

# HW 1

*Tim Vigers*

*September 1, 2018*

## Biostatistical Methods 1, Homework 1

### Exercise 1

Reproducibly simulate a sample of 10,000 from each of the following distributions. Determine the theoretical mean and standard deviation for each distribution and verify that the generated numbers have approximately the correct mean and standard deviation. Create a histogram and boxplot depicting each of the mock samples.

**Normal Distribution (m=125, s=8):**

```
set.seed(1017)
sample_size <- 10000
simvalsnormal <- rnorm(n = sample_size, mean = 125, sd = 8)
mean(simvalsnormal)
```

```
## [1] 124.8823
```

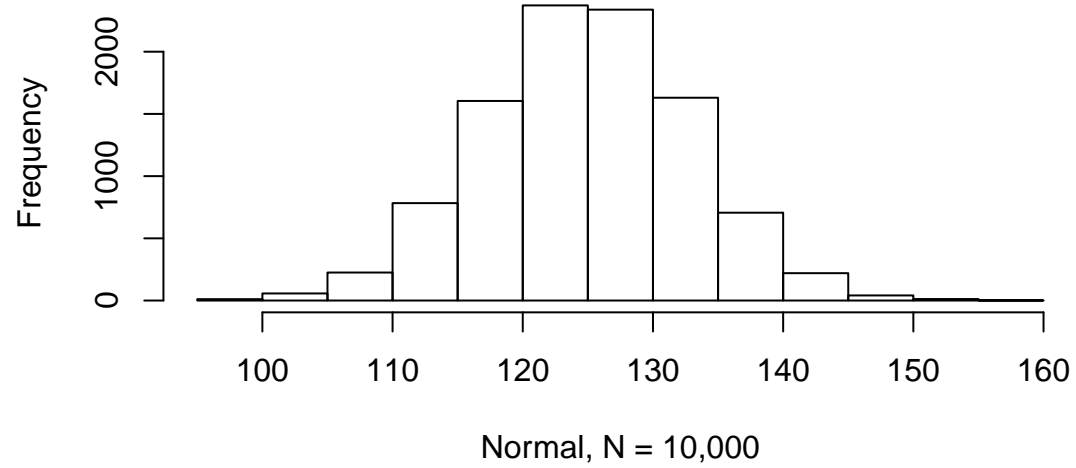
```
sd(simvalsnormal)
```

```
## [1] 7.919632
```

The means and sd of this sample are very close to the theoretical mean and sd of the distribution (125 and 8, respectively).

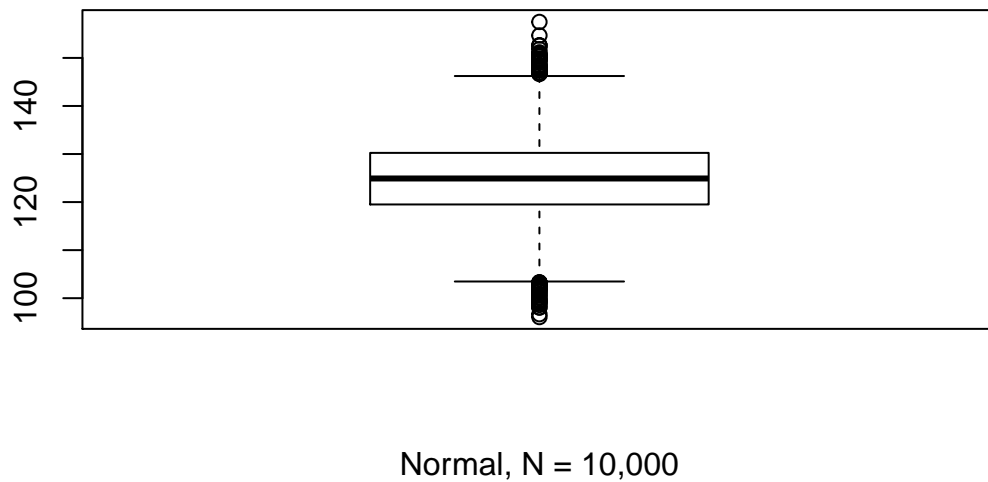
Histogram:

```
hist(simvalsnormal,main = "", xlab = "Normal, N = 10,000")
```



Boxplot:

```
boxplot(simvalsnormal, xlab = "Normal, N = 10,000")
```



### Poisson Distribution ( $\lambda=1.5$ )

```
set.seed(1017)
sample_size <- 10000
simvalspoisson <- rpois(sample_size, lambda = 1.5)
mean(simvalspoisson)
```

```
## [1] 1.4972
```

```
sd(simvalspoisson)
```

```
## [1] 1.226924
```

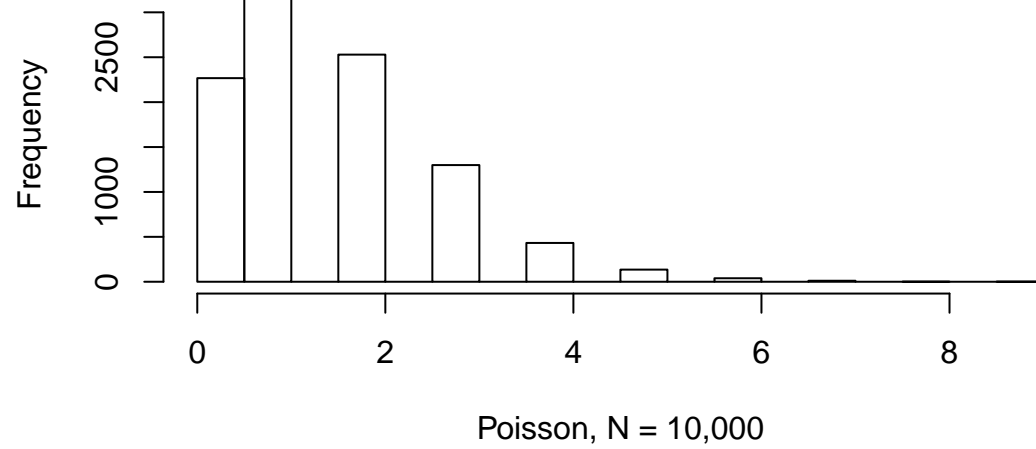
For a Poisson distribution, the mean and variance are both equal to  $\lambda$ . So here the sample mean is very close to the population mean of 1.5. Standard deviation is the square root of the variance, so for this population we would expect it to be

$$\sqrt{1.5} = 1.224745$$

This is close to our sample SD.

