

# TD MAD - EM Algorithm

Piseth KHENG, Borachhun YOU

24 October 2022

## Simulation

1. Simulate a sample of  $n = 100$  observations of Poisson distribution with parameter  $\lambda = 3$

```
poisson3 <- rpois(n=100, lambda=3)
```

2. Simulate a sample of  $n = 200$  observations of Poisson distribution with parameter  $\lambda = 15$

```
poisson15 <- rpois(n=200, lambda=15)
```

3. Create a vector of 100 “1” followed by 200 “2”

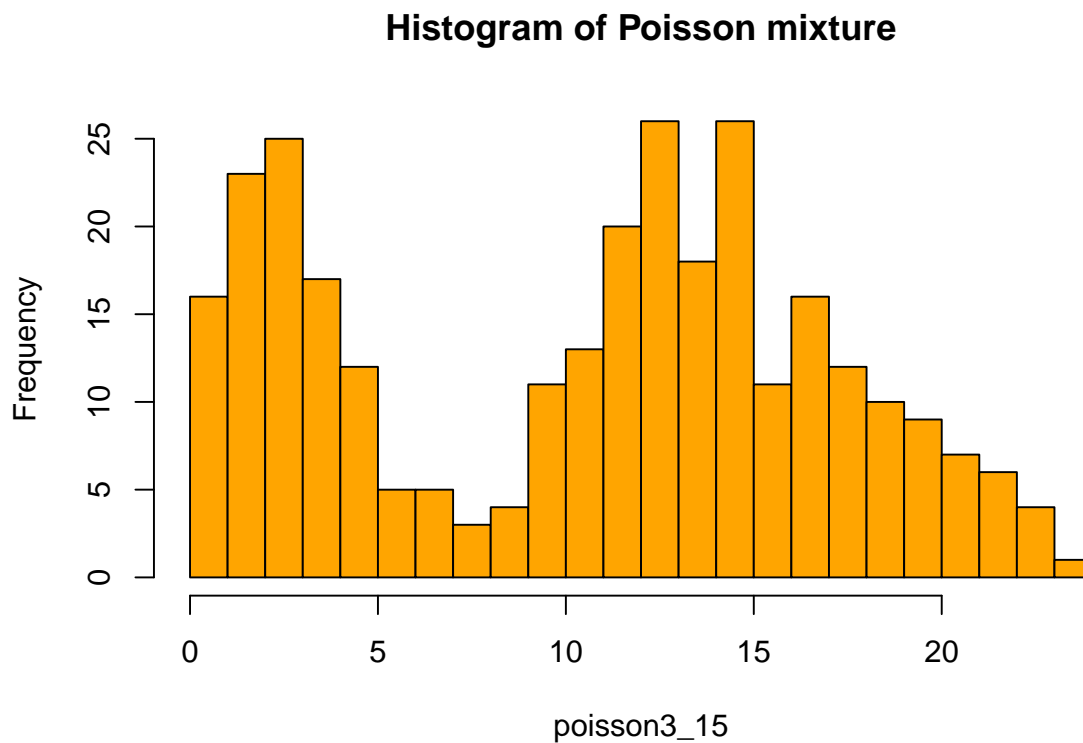
```
true_class <- rep(c(1,2), c(100,200))
```

4. Simulate a mixture of two Poisson distribution

$$\mathbb{P}(x) = \pi_1 \frac{e^{-\lambda_1}}{x!} \lambda_1^x + \pi_2 \frac{e^{-\lambda_2}}{x!} \lambda_2^x$$

with  $\lambda_1 = 3$  and  $\lambda_2 = 15$ ,  $\pi_1 = \frac{1}{3}$

```
poisson3_15 <- c(poisson3, poisson15)
hist(poisson3_15, breaks=30, col="orange", main="Histogram of Poisson mixture")
```



## EM algorithm for Poisson mixture model of $K$ components

Poisson distribution of  $K$  components:

$$\mathbb{P}(x) = \sum_{k=1}^K \pi_k \mathbb{P}_k(x) = \sum_{k=1}^K \pi_k \frac{e^{-\lambda_k} \lambda_k^x}{x!}$$

We have the observations  $X = (x_1, \dots, x_n)$ , and the missing data  $Z = (z_1, \dots, z_n)$ .

