Dt.
Pg.

B+

Name : Aditya Chetan
Roll No.: 2016217

MTH 372
Assignment 1

Q1 a] Let's assume that the points are sampled from ~~an unknown~~ a Bernoulli distribution with unknown parameter ~~p~~ θ.

$$\therefore \quad X_i \sim \text{Bernoulli}(p)$$

now,

$$E[X_i] = p(1) + (1-p)(0) = p \qquad m_1 = \frac{\sum_{i=1}^{n} X_i}{n}$$

By method of moments,

$$E[X_i] = m_1$$

$$\Rightarrow p = \sum_{i=1}^{n} X_i / n$$

$$\Rightarrow p = \bar{X}$$

$\therefore \hat{p}_{MOM} = \bar{X}$ is the method of moments estimator for p

For our given data, the ~~estim~~ MOM estimate will be,

$$\hat{p}_{MOM} = \bar{X} = \frac{0+1+1+0+0+1+1+0+0+1}{10} = \frac{5}{10} = \frac{1}{2}$$

$$\therefore \boxed{\hat{p}_{MOM} = 1/2 = \bar{X}}$$

Premium

Q1 b] Let's continue with our assumption of the ~~coin~~ data being sampled from a Bernoulli distribution with unknown parameter p.

$$\therefore X_i \sim Bernoulli(p)$$

now, likelihood of sample,

$$\mathcal{L}(p) = \prod_{i=1}^{n} p^{x_i}(1-p)^{1-x_i}$$

$$= p^{\sum_{i=1}^{n} x_i}(1-p)^{n-\sum_{i=1}^{n} x_i}$$

Taking log on both sides,

$$\log \mathcal{L}(p) = \sum_{i=1}^{n} x_i \log p + \left(n - \sum_{i=1}^{n} x_i\right)\log(1-p)$$

$\because$ log is an ↑ing f^n, maximising $\mathcal{L}(p)$ is same as maximising $\log \mathcal{L}(p)$

Let $\log \mathcal{L}(p) = l(p)$

$$\therefore l(p) = \sum_{i=1}^{n} x_i \log p + \left(n - \sum_{i=1}^{n} x_i\right)\log(1-p)$$

$$\frac{dl(p)}{dp} = \sum_{i=1}^{n} x_i / p + (-1)\left(n - \sum_{i=1}^{n} x_i\right) / (1-p)$$

$$\frac{d^2 l(p)}{dp^2} = -\sum_{i=1}^{n} x_i / p^2 + (-1)\left(n - \sum_{i=1}^{n} x_i\right) / (1-p)^2$$

Dt. Pg. B+

for maximising $l(p)$,

$$\frac{dl(p)}{dp} = 0$$

$$\Rightarrow \sum_{i=1}^{n} x_i \Big/ p + (-1)\left(n - \sum_{i=1}^{n} x_i\right) \Big/ (1-p) = 0$$

$$\Rightarrow (1-p)\sum_{i=1}^{n} x_i \cancel{+} - np + p\sum_{i=1}^{n} x_i = 0$$

$$\Rightarrow np = \sum_{i=1}^{n} x_i$$

$$\Rightarrow p = \sum_{i=1}^{n} x_i \Big/ n$$

Also, $\left.\frac{d^2 l(p)}{dp^2}\right|_{p=\sum_{i=1}^{n} x_i/n}$

$$= -\sum_{i=1}^{n} x_i \Big/ \left(\sum_{i=1}^{n} x_i / n\right)^2 + \frac{(-1)\left(n - \sum_{i=1}^{n} x_i\right)}{\left(1 - \frac{\sum_{i=1}^{n} x_i}{n}\right)^2}$$

$$= \frac{-n^2}{\sum_{i=1}^{n} x_i} + \frac{(-1)n}{\left(1 - \frac{\sum_{i=1}^{n} x_i}{n}\right)}$$

$$= \frac{-n}{\bar{x}} + \frac{(-1)n}{(1-\bar{x})}$$

$$= \frac{-n + \cancel{n\bar{x}} - \cancel{n\bar{x}}}{\bar{x}(1-\bar{x})} \qquad \left(\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}\right)$$

$$= \frac{-n}{\bar{x}(1-\bar{x})} < 0$$

[$\because \bar{x} \geq 0$ & $\bar{x}$ can't be 0 in this case as the ratio won't be defined]

$\therefore$ ~~$\hat{p} = \bar{x}$~~

Premium

Dt.

Pg.

B+

$\therefore \hat{p}_{MLE} = \bar{X}$ is the ~~ML~~ maximum likelihood estimator for p.

For our given data, the ~~#~~ MLE estimate is,

$$\hat{p}_{MLE} = \bar{X} = \frac{5}{10} = 1/2$$

$$\boxed{\therefore \hat{p}_{MLE} = \bar{X} = 1/2}$$

Q1] (c) The code is given in file 16217_1.R

The steps I followed were:

1. Write a ~~like~~ func^n to calculate the negative likelihood of a given sample of ~~B~~ data points sampled from a Bernoulli RV.

2. Then I made use of the "optim" routine in R to arrive at the maximum likelihood estimate for the given data. "optim" minimises a given objective func^n, hence it was given the NEGATIVE log likelihood as minimising it is same as maximising the normal log likelihood.

3. "optim" by default uses Nelder-Mead method for optimising routines.

4. The answer obtained was 0.5 which was equal to the theoretical value of MLE obtained

Dt.
Pg. B+

(d) As we have seen,

$$\hat{p}_{MOM} = \hat{p}_{MLE} = \bar{X} = 1/2$$

Hence, the estimate obtained by both methods, as well as by the ~~z~~ R code is the same.

