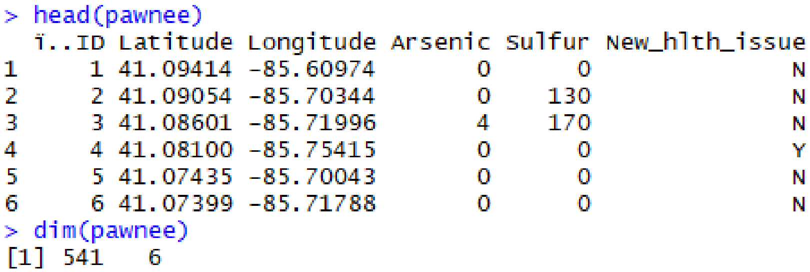
Stats 12 Lab 4 Submission

Brian Tehrani

UID: 604715464

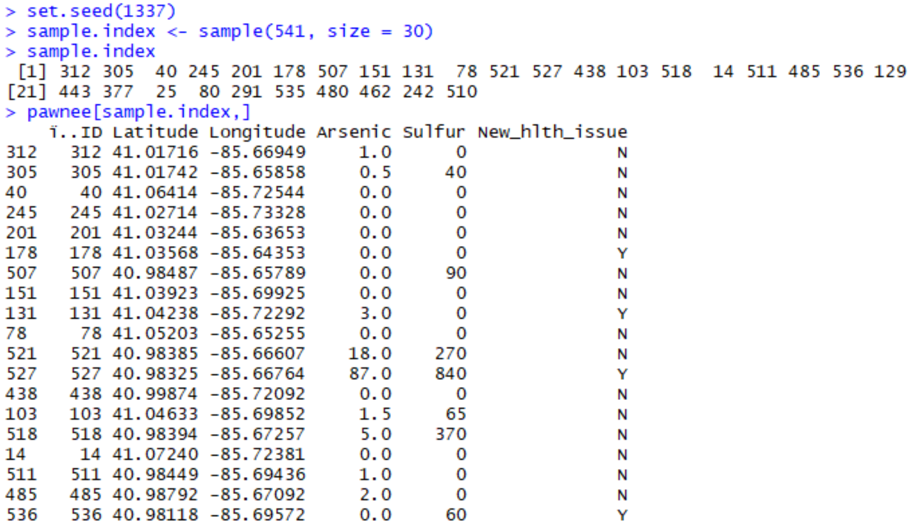
1. Exercise 1

a.



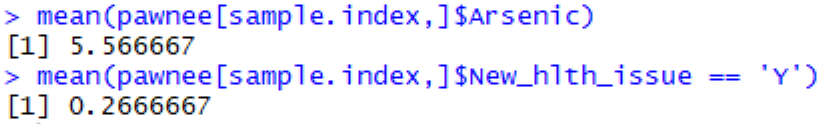
```
> head(pawnee)
  ï..ID Latitude Longitude Arsenic Sulfur New_hlth_issue
1     1 41.09414 -85.60974       0      0              N
2     2 41.09054 -85.70344       0    130              N
3     3 41.08601 -85.71996       4    170              N
4     4 41.08100 -85.75415       0      0              Y
5     5 41.07435 -85.70043       0      0              N
6     6 41.07399 -85.71788       0      0              N
> dim(pawnee)
[1] 541   6
```

b.



```
> set.seed(1337)
> sample.index <- sample(541, size = 30)
> sample.index
 [1] 312 305  40 245 201 178 507 151 131  78 521 527 438 103 518  14 511 485 536 129
[21] 443 377  25  80 291 535 480 462 242 510
> pawnee[sample.index,]
    ï..ID Latitude Longitude Arsenic Sulfur New_hlth_issue
312   312 41.01716 -85.66949     1.0      0              N
305   305 41.01742 -85.65858     0.5     40              N
40     40 41.06414 -85.72544     0.0      0              N
245   245 41.02714 -85.73328     0.0      0              N
201   201 41.03244 -85.63653     0.0      0              N
178   178 41.03568 -85.64353     0.0      0              Y
507   507 40.98487 -85.65789     0.0     90              N
151   151 41.03923 -85.69925     0.0      0              N
131   131 41.04238 -85.72292     3.0      0              Y
78     78 41.05203 -85.65255     0.0      0              N
521   521 40.98385 -85.66607    18.0    270              N
527   527 40.98325 -85.66764    87.0    840              Y
438   438 40.99874 -85.72092     0.0      0              N
103   103 41.04633 -85.69852     1.5     65              N
518   518 40.98394 -85.67257     5.0    370              N
14     14 41.07240 -85.72381     0.0      0              N
511   511 40.98449 -85.69436     1.0      0              N
485   485 40.98792 -85.67092     2.0      0              N
536   536 40.98118 -85.69572     0.0     60              Y
```

