# Homeexam 3; STA-3001

Isak Edvardsen

March 2020

## 1 1: EM, MCEM, DA

Assume data from $n = 20$ street lamps, each having two light bulbs.

If we assume that the failure times of individual light bulbs are independent and exponentially distributed with mean $1/\lambda$ then the failure times of the street lamps is gamma distributed (with $\alpha = 2$):

$$f(x|\lambda) = \lambda^2 x e^{-x\lambda}, \qquad x > 0, \qquad \lambda > 0 \tag{1}$$

Assume a random sample of n observations of failure times of street lamps, some of these are censored and we observe $\{X_i\}_{i=1}^{n+m}$.

We'll suppose $\{X_i\}_{i=1}^{m}$ are observed, but $X_i = c$ for $m + 1 \leq i \leq n$ (only know that they had not failed yet at a certain time $c$).

The likelihood is

$$L(X;\lambda) = \left(\prod_{i=1}^{m} f(x_i|\lambda)\right) \cdot \left(\prod_{i=m+1}^{n} f(x_i|\lambda)\right) \tag{2}$$

The observations $\{X_i\}_{i=m+1}^{n}$ can be treated as if they were missing.
Define the complete observations $Z = \{X_i\}_{i=m+1}^{n}$, hence Z contains the unobserved failure times.

The likelihood of $Y = (X, Z)$ (independent) is:

$$
\begin{aligned}
L(Y;\lambda) &= \left(\prod_{i=1}^{m} f(x_i|\lambda)\right) \cdot \left(\prod_{i=m+1}^{n} f(z_i|\lambda)\right) \\
&= \left(\prod_{i=1}^{m} \lambda^2 x_i e^{-\lambda x_i}\right) \cdot \left(\prod_{i=m+1}^{n} \lambda^2 z_i e^{-\lambda z_i}\right) \\
&= \lambda^{2m} e^{-\lambda \sum_{i=1}^{m} x_i} \left(\prod_{i=1}^{m} x_i\right) \cdot \lambda^{2(n-m-1)} e^{-\lambda \sum_{i=m+1}^{n} z_i} \left(\prod_{i=m+1}^{n} z_i\right) \\
&= \lambda^{2(n-1)} e^{-\lambda \left(\sum_{i=1}^{m} x_i + \sum_{i=m+1}^{n} z_i\right)} \left(\prod_{i=1}^{m} x_i\right) \cdot \left(\prod_{i=m+1}^{n} z_i\right)
\end{aligned}
$$

the completed log-likelihood is:

$$
\begin{aligned}
lnL(Y;\lambda) &= ln\left\{\prod_{i=1}^{m} f(x_i|\lambda) \cdot \prod_{i=m+1}^{n} f(z_i|\lambda)\right\} \qquad (3) \\
&= \sum_{i=1}^{m} lnf(x_i|\lambda) + \sum_{i=m+1}^{n} lnf(z_i|\lambda) \\
&= \sum_{i=1}^{m} ln\big(\lambda^2 x_i e^{-x_i\lambda}\big) + \sum_{i=1}^{n} ln\big(\lambda^2 z_i e^{-z_i\lambda}\big) \\
&= \sum_{i=1}^{m} \big(2ln(\lambda) + ln(x_i) - x_i\lambda\big) + \sum_{i=1+m}^{n} \big(2ln(\lambda) + ln(z_i) - z_i\lambda\big) \\
&= 2(n-1)ln(\lambda) - \lambda\left(\sum_{i=1}^{m} x_i + \sum_{i=m+1}^{n} z_i\right) + \sum_{i=1}^{m} ln(x_i) + \sum_{i=m+1}^{n} ln(z_i)
\end{aligned}
$$

$$
\mathbb{E}(Z_i|\boldsymbol{x}, \lambda) = \int_{c}^{\infty} z f_Z(z) dz \qquad (4)
$$

where $f_Z(z)$ is a truncated distribution where the bottom of the distribution has been removed:

$$f(z|Z>c) = \frac{f_X(z)}{1-F_Z(c)} = \frac{\lambda^2 z e^{-\lambda z}}{1-\left(1-\frac{\Gamma(2,\lambda c)}{\Gamma(2)}\right)}$$

$$= \frac{\lambda^2 z e^{-z\lambda}}{1-\Big(1-e^{-\lambda c}(1+\lambda c)\Big)} = \frac{\lambda^2 z e^{-z\lambda}}{e^{-c\lambda}\Big(1+\lambda c\Big)}$$

$$= \frac{\lambda^2 z e^{-\lambda(z-c)}}{\lambda c+1} \tag{5}$$

insert in 4 gives:

$$\mathbb{E}(Z_i|\boldsymbol{x},\lambda) = \int_c^\infty z\cdot\frac{\lambda^2 z e^{-\lambda(z-c)}}{\lambda c+1}dz$$

$$= \frac{c\lambda(c\lambda+2)+2}{\lambda(c\lambda+1)} \tag{6}$$

from equation (5) we get:

$$Q(\lambda|\lambda(t)) = \mathbb{E}\left(l(\lambda|\boldsymbol{Y})|\boldsymbol{x},\lambda^{(t)}\right)$$

$$= \mathbb{E}\Bigg(2(n-1)ln(\lambda) - \lambda\Bigg(\sum_{i=1}^{m}x_i + \sum_{i=m+1}^{n}z_i\Bigg) + \sum_{i=1}^{m}ln(x_i) + \sum_{i=m+1}^{n}ln(z_i)\Bigg)$$

