# STA2102 Final Exam

Kevin Zhang 1002225264

12/12/2020

## Question 1

### a)

For simplicity, we denote $\int_0^\infty \ln(x)\exp(-x)\,dx = c_1$ and $\int_0^\infty [ln(x)]^2 \exp(-x)\,dx = c_2$.

$$E[\ln(x)] = \int_0^\infty \left(\frac{\lambda \ln(x)}{x\sqrt{2\pi}} \exp\left[\frac{-1}{2}(\ln(x)-\mu)^2\right] + (1-\lambda)\ln(x)\exp(-x)\right) dx$$
$$= \int_{-\infty}^\infty \frac{\lambda y}{\sqrt{2\pi}} \exp\left[\frac{-1}{2}(y-\mu)^2\right] dy + (1-\lambda)c_1 = \lambda\mu + (1-\lambda)c_1$$

$$E[[\ln(x)]^2 = \int_0^\infty \left(\frac{\lambda [\ln(x)]^2}{x\sqrt{2\pi}} \exp\left[\frac{-1}{2}(\ln(x)-\mu)^2\right] + (1-\lambda)[\ln(x)]^2\exp(-x)\right) dx$$
$$= \int_{-\infty}^\infty \frac{\lambda y^2}{\sqrt{2\pi}} \exp\left[\frac{-1}{2}(y-\mu)^2\right] dy + (1-\lambda)c_2 = \lambda(\mu^2+1) + (1-\lambda)c_2$$

$$\Longrightarrow \begin{cases} E[\ln(x)] = \lambda\mu + (1-\lambda)c_1 \\ E[[\ln(x)]^2] = \lambda(\mu^2+1) + (1-\lambda)c_2 \end{cases}$$

$$\Longrightarrow \begin{cases} \lambda = \frac{E[\ln(x)]-c_1}{\mu - c_1} \\ \mu = \frac{E[[\ln(x)]^2]-c_2 \pm \sqrt{(E[[\ln(x)]^2]-c_2)^2 - 4(E[\ln(x)]-c_1)((E[\ln(x)]-c_1)(1-c_2)+c_1(E[[\ln(x)]^2]-c_2))}}{2(E[\ln(x)]-c_1)} \end{cases}$$

```
x <- scan("problem1.txt")
e1 <- mean(log(x)) # sample mean of ln(x)
e2 <- mean((log(x))^2) # sample mean of [ln(x)]^2
c1 <- -0.5772
c2 <- 1.9781
mu1 <- ((e2-c2)+sqrt((e2-c2)^2-4*(e1-c1)*((e1-c1)*(1-c2)+c1*(e2-c2))))/(2*(e1-c1))
mu2 <- ((e2-c2)-sqrt((e2-c2)^2-4*(e1-c1)*((e1-c1)*(1-c2)+c1*(e2-c2))))/(2*(e1-c1))
lambda1 <- (e1-c1)/(mu1-c1)
lambda2 <- (e1-c1)/(mu2-c1)
# mu has to take the value that forces lambda between 0 and 1
if(lambda1>=0 & lambda1<=1){
  lambda_hat <- lambda1
  mu_hat <- mu1
}else{
  lambda_hat <- lambda2
  mu_hat <- mu2
```

```r
}
cat('lambda is', format(round(lambda_hat, 4), nsmall=4),
    'and mu is', format(round(mu_hat, 4), nsmall=4))
```

```
## lambda is 0.1220 and mu is 0.1784
```

## b)

E-step: (note that $\lambda_1 = \lambda$ and $\lambda_2 = 1 - \lambda$)

We have $E(l(x, \mu, \lambda)) = \sum_{i=1}^n \sum_{j=1}^2 \delta_{ij}(\mu, \lambda)(\ln \lambda_i + \ln f(x_i, \mu))$

where $\delta_{i1}(\mu, \lambda) = \frac{\frac{\lambda}{x_i\sqrt{2\pi}} \exp \frac{-1}{2}(\ln(x)-\mu)^2}{\frac{\lambda}{x_i\sqrt{2\pi}} \exp \frac{-1}{2}(\ln(x)-\mu)^2 + (1-\lambda)\exp(-x_i)}$

M-step:

We have $\lambda_{k+1} = \frac{1}{n}\sum_{i=1}^n \delta_{i1}(\mu_k, \lambda_k)$. In addition,

$$
\begin{aligned}
\mu_{k+1} &= \arg\max \sum_{i=1}^n \delta_{i1}(\mu_k, \lambda_k) \ln \left( \frac{1}{x_i\sqrt{2\pi}} \exp\left( \frac{-1}{2}(\ln(x_i) - \mu)^2 \right)\right) \\
&= \arg\max \sum_{i=1}^n \frac{-\delta_{i1}(\mu_k, \lambda_k)}{2}(\ln(x_i) - \mu)^2 \\
&= \frac{\sum_{i=1}^n \delta_{i1}(\mu_k, \lambda_k)\ln(x_i)}{\sum_{i=1}^n \delta_{i1}(\mu_k, \lambda_k)}
\end{aligned}
$$

```r
delta <- function(mu, lambda, x){
  return(lambda*dlnorm(x, meanlog=mu)/(lambda*dlnorm(x, meanlog=mu)+(1-lambda)*exp(-x)))
}
em_iter <- function(mu0, lambda0, x){
  mu <- mu0
  lambda <- lambda0
  for(i in 1:2000){
    d <- numeric(0)
    for(x_temp in x){
      d <- c(d, delta(mu, lambda, x_temp))
    }
    lambda <- mean(d)
    mu <- sum(d*log(x))/sum(d)
  }
  return(c(mu, lambda))
}
est <- em_iter(mu0=mu_hat, lambda0=lambda_hat, x)
cat('mu is', est[1], 'and lambda is', est[2])
```

```
## mu is 0.3837367 and lambda is 0.09743101
```

