HW#1

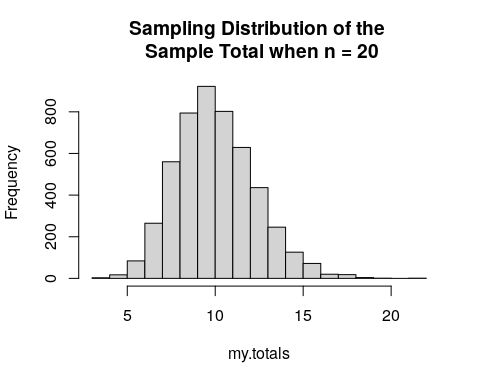
Matthew Maslow

1/31/2023

###17.

a.

n = 20 # sample size  
lambda = 2 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.totals = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.totals[i] = sum(my.sample)   
}  
hist(my.totals, main=paste("Sampling Distribution of the \n Sample Total when n =",n))



b.

mean(my.totals)

## [1] 10.02672

var(my.totals)

## [1] 5.045351

c.

mean(my.totals <= 10)

## [1] 0.529

Extra Part:

On the paper page and…

# check for part c  
pgamma(10, 20, 2)

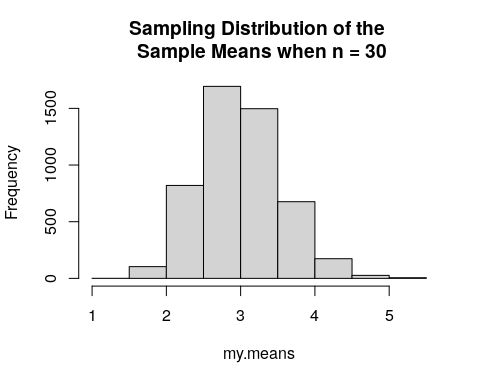
## [1] 0.5297427

The theoretical and simulation results match up quite well

###18.

a.

n = 30 # sample size  
lambda = 1/3 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.means = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.means[i] = mean(my.sample)   
}  
hist(my.means, main=paste("Sampling Distribution of the \n Sample Means when n =",n))



b.

# theoretical is written on paper page  
mean(my.means)

## [1] 2.997225

sd(my.means)

## [1] 0.5464647

c.

mean(my.means <= 3.5)

## [1] 0.8232

d.

If n=30, and if the CLT checks, then we can say that it is normally distributed, with a mean=3 and standard deviation = 3, for the theoretical results. The theoretical does not match/compare to the simulated results.

mean = 3  
sd = (3^2)/(30)  
pnorm(3.5, mean, sd)

## [1] 0.9522096

###19.

x~Exp(lambda=1/20)

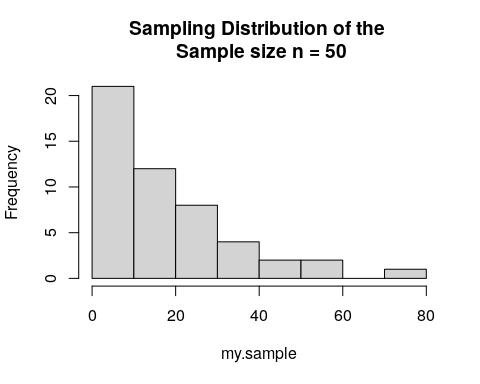
a.

qexp(0.5, 1/20)

## [1] 13.86294

b.

n = 50 # sample size  
lambda = 1/20 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
  
my.sample = round(rexp(n, lambda), 2)  
  
hist(my.sample, main=paste("Sampling Distribution of the \n Sample size n =",n))



The shape of the histogram of the sample of n=50 is very right skewed.

mean(my.sample)

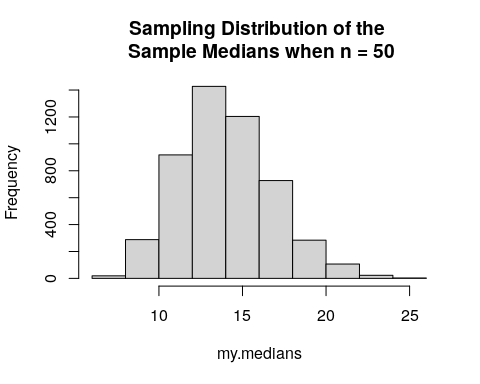
## [1] 17.6194

sd(my.sample)

