```r
library(mosaic)
```

```
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2

##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.

##
## Attaching package: 'mosaic'

## The following objects are masked from 'package:dplyr':
##
##     count, do, tally

## The following object is masked from 'package:Matrix':
##
##     mean

## The following object is masked from 'package:ggplot2':
##
##     stat

## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var

## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

Question 1 a

```r
#Question 1
#a

mean11 = 5.0 #mean given in Q11
sd11=1.5 #standard deviation given in Q11
n1 = 12 #sample size
mymean = 5.6875 #sample mean I observed in Question 11 of Assignment 1

samplemean1 = mean11
samplesd1 = sd11/sqrt(n1)

#P(X >= 5.6875) = 1 - P(X<5.6875)
probq1a = 1 - pnorm(mymean, samplemean1, samplesd1)
#the probability that another random sample of the same size will produce a sample mean
#that is at least the same value as the value of sample mean I observed in Question 11 of Assignment 1
probq1a
```

```
## [1] 0.0561756
```

Question 1 b

```
#Question 1
#b

#Calculating the probability of sample standard deviation
#fall between 0.5 to 1 by using Tansformation

mysd = 1.580369 #smaple standard devition I observed In A1
sd11=1.5 #standard deviation given in Q11

#P(0.5 <= S <= 1)

lhs = (n1-1)*0.5^2/sd11^2
#lhs
rhs = (n1-1)*1^2/sd11^2
#rhs
df1 = n1-1
probq1b = pchisq(rhs,df1) - pchisq(lhs,df1)
#the probability that another random sample will yield a sample standard deviation
#that is between 0.5 hour and 1 hour is 0.06343368
probq1b
```

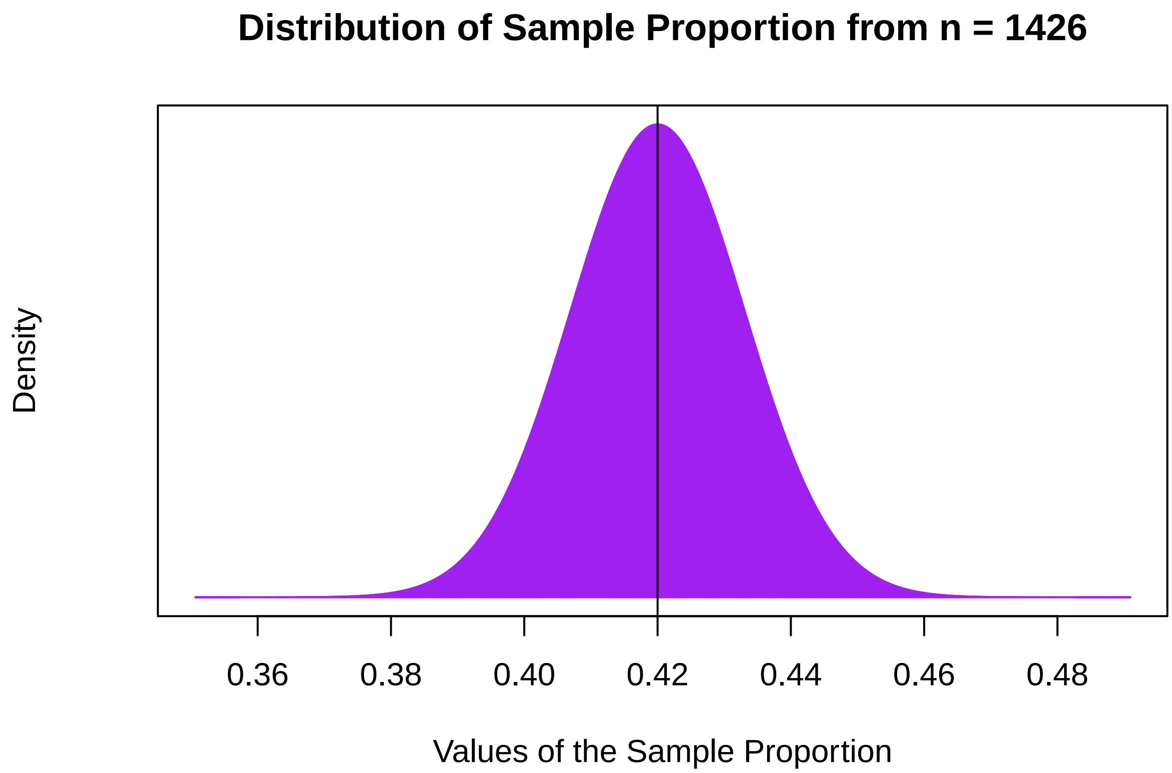
```
## [1] 0.06343368
```

Question 2 a The shape of this distribution is a normal distribution with mean of 0.42, and standard deviation of 0.0130701. A balancing point is the mean which is 0.42. A measure of spread is the standard deviation which is 0.0130701. The bell curve is steep due to the small standard deviation.

