Math 611 HW5

Jeff Gould

10/3/2020

1)

a) Write down an expression for $Q(\theta, \theta')$ using $r_{i1} = P(z_i = 1 \mid \hat{X}_i, \theta)$, $r_{i2} = P(z_i = 2 \mid \hat{X}_i, \theta)$ and the pdfs of the normals. Then give a formula for r_{i1} and r_{i2} in terms of θ and the $\hat{X}_i$

$Q(\theta, \theta') = \sum_{i=1}^N E_\theta[\log P(\hat{X}_i, z_i \mid \theta')] =$

$P(X, z_i|\theta) = \prod_{i=1}^N p_1^{z_{i1}} \mathcal{N}(X_i|\mu_1, \sigma_1^2)^{z_{i1}} \cdot p_2^{z_{i2}} \mathcal{N}(X_i|\mu_2, \sigma_2^2)^{z_{i2}} \Rightarrow$

$\log P(X, z_i|\theta) = \sum_{i=1}^N z_{i1}[\log p_1 + \log \mathcal{N}(X_i|\mu_1, \sigma_1^2)] + z_{i2}[\log p_2 + \log \mathcal{N}(X_i|\mu_2, \sigma_2^2)]$

$\sum_{i=1}^N E_\theta[\log P(\hat{X}_i, z_i \mid \theta')] = \sum_{i=1}^N P(z_i = 1)[\log p_1 + \log \mathcal{N}(X_i|\mu_1, \sigma_1^2)] + P(z_i = 2)[\log p_2 + \log \mathcal{N}(X_i|\mu_2, \sigma_2^2)] =$

$\sum_{i=1}^N r_{i1}[\log p_1 + \log \mathcal{N}(X_i|\mu_1, \sigma_1^2)] + r_{i2}[\log p_2 + \log \mathcal{N}(X_i|\mu_2, \sigma_2^2)] =$

$\sum_{i=1}^N r_{i1}[\log p_1 + \log \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-(X_i-\mu_1)^2/(2\sigma_1^2)}] + r_{i2}[\log p_2 + \log \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-(X_i-\mu_1)^2/(2\sigma_1^2)}] =$

$\sum_{i=1}^N r_{i1}[\log p_1 + -(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(2\pi\sigma_1^2)] + r_{i2}[\log p_2 + -(X_i - \mu_2)^2/(2\sigma_2^2) - \frac{1}{2}\log(2\pi\sigma_2^2)]$

$r_{i1} = \frac{p_1 \mathcal{N}(X_i|\mu_1, \sigma_1^2)}{p_1 \mathcal{N}(X_i|\mu_1, \sigma_1^2) + p_2 \mathcal{N}(X_i|\mu_2, \sigma_2^2)}$

$r_{i2} = \frac{p_2 \mathcal{N}(X_i|\mu_2, \sigma_2^2)}{p_1 \mathcal{N}(X_i|\mu_1, \sigma_1^2) + p_2 \mathcal{N}(X_i|\mu_2, \sigma_2^2)}$

b) Compute $\text{argmax}_{\theta'} Q(\theta, \theta')$

$\frac{\partial}{\partial p_1} \sum_{i=1}^N r_{i1}[\log p_1 + -(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(2\pi\sigma_1^2)] + r_{i2}[\log p_2 + -(X_i - \mu_2)^2/(2\sigma_2^2) - \frac{1}{2}\log(2\pi\sigma_2^2)] =$

$\frac{\partial}{\partial p_1} \sum_{i=1}^N r_{i1} \log p_1 + r_{i2} \log(1 - p_1) = \frac{\sum r_{i1}}{p_1} + \frac{\sum r_{i2}}{1-p_1} = \frac{(1-p_1)\sum r_{i1}}{p_1(1-p_1)} + \frac{p_1 \sum r_{i2}}{p_1(1-p_1)} = 0 \rightarrow$

$(1 - p_1)\sum r_{i1} + p_1 \sum r_{i2} = 0 \rightarrow p_1 = \frac{\sum r_{i1}}{\sum(r_{i1}+r_{i2})} = \frac{1}{N}\sum r_{i1}$, and by symmetry $p_2 = \frac{\sum r_{i2}}{\sum(r_{i1}+r_{i2})} = \frac{1}{N}\sum r_{i2}$

