PROJECT 3

Name of group members: 1. Ajay Danda (AXD180068) 2. Satyam Bhikadiya (SXB180124)

Contribution of each group member:

Ajay Danda:

- Equal Contribution in solving Q1
- Equal Contribution in solving Q2
- Equal Contribution in Documenting the Report

Satyam Bhikadiya:

- Equal Contribution in solving Q1
- Equal Contribution in solving Q2
- Equal Contribution in Documenting the Report

Question1:

1) Suppose we would like to estimate the parameter θ (> 0) of a Uniform (0, θ) population based on a random sample X1, . . , Xn from the population. In the class, we have discussed two estimators for θ — the maximum likelihood estimator, ˆθ1 = X(n), where X(n) is the maximum of the sample, and the method of moments estimator, ˆθ2 = 2X, where X is the sample mean. The goal of this exercise is to compare the mean squared errors of the two estimators to determine which estimator is better. Recall that the mean squared error of an estimator ˆθ of a parameter θ is defined as E {(ˆθ − θ) 2}. For the comparison, we will focus on n = 1, 2, 3, 5, 10, 30 and θ = 1, 5, 50, 100.

(a) Explain how you will compute the mean squared error of an estimator using Monte Carlo simulation.

Answer:

In order to obtain the Monte Carlo Simulation of Mean squared error we use the formula given below:

We know that Mean Squared error of the estimator $\hat{\theta}$ of a parameter θ is given by:

$$E\left\{\left(\hat{\theta}-\theta\right)^2\right\}$$

Hence, we need to calculate the average of the term $\left(\hat{\theta}-\theta\right)^2$ for the large number of trials of N to obtain the Monte Carlo Simulation of this function. Thus, it means that the $\hat{\theta}$ must be calculated for the larger number of times N using the proper estimator in order to obtain $\left(\hat{\theta}-\theta\right)^2$. The average of the collection of all N values obtained from the expression $\left(\hat{\theta}-\theta\right)^2$, will then be used to approximate the Expected value of $\left(\hat{\theta}-\theta\right)^2$,i.e. $E\left\{\left(\hat{\theta}-\theta\right)^2\right\}$. Hence, giving us the mean squared error of any estimator $\hat{\theta}$, using Monte Carl Simulation.

(b) For a given combination of (n, θ), compute the mean squared errors of both $\widehat{\boldsymbol{\theta}}_1$ and $\widehat{\boldsymbol{\theta}}_2$ using Monte Carlo simulation with N = 1000 replications. Be sure to compute both estimates from the same data.

Answer: As mention in the question that:

$\hat{\theta}_1$- Maximum Likelihood Estimator (MLE)

$\hat{\theta}_2$- Method of Moments Estimator (MOME)

We calculated for n=1 and $\theta = 1$ value and found the values for:
MSE ($\hat{\theta}_1$)= 0.3309007 and MSE ($\hat{\theta}_2$)= 0.3312838

Here, we can observe that for both the Mean Squared Error i.e. for Maximum Likelihood Estimator (MLE) and Method of Moments of Estimator (MOME) the values are approximately equal. Same data s et were used to calculate both values. When Monte Carlo simulation is performed for the different com bination of n and θ we observe a mixed result where, for some cases MSE ($\hat{\theta}_1$)> MSE ($\hat{\theta}_2$) and for som e cases MSE ($\hat{\theta}_1$)< MSE ($\hat{\theta}_2$) , thus creating an ambiguity over the comparison for their performance o n the all small values of n such as n=1.

(c) Repeat (b) for the remaining combinations of (n, θ). Summarize your results graphically.

Answer: For the different combination of (n, θ), the results for Mean Squared Error is shown below:

n	θ	MSE($\hat{\theta}_1$)	MSE($\hat{\theta}_2$)
1	1	0.3309007	0.3312838
1	5	8.117512	8.450327
1	50	831.8406	828.7016
1	100	3341.129	3368.579
2	1	0.1846717	0.1760687
2	5	4.178317	4.109087
2	50	409.7049	388.9889
2	100	1593.04	1625.124
3	1	0.09909183	0.11054142
3	5	2.453457	2.717393
3	50	252.1659	282.0350
3	100	1014.995	1138.687
5	1	0.05457649	0.07141098

5	5	1.211002	1.627546
5	50	123.3215	174.6433
5	100	452.6973	657.8734
10	1	0.01459196	0.03384411
10	5	0.3488868	0.7840706
10	50	36.91358	90.26275
10	100	157.0693	348.1876
30	1	0.002090203	0.010785354
30	5	0.0506085	0.2649053
30	50	4.816972	27.751980
30	100	19.86309	118.51435

The Graphical representation for the different combination of (n, θ) is given below:

The difference in the performance between the two estimators Maximum Likelihood Estimator (MLE) and Method of Moments Estimator (MOME) is observed by plotting a comparison graphs between the Mean Squared Error (MSE) vs Theta. The plotting is done for each of the given values of n to show the effect of both n and θ on the performances.

