Lab 6

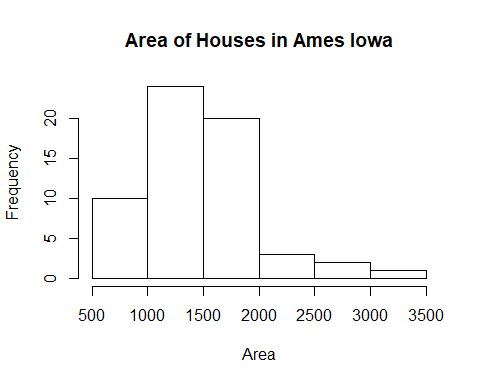
download.file("http://www.openintro.org/stat/data/ames.RData", destfile ="ames.RData")  
load("ames.RData")  
  
population <- ames$Gr.Liv.Area  
samp <- sample(population, 60)

## Exercise 1: Plot a histogram of your sample of living areas. Then, describe the shape, center, and spread of your histogram. What would you say is the “typical” living area within your sample? Explain.

summary(samp)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 759 1106 1386 1470 1744 3194

hist(samp, main = "Area of Houses in Ames Iowa", xlab = "Area")



mean(samp) #The mean of the area

## [1] 1469.55

sd(samp) #The standard deviation of the area

## [1] 505.5726

### Exercise 1 Answer

## Exercise 2: Would you expect another student’s sample distribution to be identical to yours? Would you expect it to be similar? Why or why not?

### Exercise 2 Answer: No. Another student can recieve different set of 60 from the poplation.

## Confidence Intervals

sample\_mean <- mean(samp)  
se <- sd(samp)/sqrt(60)  
lower <- sample\_mean - 2 \* se  
upper <- sample\_mean + 2 \* se  
c(lower, upper)

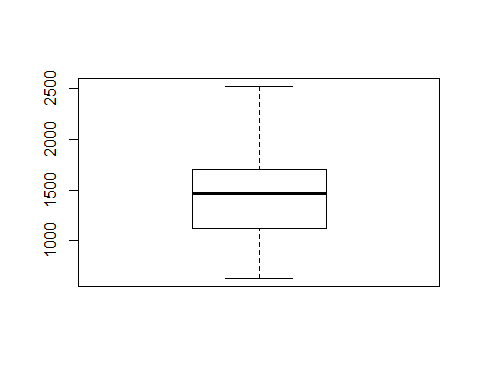
## [1] 1374.818 1611.582

## Exercise 3: For a one-sample t confidence interval to be valid, the sampling distribution of the sample mean must be normally distributed. Check this assumption using the indirect methods demonstrated during class. (Note: If any outliers are present in your sample, you will need to include the relevant calculations to classify the outlier(s) as being either mild or extreme. Extreme outliers prevent us from applying the Central Limit Theorem.)

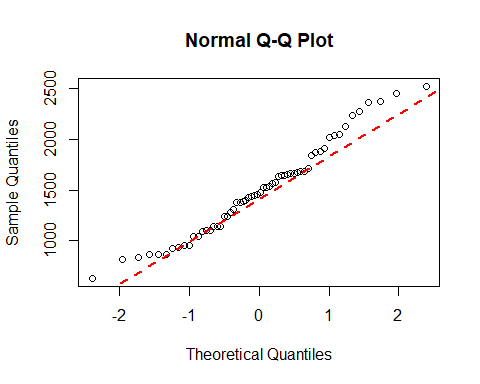
shapiro.test(samp) #Shapiro Wilk Test of samp

##   
## Shapiro-Wilk normality test  
##   
## data: samp  
## W = 0.97176, p-value = 0.1779

boxplot(samp) #Boxplot of samp



#qq Norm of Samp  
qqnorm(samp)   
qqline(samp, col=2,lwd=2,lty=2)



quantile(samp)

## 0% 25% 50% 75% 100%   
## 630.00 1127.50 1466.50 1692.75 2520.00

IQR(samp)

## [1] 565.25

quantile(samp,.25) - ((1.5)\*(IQR(samp))) #inner fence for low outlier

## 25%   
## 279.625

quantile(samp,.25) - ((3)\*(IQR(samp))) #Outer fence for low outlier

## 25%   
## -568.25

quantile(samp,.75) + ((1.5)\*(IQR(samp))) #Outer fence for low outlier

## 75%   
## 2540.625

quantile(samp,.75) + ((3)\*(IQR(samp))) #Outer fence for low outlier

## 75%   
## 3388.5

### Exercise 3 Answer: No. Another student can recieve different set of 60 from the poplation.

## Exercise 4: Report your 95% confidence interval in the form . Then, carefully interpret your confidence interval in context

### Exercise 4 Answer: Something

## Exercise 5: What does the phrase “95% confident” mean? In other words, give an interpretation of the confidence level.

### Exercise 5 Answer: Something

## Exercise 6: Did your confidence interval capture the true mean living area of houses in Ames? Explain.

mean(population) #The mean of the population

## [1] 1499.69

## Exercise 7: Each student in your class section should have gotten a slightly different confidence interval. What proportion of those intervals would you expect to successfully capture the true population mean? Why? Write your confidence interval on the board. When everybody has done so, write down the confidence intervals created by all of the students in your class section and calculate the proportion of these intervals that successfully captured the true population mean. How does this proportion compare to the expected proportion? Why might it be different? Explain.

