DATA 602 HW 2

Kane Smith

2022-09-28

Contents

Setting seed so that my explanations match my outputs from bootstrapping and simulations using random sampling match my explanations:

```
set.seed(672)
```

Question 1

```
q1_data <- read.csv('http://people.ucalgary.ca/~jbstall/DataFiles/Data602Assignment1Question11.csv')
```

a)

Sample mean calculated in Q11: 5.6875.

Because the population is assumed to follow a normal distribution, we know for certain that the sample means will also follow a normal distribution, even though they would have follow an approx. normal distribution anyway.

Mean of population: 5.0, standard deviation of population, 1.5.

```
1 - pnorm(5.6875, 5, 1.5/sqrt(12))
```

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

There is a 0.0562 probability that we will observe a sample mean of at least 5.6875 with a sample size of n = 12.

b)

We do not know what the distribution that the sample standard deviations follow so we must transform it into a chi-squared distribution. We want to calculate the probability that the sample standard deviation will be between 0.5 and 1.0. Turning these values into chi-squared using the formula $\frac{(n-1)S^2}{\sigma^2}$

$$\frac{(12-1)(0.5)^2}{(1.5)^2}\frac{2.75}{2.25} = 1.2222 \frac{(12-1)(1.0)^2}{(1.5)^2}\frac{11}{2.25} = 4.8889$$

```
pchisq(4.8889,11) - pchisq(1.2222, 11)
```

```
## [1] 0.06343421
```

There is a probability of 0.0634 of observing a sample standard deviation between 0.5 and 1.0 with the sample size n = 12.

Question 2

a)

The distribution of the sample proportion will be approx. normal with the mean (balancing point) being 0.42. The standard deviation (measure of spread) of the sample proportions is $\sqrt{\frac{0.42(1-0.42)}{1426}} = 0.01307$.

b)

```
pnorm(0.3794, 0.42, 0.01307)
```

```
## [1] 0.0009470605
```

There is a probability of 0.0009 that the proportion will be at **most** 0.3794.

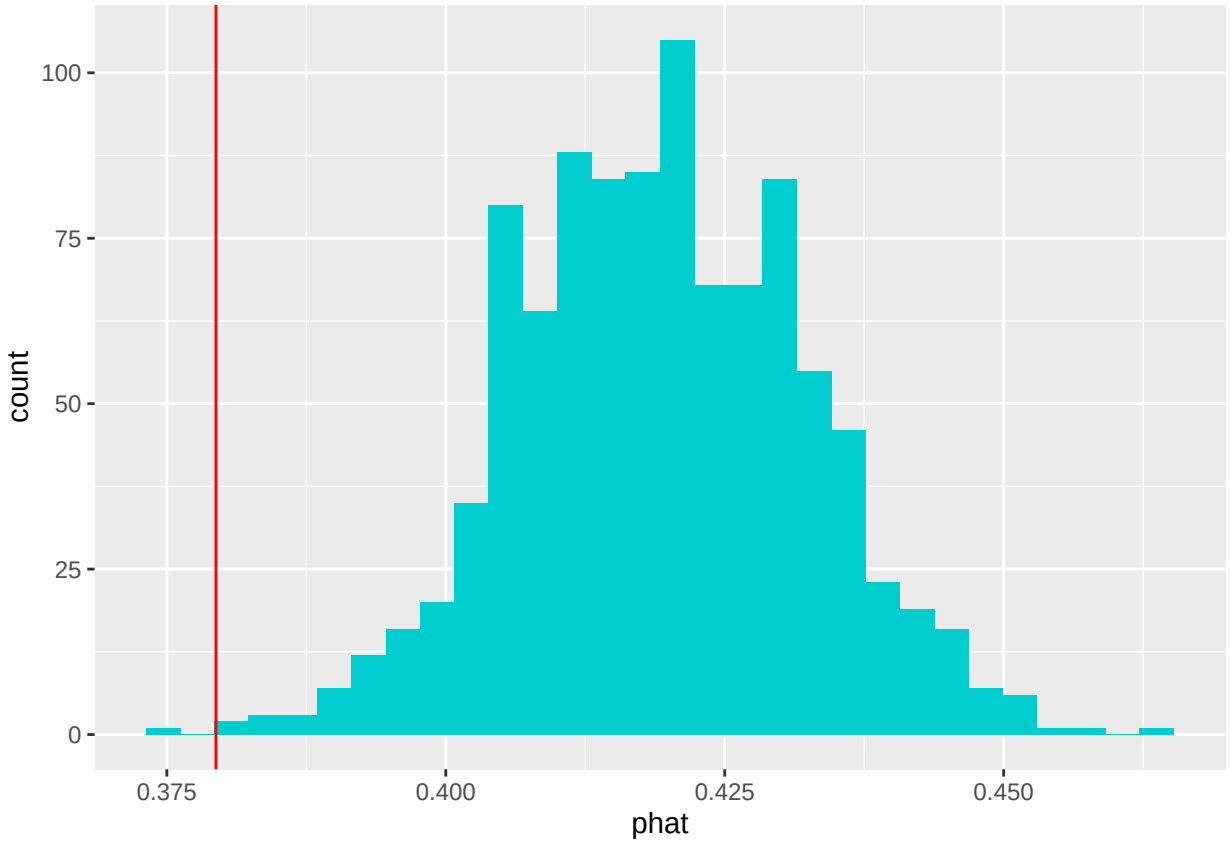
c)

```
nsamples = 1000
q2c_vec <- c(rep(0, 1426*0.58), rep(1, 1426*0.42))
phat = numeric(nsamples)
for(i in 1:nsamples){
  q2.sdata = sample(q2c_vec,size=1426, replace=TRUE)
  phat[i] = mean(q2.sdata)
}
q2.df = data.frame(phat)
numerator <- q2.df %>% filter(phat <= 0.3794)
nrow(numerator) / 1000
```

```
## [1] 0.001
```

```
ggplot(q2.df, aes(x = phat)) + geom_histogram(fill = "cyan3") + geom_vline(xintercept = 0.3794, color =
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The proportion of sample proportions that are less than or equal to 0.3794 is approx. ~0.001.

Question 3

With a mean number of matching numbers of 0.7347 in a week, and with 52 weeks in the year (n = 52), we can assume the distribution of sample means is approx. normal. We can use pnorm to calculate the area number the curve to the right of 1, giving us the probability that Billy matches at least one number in the year.