```
#Question 2
#a
#By CLT, np
sizeq2 = 1426
pq2 = 0.42
#sizeq2 * pq2 = 598.92
#sizeq2 * (1- pq2) = 827.08

#By CLT, np > 10, n(1-p)>10, phat is approximatelty normal


meanphatq2 = pq2 #mean
sdphatq2 = sqrt(pq2*(1-pq2)/sizeq2) #standard deviation
x = seq(500,700 , 0.1)
phat = (x/sizeq2)
plot(phat, dnorm(phat, pq2, sdphatq2), yaxt = 'n', xlab="Values of the Sample Proportion", ylab = "Dens
```




```
## integer(0)
```

Question 2 b

```r
#Question 2
#b
#P(phat <= 0.3794)
pnorm(0.3794, mean = meanphatq2, sd = sdphatq2)
```

```
## [1] 0.000947136
```

```r
#the probability of an new random sample of n =1426 produce a sample proportion that is at most as 0.37
```

Question 2 c

```r
#Question 2
#c
Nsim = 1000
n.sample = 1426
p.hats = numeric(Nsim)

for(i in 1:Nsim){
  outcome = numeric(n.sample)
  for(j in 1:n.sample){
```

```
    outcome[j] = rbinom(1,1,meanphatq2)
  }
  p.hats[i] = sum(outcome)/(n.sample)
}
sim.df = data.frame(p.hats)
head(sim.df,5)
```

```
##      p.hats
## 1 0.3934081
## 2 0.4165498
## 3 0.4263675
## 4 0.4249649
## 5 0.4095372
```

```
favstats(p.hats,data = sim.df )
```

```
##        min       Q1    median        Q3       max      mean         sd    n
##  0.3793829 0.411641 0.4200561 0.4291725 0.4635344 0.4199334 0.01298099 1000
##  missing
##        0
```

```
#mymeanq2 = mean(p.hats, data = sim.df)
#mysdq2 = sd(p.hats, data = sim.df)

#p.hats <= 0.3794

filter(sim.df, p.hats <= 0.3794)
```

```
##      p.hats
## 1 0.3793829
```

```
proportionq2c = nrow(filter(sim.df, p.hats <= 0.3794))/n.sample
proportionq2c
```

```
## [1] 0.0007012623
```

```
#the proportion of my p hat that are less than or equal to 0.3794 is 0.001402525
```

Question 3

```
#Question 3



pmfq3 <- function(numofmatch){
  prob = (choose(6,numofmatch)*choose(43,6 - numofmatch))/choose(49,6)
  return(prob)
}
pmfq3(0:5)
```

```
## [1] 0.4359649755 0.4130194505 0.1323780290 0.0176504039 0.0009686197
## [6] 0.0000184499
```

```
meanq3 = 0.7347
sdq3 = 0.76
n = 52 #sample size
samplemean = 0.7347
samplesd = sdq3/sqrt(n)
samplesd
```

```
## [1] 0.105393
```

```
#probability that Billy have at least one matching number on average for a sample size of 52
#Xbar is approximately normal
#P(Xbar >= 1) = 1 - P(Xbar < 0)

1 - pnorm(0,samplemean, samplesd)
```

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

```
#Billy's claim is true.
#The probability that Billy have at least one matching number on average in 52 weeks(one year) is 1.
#That means, on average, he will have at least one matching number in one year's plays.
```

Question 4 a

```
#Question 4
#a
dataq4 = c(16,5,21,19,10,5,8,2,7,2,4,9)

ntimes = 2000
nsize = 12
lc50means = numeric(ntimes)

for(i in 1:ntimes){
  datalc50 = sample(dataq4, nsize, replace = TRUE)
  lc50means[i] = mean(datalc50)
}

LC50boot = data.frame(lc50means)
head(LC50boot,10)
```

```
##    lc50means
## 1   8.666667
## 2  10.916667
## 3  10.500000
## 4   8.750000
## 5   8.250000
## 6   6.833333
## 7   7.666667
## 8   8.416667
## 9   6.416667
## 10  9.000000
```

```
favstats(~lc50means, data=LC50boot)
```

```
##       min   Q1   median    Q3      max     mean       sd    n missing
##  3.916667 7.75 8.916667 10.25 15.41667 9.009167 1.784533 2000       0
```

Question 4 b By finding the 95% bootstrap confidence interval for

$$\mu$$

I got the 2.5th-percentile and the 97.5th-percentile of X_bar. That are the lower bound and the upper bound. Since about 95% of the values of X_bar(mean) fall between 5.666667 and 12.666667
I can conclude that there is a 95% level of confidence that the unknown value of the population mean will be some point between the lower bound and the upper bound. In the given scenario, it means that DDT has a 95% level of confidence to say that the mean of LC50 is in between 5.666667 and 12.666667

```
#Question 4
#b
qdata(~lc50means,c(0.025,0.975), data = LC50boot)
```

```
##      2.5%     97.5%
##  5.666667 12.583333
```

Question 4 c

```
#Question 4
#c
lc50 = data.frame(dataq4 = c(16,5,21,19,10,5,8,2,7,2,4,9))
lc50
```

```
##    dataq4
## 1      16
## 2       5
## 3      21
## 4      19
## 5      10
## 6       5
## 7       8
## 8       2
## 9       7
## 10      2
## 11      4
## 12      9
```

