STAT W4400 Multinomial EM

Yanjin Li

April 11, 2016

0. Data loading

```
H <- matrix(readBin("/Users/yanjin1993/Documents/Academy/Columbia University /2016 Spring /STAT 4400 Me
```

1. EM algorithm Implementation:

- H: the matrix of input histograms
- K: the number of clusters
- tau: threshold parameter
- Outputs:
- m: hard assignment vector for visualization

```
MultinomialEM <- function(H,K,tau){
  p <- ncol(H) # Number of bins per histogram
  n <- nrow(H) # Number of histograms
  H[H==0] <- 0.01

  # Randomly assign centriods w/t replacement:
  centroids <- sample(c(1:n), size=K)
  t <- t(apply(H[centroids,], 1, function(row){row/sum(row)}))
  # t: centroids

  # Parameters assignment:
  a <- matrix(0, ncol=K, nrow=n) # a: assignment probabilities
  phi <- matrix(0, ncol=K, nrow=n) # phi:  partial elements
  b <- matrix(0, ncol=K, nrow=1)
  cp <- matrix(1.0/K, ncol=1, nrow=K) # cp: mixture weights
  m <- matrix(1.0/K, ncol=1, nrow=n) # m: hard assignments
  delta <- 9999 # delta: change of assignments

  while(delta > tau){
    a1 <- a
    for(k in 1:K){
      # E-step:
      for(i in 1:n) {
        phi[i, k] = exp(sum(H[i,] * log(t[k,])))
        a[i, k] = cp[k] * phi[i, k] / sum(cp * phi[i,])
      }
      a[is.nan(a)] = 0

      # M-step:
      cp[k] = sum(a[,k]) / n
      b = a[,k] %*% H
      t[k,] = b / sum(b)
    }
    delta = norm(a1 - a, "O")
```

```
    #print(delta)
  }
  for( i in 1:n){
    m[i,1] <- which.max(a[i,])
  }
  #m = sapply(c(1:n), function(i) {which.max(a[i,])})
  return(m)
}
```

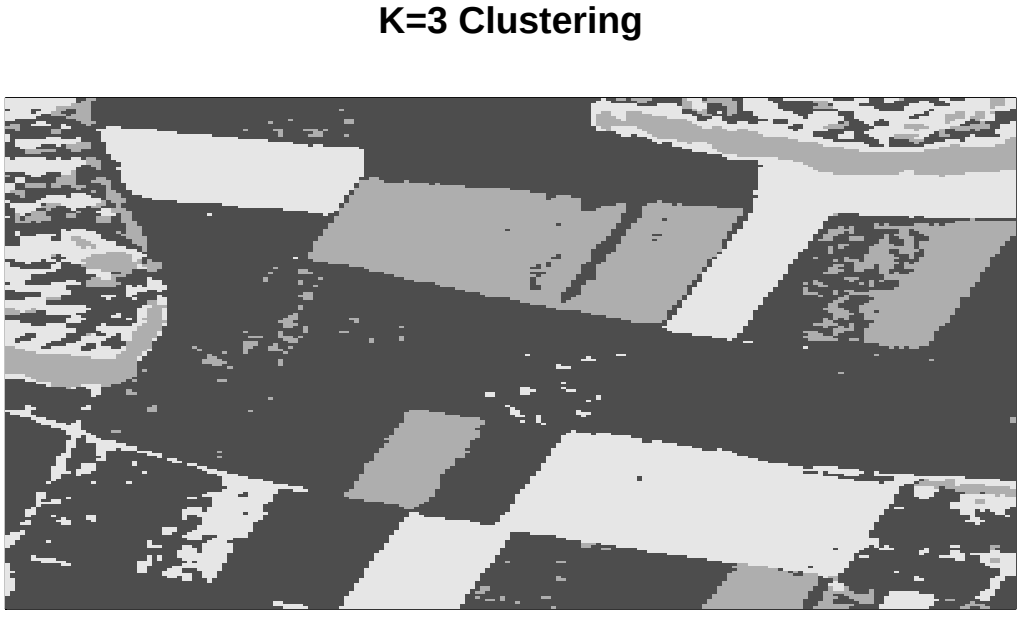
2. Algroithm training process:

```
l1 <- c(0.1)
l2 <- c(3, 4, 5)
count <- 1
M <- data.frame(matrix(NA, nrow = 40000, ncol = 0))

for(i in l1){
  for(j in l2){
    result <- MultinomialEM(H, j, i)
    result <- as.matrix(result)
    M <- cbind(M, result)
  }
}
colnames(M) <- c("K=3", "K=4", "K=5")
```

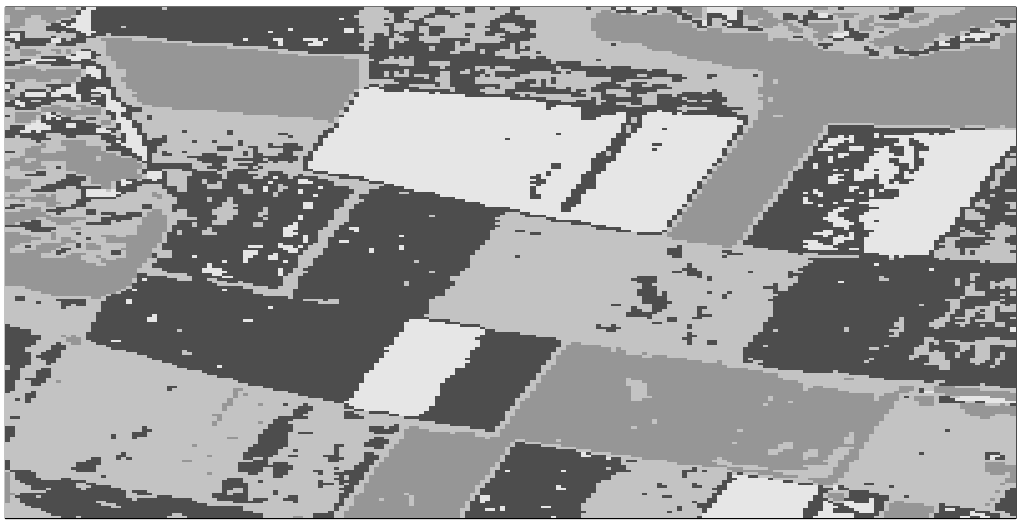
3. Hard Assigment Visualization:

```
m_matrix1 <- matrix(M[,1], nrow=200, ncol=200)
image(m_matrix1, col=gray.colors(3), xaxt="n", yaxt="n", main = "K=3 Clustering")
```




```
m_matrix2 <- matrix(M[,2], nrow=200, ncol=200)
image(m_matrix2, col=gray.colors(4), xaxt="n", yaxt="n", main = "K=4 Clustering")
```

K=4 Clustering



```
m_matrix3 <- matrix(M[,3], nrow=200, ncol=200)
image(m_matrix3, col=gray.colors(5), xaxt="n", yaxt="n", main = "K=5 Clustering")
```

K=5 Clustering