Also,

$$E[\bar{X}] = E\left[\sum_{i=1}^{n} X_i/n\right] = \frac{1}{n} E\left[\sum_{i=1}^{n} X_i\right]$$

$$= \frac{1}{n} \sum_{i=1}^{n} E[X_i]$$

$$= \frac{1}{n} \sum_{i=1}^{n} p$$

$$= \frac{\cancel{n}p}{\cancel{n}}$$

$$= p$$

$$= E[X_i]$$

$$\Rightarrow E[\bar{X}] = E[X_i]$$

$$\Rightarrow E[\hat{p}_{MOM}] = E[\hat{p}_{MLE}] = E[\bar{X}] = E[X_i]$$

Thus, MLE & MOM both give unbiased estimators for p.

Dt.

Pg. B+

Q.2] (a) Given,

~~$X_1 \overset{iid}{\sim} f_{c,\sigma}(x)\left(\frac{x}{\sigma}\right)$~~

$$X_i \overset{iid}{\sim} f_{c,\sigma}(x) = \frac{c}{\sigma}\left(\frac{x}{\sigma}\right)^{c-1} e^{-(x/\sigma)^c}, x>0, c>0, \sigma>0$$

The unknown parameters, $\underset{\sim}{\theta} = (c, \sigma)$

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} \frac{c}{\sigma}\left(\frac{x_i}{\sigma}\right)^{c-1} e^{-(x_i/\sigma)^c},$$

$$= \left(\frac{c}{\sigma}\right)^n \prod_{i=1}^{n}\left(\frac{x_i}{\sigma}\right)^{c-1} e^{-\sum_{i=1}^{n}(x_i/\sigma)^c}$$

Taking log on both sides,

$$\log \mathcal{L}(\theta) = n\log\left(\frac{c}{\sigma}\right) + (c-1)\sum_{i=1}^{n}\log\left(\frac{x_i}{\sigma}\right) - \sum_{i=1}^{n}\left(\frac{x_i}{\sigma}\right)^c$$

$\because$ log is an $\uparrow$ing func^n, $\therefore$ maximising $\log \mathcal{L}(\theta)$ is same as maximising $\log \mathcal{L}(\theta)$.

Let $\log \mathcal{L}(\theta) = \ell(\theta)$.

$$\Rightarrow \ell(\theta) = n\log c - n\log\sigma + (c-1)\sum_{i=1}^{n}\log x_i - n(c-1)\log\sigma - \sum_{i=1}^{n}\left(\frac{x_i}{\sigma}\right)^c$$

$$\Rightarrow \ell(\theta) = n\log c - nc\log\sigma + (c-1)\sum_{i=1}^{n}\log x_i - \sum_{i=1}^{n}\left(\frac{x_i}{\sigma}\right)^c$$

Dt.

Pg. B+

~~d(ii)~~ for maximum,

$\frac{d\ell(\theta)}{dc} = 0$ & $\frac{d\ell(\theta)}{d\sigma} = 0$ [necessary cond^n for maximum]

⇒ First for σ,

$$\frac{d\ell(\theta)}{d\sigma} = 0$$

$$\Rightarrow 0 - \frac{nc}{\sigma} + 0 + c\sum_{i=1}^{n} \frac{(x_i)^c}{(\sigma)^{c+1}} = 0$$

$$\Rightarrow \cancel{c}\sum_{i=1}^{n} \frac{(x_i)^c}{(\sigma)^{c\cancel{+1}}} = \frac{n\cancel{c}}{\cancel{\sigma}} \quad [\because c > 0, \sigma > 0]$$

$$\Rightarrow \sum_{i=1}^{n} \frac{(x_i)^c}{(\sigma)^c} = n$$

$$\Rightarrow \sigma^c = \sum_{i=1}^{n} \frac{(x_i)^c}{n}$$

$$\Rightarrow \sigma = \left(\sum_{i=1}^{n} \frac{(x_i)^c}{n}\right)^{1/c}$$

Now, for c,

$$\frac{d\ell(\theta)}{dc} = 0$$

$$\Rightarrow \frac{n}{c} - n\log\sigma - \sum_{i=1}^{n}\log x_i - \sum_{i=1}^{n}\left(\frac{x_i}{\sigma}\right)^c \log\left(\frac{x_i}{\sigma}\right) = 0$$

This equation can only be solved iteratively using

Premium

Dt. ______ B+

Pg. ______

numerical analysis, hence, we leave the sol^n of this point.

Q2 (b) The R code for this part is in 16217_2.R.

The steps I followed are as follows:

① First, I wrote a func^n to calculate the negative log likelihood of a given sample of data sampled from a Weibull distribution.

② Next, I used "optim" to minimise the negative log-likelihood and hence maximise the log likelihood & likelihood.

For an initial starting point of $c=1, \sigma=2$, "optim" using Nelder-Mead method gave me an MLE for c, σ as,

$$\hat{c}_{MLE} = 5.951281$$
$$\hat{\sigma}_{MLE} = 229.3823$$

To confirm that my code is indeed correct, I plotted the negative log-likelihood around these values of c, σ using the "persp" func^n in R as well as an online plotter.

These plots are shown in the next page and as can be seen, they seem to reach their lowest point ~~around~~ at values very close to $(\hat{c}_{MLE}, \hat{\sigma}_{MLE})$ only.

