Assignment 2 – **sketch** of the solutions

1. (a) Firstly, to write down the likelihood we also need the density function, which is given by

$$f(y;\theta) = \frac{\mathrm{d}}{\mathrm{d}y} F(y;\theta) = \frac{y}{\theta} e^{-y^2/(2\theta)}.$$

Also, because $F(y;\theta) + S(y;\theta) = 1$, where $S(y;\theta) = \Pr(Y > y;\theta)$, we have that $S(y;\theta) = 1 - F(y;\theta) = e^{-y^2/(2\theta)}$. The contribution of a noncensored observation to the likelihood is $f(y;\theta)$, whereas the contribution of a censored observation to the likelihood is $\Pr(Y > C;\theta) = S(C;\theta)$. Further note that when $r_i = 1$, we have that $x_i = y_i$ and when $r_i = 0$, we have that $x_i = C$. Therefore, the likelihood of the observed data $\{(x_i, r_i)\}_{i=1}^n$ can be written as

$$\begin{aligned} L(\theta;\mathbf{x},\mathbf{r}) &= \prod_{i=1}^n \left\{ f(x_i;\theta)^{r_i} S(x_i;\theta)^{1-r_i} \right\} \\ &= \prod_{i=1}^n \left\{ \left(\frac{x_i}{\theta} e^{-x_i^2/(2\theta)} \right)^{r_i} \left(e^{-x_i^2/(2\theta)} \right)^{1-r_i} \right\}, \end{aligned}$$

where $\mathbf{x} = \{x_1,\ldots,x_n\}$ and $\mathbf{r} = \{r_1,\ldots,r_n\}$. The corresponding log likelihood of the observed data is

$$\begin{aligned} \log L(\theta;\mathbf{x},\mathbf{r}) &= \sum_{i=1}^n r_i \left\{ \log x_i - \log\theta - \frac{x_i^2}{2\theta} \right\} + \sum_{i=1}^n (1-r_i) \left(-\frac{x_i^2}{2\theta} \right) \\ &= \sum_{i=1}^n r_i \log x_i - \log\theta \sum_{i=1}^n r_i - \frac{1}{2\theta} \sum_{i=1}^n x_i^2. \end{aligned}$$

The derivative is given by

$$\frac{\mathrm{d}}{\mathrm{d}\theta} \log L(\theta;\mathbf{x},\mathbf{r}) = -\frac{1}{\theta} \sum_{i=1}^n r_i + \frac{1}{2\theta^2} \sum_{i=1}^n x_i^2.$$

Equating the derivative to zero, one obtains

$$\widehat{\theta}_{\mathrm{MLE}} = \frac{\sum_{i=1}^n X_i^2}{2\sum_{i=1}^n R_i}.$$

(b) In order to obtain the expected Fisher information we start by calculating the second derivative

$$\frac{\mathrm{d}^2}{\mathrm{d}\theta^2} \log L(\theta;\mathbf{x},\mathbf{r}) = \frac{1}{\theta^2} \sum_{i=1}^n r_i - \frac{1}{\theta^3} \sum_{i=1}^n x_i^2.$$

Hence,

$$\begin{aligned} I(\theta) &= -E\left[\frac{\mathrm{d}^2}{\mathrm{d}\theta^2} \log L(\theta;\mathbf{x},\mathbf{r}) \right] \\ &= -\frac{1}{\theta^2} nE(R) + \frac{1}{\theta^3} nE(X^2). \end{aligned}$$

We have that

$$E(R) = 1 \times \Pr(R=1) + 0 \times \Pr(R=0) = \Pr(R=1) = \Pr(Y \le C) = F(C) = 1 - e^{-C^2/(2\theta)}.$$

We now proceed to compute $E(X^2)$. We start by noting that

$$X = YR + C(1-R) \Rightarrow X^2 = Y^2R + C^2(1-R) = Y^2 I(Y \leq C) + C^2 I(Y > C),$$

and therefore the law of total expectation implies that

$$E(X^2) = E(Y^2 \mid Y \leq C)\Pr(Y \leq C) + E(C^2 \mid Y > C)\Pr(Y > C).$$

Because C is fixed, $E(C^2 \mid Y > C) = C^2$. On the other hand,

$$\begin{aligned} E(Y^2 \mid Y \leq C) &= \frac{1}{F(C)}\int_0^C y^2 f(y;\theta)\mathrm{d}y \\ &= \frac{1}{F(C)}\{-C^2 e^{-C^2/2\theta} + 2\theta(1-e^{-C^2/2\theta})\}. \end{aligned}$$