```
1 - pnorm(1, 0.7347, 0.76/sqrt(52))
```

```
## [1] 0.005913843
```

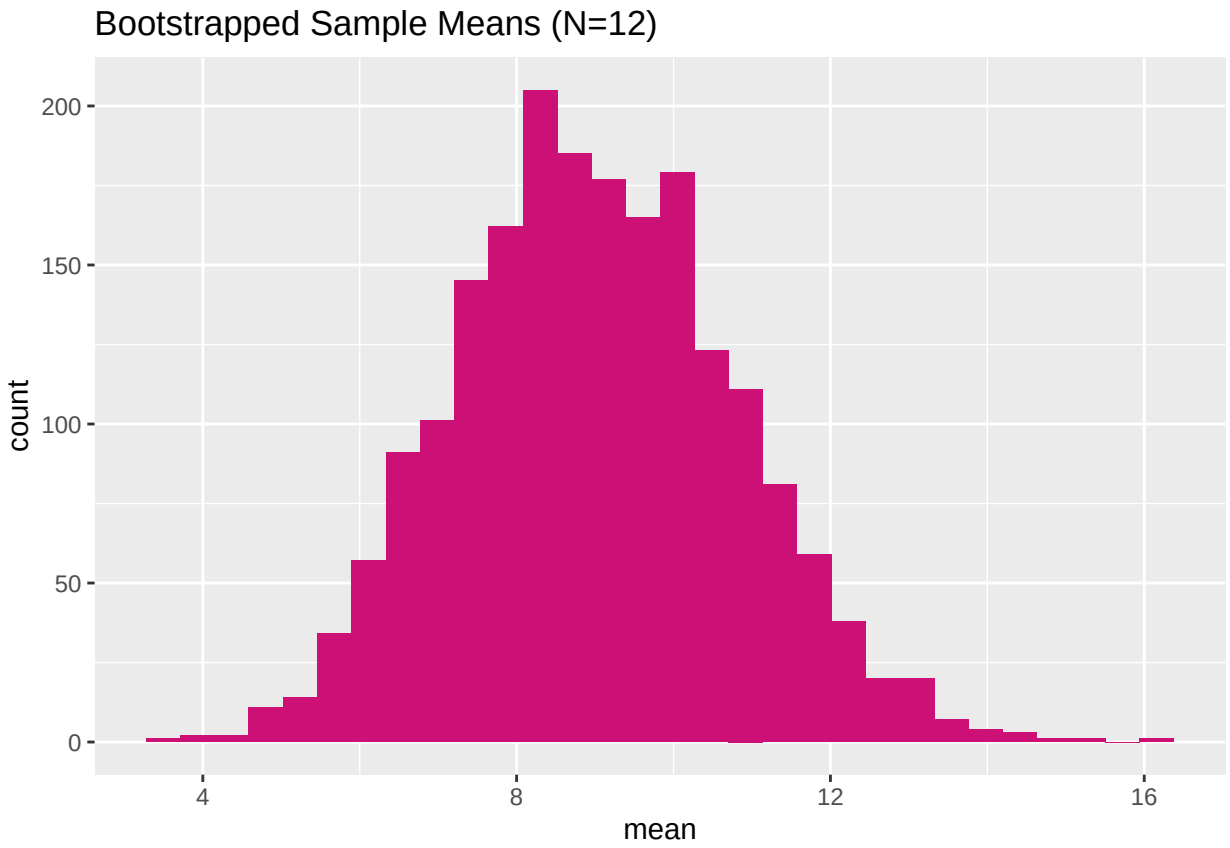
From pnorm we get 0.0059. This means it is very unlikely that Billy's claim of matching at least one number in 52 weeks is true.

Question 4

a)

```
q4_vec <- c(16,5,21,19,10,5,8,2,7,2,4,9)
bootstrap.q4<- do(2000) * mean(resample(q4_vec))
ggplot(bootstrap.q4, aes(x = mean)) + geom_histogram(fill = "deeppink3") + ggtitle("Bootstrapped Sample
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



b)

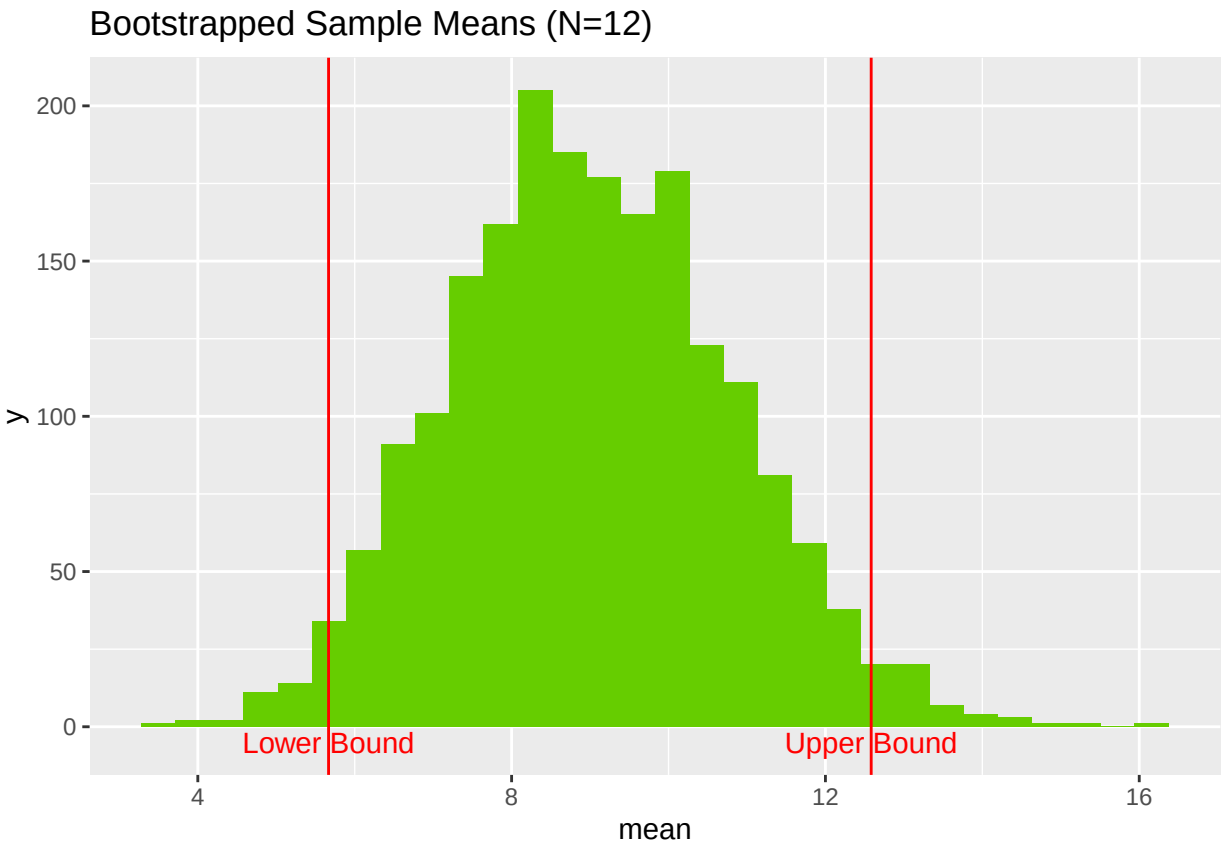
Because we want a 95% confidence interval, we must have 2.5% or our data on each side of the interval.

```
qdata(~mean, c(0.025, 0.975), data=bootstrap.q4)
```

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

```
ggplot(bootstrap.q4, aes(x = mean)) + geom_histogram(fill = "chartreuse3") + ggtitle("Bootstrapped Samp
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



This interval is saying that we can say with 95% confidence that the mean is between 5.6667 and 12.5833.

c)