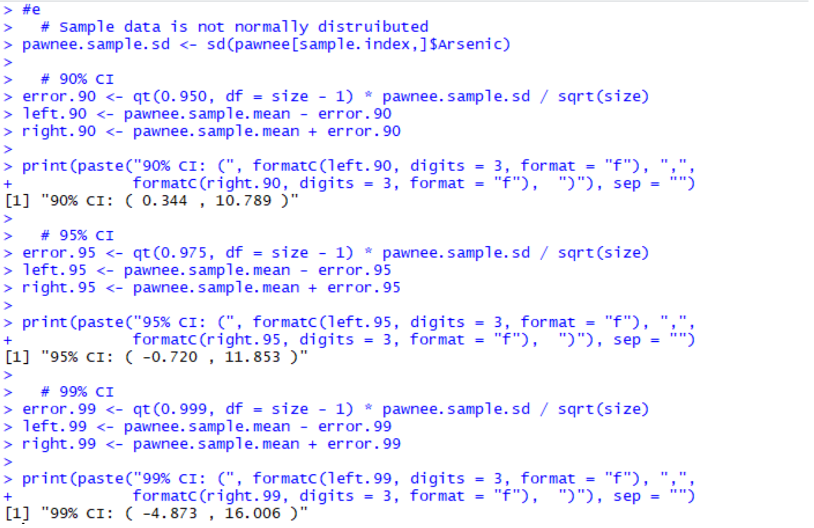
c.



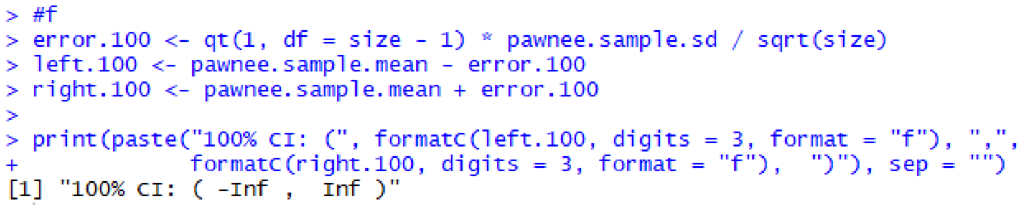
```
> mean(pawnee[sample.index,]$Arsenic)
[1] 5.566667
> mean(pawnee[sample.index,]$New_hlth_issue == 'Y')
[1] 0.2666667
```

d. The symbol for sample mean is $\bar{x}$. The symbol for sample proportion is $\hat{p}$.

e.



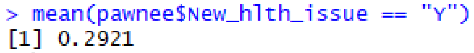
```
> #e
>   # Sample data is not normally distruibuted
> pawnee.sample.sd <- sd(pawnee[sample.index,]$Arsenic)
> 
>   # 90% CI
> error.90 <- qt(0.950, df = size - 1) * pawnee.sample.sd / sqrt(size)
> left.90 <- pawnee.sample.mean - error.90
> right.90 <- pawnee.sample.mean + error.90
> 
> print(paste("90% CI: (", formatC(left.90, digits = 3, format = "f"), ",",
+             formatC(right.90, digits = 3, format = "f"),  ")"), sep = "")
[1] "90% CI: ( 0.344 , 10.789 )"
> 
>   # 95% CI
> error.95 <- qt(0.975, df = size - 1) * pawnee.sample.sd / sqrt(size)
> left.95 <- pawnee.sample.mean - error.95
> right.95 <- pawnee.sample.mean + error.95
> 
> print(paste("95% CI: (", formatC(left.95, digits = 3, format = "f"), ",",
+             formatC(right.95, digits = 3, format = "f"),  ")"), sep = "")
[1] "95% CI: ( -0.720 , 11.853 )"
> 
>   # 99% CI
> error.99 <- qt(0.999, df = size - 1) * pawnee.sample.sd / sqrt(size)
> left.99 <- pawnee.sample.mean - error.99
> right.99 <- pawnee.sample.mean + error.99
> 
> print(paste("99% CI: (", formatC(left.99, digits = 3, format = "f"), ",",
+             formatC(right.99, digits = 3, format = "f"),  ")"), sep = "")
[1] "99% CI: ( -4.873 , 16.006 )"
```



```
> #f
> error.100 <- qt(1, df = size - 1) * pawnee.sample.sd / sqrt(size)
> left.100 <- pawnee.sample.mean - error.100
> right.100 <- pawnee.sample.mean + error.100
>
> print(paste("100% CI: (", formatC(left.100, digits = 3, format = "f"), ",",
+             formatC(right.100, digits = 3, format = "f"),  ")"), sep = "")
[1] "100% CI: ( -Inf ,  Inf )"
```