Premium

Dt. ______ B+

Pg. ______

I plotted NLL for c ranging from 1 to 10 with a step of 0.5 & σ ranging from 225 to 235 with a step of 0.5.

Q3] $X_i \overset{iid}{\sim} \text{Poisson}(\tau_i)$
$Y_i \overset{iid}{\sim} \text{Poisson}(\beta\tau_i)$ } Given

(a) ~~We are given that x_1 is missing.~~

Let us approximate it by $x_1 = \tau_1$
First, let us consider the full dataset.
Now, we can write the likelihood of the complete data as following, $\underset{\sim}{Z} = (\underset{\sim}{X}, \underset{\sim}{Y})$

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} f_{\beta\tau_i}(y_i)\, f_{\tau_i}(x_i)$$

$$= \prod_{i=1}^{n} \frac{e^{-\beta\tau_i}(\beta\tau_i)^{y_i}}{y_i!} \cdot \frac{e^{-\tau_i}(\tau_i)^{x_i}}{x_i!}$$

~~Taking log on both sides,~~

~~$\log \mathcal{L}(\theta) = \ldots$~~

$$\Rightarrow \mathcal{L}(\theta) = \frac{e^{-(1+\beta)\sum_{i=1}^{n}\tau_i}\; \beta^{\sum_{i=1}^{n} y_i} \prod_{i=1}^{n} \tau_i^{(x_i+y_i)}}{\prod_{i=1}^{n} x_i!\, y_i!}$$

Taking log on both sides,

$$\log \mathcal{L}(\theta) = -(\beta+1)\sum_{i=1}^{n}\tau_i + \sum_{i=1}^{n} y_i(\log\beta) + \sum_{i=1}^{n}(x_i+y_i)\log\tau_i - \sum_{i=1}^{n}\log x_i! - \sum_{i=1}^{n}\log y_i!$$

Dt. ______ B+

Pg. ______

Let $\log L(\theta) = l(\theta)$

$$\Rightarrow l(\theta) = -(\beta+1)\sum_{i=1}^{n}\tau_i + \log\beta\sum_{i=1}^{n} y_i + \sum_{i=1}^{n}(x_i+y_i)\tau_i$$
$$-\sum_{i=1}^{n}\log x_i! - \sum_{i=1}^{n}\log y_i!$$

$\because$ log is an ↑ing func^n, maximising log likelihood is same as maximising likelihood.

Now, for maximising

$$\frac{dl(\theta)}{d\beta} = -\sum_{i=1}^{n}\tau_i + \frac{\sum_{i=1}^{n} y_i}{\beta} + 0 - 0 - 0$$

$$\frac{d^2 l(\theta)}{d\beta^2} = 0 + \frac{\left(-\sum_{i=1}^{n} y_i\right)}{\beta^2}$$

$$\frac{dl(\theta)}{d\tau_j} = -(\beta+1)\tau_j + 0 + (x_j+y_j) - 0 - 0$$

$$\frac{d^2 l(\theta)}{d\tau_j^2} = -(\beta+1)$$

$\Rightarrow$ For maximum,

$$\frac{dl(\theta)}{d\beta} = 0 \Rightarrow \beta = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n}\tau_i} \quad \text{—} ①$$

$$\frac{dl(\theta)}{d\tau_j} = 0 \Rightarrow \tau_j = \frac{x_j+y_j}{\beta+1} \quad \forall j = 1, 2, \dots, n \quad \text{—} ②$$

~~① & ② ⇒ β~~

Premium

Dt. B+
Pg.

now, $\frac{d^2 l(\theta)}{d\tau_j^2} < 0 \quad \forall\ \tau_j \in \mathbb{R} \quad \left[\because \beta \geq 0 \right]$

~~Now, $d^2 l$~~

Now, $\cancel{\frac{d^2 l(\theta)}{d\beta^2}\Bigg|_{\hat{\beta}}}$

Also, from ① & ②,

$$\beta = \frac{\sum_{i=1}^{n} y_i\ (\beta+1)}{\sum_{i=1}^{n}(x_i+y_i)}$$

$$\Rightarrow \beta\left(\frac{\sum_{i=1}^{n} x_i}{\cancel{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}}\right) = \frac{\sum_{i=1}^{n} y_i}{\cancel{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}}$$

$$\Rightarrow \beta = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i} \quad \text{—— ③}$$

now, $$\frac{d^2 l(\theta)}{d\beta^2}\Bigg|_{\beta = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i}} = \frac{-\cancel{\sum_{i=1}^{n} y_i}\ \left(\sum_{i=1}^{n} x_i\right)^2}{\cancel{\left(\sum_{i=1}^{n} y_i\right)^2}}$$

$$= \frac{-\sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} y_i} < 0$$

~~$\because x_i \geq 0\ \forall i,\ y_i \geq 0\ \forall i$~~

$\left[\because \sum_{i=1}^{n} x_i > 0,\ \sum_{i=1}^{n} y_i > 0 \right]$

Dt. ______ B+

Pg. ______

Also, let's consider the Hessian of the likelihood funcⁿ.