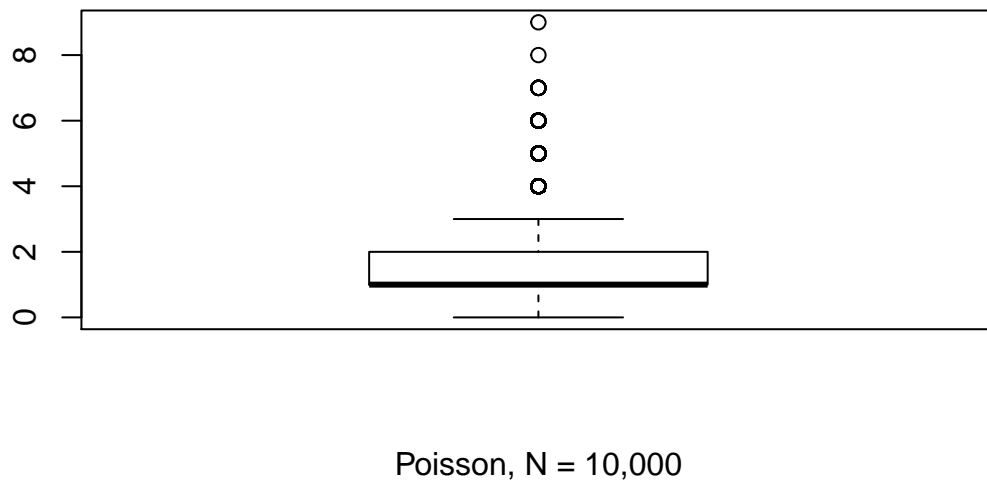
Histogram:

```
hist(simvalspoisson,main = "", xlab = "Poisson, N = 10,000")
```



Boxplot:

```
boxplot(simvalspoisson, xlab = "Poisson, N = 10,000")
```



### Binomial Distribution (n=5, p=0.15)

```
set.seed(1017)
sample_size <- 10000
simvalsbinom <- rbinom(sample_size, size = 5, prob = 0.15)
mean(simvalsbinom)
```

```
## [1] 0.7511
```

```
sd(simvalsbinom)
```

```
## [1] 0.7985065
```

The mean of a binomial distribution is  $np$ , so in this case

$$5 * 0.15 = 0.75$$

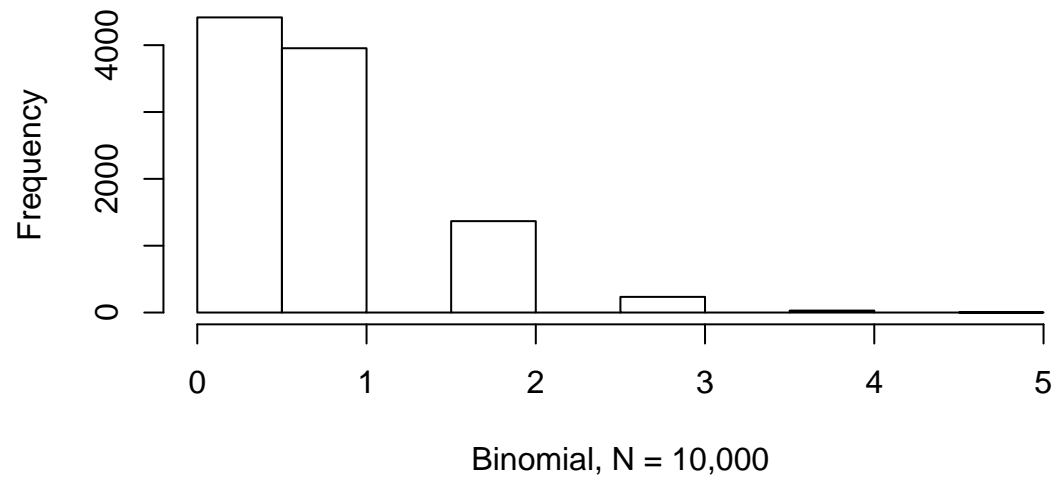
Our sample mean of 0.7511 is pretty close. The variance of a binomial distribution is  $np(1-p)$ , so in this case the standard deviation will be

$$\sqrt{(0.75 * (1 - 0.15))} = 0.798436$$

Again our sample seems to be approximating the distribution well.

