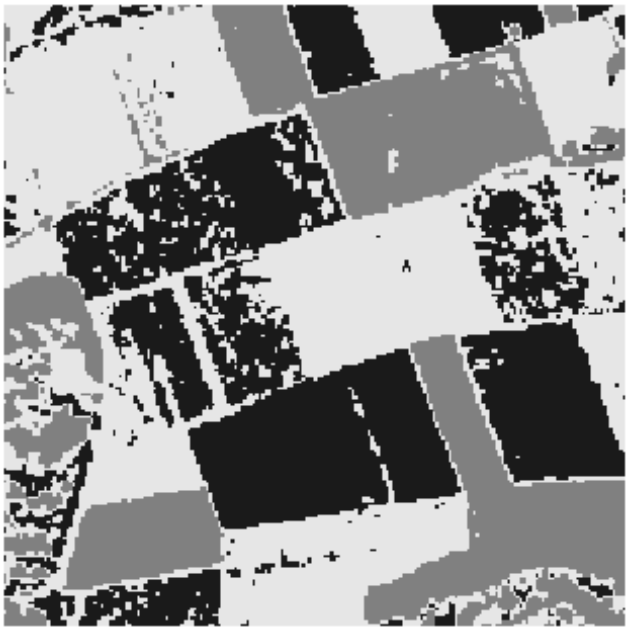
1. Implement the EM algorithm in R. The function call should be of the form m<-MultinomialEM(H,K,tau), with H the matrix of input histograms, K the number of clusters, and tau the threshold parameter.

```
# Implementation
MultinomialEM <- function(H,K,tau){

  set.seed(1)

  #install.packages("plyr")
  library("plyr")

  # Parameters + data normalization
  n <- nrow(H)
  p <- ncol(H)
  H <- H+0.01
  H <- scale(H, center = F)

  # Pre-allocation
  t <- matrix(NA, nrow = 5, ncol = 16)
  phi <- matrix(NA, nrow = n, ncol = K)
  a <- matrix(0, nrow = n, ncol = K)
  b <- matrix(NA, nrow = K, ncol = p)
```

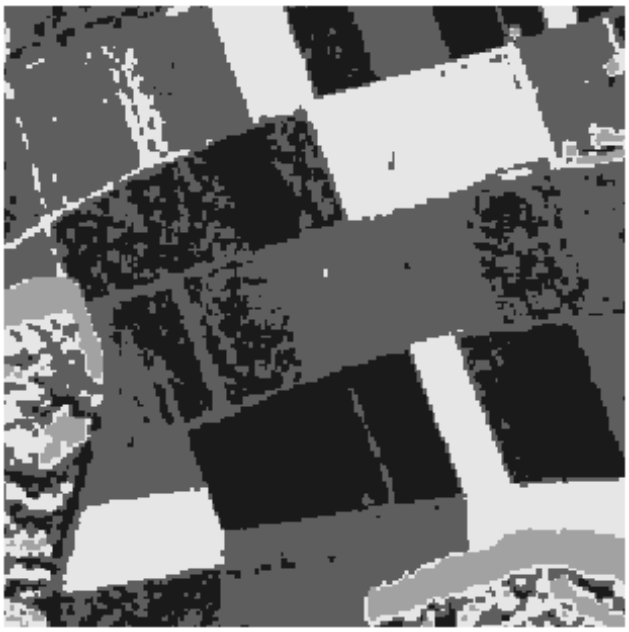
```
  # Initial inputs
  K_init <- sample((1:K), n, replace = T)
  t <- daply(data.frame(H),.(K_init),colMeans)
  c_weights <- rep(1/K,K)
  delta <- n


    while(delta > tau){

      # E Step
      phi <- exp(H %*% t(log(t)))
      a_new <- (phi %*% diag(c_weights)) / as.vector(phi %*% c_weights)

      # Assess measure of change of assignment
      delta <- norm( a_new - a ,"O")
      a <- a_new

      # M step
      c_weights <- colMeans(a)
      b <- t(a) %*% H
      t <- b / (rowSums(b))

    }

      # Soft to hard assignment
      m <- apply(a,1,which.max)
      return(m)
}
```

2. Run the algorithm on the input data for K=3, K=4 and K=5 and visualize the results as an image.

```
# Rotates the image to be displayed in the correct orientation
rotation <- function(x) {(apply(x, 2, rev))} # Source: R-bloggers

# Changes margins
par(oma=c(0,0,0,0)+1)
par(mar=c(0,0,0,0)+1)

# First simulation with 3 clusters and tau = 0.1 (axes are removed and A/R=1)
Sim1 <- MultinomialEM(H, K=3, tau=0.1)
pix1 <- matrix(Sim1,sqrt(length(Sim1)),sqrt(length(Sim1)), byrow = T)
image(t(rotation(pix1)), col = grey(seq(0.1,0.9, length.out = 3)),
      asp=1, xaxt='n', yaxt='n',bty="n")
```



```
# First simulation with 4 clusters and tau = 0.1 (axes are removed and A/R=1)
Sim2 <- MultinomialEM(H, K=4, tau=0.1)
pix2 <- matrix(Sim2,sqrt(length(Sim1)),sqrt(length(Sim1)), byrow = T)
image(t(rotation(pix2)), col = grey(seq(0.1,0.9, length.out = 4)),
      asp=1, xaxt='n', yaxt='n',bty="n")
```



```r
# First simulation with 5 clusters and tau = 0.1 (axes are removed and A/R=1)
Sim3 <- MultinomialEM(H, K=5, tau=0.1)
pix3 <- matrix(Sim3,sqrt(length(Sim1)),sqrt(length(Sim1)), byrow = T)
image(t(rotation(pix3)), col = grey(seq(0.1,0.9, length.out = 5)),
      asp=1, xaxt='n', yaxt='n',bty="n")
```