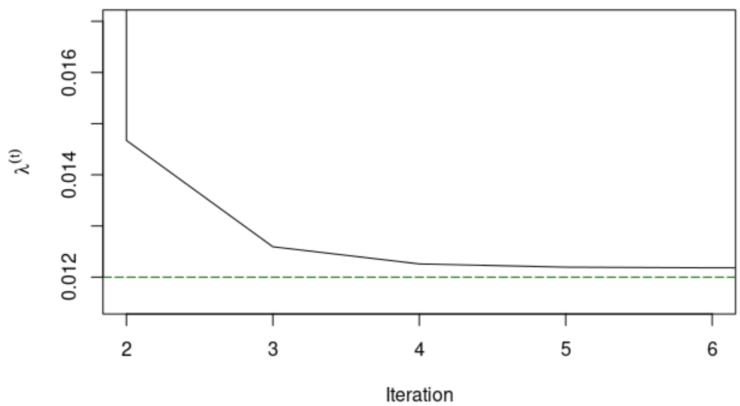
$$= 2(n-1)ln(\lambda) - \lambda\Bigg(\sum_{i=1}^{m}x_i + \sum_{i=m+1}^{n}\mathbb{E}\left(Z_i|\boldsymbol{x},\lambda^{(t)}\right)\Bigg)$$
$$+ \sum_{i=1}^{m}ln(x_i) + \sum_{i=m+1}^{n}ln\,\mathbb{E}\left(Z_i|\boldsymbol{x},\lambda^{(t)}\right)$$

Differentiate to find $\lambda^{(t+1)} = argmax_{\lambda} Q(\lambda|\lambda^{(t)})$:

$$\frac{\delta Q(\lambda|\lambda^{(t)})}{\delta \lambda} = \frac{2(n-1)}{\lambda} - \left( \sum_{i=1}^{m} x_i + \sum_{i=m+1}^{n} \mathbb{E}\left(Z_i|\boldsymbol{x}, \lambda^{(t)}\right) \right) = 0$$

$$\begin{aligned} \lambda &= 2(n-1) \Bigg/ \left( \sum_{i=1+m}^{n} \mathbb{E}\left(Z_i|\boldsymbol{x}, \lambda^{(t)}\right) + \sum_{i}^{m} x_i \right) \\ &= 2(n-1) \Bigg/ \left( \sum_{i=1+m}^{n} \frac{c_i \lambda^{(t)}(c_i \lambda^{(t)} + 2) + 2}{\lambda^{(t)}(c_i \lambda^{(t)} + 1)} + \sum_{i}^{m} x_i \right) \end{aligned}$$

E-step: Compute expectation of censored data $\mathbb{E}\left(Z_i|\boldsymbol{x}, \lambda^{(t)}\right)$
M-step: Compute $\lambda^{(t+1)}$



## 2 4: Simulated Annealing

First make some useful functions.

```r
rvec = function(n=6)
{
  # returns a vector with six elements where the first
  # element is set to 1.
  # Can be seen as a flight route, starting in London.

  vec = vector(length=n)
  vec[1] = 1
  vec[2:n] = c(sample(2:6))

  return(vec)
}

rnd_swap <- function(vec)
{
  # swaps two elements (that are not fixed)
  # and returns a alternative flight route.

  rnd_ind = sample(2:6,2)

  k = replace(vec, c(rnd_ind[1], rnd_ind[2]),
              vec[c(rnd_ind[2], rnd_ind[1])])
  return(k)
}

dist_sum = function(vec)
{
  # Return the total travel distance of the flight-route
  # (and back to starting point)

  km = NULL
  n = 6
  for (i in 1:5)
  {
    km[i] = data[vec[i],vec[i+1]]
  }
  value = sum(km) + data[vec[6],vec[1]]
  return(value)
}
```

Then do the annealing:

```
names =c("London","Mexico city","New York","Paris","Peking","Tokyo")
data <- read.csv("data.csv", header=F, sep=";")
data<- as.matrix(data)
```

```
simanneal.dist = function(n=500)
{
  p.vec = NULL
  T_high = 100
  T_low = 10
  route = rvec()

  while (T_high > T_low)
  {
    for (i in 1:n)
      {
      new_route = rnd_swap(route)

      delta = (dist_sum(new_route) - dist_sum(route))

      rho = exp(-delta/T_high)

      if (runif(1) < rho) route = new_route

      p.vec = c(p.vec,dist_sum(route))

      T_high = T_high * 0.8
      }
  }

  print(tail(route, n=6))
  print(names[tail(route, n=6)])
  print(dist_sum(tail(route, n=6)))
  return(p.vec)

}
p.vec = simanneal.dist()

## [1] 1 3 2 6 5 4
## [1] "London"      "New York"    "Mexico city" "Tokyo"       "Peking"
## [6] "Paris"
##    V1
## 19235
```