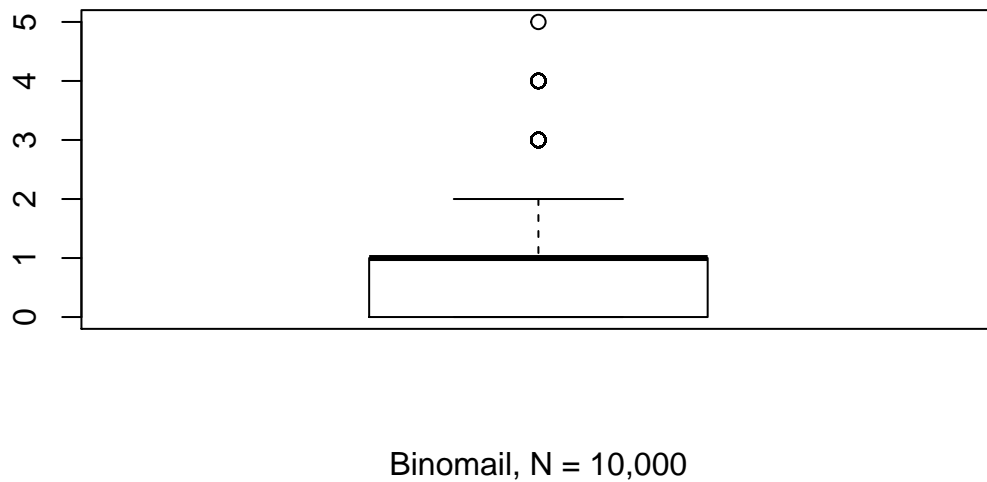
Histogram:

```
hist(simvalsbinom,main = "", xlab = "Binomial, N = 10,000")
```



Boxplot:

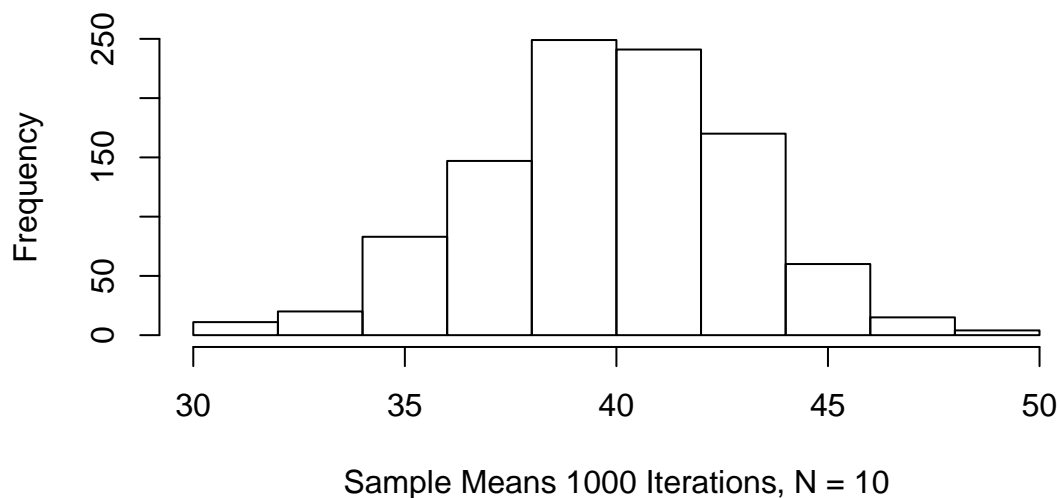
```
boxplot(simvalsbinom, xlab = "Binomial, N = 10,000")
```



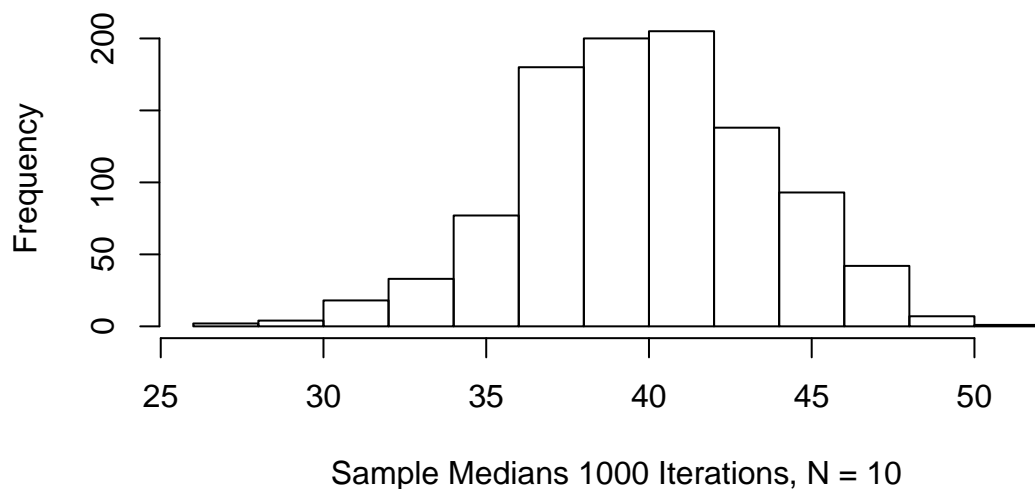
## Exercise 2

- a. For a population that is normally distributed with mean 40 and standard deviation 10, generate histograms showing the sampling distribution of the mean, median, and variance. Use 1,000 simulation iterations and a sample size of  $n = 10$ .

