Resampling Methods

Introduction

Resampling techniques are modern statistical techniques that involve taking repeated subsamples from a sample. These procedures tend to be computationally intensive, since they involve computing statistics of a subsample, creating new subsamples and repeating the process thousands or perhaps millions of times. This can allow for additional analysis of the subsamples leading to increased confidence and knowledge of the larger population. The three main techniques we will discuss here are bootstrapping, jackknife, and permutation tests.

Objectives

You will be able to:

- Identify when resampling is used
- Describe the process of bootstrapping
- Describe permutation testing

Jackknife and Bootstrapping

Let's start by defining the sampling methodology for these techniques. The bootstrap method works by taking random samples with replacement from the original sample of size n. In contrast, the jackknife, the older of the two methods, works by taking samples by removing one, or more, observations at a time. Each one of these (n-1) sized sub-samples is aggregated to create the new jackknife sample. The purpose of these resampling methods is to be able to increase the size of our samples without having to actually go out and obtain more samples. Resampling methods attempt to estimate the variability of point estimators derived from the original samples.

The motivating principle behind both is that by analyzing the variance of parameter estimates from these synthetic samples, we can also gauge the variance of our point estimate for the population itself. For example, we might take an original sample from our population and then use the jackknife or bootstrapping method to generate additional synthetic samples. By calculating the point estimate of interest for these synthetic samples, we can better gauge the confidence interval and variability of our original point estimator.

Permutation Tests

Another related methodology is permutation tests. Permutation tests can be used in lieu of assumed parameter distributions for any statistical test. For example, we discussed the central limit theorem: that when taking the mean of a repeated sample from a population, the means of these samples will form a normal distribution. From this, we were then able to extrapolate confidence intervals surrounding our estimate for the mean of the entire population by assuming that our sample mean was from a normal distribution. This allowed us to define our confidence bands associated with various levels of type I errors which we set with α. In a hypothesis test, we used the same procedure to calculate the probability of a given sample, and based on alpha, rejected or confirmed the null hypothesis. In a permutation test, rather then assume the distribution itself and calculate p-values, we would calculate all permutations of our relabeling our data and compute the parameter statistic in question for these permutations.

For example, let's say we had two samples, one with 37 observations and the other with 45 observations. We calculate the mean of both samples and wish to perform a hypothesis test with a 5% confidence interval for whether the two samples belong to the same overall population. In our previous work, we would use a t-test to perform this comparison. The permutation test alternative would be to compare the difference in these sample means to the difference in sample means of all possible permutations of 37-45 splits between our 82 data points. In other words, we compare the difference between our actual sample means to the difference in sample means between all variations of all those 82 points in order to calculate our p-values and determine whether we accept or reject the null-hypothesis.

> **Note**: While it's called a permutation test, calculating all of the possible combinations of the observations into two groups is a more pragmatic approach. After all, you are comparing the sample means of the groups and as such the order of group members is irrelevant. When you implement permutation tests in the upcoming lab, you'll use combinations to make the problem computationally feasible. Even so, as you will see, the size of possible variations can quickly explode leading to other estimations of the permutation test, which you'll investigate towards the end of the section.

Additional Resources

- http://hydrodictyon.eeb.uconn.edu/eebedia/images/9/9d/FelsensteinChap20.pdf
- https://www.scss.tcd.ie/Rozenn.Dahyot/453Bootstrap/05_Permutation.pdf

Summary

In this lesson, we continued discussing non-parametric statistics and investigated resampling techniques. This included bootstrapping, jackknife, and permutation tests. In the upcoming lab, you'll define functions that implement these techniques and then use them to conduct statistical simulations and tests.