The objective is to estimate the parameters:

$$\theta = \{\pi_1, \dots, \pi_K, \lambda_1, \dots, \lambda_K\}$$

### 1. Initialization

We take  $\pi_k = \frac{1}{K}$  and  $\lambda_k = k, \forall k \in \llbracket 1, K \rrbracket$

```
EM_init_poisson <- function(K) {
  pi <- rep(1/K, times=K)
  lambda <- 1:K
  return(c(pi, lambda))
}
```

### 2. E step

At iteration  $q$ , E step computes  $Q(\theta|\theta^{(q)})$

$$\begin{aligned}
Q(\theta|\theta^{(q)}) &= \mathbb{E}_{Z|X;\theta^{(q)}} [\log \mathbb{P}(X, Z; \theta)] = \mathbb{E}_{Z|X;\theta^{(q)}} \left[ \log \prod_{i=1}^n \mathbb{P}(x_i, z_i; \theta) \right] \\
&= \sum_{i=1}^n \mathbb{E}_{z_i|x_i;\theta^{(q)}} [\log (\mathbb{P}(z_i) \times \mathbb{P}(x_i|z_i; \theta))]
\end{aligned}$$

Let  $z_{ik} = 1_{\{z_i=k\}}$

$$\begin{aligned}
Q(\theta|\theta^{(q)}) &= \sum_{i=1}^n \mathbb{E}_{z_i|x_i;\theta^{(q)}} \left[ \sum_{k=1}^K z_{ik} \log (\mathbb{P}(z_i = k) \times \mathbb{P}(x_i|z_i = k; \theta)) \right] \\
&= \sum_{i=1}^n \sum_{k=1}^K \mathbb{P}(z_i = k|x_i; \theta^{(q)}) \log (\mathbb{P}(z_i = k) \times \mathbb{P}(x_i|z_i = k; \theta)) \\
&= \sum_{i=1}^n \sum_{k=1}^K \mathbb{P}(z_i = k|x_i; \theta^{(q)}) \log (\pi_k \times \mathbb{P}_k(x_i; \lambda_k))
\end{aligned}$$

Let

$$t_{ik}^{(q)} = \mathbb{P}(z_i = k|x_i; \theta^{(q)}) = \frac{\pi_k^{(q)} \mathbb{P}_k(x_i; \lambda_k^{(q)})}{\sum_{m=1}^K \pi_m^{(q)} \mathbb{P}_m(x_i; \lambda_m^{(q)})}$$

At each iteration  $q$ , we only need to compute  $t_{ik}^{(q)}, i \in \llbracket 1, n \rrbracket, k \in \llbracket 1, K \rrbracket$

```

# theta : c(pi_k, lambda_k)
E_step_poisson <- function(X, K, theta) {
  pi <- theta[1:K]
  lambda <- theta[(K+1):(2*K)]
  t <- matrix(0, nrow=length(X), ncol=K)
  for (i in 1:length(X)) {
    for (k in 1:K) {
      t[i,k] <- (pi[k] * dpois(X[i],lambda[k])) / sum(pi * dpois(X[i],lambda))
    }
  }
  return(t)
}

```

### 3. M step

At iteration  $q$ , M step computes

$$\theta^{(q+1)} = \arg \max_{\theta} Q(\theta|\theta^{(q)})$$

$$\begin{aligned}
\frac{\partial}{\partial \lambda_1} Q(\theta|\theta^{(q)}) &= \frac{\partial}{\partial \lambda_1} \sum_{i=1}^n \sum_{k=1}^K t_{ik}^{(q)} \log \left( \pi_k \times \frac{e^{-\lambda_k}}{x_i!} \lambda_k^{x_i} \right) \\
&= \frac{\partial}{\partial \lambda_1} \sum_{i=1}^n t_{i1}^{(q)} (\log \pi_1 - \lambda_1 - \log(x_i!) + x_i \log \lambda_1) \\
&= \sum_{i=1}^n t_{i1}^{(q)} \left( -1 + \frac{x_i}{\lambda_1} \right) = 0
\end{aligned}$$

$$\lambda_1^{(q+1)} = \frac{\sum_{i=1}^n t_{i1}^{(q)} x_i}{\sum_{i=1}^n t_{i1}^{(q)}}$$

