Problem Set 3 Solutions

Megan Hoeksema

## 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

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*. 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 to see whether the maximum likelihood estimates for and appear to be consistent and unbiased.

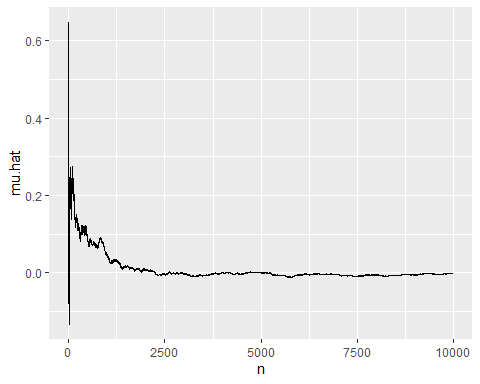
### Question 1.a

(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”. Please 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.

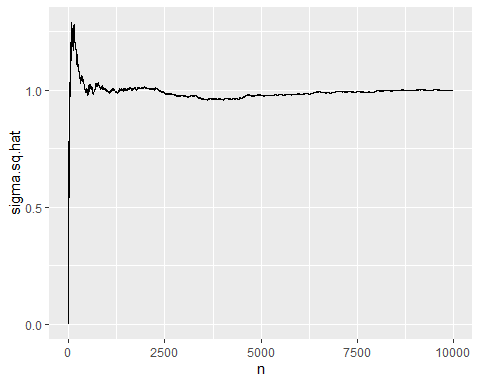
set.seed(12345)  
N<-10000  
samp<-rnorm(N)  
theta.est<-function(n,s=samp){  
 m<-mean(s[1:n])  
 s2<-sum((s[1:n]-m)^2)/n  
 return(c(m,s2))  
}  
dat.consist<-t(sapply(1:N,theta.est))  
dat.consist<-data.frame(dat.consist)  
dat.consist$n<-1:N  
names(dat.consist)<-c("mu.hat","sigma.sq.hat","n")  
  
ggplot(dat.consist,aes(x=n,y=mu.hat))+geom\_line()



dat.consist$mu.hat[c(10,100,500,1000,5000,10000)]

## [1] -0.1329441465 0.2451971965 0.0824607738 0.0461981587 0.0002191254  
## [6] -0.0006897654

ggplot(dat.consist,aes(x=n,y=sigma.sq.hat))+geom\_line()



dat.consist$sigma.sq.hat[c(10,100,500,1000,5000,10000)]

## [1] 0.5958317 1.2301986 0.9784340 0.9964993 0.9776419 0.9995999

### Conclusion: As n grows larger, the estimates for mu.hat get closer and closer to 0. As n grows larger, the estimates for sigma.sq.hat become closer and closer to 1.

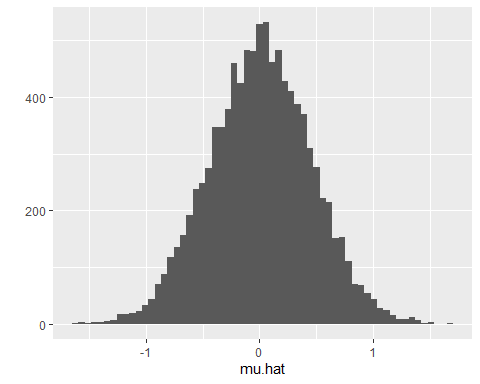
### Question 1.b

(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

Create 10,000 samples of size 5 from the standard Normal distribution and calculate the maximum likelihood estimates and of and for each sample. Compute the mean of the s and the mean of the s. Does the maximum likelihood estimate of seem to be unbiased? (You may repeat the experiment to help answer this question.) Does the maximum likelihood estimate of seem to be unbiased? Does the maximum likelihood estimate of seem to be unbiased? (Try comparing with the adjusted estimates produced by dividing the sum of the squared differences by 4 instead of 5.)

set.seed(45678)  
mat<-matrix(rnorm(10000\*5),ncol=5)  
  
mu.hat<-apply(mat,1,mean)  
 sigma2.ml<-function(x){sum((x-mean(x))^2)/5}  
sigma2.hat<-apply(mat,1,sigma2.ml)  
sigma2.b<-apply(mat,1,var)  
qplot(mu.hat, bins =60)



mean(mu.hat)