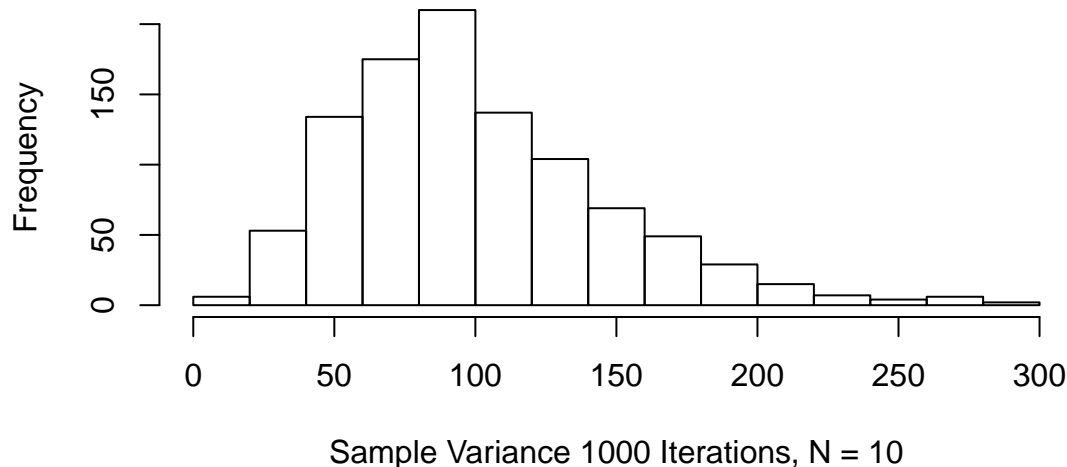
```
# Set up the simulation.
set.seed(1017)
number_of_sims <- 1000
sample_size <- 10
# Create a vector to store sample means, median, and variance.
vector_of_sample_means <- rep(-9, number_of_sims)
vector_of_sample_medians <- rep(-9, number_of_sims)
vector_of_sample_variance <- rep(-9, number_of_sims)
# For loop to generate values and store in their respective vectors.
for (i in 1:number_of_sims) {
  vector_of_sample_means[i] <- mean(rnorm(n = sample_size, mean = 40, sd = 10))
  vector_of_sample_medians[i] <- median(rnorm(n = sample_size, mean = 40, sd = 10))
  vector_of_sample_variance[i] <- var(rnorm(n = sample_size, mean = 40, sd = 10))
}
# Plot histograms
hist(vector_of_sample_means, main = "", xlab = "Sample Means 1000 Iterations, N = 10")
```



```
hist(vector_of_sample_medians, main = "", xlab = "Sample Medians 1000 Iterations, N = 10")
```



```
hist(vector_of_sample_variance, main = "", xlab = "Sample Variance 1000 Iterations, N = 10")
```



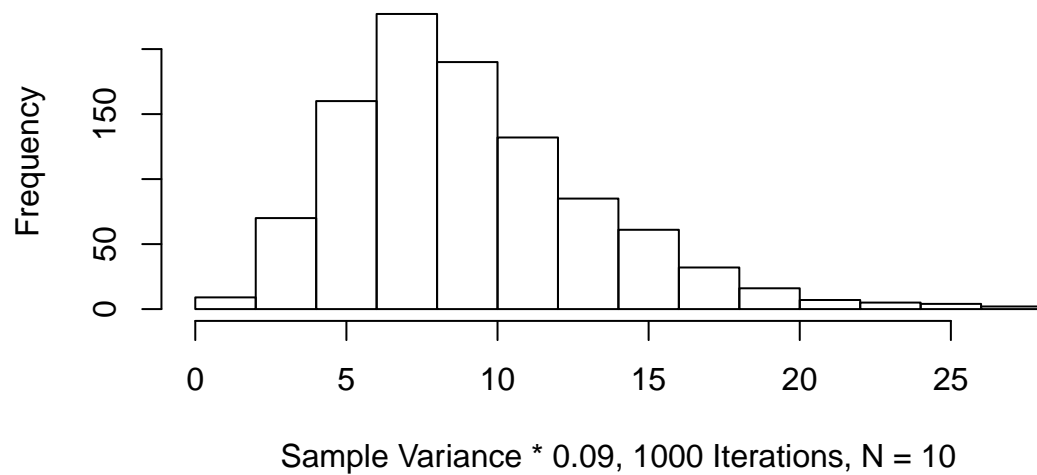
- b. According to theory, when the population is distributed as a normal with mean  $m$  and standard deviation  $s$ , the sample mean is  $\bar{X} \sim \text{Normal}(m, s/\sqrt{n})$ . So here the sampling distribution of the mean should be normal (same for the median since the mean, median, and mode of a normal distribution are equal).
- c. The `dchisq()` function requires that you specify quantiles and degrees of freedom. So to plot the theoretical distribution we can use this function setting  $df = 9$ . We don't need to look at the whole distribution, so quantiles 1 through 30 should be plenty. Also, as recommended in the hint, I've multiplied the sample variance vector from 2a by a factor of  $9/100$ . The shapes of the two plots are



very similar, both with a right skew.

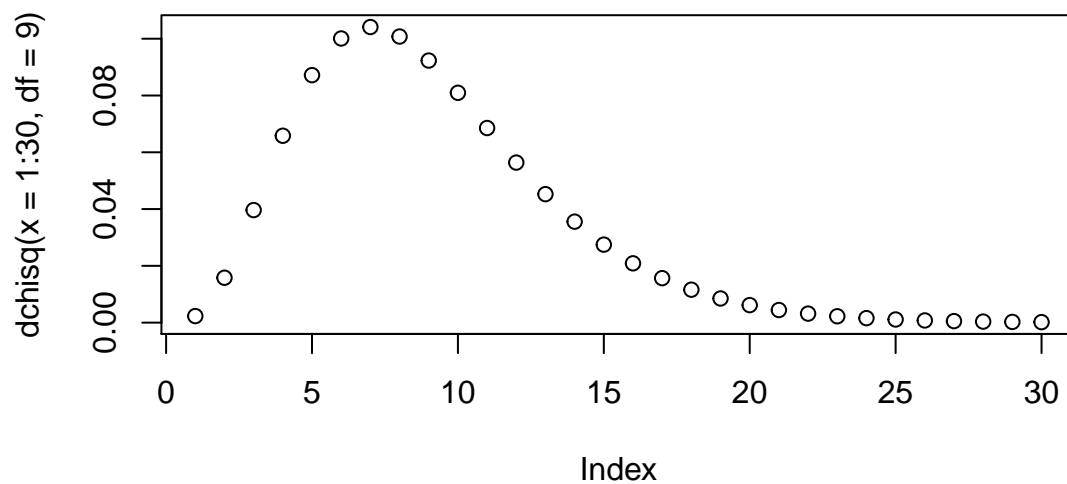
```
# Plot histogram from samples.
```

```
hist(vector_of_sample_variance * (9/100), main = "", xlab = "Sample Variance * 0.09, 1000 Iterations, N",
```



```
# Plot theoretical distribution.
```

```
plot(dchisq(x = 1:30, df = 9))
```