We thus have,

$$\begin{aligned} E(X^2) &= -C^2 e^{-C^2/2\theta} + 2\theta(1-e^{-C^2/2\theta}) + C^2 e^{-C^2/2\theta} \\ &= 2\theta(1-e^{-C^2/2\theta}). \end{aligned}$$

Hence, the expected Fisher information is

$$\begin{aligned} I(\theta) &= -\frac{1}{\theta^2}n(1-e^{-C^2/2\theta}) + \frac{1}{\theta^3}n2\theta(1-e^{-C^2/2\theta}) \\ &= \frac{n}{\theta^2}(1-e^{-C^2/2\theta}). \end{aligned}$$

(c) Appealing to the asymptotic normality of the MLE, we know that $\widehat{\theta}_{\text{MLE}}$ follows a normal distribution with mean θ and variance given by $1/I(\theta)$. Therefore, we have that a 95% confidence interval for θ is given by

$$\left(\widehat{\theta}_{\text{MLE}} - 1.96\sqrt{1/I(\widehat{\theta}_{\text{MLE}})}, \widehat{\theta}_{\text{MLE}} + 1.96\sqrt{1/I(\widehat{\theta}_{\text{MLE}})}\right),$$

where $\widehat{\theta}_{\text{MLE}}$ was found in (a) and $I(\theta)$ was found in (b).

2. (a) In the case of left censored observations, the contribution to the likelihood of a censored observation is $\Pr(Y \leq D; \mu, \sigma^2) = \Phi(D; \mu, \sigma^2)$. On the other hand, the contribution of a noncensored observation to the likelihood is given by $\phi(y; \mu, \sigma^2)$. Further note that when $r_i = 1$, we have that $x_i = y_i$ and when $r_i = 0$, we have that $x_i = D$. Therefore we shall write the likelihood of the observed data $\{(x_i, r_i)\}_{i=1}^n$ as

$$L(\mu, \sigma^2; \mathbf{x}, \mathbf{r}) = \prod_{i=1}^n \{\phi(x_i; \mu, \sigma)^{r_i}\Phi(x_i; \mu, \sigma)^{1-r_i}\},$$

where $\mathbf{x} = \{x_1, \ldots, x_n\}$ and $\mathbf{r} = \{r_1, \ldots, r_n\}$. The corresponding log likelihood of the observed data is given by

$$\log L(\mu, \sigma^2; \mathbf{x}, \mathbf{r}) = \sum_{i=1}^n \{r_i \log\phi(x_i; \mu, \sigma) + (1-r_i)\log\Phi(x_i; \mu, \sigma)\}$$

(b) The log likelihood of the observed data is not analytically tractable and so we resort to numerical methods to find the maximum likelihood estimate of μ.

```
load("dataex2.Rdata")

log_like <- function(mu, x, r){
```

```r
  sum(r*dnorm(x, mu, 1.5, log = TRUE) + (1-r)*pnorm(x, mu, 1.5, log = TRUE))
}

res <- optim(par = 2, log_like, x = dataex2$X, r = dataex2$R,
             control = list(fnscale = -1),
             method = "Brent", lower = -10, upper = 10,
             hessian = TRUE)
res
```

```
## $par
## [1] 5.532804
##
## $value
## [1] -336.3821
##
## $counts
## function gradient
##       NA       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]
## [1,] -86.53018
```

We thus have that $\widehat{\mu} = 5.5328$ (rounded to 4 decimal places). Different starting values led to the same estimate.

3. (a) Ignorable, since data are MAR and the parameters of the missingness mechanism and of the data model are distinct.

(b) If $\psi_1 \neq 0$, not ignorable as data are MNAR. If $\psi_1 = 0$, ignorable (MCAR + distinct parameters).

(c) Not ignorable, although data are MAR, μ_1 is a parameter in both the model for the missing data mechanism and the data model and, therefore, one of the parameters of the model for the missing data mechanism provides full information about one of the parameters of the data model. Thus, distinctness of the parameters (of the missingness mechanism and data model) does not hold.

