HW9

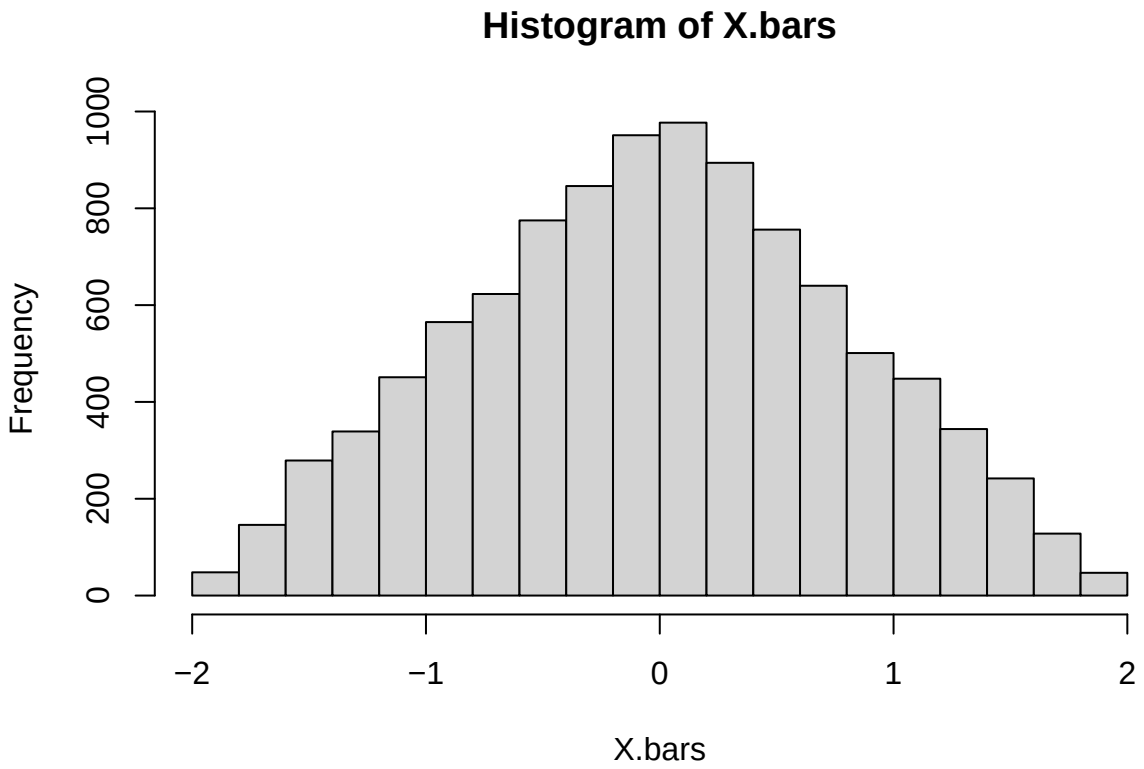
Miles Tweed

November 11, 2020

Problem 1

(a)

```
set.seed(123)
n <-2
X.bars <- vector()

for (i in 1:10000) {
  X.bars <- append(X.bars, mean(runif(n,-2,2)))

}

hist(X.bars)
```




```
print(paste('The mean of the sampling distribution is:',
            round(mean(X.bars),5)))
```

```
## [1] "The mean of the sampling distribution is: -0.01236"
```

```
print(paste('The standard deviation of the sampling distribution is:',
            round(sd(X.bars),3)))
```

```
## [1] "The standard deviation of the sampling distribution is: 0.81"
```

```
t.mu <- 1/2*(-2+2) #Theoretical mean

t.sd <- sqrt(1/12*(2+2)^2)

print(paste('Theoretical mean:',
            t.mu,'   Theoretical standard deviation:',round(t.sd,3)))
```

```
## [1] "Theoretical mean: 0    Theoretical standard deviation: 1.155"
```

