Problem Set 4

Antonio Dehesa

## Introduction

Please complete the following tasks regarding the data in R. Please generate a solution document in R markdown and upload the .Rmd document and a rendered .doc, .docx, or .pdf document. Your work should be based on the data’s being in the same folder as the .Rmd file. Please turn in your work on Canvas. Your solution document should have your answers to the questions and should display the requested plots.

These questions were rendered in R markdown through RStudio (<https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf>, <http://rmarkdown.rstudio.com> ).

# Question 1

## Context

Suppose is a parametrized family of distributions. The parameter may be vector-valued or one dimensional. Under fairly general circumstances, the maximum likelihood parameter estimate of the parameter based on a sample is *consistent*, also called *asymptotically consistent*. Informally, this means that as larger and larger samples are used to estimate the parameter, the estimate gets closer and closer to the true value.

Some parameter estimates are *unbiased*. Informally, this means that if the estimate is applied to samples of size to get a collection of estimates , the mean of the estimates, will get closer and closer to as gets larger and larger.

In this question you will perform numerical experiments on samples from a distribution to see whether the maximum likelihood estimates for and appear to be consistent and unbiased.

## Q1, part 1

(5 points)

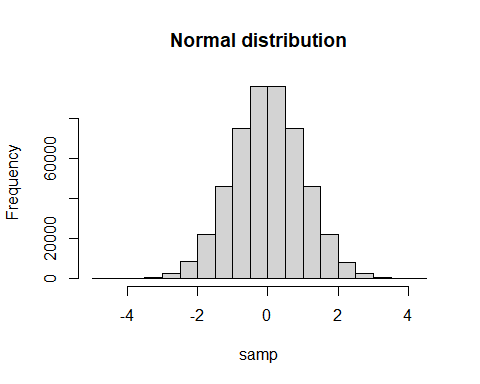
The purpose of this question is to perform numerical experiments to gain insight into the whether of maximum likelihood estimates of and are consistent for samples from .

The code provided generates samples from the standard Normal distribution, . For each value in , the maximum likelihood estimates of and are computed for the initial portion of the sample . These values are stored in order of in the data frame “dat.consist” with the variable names “mu.hat” and “sigma.sq.hat” respectively. A column of the corresponding values of is added under the variable name “n”. Below you will use this data frame to examine whether these samples provide numerical evidence that the maximum likelihood estimates of and are consistent. Plotting using “geom\_line” may be helpful.

set.seed(123456)  
N<-500000  
samp<-rnorm(N)  
# function to compute the maximum likelihood estimate of mu and the sigma-squared based on the first n values in a vector "samp" of samples from a Normal distribution:  
theta.est<-function(n,s=samp){  
 m<-mean(s[1:n])  
 s2<-sum((s[1:n]-m)^2)/n #   
 return(c(m,s2))  
}  
dat.consist<-t(sapply(seq(1000,N,by=1000),theta.est, s=samp))  
dat.consist<-data.frame(dat.consist)  
dat.consist$n<-seq(1000,N,by=1000)  
names(dat.consist)<-c("mu.hat","sigma.sq.hat","n")

1. What is the true value of the parameter for these data? Please give a numeric value. The default parameter for the mean, which is mu, for rnorm is 0. We can see this with a histogram for the normal distribution of samp.

hist(samp, main = "Normal distribution")

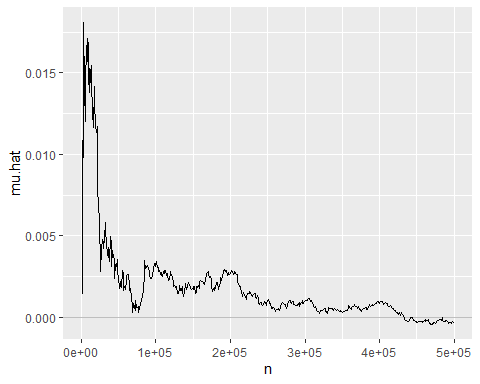


# Our value should be 0  
act\_mean<-mean(samp)  
# As we can see, it is very close to 0. But our actual value is $act\_mean$

Our value should be 0 actual mean = -2.9504208^{-4} As we can see, it is very close to 0. But our actual value is -2.9504208^{-4}

1. Do the estimates of “mu.hat” of in “dat.consist” appear to approach the true value as the sample size “n” increases?

ggplot(dat.consist,aes(x=n,y=mu.hat))+  
 geom\_hline(yintercept = 0,color="gray")+  
 geom\_line()



dat.consist$mu.hat[c(1,5,10,50,100,500)]

## [1] 0.0108535170 0.0120273564 0.0168396665 0.0024677304 0.0032766567  
## [6] -0.0002950421

Yes, we can see that the bigger the n, the closer the estimated mu\_hat is to the actual mean:

similarity <- tail(dat.consist$mu.hat, n=1) == act\_mean

The last estimation, -2.9504208^{-4}, is equal to the actual mean,-2.9504208^{-4}

1. Does this numerical experiment suggest that the maximum likelihood estimate of is consistent?

If I understood correctly, consistency is that, the more samples we have, the closer the estimate should be to the right value. This is what we observed in this case, so it should be consistent.