## [1] 15.65762

c.

n = 50 # sample size  
lambda = 1/20 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.medians = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.medians[i] = median(my.sample)   
}  
hist(my.medians, main=paste("Sampling Distribution of the \n Sample Medians when n =",n))



The sampling distribution of this chart is right-skewed.

mean(my.medians)

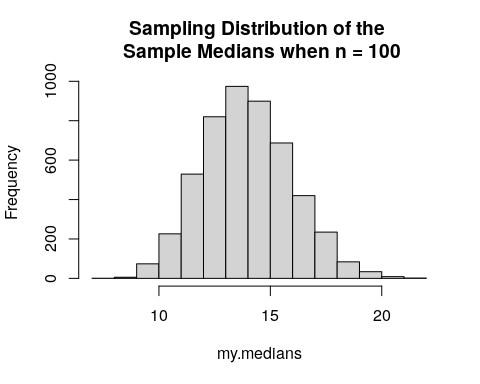
## [1] 14.01479

sd(my.medians)

## [1] 2.777449

d.

n = 100 # sample size  
lambda = 1/20 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.medians = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.medians[i] = median(my.sample)   
}  
hist(my.medians, main=paste("Sampling Distribution of the \n Sample Medians when n =",n))



Still seems to have a tiny bit of skewedness to the right, but almost normal

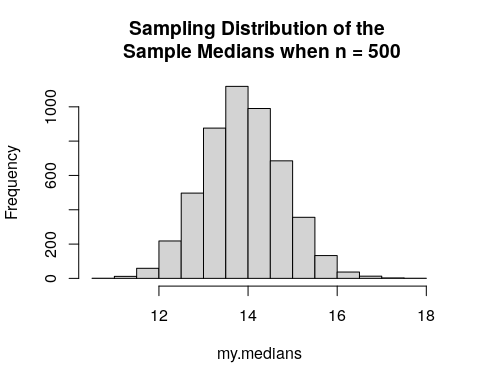
mean(my.medians)

## [1] 13.95894

sd(my.medians)

## [1] 2.005689

n = 500 # sample size  
lambda = 1/20 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.medians = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.medians[i] = median(my.sample)   
}  
hist(my.medians, main=paste("Sampling Distribution of the \n Sample Medians when n =",n))



This is almost normal

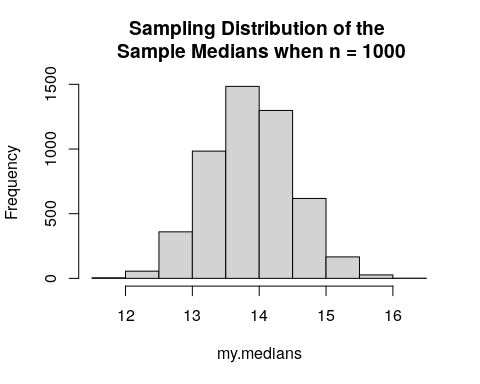
mean(my.medians)

## [1] 13.89482

sd(my.medians)

## [1] 0.8920781

n = 1000 # sample size  
lambda = 1/20 # lambda param  
mu = 1/lambda # population mean  
sigma = sqrt(1/lambda^2) # population standard deviation  
nsim = 5000  
  
set.seed(1)  
  
my.medians = numeric(nsim)   
  
for (i in 1:nsim) # loop through the iterations in the simulation  
{  
 my.sample = round(rexp(n, lambda), 2)  
 my.medians[i] = median(my.sample)   
}  
hist(my.medians, main=paste("Sampling Distribution of the \n Sample Medians when n =",n))



The dist of this chart is normal

mean(my.medians)

## [1] 13.87844

sd(my.medians)

## [1] 0.6325418

Overall as you increase the sample size, the shape of the histogram becomes less and less skewed, and becomes more normal.