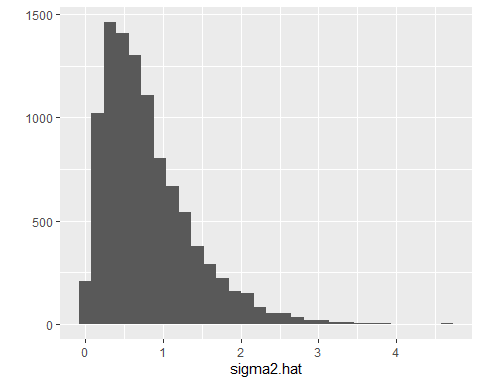
## [1] 0.008816334

mean(sigma2.hat)

## [1] 0.8049875

qplot(sigma2.hat)

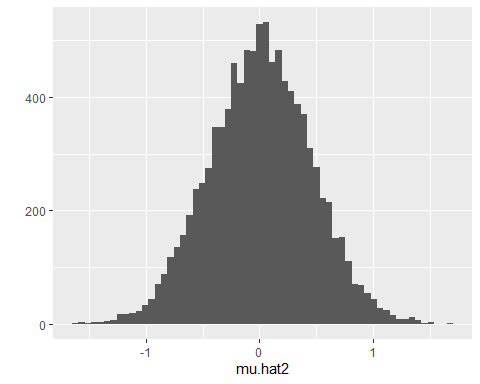
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



mean(sigma2.b)

## [1] 1.006234

## Divide by 4 instead of 5  
mu.hat2<-apply(mat,1,mean)  
 sigma2.2.ml<-function(x){sum((x-mean(x))^2)/4}  
sigma2.2.hat<-apply(mat,1,sigma2.2.ml)  
sigma2.2.b<-apply(mat,1,var)  
qplot(mu.hat2, bins =60)

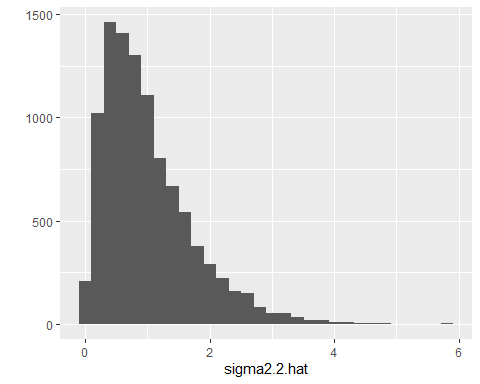


mean(sigma2.2.hat)

## [1] 1.006234

qplot(sigma2.2.hat)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



mean(sigma2.2.b)

## [1] 1.006234

##Conclusion: The maximum likelihood estimate of mu seems unnbiased. It remains close to 0, however the maximum likelihood estimate of sigma2 seems to be biased as dividing by 4 instead of 5 gives a better estimate of the variance.

## Question 2

(10 points)

The exponential distributions are a one parameter family of continuous distributions, . Given , the sample space is and the probability density function is . (The exponential distributions are used to model waiting times to events such as arrival of jobs in a queue.)

If are independent draws from an exponential distribution with parameter , the likelihood function of this sample is . Please derive the maximum likelihood value of as a function of . That is, given , what value of maximizes ?

The joint likelihood of the independent samples is

## Question 3

This question deals with the one parameter family of Poisson distributions, . For the parameter value , the sample space is and the density function is defined by for .

The Poisson distributions are among those used to model data that are counts of occurrences, such as the number of arrivals in a queue during a fixed time period. One can show that the Poisson distributions and the exponential distributions are related. If a phenomenon has the property that there exists a value such that the number of occurrences in any time interval of length has a Poisson distribution with parameter , then the distribution of the time to the next occurrence in an exponential distribution with parameter .

For very frequent occurrences, the counts may be approximated by Normal distributions to allow use of a wider range of statistical tools. The following questions examine the accuracy of this approximation.

### 3.a.

(5 points)

One step in approximating a Poisson distribution by a Normal distribution is to identify appropriate values of and .