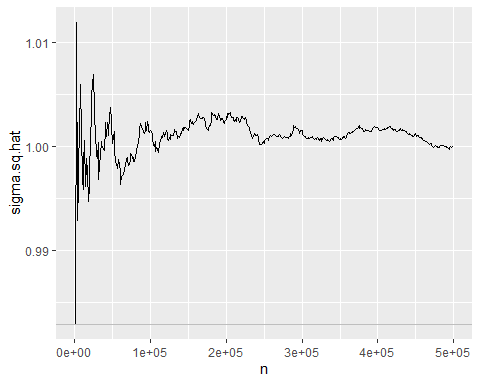
1. What is the true value of the parameter for these data? Please give a numeric value.

(act\_sd\_squared<-sd(samp)^2)

## [1] 0.999931

1. Do the estimates of “sigma.sq.hat” of in “dat.consist” appear to approach the true value as the sample size “n” increases? If you are unsure, you can calculate the estimate for some very large samples.

ggplot(dat.consist,aes(x=n,y=sigma.sq.hat))+  
 geom\_hline(yintercept = min(dat.consist$sigma.sq.hat),color="gray")+  
 geom\_line()



dat.consist$sigma.sq.hat[c(1,5,10,50,100,500)]

## [1] 0.9828973 0.9961045 0.9967854 1.0002562 1.0015304 0.9999290

Yes, we can see in the graph that the more samples we have, the more it stabilizes to the real value. F) Does this numerical experiment suggest that the maximum likelihood estimate of is consistent?

Using the same logic as in Q1, P1, C, yes, it should be consistent.

## Q1, part 2

(5 points)

The purpose of this question is to perform numerical experiments to gain insight into whether the maximum likelihood estimates of and are unbiased for samples of size 5 from

1. Create a matrix of samples of size 5 from the standard Normal distribution.

set.seed(45678)  
mat<-matrix(rnorm(10000\*5),ncol=5)

1. Please use apply to calculate the maximum likelihood estimates and of and for each sample.

If by sample, we mean by column.

N<-10000  
dat.consist<-t(apply(mat,2,theta.est, n=N))  
dat.consist<-data.frame(dat.consist)  
dat.consist$n<-seq(1,5,by=1)  
names(dat.consist)<-c("mu.hat","sigma.sq.hat","")

1. Compute the mean of the s and the mean of the s.

mat\_mu\_mean<-mean(dat.consist$mu.hat)  
mat\_sigma\_squared<-mean(dat.consist$sigma.sq.hat)  
mat\_mu\_mean

## [1] 0.008816334

mat\_sigma\_squared

## [1] 1.005367

1. Does the maximum likelihood estimate of seem to be unbiased? (You may repeat the experiment with other seeds to help answer this question.)

Yes. We can see that, although the final result of mat\_mu\_mean is not as close to the real value as one of the individual values, I tried to get the mean of only two values, the mean for three values, the mean of four values, and finally, the mean of the five values. Each step was closer to the real value. Therefore, I believe that it is unbiased.

1. Does the maximum likelihood estimate of seem to be unbiased? (You may repeat the experiment with other seeds to help answer this question. Try comparing with the adjusted estimates produced by dividing the sum of the squared differences by 4 instead of 5.)

No. I tried as suggested, by testing with the mean of two values, 3, 4, and 5. The results were inconsistent. From 2 to 3 there was an increase in the error. There was also an increase in the error from 3 to 4. Then, from 4 to 5 there was a decrease. Therefore, I believe that this is not unbiased.

sum(dat.consist$sigma.sq.hat[1:2])/2

## [1] 1.001433

sum(dat.consist$sigma.sq.hat[1:3])/3

## [1] 1.009729

sum(dat.consist$sigma.sq.hat[1:4])/4

## [1] 1.011772

sum(dat.consist$sigma.sq.hat[1:5])/5

## [1] 1.005367

rm(mat)

# Question 2

## Context

The uniform distributions are a two parameter family of continuous distributions, with and . Given , the sample space is and the probability density function is .

## Q2, part 1

(5 points)

Please compute the mean of .

mean = (1/2)\*(a+b)

## Q2, part 2

(5 points)

Please compute the variance of . The identity

may be useful in simplifying the formula.

Variance = (1/12)\*(b-a)^2 # Question 3

The data sets in these questions were downloaded 6/13/2022 from <https://ourworldindata.org/>

The code chunks below read in a data frame of world populations and a data frame of world population densities.

dat.pop<-read.csv("population-since-1800.csv",stringsAsFactors = FALSE)  
names(dat.pop)[4]<-"population"  
dat.den<-  
 read.csv("population-density.csv")  
names(dat.den)[4]<-"density"

## Q3, part 1

(2 points)

Write code to restrict both data frames to cases in which the value of “Year” is 2020 and the value of “Code” is not the empty string, ““, and is not the value for the whole world,”OWID\_WRL”. Please display the number of rows in the resulting data frames using the function nrow.