```
t.test(q4_vec)$conf
```

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

The 95% confidence interval calculated using the t.test is (4.9181, 13.0819).

d)

Confidence interval from t.test: (4.9181, 13.0819)

Confidence interval from bootstrapping: (5.6667, 12.5833)

If I had to report one the these intervals, I would report the one calculated from bootstrapping. This is because bootstrapping is assumption-less while to conduct a t.test, we must assume that the population follows a normal distribution, or the sample size is sufficiently large (n >= 25).

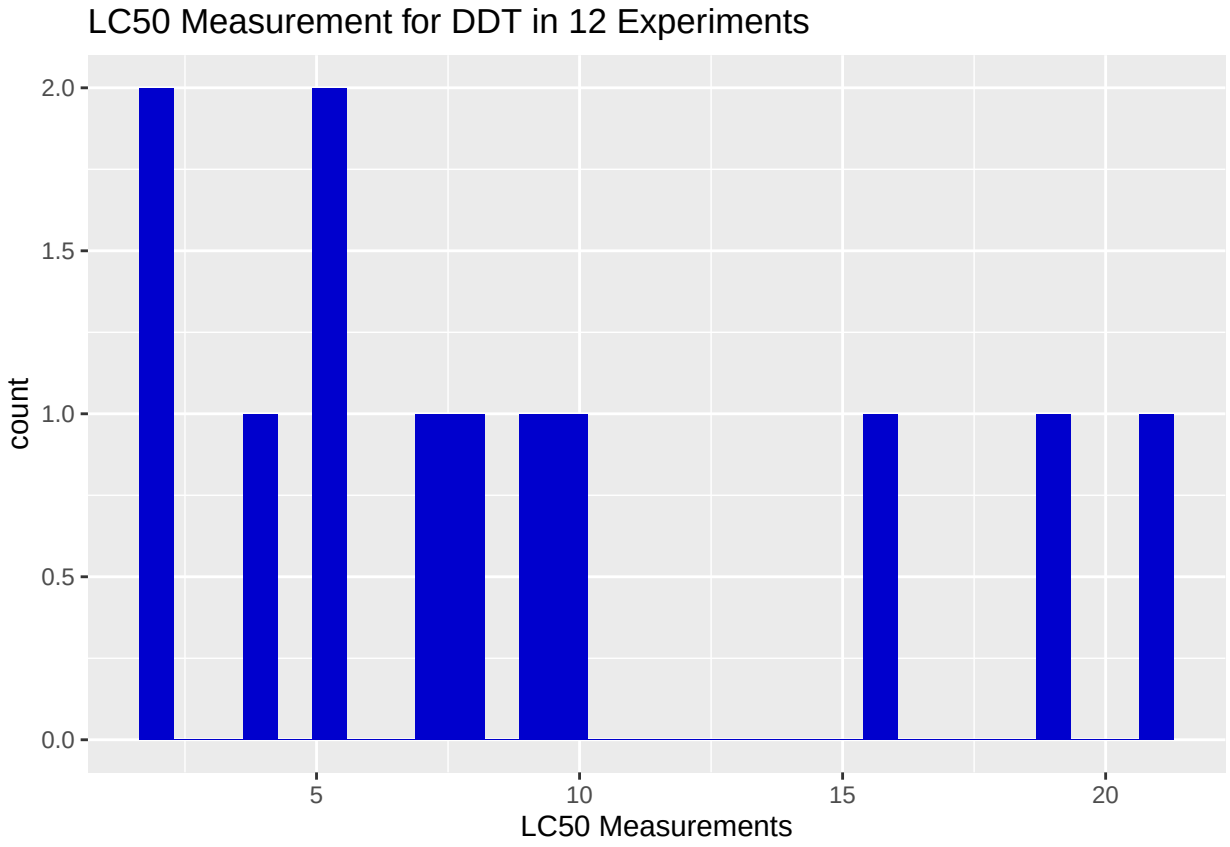
e)

One of the assumptions required to use a t.test to calculate a confidence interval is the underlying data must follow an approx. normal distribution. The main characteristics of a normal distribution is a symmetric curve where the mean lies in the middle and approx 68% of the data lies within one standard deviation of the mean. The plot of the data below is not symmetric and the mean does not appear to be in the middle.

Since the underlying data does not follow a normal distribution, we must have at least 25 observations in our sample to be able to say the sample means follow an approx. normal distribution. However, the are only 12 observations in our sample, so we should not use the confidence interval calculated using t.test as it is not valid.

```
q4_df <- data.frame(i =1:length(q4_vec), q4_vec)
ggplot(q4_df, aes(x=q4_vec)) + geom_histogram(fill = "blue3") + ggtitle("LC50 Measurement for DDT in 12
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Question 5

a)

Our expected proportion will be 571/1866 = 0.3060. We can assume the sample proportions follow an approx. normal distribution. 1866(0.3060) = 571 >= 10 and 1866(1- 0.3060) = 1295 >= 10.

```
error <- qnorm(0.975)*sqrt((0.3060*(1-0.3060))/1866)
0.3060 - error
```

```
## [1] 0.285091
```

```
0.3060 + error
```

```
## [1] 0.326909
```

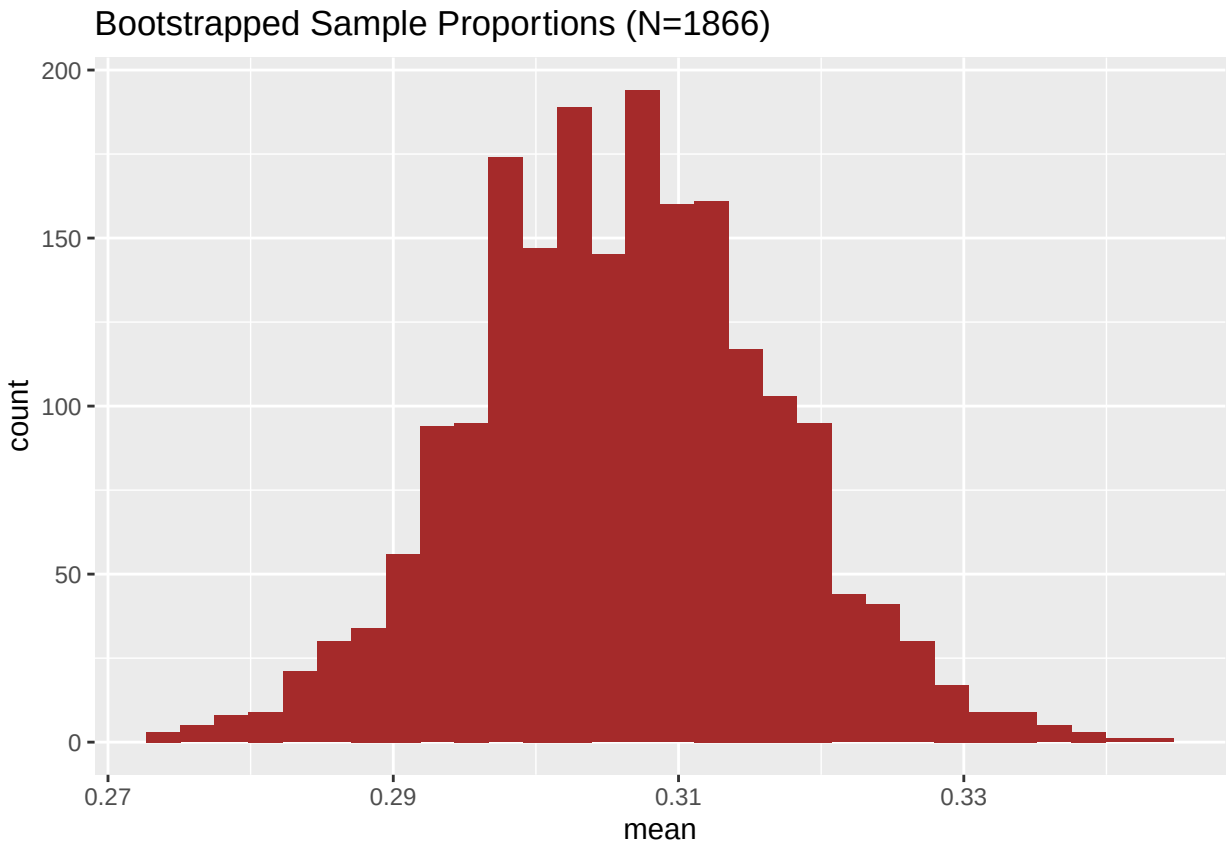
The 95% confidence interval is (0.2851, 0.3269). This means we can say with 95% confidence that the proportion is between 0.2851 and 0.3269.

b)

