Lab 4B: Foundations for Statistical Inference – Confidence Levels

Sampling from Ames, Iowa

If you have access to data on an entire population (for example, the size of every house in Ames, Iowa) it's straightforward to answer questions such as "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.[1]

The Data

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

```
filename amesh url 'http://www.openintro.org/stat/data/ames_sas.csv';

proc import datafile=amesh out=work.ames dbms=csv replace;
   getnames=yes;
   guessingrows=max;
run;
```

In this lab, we'll start with a simple random sample of size 60 from the population. Note that the data set has information about 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**. PROC SURVEYSELECT enables us to take this simple random sample. Specifying specific options in the PROC SURVEYSELECT statement enables us to obtain the type of sample that we want. To get a simple random sample of size 60, we include METHOD=SRS and SAMPSIZE=60. RANUNI requests uniform random number generation. Setting a seed value allows for replication of sampling if we want to duplicate the exact same sample. For this lab, we will not set a seed value. With the sample selected, a quick DATA step with a KEEP statement reduces the number of variables in the data set to just the one that we want, **Gr_Liv_Area**.

```
proc surveyselect data=work.ames out=work.amessample sampsize=60
                  method=srs ranuni;
run;

data work.amessample;
   set work.amessample;
   keep Gr_Liv_Area;
run;
```

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.

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?

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 PROC MEANS.

```
proc means data=work.amessample mean;
   var Gr_Liv_Area;
run;
```

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 $\bar{x}$.

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 including the CLM option in the PROC MEANS statement. ALPHA=0.05, by default, creates the 95% interval, but it can be changed.

```
proc means data=work.amessample mean clm alpha=0.05;
   var Gr_Liv_Area;
run;
```

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 lower and upper 95% confidence limits of the mean. 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?

Confidence Levels

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

In this case, we have the luxury of knowing the true population mean because we have data on the entire population. This value can be calculated using the full data set within PROC MEANS. For later usage, let's make a macro variable containing this value using PROC SQL. In the SQL code, the average of **Gr_Liv_Area** will be calculated from the **work.ames** data set. The answer will be stored in the macro variable named **popmean**. This macro variable enables us to refer directly to the mean of the variable **Gr_Liv_Area** without having to actually know the value. We can access this value now using **&popmean**.

```
proc means data=work.ames mean;
   var Gr_Liv_Area;
run;

proc sql;
   select AVG(Gr_Liv_Area) into :popmean FROM work.ames;
run;
```

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?

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 SAS, we're going to re-create many samples to learn more about how sample means and confidence intervals vary from one sample to another. Let's take 50 samples of size 60 from the population and calculate the mean and confidence limits for the mean for each sample. For each sample, name the mean **s_mean** and the confidence limits **s_lower** and **s_upper** for the lower and upper limits respectfully.

```
proc surveyselect data=work.ames out=work.amessampler sampsize=60
                  method=srs reps=50 ranuni;
run;

proc means data=work.amessampler mean clm alpha=0.05 noprint;
   by replicate;
   var Gr_Liv_Area;
   output out=work.reprun mean=s_mean lclm=s_lower uclm=s_upper;
run;
```

Let's view the results.

```
proc print data=work.reprun;
   var s_mean s_lower s_upper;
run;
```

On Your Own

1. Using the following code, determine whether the true population mean was captured by the intervals. Within the DATA step, the Boolean expression will flag **captured**=1 when the true population mean is within the confidence limits and 0 otherwise. To determine what proportion of your confidence intervals includes the true population mean, we compute the average of this binary variable with PROC MEANS. Is this proportion exactly equal to the confidence level? If not, explain why.

```
data work.reprun;
   set work.reprun;
   captured = (s_lower le &popmean le s_upper);
run;

proc means data=work.reprun mean;
   var captured;
run;
```

2. Pick a confidence level of your choosing, provided it is not 95%.

3. Calculate 50 confidence intervals at the confidence level you chose in the previous question. Adjust the code to accommodate the confidence level you selected. Calculate the proportion of intervals that include the true population mean. How does this percentage compare to the confidence level selected for the intervals?

4. What concepts from the textbook are covered in this lab? What concepts, if any, are not covered in the textbook? Have you seen these concepts elsewhere (for example, lecture, discussion section, previous labs, or homework problems)? Be specific in your answer.