$\frac{\partial}{\partial \mu_1} \sum_{i=1}^N r_{i1}[\log p_1 + -(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(2\pi\sigma_1^2)] + r_{i2}[\log p_2 + -(X_i - \mu_2)^2/(2\sigma_2^2) - \frac{1}{2}\log(2\pi\sigma_2^2)] =$

$\frac{\partial}{\partial \mu_1} \sum_{i=1}^N r_{i1}[-(X_i - \mu_1)^2/(2\sigma_1^2)] = \sum_{i=1}^N r_{i1}(X_i - \mu_1)/\sigma_1^2 = 0 \rightarrow \sum_{i=1}^N r_{i1}(X_i - \mu_1) = 0$

$\sum_{i=1}^N r_{i1}(X_i - \mu_1) = \sum_{i=1}^N (r_{i1}X_i - r_{i1}\mu_1) = \sum_{i=1}^N r_{i1}X_i - \mu_1 \sum_{i=1}^N r_{i1} = 0 \rightarrow \sum_{i=1}^N r_{i1}X_i = \mu_1 \sum_{i=1}^N r_{i1} \rightarrow \mu_1 = \frac{\sum_{i=1}^N r_{i1}X_i}{\sum_{i=1}^N r_{i1}}$

By symmetry, solving $\frac{\partial}{\partial \mu_2} \sum_{i=1}^N r_{i1}[\log p_1 + -(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(2\pi\sigma_1^2)] + r_{i2}[\log p_2 + -(X_i - \mu_2)^2/(2\sigma_2^2) - \frac{1}{2}\log(2\pi\sigma_2^2)]$ for μ_2, we get

$\mu_2 = \frac{\sum_{i=1}^N r_{i2}X_i}{\sum_{i=1}^N r_{i2}}$

$\frac{\partial}{\partial \sigma_1^2} \sum_{i=1}^N r_{i1}[\log p_1 + -(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(2\pi\sigma_1^2)] + r_{i2}[\log p_2 + -(X_i - \mu_2)^2/(2\sigma_2^2) - \frac{1}{2}\log(2\pi\sigma_2^2)] =$

$\frac{\partial}{\partial \sigma_1^2} \sum_{i=1}^N r_{i1}[-(X_i - \mu_1)^2/(2\sigma_1^2) - \frac{1}{2}\log(\sigma_1^2)] = \sum_{i=1}^N r_{i1}\left(\frac{(X_i-\mu_1)^2}{2(\sigma_1^2)^2} - \frac{1}{2\sigma_1^2}\right) = \sum_{i=1}^N r_{i1}\left(\frac{(X_i-\mu_1)^2}{2(\sigma_1^2)^2} - \frac{\sigma_1^2}{2(\sigma_1^2)^2}\right) = 0 \rightarrow$

$\sum_{i=1}^N r_{i1}\left((X_i - \mu_1)^2 - \sigma_1^2\right) = 0 \rightarrow \sum_{i=1}^N r_{i1}(X_i - \mu_1)^2 = \sum_{i=1}^N r_{i1}\sigma_1^2 \Rightarrow$

$\sigma_1^2 = \frac{\sum_{i=1}^N r_{i1}(X_i-\mu_1)^2}{\sum_{i=1}^N r_{i1}}$

Again, by symmetry, $\sigma_2^2 = \frac{\sum_{i=1}^N r_{i2}(X_i-\mu_2)^2}{\sum_{i=1}^N r_{i2}}$

c) Compare your updates for the parameters to the heuristic updates of soft EM

2)

a)

E-Step: Given $X, \theta = (\mu^{(1)}, \mu^{(2)}, p_1, p_2)$, and $p_1 + p_2 = 1$. If applying soft EM, calculate the probabily each X_i is in the first or second distribution:

$z'_{1i} = p_1 P(X_i|\mu_1) = p_1 \prod_{d=1}^{10} \mu_{1d}^{X_{id}}(1 - \mu_{1d})^{(1-X_{id})}$

