Ward\_Abigail\_HW3

Abigail Ward

7/06/2021

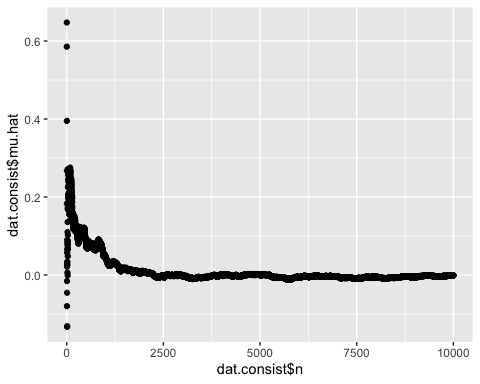
### Question 1.a

In the plots below, you can see that the maximum likelihood estimates of and are consistent and approach and as increases toward infinity.

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")  
  
maxlikelihoodplot<-ggplot(dat.consist,aes(x=dat.consist$n)) + geom\_point(data=dat.consist,aes(x=dat.consist$n, y=dat.consist$mu.hat))  
maxlikelihoodplot

## Warning: Use of `dat.consist$n` is discouraged. Use `n` instead.

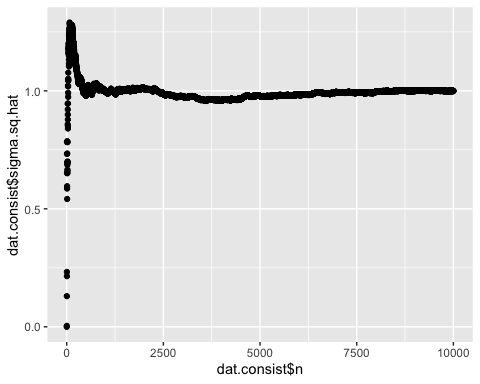
## Warning: Use of `dat.consist$mu.hat` is discouraged. Use `mu.hat` instead.



varianceplot<-ggplot(dat.consist,aes(x=dat.consist$n)) + geom\_point(data=dat.consist,aes(x=dat.consist$n, y=dat.consist$sigma.sq.hat))  
varianceplot

## Warning: Use of `dat.consist$n` is discouraged. Use `n` instead.

## Warning: Use of `dat.consist$sigma.sq.hat` is discouraged. Use `sigma.sq.hat`  
## instead.



### Question 1.b

The maximum likelihood estimate of does seem to be unbiased since the average of 10000 small samples is still very close to . The maximum likelihood estimate of seems to be biased since the expected value was , but the calculated value was . This bias was corrected by dividing the sum of the squared differences by 4 instead of 5.

set.seed(45678)  
mat<-matrix(rnorm(10000\*5),ncol=5)  
means<-c()  
sd<-c()  
newSD<-c()  
for (x in 1:10000){  
 m<-mean(mat[x,])  
 means[x] <-m  
 s<-sum((mat[x,]-m)^2)/5  
 sd[x]<-s  
 s2<-sum((mat[x,]-m)^2)/4  
 newSD[x]<-s2  
}  
  