4. The likelihood of the complete data is given by

$$L(\boldsymbol{\beta}; \mathbf{y}_{\text{obs}}, \mathbf{y}_{\text{mis}}, \mathbf{x}) = \prod_{i=1}^{n} \{p_i(\boldsymbol{\beta})^{y_i}(1 - p_i(\boldsymbol{\beta}))^{1-y_i}\}, \quad p_i(\boldsymbol{\beta}) = \frac{\exp(\beta_0 + x_i\beta_1)}{1 + \exp(\beta_0 + x_i\beta_1)},$$

where $\mathbf{y}_{\text{obs}} = \{y_1, \ldots, y_m\}$, $\mathbf{y}_{\text{mis}} = \{y_{m+1}, \ldots, y_n\}$, and $\mathbf{x} = \{x_1, \ldots, x_n\}$. The corresponding log likelihood of the complete data is

$$\log L(\boldsymbol{\beta}; \mathbf{y}_{\text{obs}}, \mathbf{y}_{\text{mis}}, \mathbf{x}) = \sum_{i=1}^{m} y_i(\beta_0 + x_i\beta_1) + \sum_{i=m+1}^{n} y_i(\beta_0 + x_i\beta_1) - \sum_{i=1}^{n} \log(1 + e^{\beta_0 + \beta_1 x_i}).$$

The E-step at iteration $(t+1)$ is given by

$$
\begin{aligned}
Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)}) &= E_{\text{Ymis}}\left[\log L(\boldsymbol{\beta}; \mathbf{y}_{\text{obs}}, \mathbf{y}_{\text{mis}}, \mathbf{x}) \mid \mathbf{y}_{\text{obs}}, \mathbf{x}, \boldsymbol{\beta}^{(t)}\right] \\
&= \sum_{i=1}^{m} y_i(\beta_0 + x_i\beta_1) - \sum_{i=1}^{n} \log(1 + e^{\beta_0 + \beta_1 x_i}) + \sum_{i=m+1}^{n} E\left[Y_i \mid \mathbf{y}_{\text{obs}}, \boldsymbol{\beta}^{(t)}\right](\beta_0 + x_i\beta_1),
\end{aligned}
$$

where $\boldsymbol{\beta}^{(t)} = \left(\beta_0^{(t)}, \beta_1^{(t)}\right)$. In what concerns the expectation term we have

$$
\begin{aligned}
E\left[Y_i \mid \mathbf{y}_{\text{obs}}, x_i, \boldsymbol{\beta}^{(t)}\right] &= \Pr(Y_i = 1 \mid x_i, \boldsymbol{\beta}^{(t)}) \\
&= \frac{e^{\beta_0^{(t)} + x_i\beta_1^{(t)}}}{1 + e^{\beta_0^{(t)} + x_i\beta_1^{(t)}}} \\
&= p_i(\boldsymbol{\beta}^{(t)}), \quad i = m+1, \dots, n.
\end{aligned}
$$

Therefore,

$$
Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)}) = \sum_{i=1}^{m} y_i(\beta_0 + x_i\beta_1) - \sum_{i=1}^{n} \log(1 + e^{\beta_0 + \beta_1 x_i}) + \sum_{i=m+1}^{n} p_i(\boldsymbol{\beta}^{(t)})(\beta_0 + x_i\beta_1).
$$

The score equations

$$
\frac{\partial}{\partial \beta_0} Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)}) = 0,
$$

and

$$
\frac{\partial}{\partial \beta_1} Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)}) = 0,
$$

do not have a closed form solution and therefore we resort to numerical methods. Below is my implementation and for the stopping criterion I have used

