Mini-Project 3

Group Members: Sathya Pooja Ramireddy (sxr176830)

Sashidhar Donthiri(sxd173730)

Contributions:

Both of us have individually tried solving the problems, and came up solutions, further discussed and put it together.

1)

(a) **Calculate Mean squared error of an estimator using Monte Carlo(MC) Simulation**:

Given $\hat{\boldsymbol{\theta}}$ be our estimator of parameter *ϑ(theta)* can be computed using MC simulation by repeating the process of simulating a sample of data and then computing $\hat{\boldsymbol{\theta}}$ from the obtained sample quite a large number of times and then computing the average of squared deviation values between $\hat{\boldsymbol{\theta}}$ and *ϑ(theta)* gives the value for mean squared error of an estimator using Monte Carlo Simulation.

(b) Given combination of (n, ϑ), $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ can be computed using Monte Carlo Simulations with N=1000 replications from independent and identical data (same data).

Assume n=1, min=0, max=5, theta=5

R-code:

```
#given (n,θ), compute mean squared errors(MSE) using Monte Carlo Simulation

#since we have to compute two estimated by simulating one sample

x_val=runif(1,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviation of theta hat from theta
```

17.96177 12.08442 #MSE for theta hat 1 and theta hat 2

(c) For different combinations of n and theta:

R-code followed by their results:

```
#nvalues= 1

#thetavalues=1,5,50,100

x_val=runif(1,0,1) #n,min,max
```

```
Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0.35702 0380365 #result


x_val=runif(1,0,50) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta

1685.252  1030.644 #result


x_val=runif(1,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 0.0584219 203995630#result


x_val=runif(1,0,100) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 2894.130    85.64365 #result
```

```
#nvalues= 2

#thetavalues=1,5,50,100



x_val=runif(2,0,1) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 .00119610   .08536198#result



x_val=runif(2,0,50) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 213.9171    309.8031#result


x_val=runif(2,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta



x_val=runif(2,0,100) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)
```

```
c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta

2078.8323      526.5876#result



#nvalues= 3

#thetavalues=1,5,50,100



x_val=runif(3,0,1) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0,04284603     0.0208220#result



x_val=runif(3,0,50) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta

253.07461    75.31336 #result


x_val=runif(3,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta
```

```
0.229658     4.474285 #result

x_val=runif(3,0,100) #n,min,max
Max_Likelihood_est=max(x_val)
moment_Val=2*mean(x_val)
c(Max_Likelihood_est,moment_Val)
#partial result
#since the replication val is 1000,n=1000
estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))
rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta
474.397670     5.032389 #result

#nvalues= 5
#thetavalues=1,5,50,100

x_val=runif(5,0,1) #n,min,max
Max_Likelihood_est=max(x_val)
moment_Val=2*mean(x_val)
c(Max_Likelihood_est,moment_Val)
#partial result
#since the replication val is 1000,n=1000
estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))
rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta
0.00528907      0.007097856 #result

x_val=runif(5,0,50) #n,min,max
Max_Likelihood_est=max(x_val)
moment_Val=2*mean(x_val)
c(Max_Likelihood_est,moment_Val)
#partial result
#since the replication val is 1000,n=1000
estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))
rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta
3.155901   225.769548#result

x_val=runif(5,0,5) #n,min,max
```

```
Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 0.74560748   0.03034125#result




x_val=runif(5,0,100) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 434.43983      81.87589#result




#nvalues= 10

#thetavalues=1,5,50,100



x_val=runif(10,0,1) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0.000196453   0.094073992#result



x_val=runif(10,0,50) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)
```

```
c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 3.715778   0.08248259#result


x_val=runif(10,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 0.1297616 0.7885489 #result




x_val=runif(10,0,100) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 367.95468   0.2592791#result



#nvalues= 30

#thetavalues=1,5,50,100



x_val=runif(30,0,1) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))
```