(Theta-1)-$\hat{\theta}_1$- Maximum Likelihood Estimator (MLE)

(Theta-2)-$\hat{\theta}_2$- Method of Moments Estimator (MOME)

(d) Based on (c), which estimator is better? Does the answer depend on n or θ? Explain. Provide justification for all your conclusions.

Answer:

As we see the graphical representation of MSE vs Theta for different combination of n and θ, it is observed that for low values of n (i.e. for n=1,2) there is no clear understanding that which estimator has edge over the other or which estimator is giving us the better estimate for θ, because in some cases MOME is better than MLE and in some cases MLE is better than MOME. However, for higher values of n i.e. for n= 3,5 and above, we can see a clearly observe that the Maximum Likelihood Estimator (MLE) provides a better estimate for θ than the Method of Moments Estimator (MOME) because MLE is repeatedly producing a lower value for the Mean Squared Error. For n=100, we can clearly observe that the difference between both the estimator is increased. Hence, by observing the graphical representation we can conclude that if we have larger number of samples (i.e. for higher values of n), we should use Maximum Likelihood Estimator as it consistently produces a low MSE. The other observation that we can see from the graphical representation is that, the MSE is directly dependent upon θ i.e. as θ increases, MSE also increases and this is because uniform distribution depends upon θ and the variability of distribution increases as the interval becomes wider and larger.

It is also observed that the effect for θ value on MSE decreases as MLE remains repeatedly lower for the larger value on n. However, the effect for θ values on MSE increases for MOME and thus we can also say that MLE converges rapidly towards the actual value of θ. Therefore, Maximum Likelihood Estimator is a better estimator for θ among the two. Hence, for larger values on n, the MLE is repeatedly better and gets less effected by θ but MOME produces high Mean Squared Error for larger values of n.

Question 2:

2. Suppose the lifetime, in years, of an electronic component can be modelled by a continuous random variable with probability density function

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

where $\theta > 0$ is an unknown parameter. Let X1, . . ,Xn be a random sample of size n from this population.

(a) Derive an expression for maximum likelihood estimator of θ.

Answer:

We know that, probability density function of the lifetime in years of an electric component is given by

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

The Maximum Likelihood function $L(\theta)$ for random variable $x_1, x_2 \ldots x_n$ is given by:

$$L(\theta) = \prod_{i=1}^{n} \frac{\theta}{x^{\{\theta+1\}}} = \theta^n \prod_{i=1}^{n} x_i^{-(\theta+1)}$$

By taking the log-likelihood function, in order to get maximum value for the estimator θ, we get:

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

Now, we differentiate above equation by θ

$$\frac{d}{d\theta}\log(L(\theta)) = 0$$

$$\therefore \frac{n}{\theta} - \sum_{i=1}^{n}\log x_i = 0$$

$$\boxed{\therefore \theta = \frac{n}{\sum_{i=1}^{n}\log x_i}}$$

This equation gives us the maximum likelihood estimate

(b) Suppose n = 5 and the sample values are x1 = 21.72, x2 = 14.65, x3 = 50.42, x4 = 28.78, x5 = 11.23. Use the expression in (a) to provide the maximum likelihood estimate for θ based on these data.

Answer:

Substituting values for n = 5 and the sample values x1 = 21.72, x2 = 14.65, x3 = 50.42, x4 = 28.78, x5 = 11.23 in the equation which we get from Q2 (a).

We get the maximum likelihood estimate of θ as $\hat{\theta}$= 0.32338742

(c) Even though we know the maximum likelihood estimate from (b), use the data in (b) to obtain the estimate by numerically maximizing the log-likelihood function using optim function in R. Do your answers match?