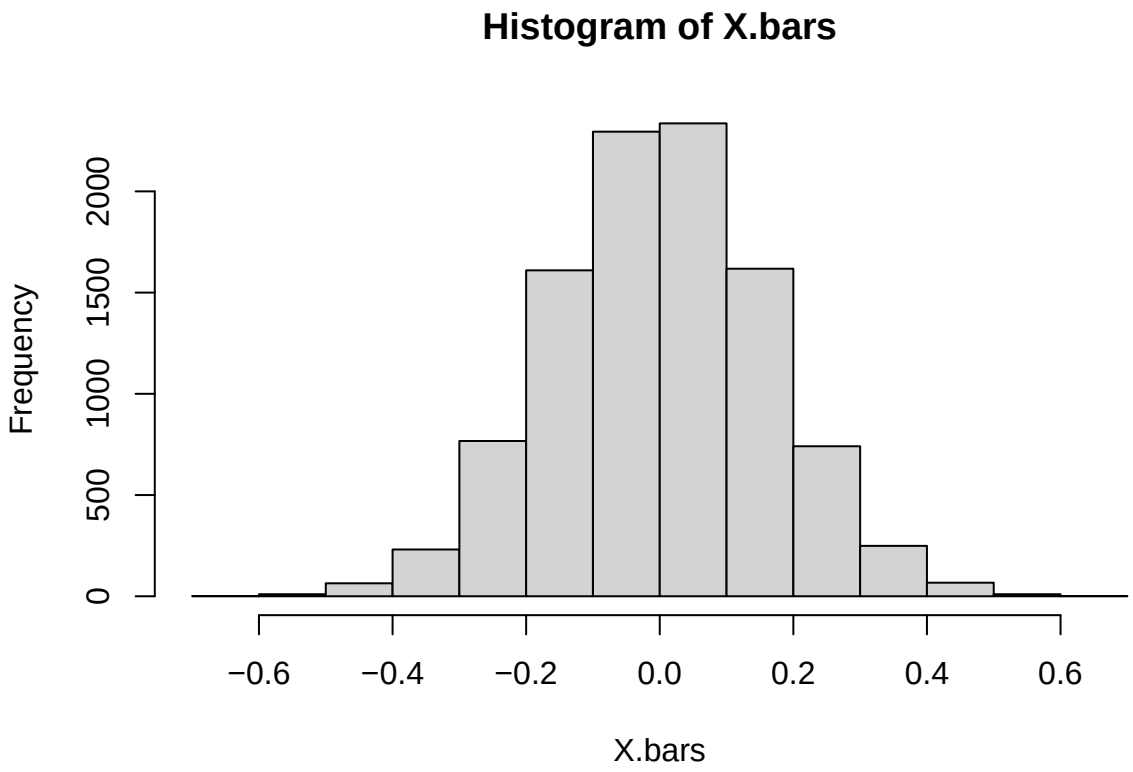
- This distribution is not quite bell-shaped and is more pyramidal instead.

- The mean is close to the theoretical value of 0 but is very slightly skewed toward the left tail at -0.012. However, I would consider the distribution to be unbiased since it is approximately center arount the mean of the population distribution.
- The standard deviation is less than the theoretical value of 1.155 at 0.81.

NOTE: The theoretical mean was found using the formula $\mu = \frac{1}{2}(a+b) = \frac{1}{2}(-2+2) = 0$ and the theoretical standard deviation was found with $\sigma = \sqrt{\frac{1}{12}(b-a)^2} = \sqrt{\frac{1}{12}(2+2)^2} = 1.155$.

(b)

```
set.seed(123)

n <-50

X.bars <- vector()

for (i in 1:10000) {
  X.bars <- append(X.bars, mean(runif(n,-2,2)))

}

hist(X.bars)
```




```
print(paste('The mean of the sampling distribution is:',
            round(mean(X.bars),5)))
```

```
## [1] "The mean of the sampling distribution is: -0.00027"
```

```
print(paste('The standard deviation of the sampling distribution is:',
            round(sd(X.bars),3)))
```

```
## [1] "The standard deviation of the sampling distribution is: 0.162"
```

- This distribution is much more bell-shaped than the previous.

- The mean is closer to the theoretical value and is unbiased.
- The standard deviation is also much smaller than both the previous sampling distribution and the theoretical one.

