MAS8403 Statistical Foundations of Data Science

R Solution

Practical 3: Maximum Likelihood

Question 1:

1. ##First Question
2. ##Generate the log-likehood value for sample size of n Poisson Process
3. #Param: theta : from Poisson Process like lambda
4. #       sample : All sample
5. #
6. Log\_Likehood = function(theta, Sample){
8. Sample\_size = length(Sample) ##Get the size of sample
10. log = 0 # Store each log value
12. final\_log = 0 # Store the log-likelihood value
14. **for** (count **in** Sample\_size){ # loop of samples to get the each likelihood value of  each theta
16. # log  value of each Xi in samples
17. log = Sample[count] \* log(theta) - theta - log(factorial(Sample[count]))

20. final\_log = final\_log + log # Get the log-likelihood value
22. }
24. **return** (final\_log) # Return the corrosponding log-likelihood value

Results:

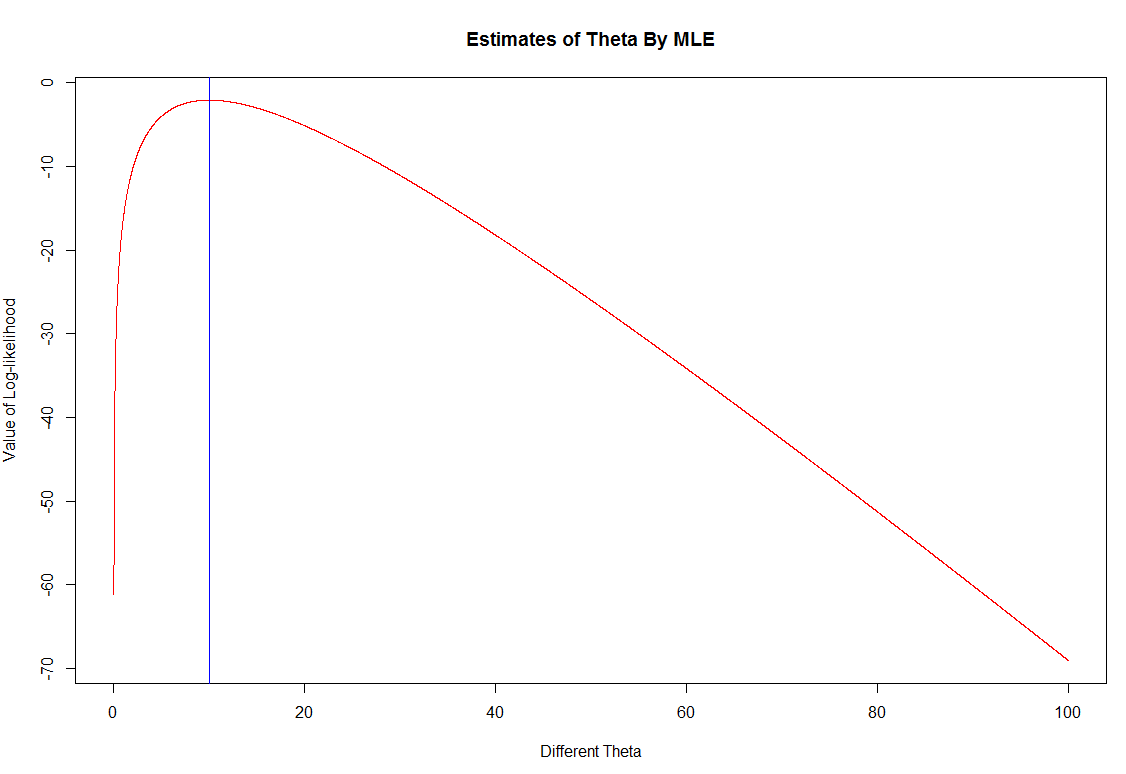
As we can check in question 2, the function works well.