First, I create a vector with 571 1's and 1295 0's to represent the sample. We can use this vector to bootstrap. Note that the order of 1's and 0's does not matter as re-sampling is random.

```
q5_vec <- c(rep(1, 571), rep(0, 1866 - 571))
bootstrap.q5 <- do(2000) * mean(resample(q5_vec))
ggplot(bootstrap.q5, aes(x = mean)) + geom_histogram(fill = "brown") + ggtitle("Bootstrapped Sample Pro
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



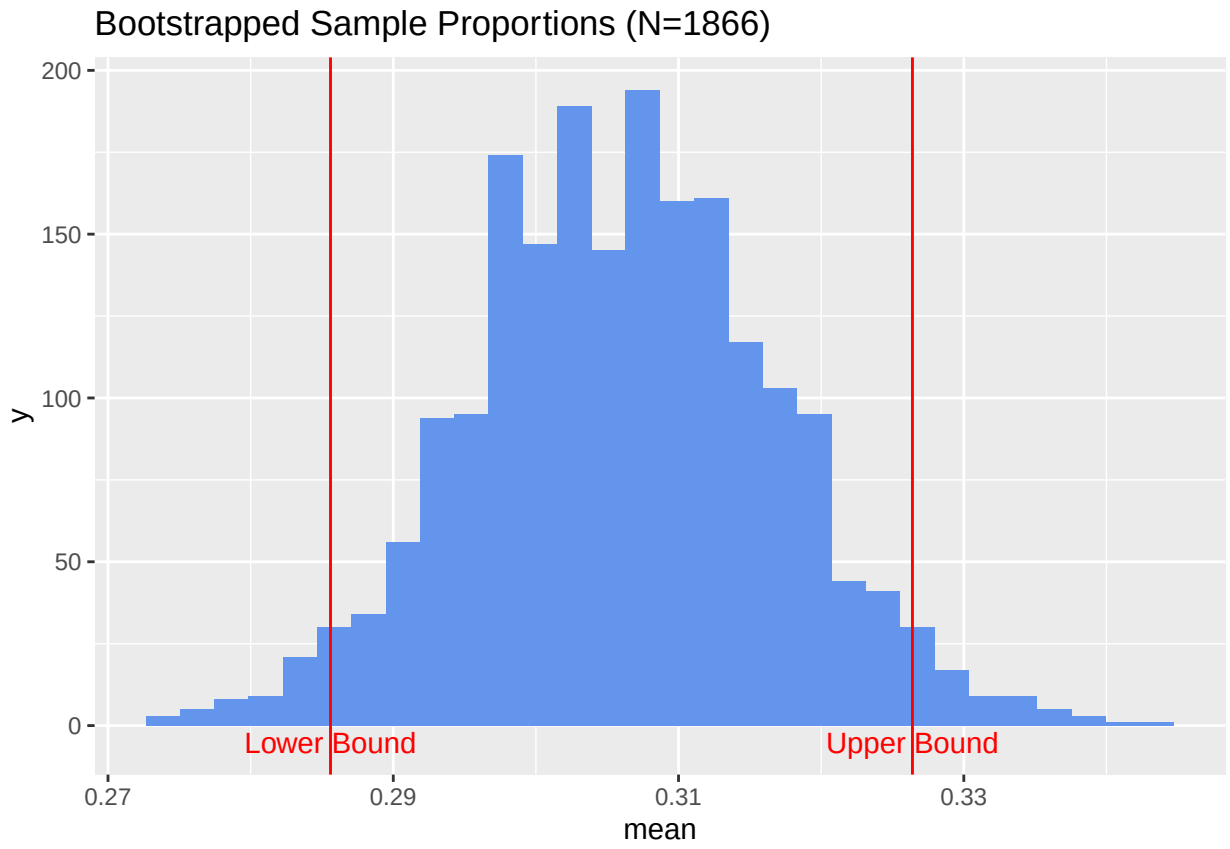
c)

```
qdata(~mean, c(0.025, 0.975), data=bootstrap.q5)
```

```
##      2.5%     97.5%
## 0.2856243 0.3274384
```

```
ggplot(bootstrap.q5, aes(x = mean)) + geom_histogram(fill = "cornflowerblue") + ggtitle("Bootstrapped Sa
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Our 95% confidence interval from bootstrapping says that with 95% confidence, we can say out proportion is between 0.2856 and 0.3264.

d)

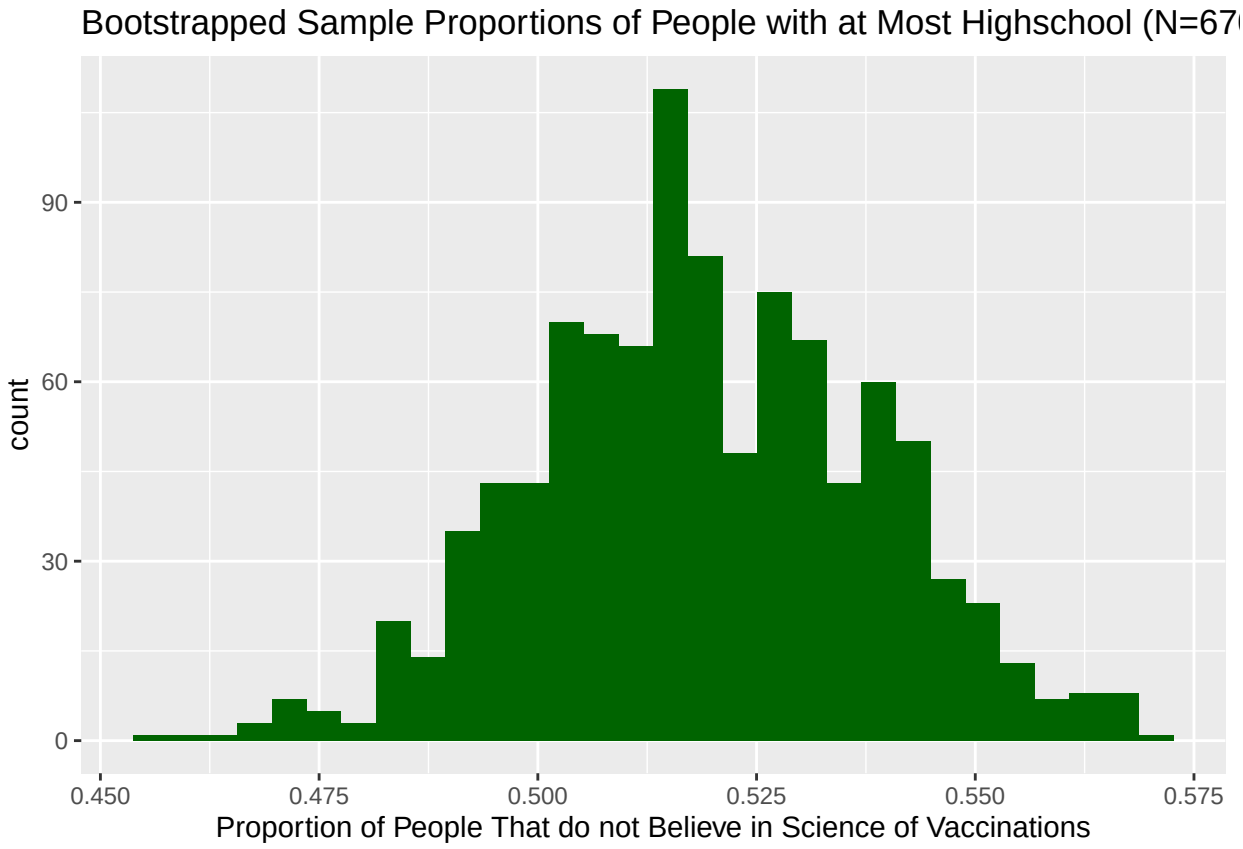
Our 95% confidence intervals calculated in a) and c) are close, but slightly different. I would report the confidence interval calculated from bootstrapping in c). This is because bootstrapping makes no assumptions of normality of the underlying distribution, so it is likely to be more accurate. In a), in order to calculate the confidence interval, we had to assume that the sample proportion were normally distributed. Although it is approximately normal, it is not exact which creates inaccuracy in our confidence interval.