$$H = \begin{bmatrix} \frac{\partial^2 l(\theta)}{\partial \beta^2} & \frac{\partial^2 l(\theta)}{\partial \beta \partial \tau_1} & \frac{\partial^2 l(\theta)}{\partial \beta \partial \tau_2} & \cdots & \frac{\partial^2 l(\theta)}{\partial \beta \partial \tau_n} \\ \frac{\partial^2 l(\theta)}{\partial \tau_1 \partial \beta} & \frac{\partial^2 l(\theta)}{\partial \tau_1^2} & \frac{\partial^2 l(\theta)}{\partial \tau_1 \partial \tau_2} & \cdots & \frac{\partial^2 l(\theta)}{\partial \tau_1 \partial \tau_n} \\ \vdots & & \ddots & & \vdots \\ \frac{\partial^2 l(\theta)}{\partial \tau_n \partial \beta} & \frac{\partial^2 l(\theta)}{\partial \tau_n \partial \tau_1} & \cdots & \cdots & \frac{\partial^2 l(\theta)}{\partial \tau_n^2} \end{bmatrix}_{(n+1)\times(n+1)}$$

$$\Rightarrow H = \begin{bmatrix} -\sum_{i=1}^{n} y_i/\beta^2 & 0 & 0 & \cdots & 0 \\ -\tau_1 & -(\beta+1) & 0 & \cdots & 0 \\ -\tau_2 & 0 & -(\beta+1) & 0 \cdots & 0 \\ \vdots & \vdots & & \ddots & 0 \\ \vdots & & & & \vdots \\ -\tau_n & 0 & \cdots & \cdots & -(\beta+1) \end{bmatrix}$$

Clearly, the hessian is a lower triangular matrix

Hence, $|H|$ = product of the diagonal elements

$$= -\frac{\sum_{i=1}^{n} y_i}{\beta^2} (\beta+1)^n (-1)^n$$

Premium

Dt.
Pg.
B+

$$\Rightarrow |H| = (-1)^{n+1} \sum_{i=1}^{n} y_i \frac{(\beta+1)^n}{\beta^2}$$

~~Also~~ Clearly, $\sum_{i=1}^{n} y_i \frac{(\beta+1)^n}{\beta^2} \geqslant 0 \quad \forall \cancel{\beta}, \because \beta > 0$ & $y_i \geqslant 0$

and in our case, ~~As~~ $n = 18$ is even,
$\Rightarrow (-1)^{n+1} = (-1)^{19} = (-1)$

$\Rightarrow |H| < 0 \qquad \forall \beta$ & $\therefore$ also for $\beta = \hat{\beta}_{MLE}$
$\Rightarrow H$ is $-$ve definite

$$\therefore \hat{\beta}_{MLE} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i} \quad -(4) \qquad , \qquad \hat{\tau}_{j\,MLE} = \frac{x_j + y_j}{\hat{\beta}_{MLE}+1} \quad j = 1, 2, \ldots, n \quad -(5)$$

are the Maximum likelihood estimates for ~~$\hat{\beta}_{MLE}$ & $\hat{\tau}_{j\,MLE}$~~ β & τ_j, $j \in \{1, 2, \ldots, n\}$

~~Now, using these closed form expressions, we run the EM algorithm using the following pseudocode.~~

Now, for incomplete data, when x_1 is missing, we have the incomplete data likelihood, given by,

$$\tilde{\mathcal{L}}^{m}(\theta) = \left(\prod_{i=2}^{n} \frac{e^{-(\beta+1)\tau_i} (\beta)^{y_i} (\tau_i)^{(x_i+y_i)}}{x_i!\, y_i!} \right) \cdot \frac{e^{-\beta\tau_1} (\beta\tau_1)^{y_1}}{y_1!}$$

Premium

Dt.
Pg.

B+

$$\tilde{\ell}(\theta) = \log \tilde{L}(\theta)$$

$$= -(\beta+1)\sum_{i=2}^{n} \tau_i + \log\beta \sum_{i=2}^{n} y_i + \sum_{i=2}^{n}(x_i+y_i)\log\tau_i$$

$$-\sum_{i=1}^{n}\log y_i! - \sum_{i=2}^{n}\log x_i!$$

$$-\beta\tau_1 + y_1\log\beta\tau_1$$

Differentiating to get MLE equations, we have,

$$\frac{d\tilde{\ell}(\theta)}{d\beta} = -\sum_{i=1}^{n}\tau_i + \frac{\sum_{i=2}^{n} y_i}{\beta} + 0 - 0 - 0 + \frac{y_1}{\beta} = 0$$

$$\Rightarrow \hat{\beta}_{MLE} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n}\tau_i}$$

$$\frac{d\tilde{\ell}(\theta)}{d\tau_j} = -(\beta+1) + \frac{x_j+y_j}{\tau_j} = 0, \quad j = 2,3,\ldots,n$$

$$\Rightarrow \hat{\tau}_{j_{MLE}} = \frac{x_j+y_j}{\hat{\beta}_{MLE}+1}, \quad j = 2,3,\ldots,n$$

$$\frac{d\tilde{\ell}(\theta)}{d\tau_1} = -\beta + \frac{y_1}{\tau_1} = 0$$

$$\Rightarrow \hat{\tau}_{1_{MLE}} = \frac{y_1}{\hat{\beta}_{MLE}}$$

Dt.
Pg.
B+

Hence, we have;

(1) $\hat{\tau}_{1_{MLE}} = \dfrac{y_1}{\hat{\beta}_{MLE}}$,

(2) $\hat{\tau}_{j_{MLE}} = \dfrac{x_j + y_j}{\hat{\beta}_{MLE}+1}$, $j = 2, 3, \ldots, n$

(3) $\hat{\beta}_{MLE} = \dfrac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} \hat{\tau}_{i_{MLE}}}$.

We use these equations in our EM algorithm.

~~We use the~~

(1) First we initialise the τ_i's & β to some value.

(2) Then, ~~we~~ in the loop, we first update $\underset{\sim}{\theta}$ using the equations we just derived.