### Exercise 3

- a. Generate and save a vector containing 500 sample means (i.e., five-hundred simulation iterations) of sample size 10 from a Binomial ( $n = 1$ ,  $p = 0.15$ ) population (recall, in `rbinom()` `size=1` and `n=10`).

```
# Set up the simulation, like in exercise 2.
set.seed(1017)
number_of_sims <- 500
sample_size <- 10
# Create a vector to store sample means.
vector_of_sample_means_10 <- rep(-9, number_of_sims)
# For loop to generate values and store in the vector, but with a binomial distribution.
for (i in 1:number_of_sims) {
  vector_of_sample_means_10[i] <- mean(rbinom(n = sample_size, size = 1, prob = 0.15))
}
```

- b. Repeat for sample sizes of  $n = 20$ ,  $n = 30$ ,  $n = 40$ , and  $n = 50$ .

```
# Create vectors for different sample sizes.
vector_of_sample_means_20 <- rep(-9, number_of_sims)
vector_of_sample_means_30 <- rep(-9, number_of_sims)
vector_of_sample_means_40 <- rep(-9, number_of_sims)
vector_of_sample_means_50 <- rep(-9, number_of_sims)
# For loops to generate values and store in the vectors.
for (i in 1:number_of_sims) {
  vector_of_sample_means_20[i] <- mean(rbinom(n = 20, size = 1, prob = 0.15))
  vector_of_sample_means_30[i] <- mean(rbinom(n = 30, size = 1, prob = 0.15))
  vector_of_sample_means_40[i] <- mean(rbinom(n = 40, size = 1, prob = 0.15))
  vector_of_sample_means_50[i] <- mean(rbinom(n = 50, size = 1, prob = 0.15))
}
```

- c. Calculate the mean and standard deviation associated with each of the five sets of  $\bar{x}$  values.

```
mean(vector_of_sample_means_10)
```

```
## [1] 0.1484
```

```
sd(vector_of_sample_means_10)
```

```
## [1] 0.1150823
```

```
mean(vector_of_sample_means_20)
```

```
## [1] 0.1473
```

```
sd(vector_of_sample_means_20)
```

```
## [1] 0.07823011
```

```
mean(vector_of_sample_means_30)
```

```
## [1] 0.1514
```

```
sd(vector_of_sample_means_30)
```

```
## [1] 0.06569292
```

```
mean(vector_of_sample_means_40)
```

```
## [1] 0.14425
```

```
sd(vector_of_sample_means_40)
```

```
## [1] 0.0537491
```

```
mean(vector_of_sample_means_50)
```

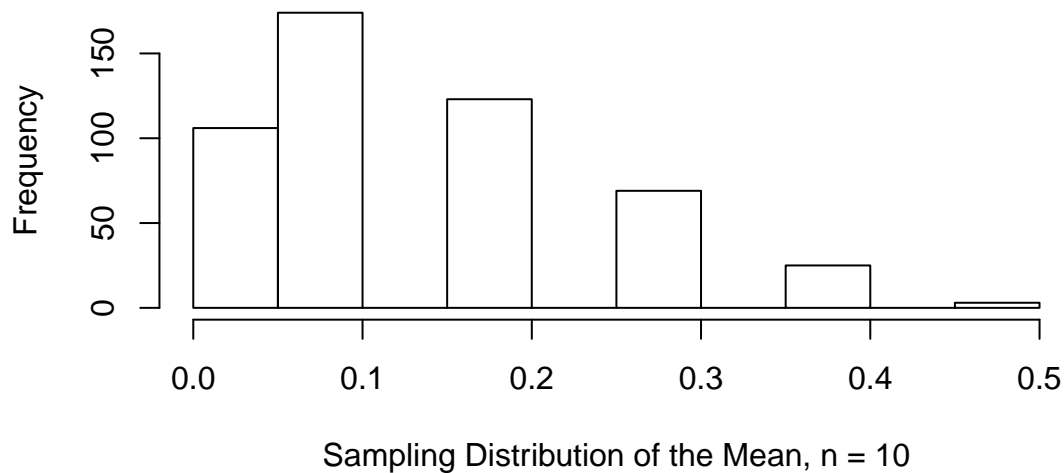
```
## [1] 0.14344
```

```
sd(vector_of_sample_means_50)
```