```
t.test(~dataq4, data=lc50)$conf
```

```
## [1]  4.91814 13.08186
## attr(,"conf.level")
## [1] 0.95
```

```
#the t-confidence interval of 95% is [4.91814 13.08186].
```

Question 4 d

If I were to report one of these confidence intervals, I would report the 95% bootstrap confidence interval from part b. Comparing two result, the t-version of confidence interval gives a wider interval than the bootstrap confidence interval. In this case, the sample size is too small. I think it is better to do a bootstrap in order to have a more precise statistical result which is more convincing. And the sample mean X_bar from bootstrap is an unbiased statistic for the average of LC50.

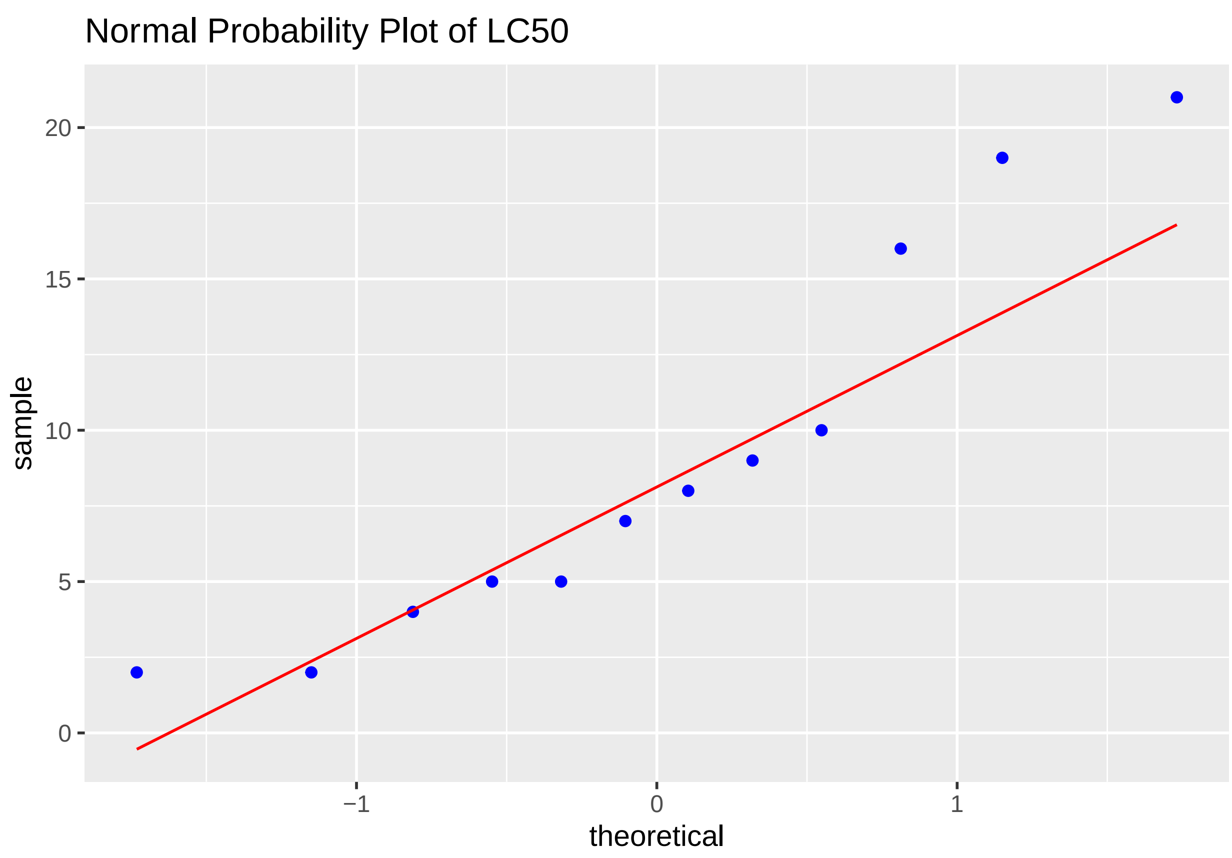
Question 4 e

From lectures we know that the t-confidence interval estimate for the population mean when the data is small, which n <= 25, the t-confidence interval estimate is valid on the condition that the data from a population of values that is modeled by the Normal distribution. We use Normal Probability Plot to check the condition. If the Normal Probability Plot produces roughly a straight line throught the middle of the points, then the data can be determined to conform to a Normal probability model.

We can see that the plot is roughly a straight line throught the middle of the points. Thus, I conclude that the condition, which the data from a population of values that is modeled by the Normal distribution, is satisfied.

```
#Question 4
#e

ggplot(data=lc50, aes(sample=dataq4)) + stat_qq(col='blue') + stat_qqline(col='red') + ggtitle("Normal P
```



Question 5 a

```r
#Question 5
#a

nq5 = 1866
nyes = 571
#Compute a 95% confidence interval for p
binom.test(nyes, nq5, ci.method="Plus4")$conf
```

```
## [1] 0.2855226 0.3273117
## attr(,"conf.level")
## [1] 0.95
## attr(,"method")
## [1] "plus4"
```

