Lab #1: Sampling Distribution of Means of Random Variables

Carley Dziewicki

Due Friday, April 17

Goals

1. Practice simulating and describing the behavior of independent random variables.
2. Explore the properties of distributions obtained by averaging independent random variables.
3. See the Central Limit Theorem for the sample mean in action through simulation experiments.

Problems

For each of the following, submit the required R commands along with your answers below. Refer to the scripts provided in the Section 5.5 and 6.1 lesson and screencasts for help writing your code. [**Hint**: Problem 1 does not require any for-loops but Problems 2 through 5 should demonstrate that you know how to use the for-loop construction.]

1. Let Z be a random variable with a Standard Normal distribution, let X be a random variable from a Uniform (0,1) distribution and let Y be a random variable with an Exponential (1) distribution. Use R to generate 1000 observations on Z, Y, and X. Obtain a histogram of the 1000 observations for each of the three variables. Confirm that they have the characteristics (shape, mean, and standard deviation) that you would expect.

Answer: The histograms of X, Y, and Z can be seen below. The shape, mean and standard deviation are all what I would expect. The normal histogram has a normal shape, and the unifrom looks approxiamtly normal and the exponenital also looks to be the correct shape. The mean and standard deviation of all the curves seems to be approximatly what we should expect for their respective distribution.

```
par(mfrow = c(2,2))
Z<-(rnorm(1000))
mean(Z)
```

```
## [1] 0.008228854
```

```
sd(Z)
```

```
## [1] 0.9990282
```

```
hist(Z)
X<- (runif(1000))
mean(X)
```

```
## [1] 0.5028909
```

```
sd(X)
```

```
## [1] 0.2844612
```

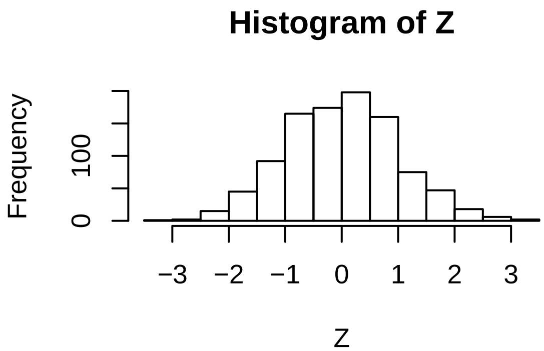
```
hist(X)
Y<- rexp(1000)
mean(Y)
```

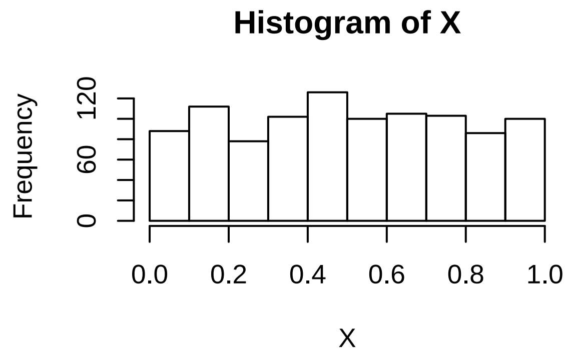
```
## [1] 0.9442483
```

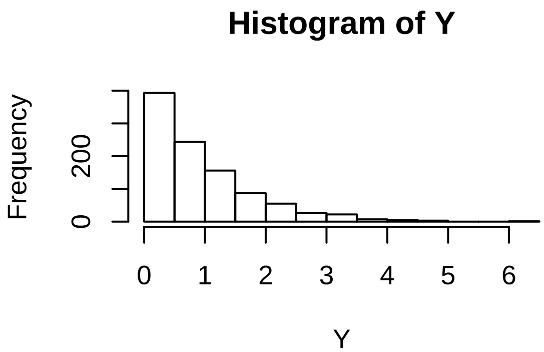
```
sd(Y)
```

```
## [1] 0.876732
```

```
hist(Y)
```







2. Use R to simulate the sampling distribution of the sample mean for samples of size 2 from a Standard Normal distribution (Z). Repeat this simulation using the following sample sizes: 5, 10, and 30. Include your R script, histograms and Normal probability plots of the sampling distribution of the sample mean, and a table of means and standard deviations for the sample mean for each value of n.

