Foundations for statistical inference - Confidence intervals

Sampling from Ames, Iowa

If you have access to data on an entire population, say the size of every house in Ames, Iowa, it's straight forward to answer questions like, "How big is the typical house in Ames?" and "How much variation is there in sizes of houses?". If you have access to only a sample of the population, as is often the case, the task becomes more complicated. What is your best guess for the typical size if you only know the sizes of several dozen houses? This sort of situation requires that you use your sample to make inference on what your population looks like.

The data

In the previous lab, ``Sampling Distributions'', we looked at the population data of houses from Ames, Iowa. Let's start by loading that data set.

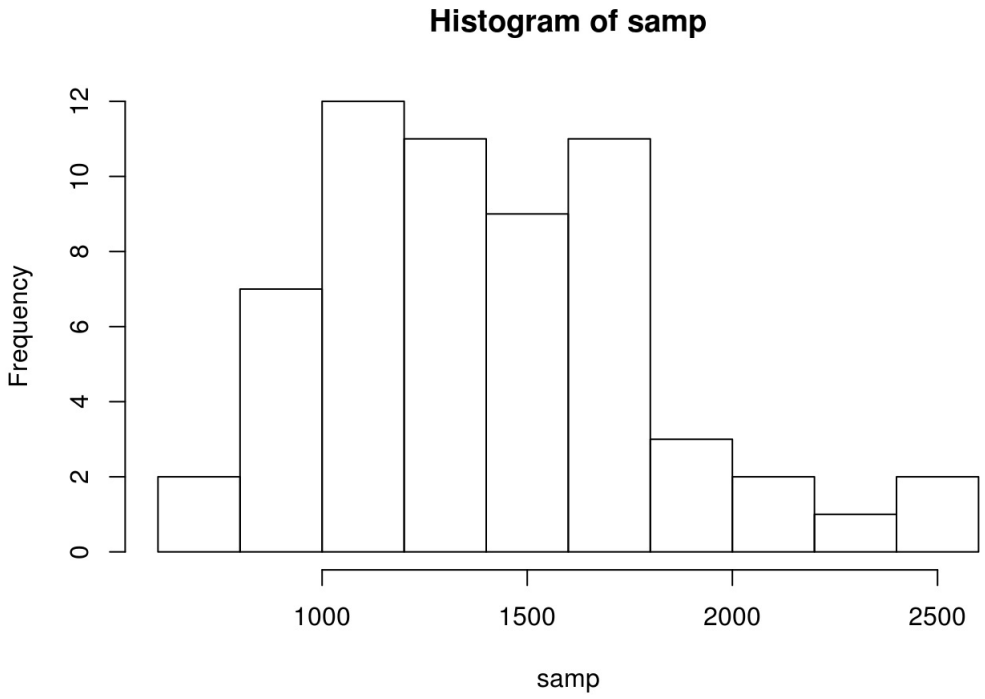
```
load("more/ames.RData")
```

In this lab we'll start with a simple random sample of size 60 from the population. Specifically, this is a simple random sample of size 60. Note that the data set has information on many housing variables, but for the first portion of the lab we'll focus on the size of the house, represented by the variable `Gr.Liv.Area`.

```
population <- ames$Gr.Liv.Area
samp <- sample(population, 60)
```

Exercise 1 Describe the distribution of your sample. What would you say is the "typical" size within your sample? Also state precisely what you interpreted "typical" to mean.

```
hist(samp)
```



Answer: The typical size is the sample size which is 60 in this case. The distribution of the sampe is bi-modal.

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

Answer: Someone else running the code would have a different distribution than myself but would be similar.

The reason is that each time you sample from the population, you get different data from the population.

Confidence intervals

One of the most common ways to describe the typical or central value of a distribution is to use the mean. In this case we can calculate the mean of the sample using,

```
sample_mean <- mean(samp)
```

Return for a moment to the question that first motivated this lab: based on this sample, what can we infer about the population? Based only on this single sample, the best estimate of the average living area of houses sold in Ames would be the sample mean, usually denoted as $\bar{x}$ (here we're calling it `sample_mean`). That serves as a good *point estimate* but it would be useful to also communicate how uncertain we are of that estimate. This can be captured by using a *confidence interval*.

We can calculate a 95% confidence interval for a sample mean by adding and subtracting 1.96 standard errors to the point estimate (See Section 4.2.3 if you are unfamiliar with this formula).

```
se <- sd(samp) / sqrt(60)
lower <- sample_mean - 1.96 * se
upper <- sample_mean + 1.96 * se
c(lower, upper)
```

```
## [1] 1306.247 1518.419
```

This is an important inference that we've just made: even though we don't know what the full population looks like, we're 95% confident that the true average size of houses in Ames lies between the values *lower* and *upper*. There are a few conditions that must be met for this interval to be valid.