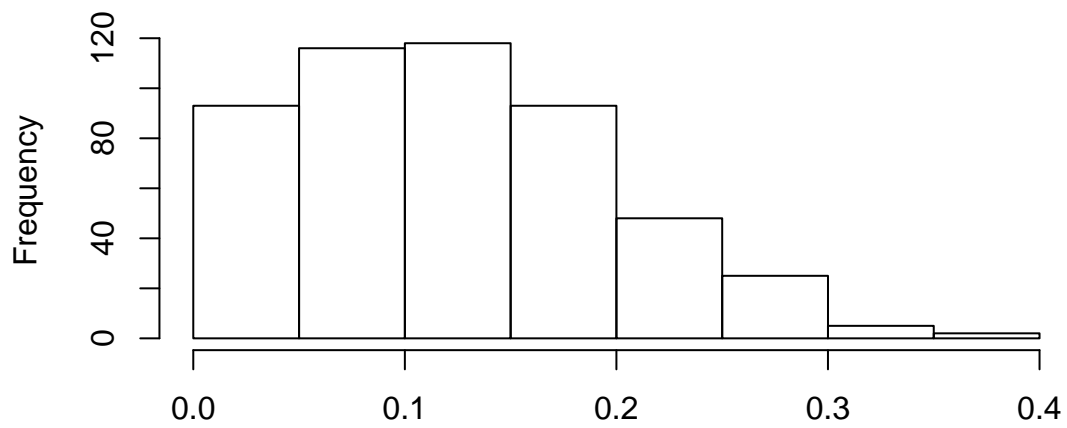
```
## [1] 0.04832429
```

d. Create histograms of the sampling distribution of the mean, for each sample size  $n$ .

```
hist(vector_of_sample_means_10, main = "", xlab = "Sampling Distribution of the Mean, n = 10")
```

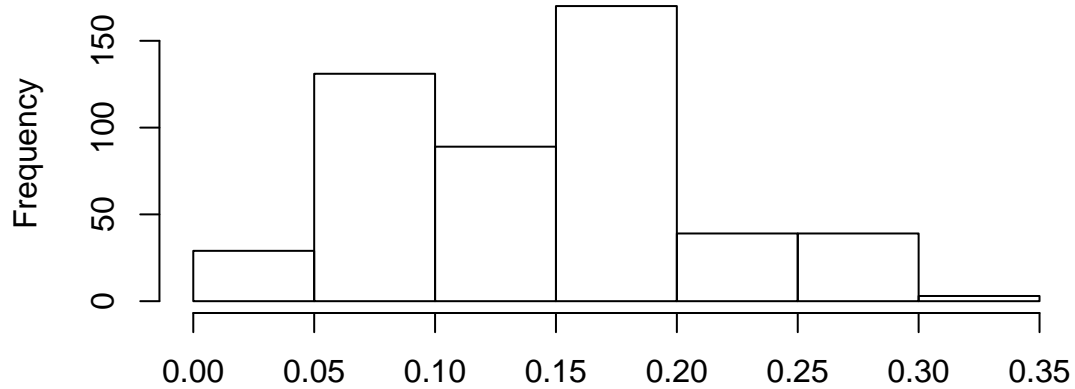


```
hist(vector_of_sample_means_20, main = "", xlab = "Sampling Distribution of the Mean, n = 20")
```



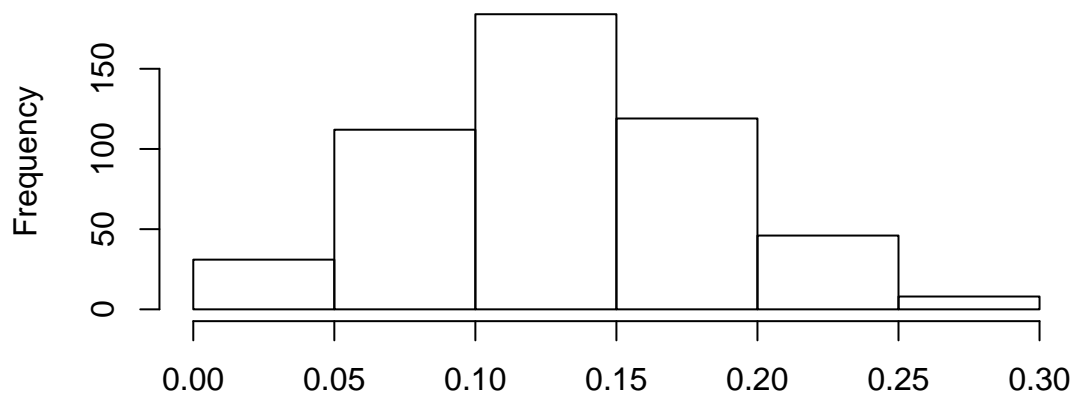
Sampling Distribution of the Mean,  $n = 20$

```
hist(vector_of_sample_means_30, main = "", xlab = "Sampling Distribution of the Mean, n = 30")
```



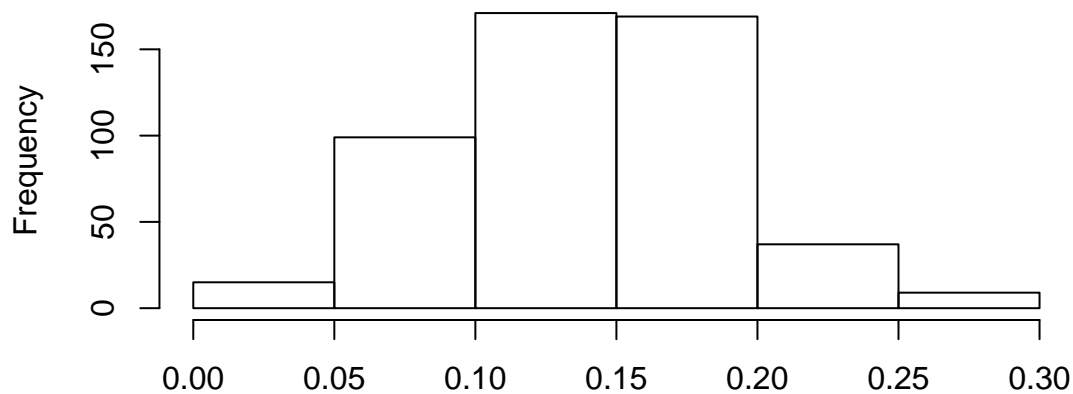
Sampling Distribution of the Mean,  $n = 30$

```
hist(vector_of_sample_means_40, main = "", xlab = "Sampling Distribution of the Mean, n = 40")
```



Sampling Distribution of the Mean,  $n = 40$

```
hist(vector_of_sample_means_50, main = "", xlab = "Sampling Distribution of the Mean, n = 50")
```



Sampling Distribution of the Mean,  $n = 50$

- e. The distribution of the means starts to look normal at  $n = 40$ , which makes sense given the  $\geq 30$  heuristic associated with the CLT.