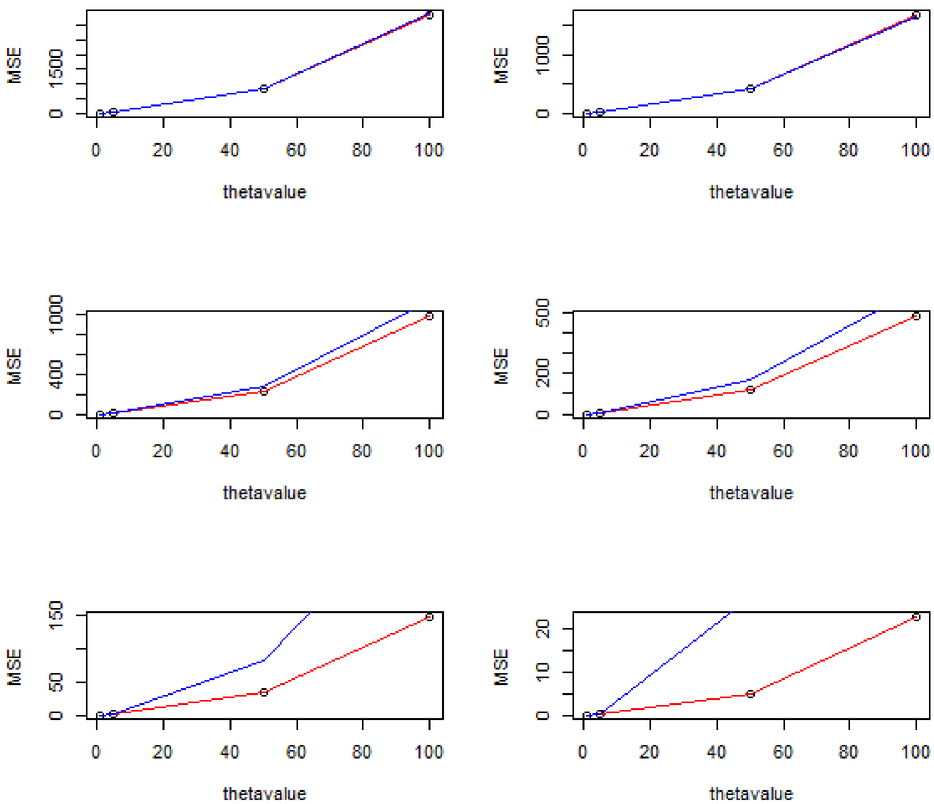
```
rowMeans((estimated_value-1)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0.0002551631   0.0161627948#result



x_val=runif(30,0,50) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-50)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0.2445082     138.617698 #result


x_val=runif(30,0,5) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-5)^2) #computing MSE by finding the deviatiosn of theta hat from theta

0.001096449  0.587780490 #result



x_val=runif(30,0,100) #n,min,max

Max_Likelihood_est=max(x_val)

moment_Val=2*mean(x_val)

c(Max_Likelihood_est,moment_Val)

#partial result

#since the replication val is 1000,n=1000

estimated_value=replicate(1000,c(Max_Likelihood_est,moment_Val))

rowMeans((estimated_value-100)^2) #computing MSE by finding the deviatiosn of theta hat from theta

 2.394698  48.662957            #result
```

R-code for graphical representation:

```
par(mfrow=c(3,2))

#a function to above written code in a simpler form to calculate MLE and MOME

fun.mle=function(nvalue,thetavalue){

  mean(replicate(1000,((max(runif(nvalue,0,thetavalue)))-thetavalue)**2))

}

fun.mome=function(nvalue,thetavalue){

  mean(replicate(1000,((2*mean(runif(nvalue,0,thetavalue)))-thetavalue)**2))

}

#calcculate MLE and MOM for each paramter theta at n=1

val.mle=c(fun.mle(1,1),fun.mle(1,5),fun.mle(1,50),fun.mle(1,100))

val.mome=c(fun.mome(1,1),fun.mome(1,5), fun.mome(1,50),fun.mome(1,100))

val.mle

val.mome

#graphical representation

plot(thetavalue,val.mle,ylab="MSE")

#Red for MLE and Blue for MOME

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")


#n=2

val.mle=c(fun.mle(2,1),fun.mle(2,5),fun.mle(2,50),fun.mle(2,100))

val.mome=c(fun.mome(2,1),fun.mome(2,5), fun.mome(2,50),fun.mome(2,100))

val.mle

val.mome

plot(thetavalue,val.mle,ylab="MSE")

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")


#n=3

val.mle=c(fun.mle(3,1),fun.mle(3,5),fun.mle(3,50),fun.mle(3,100))

val.mome=c(fun.mome(3,1),fun.mome(3,5), fun.mome(3,50),fun.mome(3,100))

val.mle

val.mome

plot(thetavalue,val.mle,ylab="MSE")

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")


#n=5

val.mle=c(fun.mle(5,1),fun.mle(5,5),fun.mle(5,50),fun.mle(5,100))
```

```
val.mome=c(fun.mome(5,1),fun.mome(5,5), fun.mome(5,50),fun.mome(5,100))

val.mle

val.mome

plot(thetavalue,val.mle,ylab="MSE")

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")


#n=10

val.mle=c(fun.mle(10,1),fun.mle(10,5),fun.mle(10,50),fun.mle(10,100))

val.mome=c(fun.mome(10,1),fun.mome(10,5), fun.mome(10,50),fun.mome(10,100))

val.mle

val.mome

plot(thetavalue,val.mle,ylab="MSE")

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")


#n=30

val.mle=c(fun.mle(30,1),fun.mle(30,5),fun.mle(30,50),fun.mle(30,100))

val.mome=c(fun.mome(30,1),fun.mome(30,5), fun.mome(30,50),fun.mome(30,100))

val.mle

val.mome

plot(thetavalue,val.mle,ylab="MSE")

lines(lowess(thetavalue,val.mle),col="red")

lines(lowess(thetavalue,val.mome),col="blue")
```