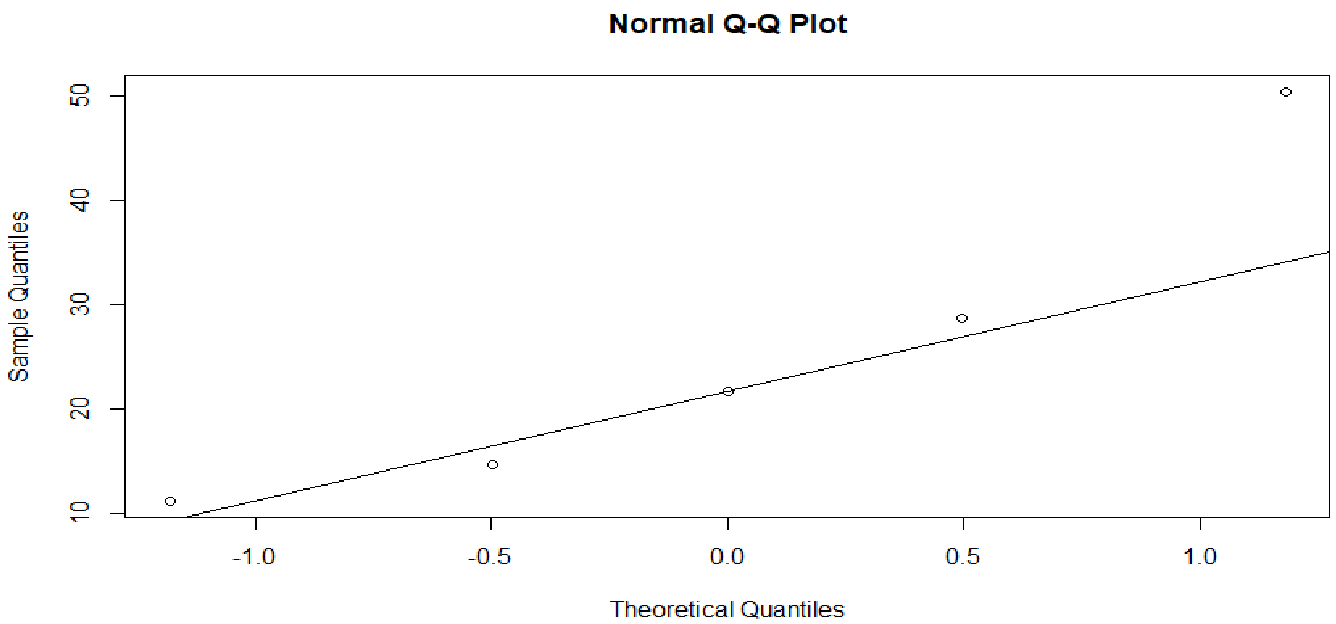
Answer:

Yes, the answer matches. By using the data in (b) to obtain the estimate by numerically maximizing the log-likelihood function using optim function in R we obtain $\hat{\theta}$= 0.3233876. Which is approximately equal to the $\hat{\theta}$= 0.32338742 that we obtained theoretically.

(d) Use the output of numerical maximization in (c) to provide approximate standard error of the maximum likelihood estimate and an approximate 95% confidence interval for θ. Are these approximations going to be good? Justify your answer.

Answer:

By assuming that n is larger enough for a normal approximation, the output of numerical maximization in (c) was used to approximate standard error of the maximum likelihood estimate and an approximate 95% confidence interval for θ. The approximate standard error was found to be 0.1446219 and the confidence interval was found to be in (0.03993379, 0.60684138). We find that the approximation is not good because the standard error is quite significant in magnitude when we compared it to the value of $\hat{\theta}$= 0.3233876, thus it observed that standard error varies by a significant amount from normal approximation and we find that the number of points is too few which leads to a poor approximation, as the data set is not from a normal distribution which we can see from the QQ plot given below. Hence, it is neither a normal distribution and it does not have enough points which are required to obtain the normal approximation and calculate proper confidence interval.



R CODE Question 1:

```
> theta_estimator = function(n,theta){        # Function to calculate estimator MLE & MOME
+   distribution= runif(n,min=0,max=theta)
+   samples=runif(n,min = 0,max = theta)     # Ensuring that the  samples remains same
+   theta1_estimator = max(samples)           # MLE Estimator
+   theta2_estimator= 2*mean(samples)         # MOME Estimator
# returning the pair of values theta1 and theta 2 computed from the same sample set
+   answer = c(theta1=theta1_estimator,theta2=theta2_estimator)
+   return (answer)
+ }

> mse_estimator = function(n,theta){
+   r=replicate(1000,theta_estimator(n,theta))
+   theta1_value=r[1,]                  #theta 1 parameters are returned
+   theta2_value=r[2,]                  #theta 2 parameters are returned
+   mse1=mean((theta1_value-theta)^2)
+   mse2=mean((theta2_value-theta)^2)
+   answer = c(mse_theta1=mse1,mse_theta2=mse2)
+   return (answer)
+ }

# Calculating MSE Estimator for different Combination of n and θ
> mse_estimator(1,1)
mse_theta1 mse_theta2
 0.3309007  0.3312838
> mse_estimator(1,5)
mse_theta1 mse_theta2
  8.117512   8.450327
> mse_estimator(1,50)
mse_theta1 mse_theta2
  831.8406   828.7016
> mse_estimator(1,100)
mse_theta1 mse_theta2
  3341.129   3368.579
> mse_estimator(2,1)
mse_theta1 mse_theta2
 0.1846717  0.1760687
> mse_estimator(2,5)
mse_theta1 mse_theta2
  4.178317   4.109087
> mse_estimator(2,50)
mse_theta1 mse_theta2
  409.7049   388.9889
> mse_estimator(2,100)
mse_theta1 mse_theta2
  1593.045   1625.124
> mse_estimator(3,1)
mse_theta1 mse_theta2
0.09909183 0.11054142
> mse_estimator(3,5)
```