source('Lab6\_class\_results.R') #Constructed file with values from class  
  
#will print in range if population mean is between upper and lower values. Will print not in range if the population mean is not between the upper and lower values  
for(i in 1:length(class\_upper)){  
 cat("\nStudent #",i)  
 if((class\_lower[i] <= mean(population)) & (mean(population >= class\_upper[i]))){  
   
 low <- cat(" - lower:", class\_lower[i])  
 high <- cat(" upper:", class\_upper[i])  
 cat(" in range")  
 }else  
 {  
 low <- cat(" - lowe:", class\_lower[i])  
 high <- cat(" upper:", class\_upper[i])  
 cat(" not in range")  
 }  
}

##   
## Student # 1 - lower: 1399.875 upper: 1627.891 in range  
## Student # 2 - lower: 1337.333 upper: 1581.5 in range  
## Student # 3 - lower: 1391.425 upper: 1623.608 in range  
## Student # 4 - lower: 1377.671 upper: 1633.396 in range  
## Student # 5 - lower: 1392.96 upper: 1623.074 in range  
## Student # 6 - lower: 1443.172 upper: 1719.528 in range  
## Student # 7 - lower: 1418.528 upper: 1660.805 in range  
## Student # 8 - lower: 1454.157 upper: 1711.476 in range  
## Student # 9 - lower: 1328.639 upper: 1595.661 in range  
## Student # 10 - lower: 1328.063 upper: 1534.971 in range  
## Student # 11 - lower: 1332.427 upper: 1599.206 in range  
## Student # 12 - lower: 1428.577 upper: 1750.156 in range  
## Student # 13 - lower: 1440.732 upper: 1650.402 in range  
## Student # 14 - lower: 1424.698 upper: 1723.235 in range  
## Student # 15 - lower: 1378.667 upper: 1652.433 in range  
## Student # 16 - lower: 1383.977 upper: 1650.223 in range  
## Student # 17 - lower: 1420.842 upper: 1691.791 in range  
## Student # 18 - lower: 1334.625 upper: 1591.275 in range  
## Student # 19 - lower: 1311.047 upper: 1569.12 in range  
## Student # 20 - lower: 1356.384 upper: 1604.416 in range  
## Student # 21 - lower: 1425.58 upper: 1704.22 in range  
## Student # 22 - lower: 1441.587 upper: 1719.713 in range  
## Student # 23 - lower: 1416.248 upper: 1704.786 in range  
## Student # 24 - lower: 1421.833 upper: 1653.467 in range  
## Student # 25 - lower: 1357.236 upper: 1606.831 in range  
## Student # 26 - lower: 1394.11 upper: 1594.257 in range  
## Student # 27 - lower: 1363.423 upper: 1631.577 in range

### Exercise 7 Answer: Something

## Homework preparation

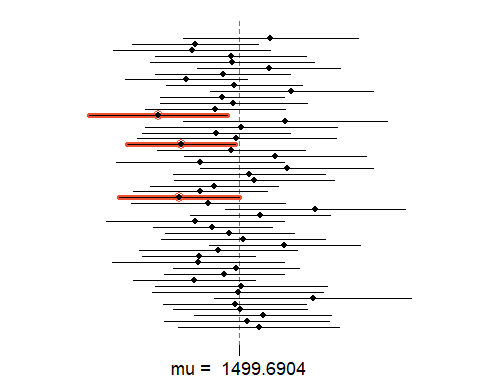
samp\_mean <- rep(NA, 50)  
samp\_sd <- rep(NA, 50)  
n <- 60  
for(i in 1:50){  
samp <- sample(population, n) # obtain a sample of size n = 60 from the population  
samp\_mean[i] <- mean(samp) # save sample mean in ith element of samp\_mean  
samp\_sd[i] <- sd(samp) # save sample sd in ith element of samp\_sd  
}  
lower <- samp\_mean - 2 \* samp\_sd/sqrt(n)  
upper <- samp\_mean + 2 \* samp\_sd/sqrt(n)  
c(lower[1], upper[1]) #Upper and lower bounds for the first interval

## [1] 1398.255 1664.178

# Homework

## 1. Using the following function (which was downloaded with the data set), plot all fifty of your 95% What proportion of your confidence intervals include the true population mean? Is this proportion exactly equal to the confidence level? Why might it differ?

plot\_ci(lower, upper, mean(population))



### Homework 1 Answer

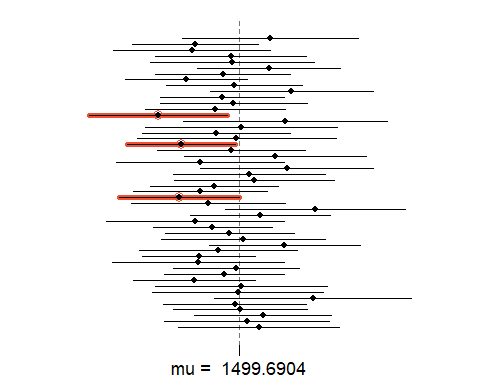
## 2. What is the appropriate critical t value for a 98% confidence level with 59 df? Include R calculations for finding this critical t. (It could be helpful to also find the critical t using the invT command on your graphing calculator. Confirm that you get the same result using both methods to ensure that you used the correct R command.)

qt(.975, 59) #98% confidence interval

## [1] 2.000995

## 3. Construct fifty 98% confidence intervals. You do not need to obtain new samples; simply calculate new intervals based on the sample means and standard deviations you have already collected; you only need to change the critical t used in the calculations (it was 2 for a 95% confidence level and 59 df). Using the plot\_ci function, plot all fifty intervals and calculate the proportion of intervals that include the true population mean. How does this percentage compare to the confidence level?

conf\_lower<- samp\_mean - qt(.975,59) \* samp\_sd/sqrt(n)  
conf\_upper<- samp\_mean + qt(.975,59) \* samp\_sd/sqrt(n)  
plot\_ci(conf\_lower, conf\_upper, mean(population))



### Homework 3 Answer1