## c)

When $\lambda = 0$, the maximum likelihood is as the following ($l(x) = -\sum_{i=1}^n x_i$):

```r
h0 <- -sum(x)
h0
```

```
## [1] -1130.93
```

In the parameter space where $\lambda \neq 0$, we plug in the maximum likelihood estimate of $(\lambda, \mu)$ and and the following likelihood:

```
h1 <- sum(log(est[2]*dlnorm(x, meanlog=est[1])+(1-est[2])*exp(-x)))
h1
```

```
## [1] -1119.759
```

The two parameter spaces differ by a degree of freedom of 1 and hence with a p value of:

```
1-pchisq(2*(-h0+h1), df=1)
```

```
## [1] 2.281013e-06
```

This is an extremely small p value and hence we do not accept the null hypothesis $\lambda = 0$.

# Question 2

## a)

First, we use Hutchinson's method to compute/estimate $tr(A)$. First, define $V$ to be a random vector in $R^n$ whose entries are independently and identically distributed on the set $\{-1, 1\}$ (with probability half and half). Second, sample $m$ samples $V_1, \ldots, V_m$ from this distribution. Then estimate $tr(A)$ by $\hat{tr}(A) = \frac{1}{m}\sum_{i=1}^{m} V_i^T A V_i$. Since $A$ is non-negative definite with all positive eigenvalues equal to $\theta$ and $r$ is known, then from the Jordan canonical form of $A$ ($tr(A) = tr(PJP^{-1}) = tr(J)$ and eigenvalue 0 should have multiplicity $n - r$) we have $tr(A) = r\theta$. Hence we can recover $\theta$ from estimate of trace of $A$.

## b)

Similar to part a, we estimate the trace of $A$ first. Notice that $E(tr(A)) = E(\sum_{i=1}^{r} \lambda_i) = r\mu$ and then we can estimate $r$ by $\hat{r} = \frac{\hat{tr}(A)}{\mu}$.

# Question 3

## a)

For this part, I use coordinate descent. First, transform $(x, y)$ to $(z, y)$ with $z = x - y$ and then $g(x, y) = g(z, y) = 4e^{z+y} + 2e^y - 3z - 5y + |z|$. Fix $z$:

$$\partial g_z(y) = 4e^{z+y} + 2e^y - 5 = 0 \implies y = \ln \frac{5}{4e^z + 2}$$

Fix $y$:

$$\partial g_y(z) = 4e^{z+y} - 3 + \partial|z|$$

$$\begin{cases} z > 0 & \partial g_y(z) = 4e^{z+y} - 3 + 1 = 0 \implies z = -y + \log\frac{1}{2} \\ z = 0 & \partial g_y(z) = 4e^{z+y} - 3 + k = 0 \quad k \in [-1, 1] \implies -1 \leq 3 - 4e^y \leq 1 \implies \log\frac{1}{2} \leq y \leq 0 \\ z < 0 & \partial g_y(z) = 4e^{z+y} - 3 - 1 = 0 \implies z = -y \end{cases}$$

$$\implies \begin{cases} z = -y + \log\frac{1}{2} & y < \log\frac{1}{2} \\ z = -y & y > 0 \\ z = 0 & \text{otherwise} \end{cases}$$

```
coord_desc <- function(x0, y0){
  z <- x0-y0
  y <- y0
  for(i in 1:1000){
```

```
    if(y<log(1/2)){
      z <- -y+log(1/2)
    }else if(y>0){
      z <- -y
    }else{
      z <- 0
    }
    y <- log(5/(4*exp(z)+2))
  }
  return(c(z+y, y))
}
res <- coord_desc(1, 1)
x <- res[1]
y <- res[2]
cat('x is', x, 'and y is', y, '\n')
```

```
## x is -0.1823216 and y is -0.1823216
```

```
cat('minimum g is', 4*exp(x)+2*exp(y)-3*x-2*y+abs(x-y))
```

```
## minimum g is 5.911608
```

## b)

For this part, I use Newton-Raphson method. We have $\frac{\partial g}{\partial x} = 4e^x - 3 + 2x - 2y$, $\frac{\partial g}{\partial y} = 2e^y - 2 - 2x + 2y$, $\frac{\partial^g}{\partial x \partial y} = -2$, $\frac{\partial^2 g}{\partial x^2} = 4e^x + 2$ and $\frac{\partial^2 g}{\partial y^2} = 2e^y + 2$.

```
newt_raph <- function(x0){
  x_temp <- x0
  for(i in 1:1000){
    gradient <- c(4*exp(x_temp[1])-3+2*(x_temp[1]-x_temp[2]),
                  2*exp(x_temp[2])-2-2*(x_temp[1]-x_temp[2]))
    hessian <- matrix(c(4*exp(x_temp[1])+2, -2, -2, 2*exp(x_temp[2])+2), nrow=2)
    x_temp <- x_temp-solve(hessian)%*%gradient
  }
  return(x_temp)
}
ans <- newt_raph(c(0,0))
cat('x is', ans[1], 'and y is', ans[2], '\n')
```

```
## x is -0.2190937 and y is -0.1126009
```

```
cat('minimum g is', 4*exp(x)+2*exp(y)-3*x-2*y+(x-y)^2)
```

```
## minimum g is 5.911608
```