$$
\left|\beta_0^{(t+1)} - \beta_0^{(t)}\right| + \left|\beta_1^{(t+1)} - \beta_1^{(t)}\right| < 0.00001.
$$

```
load("dataex4.Rdata")

reg_log_mis <- function(y, x, beta.start, eps){

  r <- ifelse(is.na(y) == TRUE, 0, 1)

  beta <- beta.start
  diff <- 1

  while(diff > eps){
    beta.old <- beta

    #E- step
    pit <- exp(beta.old[1] + beta.old[2]*x[r == 0])/
      (1 + exp(beta.old[1] + beta.old[2]*x[r == 0]))

    Qfunction <- function(beta, x, y, r){
      beta0 <- beta[1]
      beta1 <- beta[2]
      - sum(log(1 + exp(beta0+beta1*x))) + sum((y[r == 1]*(beta0 + beta1*x[r==1]))) +
        sum(pit*(beta0 + beta1*x[r == 0]))
    }
```

```r
    #M-step
    res <- optim(c(beta.old[1], beta.old[2]), Qfunction, x = x, y = y, r = r,
                 control = list(fnscale = -1), hessian = FALSE)

    beta <- c(res$par[1], res$par[2])
    diff <- sum(abs(beta - beta.old))
  }
  return(beta)
}

res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(0, 0), eps = 0.00001)
res
```

```
## [1]  0.9757079 -2.4799874
```

```r
res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(100, 100), eps = 0.00001)
res
```

```
## [1]  0.9756087 -2.4801795
```

```r
res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(-200, -200), eps = 0.00001)
res
```

```
## [1]  0.975477 -2.480356
```

```r
res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(0, 0), eps = 1e-17)
res
```

```
## [1]  0.9757079 -2.4799874
```

```r
res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(100, 100), eps = 1e-17)
res
```

```
## [1]  0.9756087 -2.4801795
```

```r
res <- reg_log_mis(y = dataex4$Y, x = dataex4$X, beta.start = c(-200, -200), eps = 1e-17)
res
```

```
## [1]  0.975477 -2.480356
```

```r
log_like_observed <- function(x, y, beta){
r <- ifelse(is.na(y) == TRUE, 0, 1)
beta0 <- beta[1]
beta1 <- beta[2]
- sum(log(1 + exp(beta0+beta1*x[r==1]))) + sum((y[r == 1]*(beta0 + beta1*x[r==1])))
}

res_observed <- optim(c(200, -200), log_like_observed, y = dataex4$Y, x = dataex4$X,
                      control = list(fnscale = -1, reltol = 1e-17))
res_observed$par
```

```
## [1]  0.9755262 -2.4803837
```

```r
res_observed$convergence
```

```
## [1] 0
```

```r
res_glm <- glm(Y ~ X, family = "binomial", data = dataex4)
res_glm$coefficients
```

```
## (Intercept)           X
##   0.9755261  -2.4803837
```

```
require(rjags)
y <- dataex4$Y
x <- dataex4$X
r <- ifelse(is.na(y) == TRUE, 0, 1)
x_obs <- x[r==1]
y_obs <- y[r==1]
n <- length(y_obs)

model_string <- "model{
for (i in 1:n){
   y_obs[i] ~ dbern(p[i])
   logit(p[i]) <- beta0 + beta1*x_obs[i]
}

beta0 ~ dnorm(0, 0.01)
beta1 ~ dnorm(0, 0.01)
}"

model <- jags.model(textConnection(model_string),
                    data = list(n = n,y_obs = y_obs, x_obs = x_obs),
                    n.chains = 1)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 405
##    Unobserved stochastic nodes: 2
##    Total graph size: 2030
##
## Initializing model
```

```
update(model, 2000, progress.bar = "none")

res_bayes <- coda.samples(model,variable.names = c("beta0","beta1"),
                    n.iter = 20000, progress.bar = "none")
summary(res_bayes)
```

```
##
## Iterations = 3001:23000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 20000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##          Mean     SD  Naive SE Time-series SE
## beta0  0.9836 0.1287 0.0009099       0.001124
## beta1 -2.5088 0.3407 0.0024092       0.003183
##
## 2. Quantiles for each variable:
```

```
##
##          2.5%     25%     50%    75%  97.5%
## beta0  0.7321  0.8961  0.9838  1.071  1.236
## beta1 -3.1946 -2.7331 -2.5015 -2.276 -1.852
```

We obtain $\widehat{\beta}_0 = 0.9755$ and $\widehat{\beta}_1 =$ -2.4804 (rounded to 4 decimal places). Different starting values led to the same estimates.