(3) Then, we update the missing data, using $x_1 = \tau_1$

(4) Keep repeating (2), (3) till convergence is reached.

Premium

Dt. ______ B+

Pg. ______

(b) In this case, y_1 is missing $\therefore$ Incomplete data likelihood in this case will be given by,

$$\tilde{\mathcal{L}}(\theta) = \left(\prod_{i=2}^{n} \frac{e^{-(\beta+1)\tau_i}(\beta)^{y_i}(\tau_i)^{(x_i+y_i)}}{x_i!\, y_i!}\right) \cdot \frac{e^{-\tau_1}(\tau_1)^{x_1}}{x_1!}$$

$$\tilde{\ell}(\theta) = \log \tilde{\mathcal{L}}(\theta)$$

$$= -(\beta+1)\sum_{i=2}^{n}\tau_i + \log\beta\sum_{i=2}^{n} y_i + \sum_{i=2}^{n}(x_i+y_i)\log\tau_i$$

$$-\sum_{i=1}^{n}\log x_i! - \sum_{i=2}^{n}\log y_i!$$

$$-\tau_1 + x_1\log\tau_1$$

Differentiating to get MLE equations, we get,

$$\frac{d\tilde{\ell}(\theta)}{d\beta} = -\sum_{i=2}^{n}\tau_i + \sum_{i=2}^{n} y_i/\beta + 0 - 0 - 0 - 0 + 0 = 0$$

$$\Rightarrow \hat{\beta}_{MLE} = \frac{\sum_{i=2}^{n} y_i}{\sum_{i=2}^{n}\tau_i}$$

$$\frac{d\tilde{\ell}(\theta)}{d\tau_j} = -(\beta+1) + 0 + \frac{x_j+y_j}{\tau_j} = 0 \quad ; j = 2,3,\dots,n$$

$$\Rightarrow \hat{\tau}_{j_{MLE}} = \frac{x_j+y_j}{\hat{\beta}_{MLE}+1}, \quad j = 2,3,\dots,n$$

$$\frac{d\tilde{\ell}(\theta)}{d\tau_1} = -1 + \frac{x_1}{\tau_1} = 0 \Rightarrow \hat{\tau}_{1_{MLE}} = x_1$$

Premium

Dt. ______ B+
Pg. ______

Hence, we have,

(1) $$\hat{\beta}_{MLE} = \frac{\sum_{i=2}^{n} y_i}{\sum_{i=2}^{n} \tau_i}$$

(2) $$\hat{\tau}_{j\,MLE} = \frac{x_j + y_j}{\hat{\beta}_{MLE} + 1}, \quad j = 2, \ldots, n$$

(3) $$\hat{\tau}_{1\,MLE} = \frac{y_1}{\hat{\beta}_{MLE}}$$

Now, using these update rules, we keep repeating the EM algorithm just as described in part (a) until convergence is reached.

b COMPARISON:

The complete values of the parameters are given in the compiled code at the end of the report.

I am discussing only a few interesting ones here.

~~(a) For complete data,~~ $\tau_1 = 3540.021$, $x_1 = 3540$

~~For incomplete data,~~ $\tau_1 = 3567.273$
$x_1 = \tau_1 = 3567.273$

Dt.
Pg. B+

~~As you can see, the initial vers~~

	Complete Data	Incomplete Data (x_1 is missing)	Incomplete Data (y_1 is missing)
y_1	3	3	2.977
x_1	3540	3567.273	3540
τ_1	3540.021	3567.273	3540
β	8.41×10^{-4}	8.40×10^{-4}	8.40×10^{-4}

As we can see, when x_1 was missing, EM was able to estimate x_1 upto an error of 27 ($|3567-3540|$)

Similarly, when y_1 was missing, EM estimated the value of y_1 to be 2.977, an absolute error of 0.023 ($|3-2.977|$).

β, on the other hand, hardly changes in any case.

16217_1.R

justachetan

Fri Mar 1 14:46:16 2019

```
# Author: Aditya Chetan
# Roll Number: 2016217

# Question 1

# Assumption: The unknown distribution is assumed
# to be a Bernoulli distribution with unknown
# parameter, theta.



# First I declare a function to calculate the
# likelihood of a given sample. The reason
# for this is to make use of the ```optim```
# command. More on this in a bit.
#
# This function ```bern_ll``` calculates the
# likeihood of a given sample given theta.
# Here, ```theta``` is unknown parameter and
# ```y``` is the given data sample.
bern_lhd<-function(theta, y) {

    # Assumption:
    #       y[i] = 1, w.p. theta
    #   &   y[i] = 0, w.p. (1 - theta)

    # ```ll``` will hold the sample likelihood.
    # Initialising it with 1 here.
    lhd <- 1

    for (i in y) {
        if (i == 1) {
            # Multiplying in accordance with assumed pdf of Bernoulli
            lhd <- lhd * theta
        }
        else{
            # Multiplying in accordance with assumed pdf of Bernoulli
            lhd <- lhd * (1 - theta)
        }
    }

    # Since ```optim``` minimises by default, multiply ```ll``` with -1
    # so as to get parameters that minimise negative likelihood
```

```
    # or maximise ```ll``` to get Maximum Likelihood Estimate
    return(-lhd)

}

# Defining the sample as given in the question
y <- c(0, 1, 1, 0, 0, 1, 1, 0, 0, 1)

# Using ```optim``` to maximise likelihood.
# This function call returns an object. The
# maximum value of parameter theta can be accessed
# using $par.
#
# ```optim``` by default uses Neldor-Mead method
# for optmisation
p <-optim(1, bern_lhd, y=y)
```

```
## Warning in optim(1, bern_lhd, y = y): one-dimensional optimization by Nelder-Me
ad is unreliable:
## use "Brent" or optimize() directly
```

```
cat("Value of MLE:", p$par, "\n")
```

```
## Value of MLE: 0.5
```

```
cat("Theoretical value of MLE: 0.5\n")
```

```
## Theoretical value of MLE: 0.5
```