$z'_{2i} = p_2 P(X_i|\mu_2) = p_2 \prod_{d=1}^{10} \mu_{2d}^{X_{id}}(1 - \mu_{2d})^{(1-X_{id})}$

$z_{1i} = \frac{z'_{1i}}{z'_{1i}+z'_{2i}}$

$z_{2i} = \frac{z'_{2i}}{z'_{1i}+z'_{2i}}$

M-Step: Given $X, Z = (z_1 z_2)$, calculate $\theta = (\mu^{(1)}, \mu^{(2)}, p_1, p_2)$

$M1 = \sum z_1, M2 = \sum z_2$

$\mu_1 = \frac{1}{M1}\sum z_{i1}X_i$

$\mu_2 = \frac{1}{M2}\sum z_{i2}X_i$

$p_1 = \frac{M1}{M1+M2}$

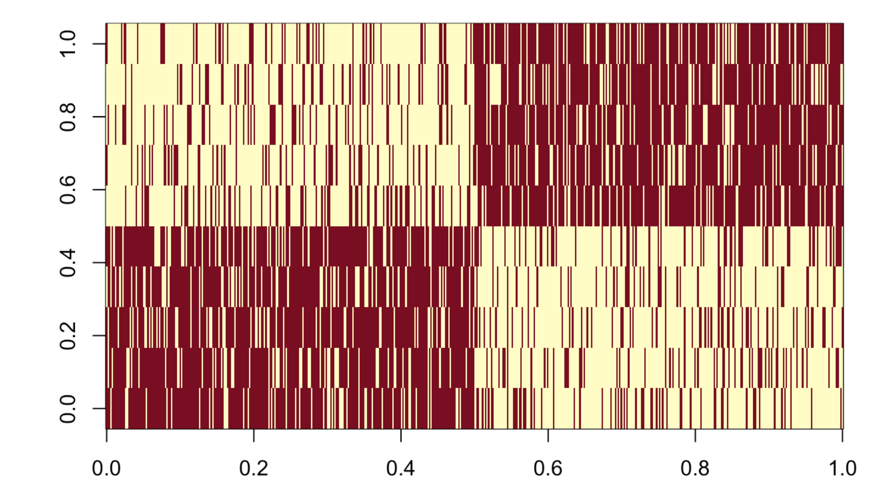
$p_2 = \frac{M2}{M1+M2}$

log-likelihood:

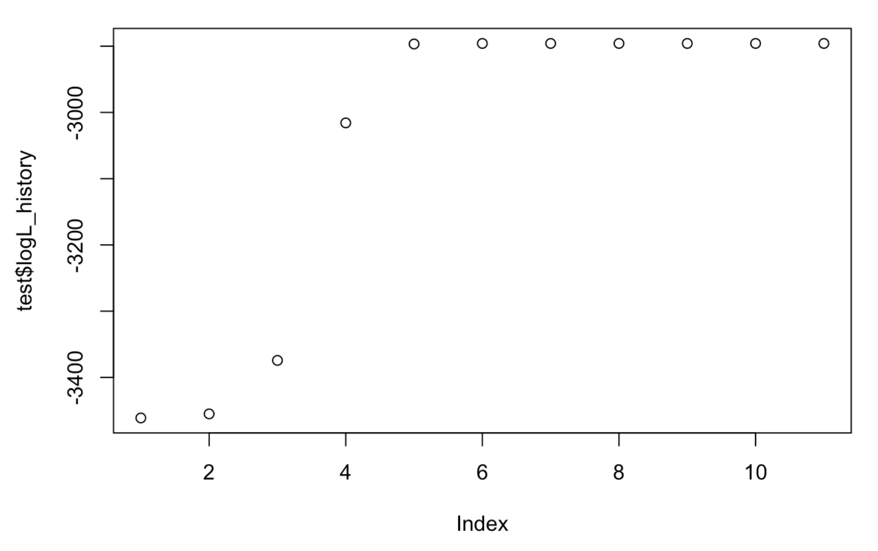
$\log p(X|\theta) = \sum_{i=1}^N \log(p_1 P(X_i|\mu_1) + p_2 P(X_i|\mu_2))$

b)