We calculate the derivative the same way for the other  $\lambda$  ( $\lambda_2, \dots, \lambda_K$ ). We get:

$$\lambda_k^{(q+1)} = \frac{\sum_{i=1}^n t_{ik}^{(q)} x_i}{\sum_{i=1}^n t_{ik}^{(q)}}, \quad k \in \llbracket 1, K \rrbracket$$

Since there is a constraint  $\sum_{k=1}^K \pi_k = 1$ , we consider Lagrangian:

$$\mathcal{L}(\theta, \alpha) = Q(\theta|\theta^{(q)}) + \alpha \left( \sum_{k=1}^K \pi_k - 1 \right)$$

$$\begin{aligned} \frac{\partial}{\partial \pi_1} \mathcal{L}(\theta, \alpha) &= \frac{\partial}{\partial \pi_1} \sum_{i=1}^n t_{i1}^{(q)} \log \left( \pi_1 \times \frac{e^{-\lambda_1}}{x_i!} \lambda_1^{x_i} \right) + \alpha = \sum_{i=1}^n \frac{t_{i1}^{(q)}}{\pi_1} + \alpha = 0 \\ \pi_1 &= - \frac{\sum_{i=1}^n t_{i1}^{(q)}}{\alpha} \end{aligned}$$

We calculate the derivative the same way for the other  $\pi$  ( $\pi_2, \dots, \pi_K$ ). We get:

$$\begin{aligned} \pi_k &= - \frac{\sum_{i=1}^n t_{ik}^{(q)}}{\alpha}, \quad k \in \llbracket 1, K \rrbracket \\ \frac{\partial}{\partial \alpha} \mathcal{L}(\theta, \alpha) &= \sum_{k=1}^K \pi_k - 1 = 0 \\ \alpha &= - \sum_{i=1}^n \sum_{k=1}^K t_{ik}^{(q)} \\ \pi_k^{(q+1)} &= \frac{\sum_{i=1}^n t_{ik}^{(q)}}{\sum_{i=1}^n \sum_{m=1}^K t_{im}^{(q)}} = \frac{\sum_{i=1}^n t_{ik}^{(q)}}{n}, \quad k \in \llbracket 1, K \rrbracket \end{aligned}$$

```
# t : result of E step
M_step_poisson <- function(X, K, t) {
  lambda <- sapply(1:K, function(k) sum(t[,k]*X) / sum(t[,k]))
  pi <- sapply(1:K, function(k) sum(t[,k]) / length(X))
  return(c(pi, lambda))
}
```

#### 4. Testing the programmed EM algorithm on the simulated data

We take the stopping condition of the algorithm:

$$\frac{\|\theta^{(q)} - \theta^{(q+1)}\|_2^2}{\|\theta^{(q)}\|_2^2} < \epsilon$$

where  $\epsilon = 10^{-6}$

```

X <- poisson3_15
K <- 2

# initialization
theta <- EM_init_poisson(K)

repeat {

  # E step
  t <- E_step_poisson(X, K, theta)

  # M step
  new_theta <- M_step_poisson(X, K, t)

  # stopping condition
  if (sum((theta - new_theta)^2) / sum(theta^2) < 1e-6) {
    break
  }

  theta <- new_theta
}

# Result of theta from EM
print(paste(c("pi_1 =", "pi_2 =", "lambda_1 =", "lambda_2 =", new_theta))

```

```

## [1] "pi_1 = 0.332860492054198"      "pi_2 = 0.667139507945802"
## [3] "lambda_1 = 3.04108390158366"    "lambda_2 = 14.8910533171639"

```

```

# Comparison between true classes and EM clusters
EM_cluster <- c()
for (row in 1:nrow(t)) {
  EM_cluster <- c(EM_cluster, which.max(t[row,]))
}
table(EM=EM_cluster, True=true_class)

```

```

##      True
## EM      1      2
##   1  98      5
##   2   2 195

```