For , create samples of size 100,000 from the Poisson distribution using “rpois” with parameter . For each sample, please find the maximum likelihood estimate of and if the samples are viewed as samples from a distribution.

lambdas<-c(4,25,100)  
set.seed(345678)  
mat.sim<-matrix(rep(NA,100000\*3),ncol=3)  
for(i in 1:3){  
 mat.sim[,i]<-rpois(100000,lambdas[i])  
}  
  
apply(mat.sim,2,mean)

## [1] 3.99768 25.00485 99.98159

apply(mat.sim,2,var)

## [1] 3.997375 25.111318 99.068502

var(1:3)

## [1] 1

var\_pop <- function(x) {  
 mean((x - mean(x))^2)  
}  
  
var\_pop(1:3)

## [1] 0.6666667

### 3.b.

(5 points)

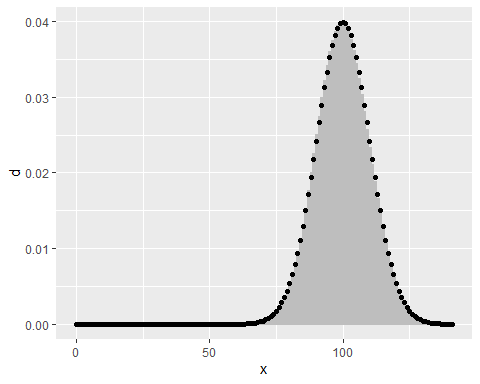
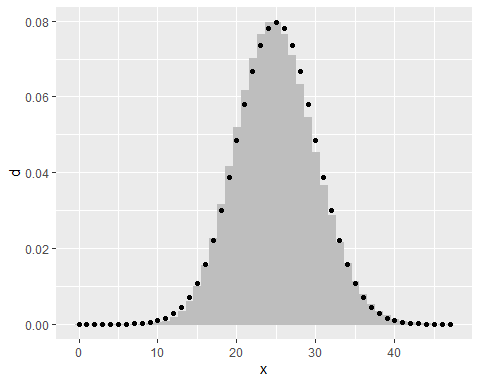
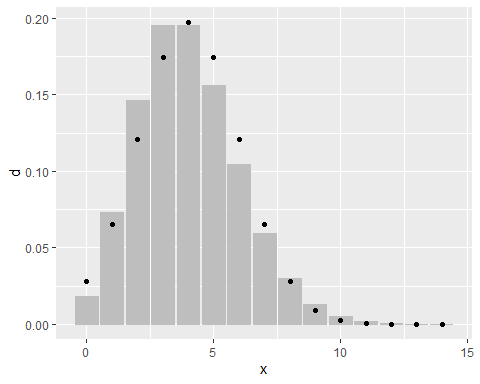
For the values of and for integers with the lower bound of 0 and the upper bounds of qpois(.99995,4), qpois(.99995,25), and qpois(.99995,100), respectively, provide visualizations comparing the probability of an outcome equal to under and the probability of the event under the Normal distribution with and .

The use of the probability of a value in under the Normal distribution adjusts for the fact that is a continuous distribution. Values in round to . For example, the probability of under , which equals 0.0132053, is approximated by the probability of the event under , 0.0129585.

Also, for each , give the sum of the absolute differences of the two probabilities, for all the values of assessed. A template is provided for this.

Does the proposed Normal approximation to improve as increases?

lambdas<-c(4,25,100)  
lim.max<-c(qpois(.99995,4), qpois(.99995,25), qpois(.99995,100))  
for(i in 1:length(lambdas)){  
 x<-0:lim.max[i]  
 # Vectorization allows the values to be computed for all x's   
 # at once.  
 lower<-pnorm(x-.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 upper<-pnorm(x+.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 p.pois<-dpois(x,lambda=lambdas[i])  
 p.norm<-upper-lower  
 dat.temp<-data.frame(x=0:lim.max[i],  
 d=dpois(0:lim.max[i],lambda=lambdas[i]),  
 d.approx=p.norm)  
 g<-ggplot(data=dat.temp,aes(x=x))+geom\_col(aes(y=d),color="gray",fill="gray")+  
 geom\_point(aes(y=d.approx))  
 print(g)  
}



lim.max<-c(qpois(.99995,4), qpois(.99995,25), qpois(.99995,100))  
for(i in 1:length(lambdas)){  
 x<-0:lim.max[i]  
 lower.prob<-pnorm(x-.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 upper.prob<-pnorm(x+.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 p.pois<-dpois(x,lambda=lambdas[i])  
 p.norm<-upper.prob-lower.prob  
 approx.error<-sum(abs(p.pois-p.norm))  
 print(approx.error)  
}

## [1] 0.1177445  
## [1] 0.05049299  
## [1] 0.02515794

### Conclusion: The normal approximation improves as lamda increases.

## Question 4

Please supply the missing code where indicated.