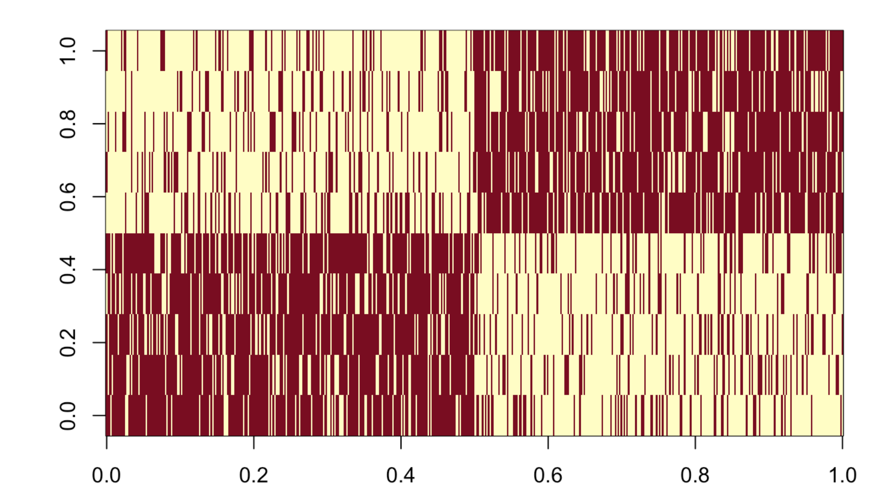
```
noisy_bits <- read_csv("noisy_bits.csv")

noisy_bits %>% as.matrix() %>% image()
```



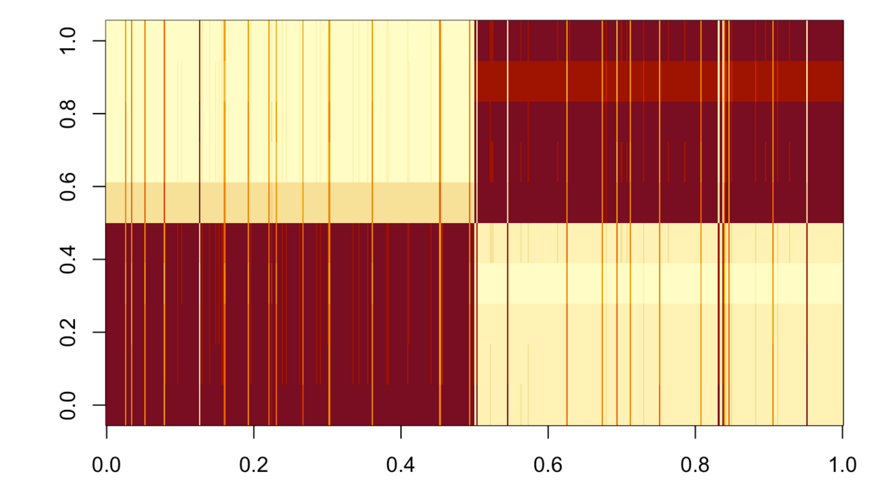
```
e_step_soft <- function(x, theta){

  mu1 <- theta$Mu[1, ]
  mu2 <- theta$Mu[2, ]

  p1 <- theta$Pi[1]
  p2 <- theta$Pi[2]

  z1 <- p1 * (apply(X, 1, function(x)mapply(dbinom, x, mu1, size = 1) %>% prod()))

  z2 <- p2 * (apply(X, 1, function(x)mapply(dbinom, x, mu2, size = 1) %>% prod()))

  Z <- cbind(z1, z2) / rowSums(cbind(z1, z2))

  return(Z)

}



m_step_soft <- function(X, Z){

  M1 <- sum(Z[ ,1])
  M2 <- sum(Z[ ,2])

  mu1 <- 1/M1 * colSums(Z[,1] * X)
  mu2 <- 1/M2 * colSums(Z[,2] * X)

  obs <- nrow(X)

  Pi <- c(M1, M2) / sum(M1, M2)

  Mu <- rbind(mu1, mu2)

  theta <- list(Mu = Mu,  Pi = Pi)

  return(theta)

}

logLike <- function(X, theta){

  mu1 <- theta$Mu[1, ]
  mu2 <- theta$Mu[2, ]

  p1 <- theta$Pi[1]
  p2 <- theta$Pi[2]

  comp1 <- p1 * (apply(X, 1, function(x)mapply(dbinom, x, mu1, size = 1) %>% prod()))
  comp2 <- p2 * (apply(X, 1, function(x)mapply(dbinom, x, mu2, size = 1) %>% prod()))

  return(sum(log(comp1 + comp2)))

}

EM_soft <- function(X, init_Z, max_iter = 50, epsilon = 1e-3){

  for (i in 1:max_iter) {
    if (i == 1) {
      # Initialization
      m.step <- m_step_soft(X, init_Z)
      e.step <- e_step_soft(X, m.step)
      cur.loglik <- logLike(X, m.step)
      loglik.vector <- cur.loglik
    } else {
      # Repeat E and M steps till convergence
      m.step <- m_step_soft(X, e.step)
      e.step <- e_step_soft(X, m.step)

      cur.loglik <- logLike(X, m.step)
      loglik.vector <- c(loglik.vector, cur.loglik)

      loglik.diff <- abs(cur.loglik - loglik.vector[i-1])
      if(loglik.diff < epsilon & i > 10) {
        break
      }
    }
  }

  return(list(
    theta = m.step,
    logL_history = loglik.vector,
    probs = e.step
  ))
}

X <- noisy_bits %>% as.matrix()

init_Zs <-  matrix(runif(1000), ncol = 2)
init_Zs <- init_Zs / rowSums(init_Zs)
init_Z <- init_Zs

test <- EM_soft(X, init_Zs)
plot(test$logL_history)
```




```
image(X)
```



```
X_result <- test[["probs"]][,1] * matrix(rep(test$theta$Mu[1,], 500), nrow = 500, byrow = T) + test[["probs"]][,2
] * matrix(rep(test$theta$Mu[2,], 500), nrow = 500, byrow = T)
image(X_result)
```



```
test$theta
```

```
## $Mu
##             V1        V2        V3        V4        V5        V6        V7
## mu1 0.2250306 0.2210770 0.2173989 0.1614023 0.2386598 0.8141468 0.7926998
## mu2 0.8090003 0.7851571 0.7808418 0.7683933 0.7758493 0.2793200 0.2053043
##             V8        V9       V10
## mu1 0.8121408 0.7537884 0.7926516
## mu2 0.2060544 0.2077925 0.1855540
##
## $Pi
## [1] 0.4948892 0.5051108
```