16217_2.R

justachetan

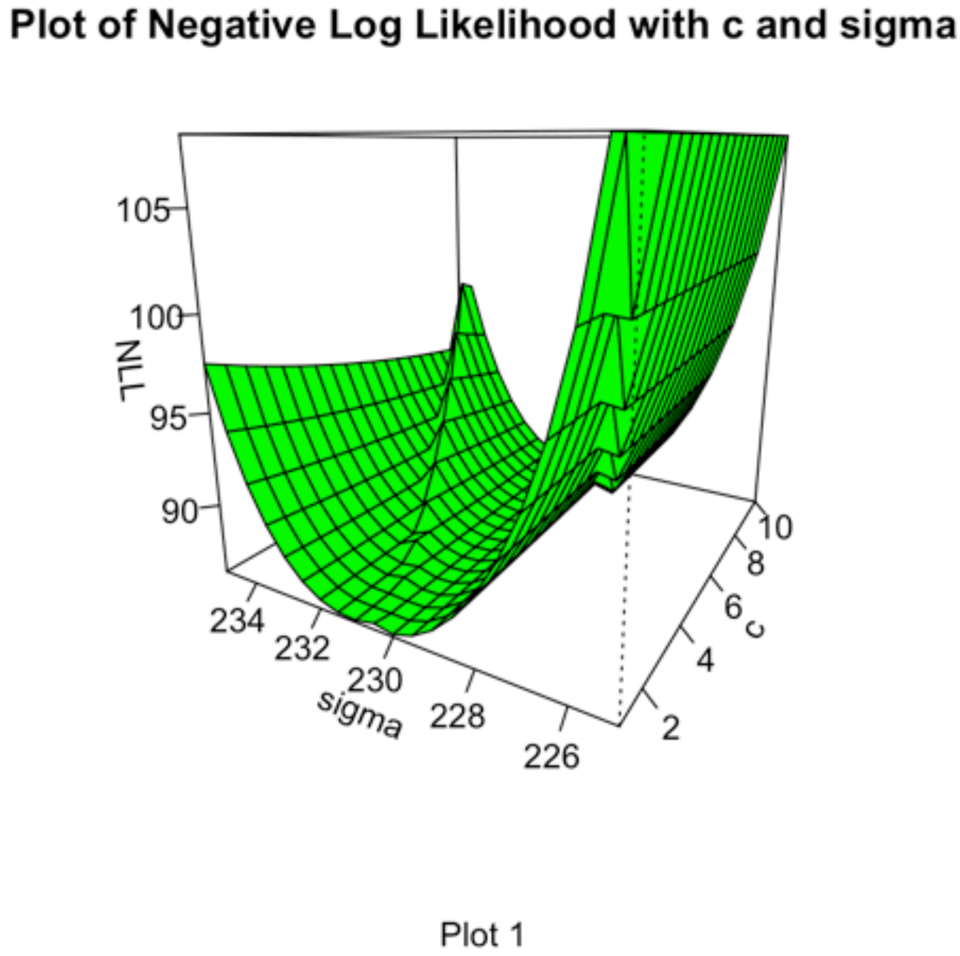
Fri Mar 1 14:47:58 2019

```
# Author: Aditya Chetan
# Roll Number: 2016217

# Question 2


# First I declare a function to calculate the
# -ve log-likelihood of a given sample. The reason
# for this is to make use of the ```optim```
# command. More on this in a bit.
#
# This function ```weibull_ll``` calculates the
# -ve log-likeihood of a given sample given theta.
# Here, ```theta```,  is unknown parameter and
# ```y``` is the given data sample.
weibull_nll<-function(theta, y) {

    # ```theta``` is a vector here which contains both
    # the unknown parameters
    c <- theta[1]
    sigma <- theta[2]

    # ```ll``` will hold the sample log-likelihood
    # Initialising it with 0 here.
    ll <- 0

    for (i in y) {

        ll <- ll + log(c / sigma) + ((c - 1) * log(i / sigma)) - ((i / sigma) ** c
)

    }

    # Since ```optim``` minimises by default, multiply ```ll``` with -1
    # so as to get parameters that minimise negative log-likelihood
    # or maximise ```ll``` to get Maximum Likelihood Estimate
    return(-ll)

}

# Defining the sample as given in the question
y <- c(143, 164, 188, 188, 190, 192, 206, 209, 213, 216, 220, 227, 230, 246, 265,
304, 234)
```

