Spark Project

Goals:

1. Learn to use Scala
2. Learn to apply Spark API
3. Practice data analysis on large data sets
4. Refresh common terms in statistics
5. Have a firsthand experience on the bootstrapping technique in statistics

(The following is from

https://www.thoughtco.com/what-is-bootstrapping-in-statistics-3126172

and is included here for the convenience. Project description starts at page 3)

Bootstrapping is a statistical technique that falls under the broader heading of resampling. This technique involves a relatively simple procedure but repeated so many times that it is heavily dependent upon computer calculations. Bootstrapping provides a method other than confidence intervals to estimate a population parameter. Bootstrapping very much seems to work like magic. Read on to see how it obtains its interesting name.

An Explanation of Bootstrapping

One goal of inferential statistics is to determine the value of a parameter of a population. It is typically too expensive or even impossible to measure this directly. So we use statistical sampling. We sample a population, measure a statistic of this sample, and then use this statistic to say something about the corresponding parameter of the population.

For example, in a chocolate factory, we might want to guarantee that candy bars have a particular mean weight. It's not feasible to weigh every candy bar that is produced, so we use sampling techniques to randomly choose 100 candy bars. We calculate the mean of these 100 candy bars and say that the population mean falls within a margin of error from what the mean of our sample is.

Suppose that a few months later we want to know with greater accuracy -- or less of a margin of error -- what the mean candy bar weight was on the day that we sampled the production line.

We cannot use today's candy bars, as too many variables have entered the picture (different batches of milk, sugar and cocoa beans, different atmospheric conditions,

different employees on the line, etc.). All that we have from the day that we are curious about are the 100 weights. Without a time machine back to that day, it would seem that the initial margin of error is the best that we can hope for.

Fortunately, we can use the technique of bootstrapping. In this situation, we randomly sample with replacement from the 100 known weights. We then call this a bootstrap sample. Since we allow for replacement, this bootstrap sample most likely not identical to our initial sample. Some data points may be duplicated, and others data points from the initial 100 may be omitted in a bootstrap sample. With the help of a computer, thousands of bootstrap samples can be constructed in a relatively short time.

An Example

As mentioned, to truly use bootstrap techniques we need to use a computer. The following numerical example will help to demonstrate how the process works. If we begin with the sample 2, 4, 5, 6, 6, then all of the following are possible bootstrap samples:

- 2 ,5, 5, 6, 6
- 4, 5, 6, 6, 6
- 2, 2, 4, 5, 5
- 2, 2, 2, 4, 6
- 2, 2, 2, 2, 2
- 4,6, 6, 6, 6

History of the Technique

Bootstrap techniques are relatively new to the field of statistics. The first use was published in a 1979 paper by Bradley Efron. As computing power has increased and becomes less expensive, bootstrap techniques have become more widespread.

Why the Name Bootstrapping?

The name “bootstrapping” comes from the phrase, “To lift himself up by his bootstraps.” This refers to something that is preposterous and impossible.

Try as hard as you can, you cannot lift yourself into the air by tugging at pieces of leather on your boots.

There is some mathematical theory that justifies bootstrapping techniques. However, the use of bootstrapping does feel like you are doing the impossible. Although it does not seem like you would be able to improve upon the estimate of a population statistic by reusing the same sample over and over again, bootstrapping can, in fact, do this.

Project steps

Step 1. Download a dataset (.csv file) from the site
https://vincentarelbundock.github.io/Rdatasets/datasets.html

For example **mtcars.csv**

Step 2. Select a categorical variable and a numeric variable and form the key-value pair and create a pairRDD called "**population**".

For example cyl (short for cylinder) can be the categorical variable since there are only three distinct values: 4, 6 and 8. The mpg (miles per gallon) can be the numerical variable. So in this step you will create a pairRDD with key cyl and value mpg.

Step 3. Compute the mean mpg and variance for each category and display as shown below:

Category	Mean	Variance
4	17.29	1.27
6	19.58	1.75
8	22.44	2.30

Step 4. Create the **sample** for bootstrapping. All you need to do is take 25% of the **population without replacement.**

Step 5. Do 1000 times

5a. Create a "resampledData". All you need to do is take 100% of the **sample with replacement.**

5b. Compute the mean mpg and variance for each category (similar to Step 3).

5c. Keep adding the values in some running sum.

Step 6. Divide each quantity by 1000 to get the average and display the result.

Category	Mean	Variance
4	17.42	1.27
6	19.31	1.67
8	22.46	2.22

OPTIONAL

These are fun steps for those who want to explore and learn more.

Step 6 Determine the absolute error percentage for each of the values being estimated.

abs(actual – estimate)*100/actual.

In this example, there are six values. For each one of those six you need to compute

abs(actual – estimate)*100/actual. For instance, for mean of 4 cyl.

= abs(17.29-1742)*100/17.29 = 0.75.

Once you finish remaining five values, you have completed one row in the following DataFrame.

Percentage	4-cyl mean	4-cyl variance	6-cyl mean	4-cyl variance	8-cyl mean	4-cyl variance
25	0.75					

Step 7 Draw a graph with x-axis percentage. Looking at the graph determine for what percentage values, each of the six error values are closer to 0.