Exercise 3 For the confidence interval to be valid, the sample mean must be normally distributed and have standard error $s/\sqrt{n}$. What conditions must be met for this to be true?

Answer: the sample size must be > or equal to 30, the sample observations must be independent and also

the data of the population is nearly normal or not too strongly skewed.

Confidence levels

Exercise 4 What does "95% confidence" mean? If you're not sure, see Section 4.2.2.

Answer: 95% confidence means we are 95% confident that the true/population mean resides within the lower and upper ranges.

In this case we have the luxury of knowing the true population mean since we have data on the entire population. This value can be calculated using the following command:

```
mean(population)
```

```
## [1] 1499.69
```

Exercise 5 Does your confidence interval capture the true average size of houses in Ames? If you are working on this lab in a classroom, does your neighbor's interval capture this value?

Answer: yes my 95% confidence interval captures the population mean of sizes of houses.

No my classmate's confidence interval may not capture the true population due to several things:

1. the sample size is different
2. the sample mean has a different value
3. the sample standard deviation is different
4. the percent used could be different.

Exercise 6 Each student in your class should have gotten a slightly different confidence interval. What proportion of those intervals would you expect to capture the true population mean? Why? If you are working in this lab in a classroom, collect data on the intervals created by other students in the class and calculate the proportion of intervals that capture the true population mean.

Using R, we're going to recreate many samples to learn more about how sample means and confidence intervals vary from one sample to another. *Loops* come in handy here (If you are unfamiliar with loops, review the Sampling Distribution Lab (http://htmlpreview.github.io/?https://github.com/andrewpbray/oiLabs/blob/master/sampling_distributions/sampling_distributions.html)).

Here is the rough outline:

1. Obtain a random sample.
2. Calculate and store the sample's mean and standard deviation.
3. Repeat steps (1) and (2) 50 times.
4. Use these stored statistics to calculate many confidence intervals.

But before we do all of this, we need to first create empty vectors where we can save the means and standard deviations that will be calculated from each sample. And while we're at it, let's also store the desired sample size as `n`.

```
samp_mean <- rep(NA, 50)
samp_sd <- rep(NA, 50)
n <- 60
```

Now we're ready for the loop where we calculate the means and standard deviations of 50 random samples.

```
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
}
```

Lastly, we construct the confidence intervals.

```
lower_vector <- samp_mean - 1.96 * samp_sd / sqrt(n)
upper_vector <- samp_mean + 1.96 * samp_sd / sqrt(n)
```

Lower bounds of these 50 confidence intervals are stored in `lower_vector`, and the upper bounds are in `upper_vector`. Let's view the first interval.

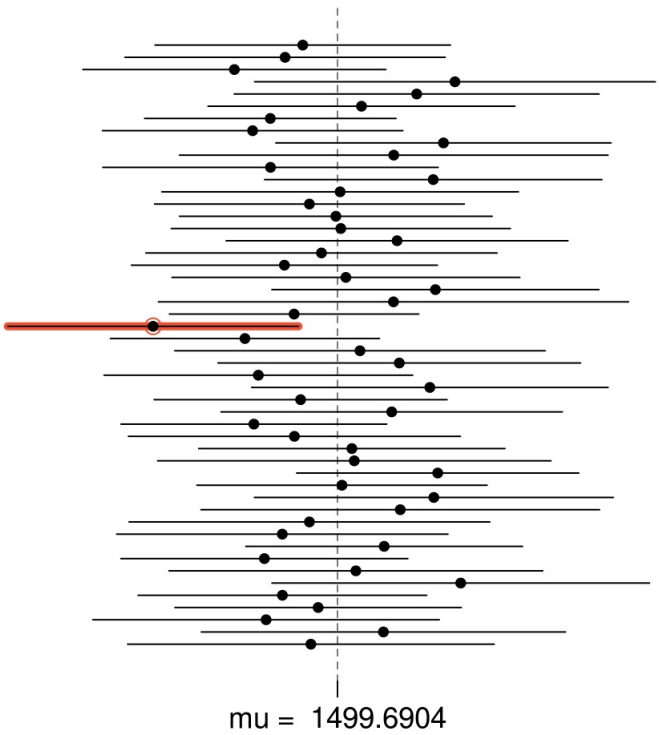
```
c(lower_vector[1], upper_vector[1])
```

```
## [1] 1341.125 1618.075
```

On your own

1. Using the following function (which was downloaded with the data set), plot all intervals. What proportion of your confidence intervals include the true population mean? Is this proportion exactly equal to the confidence level? If not, explain why.