```r
#From this sample of n=1866 Canadians homeowners aged 55 or older,
#the proportion of this population that has either downsized or plan to downsize
#is somewhere between 0.2855226  and 0.3273117, with 95% confident.
```

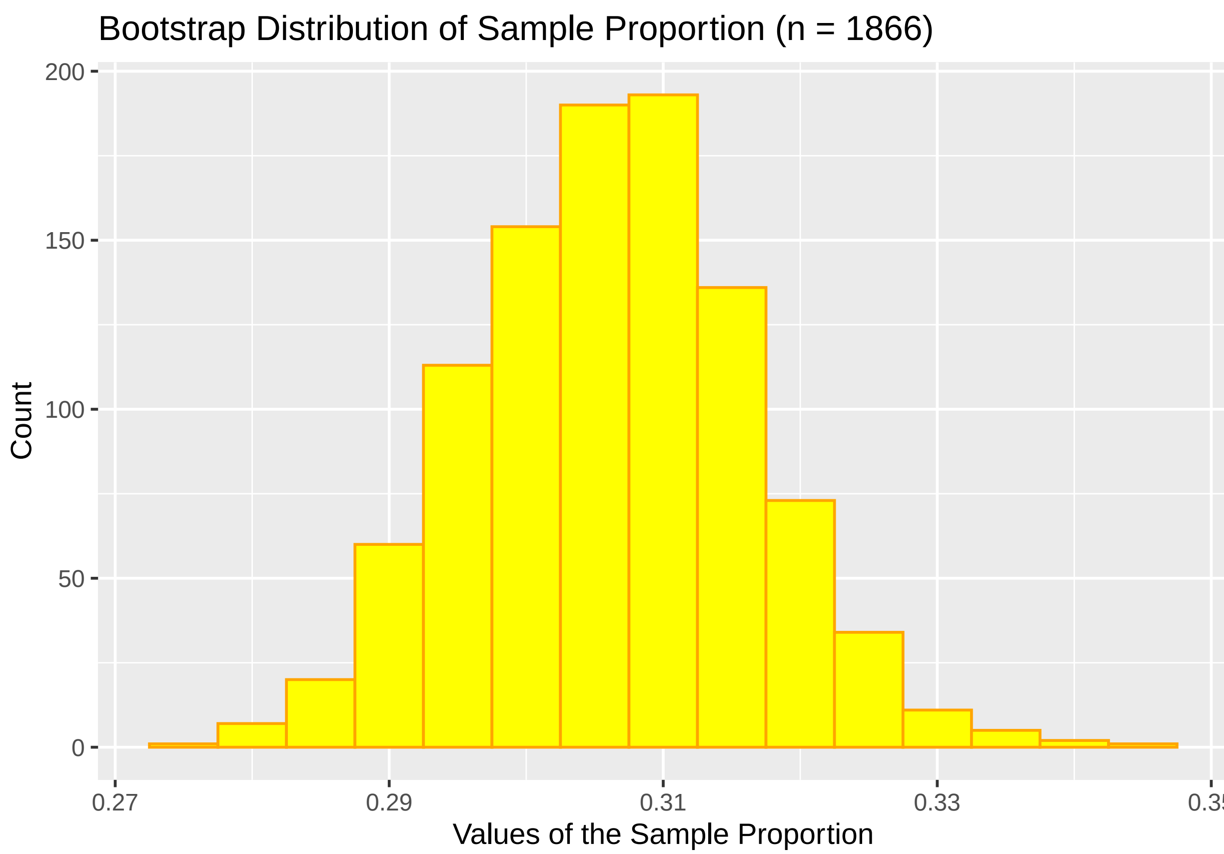
Question 5 b

```r
#Question 5
#b

pool = c(rep(0, 1866-571), rep(1, 571))
phats1866 <- numeric(1000)
for(i in 1:1000){
     temp.data <- resample(pool)
     phats1866[i] <- mean(temp.data)
}
boot_phat1866.df <- data.frame(phats1866)
head(boot_phat1866.df, 4)
```

```
##   phats1866
## 1 0.3060021
## 2 0.3161844
## 3 0.3242229
## 4 0.3285102
```

```r
ggplot(boot_phat1866.df, aes(x=phats1866)) + geom_histogram(col="orange", fill="yellow", binwidth=0.005
```




```
favstats(~phats1866, data=boot_phat1866.df)
```

```
##        min        Q1    median        Q3       max      mean          sd    n
##  0.2759914 0.2995713 0.3060021 0.3129689 0.3445874 0.3062524 0.01010674 1000
##  missing
##        0
```

Question 5 c