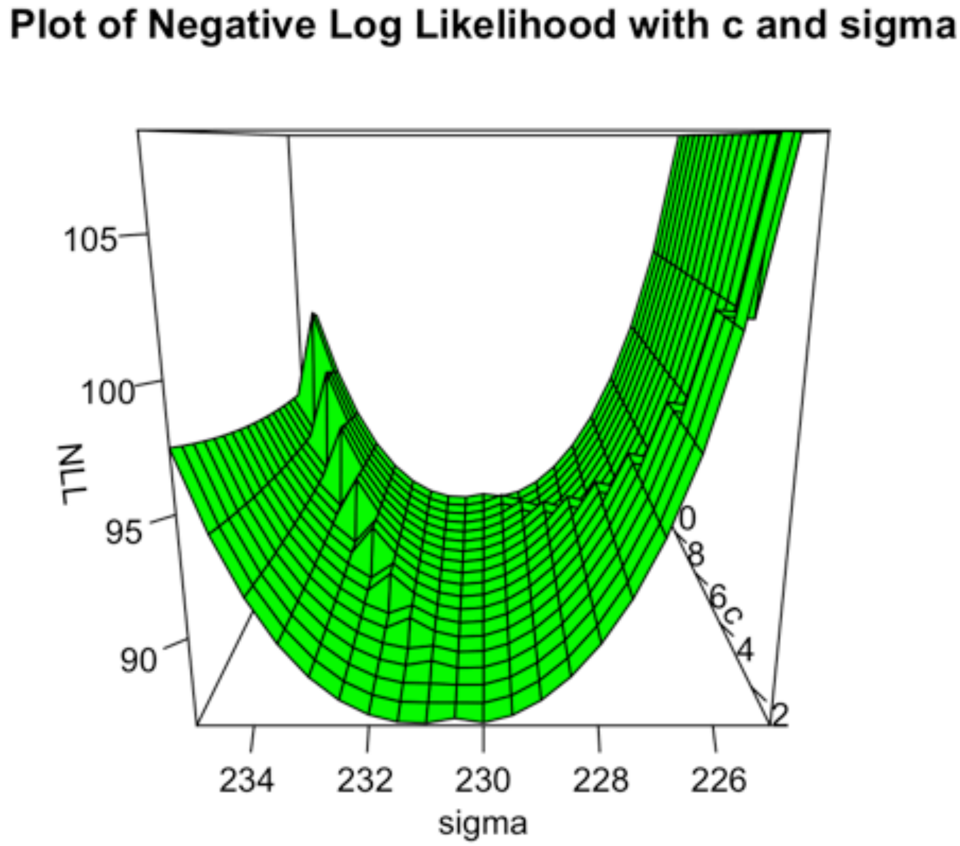
```
# Using ```optim``` to minimise -ve log-likelihood.
# This function call returns an object. The
# maximum value of parameter theta can be accessed
# using $par.
#
# ```optim``` by default uses Neldor-Mead method
# for optmisation
p <-optim(c(1,2), weibull_nll, y=y)


cat("Value of MLE:\n\tc\t= ", p$par[1], "\n\tsigma\t= ", p$par[2],"\n")
```

```
## Value of MLE:
##  c   =  5.951281
##  sigma   =  229.3823
```

```
# Ranges to c and sigma to check minimum on
cs <- seq(1, 10, 0.5)
sigmas <- seq(225, 235, 0.5)

# Code used to get data for plotting

z <- c()

for (i in cs) {
    for (j in sigmas) {
        z <- c(z, weibull_nll(c(i, j), y))
        # cat(i, ',', j, ',', weibull_nll(c(i, j), y), '\n') # Used when I wanted
a .csv of the points
    }
}

# z[i, j] if the NLL of sample with c = cs[i] and sigma = sigmas[j]
z <- matrix(z, length(cs), length(sigmas))

persp(cs, sigmas, z, col='green', theta = -60, ticktype = 'detailed', xlab='c', yl
ab='sigma', zlab='NLL', main = "Plot of Negative Log Likelihood with c and sigma",
sub ="Plot 1")
```




```
persp(cs, sigmas, z, col='green', theta = -90, ticktype = 'detailed', xlab='c', yl
ab='sigma', zlab='NLL', main = "Plot of Negative Log Likelihood with c and sigma",
sub ="Plot 2")
```


Plot of Negative Log Likelihood with c and sigma
105
100
95
90
NLL
234
232
230
228
226
sigma
0
8
6
c
4
2