Problem 2

Part 1

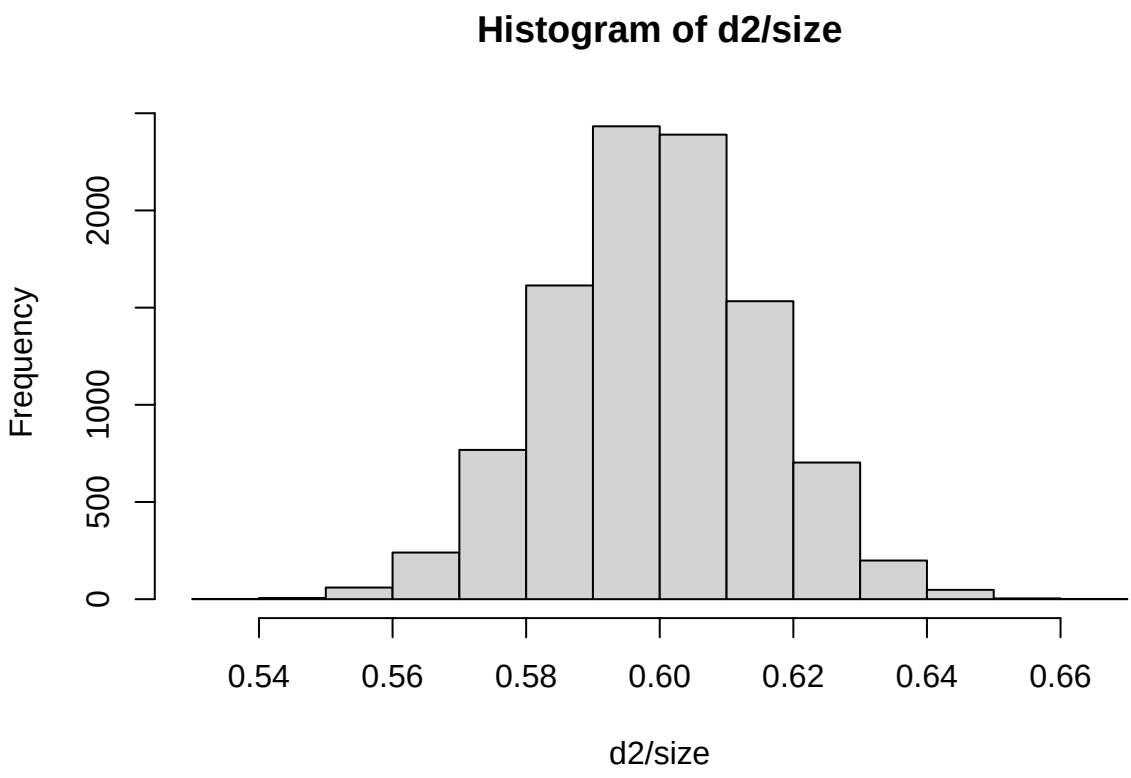
```
CI.calc <- function (n, s, cl = 95) {

              p <- s/n
              q <- 1-p
              sd <- sqrt((p*q)/n)
              if (cl==95){
                ci.low <- p-(1.96*sd)
                ci.high <- p+(1.96*sd)
              } else if (cl == 90) {
                ci.low <- p-(1.645*sd)
```

```
    ci.high <- p+(1.645*sd)
  }
  return(c(ci.low,ci.high))
}

CI.calc(1000,602)
```

```
## [1] 0.5716614 0.6323386
```

Part 2

```
set.seed(1)
n.sim <- 10000
prob <- 0.6
size <- 1000

# Placeholders for left (first column) and right (second column) ends
# of our CIs.
ci.95 <- matrix(0, nrow=n.sim, ncol=2)
ci.90 <- matrix(0, nrow=n.sim, ncol=2)

# Here you will need to
# 1) generate the 10,000 values from Bin(1000,0.6),

d2 = rbinom(n=n.sim, size = size, p=prob)
hist(d2/size)
```




```
# 2) loop through those and feed them as input to your confidence level function from part 1
#  (for cases of 95% and 90%)
```

```r
for (i in 1:length(d2)){
  ci.95[i,] <- CI.calc(1000,d2[i],95)
  ci.90[i,] <- CI.calc(1000,d2[i],90)
}


# That's an example of how you calculate the % of times your confidence interval
# contains the true parameter
mean(ci.95[,1] < prob & ci.95[,2] > prob)
```

```
## [1] 0.9438
```