Question 6

a)

```
q6a_vec <- c(rep(1, 348), rep(0, 670-348))
q6a_bootstrap = do(1000)*mean(resample(q6a_vec))
ggplot(q6a_bootstrap, aes(x = mean)) + geom_histogram(fill = "darkgreen") + ggtitle("Bootstrapped Sample
```

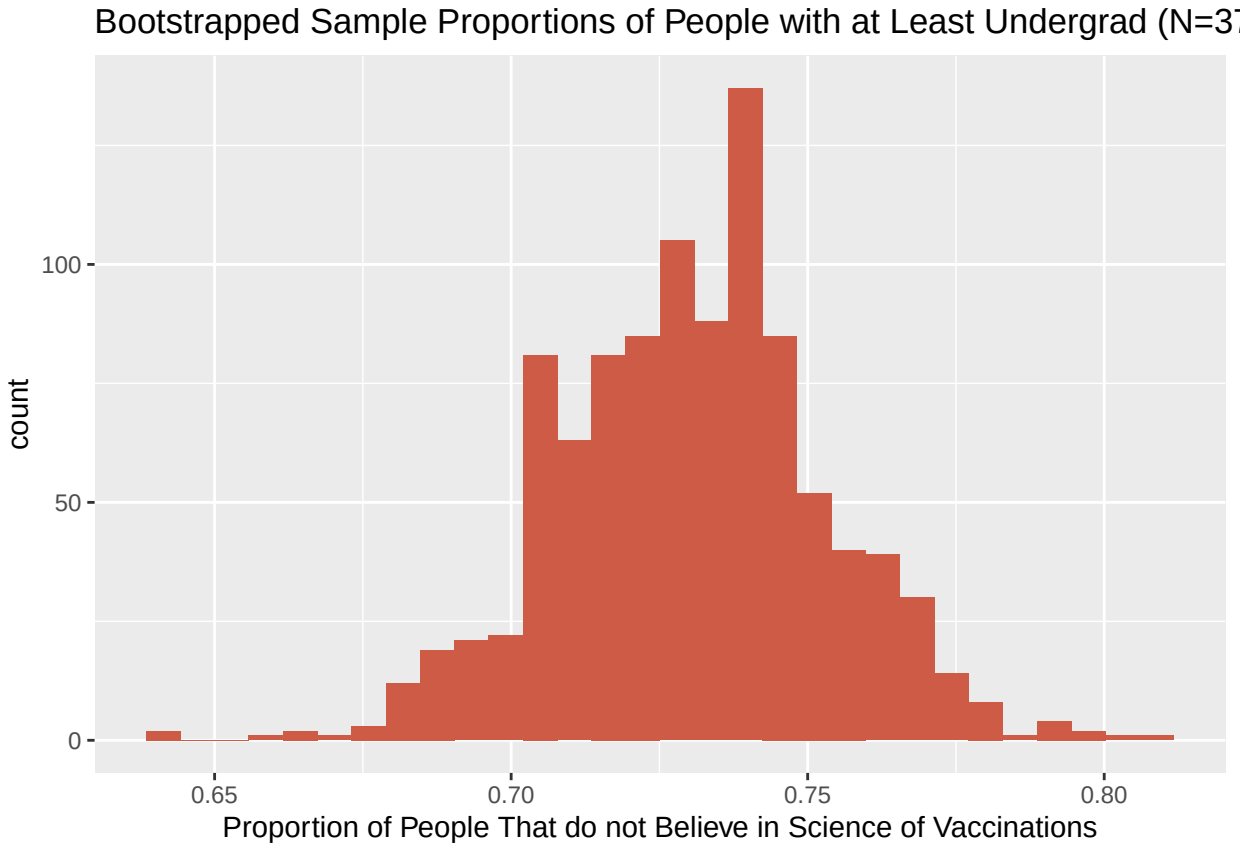
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



b)

```
q6b_vec <- c(rep(1, 274), rep(0, 376-274))
q6b_bootstrap = do(1000)*mean(resample(q6b_vec))
ggplot(q6b_bootstrap, aes(x = mean)) + geom_histogram(fill = "coral3") + ggtitle("Bootstrapped Sample Pr
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

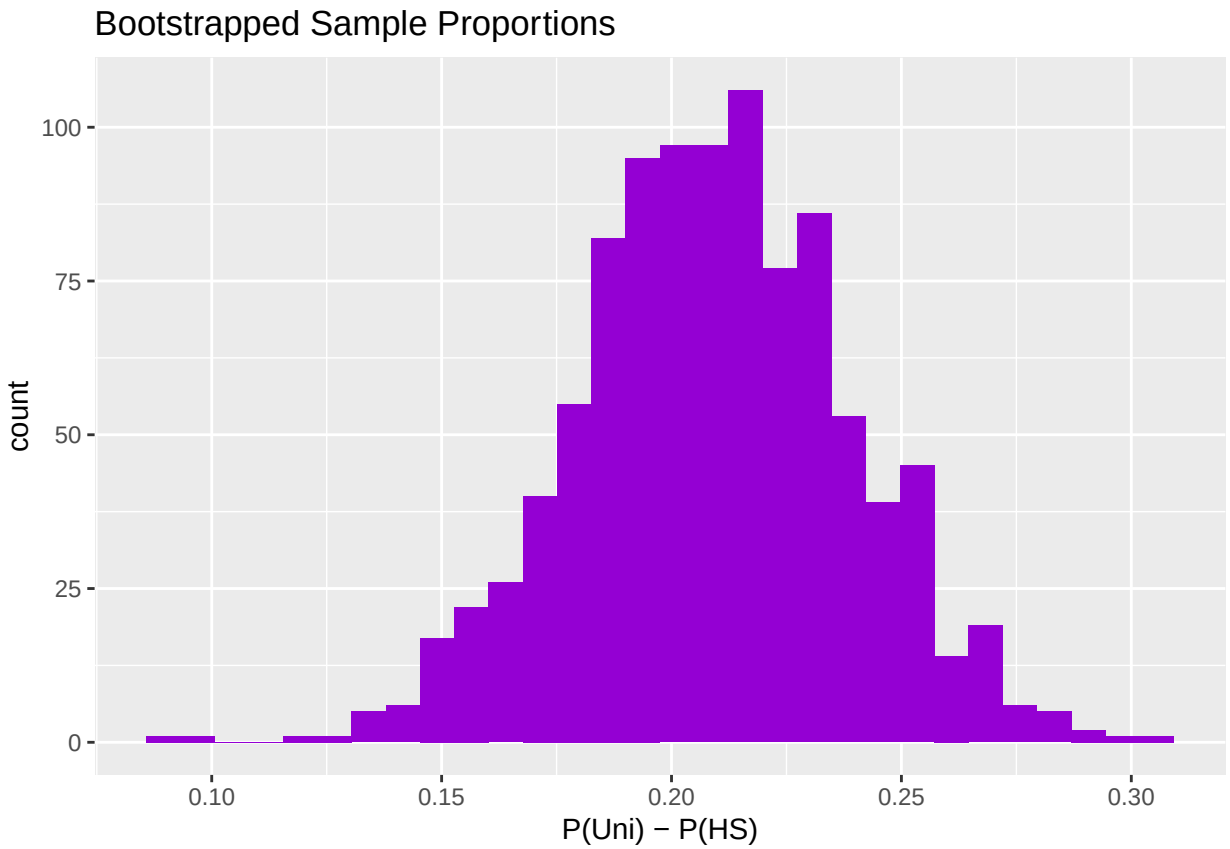


c)

To calculate $P_{Uni} - P_{HS}$, we will take sample proportions from the vector created in b) with replacement and subtract them by sample proportions from the vector created in a) with replacement.