Question 2:

1. #Second Question
2. ##Use Some Sample to plot Log-Likehood
3. #
4. Main\_Plot\_Question2 = function(){
6. sample = c(12, 8, 14, 8, 11, 6, 13, 9, 9, 10) # Set the Sample in question
8. theta = seq(0,100,0.01) # Set the range of theta
10. theta.hat = theta[which.max(Log\_Likehood(theta = theta, Sample = sample))] # Get the theta which can max the log-likelihood value
12. # Plot the loglikelihood against theta
13. plot(theta, Log\_Likehood(theta = theta, Sample = sample), type = 'l', col = 'red'
14. ,xlab = 'Theta', ylab = 'Log-likelihood',main = 'Hello')
15. abline(v = theta.hat,col = 'blue') # Plot the theta we find
16. }
18. Main\_Plot\_Question2() # Run the Question 2

Results:

theta.hat = 10

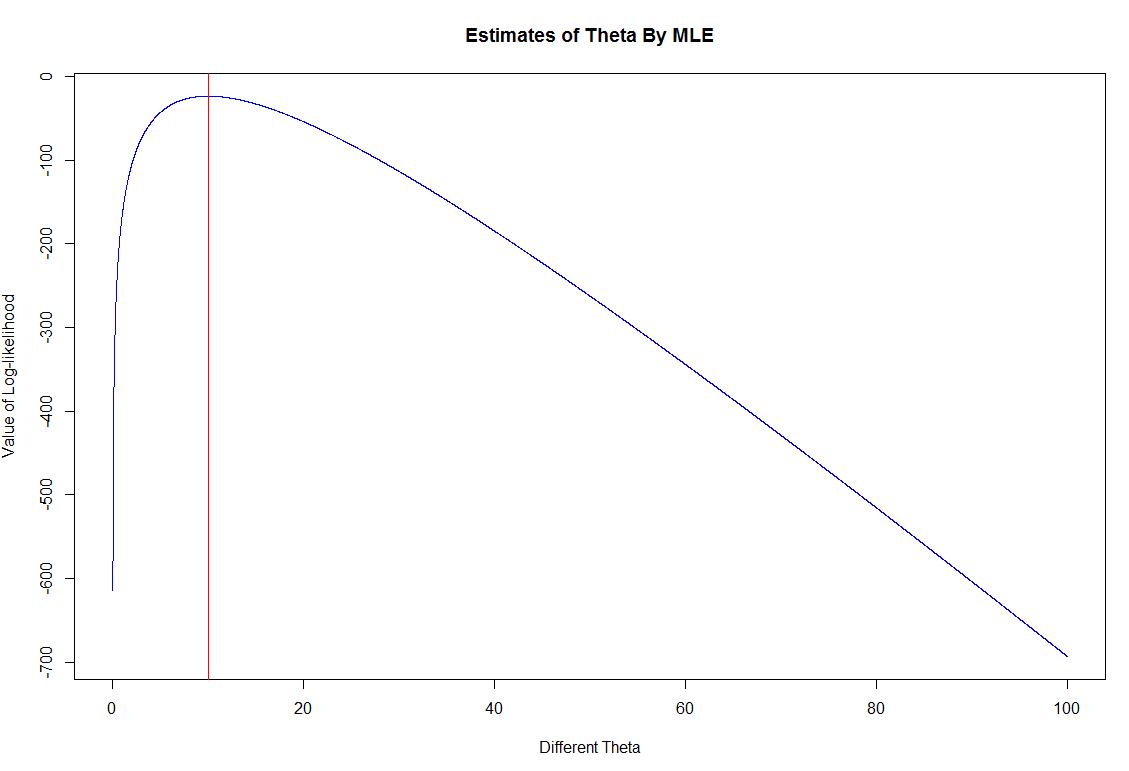


Question 3:

1. ##Third Question
2. ##New function of Q1 but output the maximum likehood estimate theta
3. #Param: sample :whatever sample needed to use to find a good theta
4. #
5. MLE\_Question3 = function(Sample){
7. theta = seq(0,100,0.01) # Give a theta a range
9. Sample\_size = length(Sample) # Get the size of samples
11. Each\_value = c() # Used to store the each value
13. count\_value = 1 # Index of above vector
15. **for** (diff\_theta **in** theta){ #Loop of different theta
17. log = 0 # Initialize the log value
19. final\_log = 0 # Initialize the log-likelihood value
21. **for** (count **in** 1:Sample\_size){ # loop of samples to get the each likelihood value of each theta
23. log = Sample[count] \* log(diff\_theta) - diff\_theta - log(factorial(Sample[count])) # log value of each Xi in samples
25. final\_log = final\_log + log # Get the log-likelihood value
27. }
29. Each\_value[count\_value] = final\_log # Store the each log-likelihood value of each theta
31. count\_value = count\_value + 1 # Count the index of anove vector to store
33. }
35. index\_max = which.max(Each\_value) # Get the index of max likelihood value
37. plot(theta, Each\_value, type = 'l', col = 'blue',xlab = 'Different Theta', ylab = 'Value of Log-likelihood',main = 'Estimates of Theta By MLE')
38. abline(v = theta[index\_max],col = 'red') # Plot the theta we find
40. **return** (theta[index\_max]) # Return the theta max the log-likelihood value
41. }
43. #Question 3
44. main\_Question3 = function(){
46. sample = c(12, 8, 14, 8, 11, 6, 13, 9, 9, 10) # Set the samples
47. max\_et\_theata = MLE\_Question3(sample) # Get the max theta
49. }
51. **print**(main\_Question3()) # Run the question 3