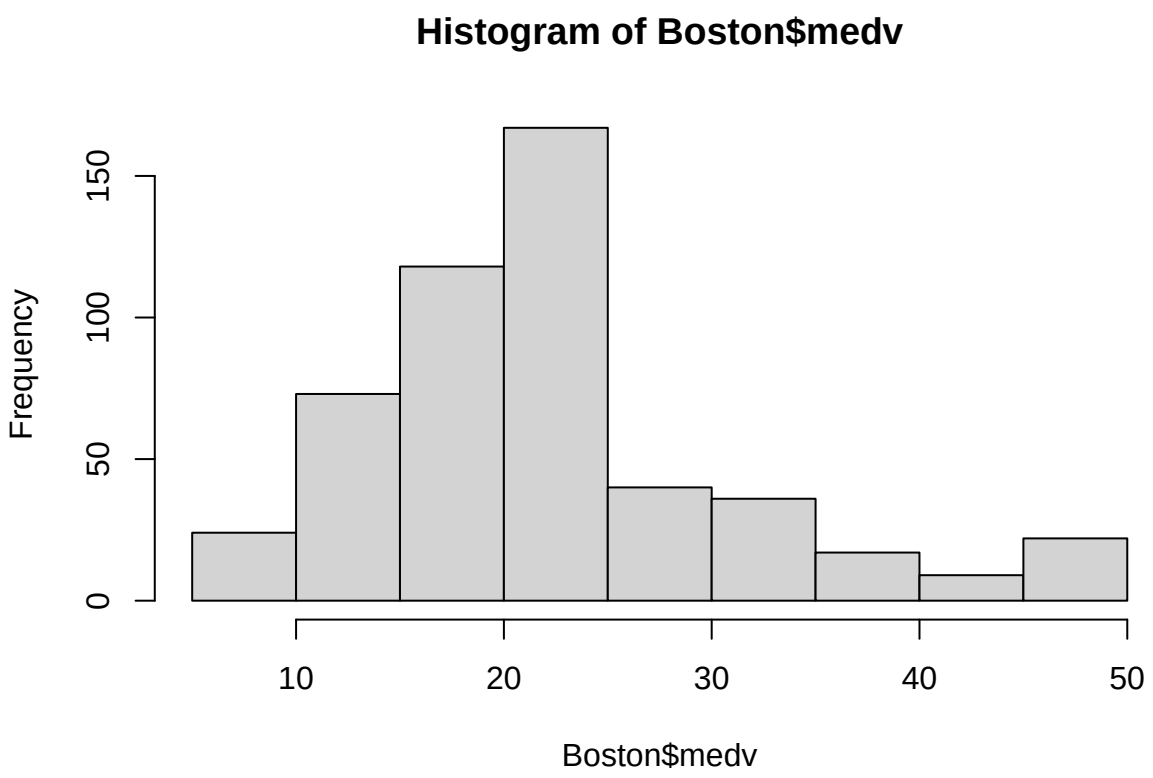
```r
#...

mean(ci.90[,1] < prob & ci.90[,2] > prob)
```

```
## [1] 0.8943
```

Problem 3

(a)

```r
library(MASS)

#Boston

hist(Boston$medv)
```




```r
mu.hat <- mean(Boston$medv)
print(paste('Estimate of the population mean:',round(mu.hat,2)))
```

```
## [1] "Estimate of the population mean: 22.53"
```

(b)

```
sigma <- sd(Boston$medv)
se <- sigma/sqrt(length(Boston$medv))
print(paste("Estimate of standard error using CLT",round(se,3)))
```

```
## [1] "Estimate of standard error using CLT 0.409"
```

```
library(boot)
n <- nrow(Boston)

x.se <- function(x,i) { sd(x[i,'medv'])/sqrt(length(x[i,'medv'])) }

res.se <- boot(data = Boston,
            statistic = x.se,
            R = 10000)

print(paste('Average value of standard error using bootstrap:',round(res.se$t0,3)))
```

```
## [1] "Average value of standard error using bootstrap: 0.409"
```

The two values are identical.