```
q6c_bootstrap = do(1000)*(mean(resample(q6b_vec))- mean(resample(q6a_vec)))
ggplot(q6c_bootstrap, aes(x = result)) + geom_histogram(fill = "darkviolet") + ggtitle("Bootstrapped Sa
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



d)

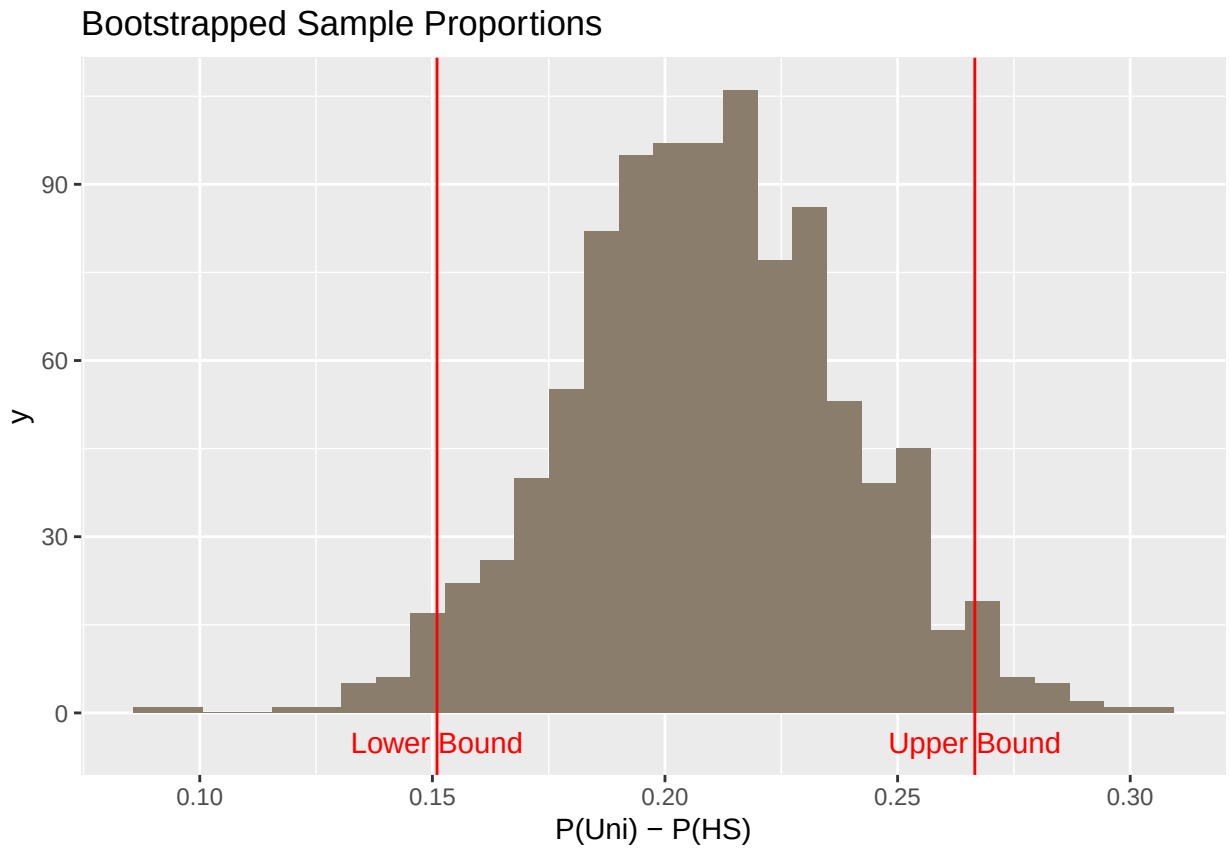
Calculation of 95% confidence interval:

```
q6c_phat.df <- data.frame(q6c_bootstrap)
qdata(~result, c(0.025, 0.975), data=q6c_phat.df)
```

```
##     2.5%    97.5%
## 0.150152 0.267338
```

```
ggplot(q6c_bootstrap, aes(x = result)) + geom_histogram(fill = "bisque4") + ggtitle("Bootstrapped Sampl
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We can say with 95% confidence that the difference between the sample proportion of people with at most high school diplomas that do not believe in the science behind the vaccine and the people with at least undergraduate degrees who do not believe in the science behind the vaccine is between 0.1510 and 0.2666. Since the difference is positive, we can say with 95% confidence that the proportion of people with at least an undergraduate degree that do not believe the science is higher than the proportion of people with at most a high school diploma.

Question 7

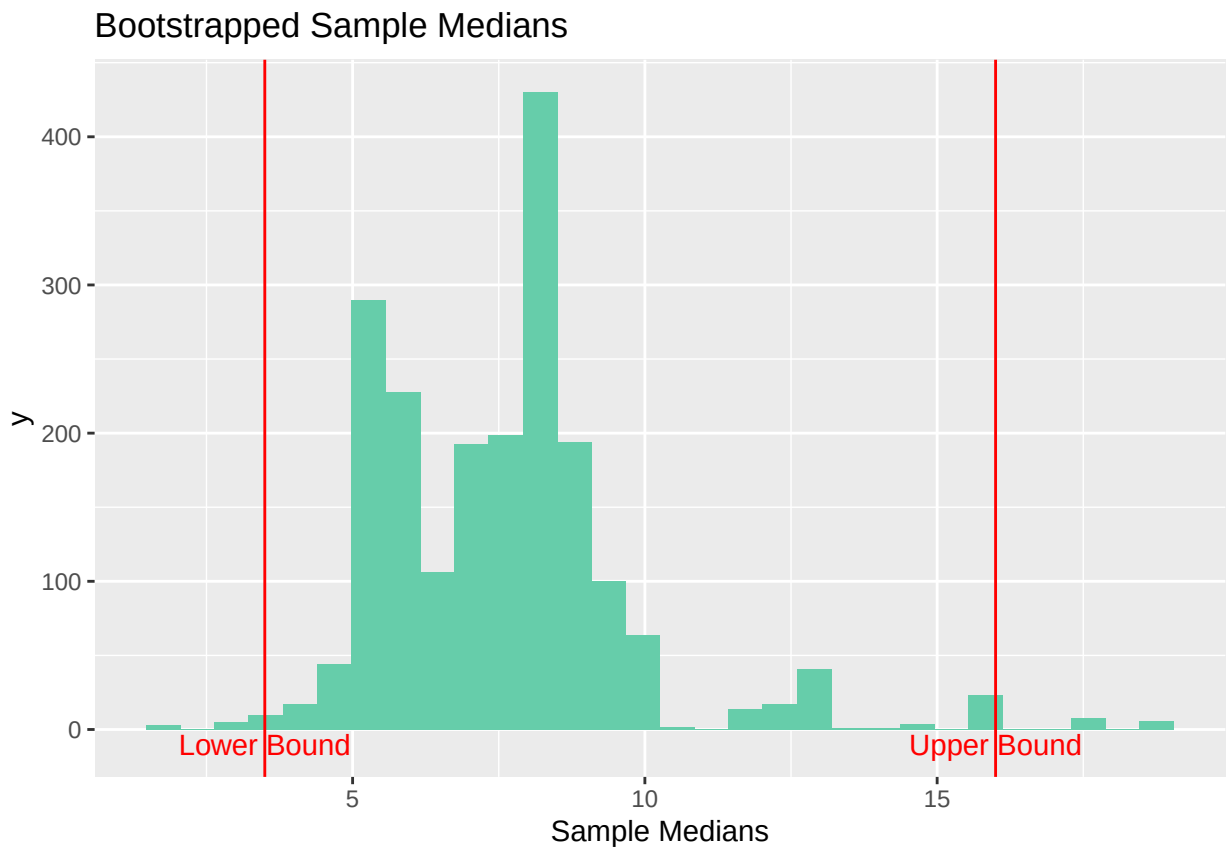
a)

```
q7_vec <- c(16,5,21,19,10,5,8,2,7,2,4,9)
bootstrap.q7a<- do(2000) * median(resample(q7_vec))
qdata(~median, c(0.005, 0.995), data=bootstrap.q7a)
```

```
##  0.5% 99.5%
##   3.5  17.5
```

```
ggplot(bootstrap.q7a, aes(x = median)) + geom_histogram(fill = "aquamarine3") + ggtitle("Bootstrapped Sa
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



In order to calculate a 99% confidence interval, we must have 0.5% of the data on each side of interval.

This 99% confidence interval means that we can say with 99% confidence that the median is between 3.4975 and 16.0.