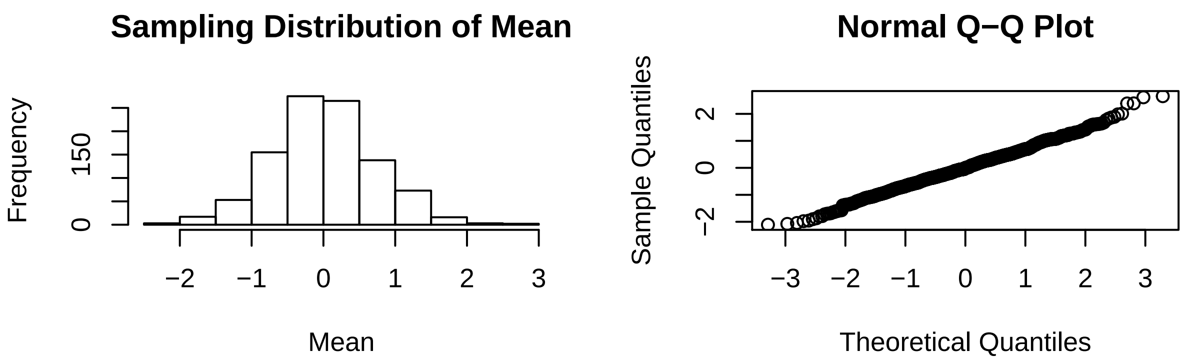
    - **Note 1:** Constructing even a simple table in R Markdown can be a bother. You can use this website https://www.tablesgenerator.com/markdown_tables to create a table and then generate the code to copy into this document. Let me know if you have questions about it.)

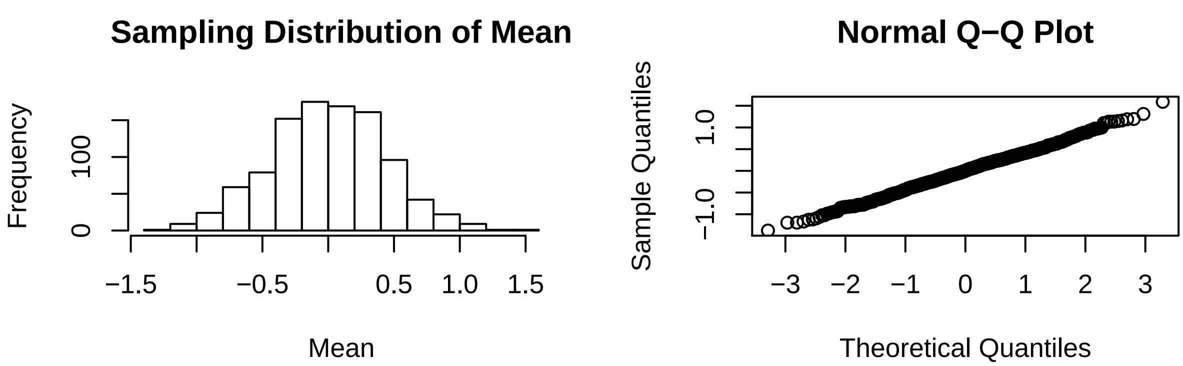
    - **Note 2:** For your Normal probability plot here and on subsequent problems, you may simply use the built-in R command `qqnorm()` since you have already demonstrated that you know how to construct this plot from scratch. The `qqnorm()` function only needs one argument, the vector of observed values.