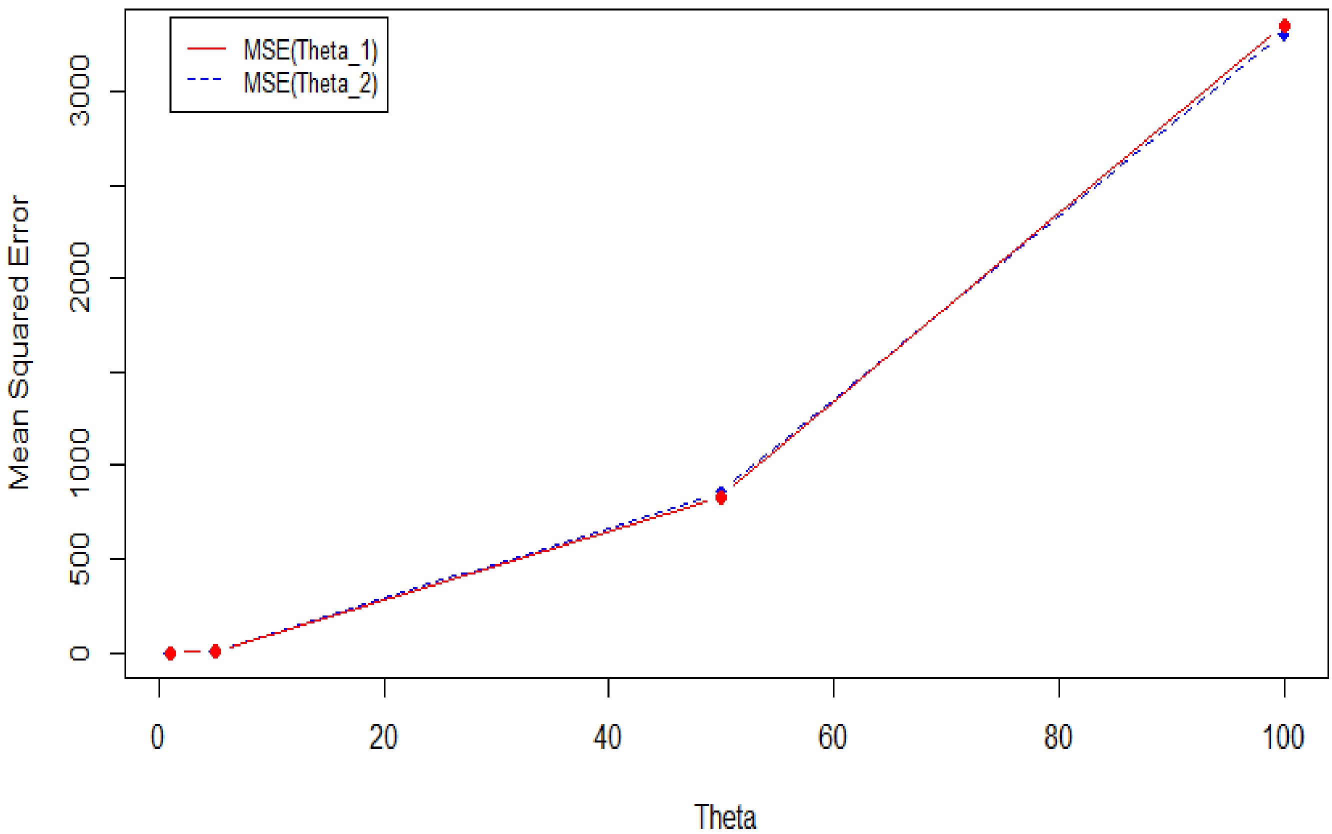
```
mse_theta1 mse_theta2
  2.453457   2.717393
> mse_estimator(3,50)
mse_theta1 mse_theta2
  252.1659   282.0350
> mse_estimator(3,100)
mse_theta1 mse_theta2
  1014.995   1138.687
> mse_estimator(5,1)
mse_theta1 mse_theta2
0.05457649 0.07141098
> mse_estimator(5,5)
mse_theta1 mse_theta2
  1.211002   1.627546
> mse_estimator(5,50)
mse_theta1 mse_theta2
  123.3215   174.6433
> mse_estimator(5,100)
mse_theta1 mse_theta2
  452.6973   657.8734
> mse_estimator(10,1)
mse_theta1 mse_theta2
0.01459196 0.03384411
> mse_estimator(10,5)
mse_theta1 mse_theta2
 0.3488868  0.7840706
> mse_estimator(10,50)
mse_theta1 mse_theta2
  36.91358   90.26275
> mse_estimator(10,100)
mse_theta1 mse_theta2
  157.0693   348.1876
> mse_estimator(30,1)
 mse_theta1  mse_theta2
0.002090203 0.010785354
> mse_estimator(30,5)
mse_theta1 mse_theta2
 0.0506085  0.2649053
> mse_estimator(30,50)
mse_theta1 mse_theta2
  4.816972  27.751980
> mse_estimator(30,100)
mse_theta1 mse_theta2
  19.86309  118.51435
```