```
#Question 5
#c
qdata(~phats1866, c(0.025,0.975), data=boot_phat1866.df)
```

```
##      2.5%     97.5%
## 0.2867095 0.3263800
```

```
#95% bootstrap confidence interval for p is somewhere between 0.2856377  and 0.3285236
```

Question 5 d

Comparing two result, they are somewhat similar. The binom Plus4 method gives a 95% confidence interval of [0.2855226 0.3273117]. The 95% bootstrap confidence interval gives a interval of [0.2851018 0.3269025]. I will report the bootstrap confidence interval due to a narrower interval it gives.

I can be 95% confident that, from this sample of n=1866 Canadians homeowners aged 55 or older, the proportion of this population that has either downsized or plan to downsize is somewhere between 0.2856377 and 0.3285236

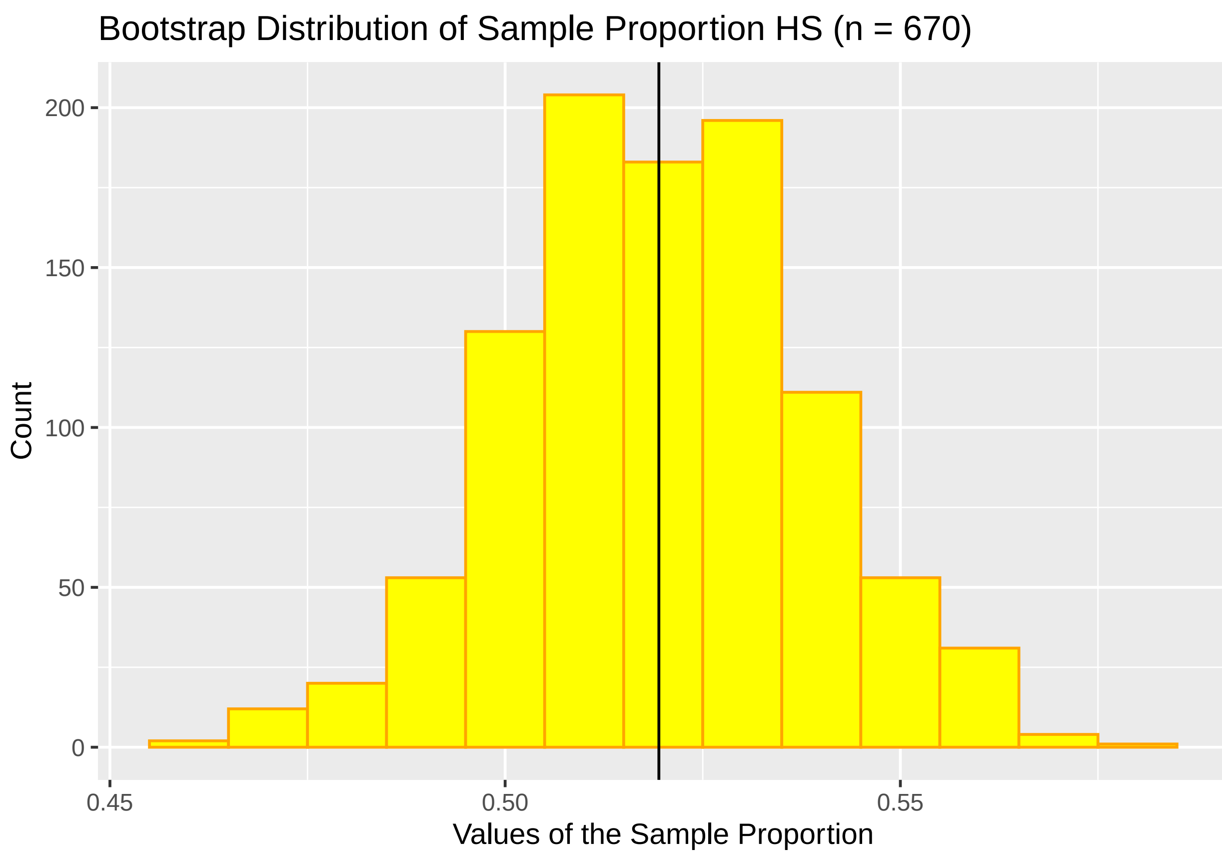
Question 6 a

```
#Question 6
#a

#0 meas agreed, 1 means disagreed
data.surveyhs = c(rep(0, 670-348), rep(1, 348 ))
phatshs670 <- numeric(1000)
for(i in 1:1000){
     temp.data <- resample(data.surveyhs)
     phatshs670[i] <- mean(temp.data)
}
boot_phaths670.df <- data.frame(phatshs670)
head(boot_phaths670.df, 4)
```

```
##   phatshs670
## 1  0.5611940
## 2  0.5447761
## 3  0.5179104
## 4  0.5268657
```

```
mean.hsdis <- favstats(~phatshs670, data=boot_phaths670.df)$mean
ggplot(boot_phaths670.df, aes(x=phatshs670)) + geom_histogram(col="orange", fill="yellow", binwidth=0.0
```




```
favstats(~phatshs670, data=boot_phaths670.df)
```

```
##       min        Q1    median        Q3       max      mean         sd    n
##  0.461194 0.5074627 0.519403 0.5313433 0.5776119 0.5194507 0.01905579 1000
##  missing
##        0
```

Question 6 b

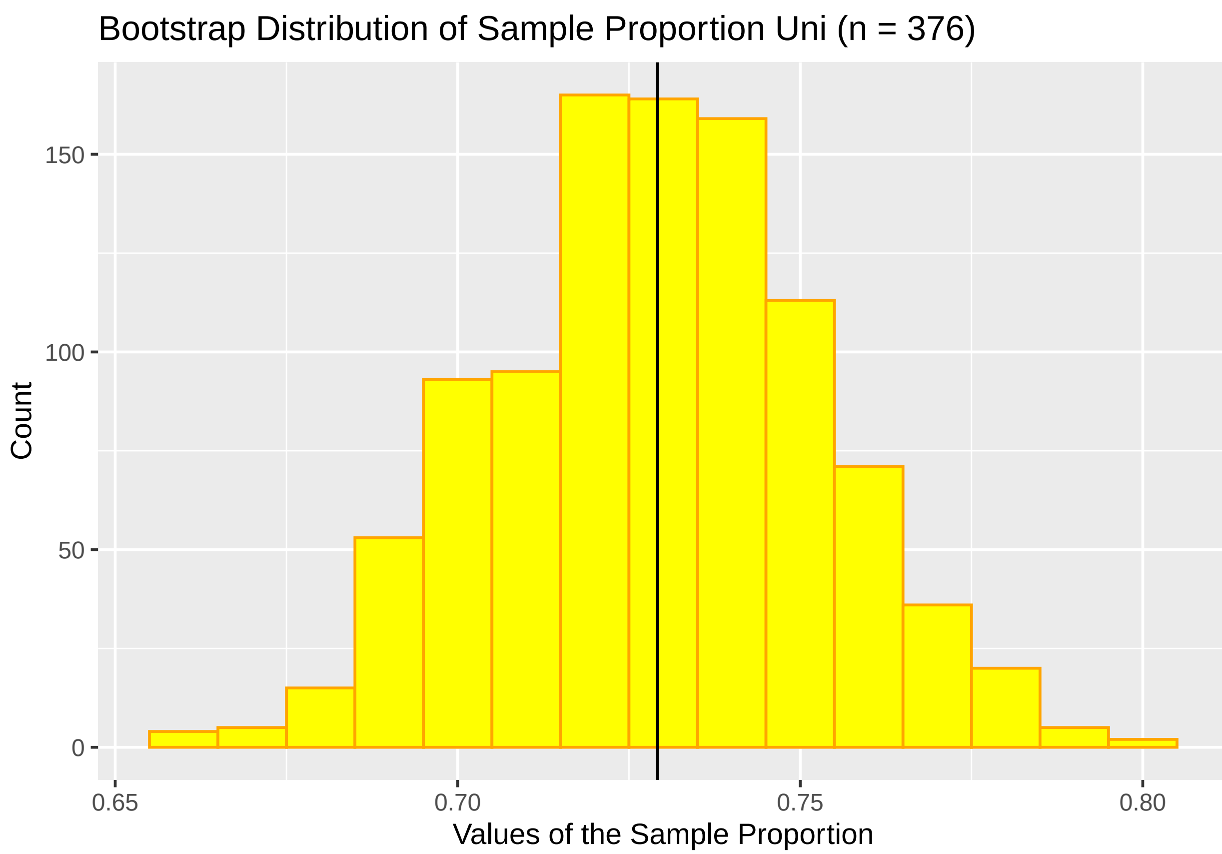
```
#Question 6
#b