We see that we did a good job of finding the true distribution, as $p_1 = p_2 = 0.5$, $\mu_1 = (0.8, 0.8, 0.8, 0.8, 0.8, 0.2, 0.2, 0.2, 0.2, 0.2)$, and $\mu_2 = (0.2, 0.2, 0.2, 0.2, 0.2, 0.8, 0.8, 0.8, 0.8, 0.8)$

Correct classifications:

```
mean(test[["probs"]][1:250,2] > 0.5)
```

```
## [1] 0.976
```

```
mean(test[["probs"]][251:500,1] > 0.5)
```

```
## [1] 0.956
```

3

a) Let $a, b \in \mathbb{R}^n$ be column vectors. Show in any way you like - by proof, through example, by intuitive explanation - that $(a \cdot b)^2 = a^T M a$ where M is an $n \times n$ matrix given by $M = bb^T$.

$(a \cdot b)^2 = (a \cdot b)(a \cdot b) = (a \cdot b)(b \cdot a) = (a \cdot b)(b^T a) = a \cdot bb^T a = a \cdot Ma = a^T Ma$, where the last step follows because M is symmetric

b) Let $\hat{\Sigma}$ be the covariance matrix of the data. Then, by definition

$$\hat{\Sigma}_{jk} = \frac{1}{N}\sum_{i=1}^N (X_j^{(i)} - \mu_j)(X_k^{(i)} - \mu_k)$$

Show that $\hat{\Sigma}$ can also be written in the following two forms

- Let $\tilde{X}$ be the $N \times n$ matrix with the $X^{(i)} - \mu$ as the rows

$$\hat{\Sigma} = \frac{1}{N}\tilde{X}^T\tilde{X}$$

Let C be the $n \times n$ matrix formed by $\tilde{X}^T\tilde{X}$. Then the c_{jk} entry in C is