## Exercise 4

For this problem I just copied the code from exercise 3, but with the `rcauchy()` function instead of `rbinom()`. Based on the means and standard deviations produced this way, it appears that sample size does not affect the distribution, and that it stays random as  $n$  increases.

```

# Set up the simulation, like in exercise 3.
set.seed(1017)
number_of_sims <- 500
# Create vectors for different sample sizes.
vector_of_sample_means_10 <- rep(-9, number_of_sims)
vector_of_sample_means_50 <- rep(-9, number_of_sims)
vector_of_sample_means_100 <- rep(-9, number_of_sims)
vector_of_sample_means_1000 <- rep(-9, number_of_sims)
# For loops to generate values and store in the vectors.
for (i in 1:number_of_sims) {
  vector_of_sample_means_10[i] <- mean(rcauchy(n = 10))
  vector_of_sample_means_50[i] <- mean(rcauchy(n = 50))
  vector_of_sample_means_100[i] <- mean(rcauchy(n = 100))
  vector_of_sample_means_1000[i] <- mean(rcauchy(n = 1000))
}
# Mean and sd of each sample group.
mean(vector_of_sample_means_10)

## [1] -10.66474
sd(vector_of_sample_means_10)

## [1] 227.3831
mean(vector_of_sample_means_50)

## [1] 0.07631917
sd(vector_of_sample_means_50)

## [1] 19.69005
mean(vector_of_sample_means_100)

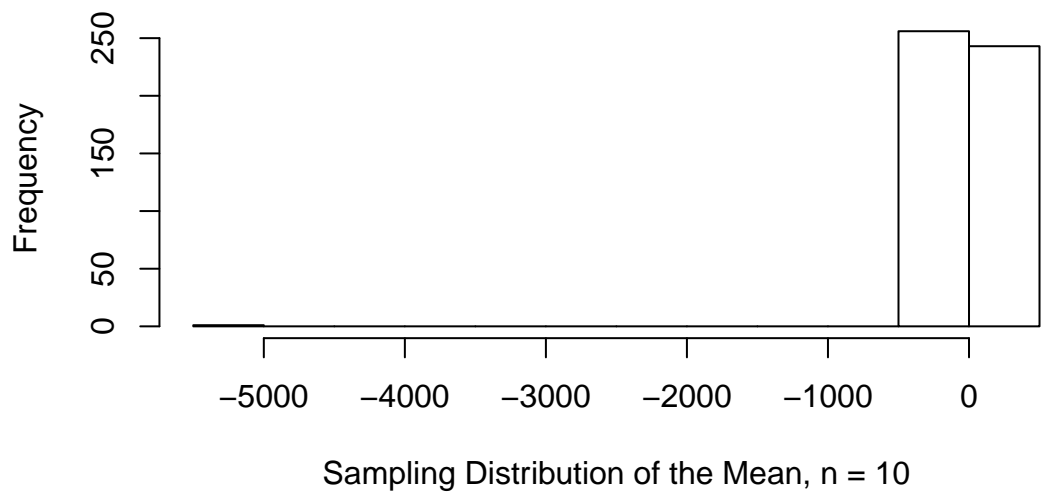
## [1] -85.69958
sd(vector_of_sample_means_100)

## [1] 1916.714
mean(vector_of_sample_means_1000)

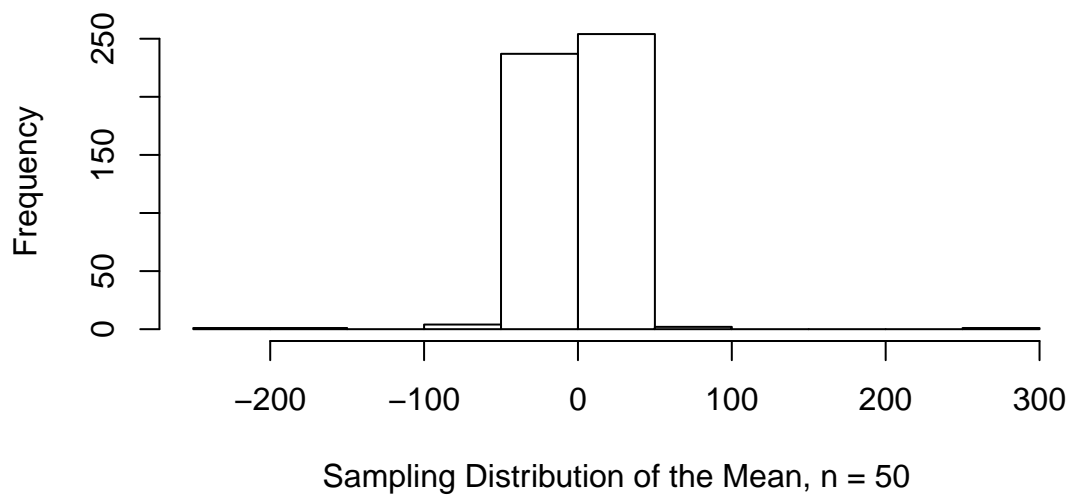
## [1] 0.5162534
sd(vector_of_sample_means_1000)

## [1] 9.971087
# Histograms of each sample group.
hist(vector_of_sample_means_10, main = "", xlab = "Sampling Distribution of the Mean, n = 10")

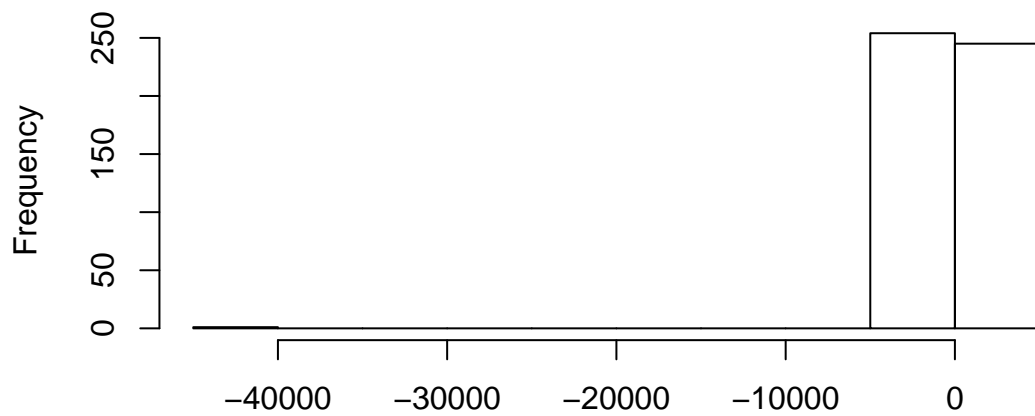
```



```
hist(vector_of_sample_means_50, main = "", xlab = "Sampling Distribution of the Mean, n = 50")
```