Plotting the MSE vs Theta Graph for n and θ

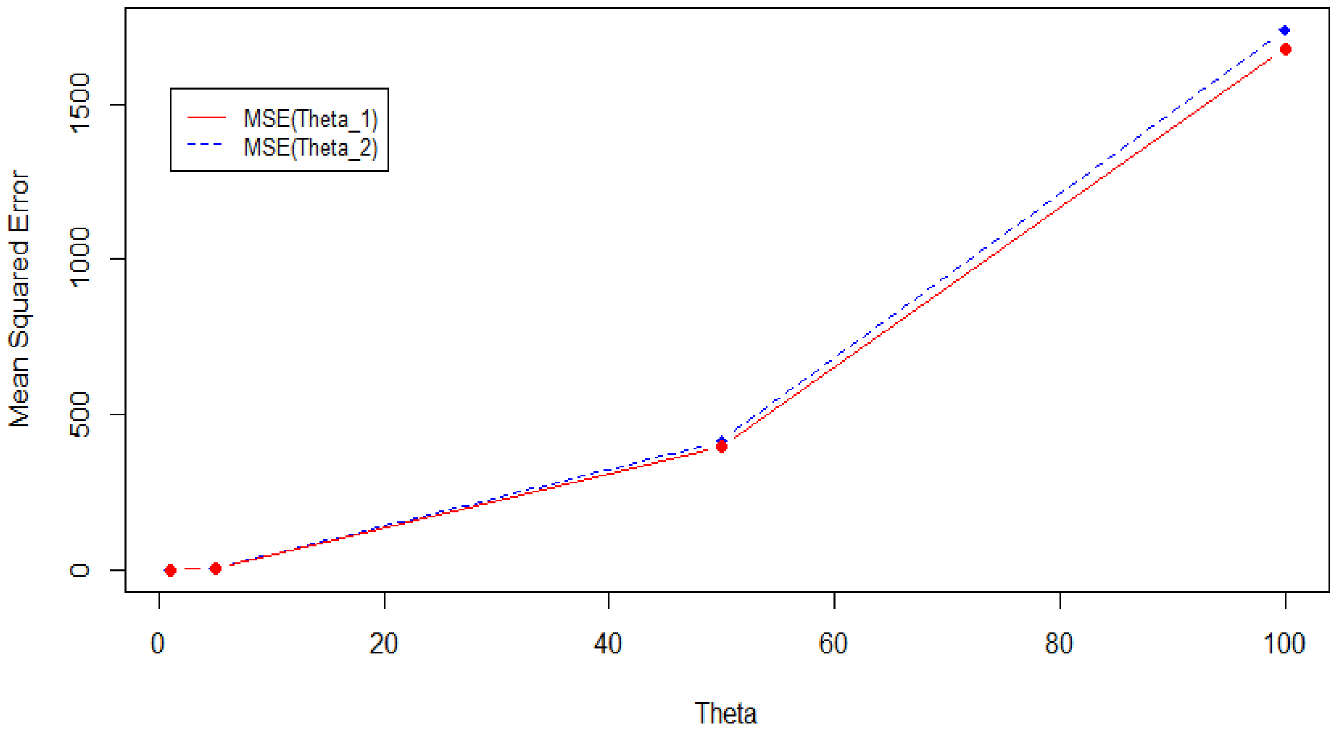
```
> mse_plot = function(n, height){
+   cumulative_answers_trials1 = c()
+   for (theta_given in c(1,5,50,100)){
+     cumulative_answers_trials1 = c(cumulative_answers_trials1,mse_estimator(n,theta_given))
+   }
+   theta = c(1,5,50,100)
+   mse.theta1 = cumulative_answers_trials1[c(TRUE,FALSE)]
+   mse.theta2 = cumulative_answers_trials1[c(FALSE,TRUE)]
+   plot(x=theta,y=mse.theta2,type="b",pch=18, col="blue",lty=2,ylab="Mean Squared Error",xlab=
"Theta")
+   lines(x=theta,y=mse.theta1,type="b",pch=19,col="red")
+   legend(1, height, legend=c("MSE(Theta_1)", "MSE(Theta_2)"),col=c("red", "blue"), lty=1:2, cex
=0.8)
+ }

> mse_plot(1,3400)
> mse_plot(2,1550)
> mse_plot(3,1000)
> mse_plot(5,500)
> mse_plot(10,300)
> mse_plot(30,100)
```