Below graph represents graphical representation of MSE with parameter theta 1 and theta 2 over n=1,2,3,5,10,30 in order. MLE-“Red” and MOME-“Blue”, thetavalue->parameter



(d) **Observations**:

- From above results we can observe that, the MSE's of both the estimators decrease as n increases, which means that as estimators become more accurate as n increases.
- They converge to the true parameter value in the given limit where MSE can be 0.
- It can also be seen that for n=1, MSEs of both estimators are almost same and when value of n is increased, like n>=5, MSE for MLE (theta 1) lies below that of MOME (theta 2).
- So, we can say that n=1, the two estimators hold good importance, but n>=5, the MLE is better when compared.

Thus, from graphs we can say that both MSEs are almost same for n=1,2 and for n>2, the ML Estimator is better than MOME.

Q2)

(a) Derive expression for MLE of theta :

(2) (a) $f(x) = \begin{cases} \dfrac{\theta}{x^{\theta+1}}, & x \geq 1 \\ 0 & x < 1 \end{cases}$

For $x = 1, 2, 3 \ldots n$, $\prod_{i=1}^{n} \dfrac{\theta}{x^{\theta+1}} \rightarrow (A)$

To find Maximum likelihood function,

we apply log to (A)

$$\Rightarrow \log\left[\prod_{i=1}^{n} \frac{\theta}{x^{\theta+1}}\right] \Rightarrow \sum_{i=1}^{n} \log \frac{\theta}{x^{\theta+1}}$$

$$\Rightarrow \sum_{i=1}^{n} \log\theta - \log x^{\theta+1}$$

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

Differentiate wrt θ & equate to '0'

$$\Rightarrow \frac{n}{\theta} - \left[\sum_{i=1}^{n}\left(\frac{1}{\cancel{x_i^{\theta+1}}}\left(\cancel{x_i^{\theta+1}}\right)\log(x_i)\right)\right] = 0$$

$$\Rightarrow \frac{n}{\theta} = \sum_{i=1}^{n} \log(x_i)$$

$$\Rightarrow \theta = \frac{n}{\sum_{i=1}^{n} \log(x_i)}$$

(b)

To compute MLE estimate based on given x values:

x=c(4.79,10.89,6.54,22.15)

logl<- 4*(1/sum(log(x)))

logl

Result: 0.4479208

(c) Using data in (b) to obtain the estimate by numerically maximizing the log-likelihood function using optim function in R:

neg_loglikelihood<- function(theta,x)

{

  logl<- 4*(1/sum(log(x)))

  return(-logl)

}

#Maximizing the Negative log likelihood function using optim function

max_lklhood.est <- optim(par=1,fn=neg_loglikelihood,method="BFGS",hessian = TRUE, x=c(4.79,10.89,6.54,22.15))

#finding the MLE

-max_lklhood.est$value

#**Result**: 0.4479208

- **Yes, my results match and are same.**

(d) Using output of numerical maximization in (c),

#standarderror of MLE

se=sqrt(diag((max_lklhood.est$hessian)))

se

#**Result** : 0.2239593

to compute Confidence Interval assuming the dist is t-dist'ed

upperCI=max_lklhood.est$par + qt(0.975,3)*se

upperCI

lowerCI=max_lklhood.est$par -qt(0.975,3)*se

lowerCI.

Reason:

- **As we have considered t-distribution taking an assumption to be a normal distribution, so in this case the approximations will not be accurate.**
- **Here, if you increase sample size, the width of confidence intervals will increase is False.**
- **Increasing the sample size decreases the width of confidence intervals, because it decreases the standard error.**
- **Note that the null value of the confidence interval for the relative risk is one. If a 95% CI for the relative risk includes the null value of 1, then there is <u>insufficient evidence</u> to conclude that the groups are statistically significantly different.**

Since the sample size is small, and width is more (approx. 1.4254766) this doesn't give a good approximation.