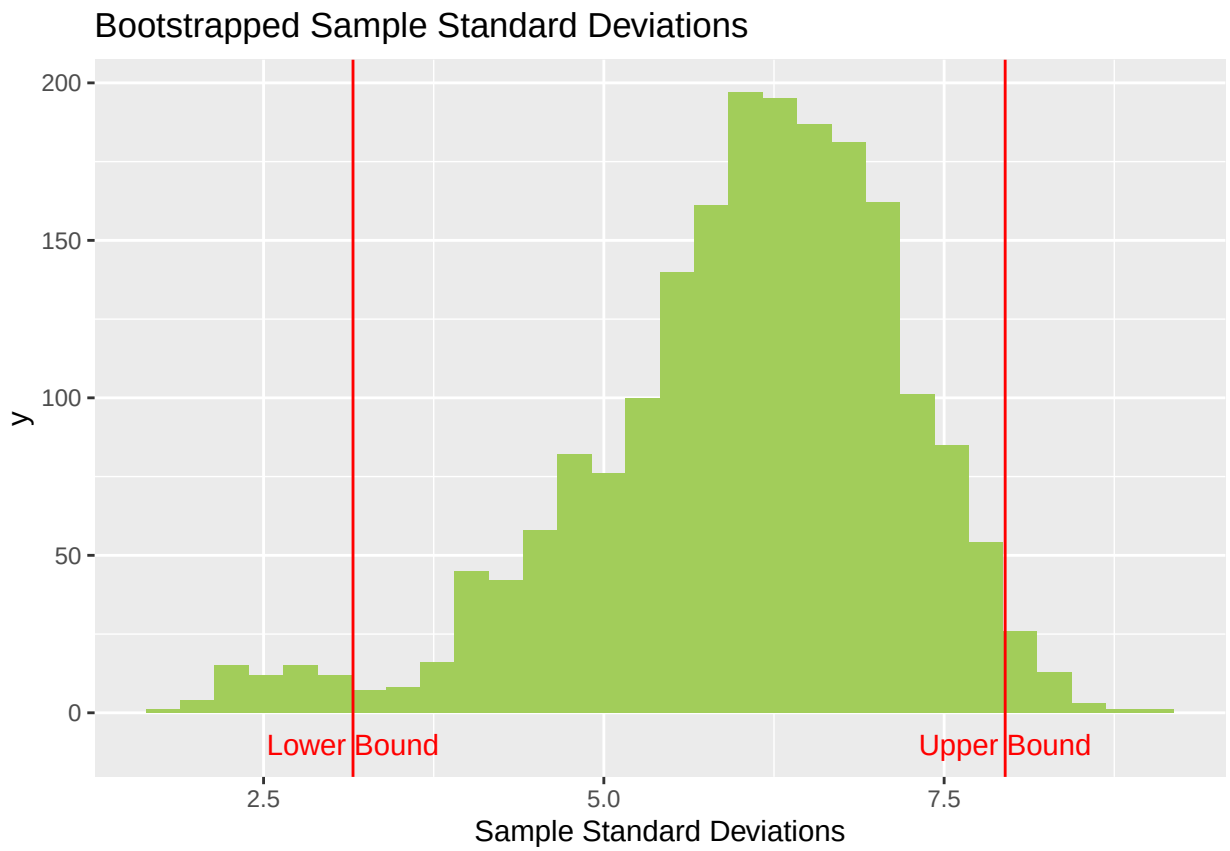
b)

```
bootstrap.q7b <- do(2000) * sd(resample(q7_vec))
qdata(~sd, c(0.025, 0.975), data=bootstrap.q7b)
```

```
##     2.5%    97.5%
## 2.937480 7.912435
```

```
ggplot(bootstrap.q7b, aes(x = sd)) + geom_histogram(fill = "darkolivegreen3") + ggtitle("Bootstrapped Sa
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



This 95% confidence interval means that we can say with 95% confidence that the standard deviation is between 3.1572 and 7.9484.

Question 8

a)

The proportion of NDP is 358 / 858 = 0.4172, and therefore its compliment is 1 - 0.4172 = 0.5828. We can assume the sample proportions follow an approx. normal distribution. 858(0.4172) = 358 >= 10 and 858(1-0.4172) = 500 >= 10.

```
q8_error <- qnorm(0.975)*sqrt((0.4172*(1-0.4172))/858)
0.4172 - q8_error
```

```
## [1] 0.3842059
```

```
0.4172 + q8_error
```

```
## [1] 0.4501941
```

The 95% confidence interval that the proportion Albertans over the age of 18 will vote NDP is (0.3842, 0.4502). This means we can say with 95% confidence that the proportion of Albertans that will vote for NDP is between 0.3842 and 0.4502.

b)

$P_{NDP} = \frac{X_{NDP}+2}{858+4}$. To find X(NDP), we will use the rbinom function that takes n = 858 and the probability of success (vote for NDP) = 0.4172. This will return a vector of size 858 with 1's and 0's, 1's being the people who voted for NDP.

```
q8_vec <- c(rep(1,358), rep(0, 858 - 358))
q8_bootstrap = do(1000)*((sum(resample(q8_vec))+2)/(858 +4))
q8_phat.df <- data.frame(q8_bootstrap)
```

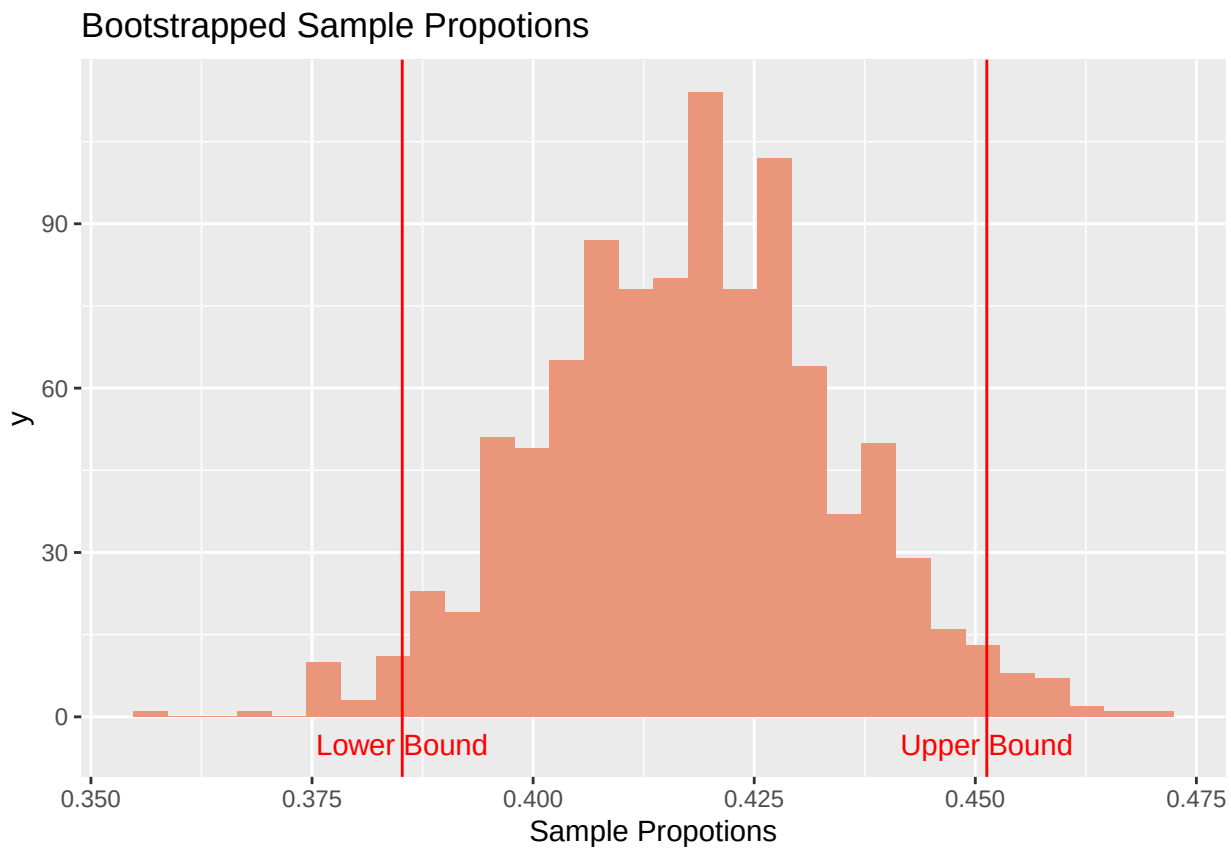
c)

```
qdata(~result, c(0.025, 0.975), data=q8_phat.df)
```

```
##      2.5%     97.5%
## 0.3851508 0.4512761
```

```
ggplot(q8_bootstrap, aes(x = result)) + geom_histogram(fill = "darksalmon") + ggtitle("Bootstrapped Samp
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Our 95% confidence interval from bootstrapping says that with 95% confidence, we can say out proportion is between 0.3852 and 0.4513.

d)

Both confidence intervals from a) and c) are similar. If we take both into consideration, we can say with 95% confidence that the proportion of Alberta voters that would vote for NDP would be between approximately 0.38 and 0.45.