avgMean<-mean(means)  
avgSD<-mean(sd)  
avgNewSD<-mean(newSD)  
avgMean

## [1] 0.008816334

avgSD

## [1] 0.8049875

avgNewSD

## [1] 1.006234

## Question 2

## Question 3.a

The maximum likelihood estimate of both and are equal to the given

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

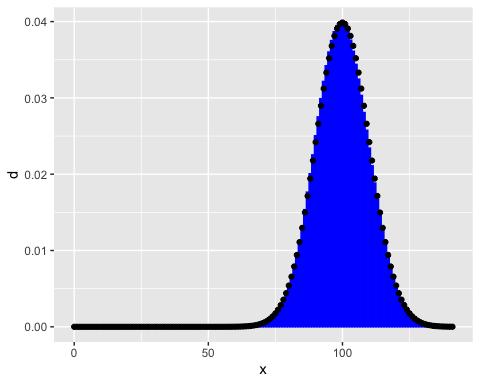
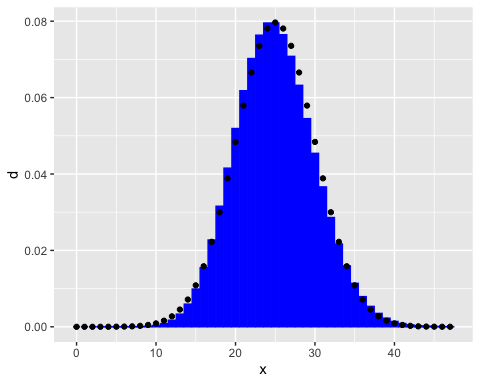
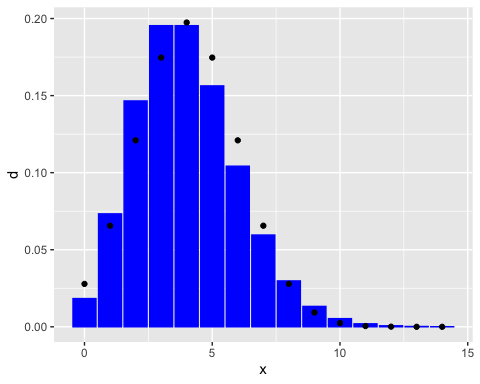
apply(mat.sim,2,var)

## [1] 3.997375 25.111318 99.068502

## Question 3.b

The proposed Normal approximation to the improves as increases as shown in the histograms better matching the normal distribution dots as well as the smaller sum of the absolute differences of the two probabilities.

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]  
 lower<-pnorm(x-.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 upper<-pnorm(x+.5,mean=lambdas[i],sd=sqrt(lambdas[i]))  
 p.norm<-upper-lower  
 instance<-data.frame(x=0:lim.max[i],  
 d=dpois(0:lim.max[i],lambda=lambdas[i]),  
 d.approx=p.norm)  
 g<-ggplot(data=instance,aes(x=x))+geom\_col(aes(y=d),color="blue",fill="blue")+  
 geom\_point(aes(y=d.approx))  
 print(g)  
}



for(i in 1:length(lambdas)){  
 x<-0:lim.max[i]  
 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  
 approx.error<-sum(abs(p.pois-p.norm))  
 print(approx.error)  
}

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

## Question 4

dat.pop<-read.csv("population.csv",stringsAsFactors = FALSE)  
dat.den<-  
 read.csv("population-density.csv",stringsAsFactors = FALSE)  
names(dat.den)[4]<-"density"  
  
#4a answer  
dat.pop<-filter(dat.pop,Year==2000,!Code %in% c("","OWID\_WRL"))  
dat.den<-filter(dat.den,Year==2000,!Code %in% c("","OWID\_WRL"))  
  
  
dat.both<-inner\_join(dat.den,dat.pop,by="Code")  
mean(dat.both$Entity.x==dat.both$Entity.y)

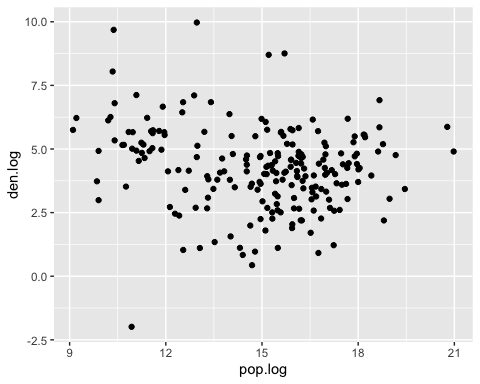
## [1] 1

#this is equal to 1

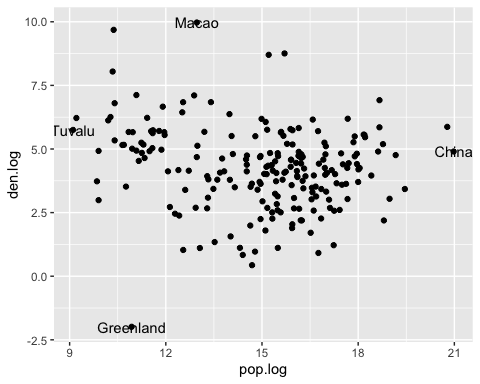
inds<-c(  
 which(dat.both$density %in%  
 c(max(dat.both$density),min(dat.both$density))),  
 which(dat.both$Population %in%  
 c(max(dat.both$Population),min(dat.both$Population)))  
)

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

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



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



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

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

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