5.(a) We have

$$f(y) = pf_{\text{LN}}(y;\mu,\sigma^2) + (1-p)f_{\text{Exp}}(y;\lambda).$$

To apply the EM algorithm we need to form the complete data. In order to do so, let us introduce $z_1,\ldots,z_n$ binary unobserved (latent) variables such that

$$z_i = \begin{cases} 1, & \text{if } y_i \text{ comes from the first component (lognormal distribution),} \\ 0, & \text{if } y_i \text{ comes from the second component (exponential distribution),} \end{cases}$$

for $i = 1,\ldots,n$. The complete data is then formed by $(\mathbf{y},\mathbf{z})$, and the complete data likelihood is given by

$$L(\theta;\mathbf{y},\mathbf{z}) = \prod_{i=1}^{n} \left\{ [pf_{\text{LN}}(y;\mu,\sigma^2)]^{z_i} [(1-p)f_{\text{Exp}}(y;\lambda)]^{1-z_i} \right\},$$

where $\theta = (p,\mu,\sigma^2,\lambda)$. The log likelihood function of the complete data is then

$$\log L(\theta;\mathbf{y},\mathbf{z}) = \sum_{i=1}^{n} z_i\{\log p + \log f_{\text{LN}}(y_i;\mu,\sigma^2)\} + \sum_{i=1}^{n}(1-z_i)\{\log(1-p) + \log f_{\text{Exp}}(y_i;\lambda)\}.$$

Let us proceed to the E-step. At iteration $(t+1)$ of the algorithm, we need to compute $Q(\theta \mid \theta^{(t)})$, where $\theta^{(t)}$ denotes an estimate of θ at iteration t.

$$\begin{aligned} Q(\theta \mid \theta^{(t)}) &= E_Z\left[\log L(\theta;\mathbf{y},\mathbf{z}) \mid \mathbf{y},\theta^{(t)}\right] \\ &= \sum_{i=1}^{n} E\left[Z_i \mid \mathbf{y},\theta^{(t)}\right] \{\log p + \log f_{\text{LN}}(y_i;\mu,\sigma^2)\} \\ &+ \sum_{i=1}^{n} \left(1 - E\left[Z_i \mid \mathbf{y},\theta^{(t)}\right]\right) \{\log(1-p) + \log f_{\text{Exp}}(y_i;\lambda)\} \end{aligned}$$

We need to find $E\left[Z_i \mid \mathbf{y},\theta^{(t)}\right]$. We know

$$\begin{aligned} E\left[Z_i \mid \mathbf{y},\theta^{(t)}\right] &= E\left[Z_i \mid y_i,\theta^{(t)}\right] \\ &= 1 \times \Pr\left(Z_i = 1 \mid y_i,\theta^{(t)}\right) + 0 \times \Pr\left(Z_i = 0 \mid y_i,\theta^{(t)}\right) \\ &= \Pr\left(Z_i = 1 \mid y_i,\theta^{(t)}\right) \end{aligned}$$

Using Bayes theorem and the law of total probability, it follows that

$$\begin{aligned} \Pr\left(Z_i = 1 \mid y_i,\theta^{(t)}\right) &= \frac{p^{(t)} f_{\text{LN}}\left(y_i;\mu^{(t)},(\sigma^{(t)})^2\right)}{p^{(t)} f_{\text{LN}}\left(y_i;\mu^{(t)},(\sigma^{(t)})^2\right) + (1-p^{(t)}) f_{\text{Exp}}(y_i;\lambda^{(t)})}. \\ &= \tilde{p}_i^{(t)} \end{aligned}$$

Thus,

$$
\begin{aligned}
Q(\theta \mid \theta^{(t)}) &= \sum_{i=1}^{n} \tilde{p}_i^{(t)} \{\log p + \log f_{\mathrm{LN}}(y_i; \mu, \sigma^2)\} \\
&+ \sum_{i=1}^{n} (1 - \tilde{p}_i^{(t)}) \{\log(1-p) + \log f_{\mathrm{Exp}}(y_i; \lambda)\} \\
&= \sum_{i=1}^{n} \tilde{p}_i^{(t)} \{\log p - \log y_i - \frac{1}{2}\log(2\pi) - \frac{1}{2}\log \sigma^2 - \frac{1}{2\sigma^2}(\log y_i - \mu)^2\} \\
&+ \sum_{i=1}^{n} (1 - \tilde{p}_i^{(t)}) \{\log(1-p) + \log \lambda - \lambda y_i\}.
\end{aligned}
$$

Having finished the E-step, let us proceed now to the M-step. Taking the derivative with respect to p, we obtain

$$
\frac{\partial}{\partial p} Q(\theta \mid \theta^{(t)}) = 0 \Rightarrow \frac{1}{p}\sum_{i=1}^{n} \tilde{p}_i^{(t)} - \frac{1}{1-p}\sum_{i=1}^{n}(1 - \tilde{p}_i^{(t)}) = 0 \Rightarrow p^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n} \tilde{p}_i^{(t)}.
$$