#0 meas agreed, 1 means disagreed
data.surveyuni = c(rep(0, 376 -274), rep(1, 274))
phatsuni376 <- numeric(1000)
for(i in 1:1000){
     temp.data <- resample(data.surveyuni)
     phatsuni376[i] <- mean(temp.data)
}
boot_phatuni376.df <- data.frame(phatsuni376)
head(boot_phatuni376.df, 4)
```

```
##   phatsuni376
## 1   0.7074468
```

```
## 2   0.7659574
## 3   0.7180851
## 4   0.7579787
```

```
mean.unidis <- favstats(~phatsuni376, data=boot_phatuni376.df)$mean
ggplot(boot_phatuni376.df, aes(x=phatsuni376)) + geom_histogram(col="orange", fill="yellow", binwidth=0
```




```
favstats(~phatsuni376, data=boot_phatuni376.df)
```

```
##        min       Q1    median        Q3       max      mean         sd    n
##  0.6569149 0.712766 0.7287234 0.7446809 0.8005319 0.7291649 0.02346285 1000
##  missing
##        0
```

Question 6 c

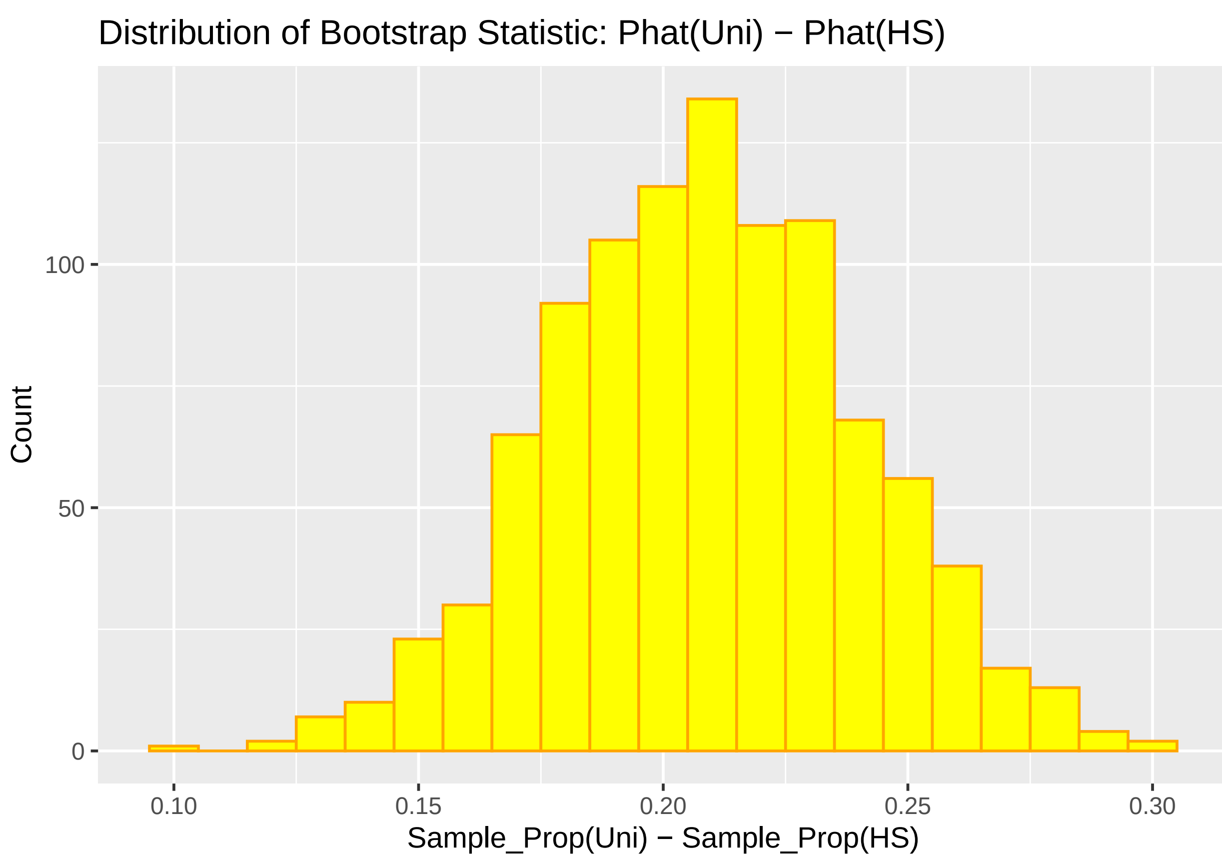
```
#Question 6
#c