dat.pop<-filter(dat.pop, Year ==2020, Code != "", Code != "OWID\_WRL")  
dat.den<-filter(dat.den, Year ==2020, Code != "", Code != "OWID\_WRL")  
nrow(dat.pop)

## [1] 234

nrow(dat.den)

## [1] 216

The following code merges the data sets, restricting to values of “Code” occurring in both data sets.

dat.both<-inner\_join(dat.den,dat.pop,by="Code")  
# check  
mean(dat.both$Entity.x==dat.both$Entity.y)

## [1] 1

## Q3, part 2

(3 points)

Write code to find the four indices in “dat.both” at which the population takes on its minimum or maximum value and at which the density takes on its minimum or maximum value. Store the resulting indices in a vector named “inds”. Use of the which function can simplify this effort. The functions which.min and which.max may also be used. Please display the “Entity.x” values of the identified rows.

inds<-c(which.min(dat.both$population),which.max(dat.both$population), which.min(dat.both$density), which.max(dat.both$density))  
dat.both$Entity.x[inds]

## [1] "Nauru" "China" "Greenland" "Macao"

## Q3, part 3

(3 points)

Use “transmute” from dplyr to modify “dat.both” to be a data frame based on “dat.both”, but with the value of “Entity.x” in a variable labeled “entity”, the log of “density” in a variable labeled “den.log”, and the log of “Population” in a variable labeled “pop.log” and no other variables. Please display first 5 rows of the new version of “dat.both”.

dat.both <- transmute(dat.both, entity = Entity.x, den.log = log(density), pop.log= log(population))  
head(dat.both, n=5)

## entity den.log pop.log  
## 1 Afghanistan 4.088108 17.47723  
## 2 Albania 4.654237 14.87254  
## 3 Algeria 2.912948 17.59631  
## 4 American Samoa 5.620347 10.91866  
## 5 Andorra 5.102266 11.25500

Create and display a data frame “dat.text” from dat.both that includes only the rows containing the extremes identified in question 3, part 2.

dat.text <- dat.both[inds,]  
dat.text

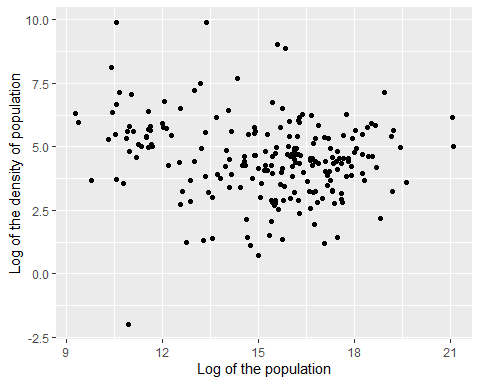
## entity den.log pop.log  
## 135 Nauru 6.294712 9.290445  
## 42 China 5.028593 21.087439  
## 78 Greenland -1.980502 10.946799  
## 115 Macao 9.890242 13.383715

## Q3, part 4

(2 points)

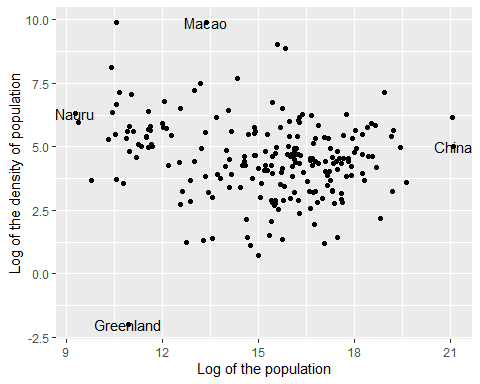
Use “ggplot” to create a point plot of the log of population (on the x-axis) versus the log of density. Store the plot in the variable g. Display the plot.

g <- ggplot(dat.both, mapping = aes(x = pop.log, y = den.log)) +  
 geom\_point()+  
 xlab("Log of the population")+  
 ylab("Log of the density of population")  
g



The following should give the previous plot with the names of the entities having extreme population or extreme density, assuming that the result of the “transmute” call was stored back in “dat.both”.

# Please uncomment and run:  
 g<-g+  
 geom\_text(data=dat.text,aes(x=pop.log,y=den.log,label=entity))  
 g



### Q3, part 5

(10 points)

Please add the least squares best fit line with “pop.log” as the -value and “den.log” as the -value in . Also plot the line minimizing the squared error again with “pop.log” as the -value and “den.log” as the -value in such a way that the points on the line are related by . That is, if is the function giving “pop.log” as an affine function of “den.log”, minimizing the square error , plot the inverse function .

y\_bar <- mean(dat.both$den.log)  
x\_bar <- mean(dat.both$pop.log)  
m<- (mean(dat.both$den.log\*dat.both$pop.log) - y\_bar\*x\_bar)/(mean(dat.both$pop.log^2) - x\_bar^2)  
b <- mean(dat.both$den.log - m\*dat.both$pop.log)  
f <- function(m,x,b){  
 return(m\*x+b)  
}  
g2 <- g + geom\_abline(slope=m,intercept = b)+  
 expand\_limits(x = 5, y = 15)# Plot of y = mx + b  
g2