Answer:

```
par(mfrow = c(2,2))

M <- 1000 # number of replications
n <- 2 # sample size
```

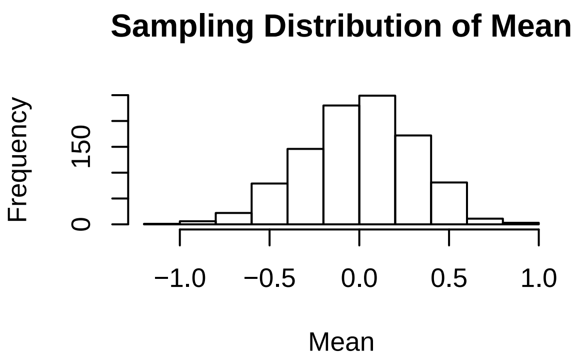
```r
tstata <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rnorm(n) # Draw sample of size n
tstata[i] <- mean(xvec) }
hist(tstata, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstata)

n <- 5 # sample size
tstatb <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rnorm(n) # Draw sample of size n
tstatb[i] <- mean(xvec) }
hist(tstatb, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatb)
```

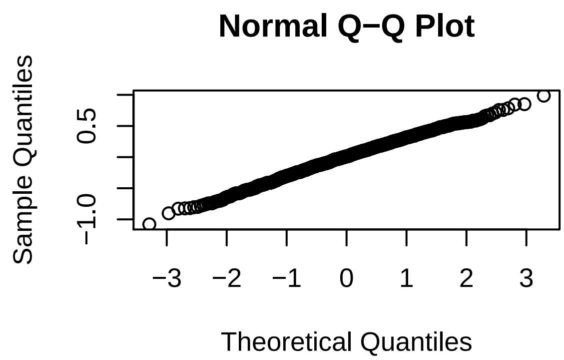


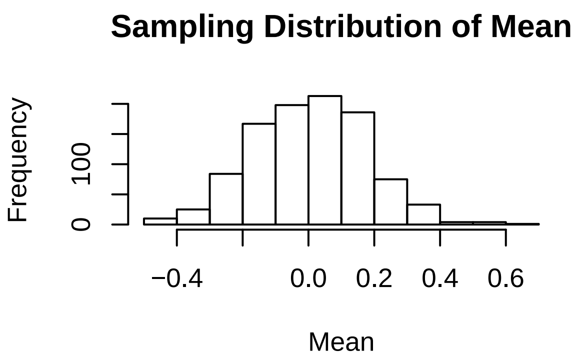


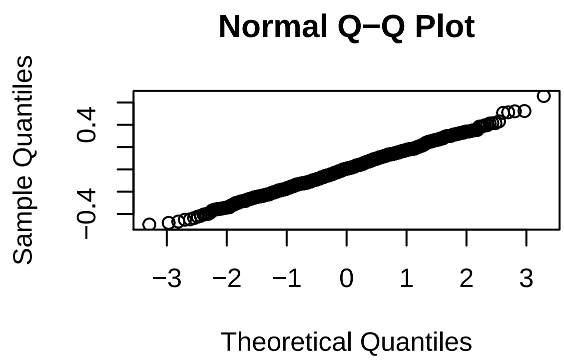

```r
n <- 10 # sample size
tstatc <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rnorm(n) # Draw sample of size n
tstatc[i] <- mean(xvec) }
hist(tstatc, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatc)

n <- 30 # sample size
tstatd <- numeric(M) # # space to save the statistic values
for(i in 1:M)
```

```r
{
xvec <- rnorm(n) # Draw sample of size n
tstatd[i] <- mean(xvec) }
hist(tstatd, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatd)
```









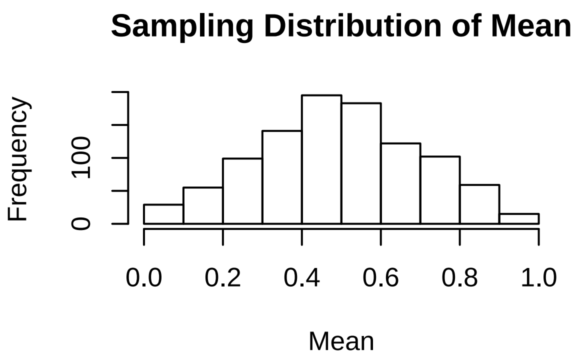
n	mean	sd
2	-0.0493	0.699
5	-0.0293	0.442
10	-0.0007	0.323
30	-.00753	0.184

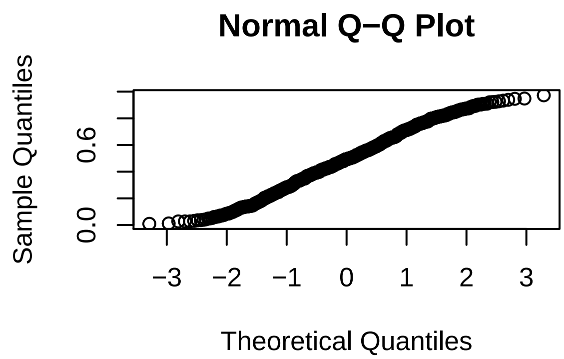
3. Repeat the previous problem but use the Uniform (0,1) distribution X instead of the Standard Normal distribution.