phat.hs <- numeric(1000)
phat.uni <- numeric(1000)
phat.difference <- numeric(1000)
for(i in 1:1000){
  temp.data1 <- sample(data.surveyhs, length(data.surveyhs), replace=TRUE)
```

```
  temp.data2 <- sample(data.surveyuni, length(data.surveyuni), replace=TRUE)
  phat.hs[i] <- mean(temp.data1)
  phat.uni[i] <- mean(temp.data2)
  phat.difference[i] <- phat.uni[i] - phat.hs[i]
}
boot.diffprops <- data.frame(phat.hs, phat.uni, phat.difference)
head(boot.diffprops, 5)
```

```
##     phat.hs  phat.uni phat.difference
## 1 0.5164179 0.7234043       0.2069863
## 2 0.5029851 0.7606383       0.2576532
## 3 0.5268657 0.7180851       0.1912194
## 4 0.5000000 0.7154255       0.2154255
## 5 0.5343284 0.7393617       0.2050333
```

```
ggplot(boot.diffprops, aes(x = phat.difference)) + geom_histogram(col="orange", fill="yellow", binwidth=
```




```
qdata(~phat.difference, c(0.025, 0.975), boot.diffprops)
```

```
##      2.5%     97.5%
## 0.1474184 0.2701903
```

```
#We have a 95% bootstrap confidence interval for p_hat(uni) - p_hat(hs) in [0.1493153 ,0.2697577]
```

Question 6 d

From my part d result, it shows that difference between two population proportions are always negative.

$$-0.266833 \leq p_{hs} - p_{uni} \leq -0.149503$$

It means that the proportion of persons with at most a high school education who disagree the science around vaccinations isn't clear greater than the similar proportion of persons with at least an undergraduate university degree In fact, the proportion of persons with at most a high school education who disagree the science around vaccinations is less than the similar proportion of persons with at least an undergraduate university degree. The result from part c also confirmed this statement.

$$0.1493153 \leq \widehat{p}uni - \widehat{p}hs \leq 0.2697577$$

is always positive.

```
#Question 6
#d
prop.test(c( 348 + 1, 274 + 1), c(670+ 2, 376 + 2), correct=FALSE)$conf
```

```
## [1] -0.266833 -0.149503
## attr(,"conf.level")
## [1] 0.95
```

Question 7 a

```
#Question 7
#a
data.q4 = c(16,5,21,19,10,5,8,2,7,2,4,9)
ntimes = 2000
nsize = 12
LC50median = numeric(ntimes)

for(i in 1:ntimes){
  lc50b = sample(data.q4, nsize, replace = TRUE)
  LC50median[i] = median(lc50b)
}

LC50bootq7a = data.frame(LC50median)
head(LC50bootq7a,10)
```

```
##    LC50median
## 1         7.5
## 2         8.0
## 3         4.5
## 4         7.0
## 5         9.0
## 6         5.0
## 7         9.0
## 8         7.5
## 9         6.0
## 10        7.5
```

```
favstats(~LC50median, data=LC50bootq7a)
```

```
##  min Q1 median  Q3 max    mean       sd    n missing
##    2  6    7.5 8.5  19 7.42625 2.122072 2000       0
```

```
qdata(~LC50median,c(0.005,0.995), data = LC50bootq7a)
```

```
##  0.5% 99.5%
##     3    16
```

```
#I can be 99% confident that,
#from the LC50 measurements (in parts per million) for DDT,
#the median of LC50 is somewhere between 4.0 and 17.5
```

Question 7 b I can be 99% confident that,from the LC50 measurements (in parts per million) for DDT, the standard deviation of LC50 is somewhere between 2.208694 and 8.310342.

```
#Question 7
#b


ntimes = 2000
nsize = 12
LC50sd = numeric(ntimes)

for(i in 1:ntimes){
  lc50b = sample(data.q4, nsize, replace = TRUE)
  LC50sd[i] = sd(lc50b)
}

LC50bootq7b = data.frame(LC50sd)
head(LC50bootq7b,4)
```

```
##     LC50sd
## 1 4.814750
## 2 7.049608
## 3 5.879471
## 4 5.757735
```

```
favstats(~LC50sd, data=LC50bootq7b)
```

```
##       min      Q1   median       Q3      max     mean       sd    n missing
##  2.050499 5.33428 6.177918 6.904242 8.691253 6.023018 1.203349 2000       0
```

```
qdata(~LC50sd,c(0.005,0.995), data = LC50bootq7b)
```

```
##     0.5%    99.5%
## 2.340227 8.306170
```

Question 8 a It can be 99% confident that, from this sample of n=858 Alberta voters, the proportion of all Alberta residents (aged 18 years of age or older) that will vote for their respective NDP MLA-candidate is somewhere between 0.3778315 and 0.4435142