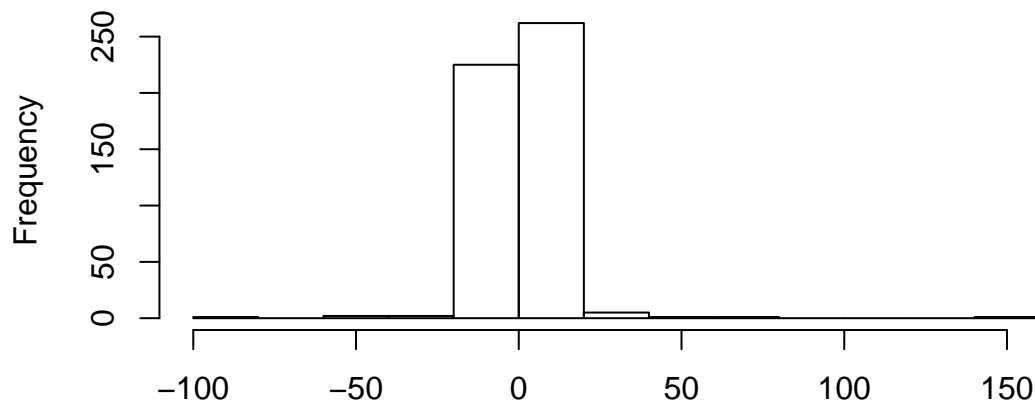


```
hist(vector_of_sample_means_100, main = "", xlab = "Sampling Distribution of the Mean, n = 100")
```



Sampling Distribution of the Mean, n = 100

```
hist(vector_of_sample_means_1000, main = "", xlab = "Sampling Distribution of the Mean, n = 1000")
```



Sampling Distribution of the Mean, n = 1000

## Exercise 6

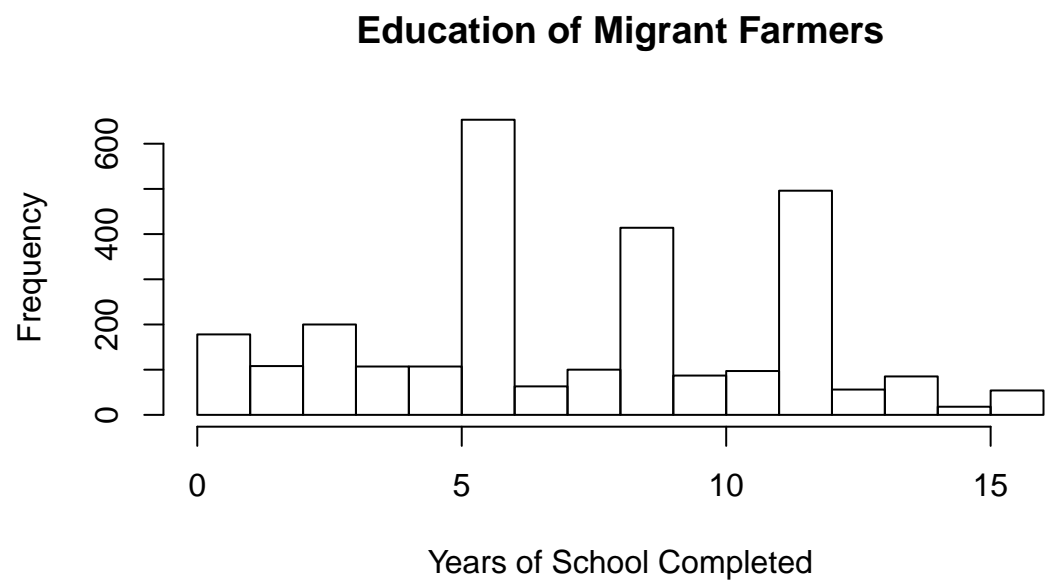
- Read NAWS2014.csv from the Canvas site into R with the name NAWS.

```
NAWS <- read.csv("C:\\Users\\timbv\\Documents\\School\\UC Denver\\Biostatistics\\Biostatistical Methods
```

- Plot a histogram of the A09 column, which asks how many years of school migrant farmers have completed.



```
hist(NAWS$A09, main = "Education of Migrant Farmers", xlab = "Years of School Completed")
```



Goodman Summary