Results:

Estimate of theta[index\_max] = 10 , agree by the log-likelihood plot



Question 4:

1. ##Question 4
2. #
3. Po\_mle = function(n){

6. count\_st = 1 # Index of vector of each mean and variance
7. mean\_store = c() # Vector of each mean and variance
8. var\_store = c() # Vector of each mean and variance
10. **for** (s **in** n){ #Loop of each n
12. Each\_th = c() # Vector to store the each theta
13. mean\_all = 0 # Initialize the mean of each n
14. var\_all = 0  # Initialize the variance of each n
15. count = 1   # Index of the vector of each theta

18. **print** (s)
20. **while**( count != 1001 ){ # Loop of 1000 samples' MLE value
22. sample\_pd = rpois(s,10)# Randomly generated from poisson distribution
24. Each\_th[count] = MLE\_Question3(sample\_pd) # Store the each theta for each n
26. count = count + 1 # Count the index of storing the theta
27. }
29. mean\_store[count\_st] = mean(Each\_th) # mean of each n
31. var\_store[count\_st] = var(Each\_th) #  variance of each n
33. count\_st = count\_st + 1 # Count the index of vector above
35. }
37. final\_list = c(mean\_store,var\_store) # Reture all the mean and variance
39. **return** (final\_list)
41. }

Result :

As we can check in question 5, it works well

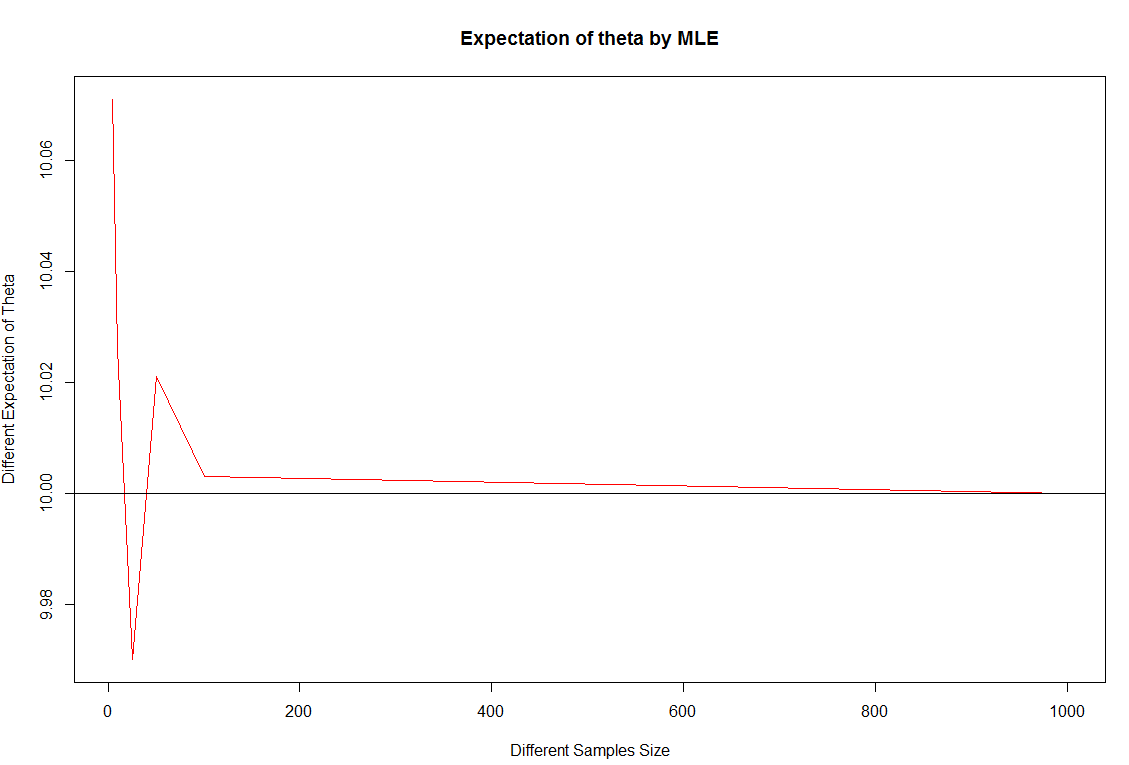
Question 5:

1. main\_Question5 = function(){
2. n = c(5,10,25,50,100,1000) # Set the n samples
4. inter = length(n) # get the Length of n
6. results = Po\_mle(n) # Get the Results
8. mean = results[1:inter] # Get the mean of all n
10. var = results[(inter + 1):(2\*inter)] # Get the variance of all n

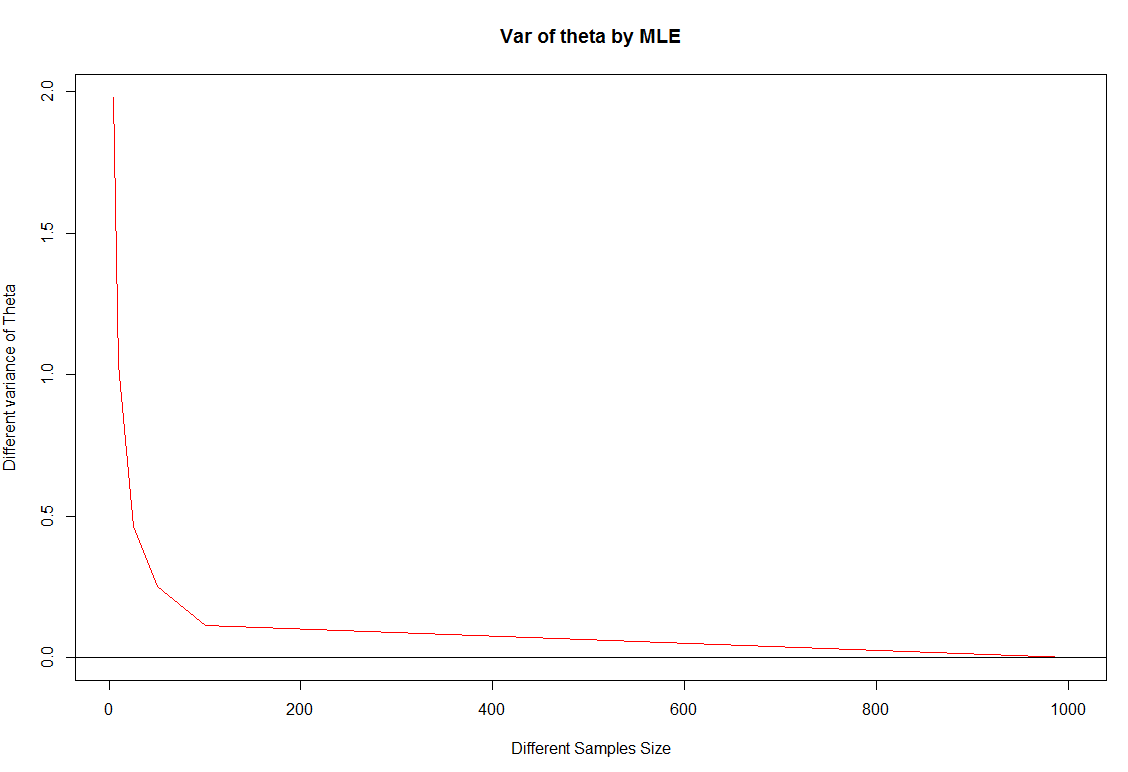
13. plot(n, mean, type = 'l', col = 'red',xlab = 'Different Samples Size', ylab = 'Different Expectation of Theta',main = 'Expectation of theta by MLE')
14. abline(h = 10,col = 'black')
16. plot(n, var, type = 'l', col = 'red',xlab = 'Different Samples Size', ylab = 'Different variance of Theta',main = 'Var of theta by MLE')
17. abline(h = 0,col = 'black')
19. }
21. main\_Question5() # Run the Question 5

Result :

**As we can see graphically, firstly**



**in the meantime**

****

**So, I think it is Asymptotically Unbiased and consistency**