```
n.vote = 858
ndp.vote = 352
#Compute a 95% confidence interval for P_NDP
binom.test(ndp.vote, n.vote, ci.method="Plus4")$conf
```

```
## [1] 0.3778315 0.4435142
## attr(,"conf.level")
## [1] 0.95
## attr(,"method")
## [1] "plus4"
```

Question 8 b

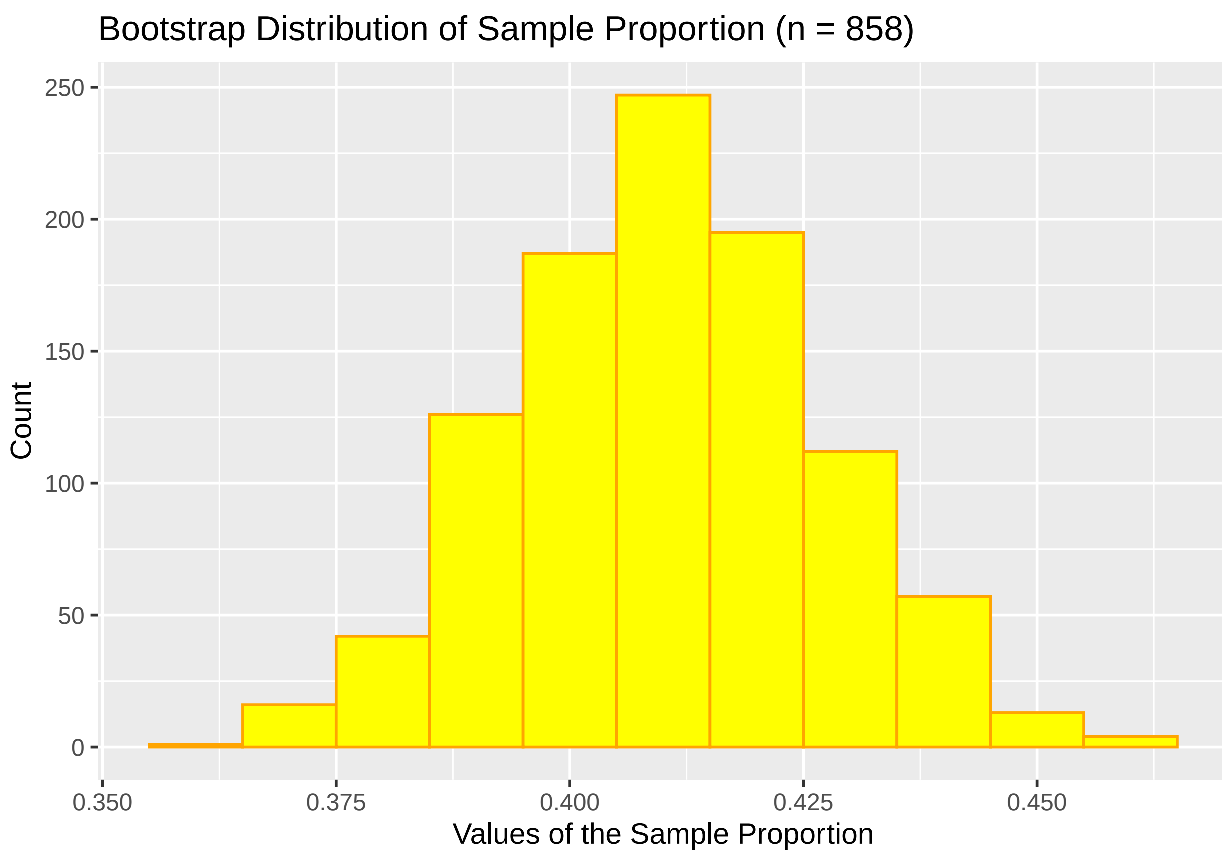
```
#0 meas vote for other partys, 1 means vote for NDP
pndp = (352+2) / (858+4)

data.votendp = c(rep(0, (858*(1-pndp))), rep(1, (858 * pndp))) #X_NDP + 2 / n + 4

phatsndp <- numeric(1000)
for(i in 1:1000){
     temp.data <- resample(data.votendp)
     phatsndp[i] <- mean(temp.data)
}
boot_phatndp.df <- data.frame(phatsndp)
head(boot_phatndp.df, 4)
```

```
##    phatsndp
## 1 0.3663944
## 2 0.4165694
## 3 0.4014002
## 4 0.3815636
```

```
ggplot(boot_phatndp.df, aes(x=phatsndp)) + geom_histogram(col="orange", fill="yellow", binwidth=0.01) +
```




```
favstats(~phatsndp, data=boot_phatndp.df)
```

```
##        min        Q1    median        Q3       max      mean         sd    n
##  0.3640607 0.3990665 0.4107351 0.4212369 0.4632439 0.4104877 0.01661464 1000
##  missing
##        0
```

Question 8 c

```
qdata(~phatsndp, c(0.025,0.975), data=boot_phatndp.df)
```

```
##      2.5%     97.5%
## 0.3803967 0.4434072
```

Question 8 d Part a gives a confidence interval of [0.3778315,0.4435142]. Part c gives a confidence interval of [0.3780630,0.4446033]. The two results are very similar. I can be 95% confident that, if a provincial election were held "tomorrow", the proportion of Alberta voters who would vote for their NDP MLA-candidate is somewhere between 0.378 and 0.444