Plot 2

16217_3.R

justachetan

Fri Mar 1 14:48:20 2019

```
# Author: Aditya Chetan
# Roll Number: 2016217

# Question 3


# We start with initialising the taus and beta
theta <- rep(2, 19)

# Next, we store the data provided in the question

x <- c(3540, 3560, 3739, 2784, 2571, 2729, 3952, 993, 1908, 948, 1172, 1047, 3138,
5485, 5554, 2943, 4969, 4828)
y <- c(3, 4, 1, 1, 3, 1, 2, 0, 2, 0, 1, 3, 5, 4, 6, 2, 5, 4)

# Part (a)
cat("Part (a) Complete Data\n-------------------------------------------\n\n")
```

```
## Part (a) Complete Data
## -------------------------------------------
```

```
# For complete data
# Equations used here have been derived in the report
beta_complete = sum(y) / sum(x)
tau_complete = (x + y) / (beta_complete + 1)

cat("Value of MLE for complete data:\n\tbeta\t=\t", beta_complete,'\n')
```

```
## Value of MLE for complete data:
##  beta    =    0.0008413892
```

```
for (i in seq(1, length(tau_complete))) {
    cat("\ttau",i,"\t=\t", tau_complete[i], '\n')
}
```

```
##  tau 1   =    3540.021
##  tau 2   =    3561.004
##  tau 3   =    3736.856
##  tau 4   =    2782.659
##  tau 5   =    2571.836
##  tau 6   =    2727.705
##  tau 7   =    3950.676
##  tau 8   =    992.1652
##  tau 9   =    1908.394
##  tau 10  =    947.203
##  tau 11  =    1172.014
##  tau 12  =    1049.117
##  tau 13  =    3140.358
##  tau 14  =    5484.385
##  tau 15  =    5555.326
##  tau 16  =    2942.524
##  tau 17  =    4969.818
##  tau 18  =    4827.938
```

```
cat("\nPart (a) Incomplete Data (x_1 is missing)\n--------------------------------
-----------\n\n")
```

```
##
## Part (a) Incomplete Data (x_1 is missing)
## -------------------------------------------
```

```
# For incomplete data
# Equations used here for EM have again been derived in the report

# x_1 from the data will be missing here
x_m <- x[2:length(x)]


# Therefore, the incomplete data can be defined as follows:
z <- c(x_m, y)


# Setting the limit of the number of iterations for EM
max_iter <- 10000

# Tolerance of error for EM
tol <- 10^-12

# Difference between scores in two iterations
diff = 1

# Variable to store the parameters during the iterations
theta_em <- theta
```

```r
# Assumptions for EM: I am considering the definition of convergence
# to be either when ```max_iter``` iterations are complete or when the
# L2 norm of the difference between the parameter vectors goes below
# ```tol```, whichever occurs first. The equations used here have been
# derived in the report
for (i in seq(1, max_iter)) {

    # Store parameters in the previous iteration
    theta_prev <- theta_em

    # If difference between iterations is less than
    # tolerance, then stop
    if (diff < tol) {
        break
    }

    # beta being updated
    theta_em[1] <- sum(y) / sum(theta_em[2:length(theta_em)])

    # tau_1 being updated
    theta_em[2] <- y[1] / theta_em[1]

    # tau_j, j = 2, 3, 4 ... ,n being updated
    theta_em[3: length(theta_em)] <- (x[2: length(x)] + y[2:length(y)]) / (theta_e
m[1] + 1)

    # Calculate the difference in parameters
    diff <- sqrt(sum((theta_em - theta_prev)))

}

cat("Value of MLE for incomplete data:\n\tbeta\t=\t", theta_em[1],'\n')
```

```
## Value of MLE for incomplete data:
##  beta    =    0.0008409786
```

```r
for (i in seq(1, length(theta_em) - 1)) {
    cat("\ttau",i,"\t=\t", theta_em[i + 1], '\n')
}
```

```
##  tau 1   =    3567.273
##  tau 2   =    3561.005
##  tau 3   =    3736.857
##  tau 4   =    2782.66
##  tau 5   =    2571.837
##  tau 6   =    2727.706
##  tau 7   =    3950.678
##  tau 8   =    992.1656
##  tau 9   =    1908.395
##  tau 10  =    947.2034
##  tau 11  =    1172.014
##  tau 12  =    1049.118
##  tau 13  =    3140.359
##  tau 14  =    5484.388
##  tau 15  =    5555.328
##  tau 16  =    2942.525
##  tau 17  =    4969.82
##  tau 18  =    4827.94
```

```
cat("\nEstimated value of x_1\t=\ttau_1\t=\t", theta_em[2], "\n\n")
```

```
##
## Estimated value of x_1   =   tau_1   =    3567.273
```

```
# Part (b) y_1 is missing

cat("\nPart (b) Incomplete Data (y_1 is missing)\n--------------------------------
-----------\n\n")
```

```
##
## Part (b) Incomplete Data (y_1 is missing)
## -------------------------------------------
```

```
# Resetting the difference
diff <- 1

# Removing y_1 from consideration
y_m <- y[2 : length(y)]

# Incomplete data in this case will be as shown below
z2 <- c(x, y_m)

# Variable to store the parameters during the iterations
theta_em2 <- theta

# EM algorithm proceeds as previously. The derivation
# of the equations for this part is shown in the report
for (i in seq(1, max_iter)) {

    # Store parameters in the previous iteration
    theta_prev <- theta_em2

    # If difference between iterations is less than
    # tolerance, then stop
    if (diff < 10 ** -12) {
        break
    }

    # beta being updated
    theta_em2[1] <- sum(y[2 : length(y)]) / sum(theta_em2[3:length(theta_em2)])

    # tau_1 being updated
    theta_em2[2] <- x[1]

    # tau_j, j = 2, 3, 4 ... ,n being updated
    theta_em2[3: length(theta_em2)] <- (x[2: length(x)] + y[2:length(y)]) / (theta
_em2[1] + 1)

    # Calculate the difference in parameters
    diff <- sqrt(sum((theta_em - theta_prev)))

}


cat("Value of MLE for incomplete data:\n\tbeta\t=\t", theta_em2[1],'\n')
```

```
## Value of MLE for incomplete data:
##  beta    =    0.0008409786
```

```r
for (i in seq(1, length(theta_em2) - 1)) {
    cat("\ttau",i,"\t=\t", theta_em2[i + 1], '\n')
}
```

```
##  tau 1   =    3540
##  tau 2   =    3561.005
##  tau 3   =    3736.857
##  tau 4   =    2782.66
##  tau 5   =    2571.837
##  tau 6   =    2727.706
##  tau 7   =    3950.678
##  tau 8   =    992.1656
##  tau 9   =    1908.395
##  tau 10  =    947.2034
##  tau 11  =    1172.014
##  tau 12  =    1049.118
##  tau 13  =    3140.359
##  tau 14  =    5484.388
##  tau 15  =    5555.328
##  tau 16  =    2942.525
##  tau 17  =    4969.82
##  tau 18  =    4827.94
```

```r
cat("\nEstimated value of y_1\t=\tbeta * tau_1\t=\t", theta_em[1] * theta_em2[2],"
\n\n")
```

```
##
## Estimated value of y_1   =   beta * tau_1    =    2.977064
```