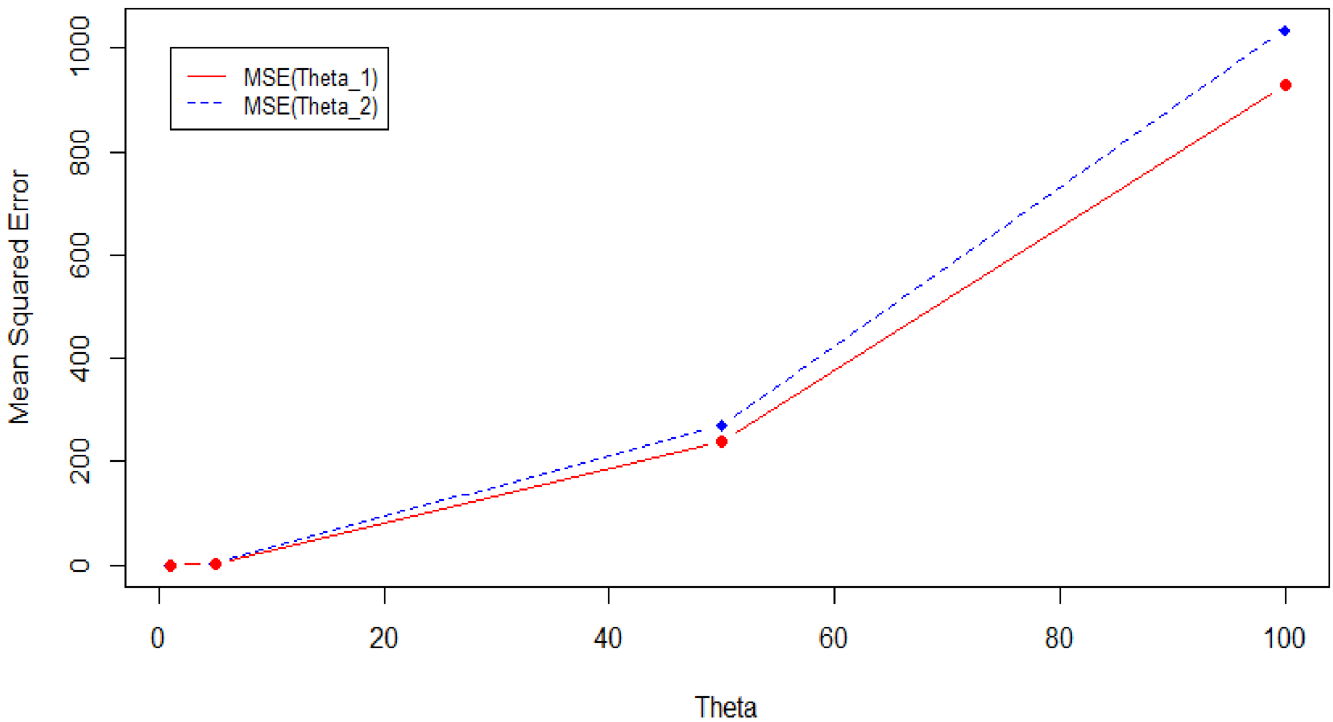
(i.) Graph of MSE vs Theta for n=1 and θ= (1,5,50,100):