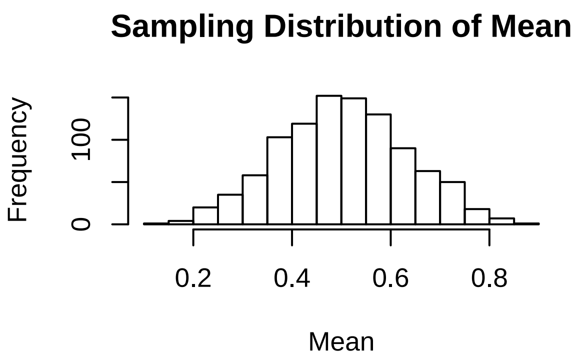
Answer:

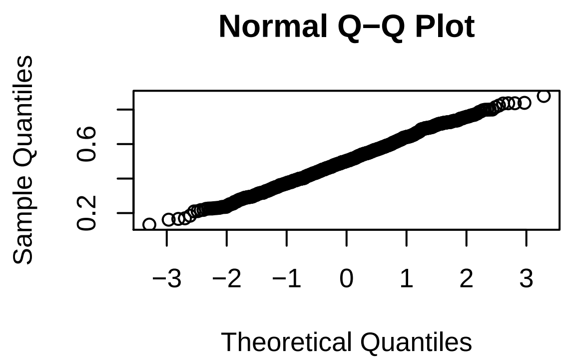
```r
par(mfrow = c(2,2))

M <- 1000 # number of replications
a<- 0
b<- 1
n <- 2 # sample size
tstata <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- runif(n,a,b) # Draw sample of size n
tstata[i] <- mean(xvec) }
```

```r
hist(tstata, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstata)

n <- 5 # sample size
tstatb <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- runif(n,a,b) # Draw sample of size n
tstatb[i] <- mean(xvec) }
hist(tstatb, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatb)
```