(c)

```
x.median <- function(x,i) { median(x[i,'medv']) }

res.med <- boot(data = Boston,
            statistic = x.median,
            R = 10000)

print(paste('Median value using bootstrap:',res.med$t0))
```

```
## [1] "Median value using bootstrap: 21.2"
```

(d)

```
meds <- res.med$t

x.se2 <- function(x,i) { sd(x[i,1])/sqrt(length(x[i,1])) }

res.med.se <- boot(data = Boston,
            statistic = x.se2,
            R = 10000)

print(paste('Standard error of median using bootstrap:',round(res.med.se$t0,3)))
```

```
## [1] "Standard error of median using bootstrap: 0.382"
```

Problem 4

7.7

(a)

```
n <- 500
p <- 0.3
```

```
sd <- sqrt((p*(1-p))/n)

print(paste("np=", n*p ,'   n(1-p)=',n*(1-p), '   sd:',round(sd,3)))
```

```
## [1] "np= 150    n(1-p)= 350    sd: 0.02"
```

The sampling distribution is approximately normal shaped because n is large enough that both np and n(1-p) are greater than 15. The mean is the same as the population proportion 0.300 and the standard deviation is 0.02.

(b) These batting averages are only one standard deviation from the mean so they would not be unusual.

7.14

(a)

```
p = 0.55
n = 200

sd <- sqrt((p*(1-p))/n)



print(paste("np=", n*p ,'   n(1-p)=',n*(1-p), '   sd:',round(sd,3)))
```

```
## [1] "np= 110    n(1-p)= 90    sd: 0.035"
```

The sampling distribution will be shaped normally so the mean would be 0.55 and the standard deviation would be 0.035.

(b) Yes it is reasonable to assume a normal shape because both np and n(1-p) are greater than 15.

(c)

```
z.score <- (0.50 - 0.55)/0.035
print(paste('The z-score associated with 50% for this sampling distribution is:',round(z.score,2)))
```

```
## [1] "The z-score associated with 50% for this sampling distribution is: -1.43"
```

This is within two standard deviation from the mean $0.55 \pm 2(0.035)$. Using the z score associated with this outcome, the probability of receiving less than 50% of the votes is 0.0764 or 7.64% which is not very likely.

(d)

```
p = 0.55
n = 1000

sd <- sqrt((p*(1-p))/n)

z.score <- (0.50 - 0.55)/sd

print(paste("np=", n*p ,'   n(1-p)=',n*(1-p), '   sd:',round(sd,3), '    z-score:', round(z.score,2)))
```

```
## [1] "np= 550    n(1-p)= 450    sd: 0.016     z-score: -3.18"
```

With a sample size n=1000 the standard deviation is much smaller and the probability of not winning the majority becomes much smaller. The probability is now 0.0007 or 0.07% which is extremely unlikely.

7.15

(a) Using the web application, I got a sample mean of 68.6. With a single small sample it is unreasonable to think that the sample will be normally distributed so the sample mean will not necessarily equal the population mean.

(b) The simulated sampling distribution is approximately normal in shape and has a sample mean of 70 which is equal to the population mean.

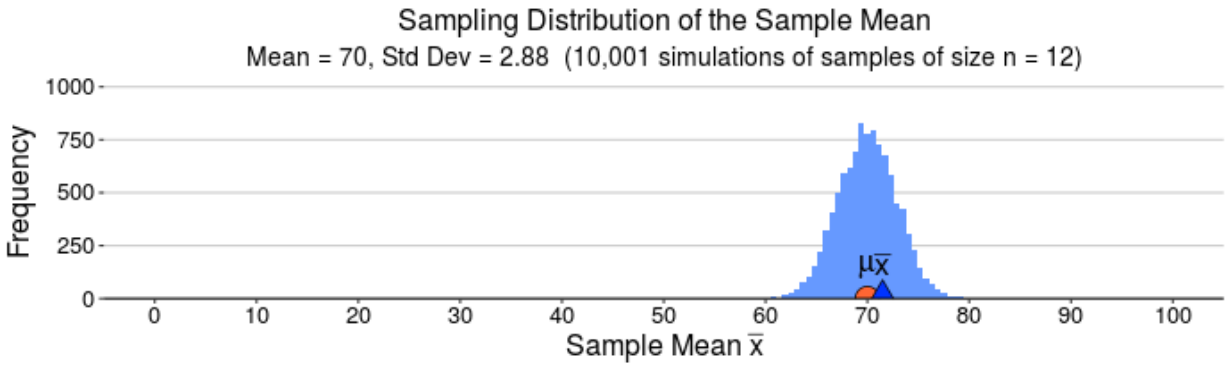


Figure 1: Sampling Distribution

(c) The sampling distribution becomes more narrow.