   ```
   plot_ci(lower_vector, upper_vector, mean(population))
   ```



Answer: the plots only have usually 46, 47, 48 or 49 out of 50 of intervals containing the possible true population mean.

When I run the confidence intervals, I don’t always get

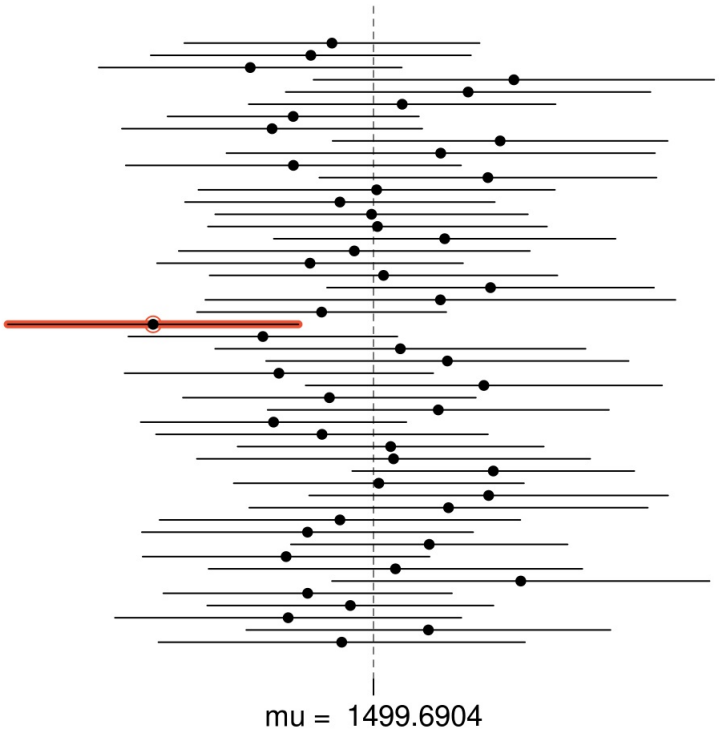
1. Pick a confidence level of your choosing, provided it is not 95%. What is the appropriate critical value?

Answer: I will pick 90% confidence level. A confidence interval of 90% is z* = 1.64

1. Calculate 50 confidence intervals at the confidence level you chose in the previous question. You do not need to obtain new samples, simply calculate new intervals based on the sample means and standard deviations you have already collected. Using the `plot_ci` function, plot all intervals and calculate the proportion of intervals that include the true population mean. How does this percentage compare to the confidence level selected for the intervals?

Answer:

```
lower_vector <- samp_mean - 1.64 * samp_sd / sqrt(n)
upper_vector <- samp_mean + 1.64 * samp_sd / sqrt(n)
plot_ci(lower_vector, upper_vector, mean(population))
```




```
data.frame(lower_vector,upper_vector)
```

```
##    lower_vector upper_vector
## 1      1363.733     1595.467
## 2      1419.263     1649.504
## 3      1336.257     1555.243
## 4      1394.582     1575.551
## 5      1366.846     1549.287
## 6      1473.473     1712.127
## 7      1395.451     1631.716
## 8      1353.769     1535.131
## 9      1447.462     1622.404
## 10     1353.287     1562.680
## 11     1364.448     1592.519
## 12     1421.159     1673.074
## 13     1459.052     1685.914
## 14     1411.351     1594.782
## 15     1486.265     1664.568
## 16     1388.049     1636.684
## 17     1413.783     1607.317
## 18     1362.240     1572.027
## 19     1352.535     1520.465
## 20     1432.777     1648.523
## 21     1379.232     1564.402
## 22     1456.804     1682.129
## 23     1342.350     1537.450
## 24     1431.795     1660.972
## 25     1399.610     1633.756
## 26     1344.726     1514.807
## 27     1268.466     1452.268
## 28     1388.112     1545.688
## 29     1393.388     1690.579
## 30     1470.305     1677.029
## 31     1396.101     1615.966
## 32     1362.862     1556.272
## 33     1376.595     1598.539
## 34     1436.678     1652.722
## 35     1395.039     1609.361
## 36     1399.658     1597.275
## 37     1380.566     1576.367
## 38     1388.990     1614.377
## 39     1465.430     1678.603
## 40     1343.099     1555.034
## 41     1406.746     1677.621
## 42     1473.837     1685.663
## 43     1340.663     1530.504
## 44     1369.492     1528.375
## 45     1420.839     1614.795
## 46     1444.228     1674.805
## 47     1461.791     1715.076
## 48     1326.013     1517.554
## 49     1358.878     1561.289
## 50     1380.091     1566.775
```

Ran the Rmd file several times got 40/50, 42/50, 46/50.

The 90% confidence interval has even less intervals containing the true population mean.

The 95% confidence interval contained more intervals and was more narrower.