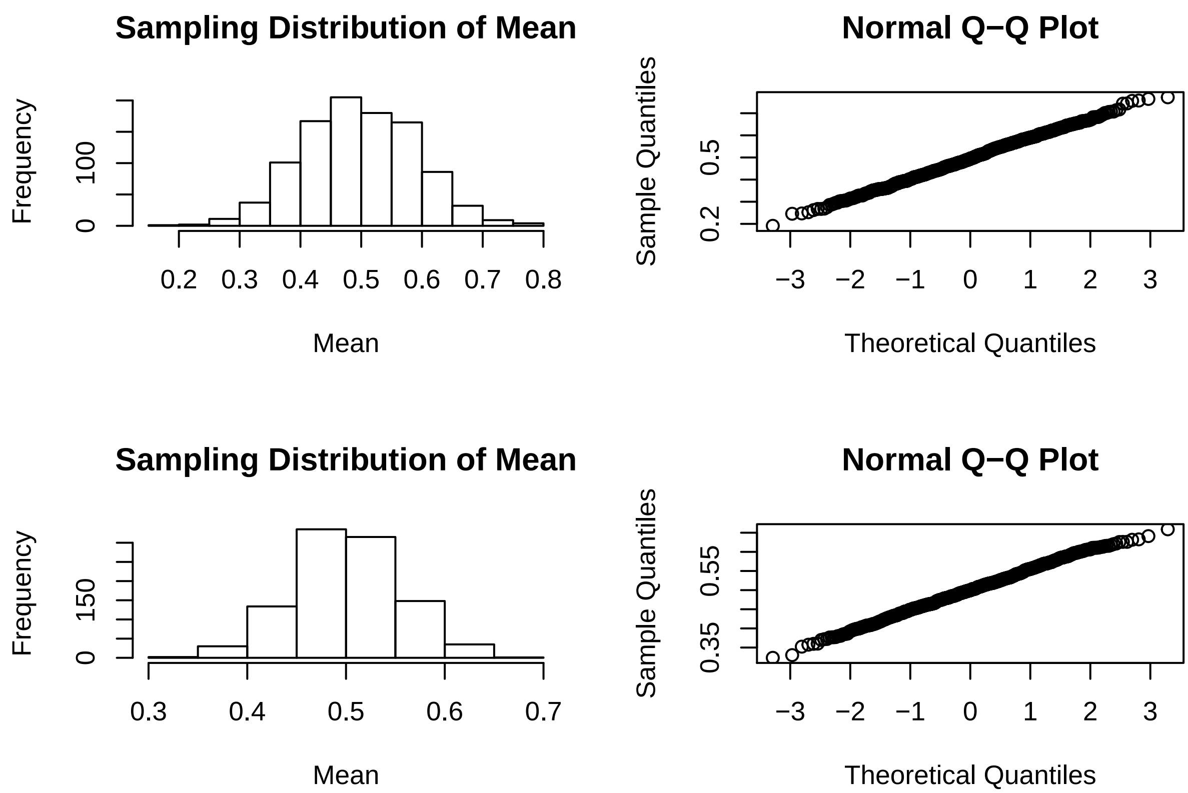









```r
n <- 10 # sample size
tstatc <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- runif(n,a,b) # Draw sample of size n
tstatc[i] <- mean(xvec) }
hist(tstatc, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatc)

n <- 30 # sample size
tstatd <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- runif(n,a,b) # Draw sample of size n
tstatd[i] <- mean(xvec) }
hist(tstatd, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatd)
```



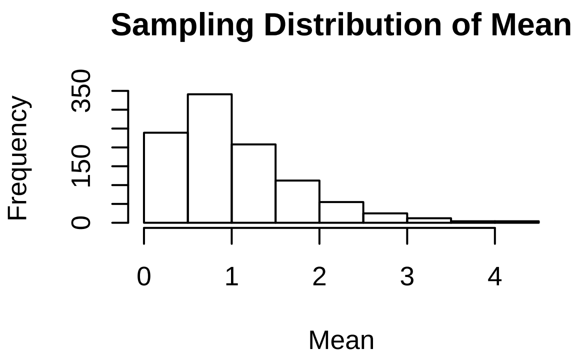
n	mean	sd
2	.509	.207
5	.497	.128
10	.497	.091
30	.498	.051

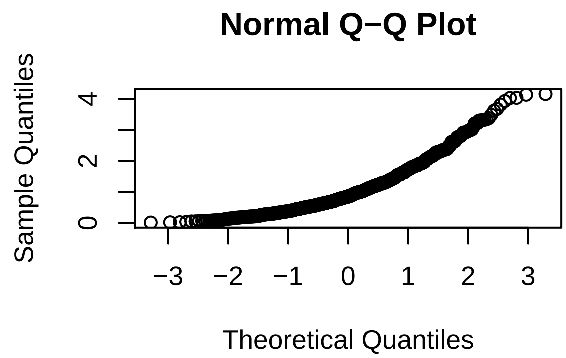
4. Again repeat the previous problem but use the Exponential (1) distribution Y this time.

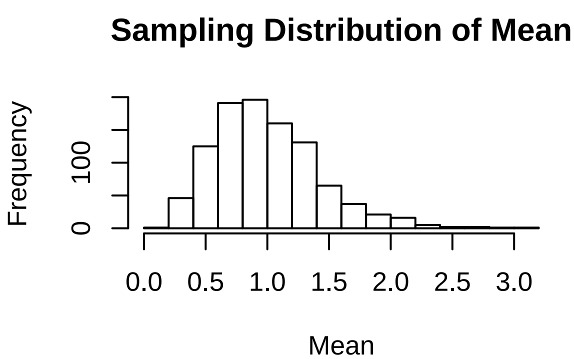
Answer:

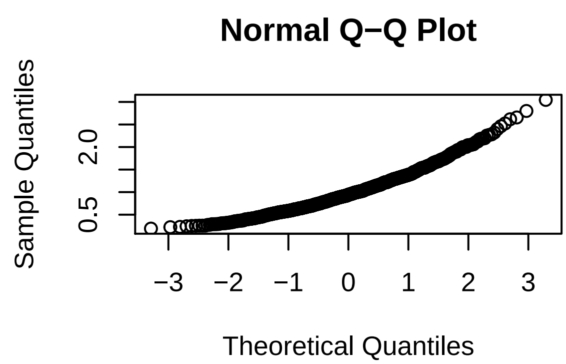
```
par(mfrow = c(2,2))

M <- 1000 # number of replications
n <- 2 # sample size
tstata <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rexp(n) # Draw sample of size n
tstata[i] <- mean(xvec) }
hist(tstata, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstata)

n <- 5 # sample size
tstatb <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rexp(n) # Draw sample of size n
```

```
tstatb[i] <- mean(xvec) }
hist(tstatb, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatb)
```

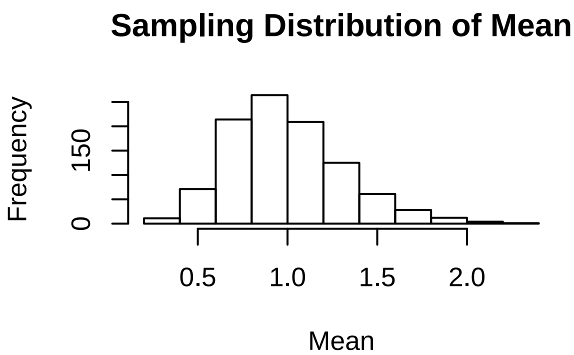


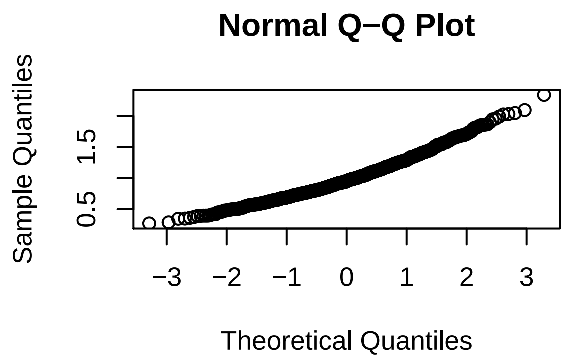


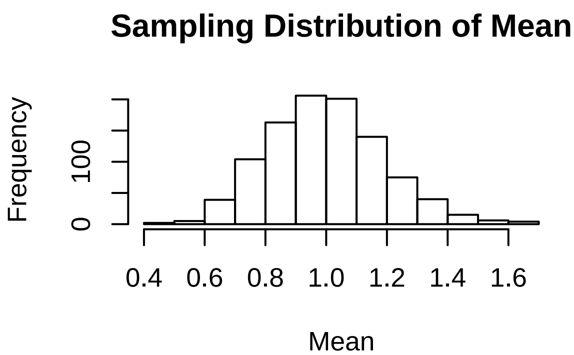


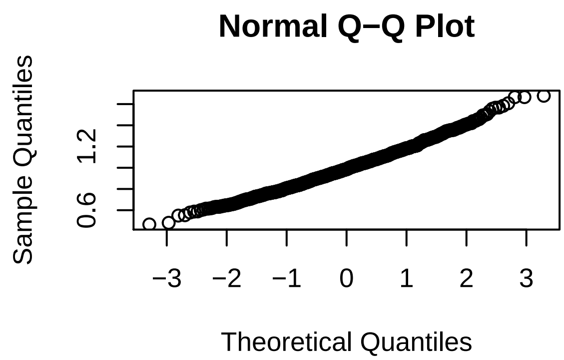



```
n <- 10 # sample size
tstatc <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rexp(n) # Draw sample of size n
tstatc[i] <- mean(xvec) }
hist(tstatc, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatc)

n <- 30 # sample size
tstatd <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rexp(n) # Draw sample of size n
tstatd[i] <- mean(xvec) }
hist(tstatd, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatd)
```