Taking the derivative with respect to μ, we obtain

$$
\frac{\partial}{\partial \mu} Q(\theta \mid \theta^{(t)}) = 0 \Rightarrow \sum_{i=1}^{n} \tilde{p}_i^{(t)} \left\{\frac{1}{\sigma^2}(\log y_i - \mu)\right\} = 0 \Rightarrow \mu^{(t+1)} = \frac{\sum_{i=1}^{n} \tilde{p}_i^{(t)} \log y_i}{\sum_{i=1}^{n} \tilde{p}_i^{(t)}}.
$$

Taking the derivative with respect to σ^2, we obtain

$$
\begin{aligned}
\frac{\partial}{\partial \sigma^2} Q(\theta \mid \theta^{(t)}) = 0 &\Rightarrow \sum_{i=1}^{n} \tilde{p}_i^{(t)} \left\{-\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}(\log y_i - \mu)^2\right\} = 0 \\
&\Rightarrow (\sigma^{(t+1)})^2 = \frac{\sum_{i=1}^{n} \tilde{p}_i^{(t)} (\log y_i - \mu^{(t+1)})^2}{\sum_{i=1}^{n} \tilde{p}_i^{(t)}}
\end{aligned}
$$

Finally, taking the derivative with respect to λ, we obtain

$$
\frac{\partial}{\partial \lambda} Q(\theta \mid \theta^{(t)}) = 0 \Rightarrow \sum_{i=1}^{n}(1 - \tilde{p}_i^{(t)}) \left(\frac{1}{\lambda} - y_i\right) = 0 \Rightarrow \lambda^{(t+1)} = \frac{\sum_{i=1}^{n}(1 - \tilde{p}_i^{(t)})}{\sum_{i=1}^{n}(1 - \tilde{p}_i^{(t)}) y_i}.
$$

(b) Below we implement the EM algorithm derived in part (a) and as stopping criterion I have used

$$
\left|p^{(t+1)} - p^{(t)}\right| + \left|\mu^{(t+1)} - \mu^{(t)}\right| + \left|\sigma^{(t+1)} - \sigma^{(t)}\right| + \left|\lambda^{(t+1)} - \lambda^{(t)}\right| < 0.00001.
$$

```r
load("dataex5.Rdata")

em.mixture <- function(y, theta0, eps){
  n <- length(y)
  theta <- theta0
  p <- theta[1]; mu <- theta[2]
  sigma <- theta[3]; lambda <- theta[4]
  diff <- 1

  while(diff>eps){
    theta.old <- theta

    #E-step
```

```r
    ptilde1 <- p*dlnorm(y, meanlog = mu, sdlog = sigma)
    ptilde2 <- (1-p)*dexp(y, rate = lambda)
    ptilde <- ptilde1/(ptilde1 + ptilde2)

    #M-step
    p <- mean(ptilde)
    mu <- sum(log(y)*ptilde)/sum(ptilde)
    sigma <- sqrt(sum(((log(y) - mu)^2)*ptilde)/sum(ptilde))
    lambda <- sum(1-ptilde)/sum(y*(1-ptilde))

    theta <- c(p, mu, sigma, lambda)
    diff <- sum(abs(theta - theta.old))
  }
  list(theta)
}

res <- em.mixture(y = dataex5, c(0.1, 1, 0.5, 2), 0.00001)
p <- res[[1]][1]; mu <- res[[1]][2]
sigma <- res[[1]][3]; lambda <- res[[1]][4]
p; mu; sigma; lambda
```

```
## [1] 0.4795916

## [1] 2.013147

## [1] 0.9294364

## [1] 1.033139
```

The estimates of the four parameters are: $\widehat{p} = 0.4796$, $\widehat{\mu} = 2.0131$, $\widehat{\sigma} = 0.9294$, and $\widehat{\lambda} = 1.0331$ (estimates rounded to four decimal places).

The plot below shows that the estimated density nicely follows the histogram of the data.

```r
hist(dataex5, freq = F,
     ylim = c(0,0.2), nclass = 20,
     ylab = "Density", xlab = expression(y),
     main = "", cex.lab = 1.2, cex.axis = 1.2)
curve(p*dlnorm(x, mu, sigma) + (1-p)*dexp(x, lambda),
      add = TRUE, lwd = 2, col = "blue2")
```