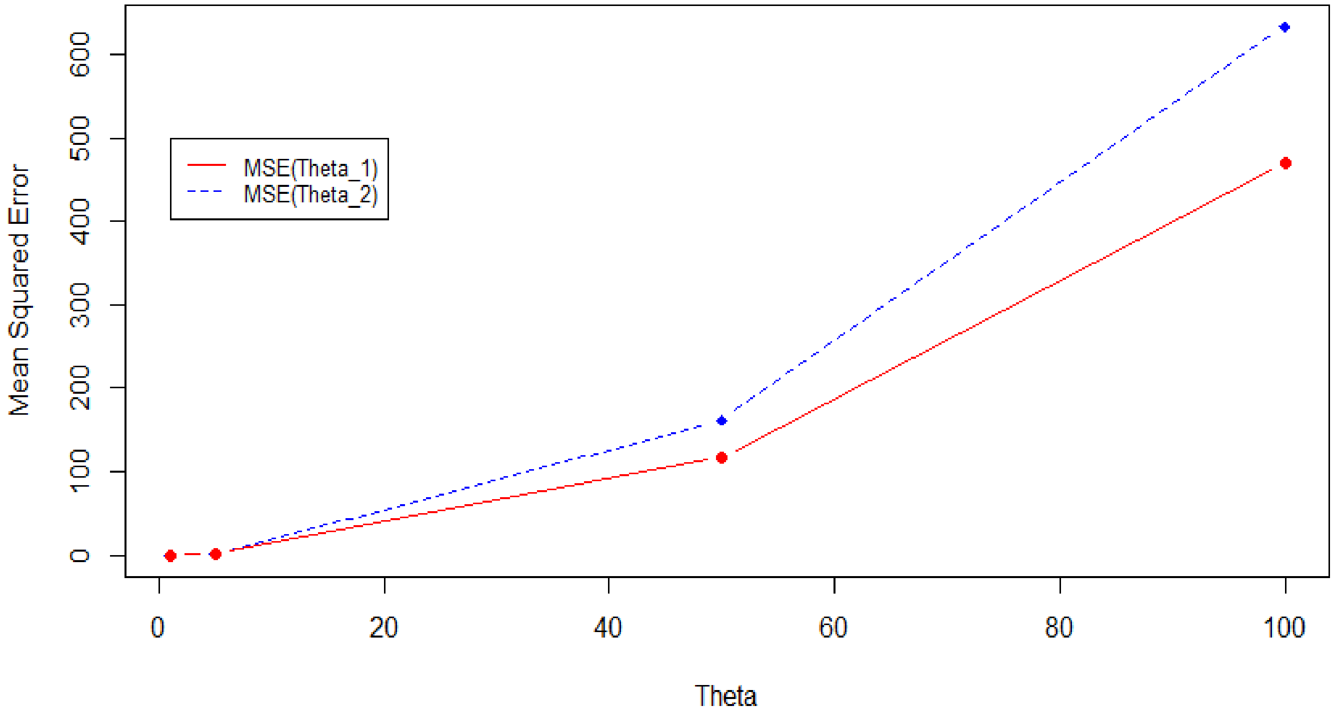
(ii.) Graph of MSE vs Theta for n=2 and θ= (1,5,50,100):



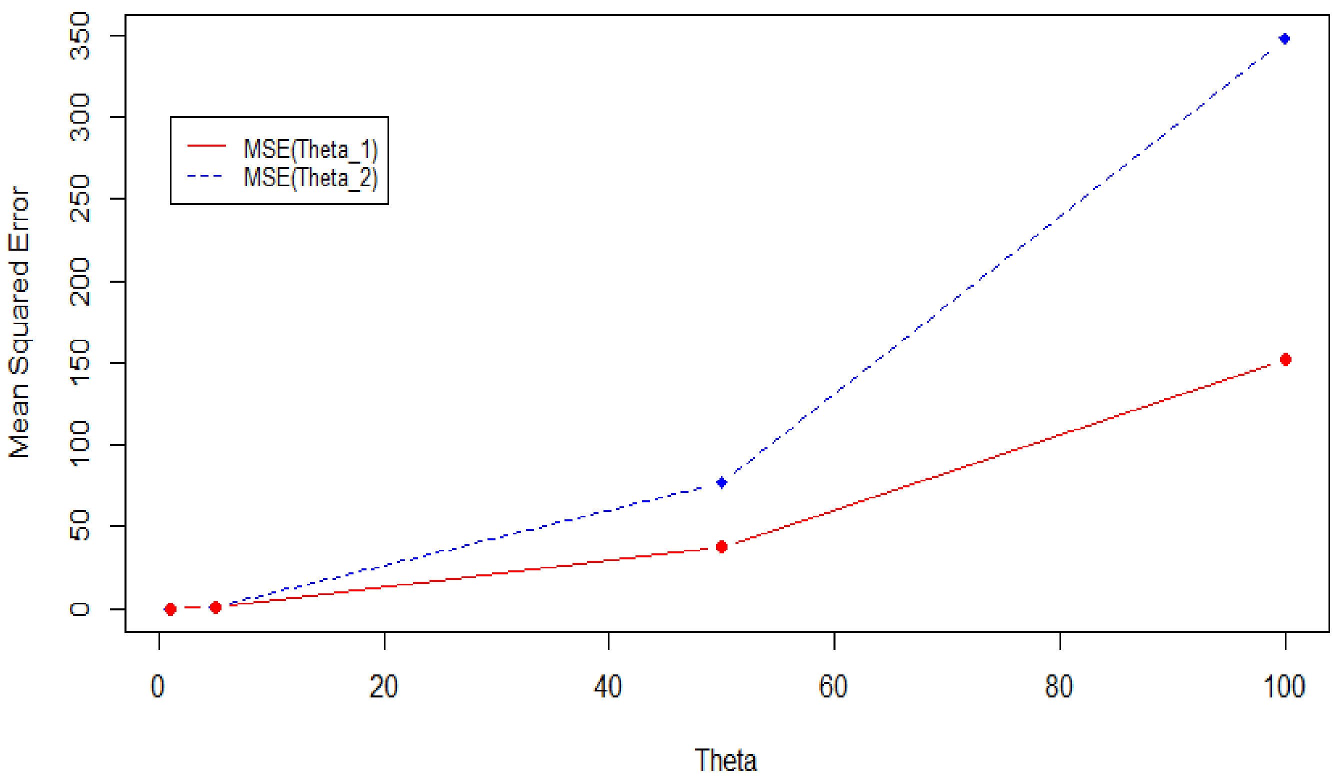
(iii.) Graph of MSE vs Theta for n=3 and θ= (1,5,50,100):