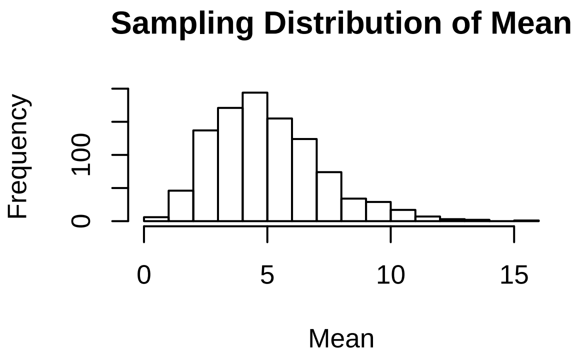
n	mean	sd
2	.996	.689
5	.996	.455
10	1.011	.312
30	1.005	.187

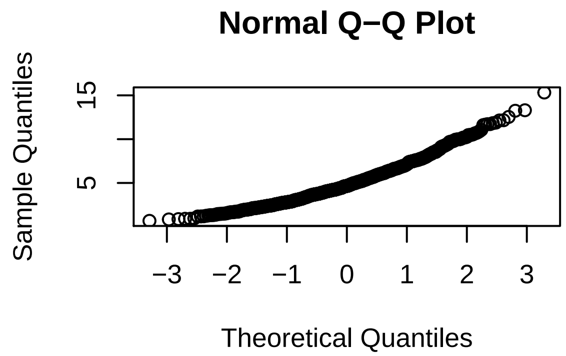
5. Repeat the previous problem but use a different non-Normal continuous distribution of your choosing, e.g., Gamma (specify α and β), Chi-Square with 5 degrees of freedom, or Weibull but not the Cauchy distribution. [**Hint:** You will need to determine the appropriate function for drawing a random sample.]

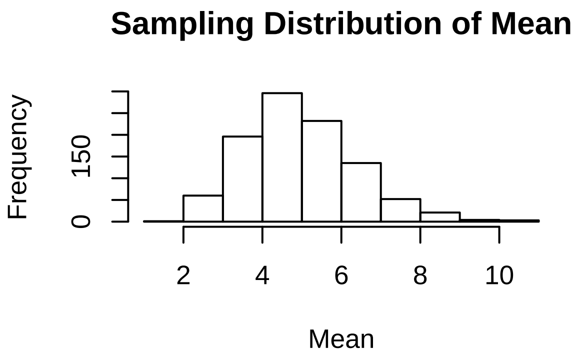
Answer:

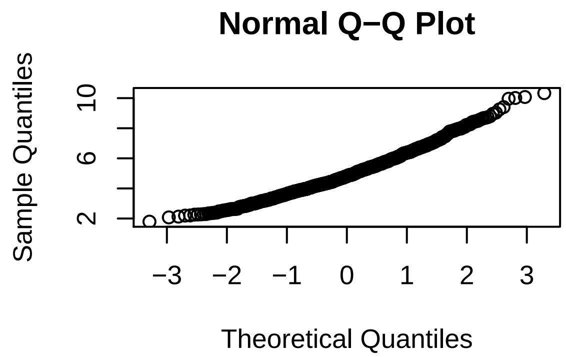
```
par(mfrow = c(2,2))

M <- 1000 # number of replications
n <- 2 # sample size
tstata <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rchisq(n, df=5) # Draw sample of size n
tstata[i] <- mean(xvec) }
hist(tstata, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstata)

n <- 5 # sample size
tstatb <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
```

```
xvec <- rchisq(n, df=5) # Draw sample of size n
tstatb[i] <- mean(xvec) }
hist(tstatb, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatb)
```