It should be noted that the confidence interval calculated in c) is more reliable because it does not require any assumptions about the distribution of sample proportions, also it is a conservative calculation of the proportion since we added a larger scalar to the denominator compared to the numerator.

Session info:

```
sessionInfo()
```

```
## R version 4.1.3 (2022-03-10)
## Platform: x86_64-conda-linux-gnu (64-bit)
## Running under: Ubuntu 22.04.1 LTS
##
## Matrix products: default
## BLAS/LAPACK: /opt/conda/lib/libopenblasp-r0.3.21.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] mosaic_1.8.3      ggridges_0.5.3    mosaicData_0.20.2 ggformula_0.10.1
## [5] ggstance_0.3.5    Matrix_1.4-1      lattice_0.20-45   dplyr_1.0.9
## [9] ggplot2_3.3.6
##
## loaded via a namespace (and not attached):
##  [1] ggrepel_0.9.1     Rcpp_1.0.9        tidyr_1.2.0       assertthat_0.2.1
##  [5] digest_0.6.29     utf8_1.2.2        ggforce_0.3.4     R6_2.5.1
##  [9] plyr_1.8.7        backports_1.4.1   labelled_2.9.1    evaluate_0.16
## [13] highr_0.9         pillar_1.8.1      rlang_1.0.4       rstudioapi_0.14
## [17] rmarkdown_2.15    labeling_0.4.2    splines_4.1.3     readr_2.1.2
## [21] stringr_1.4.1     htmlwidgets_1.5.4 polyclip_1.10-0   munsell_0.5.0
## [25] broom_1.0.0       compiler_4.1.3    xfun_0.32         pkgconfig_2.0.3
## [29] htmltools_0.5.3   tidyselect_1.1.2  tibble_3.1.8      gridExtra_2.3
## [33] mosaicCore_0.9.0  fansi_1.0.3       crayon_1.5.1      tzdb_0.3.0
## [37] withr_2.5.0       MASS_7.3-58.1     grid_4.1.3        gtable_0.3.0
## [41] lifecycle_1.0.1   DBI_1.1.3         magrittr_2.0.3    scales_1.2.1
## [45] cli_3.3.0         stringi_1.7.8     farver_2.1.1      leaflet_2.1.1
## [49] ellipsis_0.3.2    ggdendro_0.1.23   generics_0.1.3    vctrs_0.4.1
## [53] tools_4.1.3       forcats_0.5.2     glue_1.6.2        tweenr_2.0.1
## [57] purrr_0.3.4       hms_1.1.2         crosstalk_1.2.0   fastmap_1.1.0
## [61] yaml_2.3.5        colorspace_2.0-3  knitr_1.39        haven_2.5.0
```

Footnote: Adding for the record

This is the way I believe Q3 was to be done, however I was advised by the professor to submit the other answer due to how the TAs were told to mark thew question.

```
1- (choose(6,0)*choose(43,6))/choose(49,6)
```

```
## [1] 0.564035
```

```
1- pbinom(1, 52, 0.564035)
```

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

Probability of success (Matching at least one number): 0.564035

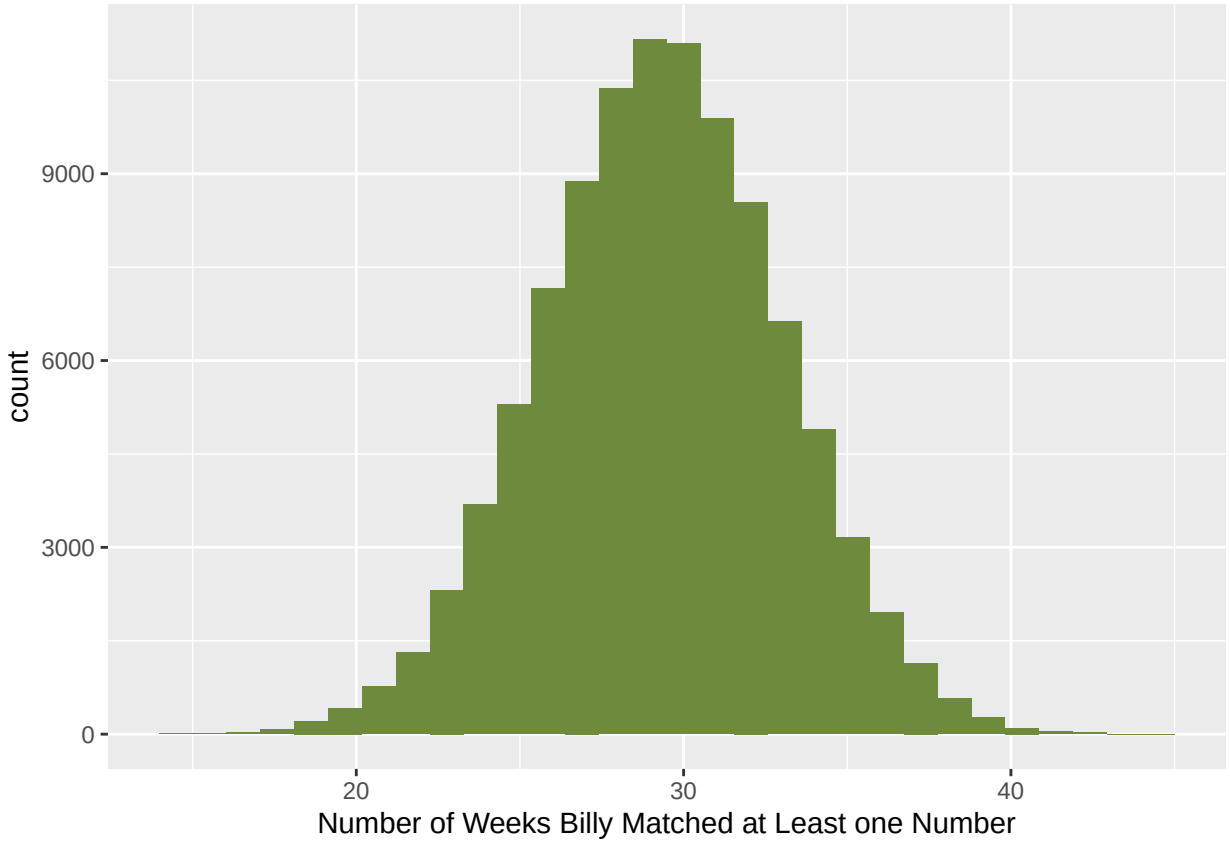
Probability of fail (Match no numbers): (1-0.564035) = 0.435965

Using pbinom to calculate the probability of observing at least one matching number, we get 1. This means that Billy's claim that he will observe at least one number throughout 52 weeks, given the probability distribution for matching numbers in a week, is almost definitely true.

Below is a distribution of the showing the frequency of the amount weeks where Billy matches at least one number. This was created using rbinom of 100,000 trials. As you can see visually, the likelihood of have 0 weeks where Billy matches at least one number is very small.

```
q3_vec <- rbinom(100000, 52, 0.564035)
matches.df <- data.frame(q3_vec)
ggplot(matches.df, aes(x = q3_vec)) + geom_histogram(fill = "darkolivegreen4") + xlab("Number of Weeks
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


9000
6000
3000
0
count
20
30
40
Number of Weeks Billy Matched at Least one Number