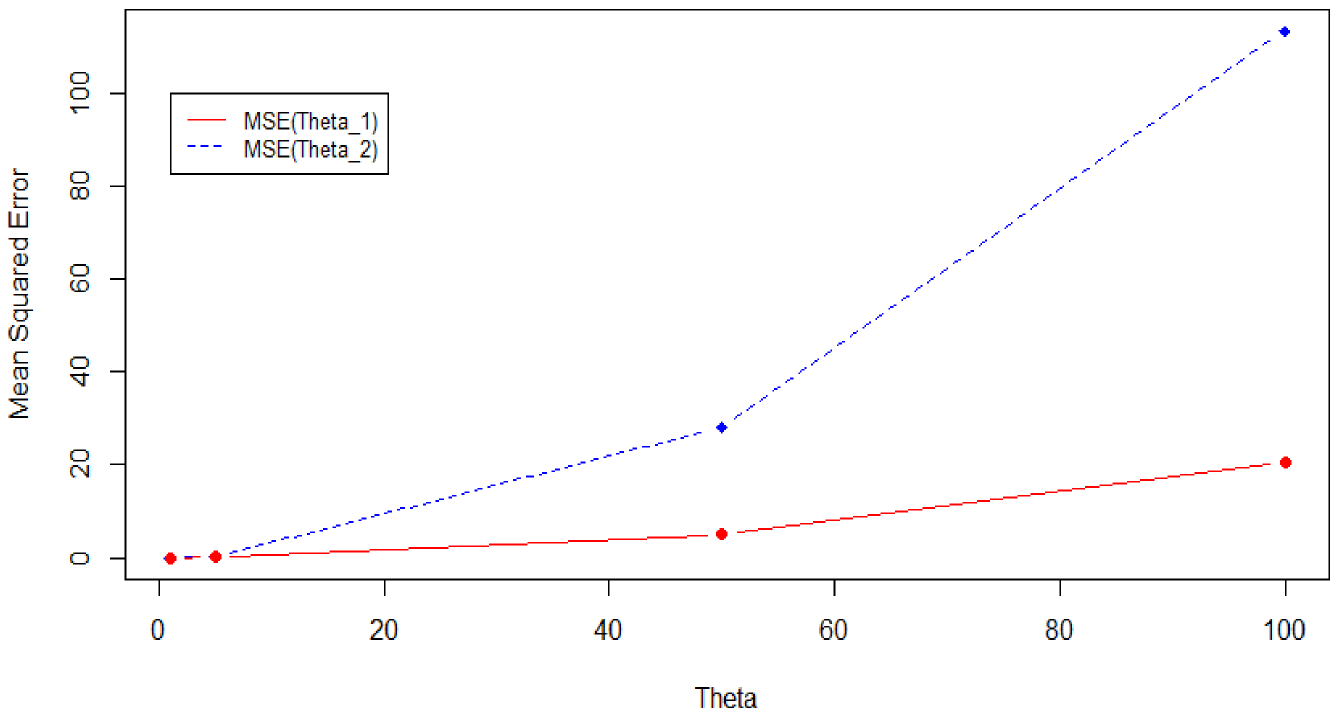
(iv.) Graph of MSE vs Theta for n=5 and θ= (1,5,50,100):



(v.) Graph of MSE vs Theta for n=10 and θ= (1,5,50,100):



(vi.) Graph of MSE vs Theta for n=100 and θ= (1,5,50,100):



R CODE QUESTION 2:

```
> dataset = c(21.72,14.65,50.42,28.78,11.23)            # The given Data set
> negative_log_estimator = function(theta,data){        # Maximum Likelihood Function
+   answer = sum(log(theta*data^(-theta-1)))
+   return (-answer)
+ }
> mle = optim(par=1.041,fn=negative_log_estimator,method="BFGS",hessian=T, data=dataset)
> mle
$par
[1] 0.3233876

$value
[1] 26.10585

$counts
function gradient
      22        9

$convergence
[1] 0

> standard_error = sqrt(diag(solve(mle$hessian)))     # Calculating Standard Error
> standard_error
[1] 0.1446219

> confidence_interval = function(SE,n,alpha){         # Confidence Interval Function
+   d = dataset
+   ci=mle$par+c(-1,1)*qnorm(1-(alpha/2))*SE
+   return (ci)
+ }
> a=confidence_interval(SE=standard_error,n=length(dataset),alpha=0.05)
> a
[1] 0.03993379 0.60684138

> qqnorm(dataset)
> qqline(dataset)
```