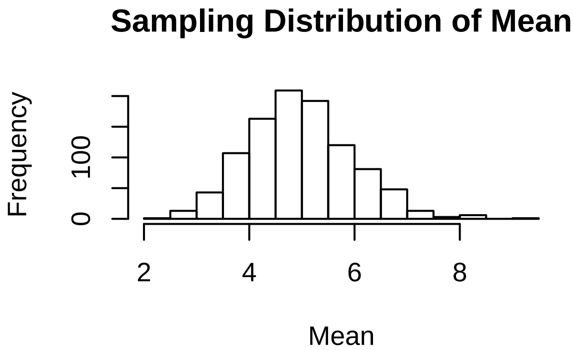


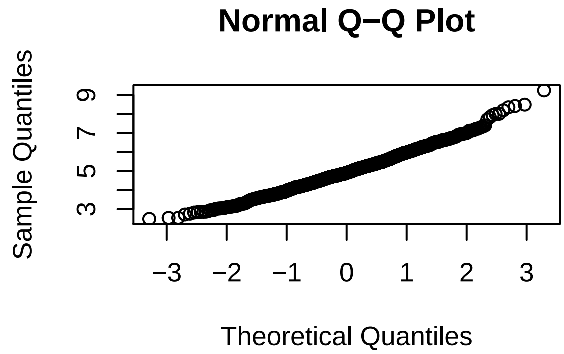


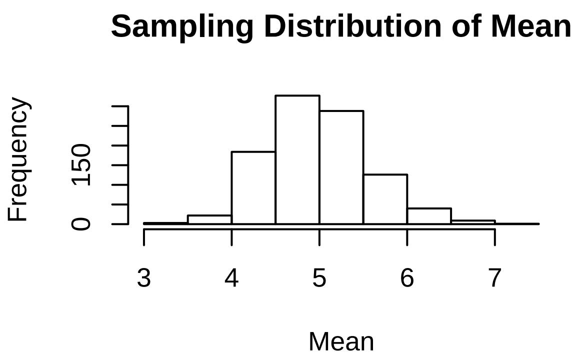


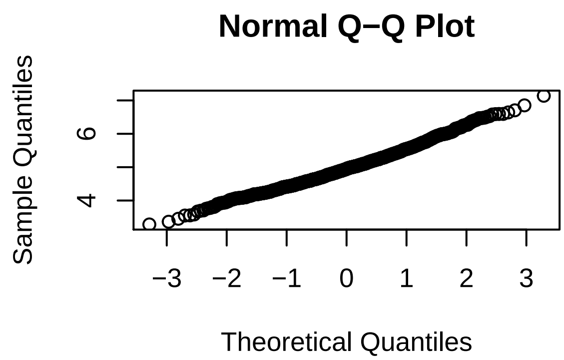



```
n <- 10 # sample size
tstatc <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rchisq(n, df=5) # Draw sample of size n
tstatc[i] <- mean(xvec) }
hist(tstatc, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatc)

n <- 30 # sample size
tstatd <- numeric(M) # # space to save the statistic values
for(i in 1:M)
{
xvec <- rchisq(n, df=5) # Draw sample of size n
tstatd[i] <- mean(xvec) }
hist(tstatd, main = "Sampling Distribution of Mean", xlab = "Mean")
qqnorm(tstatd)
```









n	mean	sd
2	5.033	2.24
5	4.960	1.41
10	5.058	.969
30	5.061	.595

6. Summarize your findings from these simulations. Describe the shapes of the sampling distributions, comment on how the shape changes as n increases and comment on similarities and differences across the three distributions. Also, note how the standard deviation of the sample mean changes as n increases.

Answer: To summarize my findings from the simulations it seems in all cases that the larger the sample size n increases the more accurate the histogram is to the normal curve. For the normal distibution the mean approached 0 and the sd deviation got smaller when n increased. For the uniform distribution, the curve also approached normal, however the mean approached .5 and the sd decreased when n increased, but we would expect a mean of .5 from a standard uniform function. For the exponential a similar trend to the others, in that the curve approached normal, mean was 1 and sd decreased, which is in line of a exponential funtion. Finally I used the Chi-sq distribtion and the curve also approched normal, then mean 5 and sd started quite large but also decreased when n got larger. In conclusion, for all the distibutions the sampleing distribution of the mean curve got closer to normal as n got larger and the mean and sd got closer to expectations and less variable.