The data sets in these questions were downloaded 1/17/2020 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.csv",stringsAsFactors = FALSE)  
dat.den<-  
 read.csv("population-density.csv",stringsAsFactors = FALSE)  
names(dat.den)[4]<-"density"

### 4.a.

(2 points)

Write code to restrict both data frames to cases in which the value of “Year” is 2000 and the value of “Code” is not the empty string, "“, or the value for the whole world,”OWID\_WRL".

dat.pop<-filter(dat.pop,Year==2000,!Code %in% c("","OWID\_WRL"))  
dat.den<-filter(dat.den,Year==2000,!Code %in% c("","OWID\_WRL"))

Merge the data sets.

dat.both<-inner\_join(dat.den,dat.pop,by="Code")  
# check: this will equal 1 if the code above is correct.  
mean(dat.both$Entity.x==dat.both$Entity.y)

## [1] 1

### 4.b.

(3 points)

Write code to find the 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.

inds<-c(  
 which(dat.both$density %in%  
 c(max(dat.both$density),min(dat.both$density))),  
 which(dat.both$Population %in%  
 c(max(dat.both$Population),min(dat.both$Population)))  
)

### 4.c.

(3 points)

Use “transmute” from dplyr to create a data frame from dat.both with the value for “entity”,the log of “density” in “den.log”, and the log of “Population” in “pop.log”.

dat.both<-transmute(dat.both,den.log=log(density),  
 pop.log=log(Population),entity=Entity.x)

Create a data frame “dat.text” from dat.both that includes only the rows containing these extremes.

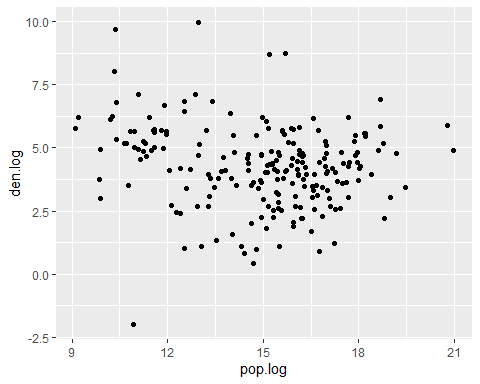
dat.text<-dat.both[inds,]

### 4.d.

(2 points)

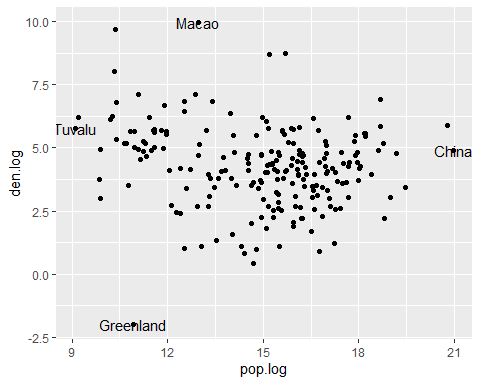
Use “ggplot” to create a point plot of the log of population (on the x-axis) versus the log of density. Store the plot in the variable g. Display the plot.

g<-ggplot(dat.both,aes(x=pop.log,y=den.log))+geom\_point()  
g



The following when uncommented 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”.

g<-g+  
 geom\_text(data=dat.text,aes(x=pop.log,y=den.log,label=entity))  
 g



### 4.e.

(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 .

lin.model<-lm(den.log~pop.log,data=dat.both)  
  
coeffs<-lm(den.log~pop.log,data=dat.both)$coefficients  
coeffs

## (Intercept) pop.log   
## 6.0776440 -0.1214985

coeffs.rev<-lm(pop.log~den.log,data=dat.both)$coefficients  
slope.rev<-1/coeffs.rev[2]  
intercept.rev<- (-coeffs.rev[1])/coeffs.rev[2]  
  
g<-g+  
 geom\_abline(slope=coeffs[2],intercept=coeffs[1],color="blue")  
g<-g+  
 geom\_abline(slope=slope.rev,intercept=intercept.rev,  
 color="orange")  
g