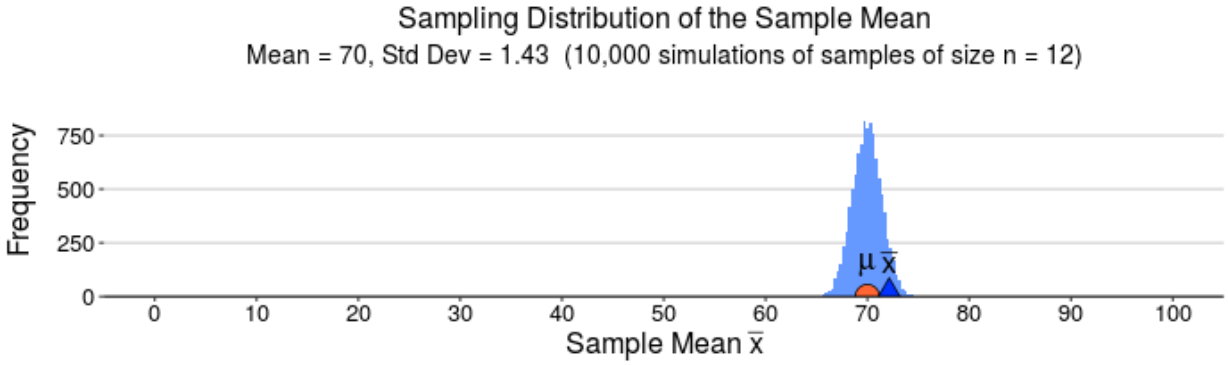


Figure 2: Sampling Distribution

7.20

(a)

```
mu <- 0.1 # probability distribution mean
sig <- 100 # probability distribution sd
n <- 1e6 # number of plays

sd <- sig/sqrt(n)
print(paste('Mean:',mu,'  Standard deviation', sd))
```

```
## [1] "Mean: 0.1   Standard deviation 0.1"
```

The mean of the sampling distribution will be the center around and the same as the probability distribution (0.1). The standard deviation of the sampling distribution is $\frac{\sigma}{\sqrt{n}}$

(b) Since the sampling distribution is normally distributed the probability of winning a dollar or more is:

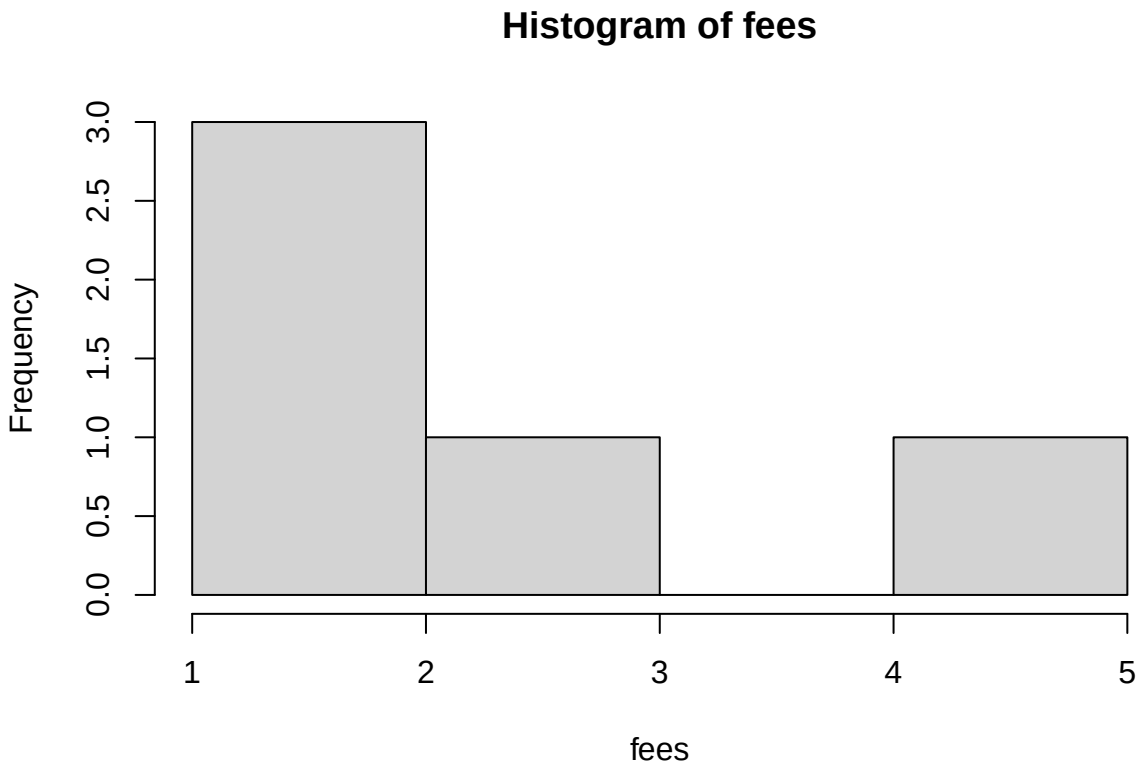
```
pnorm(1, mean=0.1, sd = 0.1, lower.tail = F)
```

```
## [1] 1.128588e-19
```

8.6

(a) Since only one small sample was taken it would likely not be normally distributed or centered around the population mean. Because of this it could not be used to find a point estimate of the mean fee paid on the platform. However, the best guess for the population mean would be the sample mean.

```
fees <- c(1.09, 4.99, 1.99, 1.99, 2.99)

x.bar <- mean(fees)
hist(fees)
```




```
print(paste('The point estimate of the mean is the mean of the sample distribution',x.bar))
```

```
## [1] "The point estimate of the mean is the mean of the sample distribution 2.61"
```

(b) This indicates how accurate the point estimate for the population mean. In other words, it says that we can be 95% confident that the actual population mean (μ) will fall within the range of $\bar{x} \pm 1.85$. This also indicates that the sample distribution's standard deviation is $s = 1.85/1.96 \approx 0.94$

8.13

(a)

```
n <- 3900
q <- 24/n
p <- 1-q

p.hat <- q
print(paste('The point estimate is:',round(p.hat,4)))
```

```
## [1] "The point estimate is: 0.0062"
```

$\hat{p} = 0.0062$

(b)

```
se <- sqrt((p*q)/n)
print(paste('The standard error is:',round(se,4)))
```

```
## [1] "The standard error is: 0.0013"
```

(c)

```
me <- 1.96*se
print(paste('The margin of error is:',round(me,4)))
```

```
## [1] "The margin of error is: 0.0025"
```

(d)

```
ci.95 <- c(round(p.hat-me,4),round(p.hat+me,4))
print(paste('The 95% confidence interval is:',ci.95))
```

```
## [1] "The 95% confidence interval is: 0.0037"
## [2] "The 95% confidence interval is: 0.0086"
```

In the long-run, the correct population proportion would be contained within the confidence interval that was generated using this method for random samples 95% of the time.

(e) Yes, we can be 95% confident since all of the values within the confidence interval are below 0.01.

8.16

(a)

```
sample.p <- 1183/1824

print(paste('Sample p:',sample.p))
```

```
## [1] "Sample p: 0.648574561403509"
```

(b) The confidence interval suggests that we can be 95% confident that the true population proportion of people that are in favor of the death penalty is between 0.63 and 0.67.

(c) In the long-run, if we sampled the population many times and constructed 95% confidence intervals for the samples, the population proportion would be contained in that interval 95% of the time.

(d) Yes, because every values in the confidence interval is over 0.5.