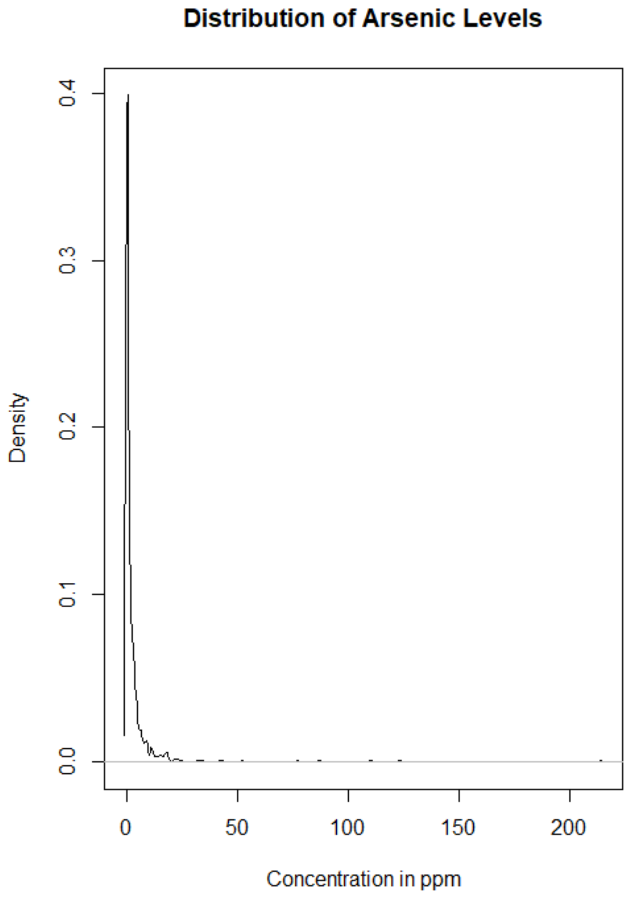
f.

The 100% CI shows as undefined, bit it incorporates the entire values in the set. This indicates that there is a 100% confidence that the population mean is contained within the set range of values.



```
> mean(pawnee$New_hlth_issue == "Y")
[1] 0.2921
```

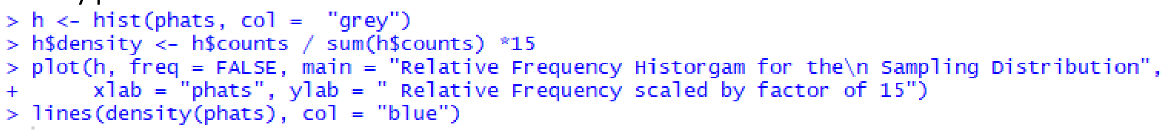
g.



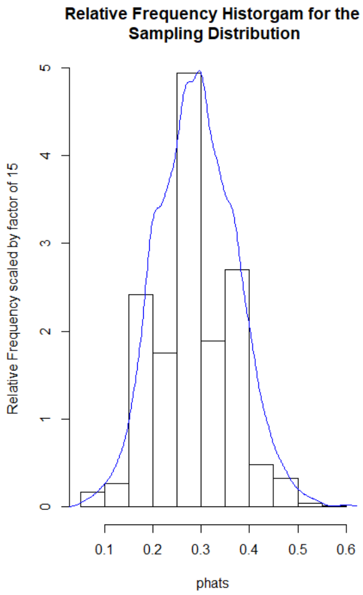
h.

2. Exercise 2

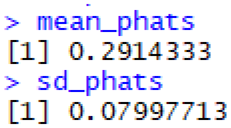
   a. Density plot



```
> h <- hist(phats, col =  "grey")
> h$density <- h$counts / sum(h$counts) *15
> plot(h, freq = FALSE, main = "Relative Frequency Historgam for the\n Sampling Distribution",
+      xlab = "phats", ylab = " Relative Frequency scaled by factor of 15")
> lines(density(phats), col = "blue")
```



b.

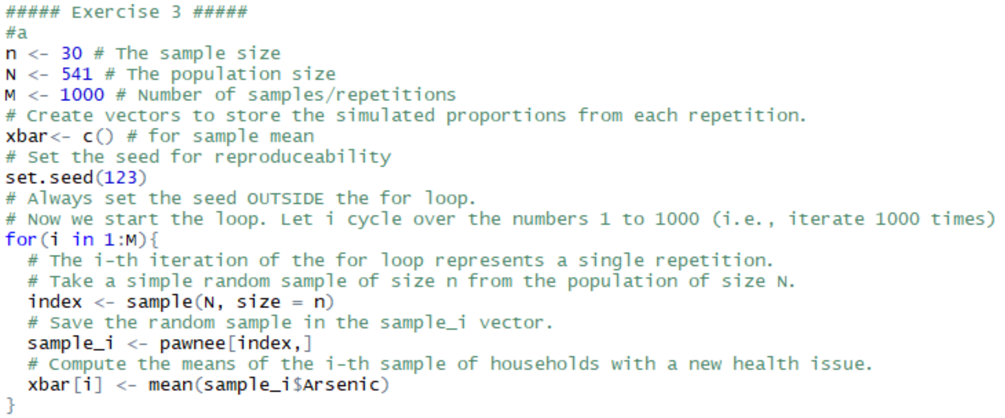


```
> mean_phats
[1] 0.2914333
> sd_phats
[1] 0.07997713
```