$\tilde{x}_{j1}^T\tilde{x}_{1k} + \tilde{x^T}_{j2}\tilde{x}_{2k} + \dots + \tilde{x}_{jN}^T\tilde{x}_{Nk} = \sum_{i=1}^N \tilde{X}_{ji}^T\tilde{X}_{ik} = \sum_{i=1}^N (X_j^{(i)} - \mu_j)(X_k^{(i)} - \mu_k)$

$\frac{1}{N}\tilde{X}^T\tilde{X} = \frac{1}{N}C$, so each entry in $\frac{1}{N}C = \frac{1}{N}\sum_{i=1}^N (X_j^{(i)} - \mu_j)(X_k^{(i)} - \mu_k)$ which is $\hat{\Sigma}$

- Thinking of the $X^{(i)}$ as column vectors,

$$\hat{\Sigma} = \frac{1}{N}\sum_{i=1}^N (X^{(i)} - \mu)(X^{(i)} - \mu)^T$$

For each $(X^{(i)} - \mu)(X^{(i)} - \mu)^T$ we get an $n \times n$ matrix. The jk^{th} entry in each matrix is $(X_j^{(i)} - \mu_j)(X_k^{(i)} - \mu_k)$

Then when we some up the matrices, the result is that each jk^{th} entry in the resulting matrix is

$\sum_{i=1}^N (X_j^{(i)} - \mu_j)(X_k^{(i)} - \mu_k)$, and divide by N to get $\hat{\Sigma}$.

See attached scan for a (very) rough outline.

c)

The 1-d PCA involves the parameters μ, $w^{(1)}$ and $c_i \in \mathbb{R}$ for $i = 1, 2, \dots, N$ that are used to approximate $X^{(i)}$ according to

$$X^{(i)} \approx \mu + c_i w^{(1)}.$$

Derive the values of c_i and $w^{(1)}$ that optimize this approximation. (We did this in class.) Then, compute the mean and variance of the c_i

$l(w^{(1)}, c_1, \dots, c_N) = \sum_{i=1}^N ||\hat{X}^{(i)} - \mu - c_i w^{(1)}||^2 \rightarrow$

$\frac{\partial l}{\partial c_k} = \frac{\partial}{\partial c_k}||\hat{X}^{(k)} - \mu - c_k w^{(1)}||^2 = \frac{\partial}{\partial c_k}\left[(\hat{X}^{(k)} - \mu - c_k w^{(1)}) \cdot (\hat{X}^{(k)} - \mu - c_k w^{(1)})\right] = -2w^{(1)} \cdot (X^{(k)} - \mu - c_k w^{(1)}) = 0 \rightarrow$

$(X^{(k)} - \mu - c_k w^{(1)}) = 0 \rightarrow X^{(k)} = \mu + c_k w^{(1)}$

- compute the mean and variance of $C = (c_1, c_2, \dots c_N)$

$X^{(i)} = \mu + c_i w^{(1)} \rightarrow c_i = \frac{X^{(i)} - \mu}{w^{(1)}}$

$E[C] = E\left[\frac{X-\mu}{w^{(1)}}\right] = 0$

$Var(C) = Var\left[\frac{X-\mu}{w^{(1)}}\right] = \frac{1}{w^{(i)2}}Var[X - \mu] = \frac{1}{w^{(i)2}}Var(X)$

$$\frac{1}{N}\sum (x^{(i)}-\mu)(x^{(i)}-\mu)^T$$

$n\times 1$ $\quad 1\times n$ $\quad \to n\times n$

$$(x^{(1)}-\mu)(x^{(1)}-\mu)^T$$

$$\begin{pmatrix} (x_{11}-\mu_1)(x_{11}-\mu_1) & (x_{11}-\mu_1)(x_{12}-\mu_2) & \cdots & (x_{11}-\mu_1)(x_{1n}-\mu_n) \\ (x_{21}-\mu_1)(x_{21}-\mu_2) & & & \\ & & & \\ & & & (x_{n1}-\mu_n)(x_{nn}-\mu_n) \end{pmatrix}$$