8.25

(a)

```
n <- 1400
n.d <- 660
n.r <- 740
p.d <- n.d/n
p.r <- n.r/n
se <- sqrt((p.d*p.r)/n)
me <- 1.96*se
ci.95.low <- round(p.d-me,3)
ci.95.high <- round(p.d+me,3)
```

```
print(paste('The 95% confidence interval for Democratic votes is:',
            ci.95.low, 'to',ci.95.high))
```

```
## [1] "The 95% confidence interval for Democratic votes is: 0.445 to 0.498"
```

Yes, with 95% confidence I would conclude that the Republican candidate would win.

(b)

```
me <- 2.58*se
ci.99.low <- round(p.d-me,3)
ci.99.high <- round(p.d+me,3)
print(paste('The 99% confidence interval for Democratic votes is:',
            ci.99.low, 'to',ci.99.high))
```

```
## [1] "The 99% confidence interval for Democratic votes is: 0.437 to 0.506"
```

No, I could not make a conclusion about a winner with 99% confidence. I would need a larger sample size in order to reduce the standard error and shrink the margin of error.

8.26

(a)

```
n <- 140
n.d <- 66
n.r <- 74
p.d <- n.d/n
p.r <- n.r/n
se <- sqrt((p.d*p.r)/n)
me <- 1.96*se
ci.95.low <- round(p.d-me,3)
ci.95.high <- round(p.d+me,3)
print(paste('The 95% confidence interval for Democratic votes is:',
            ci.95.low, 'to',ci.95.high))
```

```
## [1] "The 95% confidence interval for Democratic votes is: 0.389 to 0.554"
```

No, the confidence interval spans the region around 50% of the votes and a conclusion could not be made.

(b) With the smaller sample size the standard error becomes larger (since the sample size is the term in the denominator). A larger standard error leads to a larger margin of error which provides less information because the range of possible values for the population proportion is greater.

8.29

(a) The point estimate of the population mean is 2.56.

(b)

```
n <- 590
se <- 0.84/sqrt(n)
print(paste('Standard Error:',round(se,3)))
```

```
## [1] "Standard Error: 0.035"
```

(c) We can be 95% confident that the mean value for the response to the question "how many children for a family is ideal" is within the range of 2.49 to 2.62 for the total population of women.

(d) No, because the value of two falls outside of the confidence interval.

8.30

(a) The point estimate for the population mean is 2.51.

```
se <- 0.87 / sqrt(530)
print(paste('The standard error is:',round(se,3)))
```

```
## [1] "The standard error is: 0.038"
```

(b) in the long-run if we were to sample the population and calculate the confidence interval, we would expect that interval to contain the true population mean 95% of the time. So we could be 95% confident that the population mean was between 2.43 and 2.59.

(c) Considering the similarity of the confidence intervals, there is not much difference between males and females regarding opinions about the ideal number of children in a family. The point estimates of the mean differ by only 0.05 and since this is a measure of children and only integer values of children are sensible (i.e. you cannot have a fraction of a child) the values would have to differ by at least 1.0 for there to be a real difference in their opinions.

8.37

(a)

```
resp <- c(0,0,0,0,1,1,1,2,2,6,6,7,7,10)
n <- length(resp)
x.bar <- mean(resp)
s <- sd(resp)
se <- s/sqrt(n)
print(paste('Sample mean:',round(x.bar,3),
            '  standard deviation', round(s,3),
            '  standard error:', round(se,3)))
```

```
## [1] "Sample mean: 3.071   standard deviation 3.385   standard error: 0.905"
```

(b) The degrees of freedom for this sample is $(14 - 1) = 13$ and the t-value required to generate a 90% confidence interval at this number of degrees of freedom is $T_{.050} = 1.771$

```
me <- 1.771*se
print(paste("90% confidence interval:",round(x.bar-me,3),'to',round(x.bar+me,3)))
```

```
## [1] "90% confidence interval: 1.469 to 4.673"
```

We are 90% confident that the mean number of hours per week that females over 80 spend checking and responding to email is between 1.47 and 4.67.

(c) Looking at the sample the modal value is 0 with the next most common being one which suggests that this may be the most common response. Also, the standard deviation suggests that a majority of the of the responses will be between 0 and 6.45 since this range represents one standard deviation from the mean. The t distribution is robust enough to account for this though since it takes into account the degrees of freedom so the interval can be considered valid.