c. I think the simulated distributions of the sample proportions are approximately normal. This is because the histogram of the distributions of sample proportions looks normally distributed and normal with no apparent skew. The mean of the histogram is also centered around the mean of the calculated phats (approx.. 0.291 close to 0.3). The data values also seem to have little spread.

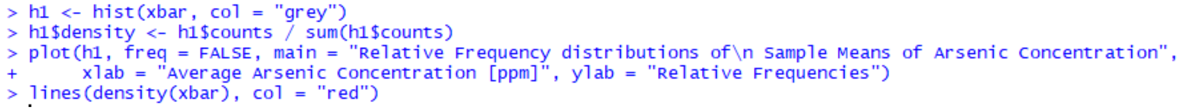
d. Using the Central Limit Theorem, since the sampling distribution is a normal distribution with a small range and no apparent outliers or skew, I would expect tat the mean and standard deviation of the sampling distribution to reflect that of the population. The mean of all the sample proportions (p-hat) would be the population proportion (p). From Exercise 1 part g, the population mean is 0.2921, which is very close to the sample proportions mean of 0.291433 (The difference is small, 0.0006667) As the number of samples increases the spread of the distribution decreases. The formula to get the standard deviation and solving for the sample standard deviation is $\sigma_{\hat{p}} = \sqrt{\frac{p(1-p)}{n}} = \sqrt{\frac{0.2921(1-0.2921)}{30}} = 0.0830216$. This value is close to the value in part b and off by 0.00304447).

3. Exercise 3



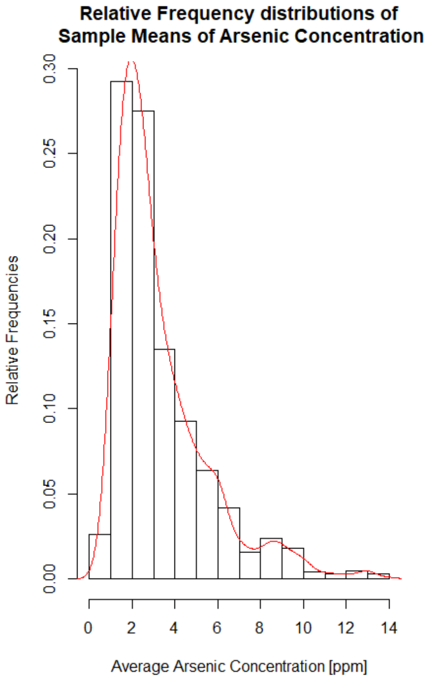
```
##### Exercise 3 #####
#a
n <- 30 # The sample size
N <- 541 # The population size
M <- 1000 # Number of samples/repetitions
# Create vectors to store the simulated proportions from each repetition.
xbar<- c() # for sample mean
# Set the seed for reproduceability
set.seed(123)
# Always set the seed OUTSIDE the for loop.
# Now we start the loop. Let i cycle over the numbers 1 to 1000 (i.e., iterate 1000 times)
for(i in 1:M){
  # The i-th iteration of the for loop represents a single repetition.
  # Take a simple random sample of size n from the population of size N.
  index <- sample(N, size = n)
  # Save the random sample in the sample_i vector.
  sample_i <- pawnee[index,]
  # Compute the means of the i-th sample of households with a new health issue.
  xbar[i] <- mean(sample_i$Arsenic)
}
```

a.



```
> h1 <- hist(xbar, col = "grey")
> h1$density <- h1$counts / sum(h1$counts)
> plot(h1, freq = FALSE, main = "Relative Frequency distributions of\n Sample Means of Arsenic Concentration",
+      xlab = "Average Arsenic Concentration [ppm]", ylab = "Relative Frequencies")
> lines(density(xbar), col = "red")
```

b.



c. The simulated distributions of sample arsenic means is not approximately normal because the histogram shows that the data is in fact slewed to the right. My answer is different than 2c because the majority of the arsenic concentrations in the area have 0 ppm concentration and thus is the cause of the skew when taking many samples. Additionally we are looking at the mean values of Arsenic concentration which is a different parameter and distribution than looking at the sample proportions in 2c.